

Bandit Learning with Delayed Impact of Actions

We consider a stochastic multi-armed bandit (MAB) problem with delayed impact of actions. In our setting, the frequencies of historical decisions impact the arm rewards in the subsequent future. This delayed impact of actions is prevalent in the real world. For example, the capability to pay back a loan for people in a certain social group might depend on historically how frequently that group has been approved loan applications. If banks keep rejecting loan applications to people in a disadvantaged group, it could create a feedback loop and further damage the chance of getting loans for people in that group. In this paper, we formulate this delayed and long-term impact of actions within the context of multi-armed bandits. We generalize the classical bandit setting to encode the dependency of this “bias” due to the action history of the learning. The goal is to maximize the collected utilities over time while taking into account the dynamics created by the delayed impacts of historical actions. We propose an algorithm that achieves a regret of $\tilde{O}(KT^{2/3})$ and show a matching regret lower bound of $\Omega(KT^{2/3})$, where K is the number of arms and T denotes the learning horizon. Our results complement the bandit literature by adding techniques to deal with actions with long-term impacts and have implications in designing fair algorithms.

ACM Reference Format:

. 2020. Bandit Learning with Delayed Impact of Actions. 1, 1 (October 2020), 44 pages. <https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

1 INTRODUCTION

Algorithms have been increasingly involved in high-stakes decision making. Examples include approving/rejecting loan applications [19, 31], deciding on employment and compensation [5, 17], and recidivism and bail decisions [1]. Automating these high-stakes decisions has raised ethical concerns on whether it amplifies the discrimination bias against protected classes [12, 43]. There have also been growing efforts towards studying algorithmic approaches to mitigate these concerns. Most of the above efforts have focused on static settings: a utility-maximizing decision maker needs to ensure her actions satisfy some fairness criteria at the decision time, without considering the long-term impacts of actions. However, in practice, these decisions may often have long-term impacts on the rewards and well-beings for the people involved. For example,

- A regional financial institute may decide on the fraction of loan applications from different social groups to approve. These decisions could affect the development of these groups: The capability of applicants from a group to pay back a loan might depend on the group’s socio-economic status, which is influenced by how frequently applications from this group have been approved [6, 16].

Author’s address:

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2020 Association for Computing Machinery.

XXXX-XXXX/2020/10-ART \$15.00

<https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

- The police department may decide on the amount of patrol time or the probability of patrol in a neighborhood (primarily populated with a demographic group). The likelihood to catch a crime in a neighborhood might depend on how frequent the police decides to patrol this area [21, 24].

These observations raise the following concerns. If being insensitive with the long-term impact of actions, the decision maker risks treating a historically disadvantaged group unfairly. Making things even worse, these unfair and oblivious decisions might reinforce existing biases and make it harder to observe the true potential for a disadvantaged group. While being a relatively under-explored (but important) topic, several recent works have looked into this problem of delayed impact of actions in algorithm design. However, these studies have so far focused on understanding the impact in a one-step delay of actions [26, 30, 38], or a sequential decision making setting without uncertainty [27, 39, 42, 54].

Our work departs from the above line of efforts by studying the long-term impact of actions in sequential decision making under uncertainty. We generalize the multi-armed bandit setting by introducing the *impact functions* that encode the dependency of the “bias” due to the action history of the learning to the arm rewards. Our goal is to learn to maximize the rewards obtained over time, in which the rewards’ evolution could depend on the past actions. This history-dependency structure of observed rewards makes our problem substantially more challenging. In particular, we first show that applying standard bandit algorithms leads to linear regret, i.e., existing approaches will obtain low rewards with a biased learning process. We then demonstrate that, under relatively mild conditions, we are able to present algorithms with theoretical guarantees.

We start by addressing a simpler setting, *action-dependent bandits*, in which the reward of each arm is influenced by the action taken by the learner at each particular round.¹ While this setting is interesting on its own, it also serves as the building block for our main problem of history-dependent bandits as discussed later. In each round of this setting, the learner first commits and announces her strategy, a probability distributions over arms, which could be interpreted as the probabilities of choosing arms or the proportions of resources allocated to arms. The reward distribution of each arm is influenced by this announced strategy. The learner then receives the reward associated with the arm pulled by the realized strategy. For this setting, we propose an algorithm that takes into account the action-dependent reward structure and achieves a regret of $\tilde{O}(KT^{2/3})$.² Furthermore, we show a matching lower regret bound of $\Omega(KT^{2/3})$.

We then consider our main setting, *history-dependent bandits*, in which the reward of each arm is influenced by the learner’s action history via an impact function. In this setting, the reward distribution of an arm is influenced by a time-discounted average of the history of the actions on the arm.³ We provide a template that helps reduce this problem to the above action-dependent setting with phases. In each phase, we start with an approaching stage to smooth out the historical bias caused by past taken actions. In the second stage of each phase, we then explore the true payoff under a targeted selection strategy, using a stylized algorithm similar to the one we developed for the first action-dependent setting. We show that a $\tilde{O}(KT^{2/3})$ regret bound is again achievable that matches the one in the action-dependent case and show a matching lower regret bound of $\Omega(KT^{2/3})$.

¹This setting can also be viewed as a special case as the history dependence only encodes the most recent action.

²We use $\tilde{O}(\cdot)$ to suppress the dependency of logarithmic factors.

³Our results apply to a general family of impact functions. See discussion in Section 5.3.

This paper contributes to the bandit literature by formulating the long-term impact of actions, which have implications in designing algorithms that are fair to different groups where learning has to be done with embedded historical biases due to past actions. We explore the algorithm design in this space and propose algorithms that achieve $\tilde{O}(KT^{2/3})$ regret for a general class of impact functions and prove a matching lower bound.

Organization of the Paper – We survey the related work in Section 2. We then provide a formal formulation of the problem in Section 3. In Section 4, we explore the action-dependent bandit problem. We propose our algorithm and analyze the regret bound of the algorithm. After establishing the results of action-dependent bandits, we study the problem of history-dependent bandits in Section 5. We present the corresponding algorithm and its theoretical guarantees. In Section 6, we present the lower bounds for both problems. A numeric experiment is included in Section 7. Finally, the paper is concluded with discussions of future work in Section 8. The full proofs of our main results are included in the appendix.

2 RELATED WORK

Our learning framework is based on the rich bandit learning literature [3, 35]. However, instead of making the standard assumption of i.i.d. or adversarial rewards, we consider the setting in which the arm reward depends on the action history. The settings most similar to ours are non-stationary bandits, including *restless* bandits [8, 18, 20, 49, 51], in which the reward of each arm changes over time regardless of whether the arm is pulled, and *rested* bandits [15, 36, 47], in which the reward of arm evolves only when it is pulled. In contrast, our model encodes a generic dependency of actions taken in the past and our setting is sort of a mix between the above two. On one hand, the reward of each arm is *restless*, because even if we do not select a particular arm at step t , the arm's underlying state will continue to evolve (this is represented by our definition of $f(t)$), which will change the expected reward to be seen in the future. On the other hand, the changing of rewards does depend on actions, so in this sense, it is related to rested bandit. Technically, due to the presence of historical bias, we allow the learner to learn the optimal strategy in a continuous space which is built on the *probabilistic* simplex over all arms. Meanwhile, our work distinguishes from prior works in that our proposed framework does not require the exact knowledge of dependency function except to the extent of a Lipschitz property and a convergence property.

Our formulation bears similarity to reinforcement learning since our impact function encodes memory (and is in fact Markovian [44, 50]), although we focus on studying the exploration-exploitation tradeoff in bandit formulation. Our techniques and approaches share similar insights with Lipschitz bandits [14, 33, 41, 48] and combinatorial bandits [9–11, 13] in that we also assume the Lipschitz reward structure and consider combinatorial action space. However, our setting is different since the arm reward explicitly depends on the learner's action history. We have a detailed regret comparisons with the regret of directly applying techniques in combinatorial bandits to our setting in Section 4.

We also note that there are several works that formulate delayed feedback in online learning [29, 32, 45, 52]. In particular, Pike-Burke et al. [45] considers the setting in which the observed reward is a sum of a number of previously generated rewards which happen to arrive in the given round. Joulani et al. [29] and Vernade et al. [52] focus on the setting where either feedback or reward is delayed. Kleinberg and Immorlica [32] studies the setting that the expected reward of an arm is an

increasing concave function of the time since it was last played. Our work differs from the above works in that, in our setting, the reward of the arm is influenced by the action history while the above works still consider stationary rewards (though the reward realization could be delayed).

Our work also has implications in algorithmic fairness. one related line of works have studied fairness in the sequential learning setting, however they do not consider long-term impact of actions [7, 22, 25, 28, 37, 40]. For the explorations of delayed impacts of actions, the studies so far have focus on addressing the one-step delayed impacts or a multi-step sequential setting with full information [6, 16, 26, 27, 38, 42]. Our work differs from the above and studies delayed impacts of actions in sequential decision making under uncertainty.

3 PROBLEM SETTING AND PRELIMINARY ANALYSIS

We formulate the setting in which an institution sequentially determines how to allocate resource to different groups. For example, a regional financial institute may decide on the fraction or overall frequency of loan applications to approve from different social groups. The police department may decide on the amount of patrol time or the patrol probability they allocate to different regions.

The institution is assumed to be a utility maximizer, aiming to maximize the expected reward associated with the allocation policy over time. If we assume the reward⁴ for allocating a unit of resource to a group is i.i.d. drawn from some unknown distribution, this problem can be reduced to a standard bandit problem, with each group representing an *arm*. The goal of the institution is then to learn a sequence of arm selections to maximize its cumulative rewards or minimize the regret.

In this work, we extend the bandit setting and consider the delayed impact of actions. Below we formalize our setup which introduces *impact functions* to bandit framework.

Action space. There are K *base arms*, indexed from $k = 1$ to K , with each base arm representing a group. At each discrete time t , the institution chooses an action, called a *meta arm*, which is a probability distribution over base arms. Let $\mathcal{P} = \Delta([K])$ be the $(K - 1)$ -dimensional probability simplex. We denote the meta arm as $\mathbf{p}(t) = \{p_1(t), \dots, p_K(t)\} \in \mathcal{P}$, where $p_k(t)$ represents the probability of choosing a base arm k ($p_k(t)$ can be equivalently interpreted as the probability of allocating a unit resource to group k or the *portion* of the resources allocated to group k). The institution only observes the reward from the arm it ends up with selecting.⁵

Delayed impacts of actions. We consider the scenario in which the rewards of actions are unknown a priori and are influenced by the action history. Formally, let $\mathcal{H}(t) = \{\mathbf{p}(s)\}_{s \in [t]}$ be the action history at time t . We define the *impact function* $\mathbf{f}(t) = F(\mathcal{H}(t))$ to summarize the impact of the learner's actions to the reward generated in each groups, where $F(\cdot)$ is the function mapping the action history to its current impact on arms' rewards. In the following discussion, we make $F(\cdot)$ implicit and use the vector $\mathbf{f}(t) = \{f_1(t), \dots, f_K(t)\}$ to denote the impact to each group, where $f_k(t)$ captures the impact of action history to arm k .

⁴The reward could be whether a crime has been stopped or whether the lender pays the monthly payment on time. For applications that require longer time periods to assess the rewards, the duration of a time step, i.e., the frequency to update the policy, would also need to be adjusted accordingly.

⁵Note that interpreting the meta-arm as a probability distribution or a proportional allocation could impact the way the rewards are generated (i.e., does the institution observe only the reward of the realized based arm, or the rewards of all base arms with non-zero allocations.) Our analysis utilizes the idea of importance weighting and could deal with both cases in the same framework. To simplify the presentation, we focus on the harder case of interpreting the meta-arm as probability distributions, though our results apply to both interpretations.

Rewards and regret. The reward for allocating resources to group k at time t depends on both $p_k(t)$ and the historical impact $f_k(t)$. In particular, when the institution allocates a unit of resource to group k , she obtains a reward (the instantaneous reward is bounded within $[0, 1]$) drawn i.i.d. from a distribution with mean $r_k(f_k(t)) \in [0, 1]$, where $r_k(\cdot)$ is unknown a priori but is Lipschitz continuous (with known Lipschitz constant $L_k \in (0, 1]$) with respect to its input, i.e., a small deviation of the institution's actions has small impacts on the unit reward from each group. When action $\mathbf{p}(t)$ is taken time t , the institution obtains an expected reward

$$U_t(\mathbf{p}(t)) = \sum_{k=1}^K p_k(t) \cdot r_k(f_k(t)). \quad (1)$$

As for the impact function, we focus on the setting in which $\mathbf{f}(t)$ is a time-discounted average, with each component $f_k(t)$ defined as

$$f_k(t) = \frac{\sum_{s=1}^t p_k(s) \gamma^{t-s}}{\sum_{s=1}^t \gamma^{t-s}}, \quad (2)$$

where $\gamma \in (0, 1)$ is the time-discounting factor. Intuitively, $f_k(t)$ is a weighted average with more weights on recent actions. We would like to highlight that our results extend to a more general family of impact functions and do not require the exact knowledge of impact functions (see discussion in Section 5.3). Let \mathcal{A} be the algorithm the institution deploys. The goal of \mathcal{A} is to choose a sequence of actions $\{\mathbf{p}(t)\}$ that maximizes the total utility. The performance of the algorithm is characterized by regret, defined as

$$\text{Reg}(T) = \sup_{\mathbf{p} \in \mathcal{P}} \sum_{t=1}^T U_t(\mathbf{p}) - \mathbb{E} \left[\sum_{t=1}^T U_t(\mathbf{p}(t)) \right],$$

where the expectation is taken on the randomness of algorithm \mathcal{A} and the utility realization⁶.

Incorporating explicit fairness constraints. As a side note, while our setting follows standard bandit settings and aims to maximize the utility, it can be extended to incorporate fairness constraints as commonly seen in the discussion of algorithmic fairness. For example, consider the notion of group fairness, which aims to achieve approximate parity of certain measures across groups. Let $\pi_i(f_i(t)) \in [0, 1]$ be the fairness measure for group i (which could reflect the socioeconomic status of the group). One common approach is to impose constraints to avoid the group disparity. Let $\tau \in [0, 1]$ be the tolerance parameter, the fairness constraints at t can be written as: $|\pi_i(f_i(t)) - \pi_j(f_j(t))| \leq \tau, \forall i, j \in [K]$. $\pi_i(\cdot)$ is unknown a priori and is dependent on the historical impact. Incorporating the fairness constraints would transform the goal of the institution as a constrained optimization problem:

$$\max_{\mathbf{p} \in \mathcal{P}} \sum_{t=1}^T U_t(\mathbf{p}(t)) \text{ s.t. } |\pi_i(f_i(t)) - \pi_j(f_j(t))| \leq \tau, \forall i, j \in [K], \forall t \in [T].$$

We can then utilize the Lagrangian relaxation: impose the fairness requirement as soft constraints and obtain an unconstrained optimization problem with a different utility function (Please see

⁶In this paper, we adopt the standard regret definition and compare against the optimal fixed policy. Another possible regret definition is to compare against the optimal dynamic policy that could change based on the history. However, calculating the optimal dynamic policy in our setting is nontrivial as it requires to solve an MDP with continuous states. Moreover, we show that, under some conditions like Definition 6.2, the optimal dynamic policy is the same as best fixed policy in hindsight.

Appendix A for detailed derivations.) As long as we also observe (bandit) feedback on the fairness measures at every time step, the techniques developed in this work can be extended to include fairness constraints.

Negative results: Classical bandit solutions might lead to linear regrets

We first present an important, though perhaps not surprising, negative result: if the institution is not aware of the delayed impact of actions, applying existing standard bandit algorithms in our setting leads to linear regrets. This negative result highlights the importance of designing new algorithms when delayed impact of actions are present.

LEMMA 3.1. *(Informal) If the institution is unaware of the delayed impact of actions, applying standard bandit algorithms (including UCB, Thompson Sampling) leads to linear regrets.*

To prove the above lemma, we construct a problem instance that the best strategy in hindsight is dependent on the learner's action. We then analyze the performance of standard algorithms, that is, arm selections that do not take into account that the actions will impact the reward generation. In addition, we show that the above lemma holds for a more general class of algorithms where the base arm selections only depends on the observed outcomes for this particular arm. This algorithm family includes well-known methods, such as versions of UCB and Thompson Sampling. The formal statement and analysis are in Appendix B.

3.1 Overview of our techniques

The key challenge of our setting is in estimating the arm rewards: when pulling the same meta arm at different time steps, the institution does not guarantee to obtain rewards drawn from the targeted distribution according to the chosen meta arm, as the arm reward depends on the historical impact $f(t)$. To address this challenge, we note if the institution keeps pulling the same meta-arm repeatedly, the impact function (and thus the arm rewards) would converge to approximately the reward that would generate under the impact of the chosen meta-arm. This observation leads to our approaches. We first develop a bandit algorithm that works with impacts that “immediately” converge. We then propose a phase-learning reduction template that reduces our general setting to the above stylish one.

We start with *action-dependent bandits*, where the impact at time t depends only on the action at t , i.e., $f(t) = \mathbf{p}(t)$. This setting not only captures the one-step impact but also offers a backbone for the phase-learning template. In this setting, when a meta-arm $\mathbf{p} = \{p_1, \dots, p_K\}$ is selected, one of the base arms k is realized with probability p_k , and the institution receives the realized reward for base arm k . However, since we also know the probability p_k for selecting each base arm, we may apply importance weighting to simulate the case as if the learner is selecting K probabilities and obtain K signals at each time step. This interpretation transforms our problem structure to a setting similar to combinatorial bandits. Furthermore, since both $r_k(\cdot)$ are Lipschitz continuous, we adopt the idea from Lipschitz bandits to discretize the continuous space of each p_k . With these ideas combined, we design a UCB-like algorithm that achieves a regret of $\mathcal{O}(KT^{2/3}(\ln T)^{1/3})$.

With the solution of action-dependent bandits, we then explore the general *history-dependent bandits* with impact functions following Equation (2). The main idea is to divided total time rounds into phases, selecting the same actions in each phase to smooth out impacts. One challenge is to construct appropriate confidence bound and adjust the length account for the historical action bias.

With a careful combination with our results for action-dependent bandits, we present an algorithm which can also achieve a regret of $\tilde{O}(KT^{2/3})$. We further proceed to show that this bound is tight.

4 ACTION-DEPENDENT BANDITS

In this section, we study action-dependent bandits, in which the impact function $\mathbf{f}(t) = \mathbf{p}(t)$. Our algorithm starts with a discretization over the space \mathcal{P} . Formally, we uniformly discretize $[0, 1]$ for each base arm into intervals of a fixed length ϵ , with carefully chosen ϵ such that $1/\epsilon$ is an positive integer.⁷ Let \mathcal{P}_ϵ be the space of discretized meta arms, i.e., for each $\mathbf{p} = \{p_1, \dots, p_K\} \in \mathcal{P}_\epsilon$, $\sum_{k=1}^K p_k = 1$ and $p_k \in \{\epsilon, 2\epsilon, \dots, 1\}$ for all k . Let $\mathbf{p}_\epsilon^* := \sup_{\mathbf{p} \in \mathcal{P}_\epsilon} \sum_{k=1}^K p_k r_k(p_k)$ denote the optimal strategy in discretized space \mathcal{P}_ϵ . After a meta arm $\mathbf{p}(t) = \{p_1(t), \dots, p_K(t)\} \in \mathcal{P}_\epsilon$ is selected, a base arm $a_t \in [K]$ drawn according to the distribution $\mathbf{p}(t)$ will be realized. From now, we use \tilde{r}_t to denote the realization of corresponding reward. The learner observes the realization of a_t and receives the instantaneous reward $\tilde{r}_t(p_{a_t}(t), a_t)$, but does not observe the rewards of other base arms. In the following discussion, we omit the second parameter and use $\tilde{r}_t(p_{a_t}(t))$ to denote $\tilde{r}_t(p_{a_t}(t), a_t)$ when it is clear from the context. We use importance weighting to construct the unbiased realized reward for each of the K elements in \mathbf{p} :

$$\hat{r}_t(p_k(t)) = \begin{cases} \tilde{r}_t(p_k(t))/p_k(t), & a_t = k \text{ and } p_k(t) \neq 0 \\ 0, & a_t \neq k \text{ or } p_k(t) = 0 \end{cases} \quad (3)$$

Since the probability of $a_t = k$ is $p_k(t)$, it is easy to see that $\mathbb{E}[\hat{r}_t(p_k(t))] = \mathbb{E}[\tilde{r}_t(p_k(t))]$. Given the importance-weighted rewards $\{\hat{r}_t(p_k(t))\}$, we re-frame our problem as choosing a K -dimensional probability measure (one value for each base arm). In particular, for each base arm k , p_k will take the value from $\{\epsilon, 2\epsilon, \dots, 1\}$, and we refer to p_k as *discretized arm*.

REMARK 4.1. *The above importance-weighting technique enables us to “observe” samples of $r_k(p_k)$ for all base arms k when selecting $\mathbf{p} = \{p_1, \dots, p_K\}$. This technique helps to bridge the gap between the interpretation of whether \mathbf{p} is a probability distribution or an allocation over base arms. Our following techniques can be applied in either interpretation.*

Algorithm 1 Action-Dependent UCB

- 1: **Input:** K, ϵ
 - 2: **Initialization:** For each discretized arm, play an arbitrary meta arm such that this discretized arm is included (if the selection of the arm is not realized, then simply initialize its reward to 0; otherwise initialize it to the observed reward divided/reweighted by the selection probability).
 - 3: **for** $t = \lceil K/\epsilon \rceil + 1, \dots, T$ **do**
 - 4: Select $\mathbf{p}(t) = \arg \max_{\mathbf{p} \in \mathcal{P}_\epsilon} \text{UCB}_t(\mathbf{p})$ where $\text{UCB}_t(\mathbf{p})$ is defined as in (4).
 - 5: Draw an arm $a_t \sim \mathbf{p}(t)$ and observe its realized reward $\tilde{r}_t(p_{a_t}(t))$.
 - 6: Update the importance-weighted rewards $\{\hat{r}_t(p_k(t))\}$ as in (3) and update the empirical mean $\{\bar{r}_t(p_k(t))\}$ for each base arm as in (4).
 - 7: **end for**
-

By doing so, our problem is now similar to combinatorial bandits, in which we are choosing K discretized arms and observe the corresponding rewards. Below we describe our UCB-like algorithm

⁷Smarter discretization generally does not lead to better regret bounds [33].

based on the reward estimation of discretized arms. We define the set $\mathcal{T}_t(p_k) = \{s \in [t] : p_k \in \mathbf{p}(s)\}$ to record all the time steps such that the deployed meta arm $\mathbf{p}(s)$ contains the discretized arm p_k . We can maintain the empirical estimates of the mean reward for each discretized arm and compute the UCB index for each meta arm $\mathbf{p} \in \mathcal{P}_\epsilon$:

$$\bar{r}_t(p_k) = \frac{\sum_{s \in \mathcal{T}_t(p_k)} \hat{r}_s(p_k)}{n_t(p_k)}, \quad \text{UCB}_t(\mathbf{p}) = \sqrt{\frac{K \ln t}{\min_{p_k \in \mathbf{p}} n_t(p_k)}} + \sum_{p_k \in \mathbf{p}} p_k \cdot \bar{r}_t(p_k), \quad (4)$$

where $n_t(p_k)$ is the cardinality of set $\mathcal{T}_t(p_k)$. With the UCB index in place, we are now ready to state our algorithm in Algorithm 1. The next theorem provides the regret bound of Algorithm 1.

THEOREM 4.2. *Let $\epsilon = \Theta((\ln T/T)^{1/3})$. The regret of Algorithm 1 (with respect to the optimal arm in non-discretized \mathcal{P}) is upper bounded as follows: $\text{Reg}(T) = O(KT^{2/3}(\ln T)^{1/3})$.*

Discussions. Our techniques have close connections to Lipschitz bandits [14, 41] and combinatorial bandits [10, 11]. Given the Lipschitz property of $r_k(\cdot)$, we are able to utilize the idea of Lipschitz bandits to discretize the strategy space and achieve sublinear regret with respect to the optimal strategy in the non-discretized strategy space. Moreover, we achieve a significantly improved regret bound by utilizing the connection between our problem setting and combinatorial bandits. In combinatorial bandits, the learner selects K actions out of action space \mathcal{M} at each time step, where $|\mathcal{M}| = \Theta(K/\epsilon)$ in our setting. Directly applying state-of-the-art combinatorial bandit algorithms [11] in our setting would achieve an instance-independent regret bound of $O(K^{3/4}T^{3/4}(\ln T)^{1/4})$, while we achieve a lower regret of $O(KT^{2/3}(\ln T)^{1/3})$.⁸ The reason for our improvement is that, for each base arm, regardless of which probability it was chosen, we can update the reward of the base arm, which provides information for all meta arms that select this arm with a different probability. This reduces the exploration and helps achieving the improvement. In addition to the above improvement, we would like to highlight that another of our main contributions is to extend the action-dependent bandits to the problem of history-dependent bandits, as discussed in Section 5.

Another natural attempt to tackle our problem is to apply EXP3 [4], which achieves sublinear regret even when the arm reward is generated adversarially. However, we would like to note that the optimal policy in our setting could be a mixed strategy, while the “sublinear” regret of EXP3 is with respect to a fixed strategy, and therefore it implies a linear regret in our setting.

4.1 Proof Sketch of Theorem 4.2

In this section, we sketch the proof of Theorem 4.2. We begin by discussing some challenges and high-level ideas to circumvent these challenges before delving into the details.

Similar to the proofs of the family of UCB-style algorithms for MAB, after an appropriate discretization, we can derive the regret as the sum of the *badness* (namely, how suboptimal a meta arm is) for all (discretized) suboptimal meta arms. However, this will cost us an exponential K in the order of final regret bound since we need to take the summation over all feasible suboptimal meta arms to achieve the regret, yet the number of suboptimal meta arms grows exponentially with K . To tackle this challenge, we focus on the derivations of badness and counting the *minimum*

⁸We compare our results with a tight regret bound achieved in Theorem 2 of [11]. The detailed derivations are deferred to Appendix C.2.1.

suboptimal selections in the space of realized actions (base arms), which enables us to reduce the exponential K to polynomial K . On a very high level, our proof proceeds in the following steps:

- In Step 1, we aim to get a high probability bound of the estimation error on the expected rewards of meta arms after discretization.
- In Step 2, we bound the probability on deploying a suboptimal meta arm when selected sufficiently many number of times, where we quantify such sufficiency via $\min_{p_k \in \mathbf{p}} n_t(p_k)$.
- In Step 3 and Wrapping-up step, we bound the expected value of $\min_{p_k \in \mathbf{p}} n_t(p_k)$ and connect the regret on playing suboptimal meta arms \mathbf{p} with the regret incurred on including discretized arms $p_k \notin \mathbf{p}_\epsilon^*$ which are not in optimal strategy (in discretized space).

Finally, the regret bound of Algorithm 1 can be achieved by optimizing the discretization parameter.

Step 1: Bounding the error of $|\bar{U}(\mathbf{p}) - U(\mathbf{p})|$. For any $\mathbf{p} = \{p_1, \dots, p_K\}$, define the empirical reward $\bar{U}_t(\mathbf{p}) = \sum_{k=1}^K p_k \bar{r}_t(p_k)$. The first step of our proof is to bound $\mathbb{P}(|\bar{U}_t(\mathbf{p}) - U(\mathbf{p})| \leq \delta)$ for each meta arm $\mathbf{p} = \{p_1, \dots, p_K\}$ with high probability.⁹ Using the Hoeffding's inequality, we obtain

$$\mathbb{P}(|\bar{U}_t(\mathbf{p}) - U(\mathbf{p})| \geq \delta) \leq 2 \exp(-2\delta^2 n_t(p_{\min}(\mathbf{p}))/K),$$

where $p_{\min}(\mathbf{p}) := \arg \min_{p_k \in \mathbf{p}} n_t(p_k)$. By choosing $\delta = \sqrt{\frac{K \ln t}{n_t(p_{\min}(\mathbf{p}))}}$ in the above inequality, for each meta arm \mathbf{p} at time t , we have that $|\bar{U}_t(\mathbf{p}) - U(\mathbf{p})| \leq \sqrt{K \ln t / n_t(p_{\min}(\mathbf{p}))}$, with the probability at least $1 - 2/t^2$.

Step 2: Bounding the probability on deploying suboptimal meta arm. With the above high probability error bound we obtain in Step 1, we can construct an UCB index for each meta arm $\mathbf{p} \in \mathcal{P}_\epsilon$:

$$\text{UCB}_t(\mathbf{p}) = \bar{U}_t(\mathbf{p}) + \sqrt{\frac{K \ln t}{n_t(p_{\min}(\mathbf{p}))}}. \quad (5)$$

The above constructed UCB index gives the following guarantee:

LEMMA 4.3. *At any time round t , for a suboptimal meta arm \mathbf{p} , if it satisfies $n_t(p_{\min}(\mathbf{p})) \geq 4K \ln t / \Delta_{\mathbf{p}}^2$, then $\text{UCB}_t(\mathbf{p}) < \text{UCB}_t(\mathbf{p}_\epsilon^*)$ with the probability at least $1 - 4/t^2$. Thus, for any t ,*

$$\mathbb{P}(\mathbf{p}(t) = \mathbf{p} | n_t(p_{\min}(\mathbf{p})) \geq 4K \ln t / \Delta_{\mathbf{p}}^2) \leq 4t^{-2},$$

where $\Delta_{\mathbf{p}}$ denotes the badness of meta arm \mathbf{p} .

PROOF SKETCH. We provide a proof sketch here. We prove this lemma by establishing two “events” which occur with high probability: (1) the UCB index of each meta arm will concentrate on the true mean reward of \mathbf{p} ; (2) the empirical mean reward of each meta arm \mathbf{p} will also concentrate on the true mean reward of \mathbf{p} . We then show that the probability of either one of the events not holding is at most $4/t^2$. By a union bound we prove the above desired lemma. \square

Intuitively, Lemma 4.3 essentially shows that for a meta arm \mathbf{p} , if its $n_t(p_{\min}(\mathbf{p}))$ is sufficiently sampled with respect to $\Delta_{\mathbf{p}}$, that is, sampled at least $4K \ln t / \Delta_{\mathbf{p}}^2$ times, we know that the probability that we hit this suboptimal meta arm is very small.

⁹We use δ to denote the estimation error, as ϵ has been used as the discretization parameter.

Step 3: Bounding the $\mathbb{E}[n_T(p_{\min}(\mathbf{p}))]$. Ideally, we would like to bound the number of the selections on deploying the suboptimal meta arm, i.e., $N_T(\mathbf{p})$, in a logarithmic order of T . However, if we proceed to bound this by separately considering each meta arm, the final regret bound will have an order with exponent in K since the number of meta arms grows exponentially in K . Instead, we turn to bound $\mathbb{E}[n_T(p_{\min}(\mathbf{p}))]$. Recall that by the definitions of $n_T(p)$ and $p_{\min}(\mathbf{p})$, the pulls of \mathbf{p} is upper bounded by its $n_T(p_{\min}(\mathbf{p}))$. This quantity will help us to reduce the exponential K to the polynomial K . This is formalized in the following lemma.

LEMMA 4.4. *For each suboptimal meta arm $\mathbf{p} \neq \mathbf{p}_\epsilon^*$, we have that $\mathbb{E}[n_T(p_{\min}(\mathbf{p}))] \leq \frac{4K \ln T}{\Delta_{\mathbf{p}}^2} + O(1)$.*

We use this connection in the following step to reduce the computation of regret on pulling all suboptimal meta arms to calculate the regret via the summation over discretized arms.

Wrapping up: Proof of Theorem 4.2. We are now ready to prove Theorem 4.2. We first define notations that are helpful for our analysis. To circumvent the summation over all feasible suboptimal arms $\{\mathbf{p}\}$, for each discretized arm p_k , we define the notion of *super set* $\mathcal{S}(p_k) := \{\mathbf{p} : p_k \in \mathbf{p}\}$ which contains all suboptimal meta arms that include this discretized arm. With a slight abuse of notations, we also sort all meta arms in $\mathcal{S}(p_k)$ as $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{I(p_k)}$ in ascending order of their expected rewards, where $I(p_k) := |\mathcal{S}(p_k)|$ is the cardinality of the super set $\mathcal{S}(p_k)$. For $\mathbf{p}_l \in \mathcal{S}(p_k)$, we also define $\Delta_l^{p_k} := \Delta_{\mathbf{p}_l}$ where $l \in [I(p_k)]$, and specifically $\Delta_{\min}^{p_k} := \min_{\mathbf{p} \in \mathcal{S}(p_k)} \Delta_{\mathbf{p}} = \Delta_{I(p_k)}^{p_k}$; $\Delta_{\max}^{p_k} := \max_{\mathbf{p} \in \mathcal{S}(p_k)} \Delta_{\mathbf{p}} = \Delta_1^{p_k}$. Let $\text{Reg}_\epsilon(T)$ denote the regret relative to the best strategy in the discretized space parameterized by ϵ . With these notations, we first establish the following instance-dependent regret.

LEMMA 4.5. *Following the UCB designed in (5), we have the following instance-dependent regret on the discretized arm space: $\text{Reg}_\epsilon(T) \leq \lceil K/\epsilon \rceil \cdot (\Delta_{\max} + O(1)) + \sum_{p_k: \Delta_{\min}^{p_k} > 0} 8K \ln T / \Delta_{\min}^{p_k}$, where $\Delta_{\max} := \max_{p_k} \Delta_{\max}^{p_k}$.*

PROOF. Note that by definitions, we can compute the regret $\text{Reg}_\epsilon(T)$ as follows:

$$\text{Reg}_\epsilon(T) = \sum_{\mathbf{p} \in \mathcal{P}_\epsilon} \mathbb{E}[N_T(\mathbf{p})] \Delta_{\mathbf{p}} \leq \sum_{p_k} \sum_{l \in [I(p_k)]} \mathbb{E}[N_T(\mathbf{p}_l)] \Delta_l^{p_k}. \quad (6)$$

Observe that, by Lemma 4.4, for each discretized arm p_k , there are two possible cases:

- There exists a meta arm $\mathbf{p}_l \in \mathcal{S}(p_k)$, and its $p_{\min}(\mathbf{p}_l) = p_k$. Then by linearity of expectation, we can bound the expectation of total number of pulls for all $\mathbf{p}_{l'} \in \mathcal{S}(p_k)$ as follows

$$\sum_{\mathbf{p}_{l'} \in \mathcal{S}(p_k)} \mathbb{E}[N_T(\mathbf{p}_{l'})] = \mathbb{E}[n_T(p_k)] \leq \frac{4K \ln T}{(\Delta_{\min}^{p_k})^2} + O(1).$$

- There exists no meta arm $\mathbf{p} \in \mathcal{S}(p_k)$, and $p_{\min}(\mathbf{p})$ for each \mathbf{p} is p_k . In this case, for each $\mathbf{p}_l \in \mathcal{S}(p_k)$, there always exists another discretized arm p' that is included in \mathbf{p}_l such that $p' = p_{\min}(\mathbf{p}_l)$ but $p' \neq p_k$. Thus, for each $\mathbf{p}_l \in \mathcal{S}(p_k)$, together with other meta arms which also include discretized arm p' as \mathbf{p}_l , we have that

$$\sum_{\mathbf{p} \in \bigcup_{p' \in \mathcal{P}} \mathcal{P}} \mathbb{E}[N_T(\mathbf{p})] = \sum_{\mathbf{p} \in \mathcal{S}(p')} \mathbb{E}[N_T(\mathbf{p})] = \mathbb{E}[n_T(p')] \leq \frac{4K \ln T}{(\Delta_{\min}^{p'})^2} + O(1).$$

The above observations imply that even though we can not find any meta arm \mathbf{p} in $\mathcal{S}(p_k)$ such that $p_{\min}(\mathbf{p}) = p_k$, we can always carry out similar analysis by finding another discretized arm $p' \in \mathbf{p}$ but $p' \neq p_k$, such that $p' = p_{\min}(\mathbf{p})$. Thus, for each discretized arm p_k , we can focus on the case where p_k is able to attain the minimum $n_t(p_k)$ for some $\mathbf{p} \in \mathcal{S}(p_k)$. For analysis convenience, instead of looking at the counter of \mathbf{p} , i.e., $n_t(p_{\min}(\mathbf{p}))$, we will define a counter $c(p_k)$ for each discretized arm p_k and the value of $c(p_k)$ at time t is denoted by $c_t(p_k)$. The update of $c_t(p_k)$ is as follows: For a round $t > \lceil K/\epsilon \rceil$ (here $\lceil K/\epsilon \rceil$ is the number of rounds needed for initialization), let $\mathbf{p}(t)$ be the meta arm selected in round t by the algorithm. Let $p_k = \arg \min_{p_k \in \mathbf{p}(t)} c_{t-1}(p_k)$. We increment $c(p_k)$ by one, i.e., $c_t(p_k) = c_{t-1}(p_k) + 1$. In other words, we find the discretized arm p_k with the smallest counter in $\mathbf{p}(t)$ and increment its counter. If such p_k is not unique, we pick an arbitrary discretized arm with the smallest counter. Note that the initialization gives $\sum_{p_k} c_{\lceil K/\epsilon \rceil}(p_k) = \lceil K/\epsilon \rceil$. It is easy to see that for any $p_k = p_{\min}(\mathbf{p})$, we have $n_t(p_k) = c_t(p_k)$.

With the above change of counters, Lemma 4.3 and Lemma 4.4 then have the implication on selecting discretized arm $p_k \notin \mathbf{p}_\epsilon^*$ given its counter $c_t(p_k)$. To see this, for each $\mathbf{p}_l \in \mathcal{S}(p_k)$, we define sufficient selection of discretized arm p_k with respect to \mathbf{p}_l as p_k being selected $4K \ln T / (\Delta_l^{p_k})^2$ times and p_k 's counter $c(p_k)$ being incremented in these selected instances. Then Lemma 4.3 tells us when p_k is sufficiently selected with respect to \mathbf{p}_l , the probability that the meta arm \mathbf{p}_l is selected by the algorithm is very small. On the other hand, when p_k 's counter $c(p_k)$ is incremented, but if p_k is under-selected with respect to \mathbf{p}_l , we incur a regret of at most $\Delta_j^{p_k}$ for some $j \leq l$.

Define $C_T(\Delta) := \frac{4K \ln T}{\Delta^2}$, the number of selection that is considered sufficient for a meta arm with reward Δ away from the optimal strategy \mathbf{p}_ϵ^* with respect to time horizon t . With the above analysis, we define following two situations for the counter of each discretized arm:

$$c_T^{l,\text{suf}}(p_k) := \sum_{t=\lceil K/\epsilon \rceil+1}^T \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k) > C_T(\Delta_l^{p_k}) \right) \quad (7)$$

$$c_T^{l,\text{und}}(p_k) := \sum_{t=\lceil K/\epsilon \rceil+1}^T \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \leq C_T(\Delta_l^{p_k}) \right). \quad (8)$$

Clearly, we have $c_T(p_k) = 1 + \sum_{l \in I(p_k)} (c_T^{l,\text{suf}}(p_k) + c_T^{l,\text{und}}(p_k))$. With these notations, we can write (6) as follows:

$$\text{Reg}_\epsilon(T) \leq \mathbb{E} \left[\sum_{p_k} \left(\Delta_{\max}^{p_k} + \sum_{l \in I(p_k)} \left(c_T^{l,\text{suf}}(p_k) + c_T^{l,\text{und}}(p_k) \right) \cdot \Delta_l^{p_k} \right) \right]. \quad (9)$$

The proof of this lemma will complete after establishing following two claims:

$$\text{Claim 1: } \mathbb{E} \left[\sum_{p_k} \sum_{l \in I(p_k)} c_T^{l,\text{suf}}(p_k) \right] \leq \lceil K/\epsilon \rceil \cdot O(1). \quad (10)$$

$$\text{Claim 2: } \mathbb{E} \left[\sum_{p_k} \sum_{l \in I(p_k)} c_T^{l,\text{und}}(p_k) \Delta_l^{p_k} \right] \leq \sum_{p_k} \left((4K \ln T) / \Delta_{\min}^{p_k} + 4K \ln T (1/\Delta_{\min}^{p_k} - 1/\Delta_{\max}^{p_k}) \right). \quad (11)$$

The proof of the above two claims is provided in Appendix C.2.4. \square

To achieve instance-independent regret bound, we need to deal with the case when the meta-arm gap $\Delta_{\min}^{p_k}$ is too small, leading the regret to approach infinite. Nevertheless, one can still show that when $\Delta_{\min}^{p_k} \leq 1/\sqrt{T}$, the regret contributed by this scenario scales at most $O(\sqrt{T})$ at time horizon T

LEMMA 4.6. *Following the UCB designed in (5), we have: $\text{Reg}_\epsilon(T) \leq O(K\sqrt{T \ln T/\epsilon})$.*

With the above lemma in hand, picking $\epsilon = \Theta((\ln T/T)^{1/3})$ will give us desired result in Theorem 4.2.

5 HISTORY-DEPENDENT BANDITS

We now describe how to utilize our results for action-dependent bandits to solve the history-dependent bandit learning problem, with the impact function specified in Equation (2). The crux of our analysis is the observation that, in history-dependent bandits, if the learner keeps selecting the same strategy \mathbf{p} for a long enough period of time, the expected one-shot utility will be approaching the utility of selecting \mathbf{p} in the action-dependent bandits. More specifically, suppose after time t , the current action impact for all arms is $\mathbf{f}(t) = \mathbf{p}^{(y)}(t) = \{p_1^{(y)}(t), \dots, p_K^{(y)}(t)\}$. Assume that the learner is interested in learning about the utility of selecting $\mathbf{p} = \{p_1, \dots, p_K\}$ next. Since the rewards are influenced by $\mathbf{f}(t)$, selecting \mathbf{p} at time $t + 1$ does not necessarily give us the utility samples at $U(\mathbf{p})$. Instead, the learner can keep pulling this meta arm for a non-negligible s consecutive rounds to ensure that $\mathbf{f}(t + s)$ approaches \mathbf{p} . Following this idea, we decompose the total number of time rounds T into $\lfloor T/L \rfloor$ phases which each phase is associated with L rounds. We denote $m \in [1, \dots, \lfloor T/L \rfloor]$ as the phase index and $\mathbf{p}(m)$ as the selected meta-arm in the m -th phase. To summarize the above phased-learning template:

- In each phase m , we start with an *approaching stage*: the first s_a rounds of the phase. This stage is used to “move” $\mathbf{f}(t + s)$ with $1 \leq s \leq s_a$ towards to \mathbf{p} .
- In the second stage, namely, *estimation stage*, of each phase: the remaining $L - s_a$ rounds. This stage is used for collecting the realized rewards and estimating the true reward mean on action \mathbf{p} .
- Finally, we leverage our tools in action-dependent bandits to decide what meta arm to select in each phase.

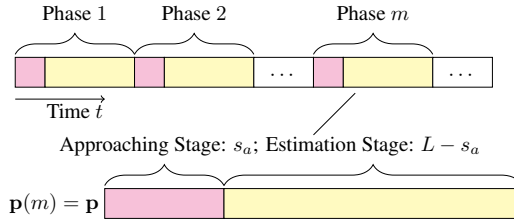


Fig. 1. A graphical illustration of Algorithm 2. We deploy \mathbf{p} for all rounds in m -th phases, therefore, we use $\mathbf{p}(m) = \mathbf{p}$ to represent $\mathbf{p}(t) = \mathbf{p}$ for simplicity.

Note that even if we keep pulling the arm k with the constant probability p_k in the approaching stage, the action impact in the estimation stage is not exactly the same as meta arm we want to learn, i.e., $\mathbf{f}(t + s) \neq \mathbf{p}$ for $s \in (s_a, L]$, due to the finite length of the stage. However, we can guarantee all $\mathbf{f}(t + s)$ for $s \in (s_a, L]$ is close enough to \mathbf{p} by bounding its approximation error w.r.t \mathbf{p} . The above idea enables a more general reduction algorithm that is compatible with any bandit algorithm that solves the action-dependent case. Let $\rho = (L - s_a)/L$ be the ratio of number of rounds in estimation stage of each phase. We present this reduction in Algorithm 2 and a graphical illustration in Figure 1.

Algorithm 2 Reduction Template

```

1: Input:  $K, T; \gamma, \epsilon, \rho \in (0, 1), s_a$ .
2: Input: A bandit algorithm  $\mathcal{A}$ : History-Dependent UCB (Algorithm 3).
3: Split all rounds into consecutive phases of  $L = s_a/(1 - \rho)$  rounds each.
4: for  $m = 1, \dots$  do
5:   Query algorithm  $\mathcal{A}$  for its meta arm selection  $\mathbf{p}(m) = \mathbf{p}$ .
6:   Each phase is separated into two stages:
   • Approaching stage:  $t = L(m - 1) + 1, \dots, L(m - 1) + s_a$ ;
   • Estimation stage:  $t = L(m - 1) + s_a + 1, \dots, Lm$ .
7:   for  $t = L(m - 1) + 1, \dots, L(m - 1) + s_a$  do
8:     Deploy the meta arm  $\mathbf{p}$ .
9:   end for
10:  for  $t = L(m - 1) + s_a + 1, \dots, Lm$  do
11:    Deploy the meta arm  $\mathbf{p}$ ;
12:    Collect the realized rewards  $\tilde{r}_t$  to estimate the mean reward as in (12).
13:  end for
14:  Update  $\bar{U}_t^e(\mathbf{p})$  as in (12).
15: end for

```

5.1 History-Dependent UCB

In this section, we show how to utilize the reduction template to achieve a $\tilde{O}(KT^{2/3})$ regret bound for history-dependent bandits.

We first introduce some notations. For each discretized arm p_k , similar to action-dependent case, we define $\Gamma_m(p_k) := \{s : s \in ((i - 1)L + s_a, iL] \text{ where } p_k \in \mathbf{p}(i), \forall i \in [m]\}$ as the set of all time indexes till the end of phase m in estimation stages such that arm k is pulled with probability p_k . We now define the following empirical $\bar{r}_m^e(p_k)$ computed from our observations and the empirical utility $\bar{U}_m^e(\mathbf{p})$:¹⁰

$$\bar{r}_m^e(p_k) = \frac{1}{n_m^e(p_k)} \sum_{s \in \Gamma_m(p_k)} \hat{r}_s(p_k^{(y)}(s)), \quad \bar{U}_m^e(\mathbf{p}) = \sum_{p_k \in \mathbf{p}} p_k \cdot \bar{r}_m^e(p_k), \quad (12)$$

where $n_m^e(p_k) := |\Gamma_m(p_k)|$ denotes the total number of rounds pulling arm k with the probability p_k in all estimation stages, and $\hat{r}_s(p_k^{(y)}(s))$ is similarly defined as in Equation (3). As mentioned, we use the smoothed-out frequency $\{p_k^{(y)}(s)\}_{s \in \Gamma_m(p_k)}$ in the estimation stage as an approximation for the discounted frequency right after the approaching stage.

We compute our UCB for each meta arm at the end of each phase. We define and compute $\text{err} := Ky^{s_a}(L^* + 1)$, the approximation error incurred when we smooth out the historical action impact.¹¹ With these preparations, we present the phased history-dependent UCB algorithm (in companion with Algorithm 2) in Algorithm 3. The main result of this section is given as follows:

¹⁰e in superscript stands for etimation stage.

¹¹The value of err is computed in Lemma 5.3 and $L^* := \arg \max_k L_k$.

Algorithm 3 History-Dependent UCB

- 1: Construct UCB for each meta arm $\mathbf{p} \in \mathcal{P}_\epsilon$ at the end of each phase $m = 1, 2, \dots$, as follows:

$$\text{UCB}_m(\mathbf{p}) = \bar{U}_m^e(\mathbf{p}) + \text{err} + 3\sqrt{\frac{K \ln(L\rho)}{\min_{p_k \in \mathbf{p}} n_m^e(p_k)}}.$$

- 2: Select $\mathbf{p}(m+1) = \arg \max_{\mathbf{p}} \text{UCB}_m(\mathbf{p})$ with ties breaking equally.
-

THEOREM 5.1. *For any constant ratio $\rho \in (0, 1)$ and $\gamma \in (0, 1)$, let $\epsilon = \Theta((\ln(T\rho)/(T\rho))^{1/3})$ and $s_a = \Theta(\ln(\epsilon^{1/3}/K)/\ln \gamma)$. The regret of Algorithm 2 with Algorithm 3 as input bounds as follows: $\text{Reg}(T) = \mathcal{O}(KT^{2/3}((\ln(T\rho))/\rho)^{1/3})$.*

For a constant ratio ρ , we match the optimal regret order for action-dependent bandits. When γ is smaller, the impact function “forgets” the impact of past-taken actions faster, therefore less rounds in approaching stage will be needed (see s_a ’s dependence in γ) and this leads to larger ρ .

REMARK 5.2. *The dependence of our regret on the phase length L is encoded in ρ . When implementing our algorithm (Section 7), we calculate L via s_a given the ratio ρ . We also run simulations of our algorithm on different ratios ρ , the results show that the performance of our algorithm are not sensitive w.r.t. specifying ρ s - in practice, we do not require the exact knowledge of ρ , instead we can afford to use a rough estimation of its upper bound to compute L .*

5.2 Proof Sketch of Theorem 5.1

In this section, we provide the analysis of Theorem 5.1. The analysis follows a similar structure to the one used in the proof of the regret bound in Theorem 4.2. However, due to the existence of historical bias, we need to perform a careful computation when handling the high-probability bounds. Specifically, we need to prove that, after deploying \mathbf{p} consecutively for moderate long rounds (tuning s_a), the approximation error $|U(\mathbf{p}) - \bar{U}_m^e(\mathbf{p})|$ is small enough. The analysis is provided below.

Step 1: Bounding the small error of $|U(\mathbf{p}) - \bar{U}_m^e(\mathbf{p})|$ with high-probability. Our first step is to ensure the empirical mean reward estimation we obtain from the information we collected in all the estimation stages will approximate well the true mean of meta arm we want to deploy.

To return a high-probability error bound, we first bound the approximation error incurred due to the dependency of history of arm selection (“historical bias”). This is summarized below.

LEMMA 5.3. *Keeping deploying $\mathbf{p} = \{p_1, \dots, p_K\}$ in the approaching stage with s_a rounds, and collect all reward feedback in the following estimation stage for the empirical estimation of rewards generated by \mathbf{p} , one can bound the approximation error as follows:*

$$\mathbb{E}[|\bar{U}_m^e(\mathbf{p}) - \bar{U}(\mathbf{p})|] \leq KY^{s_a}(L^* + 1),$$

where $\bar{U}(\mathbf{p})$ denote the empirical mean of rewards if the instantaneous reward is truly sampled from mean reward function according to \mathbf{p} .

The proof of the above lemma is mainly built on analyzing the convergence of $\mathbf{p}^{(\gamma)}$ via pulling the arms with the same probability consistently.

With the approximation error at hand, we can then bound the error of $|U(\mathbf{p}) - \bar{U}_m^e(\mathbf{p})|$ with high probability:

LEMMA 5.4. *With probability at least $1 - \frac{6}{(L\rho m)^2}$, we have*

$$|U(\mathbf{p}) - \bar{U}_m^e(\mathbf{p})| \leq \text{err} + 3\sqrt{\frac{K \ln(L\rho m)}{n_m^e(p_{\min}(\mathbf{p}))}},$$

where $p_{\min}(\mathbf{p}) = \arg \min_{p_k \in \mathbf{p}} n_m^e(p_k)$.

Step 2: Bounding the probability on deploying suboptimal meta arm. Till now, with the help of the above high probability bound on the empirical reward estimation, the history-dependent reward bandit setting is largely reduced to an action-dependent one with a certain approximation error. Then, similar to our argument on upper bound of action-dependent bandits, we have the following specific Lemma for history-dependent bandits:

LEMMA 5.5. *At the end of each phase, for a suboptimal meta arm \mathbf{p} , if it satisfies $n_m^e(p_{\min}(\mathbf{p})) \geq \frac{9K \ln(L\rho m)}{(\Delta_{\mathbf{p}}/2 - \text{err})^2}$, then with the probability at least $1 - \frac{12}{(L\rho m)^2}$, we have $\text{UCB}_m(\mathbf{p}) < \text{UCB}_m(\mathbf{p}^*)$, i.e.,*

$$\mathbb{P}\left(\mathbf{p}(m+1) = \mathbf{p} | n_m^e(p_{\min}(\mathbf{p})) \geq \frac{9K \ln(L\rho m)}{(\Delta_{\mathbf{p}}/2 - \text{err})^2}\right) \leq \frac{12}{(L\rho m)^2}.$$

The above lemma implies that we will stop deploying suboptimal meta arm \mathbf{p} and further prevent it from incurring regret as we gather more information about it such that $\text{UCB}_m(\mathbf{p}) < \text{UCB}_m(\mathbf{p}^*)$.

Step 3: Bounding the $\mathbb{E}[n_m^e(p_{\min}(\mathbf{p}))]$. The results we obtain in Step 2 implies following guarantee:

LEMMA 5.6. *For each suboptimal meta arm $\mathbf{p} \neq \mathbf{p}^*$, we have following:*

$$\mathbb{E}[n_m^e(p_{\min}(\mathbf{p}))] \leq \frac{9K \ln(L\rho m)}{(\Delta_{\mathbf{p}}/2 - \text{err})^2} + \frac{2\pi^2}{L - s_a}.$$

Wrapping up: Proof of Theorem 5.1. Following the similar analysis in Section 3, we can also get an instance-dependent regret bound for history-dependent bandits:

LEMMA 5.7. *Following the UCB designed in Algorithm 3, we have following instance-dependent regret on discretized arm space for history-dependent bandits:*

$$\text{Reg}_\epsilon(T) \leq O\left(\frac{K\Delta_{\max}}{L\epsilon\rho^2}\right) + \sum_{p_k} \left(\frac{9K \ln(T\rho)}{\rho} \left(\frac{\Delta_{\min}^{p_k}}{(\Delta_{\min}^{p_k}/2 - \text{err})^2} + \frac{2}{\Delta_{\min}^{p_k}/2 - \text{err}} \right) \right).$$

PROOF SKETCH. The proof of the above lemma follows from similar outline as in the proof of Lemma 4.5. We emphasize differences here. We define a counter $c^e(p_k)$ for each discretized arm p_k and the value of $c^e(p_k)$ at phase m is denoted by $c_m^e(p_k)$. But different from the proof of action-dependent bandits, we update the counter $c^e(p_k)$ only when we start a new phase. In particular, for a phase $m \geq 1$, let $\mathbf{p}(m)$ be the meta arm selected in the phase m by the algorithm. Let $p_k = \arg \min_{p_k \in \mathbf{p}(m)} c_m^e(p_k)$, we then increment $c_m^e(p_k)$ by one, i.e., $c_m^e(p_k) = c_{m-1}^e(p_k) + 1$. In other words, we find the discretized arm p_k with the smallest counter in $\mathbf{p}(m)$ and increment its counter. Note that the initialization gives $\sum_{p_k} c_0^e(p_k) = \lceil K/\epsilon \rceil$. Then for any $p_k = p_{\min}(\mathbf{p})$, we

have $n_m(p_k) = L\rho \cdot c_m(p_k)$. We also define $C_M^e(\Delta) := \frac{9K \ln(ML\rho)}{L\rho(\Delta/2 - \text{err})^2}$, the number of selection that is considered sufficient for a meta arm with reward Δ away from the optimal strategy \mathbf{p}_ϵ^* with respect to phase horizon M . With the above notations, we define following two situations for the counter of each discretized arm:

$$c_M^{\text{e},l,\text{sup}}(p_k) := \sum_{m=1}^M \mathbb{1} \left(\mathbf{p}(m) = \mathbf{p}_l, c_m^{\text{e}}(p_k) > c_{m-1}^{\text{e}}(p_k) > C_M^e(\Delta_l^{p_k}) \right)$$

$$c_M^{\text{e},l,\text{und}}(p_k) := \sum_{m=1}^M \mathbb{1} \left(\mathbf{p}(m) = \mathbf{p}_l, c_m^{\text{e}}(p_k) > c_{m-1}^{\text{e}}(p_k), c_{m-1}^{\text{e}}(p_k) \leq C_M^e(\Delta_l^{p_k}) \right).$$

Similarly, we have $c_M^{\text{e}}(p_k) = 1 + \sum_{l \in I(p_k)} (c_M^{\text{e},l,\text{sup}}(p_k) + c_M^{\text{e},l,\text{und}}(p_k))$. With these notations, we can write (23) as follows:

$$\text{Reg}_\epsilon(T) \leq \mathbb{E} \left[\sum_{p_k} \left(\Delta_{\max}^{p_k} + L \cdot \sum_{l \in I(p_k)} (c_M^{\text{e},l,\text{sup}}(p_k) + c_M^{\text{e},l,\text{und}}(p_k)) \cdot \Delta_l^{p_k} \right) \right].$$

The proof of this lemma will complete after establishing following two claims:

Claim 3: $\mathbb{E} \left[L \sum_{p_k} \sum_{l \in I(p_k)} c_M^{\text{e},l,\text{sup}}(p_k) \right] \leq O \left(\frac{K}{L\epsilon\rho^2} \right).$

Claim 4: $\mathbb{E} \left[L \sum_{p_k} \sum_{l \in I(p_k)} c_M^{\text{e},l,\text{und}}(p_k) \Delta_l^{p_k} \right] \leq \sum_{p_k} \left(\frac{9K \ln(ML\rho)}{\rho} \left(\frac{\Delta_{\min}^{p_k}}{(\Delta_{\min}^{p_k}/2 - \text{err})^2} + \frac{2}{\Delta_{\min}^{p_k}/2 - \text{err}} \right) \right).$

The proof of the above two claims is provided in Appendix 5.7. \square

The instance-independent regret on discretized arm space is summarized in following lemma:

LEMMA 5.8. *Following the UCB designed in Algorithm 3, the instance-independent regret is given as*
 $\text{Reg}_\epsilon(T) \leq O \left(K \cdot \sqrt{T \ln(T\rho)/(\rho\epsilon)} + K\text{Err} + K/(L\epsilon\rho^2) \right).$

Combining with the discretization error, and optimizing ϵ and s_a will yield us the result in Theorem 5.1.

5.3 Extension to General Impact Functions

So far, we demonstrate how to utilize the reduction template for history-dependent bandits, with the impact function specified as in (2). However, the same technique can be applied for a general family of impact functions. In particular, as long as the impact function converges after the learner keeps selecting the same action, the same technique applies. To be more precise, we only require $\mathbf{f}(t)$ to satisfy the condition $|\mathbf{f}_k(t+s) - \mathbf{g}(p_k)| \leq \gamma^s, \gamma \in (0, 1)$ when the learner keeps pulling arm k with probability p_k for s round. The function $\mathbf{g}(\cdot)$ can be an arbitrary monotone function as long as it is continuous differentiable, for example: $\mathbf{g}(x) = x$. In fact, the only place we utilize the property of $\mathbf{f}(t)$ is to estimate how close \mathbf{f} is to $\mathbf{g}(\mathbf{p})$ after the approaching stage when repeatedly selecting \mathbf{p} . For a different $\mathbf{f}(t)$, we can define new reward mean functions $r'_k(\cdot) = r_k(\mathbf{g}(\cdot))$, and tune parameters ϵ and s_a accordingly to bound the approximation error on $|U(\mathbf{p}) - \bar{U}_m^{\text{e}}(\mathbf{p})|$ (change the Lipschitz constant). Then we can follow the same algorithm template to achieve the same regret bound.

Moreover, we do not require the exact knowledge of the impact function $\mathbf{f}(t)$. We only require the impact functions to satisfy the above conditions for our algorithms/analysis to hold. With the same arguments, while we assume the reward function $r_k(\cdot)$ is fed with the same impact function \mathbf{f} , our formulation generalizes to different impact functions for $r_k(\cdot)$, as long as these impact functions are able to stabilize given a consecutively adoption of the desired action.

6 MATCHING LOWER BOUNDS

For both action-dependent bandits and history-dependent bandits, we proposed algorithms achieving a regret bound of $\tilde{O}(KT^{2/3})$. In this section, we show that no algorithm can perform better in terms of the dependency on K and T . In particular, we derive the lower bound of our problem as $\Omega(KT^{2/3})$ in both bandit settings. This is summarized as the following theorem:

THEOREM 6.1. *Let $T > 2K$, and $K \geq 4$, there exist some problem instances for action-dependent bandits and history-dependent bandits, respectively, such that the regret for any algorithm \mathcal{A} for these two bandit settings is lower bounded as: $\inf_{\mathcal{A}} \text{Reg}(T) \geq \Omega(KT^{2/3})$.*

For the lower bound proof of action-dependent bandits, we construct randomized problem instances and use information inequality to prove the lower bound. The full proof is provided in Appendix E.

For history-dependent bandits, we show that for a general class of reward function $\{r_k(\cdot)\}$ which satisfies the *proper* property (we will shortly elaborate this property), solving history-dependent bandits is as least as hard as solving action-dependent bandits. Armed with the above derived lower bound of action-dependent bandits, we can then conclude the lower bound of history-dependent bandits. *Proper Reward Function* is defined as below.

DEFINITION 6.2 (PROPER REWARD FUNCTION). *For any mixed strategy $\mathbf{p} \in \mathcal{P}$ and any $\mathbf{q} \in \mathcal{P}$ but $\mathbf{q} \neq \mathbf{p}$, the reward function $\{r_k(\cdot)\}$ is proper if the following holds,*

$$\sum_{p_k \in \mathbf{p}} p_k r_k(p_k) \geq \sum_{p_k \in \mathbf{p}, q_k \in \mathbf{q}} p_k r_k(q_k). \quad (13)$$

Our definition of proper reward function is coming from the literature of proper scoring rule [23], which is the tool for providing summary measures for the evaluation of probabilistic forecasts, by assigning a numerical score based on the predictive distribution and on the event or value that materializes. With the above defined proper reward at hand, we are now ready to show the regret lower bound for history-dependent bandits. Let \mathcal{I}^h denotes a history-dependent bandit instance whose reward function satisfies the above defined proper property, and \mathcal{I}^a denotes the associated action-dependent bandit instance whose reward function is the same as that in \mathcal{I}^h . Let $\mathbf{f}(t) = \{f_k^*(t)\}_{k \in [K]}$ be the discounted frequency at time t when the learner keeps deploying the

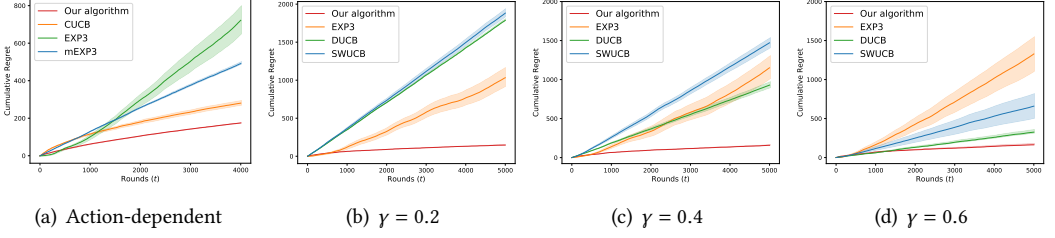


Fig. 2. (a): Behavior of the different algorithms for action-dependent bandits. (b)-(d): Behavior of the different algorithms for history-dependent bandits on different γ .

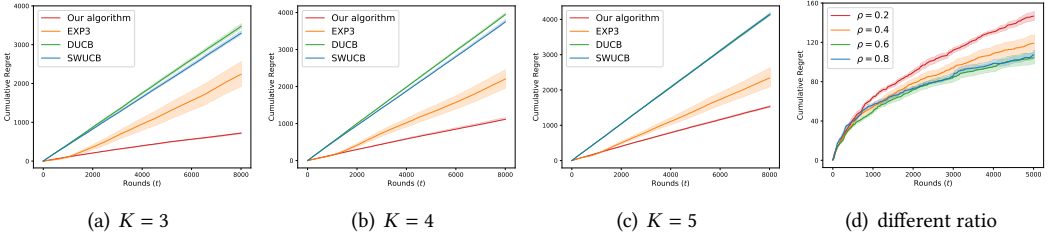


Fig. 3. (a)-(c): Behavior of the different algorithms on different K , the remaining parameters are the same with the simulations in comparison on different K . (d): The performance of our algorithms on different ratios, we set $K = 2$ and remaining parameters are also same as before.

best-in-hindsight strategy $\mathbf{p}^* = (p_1^*, \dots, p_K^*)$. Then we can show that

$$\begin{aligned}
 \mathbb{E}[\text{Reg}(T)|I^h] &= \sum_{t=1}^T \sum_k p_k^* r_k(f_k^*(t)) - \mathbb{E} \left[\sum_{t=1}^T \sum_k p_k(t) r_k(f_k(t)) \right] \\
 &\stackrel{(a)}{\geq} \sum_{t=1}^T \sum_k p_k^* r_k(f_k^*(t)) - \mathbb{E} \left[\sum_{t=1}^T \sum_k p_k(t) r_k(p_k(t)) \right] \\
 &\stackrel{(b)}{\geq} \sum_{t=1}^T \sum_k p_k^* r_k(p_k^*) - \mathbb{E} \left[\sum_{t=1}^T \sum_k p_k(t) r_k(p_k(t)) \right] - O \left(\frac{\gamma^2(1 - \gamma^{2T-2})K}{1 - \gamma^2} \right) \\
 &= \mathbb{E}[\text{Reg}(T)|I^a] - O \left(\frac{\gamma^2(1 - \gamma^{2T-2})K}{1 - \gamma^2} \right) = \Omega(KT^{2/3}),
 \end{aligned}$$

where the step (a) is due to the strict proper property of reward function, and the step (b) is due to the fact that the history-dependent bandit shares the same best-in-hindsight strategy as that in the action-dependent bandit and Lemma 5.3. By the regret reduction from the history-dependent bandit to the action-dependent bandit, we can conclude the lower bound of the history-dependent bandit.

7 NUMERIC EXPERIMENTS

We conducted a series of simulations to empirically evaluate the performance of our proposed solution with a set of baselines.

7.1 Evaluations for action-dependent bandits

We first evaluate our proposed algorithm on action-dependent bandits against the following state-of-the-art bandit algorithms.

- **EXP3**: One natural baseline is applying EXP3 [4] on the space of base arms. While EXP3 is designed for adversarial rewards, it is competing with the best fixed arm in hindsight and might not work well in our setting since the optimal strategy is randomized.
- **EXP3-Meta-Arm (mEXP3)**: To make a potentially more fair comparison, we also implement EXP3 on the meta-arm space. We denote it as mEXP3 in the following discussion.
- **CUCB [11, 53]**: This algorithm is designed to solve combinatorial semi-bandit problem, which chooses m arms out of M arms at each round and receives only the rewards of selected arms. Mapping to our setting, $M = K/\epsilon$ represents the total number of discretized arms, $m = K$ is the number of base arms, and the selection of m arms is constrained to satisfy the probability simplex constraint.

In the simulations, we set $K = 2$ for simplicity. Moreover, $r_k(p_k)$ is chosen such that $r_k(p_k)$ is maximized when $0 < p_k < 1$. In particular, we define $r_k(p_k)$ as a scaled Gaussian function : $r_k(p_k) = f(p_k|\tau_k, 0.5)/C_k$, where $f(x|\tau, \sigma^2)$ is the pdf of Gaussian distribution with the mean τ and variance σ^2 , and $C_k = f(\tau_k|\tau_k, \sigma^2)$ is a constant ensuring $r_k(p_k) \in [0, 1], \forall p_k \in [0, 1]$. For each arm k , τ_k is uniformly draw from 0.45 to 0.55 and the instantaneous reward is drawn from a Bernoulli distribution with the mean of $r_k(p_k(t))$, i.e., $\tilde{r}_t(p_k(t)) \sim \text{Bernoulli}(r_k(p_k(t)))$. And the ratio ρ is set to 0.2. For each algorithm we perform 40 runs for each of independent 40 values of the corresponding parameter, and we report the averaged results of these independent runs, where the error bars correspond to ± 2 standard deviations.

The results, shown in Figure 2(a), demonstrate that our algorithm significantly outperforms the baselines. As expected, mEXP3 works better than EXP3 algorithm when T is large, since the former searches the optimal strategies in the meta arm space. Our algorithm outperforms mEXP3 and CUCB since we utilize the problem structure, which reduces the amount of explorations.

7.2 Evaluations for history-dependent bandits

We now evaluate our proposed algorithm for history-dependent bandits via comparing against the following baselines from non-stationary bandits. Note that, while CUCB performs reasonably well in action-dependent case, it does not apply in history-dependent case, since we cannot *select* the time-discounted frequency (which maps to the arm in CUCB) as required in CUCB.

- **Discounted UCB [20, 34]**: Discounted UCB (DUCB) is an adaptation of the standard UCB policies that relies on a discount factor $\gamma_{\text{DUCB}} \in (0, 1)$. This method constructs an UCB : $\bar{r}_t(k, \gamma_{\text{DUCB}}) + c_t(k, \gamma_{\text{DUCB}})$ for the instantaneous expected reward, where the confidence is defined as $c_t(k, \gamma_{\text{DUCB}}) = 2\sqrt{\frac{\xi \ln(n_t)}{N_t(k, \gamma_{\text{DUCB}})}}$, for an appropriate parameter ξ , $N_t(k, \gamma_{\text{DUCB}}) = \sum_{s=1}^t \gamma_{\text{DUCB}}^{t-s} \mathbb{1}_{(a_s=k)}$, and the discounted empirical average is given by $\bar{r}_t(k, \gamma_{\text{DUCB}}) = \frac{1}{N_t(k, \gamma_{\text{DUCB}})} \cdot \sum_{s=1}^t \gamma_{\text{DUCB}}^{t-s} \tilde{r}_s(k) \mathbb{1}_{(a_s=k)}$.
- **Sliding-Window UCB [20]**: Sliding-Window UCB (SWUCB) is a modification of DUCB, instead of averaging the rewards over all past with a discount factor, SWUCB relies on a local empirical average of the observed rewards, for example, using only the τ last plays. Specifically, this method

also constructs an UCB : $\bar{r}_t(k, \tau) + c_t(k, \tau)$ for the instantaneous expected reward. The local empirical average is given by $\bar{r}_t(k, \tau) = \frac{1}{N_t(k, \tau)} \sum_{s=t-\tau+1}^t \tilde{r}_s(k) \mathbb{1}_{(a_s=k)}$, $N_t(k, \tau) = \sum_{s=t-\tau+1}^t Y_{\text{DUCB}}^{t-s} \mathbb{1}_{(a_s=k)}$

and the confidence interval is defined as $c_t(k, \gamma_{\text{DUCB}}) = 2\sqrt{\frac{\xi \ln(\min(t, \tau))}{N_t(k, \tau)}}$.

We use *grid searches* to determine the algorithms' parameters. For example, in DUCB, the discount factor was chosen from $\gamma_{\text{DUCB}} \in \{0.5, 0.6, \dots, 0.9\}$, while the window size of SWUCB was chosen from $\tau \in \{10^2, \dots, 5 \times 10^2\}$. Besides above algorithms, we also implement the celebrated non-stationary bandit algorithm EXP3.

We chose K and $r_k(p_k)$ to be the same as the experiments in action-dependent case. And the discount factor is chosen as $\gamma_{\text{DUCB}} = 0.8$ and the window size for SWUCB is chosen as 200 via the grid search, and ξ is set to 1. We examine the algorithm performances under different γ (the parameter in time-discounted frequency), with smaller γ indicating that arm rewards are more influenced by recent actions. As seen in Figures 2(b)-2(d), our algorithm outperforms all baselines in all γ but the improvement is more significant with small γ . This is possibly due to that most non-stationary bandit algorithms have been focusing on settings in which the change of arm rewards over time is not dramatic.

We also examine our algorithm with larger number of base arms K with comparing to above baseline algorithms and the performance of our algorithm on different ratios ρ . The results are presented in Figure 3 and show that our algorithm consistently performs better than other baselines when K goes large. The results also suggest that our algorithm is not sensitive to different ρ , though one could see the regret is slightly lower when ρ is increasing, which is expected from our regret bound.

8 CONCLUSION AND FUTURE WORK

We explore a multi-armed bandit problem in which actions have delayed impacts to the arm rewards. We propose algorithms that achieve a regret of $\tilde{O}(KT^{2/3})$ and provide a matching lower regret bound of $\Omega(KT^{2/3})$. Our results complement the bandit literature by adding techniques to deal with action history dependent biases in bandits. While our model have its limitations, it captures an important but relatively under-explored angle in algorithmic fairness, the long-term impact of actions in sequential learning settings. We hope our study will open more discussions along this direction.

REFERENCES

- [1] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica*, May, 23:2016, 2016.
- [2] Jean-Yves Audibert and Sébastien Bubeck. Regret bounds and minimax policies under partial monitoring. *Journal of Machine Learning Research*, 11:2785–2836, 2010.
- [3] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.
- [4] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.
- [5] Alexander Bartik and Scott Nelson. Credit reports as resumes: The incidence of pre-employment credit screening. 2016.
- [6] Robert Bartlett, Adair Morse, Richard Stanton, and Nancy Wallace. Consumer-lending discrimination in the era of fintech. *Unpublished working paper. University of California, Berkeley*, 2018.
- [7] Yahav Bechavod, Katrina Ligett, Aaron Roth, Bo Waggoner, and Steven Z Wu. Equal opportunity in online classification with partial feedback. In *Advances in Neural Information Processing Systems*, pages 8972–8982, 2019.

- [8] Omar Besbes, Yonatan Gur, and Assaf Zeevi. Stochastic multi-armed-bandit problem with non-stationary rewards. In *Advances in neural information processing systems*, pages 199–207, 2014.
- [9] Nicolo Cesa-Bianchi and Gábor Lugosi. Combinatorial bandits. *Journal of Computer and System Sciences*, 78(5): 1404–1422, 2012.
- [10] Wei Chen, Wei Hu, Fu Li, Jian Li, Yu Liu, and Pinyan Lu. Combinatorial multi-armed bandit with general reward functions. In *Advances in Neural Information Processing Systems*, pages 1659–1667, 2016.
- [11] Wei Chen, Yajun Wang, Yang Yuan, and Qinshi Wang. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *The Journal of Machine Learning Research*, 17(1):1746–1778, 2016.
- [12] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.
- [13] Richard Combes, Mohammad Sadegh Talebi Mazraeh Shahi, Alexandre Proutiere, et al. Combinatorial bandits revisited. In *Advances in Neural Information Processing Systems*, pages 2116–2124, 2015.
- [14] Richard Combes, Alexandre Proutière, and Alexandre Fauquette. Unimodal bandits with continuous arms: Order-optimal regret without smoothness. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 4(1): 1–28, 2020.
- [15] Corinna Cortes, Giulia DeSalvo, Vitaly Kuznetsov, Mehryar Mohri, and Scott Yang. Discrepancy-based algorithms for non-stationary rested bandits. *arXiv preprint arXiv:1710.10657*, 2017.
- [16] Bo Cowgill and Catherine E Tucker. Economics, fairness and algorithmic bias. *preparation for: Journal of Economic Perspectives*, 2019.
- [17] Bo Cowgill and Eric Zitzewitz. Incentive effects of equity compensation: Employee level evidence from google. *Dartmouth Department of Economics working paper*, 2009.
- [18] Santiago Duran and Ina Maria Verloop. Asymptotic optimal control of markov-modulated restless bandits. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 2(1):1–25, 2018.
- [19] Andreas Fuster, Paul Goldsmith-Pinkham, Tarun Ramadorai, and Ansgar Walther. Predictably unequal? the effects of machine learning on credit markets. 2018.
- [20] Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In *International Conference on Algorithmic Learning Theory*, pages 174–188, 2011.
- [21] Andrew Gelman, Jeffrey Fagan, and Alex Kiss. An analysis of the new york city police department’s “stop-and-frisk” policy in the context of claims of racial bias. *Journal of the American statistical association*, 102(479):813–823, 2007.
- [22] Stephen Gillen, Christopher Jung, Michael Kearns, and Aaron Roth. Online learning with an unknown fairness metric. In *Advances in Neural Information Processing Systems*, pages 2600–2609, 2018.
- [23] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.
- [24] Sharad Goel, Justin M Rao, Ravi Shroff, et al. Precinct or prejudice? understanding racial disparities in new york city’s stop-and-frisk policy. *The Annals of Applied Statistics*, 10(1):365–394, 2016.
- [25] Swati Gupta and Vijay Kamble. Individual fairness in hindsight. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 805–806, 2019.
- [26] Hoda Heidari, Vedant Nanda, and Krishna P Gummadi. On the long-term impact of algorithmic decision policies: Effort unfairness and feature segregation through social learning. *arXiv preprint arXiv:1903.01209*, 2019.
- [27] Lily Hu and Yiling Chen. A short-term intervention for long-term fairness in the labor market. In *Proceedings of the 2018 World Wide Web Conference*, pages 1389–1398, 2018.
- [28] Matthew Joseph, Michael Kearns, Jamie H Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. In *Advances in Neural Information Processing Systems*, pages 325–333, 2016.
- [29] Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Online learning under delayed feedback. In *International Conference on Machine Learning*, pages 1453–1461, 2013.
- [30] Sampath Kannan, Aaron Roth, and Juba Ziani. Downstream effects of affirmative action. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 240–248, 2019.
- [31] Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Cass R Sunstein. Discrimination in the age of algorithms. *Journal of Legal Analysis*, 10, 2018.
- [32] Robert Kleinberg and Nicole Immorlica. Recharging bandits. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science*, pages 309–319, 2018.

- [33] Robert Kleinberg, Aleksandrs Slivkins, and Eli Upfal. Multi-armed bandits in metric spaces. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 681–690, 2008.
- [34] Levente Kocsis and Csaba Szepesvári. Discounted ucb. In *2nd PASCAL Challenges Workshop*, volume 2, 2006.
- [35] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.
- [36] Nir Levine, Koby Crammer, and Shie Mannor. Rotting bandits. In *Advances in neural information processing systems*, pages 3074–3083, 2017.
- [37] Fengjiao Li, Jia Liu, and Bo Ji. Combinatorial sleeping bandits with fairness constraints. *IEEE Transactions on Network Science and Engineering*, 2019.
- [38] Lydia T Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. Delayed impact of fair machine learning. In *International Conference on Machine Learning*, pages 3150–3158, 2018.
- [39] Lydia T Liu, Ashia Wilson, Nika Haghtalab, Adam Tauman Kalai, Christian Borgs, and Jennifer Chayes. The disparate equilibria of algorithmic decision making when individuals invest rationally. *arXiv preprint arXiv:1910.04123*, 2019.
- [40] Yang Liu, Goran Radanovic, Christos Dimitrakakis, Debmalya Mandal, and David C Parkes. Calibrated fairness in bandits. *Proceedings of the 4th Workshop on Fairness, Accountability, and Transparency in Machine Learning*, 2017.
- [41] Stefan Magureanu, Richard Combes, and Alexandre Proutiere. Lipschitz bandits: Regret lower bound and optimal algorithms. In *Conference on Learning Theory*, pages 975–999, 2014.
- [42] Hussein Mouzannar, Mesrob I Ohannessian, and Nathan Srebro. From fair decision making to social equality. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 359–368, 2019.
- [43] Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453, 2019.
- [44] Ronald Ortner, Daniil Ryabko, Peter Auer, and Rémi Munos. Regret bounds for restless markov bandits. In *International Conference on Algorithmic Learning Theory*, pages 214–228, 2012.
- [45] Ciara Pike-Burke, Shipra Agrawal, Csaba Szepesvari, and Steffen Grunewalder. Bandits with delayed, aggregated anonymous feedback. In *International Conference on Machine Learning*, pages 4105–4113, 2018.
- [46] Sven Schmit and Carlos Riquelme. Human interaction with recommendation systems. In *International Conference on Artificial Intelligence and Statistics*, pages 862–870, 2018.
- [47] Julien Seznec, Andrea Locatelli, Alexandra Carpentier, Alessandro Lazaric, and Michal Valko. Rotting bandits are no harder than stochastic ones. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2564–2572, 2019.
- [48] Aleksandrs Slivkins. Contextual bandits with similarity information. *The Journal of Machine Learning Research*, 15(1): 2533–2568, 2014.
- [49] Aleksandrs Slivkins and Eli Upfal. Adapting to a changing environment: the brownian restless bandits. In *Conference on Learning Theory*, pages 343–354, 2008.
- [50] Cem Tekin and Mingyan Liu. Online algorithms for the multi-armed bandit problem with markovian rewards. In *Proceedings of the 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2010.
- [51] Ina Maria Verloop et al. Asymptotically optimal priority policies for indexable and nonindexable restless bandits. *The Annals of Applied Probability*, 26(4):1947–1995, 2016.
- [52] Claire Vernade, Olivier Cappé, and Vianney Perchet. Stochastic bandit models for delayed conversions. *arXiv preprint arXiv:1706.09186*, 2017.
- [53] Qinshi Wang and Wei Chen. Improving regret bounds for combinatorial semi-bandits with probabilistically triggered arms and its applications. In *Advances in Neural Information Processing Systems*, pages 1161–1171, 2017.
- [54] Xueru Zhang, Mohammadmahdi Khaliligarekani, Cem Tekin, et al. Group retention when using machine learning in sequential decision making: the interplay between user dynamics and fairness. In *Advances in Neural Information Processing Systems*, pages 15243–15252, 2019.

Supplementary Materials for “Bandit Learning with Delayed Impact of Actions”

Setup	Notations	Explanations
basic setup	$K; T$	the number of (base) arms; time horizon
	$k; t$	arm index; time round t
	\mathcal{P}	probability simplex
	ϵ	discretization parameter
	\mathcal{P}_ϵ	probability simplex after discretization with ϵ
	\mathbf{p}	meta arm/mixed strategy
	\mathbf{p}^*	optimal meta arm
	$r_k(\cdot)$	expected reward function of arm k
	λ_k	arm k 's hyperparameter on tradeoff the expected reward and fairness
	a_t	the realized arm at time t
	$p_k, p_k(t)$	the probability on pulling arm k , at time t
	L_k^r	the Lipschitz constant of r_k
	L_k^π	the Lipschitz constant of π_k
	L^*	the maximum $L^* = \max(1 + L_k^r + \lambda_k L_k^\pi)$
	\tilde{r}_t	the realized reward at time t
	\tilde{r}_t	the importance weighted reward
	$\bar{r}_t(p_k)$	the empirical reward mean of discretized arm p_k
	$\text{Reg}(t)$	cumulative regret till time t
action dependent bandit	$\Delta_{\mathbf{p}}$	the badness of meta arm \mathbf{p} t
	$\mathbf{p}(t) = \{p_k(t)\}_{k \in [K]}$	the meta arm deployed in time round t
	$\mathbf{f}(t) = \{f_k(t)\}_{k \in [K]}$	actions impact function
	$n_t(p_k)$	the number of times when pulling arm k with prob p_k till time t
	$N_t(\mathbf{p})$	number of pulls of meta arm \mathbf{p} till time t
	$S(p_k)$	the set of all meta arms which contain p_k
	$N_t(S(p_k))$	total number of pulls of all meta arms in $S(p_k)$
history dependent bandit	$p_{\min}(\mathbf{p})$	$p_{\min}(\mathbf{p}) = \arg \min_{p \in \mathbf{p}} n_t(p_k)$ for some t
	$\bar{U}_t(\mathbf{p})$	the empirical reward mean of meta arm \mathbf{p}
	$U(\mathbf{p})$	the expected reward of meta arm \mathbf{p}
	γ	time-discounted factor
	L	the length of phase
	s_a	the length of approaching stage
	ρ	the ratio of estimation stages over each phase
	m	the index of each phase
	$\Gamma_t(p_k)$	the set of all indexes which arm k is pulled with prob p_k
	$\mathbf{p}^{(y)} = \{p_k^{(y)}\}_{k \in [K]}$	the time-discounted empirical frequency
	$\bar{r}_m^e(p_k)$	the empirical reward mean of discretized arm p_k in all estimation stages
	$\bar{U}_t^e(\mathbf{p})$	the empirical reward mean of meta arm \mathbf{p} in all estimation stages
	$n_m^e(p_k)$	the number of rounds that arm k is pulled with prob p_k in the first m phases

Table 1. The summary of notations.

A LAGRANGIAN FORMULATION

To simplify the presentation, we fix a time t and drop the dependency on t in the notations.

DEFINITION A.1. *The Lagrangian $\mathcal{L} : \mathcal{P} \times \Lambda^2 \rightarrow \mathbb{R}$ of our problem can be formulated as:*

$$\mathcal{L}(\mathbf{p}, \lambda) := \sum_{k=1}^K p_k r_k(f_k) - \sum_{c=1}^{\binom{K}{2}} \lambda_c^+ (\pi_{i_c}(f_{i_c}) - \pi_{j_c}(f_{j_c}) - \tau) - \sum_{c=1}^{\binom{K}{2}} \lambda_c^- (\pi_{j_c}(f_{j_c}) - \pi_{i_c}(f_{i_c}) - \tau), \quad (14)$$

where $\lambda^+, \lambda^- \in \Lambda \subseteq \mathbb{R}_+^{\binom{K}{2}}$. The notation $(i_c, j_c) \in \{(i, j)_{1 \leq i < j \leq K}\}$ is a pair of combination and $c \in [K(K-1)/2]$ is the index of each pair of this combination.

The problem then reduces to jointly maximize over $\mathbf{p} \in \mathcal{P}$ and minimize over $\lambda^+, \lambda^- \in \Lambda$. Rearranging and with a slight abuse of notations, we have the following equivalent optimization problem:

$$\max_{\mathbf{p} \in \mathcal{P}} \min_{\lambda^+, \lambda^-} \sum_{k=1}^K p_k(t) r_k(f_k(t)) + \lambda_k \pi_k(f_k(t)) + \tau \sum_{c=1}^{\binom{K}{2}} (\lambda_c^+ + \lambda_c^-), \quad (15)$$

where $\lambda_k := -\sum_{c:i_c=k} (\lambda_c^+ - \lambda_c^-) + \sum_{c:j_c=k} (\lambda_c^+ - \lambda_c^-)$. Due to the uncertainty of reward function $r_k(\cdot)$ and fairness measure $\pi_k(\cdot)$ (recall that our fairness criteria is defined as the parity of socio-economic status cross different groups, which we can only observe the realization drawn from an unknown distribution), we treat the above optimization problem as a hyperparameter optimization: similar to choosing hyperparameters (the Lagrange multipliers: λ^+ and λ^-) based on a validation set in machine learning tasks. Therefore, given a fixed set of λ^+ and λ^- , the problem in (15) can be reduced to the following:

$$\max_{\mathbf{p} \in \mathcal{P}} \sum_{k=1}^K p_k(t) \cdot r_k(f_k(t)) + \lambda_k \cdot \pi_k(f_k(t)). \quad (16)$$

B NEGATIVE RESULTS

In this section, we show that an online algorithm which ignores its action's impact would suffer linear regret. We consider two general bandit algorithms: TS (Thompson Sampling) and a mean-converging family of algorithms (which includes UCB-like algorithms). These are the two most popular and robust bandit algorithms that can be applied to a wide range of scenarios. We prove the negative results respectively. In particular, we construct problem instances that could result in linear regret if the deployed algorithm ignore the action's impact.

EXAMPLE 1. *Considering the following Bernoulli bandit instance with two arms, indexed by arm 1 and arm 2, i.e., $K = 2$. For any $\epsilon \in [0, 1/2)$, define the expected reward of each arm as follows:*

- *arm 1: $r_1(p) = p/(1 - \epsilon) \cdot \mathbb{1}(p \leq 1 - \epsilon) + (2 - \epsilon - p) \cdot \mathbb{1}(p \geq 1 - \epsilon)$, $\forall p \in [0, 1]$*
- *arm 2: $r_2(p) = p/(2\epsilon) \cdot \mathbb{1}(p \leq \epsilon) + (-\frac{1}{2}p + \frac{1}{2}(1 + \epsilon)) \cdot \mathbb{1}(p \geq \epsilon)$, $\forall p \in [0, 1]$*

It is easy to see that $\mathbf{p}^* = \{1 - \epsilon, \epsilon\}$ is the optimal strategy for the above bandit instance.

We first prove the negative result of Thompson Sampling using the above example. The Thompson Sampling algorithm can be summarized as below.

Algorithm 4 Thompson Sampling for Bernoulli bandits

- 1: $S_i = 0, F_i = 0$.
 - 2: **for** $t = 1, 2, \dots$, **do**
 - 3: For each arm $i = 1, 2$, sample $\theta_i(t)$ from the $\text{Beta}(S_i + 1, F_i + 1)$ distribution.
 - 4: Play arm $a_t := \arg \max_i \theta_i(t)$ and observe reward \tilde{r}_t .
 - 5: If $\tilde{r}_t = 1$, then $S_{a_t} = S_{a_t} + 1$, else $F_{a_t} = F_{a_t} + 1$.
 - 6: **end for**
-

LEMMA B.1. *For the reward structure defined in Example 1, Thompson Sampling would suffer linear regret if it doesn't consider the action's impact it deploys at every time round, namely, it takes the sample mean as the true mean reward of each arm.*

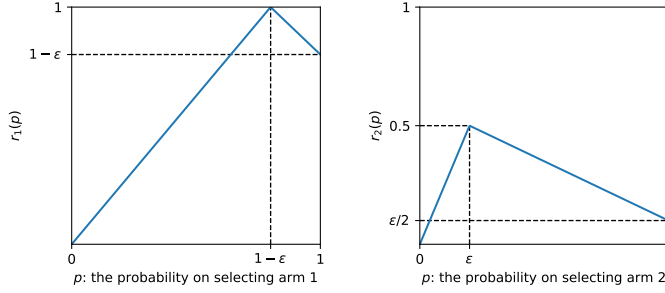


Fig. 4. The expected reward defined in Example 1: arm 1 (left) and arm 2 (right).

Before we proceed, we first prove the following strong law of large numbers in Beta distribution. We note that the below two lemmas are not new results and can be found in many statistical books. We provide proofs here for the sake of making the current work self-contained.

LEMMA B.2. Consider the Beta distribution $\text{Beta}(a\alpha + 1, b\alpha + 1)$ whose pdf is defined as $f(x, \alpha) = \frac{[x^a(1-x)^b]^\alpha}{B(a\alpha+1, b\alpha+1)}$, where $B(\cdot)$ is the beta function, then for any positive (a, b) such that $a + b = 1$, when $\alpha \rightarrow \infty$, the limit of $f(x, \alpha)$ can be characterized by Dirac delta function $\delta(x - a)$.

PROOF. By Stirling's approximation, we can write the asymptotics of beta function as follows:

$$B(x, y) \sim \sqrt{2\pi} \frac{x^{x-0.5} y^{y-0.5}}{(x+y)^{x+y-0.5}}.$$

Thus, when $\alpha \rightarrow \infty$, i.e., for large $a\alpha + 1$ and $b\alpha + 1$, we can approximate the pdf $f(x, \alpha)$ in the following:

$$f(x, \alpha) \sim \sqrt{\frac{a+2}{2\pi ab}} h^\alpha(x),$$

where $h(x) := (x/a)^a \left(\frac{1-x}{b}\right)^b$. It's easy to see that $h(x)$ has a unique maximum at a , by invoking Lemma B.3 will complete the proof. \square

LEMMA B.3. Let $h : [0, 1] \rightarrow \mathbb{R}^+$ be any bounded measurable non-negative function with a unique maximum at x^* , and suppose h is continuous at x^* . For $\lambda > 0$ define $h_\lambda(x) = C_\lambda h^\lambda(x)$ where C_λ normalizes such that $\int_0^1 h_\lambda(x) dx = 1$. Consider any continuous function f defined on $[0, 1]$ and $\epsilon > 0$, then we have $\lim_{\lambda \rightarrow \infty} \int_{h(x) \leq h(x^*) - \epsilon} h_\lambda(x) f(x) dx = 0$ and $\lim_{\lambda \rightarrow \infty} \int_0^1 h_\lambda(x) f(x) dx = f(x^*)$.

PROOF. For any $\delta > 0$, we have

$$\begin{aligned} & \left| \int_0^1 h_\lambda(x) f(x) dx - f(x^*) \right| \\ &= \left| \int_0^1 h_\lambda(x) (f(x) - f(x^*)) dx \right| \\ &\leq \left| \int_{|x-x^*| \leq \delta} h_\lambda(x) (f(x) - f(x^*)) dx \right| + \left| \int_{|x-x^*| > \delta} h_\lambda(x) (f(x) - f(x^*)) dx \right| \\ &\leq \left| \int_{|x-x^*| \leq \delta} h_\lambda(x) (f(x) - f(x^*)) dx \right| + \max |f(x) - f(x^*)| \int_{|x-x^*| > \delta} h_\lambda(x) dx. \end{aligned}$$

For any $\delta > 0$, and due to the continuous property of f on x^* , which further implies that there exists a constant $c > 0$ such that $|f(x) - f(x^*)| < \delta/2$ whenever $|x - x^*| < c$. Thus, given $c > \delta$, we have

$$\left| \int_0^1 h_\lambda(x) f(x) dx - f(x^*) \right| \leq \epsilon/2 + \max |f(x) - f(x^*)| \left| \int_{|x-x^*|>\delta} h_\lambda(x) dx \right|.$$

It suffices to show that the second term in RHS of above inequality will converge to 0 as $\lambda \rightarrow \infty$. Let $\|h\|_{\infty, \delta}$ denote the L^∞ norm of h when h is restricted to $\{|x - x^*| > \delta\}$. Note that for any nonnegative integrable functions h , we have

$$\lim_{\lambda \rightarrow \infty} \left(\int_0^1 h^\lambda(x) dx \right)^{1/\lambda} = \|h\|_\infty.$$

Recall the definition of $C_\lambda = \frac{1}{\int_0^1 h^\lambda(x) dx}$, thus, we have $\lim_{\lambda \rightarrow \infty} C_\lambda^{1/\lambda} = \frac{1}{\|h\|_\infty}$, which immediately showing that

$$\left(\int_{|x-x^*|>\delta} h_\lambda(x) dx \right)^{1/\lambda} = C_\lambda^{1/\lambda} \left(\int_{|x-x^*|>\delta} h^\lambda(x) dx \right)^{1/\lambda},$$

which further implies that $\|h\|_{\infty, \delta} / \|h\|_\infty < 1$. Thus, there must exist λ_0 such that $\forall \lambda > \lambda_0$,

$$\left(\int_{|x-x^*|>\delta} h_\lambda(x) dx \right)^{1/\lambda} < \gamma < 1. \quad (17)$$

Since $\gamma < 1$, we then have $\lim_{\lambda \rightarrow \infty} \gamma^\lambda = 0$, this implies the second term of RHS of (17) converging to 0 as $\lambda \rightarrow \infty$. \square

We now ready to prove Lemma B.1.

PROOF. We prove this by contradiction. Let $\text{Reg}(T)$ denote the expected regret incurred by TS up to time round T , and $N_t(\mathbf{p}) = \sum_{s=1}^t \mathbb{1}_{\mathbf{p}(s)=\mathbf{p}}$ denote the number of rounds when the algorithm deploys the (mixed) strategy $\mathbf{p} \in \Delta_K$. Furthermore, let $S_i(t)$ (resp. $F_i(t)$) denote the received 1_s (resp. 0_s) of arm i up to time round t . Recall that in Thompson Sampling, we have $\mathbb{P}(a_t = 1) = \mathbb{P}(\theta_1(t) > \theta_2(t))$. By the reward function defined in Example 1, it's immediate to see that

$$S_1(T) \geq (1 - \epsilon)N_T(\mathbf{p}^*); \quad F_1(T) \leq T - N_T(\mathbf{p}^*); \quad S_2(T) \geq 0.5\epsilon N_T(\mathbf{p}^*); \quad F_2(T) \geq 0.5\epsilon N_T(\mathbf{p}^*).$$

Now suppose Thompson Sampling achieves sublinear regret, i.e., $\text{Reg}(T) = o(T)$, which implies following

$$\lim_{T \rightarrow \infty} \frac{T - N_T(\mathbf{p}^*)}{T} = 0.$$

Thus, by the strong law of large numbers and invoking Lemma B.2, the sample $\theta_1(T+1) \sim \text{Beta}(S_1(T), F_1(T))$ and $\theta_2(T+1) \sim \text{Beta}(S_2(T), F_2(T))$ will converge as follows:

$$\lim_{T \rightarrow \infty} \theta_1(T+1) = 1; \quad \lim_{T \rightarrow \infty} \theta_2(T+1) = 0.5. \quad (18)$$

Then it's almost surely that $\lim_{T \rightarrow \infty} \mathbb{P}(a_{T+1} = 1) = \lim_{T \rightarrow \infty} \mathbb{P}(\theta_1(T+1) > \theta_2(T+1)) = 1$. This leads to following holds for sure

$$S_1(s+1) = S_1(s) + 1, \forall s > T.$$

Thus, consider the regret incurred from the $(T + 1)$ -th round to $(2T)$ -th round, the regret will be

$$\text{Reg}(2T) - \text{Reg}(T) = \sum_{s=T+1}^{2T} U(\mathbf{p}(s)) = 0.5T\epsilon,$$

where the second equality follows that $\mathbf{p}(s) = (1, 0)$ holds almost surely from $T + 1$ to $2T$. This shows that $\lim_{T \rightarrow \infty} \frac{\mathbb{E}[\text{Reg}(2T)]}{2T} = \epsilon/4$, which contradicts that the algorithm achieves the sublinear regret. \square

We now show that a general class of algorithms, which are based on *mean-converging*, will suffer linear regret if it ignores the action's impact. This family of algorithms includes UCB algorithm in classic MAB problems.

DEFINITION B.4 (MEAN-CONVERGING ALGORITHM [46]). Define $I_k(t) = \{s : a_s = k, s < t\}$ as the set of time rounds such the arm k is chosen. Let $\bar{r}_k(t) = \frac{1}{|I_k(t)|} \sum_{s \in I_k(t)} \tilde{r}_s$ be the empirical mean of arm k up to time t . The mean-converging algorithm \mathcal{A} assigns $s_k(t)$ for each arm k if following holds true:

- $s_k(t)$ is the function of $\{\tilde{r}_s : s \in I_k(t)\}$ and time t ;
- $\mathbb{P}(s_k(t) = \bar{r}_k(t)) = 1$ if $\liminf_t \frac{|I_k(t)|}{t} > 0$.

LEMMA B.5. For the reward structure defined in Example 1, the mean-converging Algorithm will suffer linear regret if it mistakenly take the sample mean as the true mean reward of each arm.

PROOF. We prove above lemma by contradiction. Let $N_t^{\mathcal{A}}(\mathbf{p})$ denote the number of plays with deploying the strategy \mathbf{p} by algorithm \mathcal{A} till time t . Suppose a mean-converging Algorithm \mathcal{A} achieves sublinear regret, then it must have $\lim_{T \rightarrow \infty} N_T^{\mathcal{A}}(\mathbf{p}^*)/T > 0$ and $\lim_{T \rightarrow \infty} (T - N_T^{\mathcal{A}}(\mathbf{p}^*))/T = o(T)$. By the definition of mean-converging algorithm and recall the reward structure defined in Example 1, the score $s_T(1)$ assigned to arm 1 by the algorithm \mathcal{A} must be converging to 1, and the score of $s_T(2)$ assigned to arm 2 must be converging to 0.5. By the strong law of large numbers, it suffices to show that $\mathbb{P}(\mathbf{p}(t) = \{1, 0\}) = 1, \forall t \geq T + 1$, which implies the algorithm \mathcal{A} would suffer linear regret after T time rounds and thus completes the proof. \square

C MISSING PROOFS FOR ACTION-DEPENDENT BANDITS

C.1 The naive method that directly utilize techniques from Lipschitz bandits

We first give a naive approach which directly applies Lipschitz bandit technique to our action-dependent setting. Recall that each meta arm \mathbf{p} specifies the probability $p_k \in [0, 1]$ for choosing each base arm k . We uniformly discretize each p_k into intervals of a fixed length ϵ , with carefully chosen ϵ such that $1/\epsilon$ is an positive integer. Let \mathcal{P}_ϵ be the space of discretized meta arms, i.e., for each $\mathbf{p} = \{p_1, \dots, p_K\} \in \mathcal{P}_\epsilon$, $\sum_{k=1}^K p_k = 1$ and $p_k \in \{0, \epsilon, 2\epsilon, \dots, 1\}$ for all k . We then run standard bandit algorithms on the finite set \mathcal{P}_ϵ .

There is a natural trade-off on the choice of ϵ , which controls the complexity of arm space and the discretization error. show that, with appropriately chosen ϵ , this approach can achieve sublinear regret (with respect to the optimal arm in the non-discretized space \mathcal{P}).

LEMMA C.1. Let $\epsilon = \Theta\left(\left(\frac{\ln T}{T}\right)^{\frac{1}{K+1}}\right)$. Running a bandit algorithm which achieves optimal regret $O(\sqrt{|\mathcal{P}_\epsilon|T \ln T})$ on the strategy space \mathcal{P}_ϵ attains the following regret (w.r.t. the optimal arm in non-discretized \mathcal{P}): $\text{Reg}(T) = O\left(T^{\frac{K}{K+1}} (\ln T)^{\frac{1}{K+1}}\right)$.

PROOF. As mentioned, we *uniformly* discretize the interval $[0, 1]$ of each arm into interval of a fixed length ϵ . The strategy space will be reduced as \mathcal{P}_ϵ , which we use this as an approximation for the full set \mathcal{P} . Then the original infinite action space will be reduced as finite \mathcal{P}_ϵ , and we run an off-the-shelf MAB algorithm \mathcal{A} , such as UCB1 or Successive Elimination, that only considers these actions in \mathcal{P}_ϵ . Adding more points to \mathcal{P}_ϵ makes it a better approximation of \mathcal{P} , but also increases regret of \mathcal{A} on \mathcal{P}_ϵ . Thus, \mathcal{P}_ϵ should be chosen so as to optimize this tradeoff. Let $\mathbf{p}_\epsilon^* := \sup_{\mathbf{p} \in \mathcal{P}_\epsilon} \sum_{k=1}^K p_k(t) r_k(p_k(t))$ denote the best strategy in discretized space \mathcal{P}_ϵ . At each round, the algorithm \mathcal{A} can only hope to approach expected reward $U(\mathbf{p}_\epsilon^*)$, and together with additionally suffering *discretization error*:

$$\text{DE}_\epsilon := U(\mathbf{p}^*) - U(\mathbf{p}_\epsilon^*).$$

Then the expected regret of the entire algorithm is:

$$\begin{aligned} \text{Reg}(T) &= T \cdot U(\mathbf{p}^*) - \text{Reward}(\mathcal{A}) \\ &= T \cdot U(\mathbf{p}_\epsilon^*) - \text{Reward}(\mathcal{A}) + T(U(\mathbf{p}^*) - U(\mathbf{p}_\epsilon^*)) \\ &= \mathbb{E}[\text{Reg}_\epsilon(T)] + T \cdot \text{DE}_\epsilon, \end{aligned}$$

where $\text{Reward}(\mathcal{A})$ is the total reward of the algorithm, and $\text{Reg}_\epsilon(T)$ is the regret relative to $U(\mathbf{p}_\epsilon^*)$. If \mathcal{A} attains optimal regret $O(\sqrt{NT \ln T})$ on any problem instance with time horizon T and N arms, then,

$$\text{Reg}(T) \leq O(\sqrt{|\mathcal{P}_\epsilon| T \ln T}) + T \cdot \text{DE}_\epsilon.$$

Thus, we need to choose ϵ to get the optimal trade-off between the size of \mathcal{P}_ϵ and its discretization error. Recall that $r_k(\cdot)$ is Lipschitz-continuous with the constant of L_k , thus, we could bound the DE_ϵ by restricting \mathbf{p}_ϵ^* to be nearest w.r.t \mathbf{p}^* . Let $L^* = \max_{k \in [K]} (1 + L_k)$, then it's easy to see that

$$\text{DE}_\epsilon = \Omega(KL^* \epsilon).$$

Thus, the total regret can be bounded above from:

$$\text{Reg}(T) \leq O\left(\sqrt{(1/\epsilon + 1)^{K-1} T \ln T}\right) + \Omega(TKL^* \epsilon).$$

By choosing $\epsilon = \Theta\left(\left(\frac{\ln T}{T(L^*)^2}\right)^{\frac{1}{K+1}}\right)$ we obtain:

$$\text{Reg}(T) \leq O(cT^{\frac{K}{K+1}} (\ln T)^{\frac{1}{K+1}}).$$

where $c = \Theta\left(K(L^*)^{\frac{K-1}{K+1}}\right)$. □

C.2 Missing Discussions and Proofs of Theorem 4.2

C.2.1 Regret Bound Comparison with [11]. In the work [11], the authors study the setting when pulling the meta arm, each base arm in (or possibly other base arm) this meta arm will be triggered and played as a result. Back to our setting, this is saying that when pulling a meta arm $\mathbf{p} = (p_1, \dots, p_K)$, each base arm k will be triggered with its corresponding probability (discretized arm) p_k . The authors in [11] discuss a general setting which allows complex reward structure where only requires two mild conditions. In particular, one of the condition they need for expected reward of playing a meta arm is the bounded smoothness (cf., Definition 1 in [11]). In the Theorem 2 of [11], the authors give results when the function used to characterize bounded smoothness is $f(x) = \gamma \cdot x^\omega$ for some $\gamma > 0$ and $\omega \in (0, 1]$. In more detail, they achieve a regret

bound $\mathcal{O}\left(\frac{2\gamma}{2-\omega} \left(\frac{12|\mathcal{M}|\ln T}{p^*}\right)^{\omega/2} \cdot T^{1-\omega/2} + |\mathcal{M}| \cdot \Delta_{\max}\right)^{12}$ where $p^* \in (0, 1)$ is the minimum triggering probability across all base arms and Δ_{\max} is the largest badness of the suboptimal meta arm in discretized space. Adapt to our setting, by inspection, we have $\gamma = L^*$, $\omega = 1$, $p^* = \epsilon$, $|\mathcal{M}| = \Theta(K/\epsilon)$, and $\Delta_{\max} = \Theta(KL^*)$. Substituting these values to the above bound, ignoring constant factors and combining with the discretization error, we have

$$\mathcal{O}\left(\left(\frac{K \ln T}{\epsilon^2}\right)^{1/2} \cdot T^{1/2} + K^2/\epsilon\right) + \mathcal{O}(TK\epsilon).$$

Picking $\epsilon = (\ln T / (KT))^{1/4}$ will give us result.

C.2.2 Proof of Lemma 4.3.

PROOF. We prove this lemma by establishing two “events” which occur with high probability: (1) the UCB index of each meta arm will concentrate on the true mean utility of \mathbf{p} ; (2) the empirical mean utility of each meta arm \mathbf{p} will also concentrate on the true mean utility of \mathbf{p} . We then show that the probability of either one of the events not holding is at most $4/t^2$. By a union bound we prove above desired lemma.

$$\begin{aligned} \text{UCB}_t(\mathbf{p}) &= \sum_{k=1}^K p_k \bar{r}_t(p_k) + \sqrt{\ln t \frac{K}{n_t(p_{\min}(\mathbf{p}))}} \\ &\stackrel{(a)}{\leq} \sum_{k=1}^K p_k \bar{r}_t(p_k) + \Delta_{\mathbf{p}}/2 < \left(\sum_{k=1}^K p_k r_k(p_k) + \Delta_{\mathbf{p}}/2 \right) + \Delta_{\mathbf{p}}/2 && \text{By Event 1} \\ &= \sum_{k=1}^K p_{k,\epsilon}^* r_k(p_{k,\epsilon}^*) < \sum_{k=1}^K p_{k,\epsilon}^* \bar{r}_t(p_{k,\epsilon}^*) + \sqrt{\ln t \frac{K}{n_t(p_{\min}(\mathbf{p}_{\epsilon}^*))}} && \text{By Event 2} \\ &= \text{UCB}_t(\mathbf{p}_{\epsilon}^*), \end{aligned}$$

where $\mathbf{p}_{\epsilon}^* = (p_{1,\epsilon}^*, \dots, p_{K,\epsilon}^*)$. The first inequality (a) comes from that $n_t(p_{\min}(\mathbf{p})) \geq \frac{4K \ln t}{\Delta_{\mathbf{p}}^2}$ and the probability of third inequality or fifth inequality not holding is at most $4/t^2$. \square

C.2.3 Proof of Lemma 4.4.

PROOF. To simplify notations, for each discretized arm p_k , we define the notion of *super set* $\mathcal{S}(p_k) = \{\mathbf{p} : p_k \in \mathbf{p}\}$ which contains all the meta arms that include this discretized arm. For

¹²For simplicity, the bound we present here omits a non-significant term.

suboptimal meta arm $\mathbf{p} \neq \mathbf{p}_\epsilon^*$ and its $p_{\min}(\mathbf{p})$, we have

$$\begin{aligned}
& \mathbb{E}[n_T(p_{\min}(\mathbf{p}))] \\
& \stackrel{(a)}{=} 1 + \mathbb{E} \left[\sum_{t=\lceil K/\epsilon \rceil+1}^T \mathbb{1}(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p}))) \right] \\
& = 1 + \mathbb{E} \left[\sum_{t=\lceil K/\epsilon \rceil+1}^T \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) < \frac{4K \ln t}{\Delta_p^2} \right) \right] \\
& \quad + \mathbb{E} \left[\sum_{t=\lceil K/\epsilon \rceil+1}^T \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) \geq \frac{4K \ln t}{\Delta_p^2} \right) \right] \\
& \stackrel{(b)}{\leq} \frac{4K \ln T}{\Delta_p^2} + \mathbb{E} \left[\sum_{t=\lceil K/\epsilon \rceil+1}^T \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) \geq \frac{4K \ln t}{\Delta_p^2} \right) \right] \\
& = \frac{4K \ln T}{\Delta_p^2} + \sum_{t=\lceil K/\epsilon \rceil+1}^T \mathbb{P} \left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) \geq \frac{4K \ln t}{\Delta_p^2} \right) \\
& = \frac{4K \ln T}{\Delta_p^2} + \sum_{t=\lceil K/\epsilon \rceil+1}^T \mathbb{P} \left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})) \middle| n_t(p_{\min}(\mathbf{p})) \geq \frac{4K \ln t}{\Delta_p^2} \right) \mathbb{P} \left(n_t(p_{\min}(\mathbf{p})) \geq \frac{4K \ln t}{\Delta_p^2} \right) \\
& \stackrel{(c)}{\leq} \frac{4K \ln T}{\Delta_p^2} + \frac{2\pi^2}{3}.
\end{aligned}$$

We add 1 in the first equality to account for 1 (step (a)) initial pull of every discretized arm by the algorithm (the initialization phase). In step (b), suppose for contradiction that the indicator $\mathbb{1}(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) < S)$ takes value of 1 at more than $S - 1$ time steps, where $S = \frac{4K \ln T}{\Delta_p^2}$. Let τ be the time step at which this indicator is 1 for the $(S - 1)$ -th time. Then the number of pulls of all meta arms in $\mathcal{S}(p_{\min}(\mathbf{p}))$ is at least L times until time τ (including the initial pull), and for all $t \geq \tau$, $n_t(p_{\min}(\mathbf{p})) \geq S$ which implies $n_t(p_{\min}(\mathbf{p})) \geq \frac{4K \ln t}{\Delta_p^2}$. Thus, the indicator cannot be 1 for any $t \geq \tau$, contradicting the assumption that the indicator takes value of 1 more than L times. This bounds $1 + \mathbb{E} \left[\sum_{t=\lceil K/\epsilon \rceil+1}^T \mathbb{1}(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) < S) \right]$ by S . In step (c), we apply the lemma 4.3 to bound the first conditional probability term and use the fact that the probabilities cannot exceed 1 to bound the second probability term. \square

C.2.4 Missing Proofs of Lemma 4.5.

PROOF. We now first prove the Claim 1 as in (10), i.e., for any $t > \lceil K/\epsilon \rceil$, we have following upper bound over counters of sufficiently selected discretized arms. To see this, by definition of

$c_T^{l,\text{sup}}(p_k)$, it reduces to show that for any $T \geq t > \lceil K/\epsilon \rceil$,

$$\begin{aligned} & \mathbb{E} \left[\sum_{p_k} \sum_{l \in [I(p_k)]} \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k) > C_T(\Delta_l^{p_k}) \right) \right] \\ &= \sum_{p_k} \sum_{l \in [I(p_k)]} \mathbb{P} \left(\mathbf{p}(t) = \mathbf{p}_l, p_k = p_{\min}(\mathbf{p}_l); \forall p \in \mathbf{p}_l, c_{t-1}(p) > C_T(\Delta_l^{p_k}) \right) \\ &\stackrel{(a)}{\leq} \lceil 4K/\epsilon \rceil \cdot t^{-2}, \end{aligned}$$

where the last step (a) is due to Lemma 4.3, thus (10) follows from a simple series bound.

We now proceed to analyze the discretized arms that are not sufficiently included in the meta arm chosen by the algorithm and prove the Claim 2 as in (11). For any under-selected discretized arm p_k , its counter $c(p_k)$ will increase from 1 to $C_T(\Delta_{\min}^{p_k})$. To simplify the notation, we set $C_T(\Delta_0^{p_k}) = 0$. Suppose that at round t , $c(p_k)$ is incremented, and $c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k}))$ for some $j \in [I(p_k)]$. Notice that we are only interested in the case that p_k is under-selected. In particular, if this is indeed the case, $\mathbf{p}(t) = \mathbf{p}_l$ for some $l \geq j$. (Otherwise, $\mathbf{p}(t)$ is sufficiently selected based on the counter value $c_{t-1}(p_k)$.) Thus, we will suffer a regret of $\Delta_l^{p_k} \leq \Delta_j^{p_k}$ (step (a)). As a result, for counter $c_t(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k}))$, we will suffer a total regret for those playing suboptimal meta arms that include under-selected discretized arms at most $(C_T(\Delta_j^{p_k}) - C_T(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k}$ in rounds that $c_t(p_k)$ is incremented (step (b)). In what follows we establish the above analysis rigorously.

$$\begin{aligned} & \sum_{l \in [I(p_k)]} c_T^{l,\text{und}}(p_k) \Delta_l^{p_k} \\ &= \sum_{t=\lceil K/\epsilon \rceil+1}^T \sum_{l \in [I(p_k)]} \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \leq C_T(\Delta_l^{p_k}) \right) \cdot \Delta_l^{p_k} \\ &= \sum_{t=\lceil K/\epsilon \rceil+1}^T \sum_{l \in [I(p_k)]} \sum_{j=1}^l \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})) \right) \cdot \Delta_l^{p_k} \\ &\stackrel{(a)}{\leq} \sum_{t=\lceil K/\epsilon \rceil+1}^T \sum_{l \in [I(p_k)]} \sum_{j=1}^l \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})) \right) \cdot \Delta_j^{p_k} \\ &\leq \sum_{t=\lceil K/\epsilon \rceil+1}^T \sum_{l, j \in [I(p_k)]} \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})) \right) \cdot \Delta_j^{p_k} \\ &= \sum_{t=\lceil K/\epsilon \rceil+1}^T \sum_{j \in [I(p_k)]} \mathbb{1} \left(\mathbf{p}(t) \in \mathcal{S}(p_k), c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})) \right) \cdot \Delta_j^{p_k} \\ &\stackrel{(b)}{\leq} \sum_{j \in [I(p_k)]} (C_T(\Delta_j^{p_k}) - C_T(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k}. \end{aligned}$$

Now, we can compute the regret incurred by selecting the meta arm which includes under-selected discretized arms:

$$\begin{aligned}
\sum_{p_k} \sum_{l \in [I(p_k)]} c_T^{l, \text{und}}(p_k) \Delta_l^{p_k} &\leq \sum_{p_k} \sum_{j \in [I(p_k)]} (C_T(\Delta_j^{p_k}) - C_T(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k} \\
&= \sum_{p_k} \left(C_T(\Delta_{\min}^{p_k}) \Delta_{\min}^{p_k} + \sum_{j \in [I(p_k)-1]} C_T(\Delta_j^{p_k}) \cdot (\Delta_j^{p_k} - \Delta_{j+1}^{p_k}) \right) \\
&\leq \sum_{p_k} \left(C_T(\Delta_{\min}^{p_k}) \Delta_{\min}^{p_k} + \int_{\Delta_{\min}^{p_k}}^{\Delta_{\max}^{p_k}} C_t(x) dx \right) \\
&= \sum_{p_k} \left(\frac{4K \ln T}{\Delta_{\min}^{p_k}} + 4K \ln T \left(\frac{1}{\Delta_{\min}^{p_k}} - \frac{1}{\Delta_{\max}^{p_k}} \right) \right). \tag{19}
\end{aligned}$$

Equipped with the above set of results, the bound of regret (9) follows by combing the above bounds in (10) and (11). \square

C.2.5 Proof of Lemma 4.6.

PROOF. Following the proof of Lemma 4.5, we only need to consider the meta arms that are played when they are under-sampled. We particularly need to deal with the situation when $\Delta_{\min}^{p_k}$ is too small. We measure the threshold for $\Delta_{\min}^{p_k}$ based on $c_T(p_k)$, i.e., the counter of discretized arm p_k at time horizon T . Let $\{T(p_k), \forall p_k\}$ be a set of possible counter values at time horizon T . Our analysis will then be conditioned on the event that $\mathcal{E}(p_k) = \{c_T(p_k) = T(p_k)\}$. By definition,

$$\begin{aligned}
&\mathbb{E} \left[\sum_{l \in [I(p_k)]} c_T^{l, \text{und}}(p_k) \cdot \Delta_l^{p_k} \mid \mathcal{E}(p_k) \right] \\
&= \sum_{t=\lceil K/\epsilon \rceil + 1}^T \sum_{l \in [I(p_k)]} \mathbb{1} \left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \leq C_T(\Delta_l^{p_k}) \mid \mathcal{E}(p_k) \right) \cdot \Delta_l^{p_k}. \tag{20}
\end{aligned}$$

We define $\Delta^*(T(p_k)) := \left(\frac{4K^2 \ln T}{\epsilon T(p_k)} \right)^{1/2}$. To achieve *instance-independent* regret bound, we consider following two cases:

Case 1: $\Delta_{\min}^{p_k} > \Delta^*(T(p_k))$, we thus have

$$\mathbb{E} \left[\sum_{l \in [I(p_k)]} c_T^{l, \text{und}}(p_k) \cdot \Delta_l^{p_k} \mid \mathcal{E}(p_k) \right] \leq \sqrt{4 \ln T \cdot \epsilon T(p_k)}. \tag{21}$$

Case 2: $\Delta_{\min}^{p_k} < \Delta^*(T(p_k))$. Let $l^* := \min\{l \in [I(p_k)] : \Delta_l^{p_k} > \Delta^*(T(p_k))\}$. Observe that we have $\Delta_{l^*}^{p_k} \leq \Delta^*(T(p_k))$ and the counter $c(p_k)$ never go beyond $T(p_k)$, we thus have

$$\begin{aligned}
(20) &\leq (C_T(\Delta^*(T(p_k))) - C_T(\Delta_{l^*-1}^{p_k})) \cdot \Delta^*(T(p_k)) + \sum_{j \in [l^*-1]} (C_T(\Delta_j^{p_k}) - C_T(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k} \\
&\leq C_T(\Delta^*(T(p_k))) \cdot \Delta^*(T(p_k)) + \int_{\Delta^*(T(p_k))}^{\Delta_{\max}^{p_k}} C_T(x) dx \leq 2 \cdot \sqrt{4 \ln T \cdot \epsilon T(p_k)}. \tag{22}
\end{aligned}$$

Thus, combining (21) and (22), we have

$$\mathbb{E} \left[\sum_{p_k: \Delta_{\min}^{p_k} > 0} \sum_{l \in [L(p_k)]} c_T^{l, \text{und}}(p_k) \cdot \Delta_l^{p_k} \mid \mathcal{E}(p_k) \right] \leq \sum_{p_k: \Delta_{\min}^{p_k} > 0} O(\sqrt{\ln T \cdot \epsilon T(p_k)}) \stackrel{(a)}{\leq} O(K\sqrt{KT \ln T / \epsilon}),$$

where (a) is by Jensen's inequality and $\sum_{p_k} T(p_k) \leq T$. Put all pieces together, we have the instance-independent regret bound as stated in the lemma. Observe that the final inequality does not depend on the event $\mathcal{E}(p_k)$, we thus can drop this conditional expectation. \square

D PROOF OF THEOREM 5.1 FOR HISTORY-DEPENDENT BANDITS

D.1 Proof of Lemma 5.3

PROOF. The proof of this lemma is mainly built on analyzing the convergence of $\mathbf{p}^{(y)}$ via pulling the base arms with the same probability consistently. For the ease of presentation, let us suppose $t = mL$ and let $t_m^e := \frac{t}{L}(L - s_a) = m(L - s_a)$ be the total number of estimation rounds in the first m phases. Thus, at the end of the approaching stage, we have

$$\widehat{p}_k^{(y)}(t + s_a) = \frac{p_k(t + s_a)\gamma^0 + \dots + p_k(t + 1)\gamma^{s_a-1} + (1 + \gamma + \dots + \gamma^{t-1})\gamma^{s_a}\widehat{p}_k^{(y)}(t)}{1 + \gamma + \dots + \gamma^{t+s_a-1}},$$

where $\widehat{p}_k^{(y)}(t) = \frac{p_k(t)\gamma^0 + \dots + p_k(1)\gamma^{t-1}}{1 + \gamma + \dots + \gamma^{t-1}}$. Recall that during the approaching stage, we consistently pull arm k with the same probability p_k . Thus, the approximation error of $\widehat{p}_k^{(y)}(t + s_a)$ w.r.t. p_k can be computed as:

$$|\widehat{p}_k^{(y)}(t + s_a) - p_k| = \left| \frac{p_k(1 - \gamma^{s_a}) + \widehat{p}_k^{(y)}(t)\gamma^{s_a}(1 - \gamma^t)}{1 - \gamma^{t+s_a}} - p_k \right| \leq \frac{\gamma^{s_a}(1 - \gamma^t)}{1 - \gamma^{t+s_a}} < \gamma^{s_a}.$$

Recall that $U(\mathbf{p}) = \sum_{p_k \in \mathbf{p}} p_k r_k(p_k)$. In the estimation stage, we approximate all the realized utility as the utility generated by the meta arm \mathbf{p} . However, note that we actually cannot compute the empirical value of $\bar{U}(\mathbf{p})$, instead, we use $\bar{U}_m^e(\mathbf{p}(t + s_a))$ of each phase as an approximation of $\bar{U}(\mathbf{p})$, i.e., we approximate all $\mathbf{p}^{(y)}(t + s)$, $\forall s \in (s_a, L]$ as $\mathbf{p}(t + s_a)$ and use $\mathbf{p}(t + s_a)$ as the approximation of \mathbf{p} . Recall that for any $s \in (s_a, L]$, we have:

$$|\widehat{p}_k^{(y)}(t + s) - p_k| = \left| \frac{\gamma^s(1 - \gamma^t)(\widehat{p}_k^{(y)}(t) - p_k)}{1 - \gamma^{t+s}} \right| \leq \frac{\gamma^s(1 - \gamma^t)}{1 - \gamma^{t+s}} < \frac{\gamma^{s_a}(1 - \gamma^t)}{1 - \gamma^{t+s_a}} < \gamma^{s_a}.$$

Thus, the approximation error on the empirical estimation can be computed as follows:

$$\begin{aligned} \mathbb{E} \left[\left| \bar{U}_m^e(\mathbf{p}(t + s_a)) - \bar{U}(\mathbf{p}) \right| \right] &= \mathbb{E} \left[\left| \sum_{p_k^{(y)} \in \mathbf{p}(t+s_a)} p_k^{(y)} \bar{r}_{t+s_a}^e(p_k^{(y)}) - \sum_{p_k \in \mathbf{p}} p_k \bar{r}_{t+s_a}^e(p_k) \right| \right] \\ &= \left| \sum p_k^{(y)} \mathbb{E} \left[\bar{r}_{t+s_a}^e(p_k^{(y)}) \right] - \sum p_k \mathbb{E} \left[\bar{r}_{t+s_a}^e(p_k) \right] \right| \\ &= \left| \sum p_k^{(y)} r_k(p_k^{(y)}) - \sum p_k r_k(p_k) \right| \\ &= \left| \sum \left(p_k^{(y)} \left(r_k(p_k^{(y)}) - r_k(p_k) \right) + r_k(p_k) (p_k^{(y)} - p_k) \right) \right| \\ &\leq \sum \left| \gamma^{s_a} L_k p_k^{(y)} + r_k(p_k) \gamma^{s_a} \right| \leq K \gamma^{s_a} (L^* + 1). \end{aligned}$$

□

D.1.1 Proof of Lemma 5.4.

PROOF. We first decompose $|U(\mathbf{p}) - \bar{U}_m^e(\mathbf{p}_\epsilon^{(\gamma)})|$ as $|U(\mathbf{p}) - \bar{U}(\mathbf{p})| + |\bar{U}(\mathbf{p}) - \bar{U}_m^e(\mathbf{p})|$ and then apply union bound.

$$\begin{aligned}
& \mathbb{P}\left(|U(\mathbf{p}) - \bar{U}_m^e(\mathbf{p}(t + s_a))| \geq \delta\right) \\
& \leq \mathbb{P}\left(|U(\mathbf{p}) - \bar{U}(\mathbf{p})| + |\bar{U}(\mathbf{p}) - \bar{U}_m^e(\mathbf{p}(t + s_a))| \geq \delta\right) \quad \text{By triangle inequality} \\
& = \mathbb{P}\left(|U(\mathbf{p}) - \bar{U}(\mathbf{p})| + |\bar{U}_m^e(\mathbf{p}(t + s_a)) - \mathbb{E}[\bar{U}_m^e(\mathbf{p}(t + s_a))]| - \right. \\
& \quad \left. (\bar{U}(\mathbf{p}) - \mathbb{E}[\bar{U}(\mathbf{p})]) + \mathbb{E}[\bar{U}(\mathbf{p})] - \mathbb{E}[\bar{U}_m^e(\mathbf{p}(t + s_a))]| \geq \delta\right) \\
& \leq \mathbb{P}\left(2|U(\mathbf{p}) - \bar{U}(\mathbf{p})| + |\bar{U}_m^e(\mathbf{p}(t + s_a)) - \mathbb{E}[\bar{U}_m^e(\mathbf{p}(t + s_a))]| \geq \delta - \text{err}\right) \\
& \stackrel{(a)}{\leq} 3\mathbb{P}\left(|U(\mathbf{p}) - \bar{U}(\mathbf{p})| \geq \frac{\delta - \text{err}}{3}\right) \leq 6 \exp\left(-\frac{n_m^e(p_{\min}(\mathbf{p}))(\delta - \text{err})^2}{9K}\right),
\end{aligned}$$

where in step (a), we use the Hoeffding's Inequality on Weighted Sums and Lemma 5.3. □

D.1.2 Proof of Lemma 5.5.

PROOF. To prove the above lemma, we construct two high-probability events. Event 1 corresponds to that the UCB index of each meta arm concentrates on the true mean utility of \mathbf{p} ; Event 2 corresponds to that the empirical mean utility of each approximated meta arm $\mathbf{p}^{(\gamma)}$ concentrates on the true mean utility of \mathbf{p} . The probability of Event 1 or Event 2 not holding is at most $4/t^2$. By the definition of the constructed UCB, we'll have

$$\begin{aligned}
\text{UCB}_m(\mathbf{p}) &= \bar{U}_m^e(\mathbf{p}(t + s_a)) + \text{err} + 3\sqrt{\frac{K \ln(L\rho m)}{n_m^e(p_{\min}(\mathbf{p}))}} \stackrel{(a)}{\leq} \bar{U}_m^e(\mathbf{p}(t + s_a)) + \Delta_{\mathbf{p}}/2 \\
&\stackrel{(b)}{<} (U(\mathbf{p}) + \Delta_{\mathbf{p}}/2) + \Delta_{\mathbf{p}}/2 \quad \text{By Event 1} \\
&= U(\mathbf{p}_\epsilon^*) \stackrel{(c)}{<} \text{UCB}_m(\mathbf{p}_\epsilon^*), \quad \text{By Event 2}
\end{aligned}$$

where the first inequality (a) is due to $n_m^e(p_{\min}(\mathbf{p})) \geq \frac{9K \ln(L\rho m)}{(\Delta_{\mathbf{p}}/2 - \text{err})^2}$, and the probability of step (b) or (c) not holding is at most $12/(L\rho m)^2$. □

The above lemma implies that we will stop deploying suboptimal meta arm \mathbf{p} and further prevent it from incurring regret as we gather more information about it such that $\text{UCB}_m(\mathbf{p}) < \text{UCB}_m(\mathbf{p}_\epsilon^*)$.

D.1.3 Proof of Lemma 5.6.

PROOF. For notation simplicity, suppose $t = mL$. For each suboptimal arm $\mathbf{p} \neq \mathbf{p}_\epsilon^*$, and suppose there exists $p_{\min}(\mathbf{p}) \notin \mathbf{p}_\epsilon^*$ such that $p_{\min}(\mathbf{p}) = \arg \min_{p_k \in \mathbf{p}} n_t^\epsilon(p_k)$, then

$$\begin{aligned}
& \mathbb{E}[n_t^\epsilon(p_{\min}(\mathbf{p}))] \\
&= (L - s_a) \mathbb{E} \left[\sum_{i=1}^m \mathbb{1}(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p}))) \right] \\
&= (L - s_a) \mathbb{E} \left[\sum_{i=1}^m \mathbb{1} \left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^\epsilon(p_{\min}(\mathbf{p})) < \frac{9K \ln(i(L - s_a))}{(\Delta_p/2 - \text{err})^2} \right) \right] + \\
&\quad (L - s_a) \mathbb{E} \left[\sum_{i=1}^m \mathbb{1} \left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^\epsilon(p_{\min}(\mathbf{p})) \geq \frac{9K \ln(i(L - s_a))}{(\Delta_p/2 - \text{err})^2} \right) \right] \\
&\stackrel{(a)}{\leq} \frac{9K \ln(t_m^\epsilon)}{(\Delta_p/2 - \text{err})^2} + (L - s_a) \mathbb{E} \left[\sum_{i=1}^m \mathbb{1} \left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^\epsilon(p_{\min}(\mathbf{p})) \geq \frac{9K \ln(i(L - s_a))}{(\Delta_p/2 - \text{err})^2} \right) \right] \\
&= \frac{9K \ln(t_m^\epsilon)}{(\Delta_p/2 - \text{err})^2} + (L - s_a) \sum_{i=1}^m \mathbb{P} \left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})) \middle| n_i^\epsilon(p_{\min}(\mathbf{p})) \geq \frac{9K \ln(i(L - s_a))}{(\Delta_p/2 - \text{err})^2} \right) \\
&\quad \mathbb{P} \left(n_i^\epsilon(p_{\min}(\mathbf{p})) \geq \frac{9K \ln(i(L - s_a))}{(\Delta_p/2 - \text{err})^2} \right) \\
&\leq \frac{9K \ln(t_m^\epsilon)}{(\Delta_p/2 - \text{err})^2} + (L - s_a) \sum_{i=1}^m \frac{12}{(i(L - s_a))^2} \leq \frac{9K \ln(t_m^\epsilon)}{(\Delta_p/2 - \text{err})^2} + \frac{2\pi^2}{L - s_a}.
\end{aligned}$$

In step (a), suppose for contradiction that the indicator $\mathbb{1}(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^\epsilon(p_{\min}(\mathbf{p})) < S)$ takes value of 1 at more than $S - 1$ time steps, where $S = \frac{9K \ln(i(S - s_a))}{(\Delta_p/2 - \text{err})^2}$. Let τ be the phase at which this indicator is 1 for the $(S - 1)$ -th phase. Then the number of pulls of all meta arms in $\mathcal{S}(p_{\min}(\mathbf{p}))$ is at least L times until time τ (including the initial pull), and for all $i > \tau$, $n_i^\epsilon(p_{\min}(\mathbf{p})) \geq S$ which implies $n_i^\epsilon(p_{\min}(\mathbf{p})) \geq \frac{9K \ln(i(S - s_a))}{(\Delta_p/2 - \text{err})^2}$. Thus, the indicator cannot be 1 for any $i \geq \tau$, contradicting the assumption that the indicator takes value of 1 more than S times. This bounds $1 + \mathbb{E} \left[\sum_{i=1}^m \mathbb{1}(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^\epsilon(p_{\min}(\mathbf{p})) < S) \right]$ by S . \square

D.1.4 Proof of Lemma 5.7.

PROOF. For notation simplicity, we include all initialization rounds to phase 0 and suppose the time horizon $T = ML$. Note that by definitions, we can compute the regret $\text{Reg}_\epsilon(T)$ as follows:

$$\text{Reg}_\epsilon(T) = \sum_{\mathbf{p} \in \mathcal{P}_\epsilon} \mathbb{E}[N_T(\mathbf{p})] \Delta_p \leq \sum_{p_k} \sum_{\mathbf{p}_l \in \mathcal{S}(p_k)} \mathbb{E}[N_T(\mathbf{p}_l)] \Delta_l^{p_k}. \quad (23)$$

where $N_t(\mathbf{p}) = K + L \sum_{m=1}^M \mathbb{1}(\mathbf{p}(m) = \mathbf{p})$, where K here accounts for the initialization. Follow the same analysis in action-dependent bandits, we can also define a counter $c^\epsilon(p_k)$ for each discretized arm p_k and the value of $c^\epsilon(p_k)$ at phase m is denoted by $c_m^\epsilon(p_k)$. But different from the action-dependent bandit setting, we update the counter $c^\epsilon(p_k)$ only when we start a new phase. In particular, for a phase $m \geq 1$, let $\mathbf{p}(m)$ be the meta arm selected in the phase m by the algorithm. Let $p_k = \arg \min_{p_k \in \mathbf{p}(m)} c_m^\epsilon(p_k)$. We increment $c_m^\epsilon(p_k)$ by one, i.e., $c_m^\epsilon(p_k) = c_{m-1}^\epsilon(p_k) + 1$. In other words, we find the discretized arm p_k with the smallest counter in $\mathbf{p}(m)$ and increment its counter.

If such p_k is not unique, we pick an arbitrary discretized arm with the smallest counter. Note that the initialization gives $\sum_{p_k} c_0^e(p_k) = \lceil K/\epsilon \rceil$. It is easy to see that for any $p_k = p_{\min}(\mathbf{p})$, we have $n_m(p_k) = L\rho \cdot c_m^e(p_k)$.

Like in action-dependent bandits, we also define $C_M^e(\Delta) := \frac{9K \ln(ML\rho)}{L\rho(\Delta/2 - \text{err})^2}$, the number of selection that is considered sufficient for a meta arm with reward Δ away from the optimal strategy \mathbf{p}_ϵ^* with respect to phase horizon M . With the above notations, we define following two situations for the counter of each discretized arm:

$$c_M^{e,l,\text{supf}}(p_k) := \sum_{m=1}^M \mathbb{1} \left(\mathbf{p}(m) = \mathbf{p}_l, c_m^e(p_k) > c_{m-1}^e(p_k) > C_M^e(\Delta_l^{p_k}) \right) \quad (24)$$

$$c_M^{e,l,\text{und}}(p_k) := \sum_{m=1}^M \mathbb{1} \left(\mathbf{p}(m) = \mathbf{p}_l, c_m^e(p_k) > c_{m-1}^e(p_k), c_{m-1}^e(p_k) \leq C_M^e(\Delta_l^{p_k}) \right). \quad (25)$$

Clearly, we have $c_M^e(p_k) = 1 + \sum_{l \in I(p_k)} (c_M^{e,l,\text{supf}}(p_k) + c_M^{e,l,\text{und}}(p_k))$. With these notations, we can write (23) as follows:

$$\text{Reg}_\epsilon(T) \leq \mathbb{E} \left[\sum_{p_k} \left(\Delta_{\max}^{p_k} + L \cdot \sum_{l \in [I(p_k)]} (c_M^{e,l,\text{supf}}(p_k) + c_M^{e,l,\text{und}}(p_k)) \cdot \Delta_l^{p_k} \right) \right]. \quad (26)$$

We now first show that for any $m \geq 1$, we have following upper bound over counters of sufficiently selected discretized arms:

$$\mathbb{E} \left[L \cdot \sum_{p_k} \sum_{l \in [I(p_k)]} c_M^{l,\text{supf}}(p_k) \right] \leq O \left(\frac{K}{L\epsilon\rho^2} \right). \quad (27)$$

To see this, by definition of $c_M^{e,l,\text{supf}}(p_k)$, it reduces to show that for any $M \geq m > 1$,

$$\begin{aligned} & \mathbb{E} \left[L \cdot \sum_{p_k} \sum_{l \in [I(p_k)]} \mathbb{1} \left(\mathbf{p}(m) = \mathbf{p}_l, c_m^e(p_k) > c_{m-1}^e(p_k) > C_M^e(\Delta_l^{p_k}) \right) \right] \\ &= L \cdot \sum_{p_k} \sum_{l \in [I(p_k)]} \mathbb{P} \left(\mathbf{p}(m) = \mathbf{p}_l, p_k = p_{\min}(\mathbf{p}_l); \forall p \in \mathbf{p}_l, L\rho \cdot c_{m-1}^e(p) > \frac{9K \ln(ML\rho)}{(\Delta_l^{p_k}/2 - \text{err})^2} \right) \\ &\stackrel{(a)}{\leq} \lceil 12LK/\epsilon \rceil \cdot (ML\rho)^{-2}, \end{aligned}$$

where the last step (a) is due to Lemma 5.5, thus (27) follows from a simple series bound.

We now proceed to analyze the discretized arms that are not sufficiently included in the meta arm chosen by the algorithm. For any under-selected discretized arm p_k , its counter $c^e(p_k)$ will increase from 1 to $C_M^e(\Delta_{\min}^{p_k})$. To simplify the notation, we set $C_M^e(\Delta_0^{p_k}) = 0$. Suppose that at phase $m \geq 1$, $c^e(p_k)$ is incremented, and $c_{m-1}^e(p_k) \in (C_M^e(\Delta_{j-1}^{p_k}), C_M^e(\Delta_j^{p_k})]$ for some $j \in [I(p_k)]$. Notice that we are only interested in the case that p_k is under-selected. In particular, if this is indeed the case, $\mathbf{p}(m) = \mathbf{p}_l$ for some $l \geq j$. (Otherwise, $\mathbf{p}(m)$ is sufficiently selected based on the counter value $c_{m-1}^e(p_k)$.) Thus, we will suffer a regret of $\Delta_l^{p_k} \leq \Delta_j^{p_k}$ (step (a)). As a result, for counter $c_m^e(p_k) \in (C_M^e(\Delta_{j-1}^{p_k}), C_M^e(\Delta_j^{p_k})/L]$, we will suffer a total regret for those playing suboptimal meta arms that include under-selected discretized arms at most $(C_M^e(\Delta_j^{p_k}) - C_M^e(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k}$ in rounds

that $c_m^e(p_k)$ is incremented (step (b)). In what follows we establish the above analysis rigorously.

$$\begin{aligned}
& \sum_{l \in [I(p_k)]} c_M^{e,l,\text{und}}(p_k) \Delta_l^{p_k} \\
&= \sum_{m=1}^M \sum_{l \in [I(p_k)]} \mathbb{1} \left(\mathbf{p}(m) = \mathbf{p}_l, c_m^e(p_k) > c_{m-1}^e(p_k), c_{m-1}^e(p_k) \leq C_M^e(\Delta_l^{p_k}) \right) \cdot \Delta_l^{p_k} \\
&= \sum_{m=1}^M \sum_{l \in [I(p_k)]} \sum_{j=1}^l \mathbb{1} \left(\mathbf{p}(m) = \mathbf{p}_l, c_m^e(p_k) > c_{m-1}^e(p_k), c_{m-1}^e(p_k) \in (C_M^e(\Delta_{j-1}^{p_k}), C_M^e(\Delta_j^{p_k})) \right) \cdot \Delta_l^{p_k} \\
&\stackrel{(a)}{\leq} \sum_{m=1}^M \sum_{l \in [I(p_k)]} \sum_{j=1}^l \mathbb{1} \left(\mathbf{p}(m) = \mathbf{p}_l, c_m^e(p_k) > c_{m-1}^e(p_k), c_{m-1}^e(p_k) \in (C_M^e(\Delta_{j-1}^{p_k}), C_M^e(\Delta_j^{p_k})) \right) \cdot \Delta_j^{p_k} \\
&\leq \sum_{m=1}^M \sum_{l,j \in [I(p_k)]} \mathbb{1} \left(\mathbf{p}(m) = \mathbf{p}_l, c_m^e(p_k) > c_{m-1}^e(p_k), c_{m-1}^e(p_k) \in (C_M^e(\Delta_{j-1}^{p_k}), C_M^e(\Delta_j^{p_k})) \right) \cdot \Delta_j^{p_k} \\
&= \sum_{m=1}^M \sum_{j \in [I(p_k)]} \mathbb{1} \left(\mathbf{p}(m) \in \mathcal{S}(p_k), c_m^e(p_k) > c_{m-1}^e(p_k), c_{m-1}^e(p_k) \in (C_M^e(\Delta_{j-1}^{p_k}), C_M^e(\Delta_j^{p_k})) \right) \cdot \Delta_j^{p_k} \\
&\stackrel{(b)}{\leq} \sum_{j \in [I(p_k)]} (C_M^e(\Delta_j^{p_k}) - C_M^e(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k}.
\end{aligned}$$

Now, we can compute the regret incurred by selecting the meta arm which includes under-selected discretized arms:

$$\begin{aligned}
& L \cdot \sum_{p_k} \sum_{l \in [I(p_k)]} c_M^{e,l,\text{und}}(p_k) \cdot \Delta_l^{p_k} \\
&\leq L \cdot \sum_{p_k} \sum_{j \in [I(p_k)]} (C_M^e(\Delta_j^{p_k}) - C_M^e(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k} \\
&= L \cdot \sum_{p_k} \left(C_M^e(\Delta_{\min}^{p_k}) \Delta_{\min}^{p_k} + \sum_{j \in [I(p_k)-1]} C_M^e(\Delta_j^{p_k}) \cdot (\Delta_j^{p_k} - \Delta_{j+1}^{p_k}) \right) \\
&\leq L \cdot \sum_{p_k} \left(C_M^e(\Delta_{\min}^{p_k}) \Delta_{\min}^{p_k} + \int_{\Delta_{\min}^{p_k}}^{\Delta_{\max}^{p_k}} C_M^e(x) dx \right) \\
&= \sum_{p_k} \left(\frac{9K \ln(ML\rho)}{\rho (\Delta_{\min}^{p_k}/2 - \text{err})^2} \cdot \Delta_{\min}^{p_k} + 9K \ln(ML\rho) / \rho \cdot \int_{\Delta_{\min}^{p_k}}^{\Delta_{\max}^{p_k}} \frac{1}{(x/2 - \text{err})^2} dx \right) \\
&= \sum_{p_k} \left(\frac{9\Delta_{\min}^{p_k} K \ln(ML\rho)}{\rho (\Delta_{\min}^{p_k}/2 - \text{err})^2} + \frac{9K \ln(ML\rho)}{\rho} \left(\frac{2}{\frac{\Delta_{\min}^{p_k}}{2} - \text{err}} - \frac{2}{\Delta_{\max}^{p_k}/2 - \text{err}} \right) \right) \\
&\leq \sum_{p_k} \left(\frac{9K \ln(ML\rho)}{\rho} \left(\frac{\Delta_{\min}^{p_k}}{(\Delta_{\min}^{p_k}/2 - \text{err})^2} + \frac{2}{\Delta_{\min}^{p_k}/2 - \text{err}} \right) \right).
\end{aligned}$$

Combing the bound established in (27) will complete the proof. \square

D.1.5 Proof of Lemma 5.8.

PROOF. Following the proof action-dependent bandits, we only need to consider the meta arms that are played when they are under-sampled. We particularly need to deal with the situation when $\Delta_{\min}^{p_k}$ is too small. We measure the threshold for $\Delta_{\min}^{p_k}$ based on $c_M^e(p_k)$, i.e., the counter of discretized arm p_k at phase horizon M . Let $\{M(p_k), \forall p_k\}$ be a set of possible counter values at time horizon M . Our analysis will then be conditioned on the event that $\mathcal{E}(p_k) := \{c_M^e(p_k) = M(p_k)\}$. By definition,

$$\begin{aligned} & \mathbb{E}\left[\sum_{l \in [I(p_k)]} c_M^{e, l, \text{und}}(p_k) \cdot \Delta_l^{p_k} \mid \mathcal{E}(p_k)\right] \\ &= \sum_{m=1}^M \sum_{l \in [I(p_k)]} \mathbb{1}\left(\mathbf{p}(m) = \mathbf{p}_l, c_m^e(p_k) > c_{m-1}^e(p_k), c_{m-1}^e(p_k) \leq C_M^e(\Delta_l^{p_k}) \mid \mathcal{E}(p_k)\right) \cdot \Delta_l^{p_k}. \end{aligned} \quad (28)$$

We define $\Delta^*(M(p_k)) := 2 \left(\frac{9K^2 \ln(ML\rho)}{L\rho \cdot \epsilon M(p_k)} \right)^{1/2} + 2\text{err}$. To achieve *instance-independent* regret bound, we consider following two cases:

Case 1: $\Delta_{\min}^{p_k} > \Delta^*(M(p_k))$, clearly we have $\Delta_{\min}^{p_k}/2 > \text{err}$. Thus,

$$LK/\epsilon \cdot \mathbb{E}\left[\sum_{l \in [I(p_k)]} c_M^{e, l, \text{und}}(p_k) \cdot \Delta_l^{p_k} \mid \mathcal{E}(p_k)\right] \leq 12K \cdot \sqrt{\frac{\ln(T\rho) \cdot LM(p_k)}{\rho\epsilon}} + 2K\text{err} \cdot LM(p_k) \quad (29)$$

Case 2: $\Delta_{\min}^{p_k} < \Delta^*(M(p_k))$. Let $l^* := \min\{l \in I(p_k) : \Delta_l^{p_k} > \Delta^*(M(p_k))\}$. Observe that we have $\Delta_{l^*}^{p_k} \leq \Delta^*(M(p_k))$ and the counter $c^e(p_k)$ never go beyond $M(p_k)$, we thus have

$$\begin{aligned} LK/\epsilon \cdot (28) &\leq L(C_M^e(\Delta^*(M(p_k))) - C_M^e(\Delta_{l^*-1}^{p_k})) \cdot \Delta^*(M(p_k)) + \sum_{j \in [l^*-1]} L(C_M^e(\Delta_j^{p_k}) - C_M^e(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k} \\ &\leq LC_M^e(\Delta^*(M(p_k))) \cdot \Delta^*(M(p_k)) + L \int_{\Delta^*(M(p_k))}^{\Delta_{\max}^{p_k}} C_M^e(x) dx \\ &\leq O\left(K \cdot \sqrt{\frac{\ln(T\rho) \cdot LM(p_k)}{\rho\epsilon}} + K\text{err} \cdot LM(p_k)\right). \end{aligned} \quad (30)$$

Combining (29) and (30), and with Jensen's inequality and $\sum_{p_k} M(p_k) \leq M$ will give us desired result. Put all pieces together, we have the instance-independent regret bound as stated in the lemma. The final inequality does not depend on the event $\mathcal{E}(p_k)$, we thus can drop this conditional expectation. \square

D.1.6 Wrapping-up to Prove Theorem 5.1. Combining with the discretization error, we have

$$\text{Reg}(T) \leq O\left(K \cdot \sqrt{T \ln(T\rho)/(\rho\epsilon)} + K\text{Terr} + K/(\epsilon\rho^2)\right) + O(K\epsilon T).$$

Picking

$$\epsilon = \left(\frac{\ln(T\rho)}{T\rho}\right)^{1/3}; \quad s_a = \frac{1/3 \ln\left(\frac{\ln(T\rho)}{T\rho}\right) - \ln(L^*K)}{\ln \gamma}.$$

We will obtain the results as stated in the theorem.

E LOWER BOUND OF ACTION-DEPENDENT BANDITS

In this section, we derive the lower regret bound of bandits with action-dependent feedback, showing that the upper regret bound of our Algorithm 1 is optimal in the sense that it matches this lower bound in terms of the dependency on T and K . Note that, by importance-weighting technique, we can construct an unbiased estimation of each base arm's reward. In the below discussion, we rephrase our problem as the *combinatorial Lipschitz bandit with constraint*, henceforth called CombLipBwC, which directly operates on the observations of all base arms:

DEFINITION E.1 (CombLipBwC). *Let action set \mathcal{P} available to the learner be a continuous space, consisted of K unit-range base arms, i.e., $\mathcal{P} \subset [0, 1]^K$. At each time, the learner needs to select a meta arm $\mathbf{p}(t) = \{p_1(t), \dots, p_K(t)\}$ in which each discretized arm $p_k(t) \in [0, 1]$ is selected from k -th unit range, with the constraint such that $\sum_k p_k(t) = 1$. And then the learner will observe rewards $\{\tilde{r}_t(p_k(t))\}_{k \in [K]}$ for all base arms with the mean of each $\mathbb{E}[\tilde{r}_t(p_k(t))] = r_k(p_k(t))$.*

Our main result of this section is summarized in the following theorem:

THEOREM E.2. *Let $T > 2K$ and $K \geq 4$, there exists a problem instance such that for any algorithm \mathcal{A} for our action-dependent bandits, we have $\inf_{\mathcal{A}} \text{Reg}(T) \geq \Omega(KT^{2/3})$.*

The high-level intuition for deriving the above lower bound is that we first construct a reduction from CombLipBwC to a discretized *combinatorial bandit problem with the action constraint* $\sum_k p_k(t) = 1$ - we refer to this latter problem setting as CombBwC. Then we show that the regret incurred within CombLipBwC is lower bounded by the regret incurred with CombBwC. To finish the proof, we bound the worst-case regret from below of CombBwC by taking an average over a conveniently chosen class of problem instances.

E.1 Randomized problem instances and definitions

We now construct a reduction for proving the lower bound of CombLipBwC. Specifically, we will construct a distribution \mathcal{D} over a set of problem instances (we also call each instance an adversary, since the instances are adversarially constructed) of CombLipBwC, while each problem instance will be uniquely mapped to a problem instance in CombBwC. The construction is similar to the technique used in [33].

These new instances are associated with 0 - 1 rewards. For each base arm $k \in [K]$, all the discretized arms p have mean reward $r_k(p) = 1/2$ except those near the unique best discretized arm p_k^* with $r_k(p_k^*) = 1/2 + \epsilon$. Here $\epsilon > 0$ is a parameter to be adjusted later in the analysis. Due to the requirement of Lipschitz condition, a smooth transition is needed in the neighborhood of each p_k^* . More formally, we define the following function $r_k(\cdot)$ for base arm k :

$$r_k(p) = \begin{cases} 1/2, & \forall p \in [0, 1] : |p - p_k^*| \geq \epsilon/L_k \\ 1/2 + \epsilon - L_k \cdot |p - p_k^*|, & \forall p \in [0, 1] : |p - p_k^*| < \epsilon/L_k \end{cases} \quad (31)$$

Fix $N_p \in \mathbb{N}$ and partition all base arms $[0, 1]$ into N_p disjoint intervals of length $1/N_p$. Then the above functions indicate that each interval with the length of 2ϵ will either contain a bump or be completely flat. For the sake of simplifying presentation, in the analysis below, we'll focus on the case where the Lipschitz constant is $L_k = 1, \forall k \in [K]$. Formally,

DEFINITION E.3. *We define 0-1 rewards problem instances $\mathcal{I}(\mathbf{p}^*, \epsilon)$ for CombLipBwC indexed by a random permutation $\mathbf{p}^* = \{p_k^*\}_{k \in [K]}$, which satisfies following property:*

- $\sum_k p_k^* = 1$ and each p_k^* takes the value from $\{(2j-1)\epsilon\}_{j \in [N_p]}$.
- The reward function of base arm k is defined in (31), and the optimal action of arm k is p_k^* .

In combinatorial bandits, the learner selects a subset of ground arms subject to some pre-defined constraints. Adapting to our model, we denote this discretized action space \mathcal{M} as the set of $K \times N_p$ binary matrices $\{0, 1\}^{K \times N_p}$:

$$\mathcal{M} = \{\mathbf{a} \in \{0, 1\}^{K \times N_p} : \forall k \in [K], \sum_{j=1}^{N_p} a_{k,j} = 1\},$$

where $a_{k,j} \in \{0, 1\}$ is the indicator random variable such that $a_{k,j} = 1$ means that the j -th discretized arm probability is selected for the k -th base arm. Note that this space has not included the action constraint that we're planning to impose on CombLipBwC.

We now construct the problem instances for CombBwC such that each problem instance $\mathcal{I}(\mathbf{p}^*, \epsilon)$ in CombLipBwC has a corresponding problem instance in CombBwC.

DEFINITION E.4. We define 0-1 rewards problem instances $\mathcal{J}(\mathbf{I}^*, \epsilon)$ for CombBwC indexed by $\mathbf{I}^* = \{l_k^*\}_{k \in [K]}$, such that $l_k^* = (p_k^*/\epsilon + 1)/2$. Therefore, $l_k^* \in [N_p]$ and the mean reward of $\mathcal{J}(\mathbf{I}^*, \epsilon)$ is defined as follows: for any $t \in \{1, \dots, T\}$,

$$\mathbb{E}[\tilde{r}_t(l_k(t))] = \begin{cases} 1/2, & l_k(t) \neq l_k^* \\ 1/2 + \epsilon, & l_k(t) = l_k^* \end{cases} \quad (32)$$

Observe that with one more action constraint, the feasible action space of CombBwC will be a constrained space of \mathcal{M} , which we denote by $\Pi = \{\mathbf{a} \in \{0, 1\}^{K \times N_p} : \forall k \in [K], \sum_{j=1}^{N_p} a_{k,j} = 1, \sum_{i=1}^K l_{k,j} a_{k,j} = K - 1 + N_p\}$.

We now next show that for any algorithm \mathcal{A}_I trying to solve the problem instance $\mathcal{I}(\mathbf{p}^*, \epsilon)$ in CombLipBwC, we can construct an algorithm \mathcal{A}_J that needs to solve a corresponding problem instance $\mathcal{J}(\mathbf{p}^*, \epsilon)$ in CombBwC.

The intuition of the construction routine is as follows. With the above defined KN_p intervals in hand and the deliberately designed reward structure, whenever an algorithm chooses a meta arm $\mathbf{p} = \{p_1, \dots, p_K\}$ such that each discretized arm p_k falls into an interval of this base arm k , choosing the center of this interval is best. Thus, if we restrict to discretized arms that are centers of the intervals of all base arms, we then have a family of problem instances of CombBwC, where the reward function is exactly defined in (32).

Routine A routine inbetween \mathcal{A}_I and \mathcal{A}_J

Input: A CombLipBwC instance \mathcal{I} , a CombBwC instance \mathcal{J} and an algorithm \mathcal{A}_I for solving \mathcal{I} .

for round $t = 1, \dots$ **do**

\mathcal{A}_I selects a meta arm $\mathbf{p}(t) = \{p_1(t), \dots, p_K(t)\}$;

\mathcal{A}_J selects arm $\mathbf{l}(t) = \{l_1(t), \dots, l_K(t)\}$ such that $p_k(t) \in [(2l_k(t) - 1)\epsilon - \epsilon, (2l_k(t) - 1)\epsilon + \epsilon], \forall k \in [K]$;

\mathcal{A}_J observes $\{\tilde{r}(l_k(t))\}$;

\mathcal{A}_I observes $\{\tilde{r}(p_k(t))\}$;

end for

Furthermore, with above construction routine, we have following guarantee:

LEMMA E.5. *The regret incurred by \mathcal{A}_I , which is for the problem instance $\mathcal{I}(\mathbf{p}^*, \epsilon)$, is lower bounded by the regret incurred by \mathcal{A}_J for the problem instance $\mathcal{J}(\mathbf{I}^*, \epsilon)$:*

$$\mathbb{E}[\text{Reg}_{2\epsilon}(T)|\mathcal{I}, \mathcal{A}_I] \geq \mathbb{E}[\text{Reg}_{2\epsilon}(T)|\mathcal{J}, \mathcal{A}_J]. \quad (33)$$

PROOF. As we can see, each instance $\mathcal{J}(\mathbf{I}^*, \epsilon)$ corresponds to an instance $\mathcal{I}(\mathbf{p}^*, \epsilon)$ of CombLipBwC. In particular, each k -th base arm in \mathcal{J} corresponds to the base arm k in \mathcal{I} , and more specifically, each discretized arm $j \in [N_p]$ in k -th base arm corresponds to the all possible discretized arms p such that $p \in [(2j-1) \cdot \epsilon - \epsilon, (2j-1) \cdot \epsilon + \epsilon)$. In other words, we can view \mathcal{J} as a discrete version of \mathcal{I} . In particular, we have $r_k(j|\mathcal{J}) = r_k(p)$, $\forall p \in [(2j-1)\epsilon - \epsilon, (2j-1)\epsilon + \epsilon)$, where $r_k(\cdot)$ is the reward function for base arm k in \mathcal{I} , and $r_k(\cdot|\mathcal{J})$ is the reward function for base arm k in \mathcal{J} .

Given an arbitrary algorithm \mathcal{A}_I for a problem instance \mathcal{I} of CombLipBwC, we can use it to construct an algorithm \mathcal{A}_J to solve the corresponding problem instance \mathcal{J} in CombBwC. To see this, at each round, \mathcal{A}_I is called and an action is selected $\mathbf{p}(t)$. This action corresponds to an action $\mathbf{l}(t)$ in CombBwC such that for each discretized arm $p_k(t) \in \mathbf{p}(t)$, it falls into the interval $[(2l_k(t)-1)\epsilon - \epsilon, (2l_k(t)-1)\epsilon + \epsilon)$ where $l_k(t) \in \mathbf{l}(t)$. Then algorithm \mathcal{A}_J will observe $\{\tilde{r}(l_k(t))\}$ and receive the reward $\sum_k \tilde{r}(l_k(t))$. After that, $\sum_k \tilde{r}(l_k(t))$ and $\mathbf{p}(t)$ will be further used to compute reward $\sum_k \tilde{r}(p_k(t))$ such that $\mathbb{E}[\sum_k \tilde{r}(p_k(t))] = \sum_{k \in [K]} r(l_k(t))$, and feed it back to \mathcal{A}_I .

At each round, let $\mathbf{p}(t)$ and $\mathbf{l}(t)$ denote the action chosen by the \mathcal{A}_I and \mathcal{A}_J , since we have $r_k(l_k(t)) \geq r_k(p_k(t))$ and best arm of the problem instance \mathcal{I} and \mathcal{J} has the same mean reward $K(1/2 + \epsilon)$, this completes the proof. \square

E.2 Lower bound the $\mathbb{E}[\text{Reg}_{2\epsilon}(T)|\mathcal{J}, \mathcal{A}_J]$

With Lemma E.5 stating the relationship between $\mathbb{E}[\text{Reg}_{2\epsilon}(T)|\mathcal{I}, \mathcal{A}_I]$ and $\mathbb{E}[\text{Reg}_{2\epsilon}(T)|\mathcal{J}, \mathcal{A}_J]$ as derived in (33), we can lower bound the $\mathbb{E}[\text{Reg}_{2\epsilon}(T)|\mathcal{A}_I]$ via deriving the lower bound for $\mathbb{E}[\text{Reg}_{2\epsilon}(T)|\mathcal{A}_J]$.

The structure of the proof is similar to that of [2], while the main difference is that we construct a different set of adversaries to bound the probability of the learner on achieving “good event” (will be specified later). At a high level, our proof builds on the following 4 steps: from step 1 to 3 we restrict our attention to the case of deterministic strategies for the learner, and then we show how to extend the results to arbitrary and randomized strategies by Fubini’s theorem in step 4.

Step 1: Regret Notions. We will also call that the learner is playing against the \mathbf{I}^* -adversary when the current instance is $\mathcal{J}(\mathbf{I}^*, \epsilon)$. We denote by $\mathbb{E}_{\mathbf{I}^*}[\cdot]$ the expectation with respect to the reward generation process of the \mathbf{I}^* -adversary. Without the loss of generality, we assume K is an even number. We write $\mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}$ for the probability distribution of $(j_{2h-1, t}, j_{2h, t})$ when the learner faces the \mathbf{I}^* -adversary. Thus, against the \mathbf{I}^* -adversary, we have

$$\mathbb{E}_{\mathbf{I}^*}[\text{Reg}_{2\epsilon}(T)] = \mathbb{E}_{\mathbf{I}^*} \sum_{t=1}^T \sum_{h=1}^{K/2} 2\epsilon \mathbb{1}_{\{j_{2h-1, t} \neq l_{2h-1}^*, j_{2h, t} \neq l_{2h}^*\}} = T \cdot 2\epsilon \sum_{h=1}^{K/2} \left(1 - \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}(G_T)\right),$$

where G_T denotes the *good event* such that $\{j_{2h-1, T} = l_{2h-1}^*, j_{2h, T} = l_{2h}^*\}$ holds simultaneously for base arm $2h-1$ and $2h$. For a particular distribution $\mathbf{I}^* \sim \mathcal{D}$ for all random adversaries, and let $\text{P}(\mathbf{I}^*)$ denote the support of the adversary \mathbf{I}^* . Because the maximum value is always no less than

the mean, we have

$$\sup_{\mathbf{I}^* \in \mathcal{J}_\epsilon} \mathbb{E}_{\mathbf{I}^*} [\text{Reg}_{2\epsilon}(T)] \geq T \cdot 2\epsilon \sum_{h=1}^{K/2} \left(1 - \sum_{\mathbf{I}^* \in \mathcal{J}_\epsilon} \mathbb{P}(\mathbf{I}^*) \cdot \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}(G_T) \right). \quad (34)$$

Step 2: Information Inequality. Let $\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}$ be the probability distribution of $(j_{2h-1, t}, j_{2h, t})$ against the adversary which plays like the \mathbf{I}^* -adversary except that in the $(2h-1, 2h)$ -th base arms, where the rewards of all discretized arms are drawn from a Bernoulli distribution of parameter $1/2$. We refer to it as $(-h, \mathbf{I}^*)$ -adversary. Let \mathcal{J}_ϵ denote the set of all possible \mathbf{I}^* adversaries and \mathcal{D} be the distribution over \mathbf{I}^* in which \mathbf{I}^* is sampled uniformly at random.

LEMMA E.6. *Let $n_{-h}(K-1+N_p-m), \forall m \in \{2, \dots, 1+N_p\}$ denote the total number of the combinations of $(j_k)_{k \neq 2h-1, 2h}$ such that $\sum_{i \neq 2h-1, 2h} j_k = K-1+N_p-m$. Then we have*

$$\frac{1}{|\mathcal{J}_\epsilon|} \sum_{\mathbf{I}^* \in \mathcal{J}_\epsilon} \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}(G_T) \leq \sum_{m=2}^{N_p+1} \frac{n_{-h}(K-1+N_p-m)}{|\mathcal{J}_\epsilon|} + c\epsilon \sqrt{\frac{T}{|\mathcal{J}_\epsilon|} \sum_{m=2}^{N_p+1} n_{-h}(K-1+N_p-m)}, \quad (35)$$

where c is a constant.

PROOF. Let $\text{KL}(\cdot)$ be the Kullback-Leibler divergence operator. By Pinsker's inequality, we have

$$\mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}(G_T) \leq \mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}(G_T) + \sqrt{\frac{1}{2} \text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*})}, \quad \forall \mathbf{I}^* \in \mathcal{J}_\epsilon.$$

Then by the concavity of the square root,

$$\begin{aligned} & \frac{1}{|\mathcal{J}_\epsilon|} \sum_{\mathbf{I}^* \in \mathcal{J}_\epsilon} \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}(G_T) \\ & \leq \frac{1}{|\mathcal{J}_\epsilon|} \sum_{\mathbf{I}^* \in \mathcal{J}_\epsilon} \mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}(G_T) + \sqrt{\frac{1}{2|\mathcal{J}_\epsilon|} \sum_{\mathbf{I}^* \in \mathcal{J}_\epsilon} \text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*})}. \end{aligned}$$

We introduce $n_h(m), \forall m \in \{2, \dots, 1+N_p\}$ to denote the total number of combinations of (j_{2h-1}, j_{2h}) such that $j_{2h-1} + j_{2h} = m$. Then by definition, it is easy to see that $n_h(m) = m-1$, and furthermore

$$\sum_{m=2}^{N_p+1} n_h(m) \cdot n_{-h}(K-1+N_p-m) = |\mathcal{J}_\epsilon|. \quad (36)$$

Let \mathcal{D} be the distribution over \mathbf{I}^* in which \mathbf{I}^* is sampled uniformly at random, i.e., $\mathbb{P}(\mathbf{I}^*) = \frac{1}{|\mathcal{J}_\epsilon|}$, then by the symmetry of the adversary $(-h, \mathbf{I}^*)$, we have

$$\begin{aligned}
 \sum_{\mathbf{I}^* \in \mathcal{J}_\epsilon} \mathbb{P}(\mathbf{I}^*) \cdot \mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}(G_T) &= \sum_{m=2}^{N_p+1} \sum_{\mathbf{I}^*: \sum_{k \neq 2h-1, 2h} l_k^* = K-1+N_p-m} \mathbb{P}(\mathbf{I}^*) \cdot \mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}(G_T) \\
 &= \sum_{m=2}^{N_p+1} \frac{1}{n_h(m)} \sum_{\mathbf{I}^*: \sum_{k \neq 2h-1, 2h} l_k^* = K-1+N_p-m} \mathbb{P}(\mathbf{I}^*) \\
 &= \sum_{m=2}^{N_p+1} \frac{1}{n_h(m)} \frac{n_h(m) \cdot n_{-h}(K-1+N_p-m)}{|\mathcal{J}_\epsilon|} \\
 &= \sum_{m=2}^{N_p+1} \frac{n_{-h}(K-1+N_p-m)}{|\mathcal{J}_\epsilon|}. \tag{By (36)}
 \end{aligned}$$

□

Step 3: Bounding $\text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^}, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*})$ via the chain rule.* We now proceed to bound the value of $\text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*})$.

LEMMA E.7. $\text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}^T, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}^T) \leq \frac{c\epsilon^2 T}{1-4\epsilon^2} \mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}(G_T)$, where c is the constant value.

PROOF. Given any sequence of observed rewards up to time T , which denoted by $W_T \in \{1, \dots, K\}^T$, the empirical distribution of plays, and, in particular, the probability distribution of $(j_{2h-1, t}, j_{2h, t})$ conditional on the fact that W_T will be the same for all adversaries. Thus, if we denote by $\mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}^T$ (or $\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}^T$) the probability distribution of W_T when the learner plays against the \mathbf{I}^* -adversary (or the $(-h, \mathbf{I}^*)$ -adversary), we can easily show that $\text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}) \leq \text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}^T, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}^T)$. Then we apply the chain rule for Kullback-Leibler divergence iteratively to introduce the probability distributions $\mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}^t$ of the observed rewards W_t up to time t and then will arrive desired result. More formally, we reduce to bound the $\text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}^T, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}^T)$,

$$\begin{aligned}
 &\text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}^T, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}^T) \\
 &= \text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}^1, \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}^1) + \\
 &\quad \sum_{t=2}^T \sum_{w_{t-1} \in \{1, \dots, K\}^{t-1}} \mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}^{t-1}(w_{t-1}) \text{KL}(\mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}(\cdot | w_{t-1}), \mathbb{P}_{(2h-1, 2h), \mathbf{I}^*}(\cdot | w_{t-1})) \\
 &= \text{KL}(\mathcal{B}_0, \mathcal{B}'_0) \mathbb{1}_{j_{2h-1, 1} = l_{2h-1}^*, j_{2h, 1} = l_{2h}^*} + \\
 &\quad \sum_{t=2}^T \sum_{w_{t-1}: j_{2h-1, t-1} = l_{2h-1}^*, j_{2h, t-1} = l_{2h}^*} \mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}^{t-1}(w_{t-1}) \text{KL}(\mathcal{B}_{w_{t-1}}, \mathcal{B}'_{w_{t-1}}) \\
 &= \text{KL}(\mathcal{B}_0, \mathcal{B}'_0) \mathbb{1}_{G_1} + \sum_{t=2}^T \sum_{w_{t-1}: G_{t-1}} \mathbb{P}_{-(2h-1, 2h), \mathbf{I}^*}^{t-1}(w_{t-1}) \text{KL}(\mathcal{B}_{w_{t-1}}, \mathcal{B}'_{w_{t-1}}),
 \end{aligned}$$

where $\mathcal{B}_{w_{t-1}}$ and $\mathcal{B}'_{w_{t-1}}$ are two Bernoulli random variables with parameters in $\{1/2, 1/2 + \epsilon\}$. Due to the fact that $\text{KL}(p, q) \leq \frac{(p-q^2)}{q(1-q)}$, we will have

$$\text{KL}(\mathcal{B}_{w_{t-1}}, \mathcal{B}'_{w_{t-1}}) \leq c \frac{\epsilon^2}{1 - 4\epsilon^2},$$

where c is a constant. Taking the summation will complete the proof. \square

Wrapping up: Proof of Theorem E.2 on Deterministic Strategies. Observe that we can bound

$$\sum_{m=2}^{N_p+1} n_{-h}(K - 1 + N_p - m)/|\mathcal{J}_\epsilon| = \Omega(1/N_p),$$

which follows the fact that: given $a_1 \leq a_2 \leq \dots \leq a_n$ and $b_1 \leq b_2 \leq \dots \leq b_n$, one will have $n \sum a_i b_i \geq \sum a_i \sum b_i$. Plugging back into Eqs. (35) and (34) and substituting $\epsilon = \Theta(T^{-1/3})$ will get the desired result.

Step 4: Fubini's theorem for Random Strategies. For a randomized learner, let \mathbb{E}_{rand} denote the expectation with respect to the randomization of the learner. Then

$$\frac{1}{|\mathcal{J}_\epsilon|} \sum_{\mathbf{l}^* \in \mathcal{J}_\epsilon} \mathbb{E} \sum_{t=1}^T (\mathbf{l}(t)^T \mathbf{r}_t - (\mathbf{l}^*)^T \mathbf{r}_t) = \mathbb{E}_{\text{rand}} \frac{1}{|\mathcal{J}_\epsilon|} \sum_{\mathbf{l}^* \in \mathcal{J}_\epsilon} \mathbb{E}_{\mathbf{l}^*} \sum_{t=1}^T (\mathbf{l}(t)^T \mathbf{r}_t - (\mathbf{l}^*)^T \mathbf{r}_t),$$

where $\mathbf{r}_t = (r_1(l_1(t), \dots, r_K(l_K(t)))$, and value of the reward for not realized arms are computed from Eq (3). The interchange of the integration and the expectation is justified by Fubini's Theorem. For every realization of learner's randomization, the results of all earlier steps still follow. This will give us the same lower bound for $\mathbb{E}_{\text{rand}} \frac{1}{|\mathcal{J}_\epsilon|} \sum_{\mathbf{l}^* \in \mathcal{J}_\epsilon} \mathbb{E}_{\mathbf{l}^*} \sum_{t=1}^T (\mathbf{l}(t)^T \mathbf{r}_t - (\mathbf{l}^*)^T \mathbf{r}_t)$ as we have shown above.

F OPTIMAL DYNAMIC POLICY V.S. BEST POLICY IN HINDSIGHT

As we mentioned, for action-dependent bandits, the optimal dynamic policy can be characterized by a best-in-hindsight (mixed) strategy computing from following constrained optimization problem: $\max_{\mathbf{p} \in \mathcal{P}} \sum_{k=1}^K p_k r_k(p_k)$. While for history-dependent bandits, it is possible that the optimal policy \mathbf{p}^* may not be well-defined due to the fact of reward dependence on action history. However, we argue that when competing against with best-in-hindsight policy, notwithstanding in the face of this kind of reward-history correlation, the value of the optimal strategy is always well-defined in the limit, and this limit value is also characterized by the best-in-hindsight (mixed) strategy computed from action-dependent bandits. To gain intuition, note that the time-discounted frequency $\mathbf{f}(t)$ will be exponentially approach to the fixed strategy \mathbf{p} the learner deploys. As we explain in Section 5, after consistently deploying \mathbf{p} with s rounds, the frequency $\mathbf{f}(t + s)$ will be converging to \mathbf{p} with the exponential decay error γ^s . Thus, to achieve highest expected reward, the learner should deploy the optimal strategy computed as in action-dependent case.

Furthermore, we would like to note that for a general family of reward functions satisfying Definition 6.2, solving optimal dynamic policy in history-dependent bandits can be then reduced to solve a best-in-hindsight fixed policy, this is immediately followed by the the property of Proper reward function.