PRISM: When Agents Provably Learn from Pluralistic Human Feedback

Shuo Yang [1] **Zhen Chen** [1] **Sujay Sanghavi** [2]

Abstract

Aligning AI systems with diverse human values requires learning from feedback that may be *pluralistic and inconsistent*: different individuals or groups can rank the same options differently depending on the context. In this paper, we provide theoretical foundations for this challenge in the framework of combinatorial bandits with semi-bandit feedback, where an online learner selects a size-k set of items at each step and observes set-dependent, potentially inconsistent per-item rewards. We present a simple structural assumption – *pluralistic reward inconsistency with structural monotonicity* (PRISM) – that formalizes when learning remains tractable despite inconsistent preferences. PRISM allows for intransitive and contradictory feedback, yet subsumes many widely used preference models (e.g., multinomial logit and random utility models). Most importantly, we prove that under PRISM a simple UCB-based algorithm finds the optimal set and achieves $O\left(\min(\frac{k^3 n \log T}{\epsilon}, k^2\sqrt{nT\log T})\right)$ regret, nearly matching our $\Omega(\frac{n\log T}{\epsilon})$ lower bound. Our results demonstrate that provably efficient learning from pluralistic human feedback is possible under mild structural conditions, providing a theoretical basis for the practical success of simple algorithms in the presence of value pluralism.

1. Introduction

A central challenge in aligning AI systems with human values is that human preferences are *pluralistic*: different individuals, groups, or contexts can give rise to contradictory feedback about the same set of options (Sorensen et al., 2024; Conitzer et al., 2024). Recent work has highlighted that collapsing diverse feedback into a single consistent reward model is fundamentally insufficient (Chakraborty et al., 2024), and that practical alignment must accommodate preference inconsistency rather than assume it away (Xie et al., 2025). Yet most theoretical frameworks for online learning from human feedback still rely on the assumption that preferences are consistent – that there exists an intrinsic value for each option that is independent of the context in which it is presented.

In this paper, we provide rigorous theoretical foundations for learning from pluralistic and inconsistent human feedback. We study this problem in the framework of stochastic combinatorial bandits with semi-bandit feedback (Combes et al., 2015), where at each time step a learner selects a size-k set of items from a pool, observes a stochastic reward for each item, and seeks to maximize the total reward. Crucially, we allow the reward distribution of each item to depend on the set it is presented in (Saha & Gopalan, 2019), modeling, for example, recommendation systems where the appeal of each item depends on what else is shown alongside it.

We are specifically interested in **inconsistent and possibly contradictory preferences:** that is, we can have two items a and b and two sets s_1 and s_2, so that in expectation, item a is more valuable than item b in set s_1, but b is more valuable than a in s_2. Such inconsistency is pervasive in human decision-making. For instance, evaluators often prefer student A over B, B over C and C over A when compared in pairs (Tversky, 1969); people exhibit inconsistent preferences for eco-friendly products under different contexts (MacDonald et al., 2009); and the well-known "framing effect" describes "reversals of preference induced by changes in the reference points" (Kagel & Roth, 2020). In the language of pluralistic alignment, these inconsistencies arise naturally when aggregating feedback from diverse stakeholders with conflicting values.

However, many popular parametric models (e.g., multinomial logits, random utility, etc.) do not allow for such inconsistencies. Indeed, in the worst case, general inconsistent preferences imply that the feedback of one set will reveal nothing about any other – an impossible situation for an online learner. Meanwhile, in practice, a simple approach is seen to work even with inconsistent preferences (e.g., the closely-related Sparring algorithm by Ailon et al., 2014): maintain a per-item UCB as would be done in a simple non-

[1]Department of Computer Science, University of Texas at Austin, TX, US [2]Department of Electrical and Computer Engineering, University of Texas at Austin, TX, US. Correspondence to: Shuo Yang <yangshuo_ut@utexas.edu>.

Pluralistic Alignment Workshop @ ICML 2026, Seoul, South Korea.

Algorithm	Regret	Best Set	Set-Dep. Reward	Inconsistent Preferences
CUCB (Chen et al., 2013)	$O\left(\frac{k^2 n \log T}{\epsilon}\right)$	✓	✗	✗
CombUCB1 (Kveton et al., 2015b)	$O\left(\frac{kn \log T}{\epsilon}\right)$	✓	✗	✗
ESCB (Combes et al., 2015)	$O\left(\frac{\sqrt{k} n \log T}{\epsilon}\right)$	✓	✗	✗
TS Comb. Bandits (Zhang & Combes, 2024)	$O\left(\frac{n \log k}{\epsilon} \log T + \text{poly}\right)$	✓	✗	✗
Explor.-Exploit. (Agrawal et al., 2019)	$O\left(\frac{kn \log T}{\epsilon}\right)$	✓	✓ (MNL)	✗
MNL Optimal (Lee & Oh, 2024)	$\widetilde{O}\left(n\sqrt{T}\right)$	✓	✓ (MNL)	✗
Choice Bandits (Agarwal et al., 2020)	$O\left(\frac{n^2 \log n}{\epsilon^2} + \frac{n \log T}{\epsilon^2}\right)$	✗	✓	✓
MaxMin-RLHF (Chakraborty et al., 2024)	–	✗	✓	✓
Algorithm 1 (Ours)	$O\left(\min\left(\frac{k^3 n \log T}{\epsilon}, k^2\sqrt{nT \log T}\right)\right)$	✓	✓	✓

Table 1. Regret upper bounds and settings for stochastic combinatorial bandits. "Best Set": whether the algorithm finds the best size-k set (✓) or only the best single item (✗). "Set-Dep. Reward": whether item rewards depend on the set. "Inconsistent Preferences": whether the algorithm handles contradictory orderings across sets. MaxMin-RLHF addresses diverse preferences in an LLM alignment setting (not directly comparable regret). Our algorithm (Algorithm 1) is the only method that simultaneously finds the best set, handles set-dependent rewards, and allows inconsistent preferences with provable regret guarantees.

combinatorial bandit, and in every step pick the k items for which these UCB estimates are the highest. This behavior is seen even though UCB was not designed for this setting, and there is little theoretical understanding. We, therefore, seek a theoretical understanding of the following question:

Can we provably learn optimal decisions from pluralistic and inconsistent human feedback, and why do simple UCB-based algorithms succeed in this setting?

Notice that **existing analysis of UCB does not provide a regret bound when the preference is set-dependent and inconsistent**, as there are no fixed expected rewards (or any notion of intrinsic value) associated with the items. One suboptimal item, under inconsistent preferences, can have large reward expectations in many suboptimal sets and therefore has large UCB, which potentially leads to linear regret.

In this paper, we present a surprisingly weak assumption that allows for inconsistent preferences, generalizes many popular parametric (and consistent) models, and most importantly guarantees that UCB finds the best set. In particular, our **main contributions** are summarized below:

- We present the *pluralistic reward inconsistency with structural monotonicity (PRISM)* assumption (Assumption 1), which only requires that each item in the optimal set has lower individual reward there than in any other set containing it. PRISM allows for inconsistent orderings while subsuming many standard models (MNL, RUM, etc.). It provides a natural minimal structure for pluralistic alignment: the best overall combination is the most *competitive*, so each individual option would fare better in a weaker context.
- We next present a novel analysis of the UCB-based algorithm under the PRISM assumption (Algorithm 1). We prove that this algorithm has a gap-dependent $O(nk^3 \log T/\epsilon)$ regret upper bound, as well as a gap-independent $O(k^2\sqrt{nT \log T})$ regret upper bound (Theorem 3).
- Finally, we prove a regret lower bound $\Omega(\frac{n \log T}{\epsilon})$ (Theorem 4) under PRISM. The lower bound nearly matches the regret bound of the UCB-based algorithm for constant k (which is common in practice), up to logarithmic factors.

2. Related Work

Pluralistic Alignment and Diverse Preferences. Recent work has recognized that aligning AI systems with a single consistent reward function is insufficient when human preferences are diverse and potentially contradictory. Sorensen et al. (2024) proposes a roadmap for pluralistic alignment, identifying three modes of pluralism (Overton, steerable, and distributional). Conitzer et al. (2024) argues that social choice theory should guide the aggregation of diverse human feedback. Chakraborty et al. (2024) proves an impossibility result showing that a single reward model cannot capture diverse preferences and proposes MaxMin-RLHF. Other approaches include multi-objective preference optimization (Zhou et al., 2024), personalized reward modeling (Chen et al., 2025; Poddar et al., 2024), and direct alignment with heterogeneous user types (Shirali et al., 2025). See Xie et al. (2025) for a comprehensive survey. Our work contributes to this line of research by providing *provable regret guarantees* for learning from inconsistent feedback in an online combinatorial setting, complementing the primarily empirical focus of existing pluralistic alignment work.

Combinatorial Bandits with Consistent Preferences. Most reward models for combinatorial bandits rely on an

(implicit) assumption of consistent preferences. The simplest model assumes rewards are generated independently of the selected set (Chen et al., 2013; Kveton et al., 2015b; Combes et al., 2015; Simchowitz et al., 2016). More complex models capture set-dependent rewards but still assume consistent preferences, including the Multinomial Logit Model (Abeliuk et al., 2016; Agrawal et al., 2019; Saha & Gopalan, 2019; Flores et al., 2019), Markov chain models (Désir et al., 2015; Blanchet et al., 2016), and Random Utility Models (Berbeglia, 2016). Recent advances include Thompson Sampling with polynomial (in dimension) regret for combinatorial semi-bandits (Zhang & Combes, 2024), near-minimax optimal MNL bandits (Lee & Oh, 2024), and UCB algorithms for social welfare objectives (Sarkar et al., 2025). All these models are subsumed by our PRISM assumption (Assumption 1).

Inconsistent and Intransitive Preferences. Several works consider inconsistent preferences, but focus on settings different from ours. Choice Bandits (Agarwal et al., 2020) assumes a single best arm across all sets. Dueling bandits works (Ramamohan et al., 2016; Suk & Agarwal, 2023; Sui et al., 2018) extend to inconsistent preferences (e.g., Copeland or Borda winners) but aim to find the best single arm, not the optimal set. Recent work on preference models beyond Bradley-Terry (Zhang et al., 2025) and online RLHF with non-transitive preferences (Ye et al., 2024) addresses intransitivity in LLM alignment but in pairwise comparison settings rather than set selection. Dimakopoulou et al. (2019) considers potentially inconsistent preferences in slate bandits but provides no theoretical regret guarantee. Our work is unique in providing *provable near-optimal regret* for *set selection* under inconsistent preferences.

Scope. Important related problems outside our scope include: dueling bandits (Yue et al., 2012; Saha & Gopalan, 2019) (finding the best single arm from comparisons), submodular bandits (Yue & Guestrin, 2011; Chen et al., 2017) (variable set sizes), cascade bandits (Kveton et al., 2015a; Cheung et al., 2019) (position-dependent rewards), and non-additive reward functions (Rhuggenaath et al., 2020; Agarwal et al., 2021) (set-independent item rewards with non-additive aggregation).

3. Problem Setup and PRISM Assumption

In this section, we formalize the online learning problem in which pluralistic and inconsistent feedback arises, present a motivating example, and then introduce the *pluralistic reward inconsistency with structural monotonicity* (PRISM) assumption (Assumption 1) that makes learning tractable despite inconsistency. We further define "consistent preferences" (Definition 1) and show that many widely studied models (MNL, RUM, etc.) assume consistent preferences and are special cases of PRISM.

Stochastic combinatorial multi-armed bandits problem with semi-bandit feedback. Given a fixed set of arms $\mathcal{A} = \{a_1, a_2, \cdots, a_n\}$, let $\mathcal{S}$ denote all the size-k subsets of $\mathcal{A}$. At each time step t, the online learner selects a set $s(t) \in \mathcal{S}$, and then observes a per-arm stochastic reward $X_{a,s(t)}$ of all its arms $a \in s(t)$. The stochastic reward for the set $s(t)$ is $\sum_{a\in s(t)} X_{a,s(t)}$ – i.e. the *set reward is the sum of the (set-dependent) per-arm rewards.*

This setup models, for example, click-through rates in recommendation systems: every time a set is presented to a user, the learner gets to observe which items were clicked and which were not, and is trying to maximize the total number of clicks. We allow the probability of an item being clicked to depend on the set, and be possibly inconsistent across sets. In the context of pluralistic alignment, this captures settings where the aggregate feedback from a diverse user population is inherently inconsistent – e.g., different demographic groups may prefer different items depending on what alternatives are presented alongside them.

We denote the expected reward of arm a in set s to be $Q_s(a) \triangleq \mathbb{E}[X_{a,s}]$. The optimal set is denoted by $s^* \triangleq \arg\max_s \sum_{a\in s} Q_s(a)$, and finally, the regret for time t is

$$reg(t) = \sum_{a\in s^*} Q_{s^*}(a) - \sum_{a\in s(t)} Q_{s(t)}(a),$$

and the regret up to time T is $R(T) \triangleq \mathbb{E}\left(\sum_{t=1}^{T} reg(t)\right)$. The online learner aims to minimize $R(T)$.

Before formally introducing our assumption for inconsistent preferences, we first present a motivating example.

3.1. A Motivating Example

Consider a synthetic example of providing recommendations to a customer looking for cameras. There are 6 candidates {`Nikon`, `Sony`, `Canon`, `Digital Camera`, `Keyboard`, `Shoes`}. Every time we need to offer 3 recommendations and the customer accepts at most one of them. A customer's acceptance gives the recommendation reward 1 and otherwise gives reward 0.

Suppose the user is interested in {`Nikon`, `Sony`, `Canon`}, but when some of them are not recommended, the `Digital Camera` recommendation will partially capture the corresponding interest.

Specifically, we set the expected reward to be $Q_{\cdot}(\texttt{Nikon}) = 0.35, Q_{\cdot}(\texttt{Canon}) = 0.3, Q_{\cdot}(\texttt{Sony}) = 0.25, Q_{\cdot}(\texttt{Keyboard}) = 0.01, Q_{\cdot}(\texttt{Shoes}) = 0.01$, where the dot means any set containing the concerning recommendation. Further, set $Q_s(\texttt{Digital Camera}) = 0.85 - \sum_{a\in s, a\neq \texttt{Digital Camera}} Q_s(a)$. We show 4 representative sets in Figure 1. It can be verified that the optimal

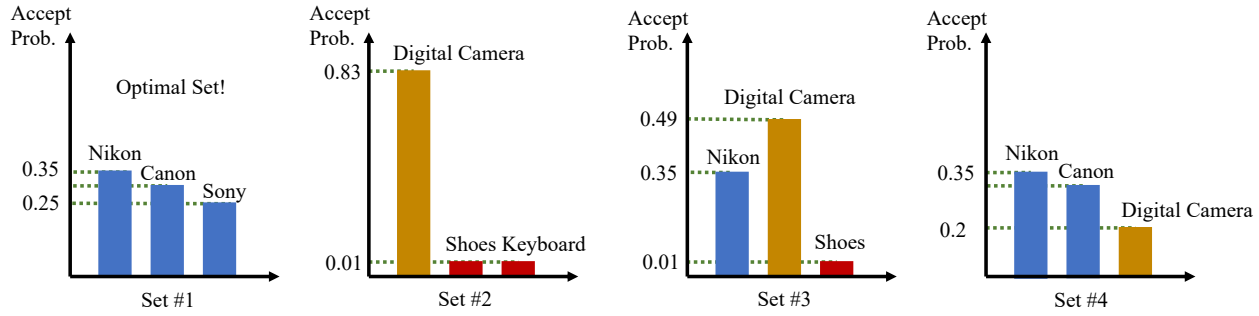

Figure 1. Four representative sets. The set #1 is optimal, as it maximizes the sum of the accepting probability of the recommendations. The `Digital Camera` has the highest accepting probability in many sub-optimal sets (even when paired with the recommendations belonging to the optimal set. See set #3). Such instances break the consistent preferences, but are covered by Assumption 1.

set is {`Nikon, Sony, Canon`} as the total expected reward is the highest 0.9.

Notice that the existence of the recommendation `Digital Camera` makes the problem harder. As shown in Figure 1, the `Digital Camera` has the highest accepting probability in many sets. Further, observe that a user is more likely to accept `Digital Camera` than `Nikon` in Set #3, whereas `Nikon` belongs to the optimal set. This makes `Digital Camera` seemingly a good recommendation, but it is not part of the optimal set.

Generally, it is conceivable that there are cases where the optimal arms are not the best in all sets, and there exist sub-optimal arms that have higher expected rewards in many sets. The UCB-based algorithm (which is widely used as a heuristic under inconsistent preferences) can, therefore, over-value some suboptimal arms and thus has linear regret.

In the following sections, however, we theoretically show that the UCB-based algorithm has near-optimal regret even under some inconsistent preferences. The crux is adopting the *PRISM* assumption, which is formally defined in the next subsection. The PRISM assumption allows for inconsistent preferences, subsumes many previously studied reward models, and most importantly, guarantees that UCB finds the optimal set.

3.2. PRISM Assumption

Here we formally define the PRISM assumption and show how it allows for inconsistent preferences.

Assumption 1 (Pluralistic Reward Inconsistency with Structural Monotonicity (PRISM)). *Let s^* be the optimal set, s be any other set and arm $a \in s \cap s^*$ be in both sets. Then PRISM requires that $Q_s(a) \geq Q_{s^*}(a)$ – i.e. arm a's individual reward in s is larger than its individual reward in s^*.*

Remark 1: PRISM means that the optimal set s^* is the most *competitive* set – while the *overall sum* of arm rewards is highest in s^*, any arm would have fared better in a different set because that set would have less competitive options.

Remark 2: One salient feature of PRISM is ***not*** assuming consistent preferences over arms $a \in \mathcal{A}$ at any time t. We first present an example that is allowed by our assumption but not other commonly seen reward models. We then formally discuss the "consistent preference" in the next subsection.

Example 1. *For any $k > 2$, without loss of generality, we take $a_1 \in s^*, a_2 \in s^*$ with $Q_{s^*}(a_1) \geq Q_{s^*}(a_2)$. For some sub-optimal set s_i, Assumption 1 allows for:*

1. ***Reversed relative reward expectation**:*

$$Q_{s^*}(a_1) \geq Q_{s^*}(a_2), \quad Q_{s_1}(a_2) > Q_{s_1}(a_1),$$
for some s_1 containing a_1, a_2.

2. ***Non-transitive relative reward expectation**: for some s_4 containing a_2, a_3, and s_5 containing a_1, a_3,*

$$Q_{s^*}(a_1) > Q_{s^*}(a_2), \quad Q_{s_4}(a_2) > Q_{s_4}(a_3),$$
$$Q_{s_5}(a_3) > Q_{s_5}(a_1).$$

Note that the s_5 in the "non-transitive" part of Example 1 also shows that Assumption 1 allows the arms not in s^* to be better than the arms belonging to s^* in some sub-optimal set. This corresponds to the `Digital Camera` in the motivating example in Section 3.1.

3.3. Existing Models are Strongly Consistent

Informally, consistent preferences mean that one arm is intrinsically more valuable than another, irrespective of the sets in which both those arms are presented. In this section, we first formally define consistent preferences, and then show that three widely used models – multinomial logit, random utility, and independent reward – implicitly assume consistent preferences. As mentioned above, our PRISM assumption covers cases that are not consistent; however,

in this section, we show that it *also* covers any strongly consistent setting.

Definition 1 (Strong Consistent Preferences)**.** *Set dependent arm rewards $\{Q_s(a)\}$ are said to represent strong consistent preferences if there exists a* **total ordering** *of arms. In particular, for any pair of arms a_i and a_j such that $a_i \succ a_j$, and any corresponding pair of size-k sets s and s' which differ only in these arms – i.e. $s = s' - a_j + a_i$ we have that the set-dependent arm rewards satisfy*

$$Q_s(a_i) \geq Q_{s'}(a_j) \text{ and } Q_s(a) \leq Q_{s'}(a),\ \forall a \in s \cap s'.$$

Further, the total set rewards also satisfy $\sum_{a\in s} Q_s(a) \geq \sum_{a\in s'} Q_{s'}(a)$.

We now show here that three widely adopted reward models all assume a "strong consistent preference", which are all also covered by PRISM (Assumption 1). For clarity of exposition, here we focus on the binary reward with $X_{a,s} \in \{0, 1\}$ and $Q_s(a)$ is therefore the probability that a receives reward 1 in set s.

Multinomial Logit (MNL): MNL assumes a deterministic utility v_i associated with each a_i and the probability of a_i receiving non-zero reward in s is $Q_s(a_i) = \frac{e^{v_i}}{e^{v_0}+\sum_{a_j\in s} e^{v_j}}$, where v_0 is some constant modeling the event of no arm receiving non-zero reward. One can verify that the v_is of MNL induce strong consistent preferences and the optimal set s^* is composed by arms with highest v_i. Assumption 1 covers MNL since $e^{v_0} + \sum_{a_j\in s} e^{v_j} \leq e^{v_0} + \sum_{a_j\in s^*} e^{v_j}, \forall s \neq s^*$.

Random utility model (RUM): RUM assumes a (random) utility associated for all $a_i \in \mathcal{A}$, with $U_i = v_i + \epsilon_i$, where v_i is a deterministic utility and ϵ_is are i.i.d. random variables drawn at every time step t. The probability of a_i in s receiving non-zero reward is given by $Q_s(a_i) = P(U_i > U_j, \forall a_j \in s \text{ and } i \neq j)$. To model the event of no arm $a \in s$ receiving non-zero reward, s can be augmented to $s \cup \{a_0\}$, with random utility U_0 of a_0 defined similarly. When U_0 is the largest, no arm $a \in s$ receives non-zero reward. It can be verified that v_is in RUM induce a strong consistent preference, and the optimal set s^* is composed by arms with highest v_i. For any arm $a \in s^*$, putting it to a sub-optimal set s leads to arm a having a larger chance of receiving non-zero reward, as other arms have smaller v_i, thus satisfies Assumption 1.

Independent reward: Independent reward model assumes a deterministic reward expectation v_i associated with arm a_i. For the arm a_i in any set s, it assumes $Q_s(a_i) = v_i$. The v_is immediately induce a strong consistent preference. The independent reward model is also covered by Assumption 1, as $Q_s(a_i)$ does not change in different s.

Finally, we show that the PRISM assumption represents a strict generalization of strong consistent preferences. That is, it subsumes all strongly consistent reward models, but also allows for inconsistent models.

Lemma 2. *Any reward model $\{Q_s(a)\}$ that represents strong consistent preferences also satisfies PRISM (Assumption 1). However, the reverse is not true; there exist reward models that satisfy PRISM but do not represent strong consistent preferences.*

We defer the proof to Section B.

4. UCB-based Algorithm and Regret Analysis

Having established PRISM as a structural condition that accommodates pluralistic preferences, we now show that a remarkably simple algorithm suffices to learn optimally under it. In this section, we describe a UCB-based algorithm and present its regret bounds (both gap-dependent and gap-independent), demonstrating a novel analysis that allows set-dependent arm rewards as long as PRISM is satisfied.

We emphasize that our main contribution is not in algorithmic innovation, but in rigorously proving that the UCB-based algorithm achieves near-optimal regret under inconsistent preferences (Assumption 1).

4.1. Algorithm

Denote $N_i(t)$ to be the number of times that a_i is included in the selected set s up to time t, $C_i(t)$ to be the cumulative reward of arm a_i at time t. We have Algorithm 1 that extends the standard α-UCB algorithm. It selects a set of arms with top-k UCB in each step. It is worth noting that Algorithm 1 only keeps track of the cumulative reward of the arms in $\mathcal{A}$, without accounting for any set-dependent information. Though it may seem contradictory to the set-dependent reward distribution, we will show that Algorithm 1 achieves near-optimal regret.

4.2. Regret Bound

Let $\epsilon = \sum_{a\in s^*} Q_{s^*}(a) - \max_{s\neq s^*} \sum_{a\in s} Q_s(a)$ denote the minimum gap in expected reward between the optimal set s^* and any sub-optimal set s. Recall that k is the size of the selected set s, and n is the size of $\mathcal{A}$. Suppose the reward $X_{a,s}$ is bounded by B (i.e., $X_{a,s} \in [0, B]$) for all a and s. Our next result provides a regret bound of Algorithm 1.

Theorem 3 (Regret Bound of Algorithm 1)**.** *For combinatorial bandits problem under Assumption 1, run Algorithm 1 with parameter $\alpha \geq 2$, we have*

$$R(T) \leq O\left(\min\left(\frac{B^2k^3n\log T}{\epsilon}, Bk^2\sqrt{nT\log T}\right)\right).$$

For the "well-separated" problem (i.e., ϵ is large), the regret scales with $\log T$; and when the sub-optimality gap ϵ

Algorithm 1 UCB-BASED ALGORITHM FOR COMBINATORIAL BANDITS WITH INCONSISTENT PREFERENCES

Online learning task: Given a set of arms, in each time choose a size-k subset so as to maximize reward. Feedback in every step is a stochastic reward for every arm in that chosen set; the distribution of these rewards can be set-dependent, and inconsistent across sets.
Input: arm set $\mathcal{A}$ of size n, set size k, time horizon T, rewards bounded by B
Parameter: A constant α, normally set to 2
Initialize: $UCB_i(1) = INF$, $N_i(1) = 0$, $C_i(1) = 0$ for all arm $a_i \in \mathcal{A}$
for $t = 1$ **to** T **do**
  Construct set $s(t)$ with arms that have top-k $UCB_i(t)$, ties break randomly. For all $a_i \in s(t)$, Set $N_i(t+1) = N_i(t) + 1$
  Observe feedback. Set $C_i(t+1) = C_i(t) + X_{a_i,s(t)}$
  $UCB_i(t+1) = \frac{C_i(t+1)}{N_i(t+1)} + B\sqrt{\frac{\alpha \log T}{N_i(t+1)}}$, for all arm $a_i \in s(t)$, and $UCB_i(t+1) = UCB_i(t)$, for others
end for

is small, the gap-independent bound $Bk^2\sqrt{nT\log T}$ will dominate the min, and this recovers the standard gap-independent $\widetilde{O}(\sqrt{T})$ regret scaling.

Further, we have the following regret lower bound for the combinatorial bandits under PRISM.

Theorem 4 (Regret Lower Bound)**.** *For any online learning algorithm that achieves $o(T^c)$ regret for all constant $c > 0$, there exists a problem instance that satisfies Assumption 1, such that the algorithm induces a regret of $\Omega\left(\frac{B^2 n \log T}{\epsilon}\right)$.*

The dependency of B, n, T, ϵ in the lower bound matches the gap-dependent upper bound (Theorem 3). For k being constant, which is commonly seen in practice (e.g., a constant number of displaying slots in online recommendation systems), our regret bound nearly matches the lower bound up to logarithmic terms. This shows the optimality of Algorithm 1 despite inconsistent preferences.

Regret Upper Bound Analysis Intuition To see how the UCB-based algorithm works under PRISM, we first present an illustrative experiment here. The environment is the motivating example presented in Section 3.1, where the optimal set is {`Nikon, Canon, Sony`} but `Digital Camera` has the highest reward expectation in many sets.

Figure 2 shows the process of the UCB-based algorithm converging to the optimal set. The observation is that the UCB of most arms decreases together, and the UCB of the optimal arms sequentially separates out.

The regret analysis follows this observation closely and can be summarized as 3 steps:

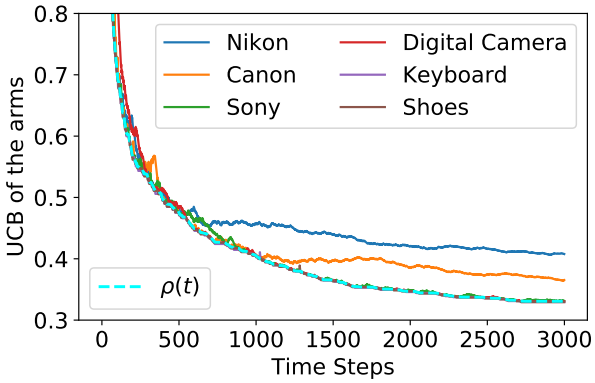

Figure 2. Evolution of UCB in the environment defined in Section 3.1. **Observation:** the UCB of all arms decreases together initially, and the arms in the optimal set separate out later. $\rho(t)$ (a "lower bound" of the played arms' UCB, formally defined Section 4.3) precisely captures this decreasing-together dynamics. As $\rho(t)$ decreases, the UCB of two recommendations `Nikon, Canon`, belonging to the optimal set, separate out from $\rho(t)$, and are therefore included in all the subsequently played sets. This happens with `Digital Camera` having the largest reward expectation in many suboptimal sets.

- *Step I:* proving that the UCB of the optimal arms stays large. The PRISM assumption is invoked here, which guarantees that the UCB of an optimal arm a_i is always larger than $Q_{s^*}(a_i)$.
- *Step II:* showing that the UCB of most of the arms decreases together and stays close to each other (characterized by $\rho(t)$, see Figure 2 and definition in Section 4.3).
- *Step III:* showing $\rho(t)$ can not stay high for a long time, which is then converted into a regret bound.

All our analysis focuses on characterizing the dynamics of UCB – how the UCB of the optimal arms and other arms decays. It does not require the arms to have fixed reward expectations or consistent preferences, which is drastically different from the standard UCB analysis.

4.3. Proof Sketch

The following results are for Algorithm 1 with $\alpha \geq 2$. We sketch the proof into 3 steps, which correspond to our discussion for analysis intuition. W.l.o.g., let $s^* = \{a_1, a_2, \cdots, a_k\}$ with $Q_{s^*}(a_1) \geq \cdots \geq Q_{s^*}(a_k)$.

Step I: UCB of optimal arms stays large.

We first show that the $UCB_i(t)$ of $a_i \in s^*$ is lower bounded by $Q_{s^*}(a_i)$ for all time steps, with high probability.

Lemma 5 (UCB is optimistic)**.** *With probability at least $1 - \frac{2}{T}$, we have $UCB_i(t) \geq Q_{s^*}(a_i)$ simultaneously for all time step $t \in [T]$ and all arm $a_i \in s^*$.*

This follows from the PRISM assumption, where the expected reward $Q_s(a_i) \geq Q_{s^*}(a_i)$ for all $s \neq s^*$ and $a_i \in s \cap s^*$. Therefore, as long as $UCB_i(t)$ is an optimistic estimate (i.e., $UCB_i(t) \geq \sum_{\tau=1}^{t} Q_{s(\tau)}(a_i)/N_i(t)$, see Corollary 10), we have that $UCB_i(t) \geq Q_{s^*}(a_i)$, with high probability.

Step II: UCB of most arms decay together as $\rho(t)$.

Here we formalize the observation in Figure 2. Let $\rho'(t) = \min_{a_i \in s(t)} UCB_i(t)$, and $\rho(t) = \min_{\tau \leq t} \rho'(\tau)$. By definition, $\rho(t)$ is monotonically non-increasing, and $UCB_i(t) \geq \rho'(t) \geq \rho(t)$, $\forall a_i \in s(t)$, (i.e., $\rho(t)$ is a lower bound for the UCB of the arms in $s(t)$).

The following lemma shows that for the arms not in $s(t)$, $\rho(t)$ is always an upper bound, and soon a tight estimate of all their UCB.

Lemma 6 (Dynamics of UCB)**.** $\rho(t) \geq UCB_i(t) \geq \rho(t)\left(1 - \frac{1}{N_i(t)}\right)$, $\forall a_i \notin s(t), \forall t \in [T]$.

Proof. For any arm $a_i \notin s(t)$, let $t' \leq t$ be the last time step that $a_i \in s(t')$. We then have

$$C_i(t') + \sqrt{\alpha N_i(t') \log T} \geq \rho'(t') N_i(t') \geq \rho(t') N_i(t') \geq \rho(t) N_i(t').$$

The last step holds as $\rho(t)$ is non-increasing. With $C_i(t) \geq C_i(t')$ and $N_i(t) = N_i(t') + 1$, we have

$$C_i(t) + \sqrt{\alpha N_i(t) \log T} \geq \rho(t)\left(N_i(t) - 1\right).$$

Dividing both sides by $N_i(t)$ gives the second inequality. It is left to show $\rho(t) \geq UCB_i(t)$, $\forall a_i \notin s(t)$. Let $t'' \leq t$ be the last time step $\rho'(t'') = \rho(t)$. It implies

$$\rho'(\tau) > \rho'(t'') = \rho(t) \geq UCB_i(t''), \quad \forall \tau \in (t'', t], a_i \notin s(t'').$$

Notice that $UCB_i(\tau+1) = UCB(\tau)$ if $a_i \notin s(\tau)$. Therefore for any $a_i \notin s(t'')$, it implies $a_i \notin s(\tau), \forall \tau \in [t'', t]$.

Notice that there are $(n-k)$ arms not in $s(t)$ and the same number of arms not in $s(t'')$, we have $a_i \notin s(t'') \iff a_i \notin s(t)$. Thus

$$UCB_i(t) = UCB_i(t'') \leq \rho'(t'') = \rho(t), \ \forall a_i \notin s(t).$$

This completes the proof. □

Note that Lemma 6 implies that all arms a_i with $UCB_i(t) \geq \rho(t)$ are included in $s(t)$. Therefore, combining with Lemma 5, we know that for all $a_i \in s^*$, with probability at least $1 - \frac{2}{T}$, once $\rho(t)$ falls below $Q_{s^*}(a_i)$, the subsequently played sets $s(t)$ will always contain a_i. As $\rho(t)$ keeps decreasing, the optimal set s^* will be recovered sequentially, from a_1 to a_k.

Step III: $\rho(t)$ can not stay large for long.

The rest of the proof focuses on characterizing how fast $\rho(t)$ decays and converting it to a regret bound. Notice that $\rho(t)$ is the rolling-min of $\rho'(t)$ and is, therefore, monotonically non-increasing by definition. For $l \leq k$, let time t_l be the last time that $\rho(t_l) \geq Q_{s^*}(a_l)$. Let t'_l be the number of times that s^* is selected before t_l. Lemma 7 presents a bound for the number of times that sub-optimal sets are played before t_l, which is measured by $t_l - t'_l$.

Lemma 7 (Bound the times of selecting sub-optimal set)**.** *With probability at least $1 - \frac{2}{T}$, we can bound $t_l - t'_l$, for all $l \in [k]$ as,*

$$t_l - t'_l \leq \frac{40\alpha B^2 lkn \log T}{(\Delta_l + \epsilon)^2}, \text{ if } \Delta_l \geq \frac{\epsilon}{10}; \text{ and}$$
$$t_l - t'_l \leq \frac{40\alpha B^2 lkn \log T}{\epsilon^2}, \text{ otherwise},$$

where $\Delta_l := \sum_{i=l}^{k} [Q_{s^*}(a_l) - Q_{s^*}(a_i)]$.

Remark 3: Suppose $\rho(T) \geq Q_{s^*}(a_l)$ for some $a_l \in s^*$ (i.e., $\rho(t)$ does not fall below $Q_{s^*}(a_l)$ for the entire time horizon), we have that $t_l = T$ by definition, and Lemma 7 still holds.

We emphasize that Lemma 7 is crucial to prove regret bound with inconsistent preferences (Assumption 1), as it does not rely on each arm having a set-independent reward expectation, which is drastically different from existing UCB analysis. The next lemma connects regret $R(T)$ to $t_l - t'_l$ for $l \leq k$.

Lemma 8 (Regret decomposition)**.** *With probability at least $1 - \frac{2}{T}$, we have*

$$R(T) \leq 2B\sqrt{\alpha kn(t_k - t'_k)\log T} + \sum_{l=1}^{k-1} \delta_{lk}\left(t_l - t'_l\right) + nQ_{s^*}(a_k).$$

where $\delta_{ij} := Q_{s^*}(a_i) - Q_{s^*}(a_j)$.

Combining Lemmas 7 and 8 proves Theorem 3.

5. Experiments

We empirically evaluate Algorithm 1 on environments with different reward models (see Figure 3) satisfying the PRISM assumption. The environments range from standard consistent models (MNL, RUM) to genuinely pluralistic settings where preferences are inconsistent across sets (Preference Matrix, Random PRISM). This demonstrates that our algorithm performs well across the full spectrum from consistent to pluralistic feedback. We summarize the environments below, with details provided in Section E.

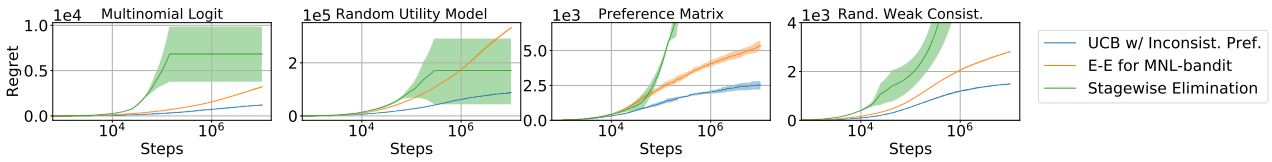

Figure 3. Synthetic experiments with different reward models. The curves are the averages and standard deviations of 5 independent runs. The "UCB w/ Inconsist. Pref." is Algorithm 1 with $\alpha = 2$. "E-E MNL-bandit" refers to the "Exploration-Exploitation algorithm for MNL-Bandit" (Agrawal et al., 2019). "Stagewise Elimination" was proposed in (Simchowitz et al., 2016). The parameters are specified as in the original papers.

Multinomial Logit: We generate $n = 20$ arms, where each arm a_i has an intrinsic value $v_i = \log(1 - 0.04i)$. The MNL model is used to determine the reward expectation $Q_s(a_i) = \frac{e^{v_i}}{e^{v_0} + \sum_{a_j \in s} e^{v_j}}$, with $v_0 = 0$. The set size is set to $k = 10$ and the number of possible sets is $184,756$.

Random Utility Model: We generate $n = 20$ arms, where each arm a_i has an intrinsic utility $v_i = 1 - 0.04i$. In every step, the random utility U_i of all arms in the set $s(t)$ are independently generated with mean μ_i and unit variance from Gaussian distribution. Besides that, the null arm a_0 will draw a $U_0 \sim \mathcal{N}(2, 1)$. If U_0 is the maximum, then the entire set receives 0 reward. Otherwise, the arm with the largest random utility U_i receives the reward 1 and others receive 0. The set size is set to $k = 5$ and the number of possible sets is $15,504$.

Preference Matrix (pluralistic): We set the total number of arms to $n = 10$ and the set size to $k = 2$, then directly specify a 10-by-10 preference matrix M to determine the probability of an arm receiving a reward. The preference is deliberately set to be inconsistent – while $s^* = \{a_1, a_2\}$ with $Q_{s^*}(a_1) = 0.47$ and $Q_{s^*}(a_2) = 0.45$, the suboptimal arm a_3 has a large reward expectation when paired with other suboptimal arms (i.e., $Q_s(a_3) = 0.675$ for s not containing a_1, a_2). This models a pluralistic setting where a "popular compromise" option (a_3) looks individually strong but is suboptimal as part of the best set.

Random PRISM (pluralistic): We randomly generate the environment that satisfies Assumption 1 via rejection sampling. We set the total number of arms to $n = 10$ and the set size to $k = 5$. These randomly generated environments need not satisfy the assumption of any parametric model (MNL, RUM, etc.) and represent the general pluralistic case where only PRISM holds.

Along with Algorithm 1, we also take "E-E for MNL-bandit" (Exploration-Exploitation algorithm for MNL, (Agrawal et al., 2019)) and "Stagewise Elimination" (Simchowitz et al., 2016) for comparisons, which are designed for "Multinomial Logit" and "Random Utility Model" environment, respectively. The algorithms are tested in the environments listed above. The average regret and standard deviation of 5 independent runs are reported in Figure 3.

"E-E for MNL-bandit" and "Stagewise Elim" perform relatively well in the environments that they are designed for. Note that in the "Preference Matrix" environment and "Random PRISM" environment, there is no consistent preference among the arms. The "Stagewise Elimination" falsely eliminates an arm that belongs to the optimal set, and therefore suffers from linear regret. Despite the inconsistent preferences, all evaluated environments satisfy PRISM (Assumption 1) and the UCB algorithm (Algorithm 1) performs the best in all the testing environments.

6. Conclusion and Discussion

In this paper, we study online learning from inconsistent human feedback in the framework of combinatorial bandits with semi-bandit feedback. We present the pluralistic reward inconsistency with structural monotonicity (PRISM) assumption, which formalizes a minimal structural condition under which learning remains tractable despite pluralistic and contradictory preferences. PRISM subsumes many existing preference models (MNL, RUM, etc.) while allowing for the intransitive and context-dependent preferences that arise naturally in pluralistic settings. Under PRISM, we prove that a simple UCB-based algorithm achieves near-optimal regret for constant set size k.

Implications for Pluralistic Alignment. Our results carry broader implications for the emerging field of pluralistic AI alignment (Sorensen et al., 2024). First, PRISM provides a concrete *possibility result*: even when individual preferences are inconsistent across contexts, provably efficient learning is achievable under mild structural conditions. This complements the impossibility results showing that a single reward model cannot capture diverse preferences (Chakraborty et al., 2024; Conitzer et al., 2024). Second, the success of the simple UCB algorithm under PRISM suggests that practical systems may not need complex multi-objective or personalized mechanisms – a standard exploration-exploitation approach can suffice when the underlying preference structure satisfies PRISM.

References

Abeliuk, A., Berbeglia, G., Cebrian, M., and Van Hentenryck, P. Assortment optimization under a multinomial logit model with position bias and social influence. *4OR*, 14(1):57–75, 2016.

Agarwal, A., Johnson, N., and Shivani, A. Choice bandits. In *Advances in Neural Information Processing Systems*, 2020.

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

Agrawal, S., Avadhanula, V., Goyal, V., and Zeevi, A. Mnl-bandit: A dynamic learning approach to assortment selection. *Operations Research*, 67(5):1453–1485, 2019.

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

Berbeglia, G. Discrete choice models based on random walks. *Operations Research Letters*, 44(2):234–237, 2016.

Blanchet, J., Gallego, G., and Goyal, V. A markov chain approximation to choice modeling. *Operations Research*, 64(4):886–905, 2016.

Chakraborty, S., Qiu, J., Yuan, H., Koppel, A., Manocha, D., Huang, F., Bedi, A., and Wang, M. Maxmin-rlhf: Alignment with diverse human preferences. In *International Conference on Machine Learning*, pp. 6116–6135. PMLR, 2024.

Chen, D., Chen, Y., Rege, A., Wang, Z., and Vinayak, R. K. Pal: Sample-efficient personalized reward modeling for pluralistic alignment. In *The Thirteenth International Conference on Learning Representations*, 2025.

Chen, L., Krause, A., and Karbasi, A. Interactive submodular bandit. In *NIPS*, pp. 141–152, 2017.

Chen, W., Wang, Y., and Yuan, Y. Combinatorial multi-armed bandit: General framework and applications. In *International Conference on Machine Learning*, pp. 151–159, 2013.

Cheung, W. C., Tan, V., and Zhong, Z. A thompson sampling algorithm for cascading bandits. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 438–447. PMLR, 2019.

Combes, R., Shahi, M. S. T. M., Proutiere, A., et al. Combinatorial bandits revisited. In *Advances in Neural Information Processing Systems*, pp. 2116–2124, 2015.

Conitzer, V., Freedman, R., Heitzig, J., Holliday, W. H., Jacobs, B. M., Lambert, N., Mosse, M., Pacuit, E., Russell, S., Schoelkopf, H., et al. Position: Social choice should guide ai alignment in dealing with diverse human feedback. In *International Conference on Machine Learning*, pp. 9346–9360. PMLR, 2024.

Désir, A., Goyal, V., Segev, D., and Ye, C. Capacity constrained assortment optimization under the markov chain based choice model. *Operations Research, Forthcoming*, 2015.

Dimakopoulou, M., Vlassis, N., and Jebara, T. Marginal posterior sampling for slate bandits. In *IJCAI*, pp. 2223–2229, 2019.

Flores, A., Berbeglia, G., and Van Hentenryck, P. Assortment optimization under the sequential multinomial logit model. *European Journal of Operational Research*, 273 (3):1052–1064, 2019.

Kagel, J. H. and Roth, A. E. *The Handbook of Experimental Economics, Volume 2*. Princeton university press, 2020.

Karp, R. M. and Kleinberg, R. Noisy binary search and its applications. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 881–890, 2007.

Kveton, B., Szepesvari, C., Wen, Z., and Ashkan, A. Cascading bandits: Learning to rank in the cascade model. In *International Conference on Machine Learning*, pp. 767–776. PMLR, 2015a.

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

Lee, J. and Oh, M.-h. Nearly minimax optimal regret for multinomial logistic bandit. volume 37, pp. 109003–109065, 2024.

MacDonald, E. F., Gonzalez, R., and Papalambros, P. Y. Preference inconsistency in multidisciplinary design decision making. *Journal of Mechanical Design*, 131(3), 2009.

Poddar, S., Wan, Y., Ivison, H., Gupta, A., and Jaques, N. Personalizing reinforcement learning from human feedback with variational preference learning. volume 37, pp. 52516–52544, 2024.

Ramamohan, S. Y., Rajkumar, A., and Agarwal, S. Dueling bandits: Beyond condorcet winners to general tournament solutions. In *Advances in Neural Information Processing Systems*, pp. 1253–1261, 2016.

Rhuggenaath, J., Akcay, A., Zhang, Y., and Kaymak, U. Algorithms for slate bandits with non-separable reward functions. *arXiv preprint arXiv:2004.09957*, 2020.

Saha, A. and Gopalan, A. Combinatorial bandits with relative feedback. In *Advances in Neural Information Processing Systems*, pp. 983–993, 2019.

Sarkar, D., Pandey, N., and Chowdhury, S. R. Revisiting social welfare in bandits: Ucb is (nearly) all you need. *arXiv preprint arXiv:2510.21312*, 2025.

Shirali, A., Nasr-Esfahany, A., Alomar, A., Mirtaheri, P., Abebe, R., and Procaccia, A. Direct alignment with heterogeneous preferences. *arXiv preprint arXiv:2502.16320*, 2025.

Simchowitz, M., Jamieson, K., and Recht, B. Best-of-k-bandits. In *Conference on Learning Theory*, pp. 1440–1489, 2016.

Sorensen, T., Moore, J., Fisher, J., Gordon, M. L., Mireshghallah, N., Rytting, C. M., Ye, A., Jiang, L., Lu, X., Dziri, N., et al. Position: A roadmap to pluralistic alignment. In *International Conference on Machine Learning*, pp. 46280–46302. PMLR, 2024.

Sui, Y., Zoghi, M., Hofmann, K., and Yue, Y. Advancements in dueling bandits. In *IJCAI*, pp. 5502–5510, 2018.

Suk, J. and Agarwal, A. When can we track significant preference shifts in dueling bandits? volume 36, pp. 38347–38369, 2023.

Tversky, A. Intransitivity of preferences. *Psychological review*, 76(1):31, 1969.

Xie, Z., Wu, J., Shen, Y., Xia, Y., Li, X., Chang, A., Rossi, R., Kumar, S., Majumder, B. P., Shang, J., et al. A survey on personalized and pluralistic preference alignment in large language models. *arXiv preprint arXiv:2504.07070*, 2025.

Ye, C., Xiong, W., Zhang, Y., Dong, H., Jiang, N., and Zhang, T. Online iterative reinforcement learning from human feedback with general preference model. In *Advances in Neural Information Processing Systems*, volume 37, pp. 81773–81807, 2024.

Yue, Y. and Guestrin, C. Linear submodular bandits and their application to diversified retrieval. In *Advances in Neural Information Processing Systems*, pp. 2483–2491, 2011.

Yue, Y., Broder, J., Kleinberg, R., and Joachims, T. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.

Zhang, R. and Combes, R. Thompson sampling for combinatorial bandits: Polynomial regret and mismatched sampling paradox. In *Advances in Neural Information Processing Systems*, volume 37, pp. 89437–89467, 2024.

Zhang, Y., Zhang, G., Wu, Y., Xu, K., and Gu, Q. Beyond bradley-terry models: A general preference model for language model alignment. In *International Conference on Machine Learning*, pp. 76939–76965. PMLR, 2025.

Zhou, Z., Liu, J., Shao, J., Yue, X., Yang, C., Ouyang, W., and Qiao, Y. Beyond one-preference-fits-all alignment: Multi-objective direct preference optimization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 10586–10613, 2024.

A. Regret Lower Bound Proof

Proof. We prove the lower bound by constructing a family of environments $\mathcal{E}_i, i \in [n]$. We define the arm set as $\mathcal{A} = \{a_1, \cdots, a_{n+k-1}\}$ and consider the binary reward $X_{a_i,s} \in \{0, B\}$. In particular, we focus on the regime where $n \gg k$ and $B > 1$.

In environment $\mathcal{E}_i$, the optimal set is $\{a_i, a_{n+1}, a_{n+2} \cdots, a_{n+k-1}\}$. We assume the arms $\{a_1, a_{n+1}, a_{n+2} \cdots, a_{n+k-1}\}$ to have $\frac{1}{2}$ probability of receiving reward B in any set in any environment, and arm a_i has probability $\frac{1}{2} + \frac{\epsilon}{B}$ of receiving reward B in any set in environment $\mathcal{E}_i$ for $i \in [2, n]$. All other arms not belonging to the optimal set have $\frac{1}{2} - \frac{\epsilon}{B}$ probability of receiving positive reward in any set. It's easy to verify that all environments $\mathcal{E}_i$ satisfies Assumption 1 and the minimum gap between optimal and sub-optimal set is ϵ. We then have the following regret lower bound.

Let q_i be the probability measure in environment $\mathcal{E}_i$. The proof follows by showing that, for all $j \in [2, n]$, any algorithm has $\mathbb{E}_{q_1}(N_j(T)) = \Omega(B^2 \log T/\epsilon^2)$ when the algorithm achieves $o(T^a)$ regret in environment $\mathcal{E}_j$ and $\mathcal{E}_1$.

For any $j \in [2, n]$, define the event $B_j = \left\{N_j(T) \leq B^2 \log T/\epsilon^2\right\}$. We prove the lower bound on $\mathbb{E}_{q_1}(N_j(T))$ by two cases. We first start with the simple one:

Case I: $q_1(B_j) < 1/3$**.** We have

$$\mathbb{E}_{q_1}(N_j(T)|q_1(B_j) < 1/3) \geq q_1(B_j^c)B^2 \log T/\epsilon^2 = \Omega(B^2 \log T/\epsilon^2).$$

Case II: $q_1(B_j) \geq 1/3$**.** Note that in environment $\mathcal{E}_j$, the algorithm will incur at least ϵ regret if not selecting a_j, Therefore we have $\mathbb{E}_{q_j}(T - N_j(T)) = o(T^c)$ for any constant $c > 0$. By Markov's inequality, we have

$$q_j(B_j) = q_j\left(\left\{T - N_j(T) > T - B^2 \log T/\epsilon^2\right\}\right) \leq \frac{\mathbb{E}_{q_j}(T - N_j(T))}{T - B^2 \log T/\epsilon^2} = o(T^{c-1}).$$

From (Karp & Kleinberg, 2007), we know that for any event E and two distributions p, q with $p(E) > 1/3$ and $q(E) < 1/3$, we have

$$D_{\text{KL}}(p; q) \geq \frac{1}{3}\log(\frac{1}{3q(E)}) - \frac{1}{e},$$

where $D_{\text{KL}}(p; q)$ is the KL-divergence of p and q. Putting q_1, q_j and B_j into the inequality above, we have

$$D_{\text{KL}}(q_1; q_j) \geq \frac{1}{3}\log(\frac{1}{3o(T^{c-1})}) - \frac{1}{e} = \Omega(\log T).$$

On the other hand, we need to bound the KL-divergence of q_1 and q_j by playing any set containing a_j. Suppose p is a categorical distribution with parameters $p_1, ..., p_k$ for k items and p' is another categorical distribution with parameters $p_1 - \epsilon_1, ..., p_k - \epsilon_k$, We have

$$D_{\text{KL}}(p, p') = \sum_{i=1}^{k}(p'_i + \epsilon_i)\log\frac{p'_i + \epsilon_i}{p'_i} \leq \sum_{i=1}^{k}(p'_i + \epsilon_i)\frac{\epsilon_i}{p'_i} = \sum_{i=1}^{k}\frac{\epsilon_i^2}{p'_i},$$

where the last inequality holds because $\sum_{i=1}^{k}\epsilon_i = 0$. Since the only different arm between $\mathcal{E}_1$ and $\mathcal{E}_j$ is arm a_j and the probability of a_j receiving reward B in environment $\mathcal{E}_j$ is $\frac{1}{2} + \frac{\epsilon}{B} \geq \frac{1}{3}$. We can directly bound the KL-divergence of q_1 and q_j by

$$D_{\text{KL}}(q_1; q_j) \leq 3N_j(T)\frac{\epsilon^2}{B^2},$$

It then directly implies that

$$3N_j(T)\frac{\epsilon^2}{B^2} = \Omega(\log T) \implies \mathbb{E}_{q_1}(N_j(T)|q_1(B_j) \geq 1/3) = \Omega\left(\frac{B^2 \log T}{\epsilon^2}\right).$$

Therefore, combining **Case I** and **Case II**, we have $\mathbb{E}_{q_1}(N_j(T)) = \Omega(B^2 \log T/\epsilon^2)$, which holds for all $j \in [2, n]$. Notice that playing a_j induces ϵ regret, considering all $j \in [2, n]$ gives a regret lower bound $\Omega(B^2 n \log T/\epsilon)$. □

B. Proof for Section 3

B.1. Proof of Lemma 2

Proof. We first show that any model assuming strong consistent preferences (Definition 1) is covered by PRISM (Assumption 1). By Definition 1, we have the optimal set s^* is composed by the arms with largest v_i. For any sub-optimal set s, we can construct a sequence of sets $s_1, s_2, \cdots, s_m$, with $s_1 = s^*, s_m = s$ and each intermediate set s_{i+1} changes one arm from s_i into an arm from s. As we start from s^* and going from s_i to s_{i+1}, we always change an arm into some other arm with smaller v. Therefore for any arm $a \in s^* \cap s$, we have $Q_{s_{i+1}}(a) \geq Q_{s_i}(a)$. This implies that $Q_s(a) \geq Q_{s^*}(a)$, which matches Assumption 1.

On the other hand, a model assuming PRISM may not satisfy strong consistent preferences. Consider sets s_1, s_2 containing a_i and s_1', s_2' obtained by replacing a_i by a_j, such that $Q_{s_1}(a_i) > Q_{s_1'}(a_j)$ and $Q_{s_2}(a_i) < Q_{s_2'}(a_j)$, the PRISM (Assumption 1) allows for such case but it does not satisfy Definition 1.

This completes the proof. □

C. Technical Lemmas

Lemma 9. *For B-bounded rewards $X_{a,s}$ (i.e., $X_{a,s} \in [0, B]$) and $\alpha \geq 2$, with probability at least $1 - \frac{2}{T}$, we have the following inequality holds for all arm a_i and all time step $t \in [T]$ simultaneously:*

$$\left| C_i(t) - \sum_{\tau=1}^{t} Q_{s(\tau)}(a_i) \right| \leq B\sqrt{\alpha N_i(t) \log T}.$$

Proof. Recall that $N_i(t)$ is the number of times that arm a_i is played up to time t. Let τ_j be the time step of the j-th pulling of arm a_i. Notice that if a_i is not played at t', then both $C_i(t') = C_i(t'-1)$ and $Q_{s(t')}(a_i) = 0$. Therefore, we have that

$$C_i(t) - \sum_{\tau=1}^{t} Q_{s(\tau)}(a_i) = C_i(\tau_{N_i(t)}) - \sum_{j=1}^{N_i(t)} Q_{s(\tau_j)}(a_i).$$

Consider the quantity

$$D_i(q) = C_i(\tau_q) - \sum_{j=1}^{q} Q_{s(\tau_j)}(a_i),$$

where $q \in \{0, 1, \cdots, N_i(t)\}$. $D_i(0)$ to $D_i(N_i(t))$ is a martingale and $D_i(N_i(t)) = C_i(t) - \sum_{\tau=1}^{t} Q_{s(\tau)}(a_i)$. Consider the bad event $\mathcal{B}_{i,m}(t) := \left\{ N_i(t) = m \quad \text{and} \quad |D_i(N_i(t))| > B\sqrt{\alpha N_i(t) \log T} \right\}$, which can be interpreted as "the desired inequality fails at time step t and the arm a_i is played for m times". By Azuma's inequality, we have

$$P(\mathcal{B}_{i,m}(t)) \leq 2\exp\left(\frac{-2\alpha m B^2 \log T}{mB^2}\right) = \frac{2}{T^{2\alpha}}.$$

Consider event $\mathcal{B}_i(t) := \left\{ |D_i(N_t(t))| > B\sqrt{\alpha N_i(t) \log T} \right\}$, which can be interpreted as "the desired inequality fails at time step t for arm a_i". We have $\mathcal{B}_i(t) = \cup_{m \in [t]} \mathcal{B}_{i,m}(t)$ and, therefore, with a union bound over $m \in [t]$,

$$P(\mathcal{B}_i(t)) \leq \frac{2t}{T^{2\alpha}} \leq \frac{2}{T^{2\alpha-1}}.$$

Further, with a union bound for all $i \in [n]$ and $t \in [T]$, we have that

$$P\left(\exists t \leq T, \exists i \in [n], \text{ s.t. } \left| C_i(t) - \sum_{\tau=1}^{t} Q_{s(\tau)}(a_i) \right| > B\sqrt{\alpha N_i(t) \log T} \right) \leq \frac{2n}{T^{2\alpha-2}}.$$

Notice that for $\alpha = 2$ and $T > n$, the inequality above implies that

$$\left| C_i(t) - \sum_{\tau=1}^{t} Q_{s(\tau)}(a_i) \right| \leq B\sqrt{\alpha N_i(t) \log T}.$$

holds simultaneously for all $i \in [n]$ and $t \in [T]$, with probability at least $1 - \frac{2}{T}$. □

Corollary 10 (UCB is optimistic)**.** *For B-bounded rewards $X_{a,s}$ (i.e., $X_{a,s} \in [0, B]$) and $\alpha \geq 2$, with probability at least $1 - \frac{2}{T}$, we have the following inequality holds for all arm a_i and all time step $t \in [T]$ simultaneously:*

$$\mathit{UCB}_i(t) \geq \frac{\sum_{\tau=1}^{t} Q_{s(\tau)}(a_i)}{N_i(t)}.$$

Corollary 11 (Corollary of Lemma 9)**.** *For B-bounded rewards $X_{a,s}$ (i.e., $X_{a,s} \in [0, B]$) and $\alpha \geq 2$, with probability at least $1 - \frac{2}{T}$, we have the following inequality holds for all arm a_i and all time step $t \in [T]$ simultaneously:*

$$2B\sqrt{\alpha N_i(t) \log T} \geq N_i(t)\mathit{UCB}_i(t) - \sum_{\tau=1}^{t} Q_{s(\tau)}(a_i).$$

Without loss of generality, we assume $s^* = \{a_1, a_2, \cdots, a_k\}$ with $Q_{s^*}(a_1) \geq Q_{s^*}(a_2) \geq \cdots \geq Q_{s^*}(a_k)$. Recall that $\rho(t)$ is monotonically non-increasing by definition. Let time t_l be the last time t that we have $\rho(t) \geq Q_{s^*}(a_l)$ for $l \leq k$, we have the following result. Notice that if there exists an l such that $\rho(T) \geq Q_{s^*}(a_l)$, we have $t_l = T$ by definition.

Lemma 12. *With probability at least $1 - \frac{2}{T}$, for all time steps $t > t_l$, we have $\{a_1, a_2, \cdots, a_l\} \subset s(t)$.*

Proof. The rest of the proof conditions on the event that the inequality in Lemma 9 holds, which happens with probability at least $1 - \frac{2}{T}$.

By definition of $\mathit{UCB}_i(t)$, we have

$$\mathit{UCB}_i(t) = \frac{C_i(t)}{N_i(t)} + \sqrt{\frac{\alpha \log T}{N_i(t)}} \geq \frac{\sum_{\tau=1}^{t} Q_{s(\tau)}(a_i)}{N_i(t)}, \quad \forall a_i \in s^*, \forall t \in [T]$$

Further, by PRISM, we have that $Q_s(a_i) \geq Q_{s^*}(a_i), \forall a_i \in s^*$ and all s contains a_i. Therefore we have $\frac{\sum_{\tau=1}^{t} Q_{s(\tau)}(a_i)}{N_i(t)} \geq Q_{s^*}(a_i)$ and thus $UCB_i(t) \geq Q_{s^*}(a_i), \forall a_i \in s^*, \forall t \in [T]$.

Notice that Lemma 6 states that for all $a_i \notin s(t)$, we have $\rho(t) \geq \mathit{UCB}_i(t)$. It therefore implies $a_i \in s(t)$ if $\mathit{UCB}_i(t) > \rho(t)$. When $\rho(t) < Q_{s^*}(a_l)$ (which happens after t_l by the definition of t_l and the fact that $\rho(t)$ is non-increasing), we have $UCB_i(t) \geq Q_{s^*}(a_i) > \rho(t), \forall i \in [l]$. Therefore we have $a_i \in s(t), \forall i \in [l], \forall t > t_l$.

Note that when $t_l = T$, the Lemma 12 trivially holds true as there is no time step t after T. Therefore, it concludes the proof that with probability at least $1 - \frac{2}{T}$, for all time steps $t > t_l$, we have $\{a_1, a_2, \cdots, a_l\} \subset s(t)$. □

Let $\delta_{ij} \triangleq Q_{s^*}(a_i) - Q_{s^*}(a_j)$. Recall that t_l is the last time step with $\rho(t_l) \geq Q_{s^*}(a_l)$ we have the following result.

Lemma 13. *With probability at least $1 - \frac{2}{T}$, we have the following results hold simultaneously for all $l \in [k]$:*

$$B\sqrt{4\alpha kn\left(t_l - \frac{l}{k}t'_l\right)\log T} \geq \sum_{i=1}^{l} Q_{s^*}(a_i)t_l + (k-l)Q_{s^*}(a_l)t_l - \sum_{i=1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - \sum_{i=1}^{l-1}\delta_{il}(t_i - t'_i) - nQ_{s^*}(a_l),$$

and

$$B\sqrt{4\alpha kn\,(t_l - t'_l)\log T} \geq \sum_{i=k+1}^{n} Q_{s^*}(a_l)N_i(t_l) + \sum_{i=1}^{k} Q_{s^*}(a_i)N_i(t_l) - \sum_{i=1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - nQ_{s^*}(a_l).$$

Proof. The rest of the proof conditions on the event that the inequality in Lemma 9 holds (and therefore the inequalities in Corollary 11 and lemma 12 hold), which happens with probability at least $1 - \frac{2}{T}$.

Proof of the first inequality. Recall that by Lemma 6, we have

$$\mathit{UCB}_i(t) \geq \rho(t)\left(1 - \frac{1}{N_i(t)}\right), \forall a_i \notin s(t),$$

and recall the definition of $\rho'(t) = \min_{a_i \in s(t)} \mathit{UCB}_i(t)$ and $\rho(t) = \min_{\tau \leq t} \rho'(\tau)$, we have

$$\mathit{UCB}_i(t) \geq \rho'(t) \geq \rho(t) \geq \rho(t)\left(1 - \frac{1}{N_i(t)}\right), \forall a_i \in s(t).$$

Therefore, by Corollary 11, for all $i \in [n]$ and all $l \in [k]$, we have

$$\begin{aligned}
2B\sqrt{\alpha N_i(t_l)\log T} \geq & N_i(t_l)\mathit{UCB}_i(t_l) - \sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) \\
\geq & N_i(t_l)\rho(t_l)\left(1 - \frac{1}{N_i(t)}\right) - \sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) \\
\geq & N_i(t_l)Q_{s^*}(a_l) - \sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - Q_{s^*}(a_l), \qquad (1)
\end{aligned}$$

where the last step follows from that t_l is the last time step with $\rho(t_l) \geq Q_{s^*}(a_l)$. Summing up for $i \geq l+1$, we have

$$2B\sum_{i=l+1}^{n}\sqrt{\alpha N_i(t_l)\log T} \geq \sum_{i=l+1}^{n} Q_{s^*}(a_l)N_i(t_l) - \sum_{i=l+1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - nQ_{s^*}(a_l).$$

Notice that for arms a_i with $i \leq l$, we have $\sum_{i=1}^{l} Q_{s^*}(a_i)N_i(t_l) - \sum_{i=1}^{l}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) \leq 0$ as $Q_{s(\tau)}(a_i) \geq Q_{s^*}(a_i)$ for all τ and $i \leq l$, by Assumption 1. Therefore we have

$$2B\sum_{i=l+1}^{n}\sqrt{\alpha N_i(t_l)\log T} \geq \sum_{i=1}^{l} Q_{s^*}(a_i)N_i(t_l) + \sum_{i>l}^{n} Q_{s^*}(a_l)N_i(t_l) - \sum_{i=1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - nQ_{s^*}(a_l).$$

For the first two terms on the right side, we have $\sum_{i=1}^{l} Q_{s^*}(a_i)N_i(t_l) = \sum_{i=1}^{l} Q_{s^*}(a_i)t_l - \sum_{i=1}^{l} Q_{s^*}(a_i)\left(t_l - N_i(t_l)\right)$ and $\sum_{i>l} Q_{s^*}(a_l)N_i(t_l) = Q_{s^*}(a_l)\left[(k-l)t_l + \sum_{i=1}^{l}\left(t_l - N_i(t_l)\right)\right]$. Therefore, we have

$$\begin{aligned}
2B\sum_{i=l+1}^{n}\sqrt{\alpha N_i(t_l)\log T} \geq & \sum_{i=1}^{l} Q_{s^*}(a_i)t_l + (k-l)Q_{s^*}(a_l)t_l - \sum_{i=1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - \sum_{i=1}^{l-1}\delta_{il}\left(t_l - N_i(t_l)\right) - nQ_{s^*}(a_l) \\
\geq & \sum_{i=1}^{l} Q_{s^*}(a_i)t_l + (k-l)Q_{s^*}(a_l)t_l - \sum_{i=1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - \sum_{i=1}^{l-1}\delta_{il}\left(t_i - t_i'\right) - nQ_{s^*}(a_l).
\end{aligned}$$

Recall that t_i' is the number of optimal set played before t_i. Combined with Lemma 12, which states that a_i is always played after t_i, the second inequality above follows from $t_l - N_i(t_l) = t_i - N_i(t_i) \leq t_i - t_i'$. The first inequality in Lemma 13 follows from

$$2B\sum_{i=l+1}^{n}\sqrt{\alpha N_i(t_l)\log T} \leq B\sqrt{4n\sum_{i=l+1}^{n}\alpha N_i(t_l)\log T} \leq B\sqrt{4\alpha kn\left(t_l - \frac{l}{k}t_l'\right)\log T}.$$

Proof of the second inequality. We reuse the following inequality that we proved at Equation (1), for all $i \in [n]$ and all $l \in [k]$, we have:

$$2B\sqrt{\alpha N_i(t_l)\log T} \geq N_i(t_l)Q_{s^*}(a_l) - \sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - Q_{s^*}(a_l).$$

Now, instead of summing over $i \geq l+1$, we sum over $i > k$ and have

$$\sum_{i=k+1}^{n} 2B\sqrt{\alpha N_i(t_l)\log T} \geq \sum_{i=k+1}^{n} N_i(t_l)Q_{s^*}(a_l) - \sum_{i=k+1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - nQ_{s^*}(a_l).$$

Notice that for arms a_i with $i \leq k$, we have $\sum_{i=1}^{k} Q_{s^*}(a_i)N_i(t_l) - \sum_{i=1}^{k}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) \leq 0$ as $Q_{s(\tau)}(a_i) \geq Q_{s^*}(a_i)$ for all τ and $i \leq k$, by Assumption 1. Therefore we have

$$\sum_{i=k+1}^{n} 2B\sqrt{\alpha N_i(t_l)\log T} \geq \sum_{i=k+1}^{n} N_i(t_l)Q_{s^*}(a_l) + \sum_{i=1}^{k} Q_{s^*}(a_i)N_i(t_l) - \sum_{i=1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - nQ_{s^*}(a_l).$$

The first inequality in Lemma 13 follows by

$$2B\sum_{i=k+1}^{n}\sqrt{\alpha N_i(t_l)\log T} \leq B\sqrt{4n\sum_{i=k+1}^{n}\alpha N_i(t_l)\log T} \leq B\sqrt{4\alpha kn\,(t_l - t_l')\log T}.$$

This completes the proof. □

Recall that we assumed $a_1, \cdots, a_k$ all belong to s^*, with $Q_{s^*}(a_1) \geq Q_{s^*}(a_2) \geq \cdots \geq Q_{s^*}(a_k)$. Recall $\delta_{ij} = Q_{s^*}(a_i) - Q_{s^*}(a_j)$ and $\Delta_l = \sum_{i=l}^{k}\delta_{li}$.

Lemma 14. *Let $\sigma_{ij} = \frac{4\delta_{ij}(\Delta_j+\epsilon)}{(\Delta_i+\epsilon)^2}$, we have*

$$\sum_{j=i}^{k}\sigma_{ij} \leq 2,\ \forall i \leq k, \forall \epsilon \geq 0.$$

Proof. Expanding the summation, we have

$$\sum_{j=i}^{k}\sigma_{ij} = \sum_{j=i}^{k}\frac{4\delta_{ij}(\Delta_j+\epsilon)}{(\Delta_i+\epsilon)^2} = 4\sum_{j=i}^{k}\frac{\delta_{ij}}{\Delta_i+\epsilon}\left(\sum_{m=j}^{k}\frac{\delta_{jm}}{\Delta_i+\epsilon} + \frac{\epsilon}{\Delta_i+\epsilon}\right).$$

Note that

$$\sum_{m=j}^{k}\delta_{jm} + \sum_{m=i}^{j}\delta_{im} + \epsilon \leq \Delta_i + \epsilon \implies \sum_{m=j}^{k}\frac{\delta_{jm}}{\Delta_i+\epsilon} + \frac{\epsilon}{\Delta_i+\epsilon} \leq 1 - \sum_{m=i}^{j}\frac{\delta_{im}}{\Delta_i+\epsilon}.$$

For brevity, let $x_m = \frac{\delta_{im}}{\Delta_i+\epsilon}$, we have $\sum_{m=i}^{k} x_m \leq 1$ and

$$\sum_{j=i}^{k}\sigma_{ij} \leq 4\sum_{j=i}^{k} x_j(1 - \sum_{m=i}^{j} x_m) \leq 4\int_0^1 (1-x)dx \leq 2.$$

This completes the proof. □

Follow the definition of $\sigma_{ij} = \frac{4\delta_{ij}(\Delta_j+\epsilon)}{(\Delta_i+\epsilon)^2}$ in Lemma 14. We have the following result.

Lemma 15. *For any $1 \leq i < j \leq k$, define function $f(i,j) = 0.4\sigma_{ij} + \sum_{m=i+1}^{j-1} 0.4\sigma_{im}f(m,j)$, we have*

1. $f(i,j) = 0.4\sigma_{ij} + \sum_{m=i+1}^{j-1} 0.4 f(i,m)\sigma_{mj}, \quad \forall 1 \le i < j \le k$
2. $f(i,j) \le 1, \quad \forall 1 \le i < j \le k$

Proof. We first prove the first part. Let $\Pi(i,j)$ be the power set of $\{i, i+1, \cdots, j-1, j\}$. Let $\Gamma(i,j) = \{x | x \in \Pi(i,j), i \in x, j \in x\}$. Further, for $x \in \Gamma(i,j)$defining

$$g(x) = \sigma_{x_1 x_2} \cdot \sigma_{x_2 x_3} \cdots \sigma_{x_{|x|-1} x_{|x|}}.$$

For example, for $x = \{2,3,5,7\}$, we have $g(x) = \sigma_{23} \cdot \sigma_{35} \cdot \sigma_{57}$.

We first show by induction that $f(i,j) = \sum_{x \in \Gamma(i,j)} 0.4^{|x|-1} g(x)$. For base case $j = i+1$, we have $f(i,j) = 0.4\sigma_{ij}$. Now suppose $f(i,j) = \sum_{x \in \Gamma(i,j)} 0.4^{|x|-1} g(x)$ holds for $j - i \le c$, we proceed to prove it holds for $j - i = c + 1$.

Take $j = i + c + 1$, we have

$$\begin{aligned}
f(i,j) &= 0.4\sigma_{ij} + \sum_{m=i+1}^{j-1} 0.4\sigma_{im} f(mj) \\
&= 0.4\sigma_{ij} + \sum_{m=i+1}^{j-1} \left(0.4\sigma_{im} \sum_{x \in \Gamma(m,j)} 0.4^{|x|-1} g(x) \right) \\
&= 0.4\sigma_{ij} + \sum_{m=i+1}^{j-1} \sum_{x \in \Gamma(m,j)} 0.4^{|x|} \sigma_{im} g(x) \\
&= \sum_{x \in \Gamma(i,j)} 0.4^{|x|-1} g(x),
\end{aligned}$$

where the last steps holds as a element x belongs to $\Gamma(i,j)$ if and only if x is one of the following two cases:

1. $x = \{i, j\}$,
2. $x = \{i, m, \cdots, j\}$ for some $m \in [i+1, j-1]$, and $\{m, \cdots, j\} \in \Gamma(m,j)$ by definition.

Therefore, we conclude via induction that

$$f(i,j) = \sum_{x \in \Gamma(i,j)} 0.4^{|x|-1} g(x). \tag{2}$$

Further, Equation (2) implies the first equation in Lemma 15, since a element x belongs to $\Gamma(i,j)$ if and only if x is one of the following two cases:

1. $x = \{i, j\}$,
2. $x = \{i, \cdots, m, j\}$ for some $m \in [i+1, j-1]$, and $\{i, \cdots, m\} \in \Gamma(i,m)$ by definition.

For the second part of the proof, we prove by induction. For the base case $j - i = 1$, we have $f(i,j) = 0.4\sigma_{ij} \le 0.4 \times 4 \times 0.5 \le 1$. Now, suppose that the inequality holds for any i, j with $j - i = c$, then for any $i, j \le k$ with $j - i = c + 1$, we have

$$f(i,j) \le 0.4\sigma_{ij} + \sum_{m=i+1}^{j-1} 0.4\sigma_{im} = 0.4 \sum_{m=i+1}^{j} \sigma_{im} \le 0.4 \sum_{m=i}^{k} \sigma_{im} \le 0.8.$$

The last inequality follows from Lemma 14, which states $\sum_{m=i}^{k} \sigma_{im} \le 2$. □

D. Proof for Section 4

D.1. Proof of Lemma 5

Proof. The rest of the proof conditions on the event that the inequality in Lemma 9 holds (and therefore the inequality in Corollary 10 holds), which happens with probability at least $1 - \frac{2}{T}$.

By Corollary 10, we have

$$UCB_i(t) \geq \frac{\sum_{\tau=1}^{t} Q_{s(\tau)}(a_i)}{N_i(t)},$$

for all a_i and all $t \in [T]$. By Assumption 1, we have that $Q_s(a_i) \geq Q_{s^*}(a_i)$ for all $a_i \in s^*$ and all s containing a_i. Therefore $\frac{\sum_{\tau=1}^{t} Q_{s(\tau)}(a_i)}{N_i(t)} \geq Q_{s^*}(a_i)$. This completes the proof that with probability at least $1 - \frac{2}{T}$, we have $UCB_i(t) \geq Q_{s^*}(a_i)$ simultaneously for all $a_i \in s^*$ and $t \in [T]$. □

D.2. Proof of Lemma 7

Proof. The rest of the proof conditions on the event that the inequality in Lemma 9 holds (and therefore the inequalities in Corollary 11 and lemmas 12 and 13 hold), which happens with probability at least $1 - \frac{2}{T}$.

Define $\delta_{ij} = Q_{s^*}(a_i) - Q_{s^*}(a_j)$, $\Delta_l = \sum_{i=l}^{k} \delta_{li}$. Define t_l to be the last time step with $\rho(t_l) \geq Q_{s^*}(a_l)$. Denote t'_l to be the number of times $s(t) = s^*$ for $t \leq t_l$. Note that if $\rho(T) \geq Q_{s^*}(a_l)$, then $t_l = T$ by definition.

Case I: $\Delta_l \geq \frac{\epsilon}{10}$.

By the first inequality of Lemma 13, we have

$$B\sqrt{4\alpha kn\left(t_l - \frac{l}{k}t'_l\right)\log T} \geq \sum_{i=1}^{l} Q_{s^*}(a_i)t_l + (k-l)Q_{s^*}(a_l)t_l - \sum_{i=1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - \sum_{i=1}^{l-1}\delta_{il}(t_i - t'_i) - nQ_{s^*}(a_l).$$

Note that

$$\sum_{i=1}^{l} Q_{s^*}(a_i) + (k-l)Q_{s^*}(a_l) - \sum_{i=1}^{k} Q_{s^*}(a_i) = \sum_{i=l}^{k}\delta_{li} = \Delta_l.$$

By the fact $\sum_{i=1}^{k} Q_{s^*}(a_i) - \sum_{i=1}^{n} Q_{s(t)}(a_i) \geq \epsilon$ for all suboptimal set $s(t)$, we have

$$\begin{aligned} B\sqrt{4\alpha kn\left(t_l - \frac{l}{k}t'_l\right)\log T} &\geq \Delta_l t_l + \epsilon\left(t_l - t'_l\right) - \sum_{i=1}^{l-1}\delta_{il}\left(t_i - t'_i\right) - nQ_{s^*}(a_l) \\ &\geq (\Delta_l + \epsilon)\left(t_l - \frac{l}{k}t'_l\right) - \frac{\epsilon(k-l) - \Delta_l l}{k}t'_l - \sum_{i=1}^{l-1}\delta_{il}(t_i - t'_i) - nQ_{s^*}(a_l). \end{aligned} \tag{3}$$

Next, we prove by mathematical induction that $(t_i - t'_i) \leq \left(3.08 + 10\sum_{j=1}^{i-1} f(j,i)\right)\frac{4\alpha B^2 kn\log T}{(\Delta_i+\epsilon)^2}$, where the function $f(i,j)$ is defined in Lemma 15. For notation simplicity, we write $t_i - t'_i$ in the following form

$$t_i - t'_i \leq c_i\frac{4\alpha B^2 kn\log T}{(\Delta_i+\epsilon)^2},$$

and proceed to bound c_i. With the new convention, we can rewrite Equation (3) as

$$B\sqrt{4\alpha kn\left(t_l - \frac{l}{k}t'_l\right)\log T} \geq (\Delta_l+\epsilon)\left(t_l - \frac{l}{k}t'_l\right) - \frac{\epsilon(k-l) - \Delta_l l}{k}t'_l - \sum_{i=1}^{l-1}\delta_{il}c_i\frac{4\alpha B^2 kn\log T}{(\Delta_i+\epsilon)^2} - nQ_{s^*}(a_l). \tag{4}$$

For $\log T \geq 2.5$, with the fact $k \geq \Delta_l + \epsilon, 1 \geq Q_{s^*}(a_l), \alpha \geq 2$, we have $4n(\Delta_l+\epsilon)Q_{s^*}(a_l) \leq 0.8\alpha kn \log T$ and therefore solving Equation (4) for the bound for $\sqrt{t_l - \frac{l}{k}t'_l}$, we have

$$\sqrt{t_l - \frac{l}{k}t'_l} \leq \frac{1}{2}\left(1 + \sqrt{1.2 + \sum_{i=1}^{l-1} \frac{4\delta_{il}\left(\Delta_l+\epsilon\right)}{\left(\Delta_i+\epsilon\right)^2}c_i + \frac{4\left(\Delta_l+\epsilon\right)\frac{\epsilon(k-l)-\Delta_l l}{k}}{4\alpha B^2 kn\log T}t'_l}\right)\frac{B\sqrt{4\alpha kn\log T}}{\Delta_l+\epsilon}.$$

Further, define $\sigma_{il} = \frac{4\delta_{il}(\Delta_l+\epsilon)}{(\Delta_i+\epsilon)^2}$, we have

$$\sqrt{t_l - \frac{l}{k}t'_l} \leq \frac{1}{2}\left(1 + \sqrt{1.2 + \sum_{i=1}^{l-1} \sigma_{il}c_i + \frac{4\left(\Delta_l+\epsilon\right)\frac{\epsilon(k-l)-\Delta_l l}{k}}{4\alpha B^2 kn\log T}t'_l}\right)\frac{B\sqrt{4\alpha kn\log T}}{\Delta_l+\epsilon}.$$

By the fact $(1+a)^2 \leq 1.1a^2 + 11$ for any real number a, we have

$$t_l - \frac{l}{k}t'_l \leq \frac{1}{4}\left(11 + 1.32 + 1.1\sum_{i=1}^{l-1}\sigma_{il}c_i\right)\frac{4\alpha B^2 kn\log T}{\left(\Delta_l+\epsilon\right)^2} + 1.1\frac{\epsilon(k-l)-\Delta_l l}{k\left(\Delta_l+\epsilon\right)}t'_l.$$

Since $\Delta_l \geq \frac{\epsilon}{10}$, we have $\frac{1.1\epsilon k - 1.1(\Delta_l+\epsilon)l}{(\Delta_l+\epsilon)k} \leq \frac{(\Delta_l+\epsilon)k-(\Delta_l+\epsilon)l}{(\Delta_l+\epsilon)k} = \frac{k-l}{k}$. Therefore we have

$$\begin{aligned} t_l - \frac{l}{k}t'_l &\leq \left(3.08 + 0.275\sum_{i=1}^{l-1}\sigma_{il}c_i\right)\frac{4\alpha B^2 kn\log T}{\left(\Delta_l+\epsilon\right)^2} + \frac{k-l}{k}t'_l \\ \implies t_l - t'_l &\leq \left(3.08 + 0.275\sum_{i=1}^{l-1}\sigma_{il}c_i\right)\frac{4\alpha B^2 kn\log T}{\left(\Delta_l+\epsilon\right)^2}. \end{aligned}$$

Plug in the convention of c_l, we have

$$c_l \leq 3.08 + 0.275\sum_{i=1}^{l-1}\sigma_{il}c_i.$$

Now we proceed to show $c_i \leq 3.08 + 10\sum_{j=1}^{i-1} f(j,i)$ by induction. For base case $i = 1, 2$, we have

$$c_1 \leq 3.08, \quad c_2 \leq 3.08 + 0.275\sigma_{12}c_1 \leq 3.08 + \sigma_{12} \leq 0.308 + 10f(1,2).$$

Next, suppose $c_i \leq 3.08 + 10\sum_{j=1}^{i-1} f(j,i)$ holds for all $i \leq l-1$, for $i = l$ we have

$$\begin{aligned} c_l &\leq 3.08 + 0.275\sum_{i=1}^{l-1}\sigma_{il}c_i \leq 3.08 + 0.275\sum_{i=1}^{l-1}\left[3.08 + 10\sum_{j=1}^{i-1}f(j,i)\right]\sigma_{il} \\ &\leq 3.08 + \sum_{i=1}^{l-1}\left[2.75\sigma_{il} + 2.75\sum_{j=1}^{i-1}f(j,i)\sigma_{il}\right] \\ &\leq 3.08 + \sum_{j=1}^{l-1}\sum_{i=j+1}^{l-1}2.75f(j,i)\sigma_{il} + \sum_{i=1}^{l-1}2.75\sigma_{il} \\ &\leq 3.08 + \sum_{j=1}^{l-1}\left[2.75\sigma_{jl} + \sum_{i=j+1}^{l-1}2.75f(j,i)\sigma_{il}\right] \\ &\leq 3.08 + 10\sum_{j=1}^{l-1}f(j,l). \end{aligned}$$

The last inequality follows from the first equation in Lemma 15: $f(i,j) = 0.4\sigma_{ij} + \sum_{m=i+1}^{j-1} 0.4 f(i,m)\sigma_{mj}$. This completes induction.

Combining with the second inequality in Lemma 15 which shows that $f(i,l) \le 1, \forall i < l \le k$, we have $c_l \le 10l$. It therefore implies that $t_l - t_l' \le \frac{40\alpha B^2 lkn \log T}{(\Delta_l+\epsilon)^2}$. This completes the proof of the first case in Lemma 7.

Case II: $\Delta_l < \frac{\epsilon}{10}$**.**

Denote l' to be the largest i with $\Delta_i \ge \epsilon/10$, and let $l' = 0$ if $\Delta_i < \epsilon/10$ for all $i \in [k]$. By definition, we know $l > l'$. Applying the second inequality in Lemma 13, we have that

$$\begin{aligned}
2B\sqrt{\alpha kn\,(t_l - t_l')\log T} &\ge \sum_{i=k+1}^{n} Q_{s^*}(a_l)N_i(t_l) + \sum_{i=1}^{k} Q_{s^*}(a_i)N_i(t_l) - \sum_{i=1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - nQ_{s^*}(a_l) \\
&\ge \sum_{i=1}^{k} Q_{s^*}(a_i)t_l - \sum_{i=1}^{l-1}\left(Q_{s^*}(a_i) - Q_{s^*}(a_l)\right)\left(t_l - N_i(t_l)\right) - \sum_{i=1}^{n}\sum_{\tau=1}^{t_l} Q_{s(\tau)}(a_i) - nQ_{s^*}(a_l) \\
&\ge \epsilon\,(t_l - t_l') - \sum_{i=1}^{l-1}\delta_{il}(t_l - N_i(t_l)) - nQ_{s^*}(a_l) \\
&\ge \epsilon\,(t_l - t_l') - \sum_{i=1}^{l-1}\delta_{il}(t_i - t_i') - nQ_{s^*}(a_l). \qquad (5)
\end{aligned}$$

Recall that t_i' is the number of optimal set played before t_i. Combined with Lemma 12, which states that a_i is always played after t_i, the last inequality above follows from $t_l - N_i(t_l) = t_i - N_i(t_i) \le t_i - t_i'$.

Next, we prove by mathematical induction that $(t_i - t_i') \le \left(3.08 + 10\sum_{j=1}^{i-1} f(j,i)\right)\frac{4\alpha B^2 kn\log T}{\epsilon^2}$ for all $i > l'$, where the function $f(i,j)$ is defined in Lemma 15. For notation simplicity, we write $t_i - t_i'$ for $i > l'$ in the following form

$$t_i - t_i' \le d_i \frac{4\alpha B^2 kn\log T}{\epsilon^2},$$

and proceed to bound d_i. With this convention, we can rewrite Equation (5) as

$$2B\sqrt{\alpha kn\,(t_l - t_l')\log T} \ge \epsilon\,(t_l - t_l') - \sum_{i=1}^{l'}\delta_{il}c_i\frac{4\alpha B^2 kn\log T}{(\Delta_i+\epsilon)^2} - \sum_{i=l'+1}^{l-1}\delta_{il}d_i\frac{4\alpha B^2 kn\log T}{\epsilon^2} - nQ_{s^*}(a_l)$$

For $\log T \ge 2.5$, with the fact $k \ge \Delta_l + \epsilon, 1 \ge Q_{s^*}(a_l), \alpha \ge 2$, we have $4n(\Delta_l+\epsilon)Q_{s^*}(a_l) \le 0.8\alpha kn\log T$ and therefore solving Equation (5) for the bound for $\sqrt{t_l - t_l'}$, we have

$$\begin{aligned}
\sqrt{t_l - t_l'} &\le \frac{B\sqrt{4\alpha kn\log T}\left(1 + \sqrt{1.2 + \sum_{i=1}^{l'}\frac{4\delta_{il}\epsilon}{(\Delta_i+\epsilon)^2}c_i + \sum_{i=l'+1}^{l-1}\frac{4\delta_{il}\epsilon}{\epsilon^2}d_i}\right)}{2\epsilon} \\
&\le \frac{B\sqrt{4\alpha kn\log T}\left(1 + \sqrt{1.2 + \sum_{i=1}^{l'}\frac{4\delta_{il}(\Delta_l+\epsilon)}{(\Delta_i+\epsilon)^2}c_i + 1.21\sum_{i=l'+1}^{l-1}\frac{4\delta_{il}(\Delta_i+\epsilon)}{(\Delta_i+\epsilon)^2}d_i}\right)}{2\epsilon}.
\end{aligned}$$

The second inequality follows from $\frac{(\Delta_i+\epsilon)^2}{\epsilon^2} \le 1.21$ as $\Delta_i < \frac{\epsilon}{10}$ for all $i > l'$. Define $\sigma_{il} = \frac{4\delta_{il}(\Delta_l+\epsilon)}{(\Delta_i+\epsilon)^2}$, with the convention of d_l, we have

$$\sqrt{d_l} \le \frac{1}{2}\left(1 + \sqrt{1.2 + \sum_{i=1}^{l'}\sigma_{il}c_i + 1.21\sum_{i=l'+1}^{l-1}\sigma_{il}d_i}\right).$$

Again use the fact that $(1+a)^2 \le 11 + 1.1a^2$, we have

$$\begin{aligned} d_l &\le \frac{1}{4}\left(11 + 1.32 + 1.1\sum_{i=1}^{l'}\sigma_{il}c_i + 1.331\sum_{i=l'+1}^{l-1}\sigma_{il}d_i\right) \\ &\le 3.08 + 0.275\sum_{i=1}^{l'}\sigma_{il}c_i + 0.34\sum_{i=l'+1}^{l-1}\sigma_{il}d_i. \end{aligned}$$

We next prove by induction that $d_i \le 3.08 + 10\sum_{j=1}^{i-1} f(j,i)$ for all $i > l'$. For the base case, $i = l'+1$, we immediately have $d_i \le 3.08 + 0.275\sum_{j=1}^{i-1}\sigma_{ji}c_j = 3.08 + 10\sum_{j=1}^{i-1} f(j,i)$, which follows from the proof in **Case I**.

Next, assuming that $d_i \le 3.08 + 10\sum_{j=1}^{i-1} f(j,i)$ holds for $i \le l-1$, then for $i = l$ we have

$$\begin{aligned} d_l &\le 3.08 + 0.275\sum_{i=1}^{l'}\sigma_{il}c_i + 0.34\sum_{i=l'+1}^{l-1}\sigma_{il}d_i \\ &\le 3.08 + 0.275\sum_{i=1}^{l'}\left[3.08 + 10\sum_{j=1}^{i-1} f(j,i)\right]\sigma_{il} + 0.34\sum_{i=l'+1}^{l-1}\left[3.08 + 10\sum_{j=1}^{i-1} f(j,i)\right]\sigma_{il} \\ &\le 3.08 + \sum_{i=1}^{l-1}\left[3.4\sigma_{il} + 3.4\sum_{j=1}^{i-1} f(j,i)\sigma_{il}\right] \\ &\le 3.08 + \sum_{j=1}^{l-1}\sum_{i=j+1}^{l-1} 3.4 f(j,i)\sigma_{il} + \sum_{i=1}^{l-1} 3.4\sigma_{il} \\ &\le 3.08 + \sum_{j=1}^{l-1}\left[3.4\sigma_{jl} + \sum_{i=j+1}^{l-1} 3.4 f(j,i)\sigma_{il}\right] \\ &\le 3.08 + 10\sum_{j=1}^{l-1} f(j,l). \end{aligned}$$

The last inequality follows from the first equation in Lemma 15: $f(i,j) = 0.4\sigma_{ij} + \sum_{m=i+1}^{j-1} 0.4 f(i,m)\sigma_{mj}$. This completes induction.

Combining with the second inequality in Lemma 15 which shows that $f(i,l) \le 1, \forall i < l \le k$, we have $d_l \le 10l$. It therefore implies that $t_l - t_l' \le \frac{40\alpha B^2 lkn\log T}{\epsilon^2}$. This completes the proof of the second case in Lemma 7. □

D.3. Proof of Lemma 8

Proof. The rest of the proof conditions on the event that the inequality in Lemma 9 holds (and therefore inequalities in Lemmas 5 and 13 hold), which happens with probability at least $1 - \frac{2}{T}$.

Recall that for $a_l \in s^*$, t_l is the last time step with $\rho(t_l) \ge Q_{s^*}(a_l)$ and t_l' is the number of times $s(t) = s^*$ for $t \le t_l$. From Lemma 5, we know that the arms $a_i \in s^*$ all have $UCB_i(t) \ge Q_{s^*}(a_k)$ for all $t \in [T]$. Therefore, we have $\rho'(t) \ge Q_{s^*}(a_k)$ for all t and thus $\rho(T) \ge Q_{s^*}(a_k)$. It implies that $t_k = T$ and $R(T) = R(t_k)$. Plug in the first inequality in Lemma 13 with $l = k$, we have

$$B\sqrt{4\alpha kn\,(t_k - t_k')\log T} \ge \sum_{i=1}^{k} Q_{s^*}(a_i)t_k - \sum_{i=1}^{n}\sum_{\tau=1}^{t_k} Q_{s(\tau)}(a_i) - \sum_{i=1}^{k-1}\delta_{ik}(t_i - t_i') - nQ_{s^*}(a_k).$$

Note that $R(t_k) = \sum_{i=1}^{k} Q_{s^*}(a_i)t_k - \sum_{i=1}^{n}\sum_{\tau=1}^{t_k} Q_{s(\tau)}(a_i)$, rearranging the terms, we have

$$R(T) = R(t_k) \le B\sqrt{4\alpha k n\,(t_k - t_k')\log T} + \sum_{l=1}^{k-1} \delta_{lk}\,(t_l - t_l') + nQ_{s^*}(a_k).$$

This completes the proof. □

D.4. Proof of Theorem 3

Now we are ready to prove Theorem 3.

Proof. We first consider when the inequality in Lemma 9 does not hold for some t and arm a_i. Since this happens with probability at most $2/T$ and the regret is trivially bounded by T. Therefore it induces at most 2 regret in expectation.

The rest of the proof conditions on the event that the inequality in Lemma 9 holds (and therefore the inequalities in Lemmas 7 and 8 hold), which happens with probability at least $1 - \frac{2}{T}$.

We first prove the gap-dependent regret bound. Combining Lemmas 7 and 8, we have

$$\begin{aligned}
R(T) \le& 2B\sqrt{\alpha k n(t_k - t_k')\log T} + \sum_{l=1}^{k-1} \delta_{lk}\,(t_l - t_l') + nQ_{s^*}(a_k)\\
\le& \frac{14\alpha B^2 k^{\frac{3}{2}} n\log T}{\epsilon} + \sum_{l=1}^{k-1} \delta_{lk}\,(t_l - t_l') + nQ_{s^*}(a_k)\\
\overset{(a)}{\le}& \frac{14\alpha B^2 k^{\frac{3}{2}} n\log T}{\epsilon} + \sum_{l=1}^{k-1} \frac{40\alpha B^2 lkn\log T}{\epsilon} + n\\
\le& \frac{35\alpha B^2 k^3 n\log T}{\epsilon}.
\end{aligned}$$

Recall that $\delta_{ik} = Q_{s^*}(a_i) - Q_{s^*}(a_k)$, and $\Delta_l = \sum_{i=l}^{k}\delta_{li}$. The inequality (a) used the fact that

$$\begin{aligned}
\delta_{lk}\cdot(t_l - t_l') \le \delta_{lk}\cdot\frac{40\alpha B^2 lkn\log T}{(\Delta_l+\epsilon)^2} \le \frac{40\alpha B^2 lkn\log T}{\epsilon}, \quad \forall \Delta_l \ge \epsilon/10; \textit{and}\\
\delta_{lk}\cdot(t_l - t_l') \le \delta_{lk}\cdot\frac{40\alpha B^2 lkn\log T}{\epsilon^2} \le \frac{40\alpha B^2 lkn\log T}{\epsilon}, \quad \forall \Delta_l < \epsilon/10.
\end{aligned}$$

This completes the proof of the gap-dependent regret bound $O\left(\frac{B^2k^3 n\log T}{\epsilon}\right)$.

For the gap-independent part, recall that $\delta_{ik} = Q_{s^*}(a_i) - Q_{s^*}(a_k)$, and $\Delta_l = \sum_{i=l}^{k}\delta_{li}$.

If $\Delta_1 + \epsilon < 10kB\sqrt{\frac{\alpha n\log T}{T}}$, from Lemma 8, we have

$$\begin{aligned}
R(T) \le& 2B\sqrt{\alpha k n T\log T} + \sum_{i=1}^{k}\delta_{ik}T + n\\
\le& 2B\sqrt{\alpha k n T\log T} + \sum_{i=1}^{k}\Delta_1 T + n = O\left(Bk^2\sqrt{nT\log T}\right),
\end{aligned}$$

where we used the fact that $t_l - t_l' \le t_l \le T$ for all $l \in [k]$, and $\delta_{ik} \le \Delta_1$ for all $i \in [k]$.

On the other hand, if $\Delta_1 + \epsilon < 10kB\sqrt{\frac{\alpha n\log T}{T}}$, let m denote the largest $i \in [1,k]$ such that $\Delta_i + \epsilon \ge 10kB\sqrt{\frac{\alpha n\log T}{T}}$. To invoke Lemma 7, we need to discuss the relationship between Δ_m and $\epsilon/10$

(a) If $\Delta_m \geq \epsilon/10$, combining Lemmas 7 and 8, we have

$$R(T) \leq 2B\sqrt{\alpha knT\log T} + \sum_{l=1}^{m}\frac{40\alpha B^2 lkn\log T}{\Delta_m+\epsilon} + \sum_{i=m+1}^{k}\delta_{ik}T + n$$
$$\leq 2B\sqrt{\alpha knT\log T} + \frac{20\alpha B^2 k^3 n\log T}{\Delta_m+\epsilon} + \sum_{i=m+1}^{k}\delta_{ik}T + n.$$

If $m = k$, we do not have the third term. Otherwise, by definition of Δ_{m+1}, we have $\delta_{ik} \leq \Delta_{m+1}, \forall i \geq m+1$. Therefore, we have

$$R(t) \leq 2B\sqrt{\alpha knT\log T} + \frac{20\alpha B^2 k^3 n\log T}{\Delta_m+\epsilon} + (k-m)\Delta_{m+1}T + n.$$

With $\Delta_m + \epsilon \geq 10kB\sqrt{\frac{\alpha n\log T}{T}} \geq \Delta_{m+1} + \epsilon$ and the fact that $T > n$, we have the $O(Bk^2\sqrt{nT\log T})$ gap-independent regret bound.

(b) If $\Delta_m < \epsilon/10$, $\Delta_m + \epsilon \geq 10kB\sqrt{\frac{\alpha n\log T}{T}}$ implies that $\epsilon \geq 9kB\sqrt{\frac{\alpha n\log T}{T}}$. Combining Lemmas 7 and 8, we have

$$R(T) \leq 2B\sqrt{\alpha knT\log T} + \sum_{l=1}^{m}\frac{40\alpha B^2 lkn\log T}{\epsilon} + \sum_{i=m+1}^{k}\delta_{ik}T + n$$
$$\leq 2B\sqrt{\alpha knT\log T} + \frac{20\alpha B^2 k^3 n\log T}{\epsilon} + \sum_{i=m+1}^{k}\delta_{ik}T + n.$$

If $m = k$, we do not have the third term. Otherwise, by definition of Δ_{m+1}, we have $\delta_{ik} \leq \Delta_{m+1}, \forall i \geq m+1$. Therefore, we have

$$R(t) \leq 2B\sqrt{\alpha knT\log T} + \frac{20\alpha B^2 k^3 n\log T}{\epsilon} + (k-m)\Delta_{m+1}T + n.$$

With $\epsilon \geq 9kB\sqrt{\frac{\alpha n\log T}{T}}$ and $10kB\sqrt{\frac{\alpha n\log T}{T}} \geq \Delta_{m+1} + \epsilon$ and the fact that $T > n$, we have the $O(Bk^2\sqrt{nT\log T})$ gap-independent regret bound.

Combining all the cases completes the proof. □

E. Experiment Setup

E.1. Multinomial Logit

In this environment, the reward is generated according to a multinomial logit model

$$Q_{s(t)}(a_i) = \frac{v_i}{1+\sum_{a_i\in s(t)} v_i}, \quad Q_{s(t)}(a_0) = \frac{1}{1+\sum_{a_i\in s(t)} v_i}.$$

where v_i is the value associated with each arm a_i, determining the reward probability. In this experiment, we set $v_i = 1 - 0.04i$ with $i \in [20]$. The size of set is set to $k = 10$, and the optimal set is s^* is composed by arms from a_1 to a_{10}. The regret of set $s(t)$ is given by

$$reg(s(t)) = \sum_{a_i\in s^*} Q_{s^*}(a_i) - \sum_{a_i\in s(t)} Q_{s(t)}(a_i).$$

E.2. Random Utility Model

In this environment, for an set $s(t)$ at time step t, each arm $a_i \in s(t)$ will independently draw a Gaussian distributed random variable $U_i \sim \mathcal{N}(\mu_i, 1)$, where μ_i is the mean associated with each arm a_i. Along with that a_0 will draw a $U_0 \sim \mathcal{N}(2, 1)$. The arm a_i (including a_0) with highest U_i will receive reward. Thus we have the probability of a_i getting reward as

$$Q_{s(t)}(a_i) = \mathbb{P}(U_i = \max_{a_j\in s(t)\cup\{a_0\}} U_j).$$

Here, we set $\mu_i = 1 - 0.04i$ with $i \in [20]$. The size of set is set to $k = 5$, and the optimal set s^* is composed by the arms from a_1 to a_5. For the convenience of computation, the regret of set $s(t)$ is defined slightly different as

$$reg(s(t)) = \sum_{a_i \in s^*} \mu_i - \sum_{a_i \in s(t)} \mu_i.$$

Once $s(t)$ recovers the optimal set s^*, which maximizes the probability of $s(t)$ receiving reward, we will have this regret $reg(s(t)) = 0$.

E.3. Preference Matrix

In this environment, the probability of one arm a_i getting reward is fully specified by a preference matrix. For ease of representation, we set the number of arms to $n = 10$ and the size of set to $k = 2$. The total number of sets is 45, much fewer than the previous two environments. However, with a specially designed preference matrix (including the loop in preference, etc), the environment turns out to be the hardest.

We set M to be the preference matrix with $M_{i,j} = Q_{\{a_i,a_j\}}(a_i) - Q_{\{a_i,a_j\}}(a_j)$. We set the optimal set to be $s^* = \{a_1, a_2\}$ with $Q_{\{a_1,a_2\}}(a_1) + Q_{\{a_1,a_2\}}(a_2) = 0.92$. For all other sets s which are sub-optimal, we set $Q_{\{a_i,a_j\}}(a_i) + Q_{\{a_i,a_j\}}(a_j) = 0.9$. The preference matrix M is given in Table 2.

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}
a_1	–	0.02	0.05	0.1	0.1	0.2	0.25	0.3	0.3	0.3
a_2	-0.02	–	0.05	0.1	0.1	0.2	0.25	0.3	0.3	0.3
a_3	-0.05	-0.05	–	0.45	0.45	0.45	0.45	0.45	0.45	0.45
a_4	-0.1	-0.1	-0.45	–	-0.3	0.3	0	0	0	0
a_5	-0.1	-0.1	-0.45	0.3	–	-0.3	0	0	0	0
a_6	-0.2	-0.2	-0.45	-0.3	0.3	–	0	0	0	0
a_7	-0.25	-0.25	-0.45	0	0	0	–	0	0	0
a_8	-0.3	-0.3	-0.45	0	0	0	0	–	0	0
a_9	-0.3	-0.3	-0.45	0	0	0	0	0	–	0
a_{10}	-0.3	-0.3	-0.45	0	0	0	0	0	0	–

Table 2. Preference Matrix M

We can see that when a_3 pairs with any other sub-optimal arm, it will have a higher chance of getting reward than a_1 and a_2. It makes a_3 the seemingly best single arm. Also note that when a_4 pairs with a_5, a_5 will have a higher chance of getting reward. Similarly, a_6 will win over a_5 and a_4 will win over a_6. The preference therefore forms a loop among a_4, a_5, a_6.

The regret of $\{a_i, a_j\}$ is given by

$$reg(\{a_i, a_j\}) = Q_{\{a_1,a_2\}}(a_1) + Q_{\{a_1,a_2\}}(a_2) - Q_{\{a_i,a_j\}}(a_i) - Q_{\{a_i,a_j\}}(a_j).$$

E.4. Random PRISM

In this environment, we randomly generate the environment with Algorithm 2 that satisfies the Assumption 1.

By construction, the environment satisfies Assumption 1. Moreover, as we randomly sample the feedback for each set randomly, it is not necessary for the generated environment to satisfy any stronger assumption, e.g., a strict preference order. The regret of set $s(t)$ is given by

$$reg(s(t)) = \sum_{a^* \in s^*} Q_{s^*}(a^*) - \sum_{a \in s(t)} Q_{s(t)}(a).$$

Algorithm 2 GENERATING ENVIRONMENT SATISFIES ASSUMPTION 1.

Input: Number of Arms n, set Size k.
Set set $s^* = \{1, 2, \cdots, k\}$ be the optimal set. Randomly Sample $Q_{s^*}(a) \sim \text{Uniform}(0, \frac{1}{k})$. The rewards are binary reward, with expectation generated as following:
for set $s \neq s^*$ **do**
  while $\sum_{a \in s} Q_s(a) > \sum_{a^* \in s^*} Q_{s^*}(a^*)$ **do**
    for $a \in s$ **do**
      if $a \in s^*$ **then**
        Sample $Q_s(a) \sim \text{Uniform}(Q_{s^*}(a), \frac{1}{k})$.
      else
        Sample $Q_s(a) \sim \text{Uniform}(0, \frac{1}{k})$.
      end if
    end for
  end while
end for