Cascading Bandits With Feedback

R Sri Prakash
IIITDM Kancheepuram
sriprakash@iiitdm.ac.in

Nikhil Karamchandani
IIT Bombay
nikhilk@ee.iitb.ac.in

Sharayu Moharir
IIT Bombay
sharayum@ee.iitb.ac.in

***Abstract*—Motivated by the challenges of edge inference, we study a variant of the cascade bandit model in which each arm corresponds to an inference model with an associated accuracy and error probability. We analyse four decision-making policies—Explore-then-Commit, Action Elimination, Lower Confidence Bound (LCB), and Thompson Sampling—and provide sharp theoretical regret guarantees for each. Unlike in classical bandit settings, Explore-then-Commit and Action Elimination incur suboptimal regret because they commit to a fixed ordering after the exploration phase, limiting their ability to adapt. In contrast, LCB and Thompson Sampling continuously update their decisions based on observed feedback, achieving constant $\mathcal{O}(1)$ regret. Simulations corroborate these theoretical findings, highlighting the crucial role of adaptivity for efficient edge inference under uncertainty.**

***Index Terms*—cascade bandits**

I. Introduction

The increasing penetration of Artificial Intelligence (AI) and the Internet of Things (IoT) across diverse application domains has led to a growing demand for Mobile Edge Computing (MEC). To achieve faster inference from models trained for specific tasks, it is essential to host them at the edge, thereby reducing latency and improving responsiveness. In this work, we adopt the approach of using a collection of simpler, task-specific models organized in a cascade to perform inference at the edge. A user submits an inference request to the edge, thereby alleviating the computational burden on the User Equipment (UE).

Our system consists of multiple cascaded machine learning (ML) models, each paired with a dedicated scoring module, as shown in Fig. 1, following a design similar to [1]. For a given query, each ML model generates an output, which is then evaluated by its corresponding scoring module. The scoring module maps the model's output to a binary decision: a value of one indicates that the response is capable of satisfying the user's query, while a value of zero indicates otherwise. If the scoring module outputs one, the response from that ML model is forwarded to the user, who subsequently provides binary feedback indicating whether the response is satisfactory (one) or unsatisfactory (zero). Conversely, if the scoring module outputs zero, the query is passed to the next ML model in the cascade. This process continues until a scoring module outputs one or until all scoring modules return zero. In the latter case, the cascade model is unable to satisfy the user's query.

A critical consideration in our model is that a scoring module's output of 1 does not guarantee a satisfactory user experience, as the user may still provide feedback of 0. We account for this by modeling a stochastic error probability for each scoring module. This error is only detectable when a scoring module returns a 1, but the user's subsequent feedback is 0. The reward obtained in this setting is defined as 1 if the user feedback is satisfactory, and 0 if the feedback is unsatisfactory or if the system fails to serve the request. The central objective of this work is to determine the optimal ordering of the ML models within the cascade so as to maximize the total reward or, equivalently, to minimize the regret, defined as the difference between the reward of the policy under consideration and that of the static optimal policy.

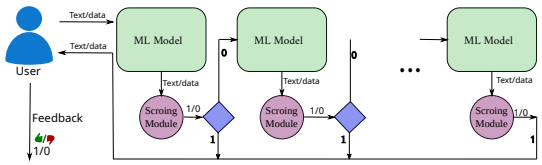

Fig. 1: Cascade ML models with scoring modules

We model our edge inference setup as a cascade of ML models, each paired with a scoring module as a cascade bandit problem with feedback. Each model–scorer pair is treated as a bandit arm, and the system output corresponds to the first model in the cascade whose scorer outputs one. This setting is closely related to the cascade bandit framework [2], [3], where a list of items is presented to a user and a reward is obtained if an item is clicked. In our case, the agent selects the first model with a positive score, presents its response, and receives binary user feedback: one if satisfactory, zero otherwise. If all scorers output zero, the request is unserved, analogous to the "no-click" outcome in cascade bandits. Unlike [4], which studies best-arm identification, we focus on regret minimisation; and unlike [5], our feedback is immediate rather than randomly delayed.

Our model differs from the conventional cascade bandit setting in that the reward depends directly on the order of models in the cascade, unlike the standard setup, where order does not affect slot reward. It is also related to cost-aware cascading bandits [6]–[9], where ordering impacts the total reward due to examination costs. The closest prior work is [10], where an agent presents a list of items and optimises the order to maximise reward. While our LCB algorithm differs from their UCB variant, we adapt their theoretical techniques and further leverage results from [11]–[14] to establish sharp regret guarantees.

The key contributions of this work are as follows. We show that any policy which becomes static after a certain number

Nikhil Karamchandani's work was supported by a SERB MATRICS grant.

of time slots—such as explore-and-commit algorithms—will incur a regret of order $\Omega(\log T)$. We then analyze the performance of the Lower Confidence Bound (LCB) algorithm and the Thompson Sampling algorithm in our cascade bandit setting, and demonstrate that both achieve $\mathrm{O}(1)$ regret. Furthermore, we validate these theoretical results through extensive simulations, showing strong consistency between the theoretical guarantees and empirical performance.

II. Problem Setup

We consider a K-arm cascade bandit problem, where the set of arms is denoted by $[K] = \{1, 2, \ldots, K\}$. Each arm produces a binary output independently of the others. Let $X_i(t)$ denote the output of arm i at time slot t (representing the scoring module's decision), where $\{X_i(t)\}_{t\geq 1}$ are i.i.d. Bernoulli random variables with mean $\mu_i = \mathbb{E}[X_i(t)]$.

At each time slot t, the learner selects an ordering of the arms, denoted by $\mathcal{L}_t = (l_1^{(t)}, l_2^{(t)}, \ldots, l_K^{(t)})$, where $l_j^{(t)}$ denotes the arm placed in position j of the cascade. The cascade is traversed sequentially in this order until the first arm that outputs one is encountered. Formally, the selected arm index at time t is $I_t = \min\{j \in [K] : X_{l_j^{(t)}}(t) = 1\}$, with the convention that $I_t = \infty$ if all arms output zero.

The response of the selected arm $l_{I_t}^{(t)}$ is shown to the user, who provides feedback $Y_{l_{I_t}^{(t)}}(t) \in \{0, 1\}$ indicating whether the displayed result was relevant. The user feedback is treated as the reward in slot t. If $I_t = \infty$, no arm is displayed and the reward is zero. An arm i is thus observed only if $X_i(t) = 1$ and all arms preceding it in the ordering $\mathcal{L}_t$ output zero.

An error is said to occur if $X_i(t) = 1$ and $Y_i(t) = 0$. These errors are stochastic, with error probability defined as $p_i = \mathbb{P}(Y_i(t) = 0 \mid X_i(t) = 1), \quad \forall i \in [K]$. Without loss of generality, we assume the arms are indexed such that $p_1 < p_2 < \cdots < p_K$. For notational convenience, we also define the gaps $\Delta_i = p_i - p_{i-1}$ for $2 \leq i \leq K$.

The goal is to maximize the cumulative reward, or equivalently, minimize the cumulative regret, by dynamically adapting the ordering of the arms in the cascade. Let $\mathcal{L}^* = (l_1^*, l_2^*, \ldots, l_K^*)$ denote the optimal ordering of the arms, which sorts arms in increasing order of their error probabilities. Furthermore, let $l_t^{-1}(i)$ denote the position of arm i in the ordering $\mathcal{L}_t$. The regret of a policy then quantifies the expected reward loss incurred relative to always playing $\mathcal{L}^*$.

We define the expected reward of a cascade model when the arms are ordered according to the ordering $\mathcal{L} = (l_1, l_2, \ldots, l_K)$ as

$$r_{\mathcal{L}} = \sum_{i=1}^{K} (1 - p_{l_i}) \mu_{l_i} \prod_{j=1}^{i-1} (1 - \mu_{l_j}).$$

This expression reflects that the i-th arm in the ordering contributes to the reward only if all preceding arms $1, \ldots, i-1$ fail to produce an output 1, which occurs with probability $\prod_{j=1}^{i-1}(1 - \mu_{l_j})$. Given that arm i is shown, it produces a satisfactory feedback with probability $(1 - p_{l_i})$, and the event of the arm being triggered occurs with probability μ_{l_i}. Hence, the summation accounts for the expected contribution of each arm to the overall reward under ordering $\mathcal{L}$.

We define the suboptimality gap of an ordering $\mathcal{L}$ as $\tilde{\Delta}_{\mathcal{L}} = r_{\mathcal{L}^*} - r_{\mathcal{L}}$, which quantifies the loss in expected reward when using $\mathcal{L}$ instead of the optimal ordering $\mathcal{L}^*$. For each arm i, let $\mathcal{M}_i$ be the set of all orderings starting with i, and define $\tilde{\Delta}_i = \max_{\mathcal{L} \in \mathcal{M}_i} \tilde{\Delta}_{\mathcal{L}}$ as the maximum suboptimality incurred by choosing i as the first arm. The largest such value across arms, $\tilde{\Delta}_{max} = \max_i \tilde{\Delta}_i$, represents the worst-case loss from starting with a poor arm, while the smallest nonzero gap, $\tilde{\Delta}_{min} = \min_{\mathcal{L} \neq \mathcal{L}^*} \tilde{\Delta}_{\mathcal{L}}$, captures how hard it is to distinguish the optimal ordering from its closest suboptimal alternative.

Regret is defined as the difference in expected reward between the optimal policy and the policy under consideration. Let $\mathcal{R}^\pi(T)$ represent the expected regret of policy π until time T, and let $\mathcal{R}_t^\pi$ denote the expected regret incurred in slot t:

$$\mathcal{R}_t^\pi = r_{\mathcal{L}^*} - r_{\mathcal{L}_t}$$
$$\mathcal{R}^\pi(T) = \sum_{t=1}^{T} \mathcal{R}_t^\pi$$

The objective is to select the order of arms $\mathcal{L}_t$ in each round to minimize the cumulative regret. This essentially means we want to find a policy that, over time, consistently chooses an arm ordering that is as close as possible to the optimal one.

III. Results

In this section, we will first define the optimal static policy for our problem and then introduce several online algorithms designed to minimise regret. These algorithms—Explore and Commit (EC), Action Elimination (AE), Lower Confidence Bound (LCB), and Thompson Sampling (TS) — each come with specific performance guarantees.

A. Static Optimal Policy

Theorem 1. *The optimal static policy will order the arms in increasing order of their error probabilities* (p_i)

The proof of Theorem 1 is provided in the Appendix [15].

Remark 1. *The optimal ordering of arms is independent of their means* (μ_i)*, which are used for sampling. Instead, the ordering relies exclusively on the probability of error* (p_i)*, meaning arms with a lower error rate are given priority and positioned first*

B. Explore and Commit

The Explore and Commit (EC) algorithm, formally described in Algorithm 1, is a two-phase strategy designed to balance exploration with exploitation in the cascade bandit setting. It begins with an exploration phase, where each arm is pulled an equal number of times in order to estimate its probability of error. The number of pulls per arm is defined as $N = \max_i n_i$, where $n_i = \lceil \frac{16 \log T}{\Delta_i^2 \mu_i} \rceil$, and the total exploration period lasts for $T_s^{EC} = NK$ slots. During this phase, the algorithm cycles through the arms in a rotating order, ensuring that each arm appears in every position of the cascade. At

each time slot, the cascade presents the output of the first triggered arm to the user, whose feedback is then used to update the empirical error probability of the selected arm and to compute lower confidence bounds on these estimates. Once the exploration phase is complete, the algorithm enters the commit phase, where the arms are permanently ordered in ascending order of their estimated error probabilities. This committed ordering is then used for the remainder of the horizon.

Algorithm 1: Explore and Commit

Input: Δ_i, T, μ_i
Output: Ordered list $\mathcal{L}_t$

Initialize: $\mathcal{L}_1 =$ random order of arms, $t = 1$
  $N = \max_i \lceil \frac{16 \log T}{\Delta_i^2 \mu_i} \rceil$
Explore:
for $t \leq NK$ **do**
  Order arms according to $\mathcal{L}_t$
  The cascade model shows the result of arm I_t to the user
  Feedback $Y_{I_t}(t)$ is observed
  for $i \in [K]$ **do**
    $S_i(t) = S_i(t-1) + \mathbb{1}_{\{i=I_t\}}$
    $\hat{p}_i(t) = \left\{ \sum_{n=1}^{t} \mathbb{1}_{\{i=I_n\}} (1 - Y_i(t)) \right\} / S_i(t)$
    $L_i(t) = \hat{p}_i(t) - \sqrt{\frac{2 \log T}{S_i(t)}}$
  end
  $\mathcal{L}_{t+1} = (l_2^{(t)}, l_3^{(t)}, \cdots, l_K^{(t)}, l_1^{(t)})$
  $t++$
end
Commit:
$\mathcal{L}'$= List arms in ascending order of $L_i(t-1)$
for $t < T$ **do**
  $\mathcal{L}_t = \mathcal{L}'$
  $t++$
end

Lemma 1. *The probability that Algorithm 1 selects a suboptimal ordering in the commit phase is bounded by*

$$\mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*) \leq \frac{K}{T^2} + \frac{K^2}{T^4}.$$

The proof of Lemma 1 is provided in the Appendix [15].

Theorem 2. *The regret obtained by Algorithm 1 satisfies*

$$\mathcal{R}^{EC}(T) = \mathrm{O}(\log T).$$

Proof. The total regret can be decomposed into contributions from the exploration phase and the commit phase:

$$\mathcal{R}^{EC}(T) = \sum_{t=1}^{T_s^{EC}} \mathbb{E}[\mathcal{R}_t] + \sum_{t=T_s^{EC}+1}^{T} \mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*) \, \tilde{\Delta}_{\mathcal{L}_t}.$$

Using Lemma 1, we obtain

$$\mathcal{R}^{EC}(T) \leq NK \, \tilde{\Delta}_{\max} + T \tilde{\Delta}_{\max} \left(\frac{K}{T^2} + \frac{K^2}{T^4} \right).$$

Since $N = \mathrm{O}(\log T)$, the first term scales as $\mathrm{O}(\log T)$, and the second term is asymptotically negligible. Hence,

$$\mathcal{R}^{EC}(T) = \mathrm{O}(\log T).$$

□

Theorem 3. *The regret of Algorithm 1 is lower bounded as*

$$\mathcal{R}^{EC}(T) = \Omega(\log T).$$

The proof of Theorem 3 is provided in Appendix [15].

However, a key limitation of EC is that the required number of pulls N depends on the gap parameters Δ_i, which are generally unknown in practice, making the algorithm difficult to implement in real-world settings. This naturally motivates the use of adaptive strategies such as *Action Elimination*, which overcome this drawback by eliminating suboptimal arms based on observed feedback without requiring prior knowledge of Δ_i.

C. Action Elimination

In this subsection, we analyse the Action Elimination (AE) algorithm, formally defined in Algorithm 2. The algorithm maintains two disjoint sets of arms: an *active set* $\mathcal{A}_t$, containing arms that are not yet sufficiently explored, and an *inactive set* $\mathcal{B}_t$, with $\mathcal{A}_t \cap \mathcal{B}_t = \varnothing$. Initially, $\mathcal{A}_t = [K]$ and $\mathcal{B}_t = \varnothing$. Let $\mathcal{L}_{\mathcal{A}_t}$ and $\mathcal{L}_{\mathcal{B}_t}$ denote the ordered lists of active and inactive arms, respectively. If an arm is removed from $\mathcal{A}_t$, the size of $\mathcal{L}_{\mathcal{A}_t}$ decreases but the relative ordering among the remaining active arms is preserved, while $\mathcal{L}_{\mathcal{B}_t}$ may be reordered when new arms are added. The overall ortdering $\mathcal{L}_t$ is formed by concatenating $\mathcal{L}_{\mathcal{A}_t}$ and $\mathcal{L}_{\mathcal{B}_t}$. To ensure that every active arm continues to receive exploration opportunities, $\mathcal{L}_{\mathcal{A}_t}$ is rotated in a round-robin manner across slots, whereas $\mathcal{L}_{\mathcal{B}_t}$ is maintained in ascending order of LCB values.

While $|\mathcal{A}_t| > 1$ (active phase), the algorithm updates empirical error probabilities and their confidence intervals (LCB and UCB). Arms are eliminated from $\mathcal{A}_t$ and permanently added to $\mathcal{B}_t$ whenever their confidence intervals no longer overlap with others. Once only one arm remains active (commit phase), all arms are ordered by LCB values, and this final order is fixed for the rest of the horizon. Thus, AE progressively eliminates suboptimal arms while refining the cascade ordering.

Unlike the Explore and Commit (EC) algorithm, which explores uniformly for a fixed budget before committing, AE adaptively eliminates inferior arms as soon as enough evidence is gathered. This reduces unnecessary exploration and can yield tighter regret guarantees.

For the analysis, we introduce the following notations. Let $\Delta_i' = \min\{p_i - p_{i-1}, p_{i+1} - p_i\}$ denote the minimum gap in error probability that distinguishes arm i from its immediate neighbors. Define N_i as the number of pulls required for arm i to collect $\frac{16 \log T}{\Delta_i'^2}$ feedback samples when placed at the head of the cascade, and let $N_i(t)$ denote the number of times arm i has occupied the first position up to time t. Since Algorithm 2 rotates arms in a round-robin fashion, each active arm is guaranteed opportunities to accumulate these samples.

Algorithm 2: Action elimination

Output: Ordered list $\mathcal{L}_t$

Initialize $\mathcal{A}_1 = \{1, 2, \cdots, K\}$, $\mathcal{B}_1 = \{\}$ $\mathcal{L}_1 =$ random order of arms, $t = 1$, $\mathcal{L}_{\mathcal{A}_t} = \mathcal{L}_1$, $\mathcal{L}_{\mathcal{B}_t} = ()$, $S_i(0) = 0$, $\hat{p}_i(0) = 0$

for $t \leq T$ **do**

  Order arms as in list $\mathcal{L}_t$

  The cascade model shows the result of arm I_t to the user

  Feedback of result $Y_{I_t}(t)$ is observed

  for $i \in [K]$ **do**

    $S_i(t) = S_i(t-1) + \mathbb{1}_{\{i=I_t\}}$

    $\hat{p}_i(t) = \left\{\sum_{n=1}^{t} \mathbb{1}_{\{i=I_n\}}(1 - Y_i(t))\right\} / S_i(t)$

    $L_i(t) = \hat{p}_i(t) - \sqrt{\frac{2\log T}{S_i(t)}}$

    $U_i(t) = \hat{p}_i(t) + \sqrt{\frac{2\log T}{S_i(t)}}$

  end

  if $[L_i(t), U_i(t)] \cap [L_j(t), U_j(t)] = \phi, \forall i \neq j$ **then**

    $\mathcal{A}_{t+1} = \mathcal{A}_t / \{j\}$

    $\tilde{\mathcal{L}}_{\mathcal{A}_{t+1}} = \mathcal{L}_{\mathcal{A}_t} / \{j\}$

    $\mathcal{B}_{t+1} = \mathcal{B}_t \cup \{j\}$

  end

  if $|\mathcal{A}_{t+1}| > 1$ **then**

    $\mathcal{L}_{A_{t+1}} = (\tilde{l}_2^{A_t}, \tilde{l}_3^{A_t}, \cdots, \tilde{l}_{|\mathcal{A}_t|}^{A_t}, \tilde{l}_1^{A_t})$

    $\mathcal{L}_{\mathcal{B}_{t+1}} =$ ascending order over $L_i(t)$, $i \in \mathcal{B}_{t+1}$

    $\mathcal{L}_{t+1} = (\mathcal{L}_{\mathcal{A}_{t+1}}, \mathcal{L}_{\mathcal{B}_{t+1}})$

  end

  else

    $\mathcal{L}_{t+1} =$ ascending order over $L_i(t)$

  end

  $t++$

end

Let T_s^{AE} represent the time at which the active phase ends. Finally, define the event $\mathcal{E}_t$ as

$$\mathcal{E}_t = \left\{|\hat{p}_i(t) - p_i| < \epsilon_i(t), \quad \forall i\right\},$$

where $\epsilon_i(t) = \sqrt{\frac{2\log T}{S_i(t)}}$ and $S_i(t)$ is the number of times arm i has been observed up to time t.

Theorem 4. *The regret obtained by Algorithm 2 is*

$$\mathcal{R}^{AE}(T) = \mathrm{O}(\log T).$$

Proof.

$$\begin{aligned}
\mathcal{R}^{AE}(T) &\leq \sum_{t=1}^{T_s^{AE}} \mathbb{E}[\mathcal{R}_t] + \sum_{t=T_s^{AE}+1}^{T} \mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*)\tilde{\Delta}_{max} \\
&\overset{(a)}{\leq} \sum_{i=1}^{K} \mathbb{E}[N_i]\tilde{\Delta}_i + \frac{K\tilde{\Delta}_{max}}{T} + \frac{K\tilde{\Delta}_{max}}{T} \\
&\leq \sum_{i=1}^{K} \frac{16\log T}{\Delta_i'^2 \mu_i}\tilde{\Delta}_i + \frac{2K\tilde{\Delta}_{max}}{T} \\
&= \mathrm{O}(\log T),
\end{aligned}$$

where (a) is obtained by using Lemma 2 and Lemma 3. □

Lemma 2. *The probability of Algorithm 2 choosing a sub-optimal ordering in the commit phase is bounded as follows*

$$\mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*) \leq \frac{K}{T^2}.$$

Lemma 3. *For Algorithm 2, regret in active phase is bounded as follows*

$$\sum_{t=1}^{T_s^{AE}} \mathbb{E}[\mathcal{R}_t] \leq \sum_{i=1}^{K} \mathbb{E}[N_i]\tilde{\Delta}_i + \frac{K\tilde{\Delta}_{max}}{T}.$$

Lemma 4. *Let $f(t)$ be a function of t such that $0 \leq f(t) \leq t$. Then, for constants $\alpha_1, \alpha_2, \alpha_3 > 0$, we have*

$$\alpha_1 f(t) + \alpha_2 e^{-\alpha_3 f(t)}\,(t - f(t)) = \Omega(\log t).$$

The proof of Lemma 2, 3, 4 is provided in the Appendix [15].

Theorem 5. *The regret of Algorithm 2 is lower bounded as*

$$\mathcal{R}^{AE}(T) = \Omega(\log(T)).$$

Proof. Let δ denote the probability that the algorithm commits to a sub-optimal ordering, and let T_s^{AE} represent the number of rounds spent in the active phase before committing.

By results on best arm identification with fixed confidence in the full-information setting [16], the expected length of the active phase must satisfy

$$\mathbb{E}[T_s^{AE}] \;\geq\; \beta' \log\left(\tfrac{1}{\delta}\right),$$

for some constant $\beta' > 0$.

An important property of Algorithm 2 is that all arms in the active set $\mathcal{A}_t$ are explored in a round-robin manner through the orderings $\mathcal{L}_{\mathcal{A}_t}$. The algorithm remains in the active phase as long as at least two arms are active. Consequently, the maximum number of times any single ordering can be selected is $T_s^{AE}/2$, which implies that a sub-optimal ordering is chosen in at least half of the slots. This structural property of the algorithm directly contributes to the regret incurred during the active phase. Therefore,

$$\mathcal{R}(T_s^{AE}) \;\geq\; \frac{\mathbb{E}[T_s^{AE}]}{2}\,\tilde{\Delta}_{\min} \;\geq\; \frac{1}{2}\beta' \log\left(\tfrac{1}{\delta}\right)\tilde{\Delta}_{\min}.$$

After the active phase, if the commit phase chooses a sub-optimal ordering (which happens with probability at least δ), regret continues to accumulate linearly with rate $\tilde{\Delta}_{\min}$. Thus, the total regret satisfies

$$\begin{aligned}
\mathcal{R}^{AE}(T) \;&\geq\; \mathcal{R}(T_s^{AE}) + \sum_{t=T_s^{AE}+1}^{T} \delta\,\tilde{\Delta}_{\min} \\
&\geq\; \frac{1}{2}\beta' \log\left(\tfrac{1}{\delta}\right)\tilde{\Delta}_{\min} + (T - T_s^{AE} - 1)\delta\tilde{\Delta}_{\min}.
\end{aligned} \tag{1}$$

Finally, applying Lemma 4 to (1) yields the claimed lower bound,

$$\mathcal{R}^{AE}(T) = \Omega(\log(T)).$$

□

Remark 2. *From the lower bound analysis, we can conclude that any policy that commits after a certain exploration period will necessarily incur a regret of order* $\Omega(\log T)$.

D. LCB

In this subsection, we analysed the Lower Confidence Bound (LCB) algorithm, formally described in Algorithm 3. The key idea of LCB is to order the arms at each round according to their estimated error probabilities, adjusted by a confidence term that encourages exploration. Initially, each arm is pulled until at least one user feedback is obtained to ensure a valid estimate. At every round t, the arms are ranked in ascending order of their lower confidence bounds $L_i(t)$, defined as the empirical error estimate $\hat{p}_i(t)$ reduced by a confidence margin $\sqrt{\frac{2\log t}{S_i(t)}}$, where $S_i(t)$ is the number of times feedback for arm i has been observed. This construction balances exploration and exploitation by prioritising arms that either appear to have lower error probability or are still under-explored. Over time, the ordering of arms converges toward the optimal cascade order as the confidence intervals shrink with more observations.

Algorithm 3: LCB

Output: Ordered list $\mathcal{L}_t$

Pull each arm till at least one user feedback is obtained

while $t \leq T$ **do**

  $\mathcal{L}_t$= List arms in ascending order of $L_i(t)$

  Observe user feedback: I_t

  $S_{I_t}(t) = S_{I_t}(t-1) + 1$

  $\hat{p}_{I_t}(t) = \hat{p}_{I_t}(t-1) + \frac{1}{S_{I_t}(t)}(1 - Y_{I_t}(t) - \hat{p}_{I_t}(t-1))$

  $L_i(t) = \hat{p}_i(t) - \sqrt{\frac{2\log t}{S_i(t)}}$, for all $i \in [K]$

end

Theorem 6. *The regret obtained by Algorithm 3 is*

$$\mathcal{R}^{LCB}(T) = \mathrm{O}(1).$$

The proof of Theorem 6 is provided in the Appendix [15].

Theorem 6 shows that the regret of the LCB algorithm is $\mathrm{O}(1)$, which is significantly stronger than the $\Omega(\log T)$ lower bounds established for EC (Theorem 3) and AE (Theorem 5). It is worth noting that, in the standard stochastic bandit setting, both Explore-then-Commit and Action Elimination are known to achieve order-optimal regret guarantees. However, this is no longer the case in our problem, where the commitment inherent in these algorithms leads to $\Omega(\log T)$ regret. In contrast, the LCB algorithm avoids committing to a single arm, instead adapting continuously through confidence-bound updates, which allows it to achieve significantly lower regret in our setting.

E. Thompson Sampling

In this subsection, we analysed the Thompson Sampling (TS) algorithm, formally defined in Algorithm 4. TS is a Bayesian approach that maintains a posterior distribution over the error probability of each arm, modelled using Beta priors. At each round t, a sample $\theta_i(t)$ is drawn from the Beta distribution $\text{Beta}(\alpha_i(t), \beta_i(t))$ for each arm $i \in [K]$. The arms are then ordered in ascending order of these sampled values, thereby balancing exploration and exploitation through randomisation. After observing the user feedback $Y_{I_t}(t)$ for the triggered arm I_t, the corresponding posterior parameters are updated: α_{I_t} is incremented when the feedback indicates success, and β_{I_t} is incremented otherwise. This sampling-based update mechanism ensures that arms with higher uncertainty are explored more frequently, while arms with consistently low error probabilities are more likely to appear earlier in the cascade, leading to convergence toward the optimal ordering.

Algorithm 4: Thompson sampling

Output: Ordered list $\mathcal{L}_t$

Initialize: $\alpha_i = 1$, $\beta_i = 1$ for all $i \in [K]$

while $t \leq T$ **do**

  Generate Thompson sample $\theta_i(t) \sim \text{Beta}(\alpha_i(t), \beta_i(t))$ for all $i \in [K]$

  $\mathcal{L}_t$= List arms in ascending order of $\theta_i(t)$

  The cascade model shows the result of arm I_t to the user

  Feedback of result $Y_{I_t}(t)$ is observed

  $\alpha_{I_t}(t+1) = \alpha_{I_t}(t) + 1 - Y_{I_t}(t)$, $\beta_{I_t}(t+1) = \beta_{I_t}(t) + Y_{I_t}(t)$.

end

Theorem 7. *The regret obtained by Algorithm 4 satisfies*

$$\mathcal{R}^{TS}(T) = \mathrm{O}(1).$$

Proof. Regret is incurred at time slot t if $I_t = i$ and there exists k such that $p_i > p_k$ while arm i appears ahead of arm k in the cascade, i.e., $l_t^{-1}(i) < l_t^{-1}(k)$. Hence, we can upper bound the cumulative regret as

$$\begin{aligned}&\mathcal{R}(T)\\ &\leq \tilde{\Delta}_{max}\cdot\\ &\quad \mathbb{E}\left[\sum_{t=1}^{T}\sum_{i=2}^{K} \mathbb{1}\{I_t = i, \exists k \text{ s.t } p_i > p_k, l_t^{-1}(i) < l_t^{-1}(k)\}\right].\end{aligned}$$

Define the event $A_{i,k}(t)$ as $\{I_t = i, l_t^{-1}(i) < l_t^{-1}(k), p_i > p_k\}$. Let $E^p_{i,k}(t)$ denote the event $\{\hat{p}_i(t) < p_i - \Delta_{i,k}/4\}$ and $E^\theta_{i,k}(t)$ denote $\{\theta_i(t) < p_k + \Delta_{i,k}/4\}$, where $\Delta_{i,k} = p_i - p_k$.

Therefore, the regret decomposition becomes

$$\begin{aligned}\mathcal{R}(T) \leq \tilde{\Delta}_{\max} \sum_{i=2}^{K}\sum_{k=1}^{K-1} \mathbb{E}\Bigg[\sum_{t=1}^{T} &\mathbb{1}\{A_{i,k}(t), E^p_{i,k}(t)\}\\ &+ \mathbb{1}\{A_{i,k}(t), \bar{E}^p_{i,k}(t), E^\theta_{i,k}(t)\}\\ &+ \mathbb{1}\{A_{i,k}(t), \bar{E}^p_{i,k}(t), \bar{E}^\theta_{i,k}(t)\}\Bigg]. \qquad (2)\end{aligned}$$

Finally, by applying Lemmas 5, 6, and 7, which respectively control the contributions of each of the three terms in (2), we

conclude that the cumulative regret is bounded by a constant, i.e., $\mathcal{R}^{TS}(T) = \mathrm{O}(1)$. □

Lemma 5. *The regret of the first term in* (2) *is bounded as follows*

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), E^{p}_{i,k}(t)\}\right] \leq 1 + \frac{16}{\Delta^2_{i,k}}.$$

Lemma 6. *The regret of the second term in* (2) *is bounded as follows*

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), \bar{E}^{p}_{i,k}(t), E^{\theta}_{i,k}(t)\}\right] \leq \frac{32}{\Delta^2_{i,k}\bar{\mu}_i} + \frac{2}{\bar{\mu}_i^2} + \frac{\pi^2}{6}.$$

Lemma 7. *The regret of the third term in* (2) *is bounded as follows*

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), \bar{E}^{p}_{i,k}(t), \bar{E}^{\theta}_{i,k}(t)\}\right] \\ \leq \frac{24}{\epsilon^2} + \frac{2c_1}{\epsilon^2} + \frac{c_1 e^{\epsilon^2/2}}{\epsilon^2}\left(\frac{1}{16\epsilon}\mathbb{1}\{16\epsilon < 1\} + \frac{1}{e}\right) \\ + \frac{8c_1}{\epsilon} - \frac{4c_1 \log(e^{2\epsilon} - 1)}{\epsilon^2},\end{aligned}$$

where c_1 is a constant.

The proof of Lemma 5, 6, 7 are provided in the Appendix [15].

From Theorem 7 we observe that Thompson Sampling also outperforms EC and AE in terms of regret minimization. Similar to LCB, Thompson Sampling avoids committing to a fixed arm after an exploration phase. Instead, it maintains a posterior distribution over arm parameters and samples from it at each round, thereby naturally balancing exploration and exploitation throughout the horizon. This probabilistic updating enables Thompson Sampling to achieve constant regret, whereas EC and AE incur $\Omega(\log T)$ regret due to their one-time commitment strategy.

IV. Simulations

In this section, we validate our theoretical findings through simulations. We consider the case of $K = 5$ arms with parameters $\mu = [0.85, 0.9, 0.95, 0.92, 0.87]$ and $p = [0.1, 0.25, 0.4, 0.55, 0.7]$. The results are averaged over 20 independent experiments. As shown in Fig. 2, we compare the regret of the policies from Section III for varying horizon T. The results indicate that the Thompson Sampling and LCB algorithms achieve constant regret, whereas the Action Elimination and Explore-and-Commit algorithms exhibit logarithmic regret growth. These observations are consistent with, and hence validate, the theoretical guarantees established in Section III. The superior performance of Thompson Sampling and LCB arises from their ability to continuously balance exploration and exploitation, thereby adapting to uncertainty throughout the horizon, while Action Elimination and Explore-and-Commit commit after the exploration period, which inherently leads to $\Omega(\log T)$ regret. To ensure fair initialisation, we assign a large value to the LCB at the beginning and break ties randomly, thereby avoiding additional delays for initial sampling.

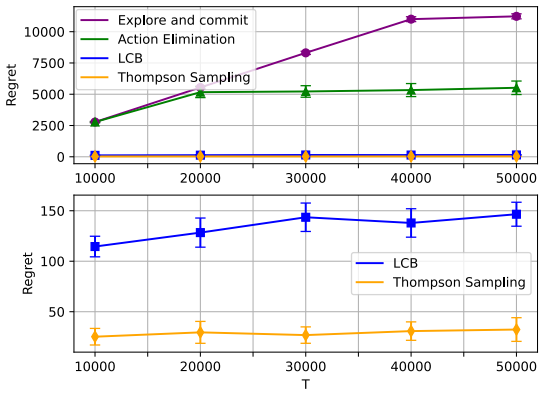

Fig. 2: Comparison of regret for different policies

To analyze how regret evolves over time and how frequently each policy selects a suboptimal ordering, we plot the cumulative regret up to $T = 5 \times 10^4$ rounds for the case of $K = 5$ arms with parameters $\mu = [0.85, 0.9, 0.95, 0.92, 0.87]$ and $p = [0.1, 0.25, 0.4, 0.55, 0.7]$. The results are presented in Fig. 3. We observe that the Explore-and-Commit and Action Elimination algorithms incur constant regret once they enter the commit phase. In contrast, the LCB and Thompson Sampling algorithms continue to accumulate regret, but at a much slower rate. While they initially incur small regret due to exploration, their regret growth eventually saturates and becomes sublinear, highlighting their superior ability to balance exploration and exploitation over time.

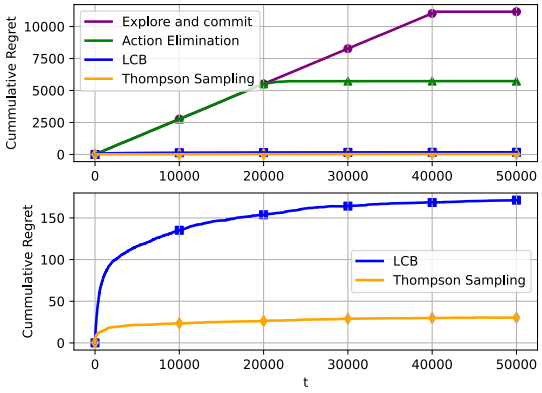

Fig. 3: Comparison of cumulative regret of different polices

REFERENCES

[1] L. Chen, M. Zaharia, and J. Zou, "Frugalgpt: How to use large language models while reducing cost and improving performance," *arXiv preprint arXiv:2305.05176*, 2023.

[2] B. Kveton, C. Szepesvari, Z. Wen, and A. Ashkan, "Cascading bandits: Learning to rank in the cascade model," in *International conference on machine learning*. PMLR, 2015, pp. 767–776.

[3] S. Zong, H. Ni, K. Sung, N. R. Ke, Z. Wen, and B. Kveton, "Cascading bandits for large-scale recommendation problems," *arXiv preprint arXiv:1603.05359*, 2016.

[4] Z. Zhong, W. C. Cheung, and V. Tan, "Best arm identification for cascading bandits in the fixed confidence setting," in *International Conference on Machine Learning*. PMLR, 2020, pp. 11 481–11 491.

[5] D. Wang, J. Cao, Y. Zhang, and W. Qi, "Cascading bandits: optimizing recommendation frequency in delayed feedback environments," *Advances in Neural Information Processing Systems*, vol. 36, pp. 78 894–78 905, 2023.

[6] C. Gan, R. Zhou, J. Yang, and C. Shen, "Cost-aware cascading bandits," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3692–3706, 2020.

[7] L. Tran-Thanh, A. Chapman, A. Rogers, and N. Jennings, "Knapsack based optimal policies for budget–limited multi–armed bandits," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 26, no. 1, 2012, pp. 1134–1140.

[8] A. Burnetas and O. Kanavetas, "Adaptive policies for sequential sampling under incomplete information and a cost constraint," in *Applications of mathematics and informatics in military science*. Springer, 2012, pp. 97–112.

[9] W. Ding, T. Qin, X.-D. Zhang, and T.-Y. Liu, "Multi-armed bandit with budget constraint and variable costs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 27, no. 1, 2013, pp. 232–238.

[10] D. Cheng, R. Huang, C. Shen, and J. Yang, "Cascading bandits with two-level feedback," in *2022 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2022, pp. 1892–1896.

[11] J.-Y. Audibert and S. Bubeck, "Best arm identification in multi-armed bandits," in *COLT-23th Conference on learning theory-2010*, 2010, pp. 13–p.

[12] S. Wang and W. Chen, "Thompson sampling for combinatorial semi-bandits," in *International Conference on Machine Learning*. PMLR, 2018, pp. 5114–5122.

[13] S. Agrawal and N. Goyal, "Near-optimal regret bounds for thompson sampling," *Journal of the ACM (JACM)*, vol. 64, no. 5, pp. 1–24, 2017.

[14] Z. Zhong, W. C. Chueng, and V. Y. Tan, "Thompson sampling algorithms for cascading bandits," *Journal of Machine Learning Research*, vol. 22, no. 218, pp. 1–66, 2021.

[15] https://tinyurl.com/ynjwpcf8/, 2025, [Online; accessed 30-Oct-2025].

[16] T. Lattimore and C. Szepesvári, *Bandit algorithms*. Cambridge University Press, 2020.

1

V. APPENDIX

Proof of Theorem 1. We prove this result by contradiction. Assume that the optimal arm ordering, denoted by $\mathcal{L}^*$, is not sorted by increasing p_i values. This implies there exists an adjacent pair of arms, l_i^* and l_{i+1}^*, such that $p_{l_i^*} > p_{l_{i+1}^*}$.

The expected reward of the assumed optimal ordering $\mathcal{L}^*$ is given by

$$r^* = \sum_{i=1}^{K}(1-p_{l_i^*})\mu_{l_i^*}\prod_{j=1}^{i-1}(1-\mu_{l_j^*}).$$

Now, let's consider a new ordering, $\tilde{\mathcal{L}}$, obtained by swapping arms l_i^* and l_{i+1}^*. The difference in expected reward between the two orderings is

$$\begin{aligned}
r^* - \tilde{r} &= \left(\prod_{j=1}^{i-1}(1-\mu_{l_j^*})\right)\Big[(1-p_{l_i^*})\mu_{l_i^*} + (1-p_{l_{i+1}^*})\mu_{l_{i+1}^*}(1-\mu_{l_i^*}) \\
&\quad - \Big((1-p_{l_{i+1}^*})\mu_{l_{i+1}^*} + (1-p_{l_i^*})\mu_{l_i^*}(1-\mu_{l_{i+1}^*})\Big)\Big] \\
&= \left(\prod_{j=1}^{i-1}(1-\mu_{l_j^*})\right)\Big[(1-p_{l_i^*})\mu_{l_i^*}\Big(1-\Big(1-\mu_{l_{i+1}^*}\Big)\Big) \\
&\quad -(1-p_{l_{i+1}^*})\mu_{l_{i+1}^*}\big(1-\big(1-\mu_{l_i^*}\big)\big)\Big] \\
&= \left(\prod_{j=1}^{i-1}(1-\mu_{l_j^*})\mu_{l_i^*}\mu_{l_{i+1}^*}\right)\Big[(1-p_{l_i^*}) - (1-p_{l_{i+1}^*})\Big] \\
&= \left(\prod_{j=1}^{i-1}(1-\mu_{l_j^*})\mu_{l_i^*}\mu_{l_{i+1}^*}\right)(p_{l_{i+1}^*} - p_{l_i^*}).
\end{aligned}$$

Since probabilities are non-negative, the term $\left(\prod_{j=1}^{i-1}(1-\mu_{l_j^*})\mu_{l_i^*}\mu_{l_{i+1}^*}\right)$ is non-negative. From our assumption, $p_{l_i^*} > p_{l_{i+1}^*}$, which implies $p_{l_{i+1}^*} - p_{l_i^*} < 0$. Therefore, $r^* - \tilde{r} < 0$, which means $\tilde{r} > r^*$. Thus $\tilde{r} > r^*$, contradicting the optimality of $\mathcal{L}^*$. Therefore no such index i can exist, which proves the claimed ordering $p_{l_1^*} < p_{l_2^*} < \cdots < p_{l_K^*}$.

□

Proof of Lemma 4. We analyse the asymptotic behaviour of

$$\frac{\alpha_1 f(t) + \alpha_2 e^{-\alpha_3 f(t)}(t - f(t))}{\log(t)}$$

under different growth rates of $f(t)$.

Case 1: If $\liminf_{t\to\infty} \frac{f(t)}{\log(t)} = \infty$, then

$$\begin{aligned}
\liminf_{t\to\infty} \frac{\alpha_1 f(t) + \alpha_2 e^{-\alpha_3 f(t)}(t - f(t))}{\log(t)} &\geq \liminf_{t\to\infty} \frac{\alpha_1 f(t)}{\log(t)} \\
&= \infty.
\end{aligned}$$

Case 2: If $\liminf_{t\to\infty} \frac{f(t)}{\log(t)} = \ell < \infty$, then

$$\begin{aligned}
&\liminf_{t\to\infty} \frac{\alpha_1 f(t) + \alpha_2 e^{-\alpha_3 f(t)}(t - f(t))}{\log(t)} \\
&= \alpha_1 \ell + \alpha_2 \liminf_{t\to\infty} t^{-\alpha_3 f(t)/\log(t)}\Big(\tfrac{t}{\log(t)} - \ell\Big).
\end{aligned}$$

Since $\liminf_{t\to\infty} \frac{f(t)}{\log(t)} = \ell$, we obtain

$$\begin{aligned}
&\liminf_{t\to\infty} \frac{\alpha_1 f(t) + \alpha_2 e^{-\alpha_3 f(t)}(t - f(t))}{\log(t)} \\
&= \begin{cases} \alpha_1 \ell, & \alpha_3 \ell > 1, \\ \alpha_1 \ell + \alpha_2, & \alpha_3 \ell = 1, \\ \infty, & \alpha_3 \ell < 1. \end{cases}
\end{aligned}$$

Combining both cases, the quantity is always bounded below by a positive constant multiple of $\log(t)$. Hence,

$$\alpha_1 f(t) + \alpha_2 e^{-\alpha_3 f(t)}(t - f(t)) = \Omega(\log(t)).$$

□

Proof of Theorem 3. Let $N_i(t)$ denote the number of times arm i is placed first in the cascade up to time t. Let $\delta(T_s^{EC})$ represent the probability that Algorithm 1 chooses a

[1]AI tools are used throughout the paper for grammar and editing.

suboptimal ordering in the commit phase after the exploration horizon T_s^{EC}.

Exploration phase: Since the algorithm rotates arms uniformly, each arm appears first approximately T_s^{EC}/K times. The regret incurred during this phase comes from pulling suboptimal arms in the first position, and can be written as

$$\mathcal{R}^{EC}(T_s^{EC}) \ \geq \ \sum_{i=2}^{K} \mathbb{E}[N_i(T_s^{EC})]\tilde{\Delta}_{\min} = \frac{K-1}{K} T_s^{EC} \tilde{\Delta}_{\min}.$$

Commit phase: If a suboptimal ordering is chosen after exploration, then the regret in the commit phase is at least

$$\delta(T_s^{EC})(T - T_s^{EC} - 1)\tilde{\Delta}_{\min}.$$

During exploration, the algorithm collects at most T_s^{EC} effective samples of arms in the first position. Thus, the problem of finding the optimal cascade ordering contains, as a subproblem, best-arm identification with a fixed budget of T_s^{EC} samples. Therefore, any lower bound on the error probability of fixed-budget best-arm identification directly applies to our setting. By the result of [11], there exists a constant $\beta > 0$ such that

$$\delta(T_s^{EC}) \ \geq \ e^{-\beta T_s^{EC}}.$$

This inequality means that the probability of choosing a suboptimal ordering in the commit phase cannot be made arbitrarily small. Even after exploring each arm for T_s^{EC} rounds, there is still a nonzero chance that the algorithm misidentifies the best arm for the first position. Therefore, the total regret satisfies

$$\begin{aligned}\mathcal{R}^{EC}(T) &\geq \mathcal{R}^{EC}(T_s^{EC}) + \delta(T_s^{EC})(T - T_s^{EC} - 1)\tilde{\Delta}_{\min} \\ &\geq \frac{K-1}{K} T_s^{EC} \tilde{\Delta}_{\min} + e^{-\beta T_s^{EC}}(T - T_s^{EC} - 1)\tilde{\Delta}_{\min}.\end{aligned}$$

Finally, by Lemma 4, this simplifies to

$$\mathcal{R}^{EC}(T) = \Omega(\log T).$$

□

Proof of Lemma 1. The arms' ordering is not optimal if the LCBs are not ordered correctly. Which means $\exists i$ such that $L_i(t) > L_{i+1}(t)$, $t = T_s^{EC}$.

$$\mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*) \leq \sum_{i=1}^{K-1} \mathbb{P}(L_i(t) > L_{i+1}(t)).$$

Let $\mathcal{E}_t$ be the event that $|\hat{p}_i(t) - p_i| < \sqrt{\frac{2\log T}{S_i(t)}}$ for all i.

$$\begin{aligned}&\mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*) \\ &\leq \sum_{i=1}^{K-1} \mathbb{P}(L_i(t) > L_{i+1}(t), \mathcal{E}_t) + \mathbb{P}(\mathcal{E}_t^c) \\ &\overset{(a)}{\leq} \sum_{i=1}^{K-1} \mathbb{P}\left(\hat{p}_i(t) - \sqrt{\frac{2\log T}{S_i(t)}} > \hat{p}_{i+1}(t) - \sqrt{\frac{2\log T}{S_{i+1}(t)}}, \mathcal{E}_t\right) \\ &\quad + \frac{K}{T^4} \\ &\leq \sum_{i=1}^{K-1} \mathbb{P}\left(p_i > p_{i+1} - 2\sqrt{\frac{2\log T}{S_{i+1}(t)}}\right) + \frac{K}{T^4} \\ &= \sum_{i=1}^{K-1} \mathbb{P}\left(S_{i+1}(t) < \frac{8\log T}{\Delta_{i+1}^2}\right) + \frac{K}{T^4} \\ &= \sum_{i=2}^{K} \mathbb{P}\left(S_i(T_s^{EC}) < \frac{8\log T}{\Delta_i^2}\right) + \frac{K}{T^4} \\ &\leq \sum_{i=2}^{K} \mathbb{P}\left(\sum_{\tau=1}^{T_s^{EC}} X_i(\tau)\mathbb{1}\{l_1^{(\tau)} = i\} < \frac{8\log T}{\Delta_i^2}\right) + \frac{K}{T^4} \\ &\leq \sum_{i=2}^{K} \mathbb{P}\left(\sum_{\tau=1}^{T_s^{EC}} X_i(\tau)\mathbb{1}\{l_1^{(\tau)} = i\} < N\mu_i/2\right) + \frac{K}{T^4} \\ &\overset{(b)}{\leq} \sum_{i=2}^{K} e^{-N\mu_i/8} + \frac{K}{T^4} \\ &\leq \sum_{i=2}^{K} e^{-n_i\mu_i/8} + \frac{K}{T^4} \\ &\leq \sum_{i=2}^{K} \frac{1}{T^{2/\Delta_i^2}} + \frac{K}{T^4} \\ &\leq \frac{K}{T^2} + \frac{K^2}{T^4}.\end{aligned}$$

Where (a) is obtained using Hoeffding's inequality, (b) is obtained by using the fact that if $X \sim Ber(n, p)$ and $\mathbb{E}[X] = \mu$, then for $0 < \epsilon < 1$, $\mathbb{P}(X < (1-\epsilon)\mu) \leq e^{-\epsilon^2\mu/2}$. □

Lemma 8. *If $\mathcal{E}_t$ holds then in commit phase for Algorithm 2, $\mathcal{L}_t = \mathcal{L}^*$.*

Proof. Let us assume $\mathcal{E}_t$ holds and $\mathcal{L}_t \neq \mathcal{L}^*$ that is $\exists i$ in the ordering $\mathcal{L}_t$ such that $p_{l_i^t} < p_{l_{i-1}^t}$. Algorithm 2 is in commit phase.

$$\begin{aligned}\implies \hat{p}_{l_i^t}(t) - \epsilon_{l_i^t}(t) &> \hat{p}_{l_{i-1}^t}(t) + \epsilon_{l_{i-1}^t}(t) \\ \implies p_{l_i^t} &> p_{l_{i-1}^t},\end{aligned}$$

which is a contradiction. □

Proof of Lemma 2.

$$\begin{aligned}\mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*) &= \mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*, \mathcal{E}_t) + \mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*, \mathcal{E}_t^c) \\ &\overset{(a)}{=} \mathbb{P}(\mathcal{L}_t \neq \mathcal{L}^*, \mathcal{E}_t^c) \\ &\leq \mathbb{P}(\mathcal{E}_t^c) \\ &\leq \frac{K}{T^2},\end{aligned}$$

where (a) is obtained by using Lemma 8. □

Lemma 9. *For Algorithm 2, if $\mathcal{E}_t$ holds and $S_i(t) > \frac{16\log T}{\Delta_i'^2}$ for all i where, $\Delta_i' = \min\{p_i - p_{i-1}, p_{i+1} - p_i\}$, then algorithm is not in the active phase.*

Proof. Let us assume $\mathcal{E}_t$ holds and algorithm is in active phase then $\exists i, j, k$ such that

$$\hat{p}_i(t) + \epsilon_i(t) > \hat{p}_j(t) - \epsilon_j(t) \text{ or}$$

$$
\begin{aligned}
&\hat{p}_i(t) - \epsilon_i(t) < \hat{p}_k(t) + \epsilon_k(t) \\
\implies & p_i + 2\epsilon_i(t) > p_j - 2\epsilon_j(t) \text{ or } p_i - 2\epsilon_i(t) < p_k + 2\epsilon_k(t) \\
\implies & p_i + 2\epsilon_i(t) \geq \frac{p_j + p_i}{2} \text{ or } \frac{p_j + p_i}{2} \geq p_j - 2\epsilon_j(t) \text{ or} \\
& p_i - 2\epsilon_i(t) \leq \frac{p_i + p_k}{2} \text{ or } \frac{p_i + p_k}{2} \leq p_k + 2\epsilon_k(t) \\
\implies & 2\epsilon_i(t) \geq \frac{p_j - p_i}{2} \text{ or } 2\epsilon_j(t) \geq \frac{p_j - p_i}{2} \text{ or} \\
& 2\epsilon_i(t) \geq \frac{p_i - p_k}{2} \text{ or } 2\epsilon_k(t) \geq \frac{p_i - p_k}{2} \\
\implies & S_i(t) \leq \frac{16 \log T}{(p_j - p_i)^2} \text{ or } S_j(t) \leq \frac{16 \log T}{(p_j - p_i)^2} \text{ or} \\
& S_i(t) \leq \frac{16 \log T}{(p_i - p_k)^2} \text{ or } S_k(t) \leq \frac{16 \log T}{(p_i - p_k)^2}. \qquad (3)
\end{aligned}
$$

Therefore if $S_i(t) > \frac{16 \log T}{\Delta_i'^2}$ for all i where, $\Delta_i' = \min\{p_i - p_{i-1}, p_{i+1} - p_i\}$, then $\nexists j, k$ such that $\hat{p}_i(t) + \epsilon_i(t) > \hat{p}_j(t) - \epsilon_j(t)$ or $\hat{p}_i(t) - \epsilon_i(t) < \hat{p}_k(t) + \epsilon_k(t)$ for all i. This means no active arms exist, and Algorithm 2 is in the commit phase. □

Proof of Lemma 3. Let us consider an algorithm $\tilde{\mathcal{A}}$, where samples are updated only when it is the head of the cascade and everything is the same as Algorithm 2. Let T_s^{AE} and $\tilde{T_s^{\mathcal{A}}}$ be the time after which Algorithm 2 and $\tilde{\mathcal{A}}$ enter commit phase. Since samples are updated less often, $\tilde{A}$ takes more time to enter the commit phase; therefore, $T_s^{AE} < \tilde{T}_s^{\mathcal{A}}$.

Note that both algorithms modify the ordering similarly (round robin). Therefore, at any given time, $N_i(t), t \leq T_s^{AE}$ is the same for both algorithms. Since $N_i(t)$ is a monotone function $N_i(T_s^{AE}) \leq N_i(\tilde{T}_s^{\mathcal{A}})$.

If $\mathcal{E}_t$ holds $\forall t$, then Lemma 9 also holds and arm i is removed from active set if $S_i(t) > \frac{16 \log T}{\Delta_i'^2}$. Then N_i represents the upper bound on the number of times arm i is head of the cascade in active phase for $\tilde{\mathcal{A}}$. Thus $N_i(T_s^{AE}) \leq N_i(\tilde{T}_s^{\mathcal{A}}) \leq N_i$ when $\mathcal{E}_t$ occurs $\forall t$.

$$
\begin{aligned}
\sum_{t=1}^{T_s^{AE}} \mathbb{E}[R_t] &= \sum_{t=1}^{T_s^{AE}} \mathbb{E}[R_t|\mathcal{E}_t]\mathbb{P}(\mathcal{E}_t) + \mathbb{E}[R_t|\mathcal{E}_t^c]\mathbb{P}(\mathcal{E}_t^c) \\
&\leq \sum_{t=1}^{T_s^{AE}} \mathbb{E}[R_t|\mathcal{E}_t] + \tilde{\Delta}_{max}\mathbb{P}(\mathcal{E}_t^c) \\
&\leq \sum_{i=1}^{K} \mathbb{E}[N_i(T_s^{AE})]\tilde{\Delta}_i + \sum_{t=1}^{T_s^{AE}} \mathbb{P}(\mathcal{E}_t^c)\tilde{\Delta}_{max} \\
&\overset{(a)}{\leq} \sum_{i=1}^{K} \mathbb{E}[N_i]\tilde{\Delta}_i + \frac{K\tilde{\Delta}_{max}}{T},
\end{aligned}
$$

where (a) is obtained by using Lemma 2. □

Proof of Theorem 6. Let $\mathcal{E}_t$ be the event that $|\hat{p}_i(t) - p_i| < \sqrt{\frac{2 \log t}{S_i(t)}}$ for all i. By Hoeffding's inequality we have $\mathbb{P}(\mathcal{E}_t^c) \leq \frac{K}{t^2}$. Let $\mathcal{G}_t$ be the event that all arms are ordered correctly in time slot t. Thus $\mathcal{G}_t^c$ represents the event that there $\exists i \in [K]$ such that $L_i(t) > L_{i+1}(t)$. Therefore,

$$
\begin{aligned}
\mathbb{P}(\mathcal{G}_t^c, \mathcal{E}_t) &\leq \sum_{i=1}^{K-1} \mathbb{P}(L_i(t) > L_{i+1}(t), \mathcal{E}_t) \\
&= \sum_{i=1}^{K-1} \mathbb{P}(\hat{p}_i(t) - \epsilon_i(t) > \hat{p}_{i+1}(t) - \epsilon_{i+1}(t), \mathcal{E}_t) \\
&\leq \sum_{i=1}^{K-1} \mathbb{P}(p_i > p_{i+1} - 2\epsilon_{i+1}(t)) \\
&= \sum_{i=1}^{K-1} \mathbb{P}\left(\epsilon_{i+1}(t) > \frac{p_{i+1} - p_i}{2}\right) \\
&= \sum_{j=2}^{K} \mathbb{P}\left(S_{j,t} < \frac{8 \log t}{\Delta_j^2}\right).
\end{aligned}
$$

Let us define a new random variable

$$
Z_i(t) = \begin{cases} 1 & \text{if } X_i(t) = 1 \text{ and } X_j(t) = 0, \forall j \neq i, \\ 0 & \text{otherwise.} \end{cases}
$$

Note that arrivals are independent therefore $\{Z_i(t)\}_{t \geq 1}$ are also independent across time. Note that $Z_i(t) \sim Ber(\mu_i \prod_{j \neq i}(1 - \mu_j))$ and let us define $\bar{\mu}_i = \mathbb{E}[Z_i(t)] = \mu_i \prod_{j \neq i}(1 - \mu_j)$. The number of user feedback samples is lower bounded as follows $\sum_{n=1}^{t} Z_i(t) \leq S_i(t)$. Therefore

$$
\mathbb{P}(\mathcal{G}_t^c, \mathcal{E}_t) \leq \sum_{j=2}^{K} \mathbb{P}\left(Z_i(t) < \frac{8 \log t}{\Delta_j^2}\right).
$$

The regret is obtained only when the ordering is incorrect. Let $\mathbb{E}[R_t]$ be the regret incurred in time slot t.

$$
\begin{aligned}
\mathbb{E}[R_t] =& \mathbb{E}[R_t|\mathcal{G}_t]\mathbb{P}(\mathcal{G}_t) + \mathbb{E}[R_t|\mathcal{G}_t^c]\mathbb{P}(\mathcal{G}_t^c) \\
=& \mathbb{E}[R_t|\mathcal{G}_t^c]\mathbb{P}(\mathcal{G}_t^c) \\
=& \mathbb{E}[R_t|\mathcal{E}_t, \mathcal{G}_t^c]\mathbb{P}(\mathcal{G}_t^c, \mathcal{E}_t) + \mathbb{E}[R_t|\mathcal{E}_t^c, \mathcal{G}_t^c]\mathbb{P}(\mathcal{G}_t^c, \mathcal{E}_t^c) \\
\leq& \tilde{\Delta}_{max} \sum_{j=2}^{K} \mathbb{P}\left(Z_i(t) < \frac{8 \log t}{\Delta_j^2}\right) + \tilde{\Delta}_{max}\mathbb{P}(\mathcal{E}_t^c).
\end{aligned}
$$

The overall regret is bounded as follows

$$
\begin{aligned}
&\mathcal{R}^{LCB}(T) \\
&= \sum_{t=1}^{T} \mathbb{E}[R_t] \\
&\leq \sum_{t=1}^{T} \tilde{\Delta}_{max} \sum_{j=2}^{K} \mathbb{P}\left(Z_i(t) < \frac{8 \log t}{\Delta_j^2}\right) + \tilde{\Delta}_{max}\mathbb{P}(\mathcal{E}_t^c) \\
&\leq \tilde{\Delta}_{max} \sum_{j=2}^{K} \sum_{t=1}^{T} \mathbb{P}\left(Z_i(t) < \frac{8 \log t}{\Delta_j^2}\right) + \sum_{t=1}^{T} \tilde{\Delta}_{max} \frac{K}{t^2}.
\end{aligned}
$$

Note that for $t \geq T'$, where $T' = \frac{16}{\Delta_i^2 \bar{\mu}_i}$ we have $\frac{8 \log t}{\Delta_i^2} < \frac{\bar{\mu}_j t}{2}$, therefore

$$
\begin{aligned}
\sum_{t=1}^{T} \mathbb{P}\left(Z_i(t) \leq \frac{8 \log t}{\Delta_j^2}\right) \leq& \frac{16}{\Delta_j^2 \bar{\mu}_j} + \sum_{t=T'}^{T} \mathbb{P}\left(Z_i(t) < \frac{t\bar{\mu}_j}{2}\right) \\
\leq& \frac{16}{\Delta_j^2 \bar{\mu}_j} + \sum_{t=T'}^{T} e^{-t\bar{\mu}_j^2/2} \\
\leq& \frac{16}{\Delta_j^2 \bar{\mu}_j} + \frac{2}{\bar{\mu}_j^2}.
\end{aligned}
$$

Now, we bound the regret as follows

$$\begin{aligned}\mathcal{R}^{LCB}(T) \leq & \tilde{\Delta}_{max} \sum_{j=2}^{K} \left(\frac{16}{\Delta_j^2 \bar{\mu}_j} + \frac{2}{\bar{\mu}_j^2} \right) + \tilde{\Delta}_{max} \frac{K\pi^2}{6} \\ = & \mathrm{O}(1).\end{aligned}$$

□

Proof of Lemma 5. Let $\tau_0 = 0$ and $\tau_1, \tau_2, \cdots$ be the time slots in which sample for p_i is obtained i.e. $I_t = i$.

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), E_{i,k}^p(t)\}\right] \leq & \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{E_{i,k}^p(t), I_t = i\}\right] \\ \leq & \mathbb{E}\left[\sum_{k=0}^{T} \mathbb{1}\{E_{i,k}^p(\tau_k)\}\right] \\ \overset{(a)}{\leq} & 1 + \sum_{k=1}^{T} e^{-2k\epsilon^2} \\ \leq & 1 + \frac{1}{2\epsilon^2} \\ = & 1 + \frac{8}{\Delta_{i,k}^2},\end{aligned}$$

where (a) is obtained by using Hoeffdings' inequality. □

Lemma 10. *If $S_i(t)$ represents the number of samples observed by arm i till time t then we have,*

$$\sum_{t=1}^{T} \mathbb{P}\left(S_i(t) \leq \frac{16 \log t}{\Delta_{i,k}^2} \right) \leq \frac{32}{\Delta_{i,k}^2 \bar{\mu}_i} + \frac{2}{\bar{\mu}_i^2}.$$

Proof. Let us define a new random variable

$$Z_i(t) = \begin{cases} 1 & \text{if } X_i(t) = 1 \text{ and } X_j(t) = 0, \forall j \neq i, \\ 0 & \text{otherwise.} \end{cases}$$

Note that arrivals are independent therefore $\{Z_i(t)\}_{t \geq 1}$ are also independent across time. Note that $Z_i(t) \sim Ber(\mu_i \prod_{j \neq i}(1 - \mu_j))$ and let us define $\bar{\mu}_i = \mathbb{E}[Z_i(t)] = \mu_i \prod_{j \neq i}(1 - \mu_j)$. The number of user feedback samples is lower bounded as follows $\sum_{n=1}^{t} Z_i(n) \leq S_i(t)$. Therefore,

$$\sum_{t=1}^{T} \mathbb{P}\left(S_i(t) \leq \frac{16 \log t}{\Delta_{i,k}^2} \right) \leq \sum_{t=1}^{T} \mathbb{P}\left(\sum_{n=1}^{t} Z_i(n) \leq \frac{16 \log t}{\Delta_{i,k}^2} \right)$$

Note that for $t \geq T'$, where $T' = \frac{32}{\Delta_{i,k}^2 \bar{\mu}_i}$ we have $\frac{16 \log t}{\Delta_{i,k}^2} < \frac{\bar{\mu}_j t}{2}$, therefore

$$\begin{aligned}\sum_{t=1}^{T} \mathbb{P}\left(Z_i(t) \leq \frac{16 \log t}{\Delta_{i,k}^2} \right) \leq & \frac{32}{\Delta_{i,k}^2 \bar{\mu}_i} + \sum_{t=T'}^{T} \mathbb{P}\left(Z_i(t) < \frac{t \bar{\mu}_i}{2} \right) \\ \leq & \frac{32}{\Delta_{i,k}^2 \bar{\mu}_i} + \sum_{t=T'}^{T} e^{-t\bar{\mu}_i^2/2} \\ \leq & \frac{32}{\Delta_{i,k}^2 \bar{\mu}_i} + \frac{2}{\bar{\mu}_i^2}.\end{aligned}$$

□

Proof of Lemma 6. Let $L_i(t) = \frac{16 \log t}{\Delta_{i,k}^2}$ then,

$$\begin{aligned}& \sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), \bar{E}_{i,k}^p(t), E_{i,k}^{\theta}(t)\} \\ & = \sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), \bar{E}_{i,k}^p(t), E_{i,k}^{\theta}(t), S_i(t) \leq L_i(t)\} \\ & \quad + \sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), \bar{E}_{i,k}^p(t), E_{i,k}^{\theta}(t), S_i(t) > L_i(t)\}.\end{aligned}$$

Consider,

$$\begin{aligned}& \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), \bar{E}_{i,k}^p(t), E_{i,k}^{\theta}(t), S_i(t) \leq L_i(t)\}\right] \\ & \leq \sum_{t=1}^{T} \mathbb{P}(S_i(t) \leq L_i(t)) \\ & \overset{(b)}{\leq} \frac{32}{\Delta_{i,k}^2 \bar{\mu}_i} + \frac{2}{\bar{\mu}_i^2}, \end{aligned} \tag{4}$$

where (b) is obtained from Lemma 10. Now consider,

$$\begin{aligned}& \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), \bar{E}_{i,k}^p(t), E_{i,k}^{\theta}(t), S_i(t) > L_i(t)\}\right] \\ & \leq \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\{\hat{p}_i(t) \geq p_i - \epsilon, \theta_i(t) < p_k + \epsilon, S_i(t) > L_i(t)\}\right] \\ & \leq \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left(\theta_i(t) \leq \hat{p}_i(t) - \Delta_{i,k} + 2\epsilon, S_i(t) > \frac{16 \log t}{\Delta_{i,k}^2}\right)\right] \\ & = \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left(\theta_i(t) \leq \hat{p}_i(t) - \frac{\Delta_{i,k}}{2}, S_i(t) > \frac{16 \log t}{\Delta_{i,k}^2}\right)\right] \\ & \leq \sum_{t=1}^{T} \mathbb{P}\left(\theta_i(t) \leq \hat{p}_i(t) - \sqrt{\frac{4 \log t}{S_i(t)}}\right) \\ & \overset{(c)}{\leq} \sum_{t=1}^{T} \frac{1}{t^2}, \end{aligned} \tag{5}$$

where (c) is obtained by using Lemma 4 of [12]. By using (4) and (5), we get the result stated. □

Proof of Lemma 7. Let $\mathcal{F}_t$ represents the history till time t that is, $\mathcal{F}_t = (I_1, Y_{I_1}, I_2, Y_{I_2}, \cdots I_t, Y_{I_t})$ and define $\mathcal{F}_0 = \{\}$. Note that $\hat{p}_i(t)$, distribution of $\theta_i(t)$, and either $E_i^p(t)$ is true or not is determined by $\mathcal{F}_{t-1}$. Let F_{t-1} be the instantiation of $\mathcal{F}_{t-1}$ where $\bar{E}_i^p$ is true. We define $q_{k,t} \coloneqq \mathbb{P}(\theta_k(t) < p_k + \epsilon | \mathcal{F}_{t-1} = F_{t-1})$ and $\boldsymbol{\theta}_{-k}(t)$ represents the vector $\boldsymbol{\theta}(t)$ without $\theta_k(t)$. Let $\Theta_{i,k}(t)$ represents the collection of all possible values of $\boldsymbol{\theta}(t)$ for which $A_{i,k}(t)$ and $\bar{E}_{i,k}^{\theta}(t)$ holds. Let $\Theta_{i,-k}(t) \coloneqq \{\boldsymbol{\theta}_{-k}(t) : \boldsymbol{\theta}(t) \in \Theta_{i,k}(t)\}$. Let $M_i = \{j : l_t^{-1}(j) > l_t^{-1}(i)\}$ represents the arms after arm i in cascade. Then,

$$\begin{aligned}& \mathbb{E}\left[\mathbb{1}\{A_{i,k}(t), \bar{E}_{i,k}^p(t), \bar{E}_{i,k}^{\theta}(t)\}\right] \\ & = \mathbb{E}\left[\mathbb{1}\{A_{i,k}(t), \bar{E}_{i,k}^p(t), \bar{E}_{i,k}^{\theta}(t)\} | \mathcal{F}_{t-1} = F_{t-1}\right] \\ & \leq \mathbb{P}(\theta_j(t) \geq p_k + \epsilon, \forall j \in M_i, I_t = i\end{aligned}$$

$$\begin{aligned}
&\boldsymbol{\theta}_{-k}(t) \in \Theta_{i,-k}(t) | \mathcal{F}_{t-1} = F_{t-1}) \\
=&\mathbb{P}(\theta_k(t) \geq p_k + \epsilon | \mathcal{F}_{t-1} = F_{t-1}). \\
&\mathbb{E}\left[\prod_{j<l_t^{-1}(i)} (1 - X_{l_j}(t)) | \boldsymbol{\theta}_t \right]. \\
&\mathbb{P}(\theta_j(t) \geq p_k + \epsilon \forall j \in M_i/\{k\}, \\
&\quad \boldsymbol{\theta}_{-k}(t) \in \Theta_{i,-k}(t) | \mathcal{F}_{t-1} = F_{t-1}) \\
=&(1 - q_{k,t}).\mathbb{E}\left[\prod_{j<l_i^{-1}(t)} (1 - X_{l_j}(t)) | \boldsymbol{\theta}_{-k}(t) \right]. \\
&\mathbb{P}(\theta_j(t) \geq p_k + \epsilon \forall j \in M_i/\{k\}, \\
&\quad \boldsymbol{\theta}_{-k}(t) \in \Theta_{i,-k}(t) | \mathcal{F}_{t-1} = F_{t-1}). \qquad (6)
\end{aligned}$$

Consider the instance where $\theta_k(t)$ is modified such that $\theta_k(t) < \theta_i(t)$ and $\boldsymbol{\theta}_{-k}(t)$ is not modified, then

$$\begin{aligned}
&\mathbb{E}\Big[\mathbb{1}\{I_t = k, l_t^{-1}(k) < l_t^{-1}(i), \bar{E}^{\theta}_{i,k}(t), \\
&\quad \boldsymbol{\theta}_{-k}(t) \in \Theta_{i,-k}(t) | \mathcal{F}_{t-1} = F_{t-1}\} \Big] \\
>&\mathbb{P}(\theta_k(t) < p_k + \epsilon \leq \theta_j(t), \forall j \in M_i/\{k\}, I_t = k, \\
&\quad \boldsymbol{\theta}_{-k}(t) \in \Theta_{i,-k}(t) | \mathcal{F}_{t-1} = F_{t-1}). \\
=&\mathbb{P}(\theta_k(t) < p_k + \epsilon | \mathcal{F}_{t-1} = F_{t-1}). \\
&\mathbb{E}\left[\prod_{j<l_k^{-1}(t)} (1 - X_{l_j}) | \boldsymbol{\theta}_{-k}(t) \right]. \\
&\mathbb{P}(\theta_j(t) \geq p_k + \epsilon, \forall j \in M_i/\{k\}, \\
&\quad \boldsymbol{\theta}_{-k}(t) \in \Theta_{i,-k}(t) | \mathcal{F}_{t-1} = F_{t-1}) \\
\geq& q_{k,t}.\mathbb{E}\left[\prod_{j<l_i^{-1}(t)-1} (1 - X_{l_j}) | \boldsymbol{\theta}_{-k}(t) \right]. \\
&\mathbb{P}(\theta_j(t) \geq p_k + \epsilon, \forall j \in M_i/\{k\}, \\
&\quad \boldsymbol{\theta}_{-k}(t) \in \Theta_{i,-k}(t) | \mathcal{F}_{t-1} = F_{t-1}). \qquad (7)
\end{aligned}$$

From (6), (7) we get

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), \bar{E}^{p}_{i,k}(t), \bar{E}^{\theta}_{i,k}(t)\} \right] \\
&\leq \sum_{t=1}^{T} \mathbb{E}\Bigg[\frac{1 - q_{k,t}}{q_{k,t}} \mathbb{1}\{I_t = k, l_t^{-1}(k) < l_t^{-1}(i), \\
&\qquad \bar{E}^{\theta}_{i,k}(t), \boldsymbol{\theta}_{-k}(t) \in \Theta_{i,-k}(t) | \mathcal{F}_{t-1} = F_{t-1}\} \Bigg].
\end{aligned}$$

Let $\tau_{k,s}$ be the time slot in which arm k is chosen for s-th time, then we have

$$\begin{aligned}
&\sum_{t=1}^{T} \mathbb{E}\Bigg[\frac{1 - q_{k,t}}{q_{k,t}} \mathbb{1}\{I_t = k, l_t^{-1}(k) < l_t^{-1}(i), \\
&\qquad \bar{E}^{\theta}_{i,k}(t), \boldsymbol{\theta}_{-k}(t) \in \Theta_{i,-k}(t) | \mathcal{F}_{t-1} = F_{t-1}\} \Bigg] \\
&\leq \sum_{s=1}^{T} \mathbb{E}\left[\frac{1 - q_{k,\tau_{k,s}}}{q_{k,\tau_{k,s}}} \right].
\end{aligned}$$

Lemma (Implied by Lemma 2.9 [13])**.** *If $\tau_{k,s}$ denote the time step at which s-th sample of arm k is observed then we have*

$$\begin{aligned}
&\mathbb{E}\left[\frac{1 - q_{k,\tau_{k,s}}}{q_{k,\tau_{k,s}}} \right] \\
&\leq \begin{cases} \frac{3}{\epsilon} & \text{for } s < \frac{8}{\epsilon} \\ \Theta\left(e^{-\epsilon^2 s/2} + \frac{1}{(s+1)\epsilon^2} e^{-sD_k} + \frac{1}{e^{\epsilon^2 s/4} - 1} \right) & \text{else,} \end{cases}
\end{aligned}$$

where $D_k = KL(p_k, p_k + \epsilon)$.

Now, we follow a similar analysis from Lemma 3.3 of [14] and improve the bound stated in [14].

$$\begin{aligned}
&\mathbb{E}\left[\frac{1 - q_{k,\tau_{k,s}}}{q_{k,\tau_{k,s}}} \right] \\
&\leq \sum_{0 \leq s \leq 8/\epsilon} \frac{3}{\epsilon} + c_1 . \sum_{8/\epsilon \leq s \leq T-1} e^{-\epsilon^2 s/2} + \frac{1}{(s+1)\epsilon^2} e^{-sD_k} \\
&\quad + \frac{1}{e^{\epsilon^2 s/4} - 1} \\
&\overset{(a)}{\leq} \frac{24}{\epsilon^2} + c_1 . \sum_{s=1}^{\infty} e^{-\epsilon^2 s/2} + c_1 . \int_{8/\epsilon}^{T-1} \frac{1}{(s+1)\epsilon^2} e^{-2\epsilon^2 s} ds \\
&\quad + c_1 \int_{8/\epsilon}^{T-1} \frac{1}{e^{\epsilon^2 s/4} - 1} ds \\
&\overset{(b)}{\leq} \frac{24}{\epsilon^2} + \frac{2c_1}{\epsilon^2} + \frac{c_1 e^{\epsilon^2/2}}{\epsilon^2} \left(\frac{1}{16\epsilon} \mathbb{1}\{16\epsilon < 1\} + \frac{1}{e} \right) \\
&\quad + c_1 \int_{8/\epsilon}^{T-1} \frac{1}{e^{\epsilon^2 s/4} - 1} ds,
\end{aligned}$$

where c_1 is a constant. (a) follows from the fact $KL(p, q) \geq \frac{|p-q|^2}{2}$. (b) is obtained using the result from [14] and fact that $\sum_{t=1}^{\infty} e^{-at} \leq \frac{1}{a}, a > 0$. Consider

$$\begin{aligned}
\int_{8/\epsilon}^{T-1} \frac{1}{e^{\epsilon^2 s/4} - 1} ds &\leq \int_{8/\epsilon}^{\infty} \frac{1}{e^{\epsilon^2 s/4} - 1} ds \\
&= \frac{8}{\epsilon} - \frac{4 \log(e^{2\epsilon} - 1)}{\epsilon^2}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{A_{i,k}(t), \bar{E}^{p}_{i,k}(t), \bar{E}^{\theta}_{i,k}(t)\} \right] \\
&\leq \frac{24}{\epsilon^2} + \frac{2c_1}{\epsilon^2} + \frac{c_1 e^{\epsilon^2/2}}{\epsilon^2} \left(\frac{1}{16\epsilon} \mathbb{1}\{16\epsilon < 1\} + \frac{1}{e} \right) \\
&\quad + \frac{8c_1}{\epsilon} - \frac{4c_1 \log(e^{2\epsilon} - 1)}{\epsilon^2}.
\end{aligned}$$

□