Transfer in Sequential Multi-armed Bandits via Reward Samples

NR Rahul and Vaibhav Katewa

***Abstract*— We consider a sequential stochastic multi-armed bandit problem where the agent interacts with the bandit over multiple episodes. The reward distribution of the arms remains constant throughout an episode but can change over different episodes. We propose an algorithm based on UCB to transfer the reward samples from the previous episodes and improve the cumulative regret performance over all the episodes. We provide regret analysis and empirical results for our algorithm, which show significant improvement over the standard UCB algorithm without transfer.**

I. INTRODUCTION

The Multi-armed Bandit (MAB) problem [1], [2], [3] is a popular sequential decision-making problem where an agent interacts with the environment by taking actions at every time step and, in return, gets a random reward. The goal of the agent is to maximize the average reward received. Recently, there has been a lot of interest in applying the MAB problem in the context of online advertisements and recommender systems[4], [5]. One of the problems highlighted in [5] is the user cold start problem, which is the inability of a recommender system to make a good recommendation for a new user in the absence of any prior information. In this scenario, it is useful to transfer knowledge from other related users in order to make better initial recommendations to the new user. In the context of a MAB problem, transfer learning uses knowledge from one bandit problem in order to improve the performance of another related bandit problem [6], [7]. In particular, it helps to accelerate learning and make better decisions quickly.

In this paper, we consider a sequential stochastic MAB problem where the agent interacts with the environment sequentially in episodes (similar to [6]), where different episodes are synonymous with different tasks or different bandit problems. The reward distributions of the arms remain constant throughout the episode but change over different episodes. This scenario is useful, for instance, in recommender systems where the reward distributions of recommended items change in order to capture the changing user preferences over time. The goal is to leverage the knowledge from previous episodes in order to improve the performance in the current episode, thereby leading to an overall performance improvement. Towards this, we use reward samples from previous episodes to make decisions in the current episode. Our algorithm is based on the UCB algorithm for bandits [8].

Related Work: In [6], the authors consider sequential transfer when encountering a fixed number of tasks. In [9], the sequential transfer of related tasks in linear bandits has been studied. The authors have captured the relatedness of tasks by the L_2 distance of the parameter vectors of reward functions. However, we consider the sequential transfer in stochastic multi-armed bandits, similar to [6], for related tasks captured by the L_∞ distance of the means of the reward functions. This notion is similar to [10], which studies the transfer across multiple bandit tasks simultaneously, rather than sequentially. Furthermore, paper [11] considers representational transfer in sequential linear bandits. The other related frameworks include meta-learning [12],[13], where the algorithm learns to adapt to a new task after learning from a few tasks drawn from the same task distribution.

The main contributions of the paper are:

(i) We develop an algorithm based on UCB to transfer knowledge using the reward samples from the previous episodes in a sequential stochastic MAB setting. Our algorithm has a better performance compared to UCB with no transfer.

(ii) We provide the regret analysis for the proposed algorithm, and our regret upper bound explicitly captures the performance improvement due to transfer.

(iii) We show via numerical simulations that our algorithm is able to effectively transfer knowledge from previous episodes.

Notations: $\mathbb{1}\{E\}$ denotes the indicator function whose value is 1 if the event (condition) E is true, and 0 otherwise. Similarly, for n events E_1, E_2, $\cdots$, E_n, where $n \in \mathbb{N}$, we define $\mathbb{1}\{E_1, E_2, \cdots, E_n\}$ as the indicator function whose value is 1 if all the events are true, and 0 otherwise. Further, let $\emptyset$ denote the null set.

II. PRELIMINARIES AND PROBLEM STATEMENT

We consider the Multi-Armed Bandit problem with K arms and J episodes. The length of each episode is n. Define $[K] \triangleq \{1, 2, \cdots, K\}$ and $[J] \triangleq \{1, 2, \cdots, J\}$. At any given integer time $t > 0$, one among the K arms is pulled, and a random reward is received. Let $I_t \in [K]$ and r_{I_t}, denote the arm pulled at time t and the corresponding random reward, respectively. We assume that $r_{I_t} \in [0, 1]$ and the rewards are independent across time and across all arms. In any given episode, the distributions of the arms do not change. However, they are allowed to be different over different episodes.

Let μ_k^j be the mean reward of arm k in episode j. Let $\mu^j \triangleq [\mu_1^j, \mu_2^j, \cdots, \mu_K^j]^T$ denote the vector containing the

NR Rahul is with the Department of Electrical Communication Engineering (ECE) at the Indian Institute of Science, Bengaluru, India. Email: rahulnr@iisc.ac.in.

Vaibhav Katewa is with the Robert Bosch Center for Cyber-Physical Systems and the Department of ECE at the Indian Institute of Science, Bengaluru, India. Email:vkatewa@iisc.ac.in

This work is supported by SERB Grants SRG/2021/000292 and MTR/2022/000522.

mean rewards of all arms for episode j. Further, let $k_*^j \in \mathcal{A}^j \triangleq \arg\max_{k\in[K]}\{\mu_k^j\}$ and $\mu_*^j = \max_{k\in[K]}\{\mu_k^j\}$ denote an optimal arm[1] in episode j and it's mean reward, respectively. Define $\Delta_k^j = \mu_*^j - \mu_k^j > 0$ as the sub-optimality gap of arm $k \notin \mathcal{A}^j$ in episode j. Note that the mean rewards of the arms are unknown.

We assume that the episodes in the MAB problem are related in the sense that the mean rewards of the arms across episodes do not change considerably. We capture this by the following assumption.

Assumption 1. *We assume that $||\mu^{j_1} - \mu^{j_2}||_\infty \leq \epsilon$ for any $j_1, j_2 \in [J]$, where the parameter $0 < \epsilon < 1$ is assumed to be known.*

This assumption implies that for each arm, the mean rewards across all episodes do not differ by more than ϵ. In applications like online advertising and recommender systems, the user preferences change over time only gradually, and therefore, the parameter ϵ can be used to capture this behaviour.

Let $N_k^j(t)$ denote the number of pulls of arm k in the time interval $[(j-1)n+1, t]$. Thus, $N_k^j(t)$ counts the number of times arm k is pulled from the beginning of episode j until time t. Note that for episode j, the allowable values of t in $N_k^j(t)$ are $[(j-1)n+1, nj]$. Further, let $S_k(t)$ denote the number of pulls of arm k in the time interval $[1, t]$. Thus, $S_k(t)$ counts the number of times arm k is pulled from the beginning of episode 1 until time t. For example, if $n = 5$ and $j = 2$, then $N_k^2(8)$ counts the number of times arm k is pulled in time instants $6, 7$, and 8. Further, $S_k(8)$ counts the number of times arm k is pulled in the interval $[1, 8]$.

The goal of the agent is to decide which arm to pull (what should be the value of I_t) at any given time t based on the information $\{r_{I_1}, r_{I_2}, \cdots, r_{I_{t-1}}\}$ in order to maximize the average reward over all episodes. This is captured by the pseudo-regret R_J of the MAB problem over J episodes:

$$\begin{aligned} R_J &= \sum_{j=1}^{J} \mathbb{E}\left[\sum_{t=(j-1)n+1}^{jn} (r_{k_*^j} - r_{I_t})\right] \\ &= \sum_{j=1}^{J}\left(n\mu_*^j - \mathbb{E}\left[\sum_{t=(j-1)n+1}^{jn} \mu_{I_t}^j\right]\right) \\ &= \sum_{j=1}^{J}\sum_{k=1}^{K} \Delta_k^j \mathbb{E}[N_k^j(jn)], \end{aligned} \tag{1}$$

where the last equality follows since $\sum_{k=1}^K N_k^j(jn) = n$ for any $j \in [J]$. Thus, the goal is to make decisions $\{I_t : 1 \leq t \leq nJ\}$ to minimize the regret in (1).

In this paper, we exploit the relation among the mean rewards of arms in different episodes (c.f. Assumption 1) in order to minimize the regret R_J. This is achieved by reusing (transferring) reward samples from previous episodes to make decisions in the current episode. We describe the approach and the proposed algorithm in detail in the next section.

[1]There may be more than one optimal arms which have equal maximum mean rewards.

III. ALL SAMPLE TRANSFER UCB (AST-UCB)

Our approach of reusing samples from previous episodes builds on the standard UCB algorithm for bandits. In this section, we first describe the UCB algorithm and then our proposed algorithm, which we call All Sample Transfer UCB (AST-UCB).

A. UCB Algorithm [8]

Intuitively, the arm-pulling decisions should be made on the reward samples obtained from each arm. Since the mean rewards of the arms are unknown, the UCB algorithm computes their sample-average estimates and the corresponding confidence intervals. Then, based on the principle of optimism in the face of uncertainty, the upper (maximum) value in the confidence interval of each arm is treated as the optimistic mean reward of that arm. Then, the arm with the highest optimistic mean reward is pulled.

As time progresses and more reward samples are received, the estimates become better, and the confidence intervals become smaller. Thus, the upper value in the confidence interval approaches the true mean. Eventually, the optimistic mean reward of the optimal arm becomes larger than all other sub-optimal arms, and thereafter, only the optimal arm is pulled.

The standard UCB algorithm is used when the arm distributions are assumed to be the same at all times. However, in our setting, the distributions change over episodes. Therefore, one approach would be to implement the UCB algorithm separately in each episode by using only the samples of that particular episode. In other words, the UCB algorithm is restarted at the beginning of every episode and it uses only the reward samples received during the current episode. We call this approach the No Transfer UCB (NT-UCB) algorithm. Next, we explain the NT-UCB algorithm for episode j.

Let $\hat{\mu}_{1k}^j(t)$ denote the sample-average estimate of the mean reward of arm k at time t, and is computed as:

$$\hat{\mu}_{1k}^j(t) = \frac{\sum_{\tau=(j-1)n+1}^{t} r_{I_\tau} \mathbb{1}\{I_\tau = k\}}{N_k^j(t)}, \tag{2}$$

where $N_k^j(t)$ denotes the number of times arm k is pulled until time t since the beginning of episode j. Next, we compute the optimistic mean reward corresponding to $\hat{\mu}_{1k}^j(t)$. For this, we require the following result.

Lemma 1. *Let $\alpha > 2$. For a given $N_k^j(t)$, with probability at least $1 - \frac{2}{(t-(j-1)n)^\alpha}$, the following equation is satisfied*

$$|\hat{\mu}_{1k}^j(t) - \mu_k^j| \leq p_{1k}^j(t) \triangleq \sqrt{\frac{\alpha \log\left(t-(j-1)n\right)}{2N_k^j(t)}}. \tag{3}$$

Proof. Follows from Hoeffding's inequality [14]. □

Using Lemma 1, we form a confidence interval for mean reward μ_k^j using the estimate $\hat{\mu}_{1k}^j(t)$ at time t in episode j as

$$D_1^j(t) = [\hat{\mu}_{1k}^j(t) - p_{1k}^j(t), \hat{\mu}_{1k}^j(t) + p_{1k}^j(t)].$$

Next, the NT-UCB algorithm pulls the arm with maximum optimistic reward:

$$I_t = \arg\max_{k\in[K]} \left\{ \hat{\mu}_{1k}^j(t-1) + p_{1k}^j(t-1) \right\}.$$

The above steps are repeated until the end of episode j. Next, we provide an upper bound on the pseudo-regret of the NT-UCB algorithm.

Lemma 2. *Let $\alpha > 2$. The pseudo-regret of NT-UCB satisfies*

$$R_J \leq \sum_{k=1}^{K} \left[2\alpha \log(n) \Bigg(\sum_{\substack{j=1 \\ \Delta_k^j > 0}}^{J} \frac{1}{\Delta_k^j} \Bigg) + \frac{\alpha}{\alpha - 2} \Bigg(\sum_{j=1}^{J} \Delta_k^j \Bigg) \right]. \tag{4}$$

Proof. Follows from the per episode regret bound of the standard UCB algorithm [1]. □

B. AST-UCB Algorithm

For any particular episode, the NT-UCB algorithm mentioned above uses samples only in that episode to compute the estimates. However, as per Assumption 1, the mean rewards across the episodes are related, and therefore, reward samples in previous episodes carry information about the mean reward in the current episode. In order to capture this information, we construct an auxiliary estimate (in addition to the UCB estimate) that uses the reward samples from the beginning of the first episode. Then, we combine these two estimates to make the decisions. Next, we describe this approach for episode j.

Let $\hat{\mu}_{2k}(t)$ denote the auxiliary sample-average estimate of the mean reward of arm k at time t, computed as:

$$\hat{\mu}_{2k}(t) = \frac{\sum_{\tau=1}^{t} r_{I_\tau} \mathbb{1}\{I_\tau = k\}}{S_k(t)}, \tag{5}$$

where $S_k(t)$ denotes the number of times arm k is pulled until time t since the beginning of episode 1. Note that estimate $\hat{\mu}_{2k}(t)$ captures the information of reward samples of arm k from all previous episodes [2]. Next, we compute the optimistic mean reward corresponding to $\hat{\mu}_{2k}(t)$. For this, we require the following result.

Lemma 3. *Let $\alpha > 2$. For a given $N_k^j(t)$ and $S_k(t)$, with probability at least $1 - \frac{2}{(t(t-(j-1)n))^\alpha}$, the following equation is satisfied*

$$|\hat{\mu}_{2k}(t) - \mu_k^j| \leq p_{2k}^j(t) \triangleq \sqrt{\frac{\alpha \log\left(t(t-(j-1)n)\right)}{2S_k(t)}} + U_k^j(t)\epsilon, \tag{6}$$

$$\text{where } U_k^j(t) = \frac{S_k(t) - N_k^j(t)}{S_k(t)}.$$

Proof. The rewards are independent random variables with support $[0, 1]$. Using McDiarmid's inequality[15] for estimate $\hat{\mu}_{2k}(t)$, we get

$$\Pr\{|\hat{\mu}_{2k}(t) - \mathbb{E}[\hat{\mu}_{2k}(t)]| \geq \delta\} \leq 2\exp(-2S_k(t)\delta^2).$$

Setting $\delta = \sqrt{\frac{\alpha \log(t(t-(j-1)n))}{2S_k(t)}}$ for $S_k(t) \geq 1$, we get

$$\Pr\left\{ |\hat{\mu}_{2k}(t) - \mathbb{E}[\hat{\mu}_{2k}(t)]| \geq \sqrt{\frac{\alpha \log(t(t-(j-1)n))}{2S_k(t)}} \right\} \leq \frac{2}{(t(t-(j-1)n))^\alpha}.$$

Hence, with probability at least $1 - \frac{2}{(t(t-(j-1)n))^\alpha}$, the following holds

$$|\hat{\mu}_{2k}(t) - \mathbb{E}[\hat{\mu}_{2k}(t)]| \leq \sqrt{\frac{\alpha \log(t(t-(j-1)n))}{2S_k(t)}}. \tag{7}$$

Next, we bound $\mathbb{E}[\hat{\mu}_{2k}(t)]$ for $S_k(t) \geq 1$, $t \in [(j-1)n + 1, jn]$:

$$\begin{aligned}
\mathbb{E}[\hat{\mu}_{2k}(t)] &= \frac{\sum_{l=1}^{j-1} N_k^l(ln)\mu_k^l + N_k^j(t)\mu_k^j}{S_k(t)} \\
&= \mu_k^j + \frac{\sum_{l=1}^{j-1} N_k^l(ln)(\mu_k^l - \mu_k^j)}{S_k(t)} \\
&\leq \mu_k^j + \frac{(S_k(t) - N_k^j(t))\epsilon}{S_k(t)} \\
&= \mu_k^j + U_k^j(t)\epsilon,
\end{aligned} \tag{8}$$

where the inequality follows from $\mu_k^l - \mu_k^j \leq \epsilon$ (Asssumption 1). Similarly, using $\mu_k^l - \mu_k^j \geq -\epsilon$ (Asssumption 1), we get

$$\mathbb{E}[\hat{\mu}_{2k}(t)] \geq \mu_k^j - U_k^j(t)\epsilon. \tag{9}$$

Conditions (8) and (9) yield $|\mathbb{E}[\hat{\mu}_{2k}(t)]| \leq \mu_k^j + U_k^j(t)\epsilon$. Using this in (7), we get the result in (6). □

Using Lemma 3, we form a confidence interval for mean reward μ_k^j using the estimate $\hat{\mu}_{2k}(t)$ at time step t in episode j as

$$D_2^j(t) = [\hat{\mu}_{2k}(t) - p_{2k}^j(t), \hat{\mu}_{2k}(t) + p_{2k}^j(t)].$$

Next, we present two key steps of the AST-UCB algorithm.

(i) Combine the optimistic rewards of the two estimates $\hat{\mu}_{1k}^j(t)$ and $\hat{\mu}_{2k}(t)$ given in (2) and (5) as:

$$q_k^j(t) = \min\{\hat{\mu}_{1k}^j(t) + p_{1k}^j(t), \hat{\mu}_{2k}(t) + p_{2k}^j(t)\}. \tag{10}$$

[2] An alternate strategy would be to construct the auxiliary estimate from a fixed number of previous episodes. However, our strategy is better since the confidence interval corresponding to estimate (5) is always better than this alternate strategy.

(ii) Pull arm

$$I_t = \arg\max_{k\in[K]}\{q_k^j(t-1)\}.$$

The above steps are repeated until the end of episode j. All the steps of AST-UCB are given below in Algorithm 1.

Algorithm 1 AST-UCB

Require: Episode length n, Number of episodes J, Parameters α, ϵ and Number of arms K

for episode $j = 1, 2, ..., J$ **do**
  for $t = (j-1)n+1, \cdots, (j-1)n+K$ **do**
    $I_t = t-(j-1)n$ (Pull each arm once)
  end for
  for $t = (j-1)n+K+1, \cdots, jn$ **do**
    compute $\hat{\mu}_{1k}^j(t-1)$, $p_{1k}^j(t-1)$ using (2), (3)
    compute $\hat{\mu}_{2k}(t-1)$, $p_{2k}^j(t-1)$ using (5), (6)
    compute optimistic reward $q_k^j(t-1)$ using (10)
    select arm $I_t = \arg\max_{k\in[K]}\{q_k^j(t-1)\}$
    update number of pulls $N_k^j(t)$ and $S_k(t)$
  end for
end for

Next, we explain the motivation for Step (i). We combine the confidence intervals $D_{1k}^j(t)$ and $D_{2k}^j(t)$ by taking their intersection to get a better confidence interval. Note that by taking the intersection, the new confidence interval $D_{1k}^j(t) \cap D_{2k}^j(t)$ is always smaller than the original two confidence intervals, as illustrated in Figure 1. This smaller interval results in a better estimate of μ_k^j. We then pick the optimistic reward in the new confidence interval[3]. Further, Step (ii) is similar to the UCB algorithm, where we pull the arm with the maximum optimistic reward. The next result presents a bound on the probability of μ_k^j lying in the new confidence interval (the new confidence interval being non-empty).

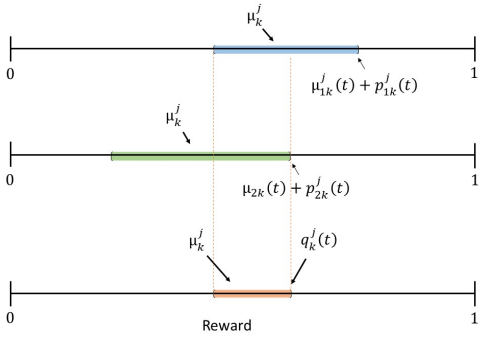


Fig. 1: The blue and green intervals represent confidence intervals $D_{1k}^j(t)$ and $D_{2k}^j(t)$ for mean μ_k^j, respectively. The orange interval is the intersection of the two intervals, which is clearly smaller (and hence better). The optimistic reward of the orange interval is given by $q_k^j(t)$.

[3]Note that the Step (i) is valid even when $D_{1k}^j(t)$ and $D_{2k}^j(t)$ do not intersect.

Lemma 4. *For episode j, time $t \in [(j-1)n+1, jn]$ and arm k, with probability at least $1-\big(\frac{2}{(t-(j-1)n)^\alpha}+\frac{2}{(t(t-(j-1)n))^\alpha}\big)$, the following equations are satisfied*

$$(i) \quad \mu_k^j \in D_{1k}^j(t) \cap D_{2k}^j(t). \tag{11}$$

$$(ii) \quad D_{1k}^j(t) \cap D_{2k}^j(t) = \emptyset. \tag{12}$$

Proof. Define events $\mathcal{E}_1 = \{\mu_k^j \notin D_{1k}^j(t)\}$ and $\mathcal{E}_2 = \{\mu_k^j \notin D_{2k}^j(t)\}$. Then we have

$$\begin{aligned}\Pr\{\mu_k^j \notin D_{1k}^j(t) \cap D_{2k}^j(t)\} &= \Pr\{\mathcal{E}_1 \cup \mathcal{E}_2\} \\ &\le \Pr\{\mathcal{E}_1\} + \Pr\{\mathcal{E}_2\} \\ &\le \frac{2}{(t-(j-1)n)^\alpha} + \frac{2}{(t(t-(j-1)n))^\alpha},\end{aligned}$$

where the last inequality follows from Lemmas 1 and 3. Hence, condition (i) in the lemma follows. Same arguments are valid for condition (ii) as well. □

Note that although the new confidence interval is smaller, Lemma 4 shows that the probability bound of the mean reward belonging to this new interval has reduced as compared to that in (3) or (6). However, we show in Theorem 1 that the negative effect of the reduction of the probability is not significant, and the smaller interval leads to an overall reduction in the regret.

IV. REGRET ANALYSIS

In this section, we derive the regret of the AST-UCB algorithm and then provide the analysis of the result.

Theorem 1. *Let $\Delta_k^{max} \triangleq \max_{j\in[J]}\{\Delta_k^j\}$ and $\Delta_k^{min} \triangleq \min_{j\in[J],\Delta_k^j>2\epsilon}\{\Delta_k^j\}$. The pseudo-regret of AST-UCB with $\alpha > 2$ satisfies*

$$R_J \le \sum_{k=1}^{K}\Delta_k^{max}\Bigg[\min\Bigg\{\sum_{\substack{j=1\\ \Delta_k^j>2\epsilon}}^{J}\frac{2\alpha\log(n)}{(\Delta_k^j)^2}, \frac{2\alpha\log(Jn^2)}{(\Delta_k^{\min}-2\epsilon)^2}\Bigg\} + \sum_{\substack{j=1\\ \Delta_k^j\le 2\epsilon}}^{J}\frac{2\alpha\log(n)}{(\Delta_k^j)^2} + J\Big(\frac{\alpha}{\alpha-2}+\frac{2}{2\alpha-3}\Big)\Bigg]. \tag{13}$$

Proof. Refer to the appendix. □

Next, we compare the regret bounds of our algorithm (13) and NT-UCB (4), and highlight the benefit of transfer. The transfer happens due to the first and second terms in (13). Hence, we compare the first and second terms of (13) with the first term of (4). To this end, we define the following terms that capture the dependence on J:

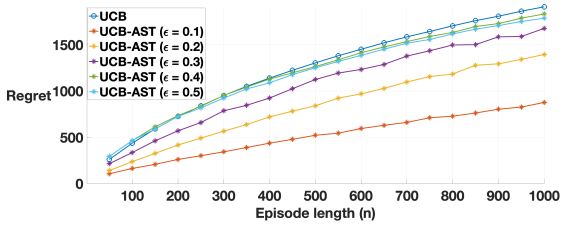
$$A_k^J = \sum_{\substack{j=1\\ \Delta_k^j>2\epsilon}}^{J}\frac{\Delta_k^{\max}\log(n)}{(\Delta_k^j)^2}, \quad B_k^J = \frac{\Delta_k^{\max}\log(Jn^2)}{(\Delta_k^{\min}-2\epsilon)^2},$$

$$C_k^J = \sum_{\substack{j=1\\ \Delta_k^j\le 2\epsilon}}^{J}\frac{\Delta_k^{\max}\log(n)}{(\Delta_k^j)^2}, \quad D_k^J = \sum_{\substack{j=1\\ \Delta_k^j>0}}^{J}\frac{\log(n)}{\Delta_k^j}.$$

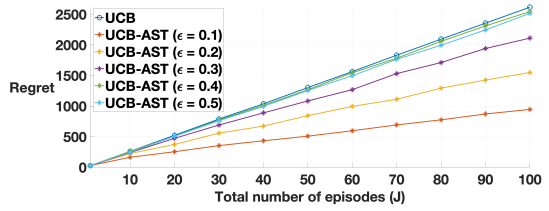
Several comments are in order. First, observe that, for transfer to be beneficial, we need $\min\{A_k^J, B_k^J\}+C_k^J < D_k^J$. Since $A_k^J + C_k^J \geq D_k^J$, this can happen only if $B_k^J + C_k^J < D_k^J$. Next, consider the case $C_k^J = 0$ (which happens when $\Delta_k^j > 2\epsilon, \forall j \in [J], k \in [K]$), and since the term B_k^J has logarithmic dependence with number of episodes J, for some large enough $J^m(\epsilon)$, we get $B_k^J < D_k^J$ which leads to the maximum decrease in the regret (in other words maximum transfer) as compared to NT-UCB. Further, as the sub-optimality gap Δ_k^j decreases, i.e., $\Delta_k^j \leq 2\epsilon$ is satisfied for some episodes, the term C_k^J increases. As a result, the large enough $J^m(\epsilon)$ required to get $B_k^J + C_k^J < D_k^J$ also increases. Note that when the sub-optimality gaps are small, i.e., when the condition $\Delta_k^j \leq 2\epsilon, \forall j \in [J], k \in [K]$ is satisfied, then we get $C_k^J > D_k^J$ which means there is no decrease in the regret and hence no transfer. Second, the dependence of ϵ on $B_k^J + C_k^J < D_k^J$ is analyzed similarly as above, and we get maximum decrease in the regret when $\epsilon = 0$, the large enough $J^m(\epsilon)$ required for $B_k^J + C_k^J < D_k^J$ increases as ϵ increases, and no decrease in regret for $\epsilon \geq 0.5$. Third, we have a logarithmic dependence of episode length n on the regret (which is the case with NT-UCB). Fourth, the third term in the regret bound of AST-UCB (13) is higher than the second term in NT-UCB bound (4) due to the decreased probability bound in Lemma 4 as compared to Lemmas 1 and 3.

V. Numerical Simulations

In this section, we present the numerical results for AST-UCB algorithm.



(a) Regret as function of episode length n, with constant $J = 50$.

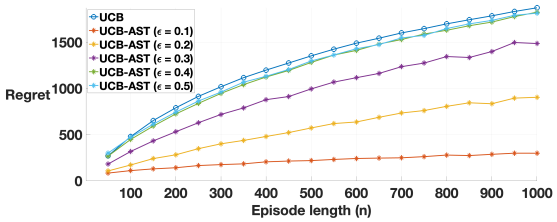


(b) Regret as function of total number of episodes J, with constant $n = 500$.

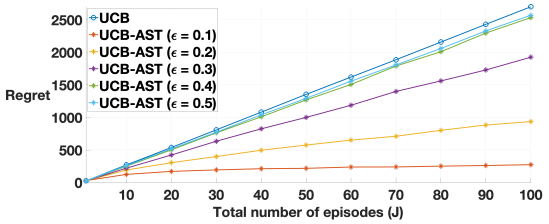
Fig. 2: Empirical regret of NT-UCB and AST-UCB for different values of ϵ for Case I.

We consider $K = 4$ armed bandit problem. In numerical simulations, we need to select the mean reward (μ_k^j) of each arm for each episode which should satisfy Assumption 1. Towards this end, we fix a seed interval of length ϵ for each arm. Then, at the beginning of each episode, we uniformly sample the value of μ_k^j from this seed interval. This ensures that Assumption 1 is satisfied. Once the mean reward value μ_k^j is obtained, we construct a uniform distribution with mean μ_k^j and width $d = 0.2$. In case the support of this uniform distribution lies outside the interval $[0, 1]$, we reduce d to avoid this. The reward samples are then generated from the uniform distribution. For each scenario, we compute the regret R_J by taking an empirical average over 30 independent realizations of that scenario.

We simulate AST-UCB and NT-UCB for two cases (two sets of seed intervals). Note that the seed intervals for each arm are of length ϵ. The mid-points of the seed intervals of the four arms for Case I and Case II are $(0.4, 0.6, 0.6, 0.4)$ and $(0.35, 0.7, 0.3, 0.4)$, respectively.



(a) Regret as function of episode length n, with constant $J = 50$.



(b) Regret as function of total number of episodes J, with constant $n = 500$.

Fig. 3: Empirical regret of NT-UCB and AST-UCB for different values of ϵ for Case II.

In Figure 2a, we observe that the regret of AST-UCB is considerably smaller as compared to NT-UCB. This is particularly true for smaller values of ϵ. As ϵ increases , the regret of AST-UCB approaches to that of NT-UCB. This is in accordance with the fact that when ϵ is large, the confidence interval of the auxiliary estimate in (6) is large and transfer is not much beneficial. Further, we observe a logarithmic dependence of the regret on n, as quantified by the regret bounds in (4) and (13).

In Figure 2b, we again observe that AST-UCB performs better than NT-UCB, particularly for small values of ϵ. We also observe that the regret has a "approximate" linear dependence on J. The plots in Figures 2 show that for any value of ϵ the difference between the regret of NT-UCB and AST-UCB increases with episode length (n) or total number of episodes (J). This is because a larger number of reward samples from previous episodes become available, thereby increasing the transfer.

Similar observations can be seen in Figures 3a and 3b for

Case II. However, the improvement of AST-UCB over NT-UCB in terms of regret is more in Case II as compared to Case I. This happens because the seed intervals in Case II are farther apart, which helps in distinguishing the best arm more quickly using the samples of previous episodes.

VI. CONCLUSION

We analyzed the transfer of reward samples in a sequential stochastic multi-armed bandit setting. We proposed a transfer algorithm based on UCB and showed that its regret is lower than UCB with no transfer. We provide regret analysis of our algorithm and validate our approach via numerical experiments. Future research directions include extending the work to the case when the parameter ϵ is unknown, and studying a similar transfer problem in the context of reinforcement learning.

APPENDIX: PROOF OF THEOREM 1

To simplify the notation, we re-denote several variables as $\mu = \mu_k^j$, $\mu_* = \mu_*^j$, $\hat{\mu}_1 = \hat{\mu}_{1k}^j(t-1)$, $\hat{\mu}_{1*} = \hat{\mu}_{1k_*^j}^j(t-1)$, $\hat{\mu}_2 = \hat{\mu}_{2k}(t-1)$, $\hat{\mu}_{2*} = \hat{\mu}_{2k_*^j}(t-1)$, $t_j^n = (j-1)n+1$,

$$p_1 = \sqrt{\frac{\alpha \log{(t-t_j^n)}}{2N_k^j(t-1)}},\ p_{1*} = \sqrt{\frac{\alpha \log{(t-t_j^n)}}{2N_{k_*^j}^j(t-1)}},$$

$$p_2 = \sqrt{\frac{\alpha \log{((t-1)(t-t_j^n))}}{2S_k(t-1)}} + U_k^j(t-1)\epsilon,$$

$$p_{2*} = \sqrt{\frac{\alpha \log{((t-1)(t-t_j^n))}}{2S_{k_*^j}(t-1)}} + U_{k_*}^j(t-1)\epsilon,$$

$$u_{1k}^j = \frac{2\alpha \log{(n)}}{(\Delta_k^j)^2},\ u_{2k}^j = \frac{2\alpha \log{(Jn^2)}}{(\Delta_k^j - 2\epsilon)^2}.$$

For arm k to be pulled at time t ($I_t = k$), at least one of the following five conditions should be true:

$$\hat{\mu}_1 - p_1 > \mu, \tag{14}$$

$$\hat{\mu}_{1*} + p_{1*} \leq \mu_*, \tag{15}$$

$$\hat{\mu}_{2*} + p_{2*} \leq \mu_*, \tag{16}$$

$$\hat{\mu}_2 - p_2 > \mu, \tag{17}$$

$$\sqrt{\frac{\alpha \log n}{2N_k^j(t-1)}} > \frac{\Delta_k^j}{2} \quad \text{and} \quad \sqrt{\frac{\alpha \log{(Jn^2)}}{2S_k(t-1)}} + \epsilon > \frac{\Delta_k^j}{2}. \tag{18}$$

We show this by contradiction. Assume that none of the conditions in (14)-(17) is true and the first condition in (18) is false. Then, using the fact that $p_1 < \sqrt{\frac{\alpha \log n}{2N_k^j(t-1)}}$, we have

$$\hat{\mu}_{1*} + p_{1*} > \mu_* = \Delta_k^j + \mu \geq 2p_1 + \mu \geq \hat{\mu}_1 + p_1, \tag{19}$$

$$\hat{\mu}_{2*} + p_{2*} > \mu_* = \Delta_k^j + \mu \geq 2p_1 + \mu \geq \hat{\mu}_1 + p_1. \tag{20}$$

Conditions in (19) and (20) imply

$$\min\{\hat{\mu}_{1*} + p_{1*}, \hat{\mu}_{2*} + p_{2*}\} > \hat{\mu}_1 + p_1. \tag{21}$$

Similarly, when none of the conditions in (14)-(17) is true and the second condition in (18) is false, we get

$$\min\{\hat{\mu}_{1*} + p_{1*}, \hat{\mu}_{2*} + p_{2*}\} > \hat{\mu}_2 + p_2. \tag{22}$$

Thus, at least one of the conditions in (21) and (22) is true, and this yields

$$\min\{\hat{\mu}_{1*} + p_{1*}, \hat{\mu}_{2*} + p_{2*}\} > \min\{\hat{\mu}_1 + p_1, \hat{\mu}_2 + p_2\}.$$

The above condition implies that the AST-UCB algorithm will not pull arm k, and hence, we have a contradiction. The cumulative regret after J episodes (each with length n) is given by

$$R_J = \sum_{j=1}^{J}\sum_{k=1}^{K} \Delta_k^j \mathbb{E}[N_k^j(jn)] \leq \sum_{k=1}^{K} \Delta_k^{\max} \mathbb{E}[\tilde{S}_k(Jn)],$$

where $\tilde{S}_k(Jn)$ is the total number of sub-optimal pulls to arm k over all episodes. Next, we bound the regret by bounding the term $\mathbb{E}[\tilde{S}_k(Jn)]$. For an arbitrary sequence I_t, $t = 1, 2, \cdots, Jn$, we have

$$\begin{aligned}
\tilde{S}_k(Jn) &= \sum_{j=1}^{J}\sum_{t=t_j^n}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^j\} \\
&= \sum_{j=1}^{J}\left(\mathbb{1}\{k \neq k_*^j\} + \sum_{t=t_j^n+K}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^j\}\right) \\
&= \sum_{j=1}^{J}\sum_{t=t_j^n+K}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^j, \text{(18) is True}\} \\
&\quad + \sum_{j=1}^{J}\Bigg(\mathbb{1}\{k \neq k_*^j\} + \sum_{t=t_j^n+K}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^j, \\
&\qquad \text{(18) is False}\}\Bigg).
\end{aligned} \tag{23}$$

$$\begin{aligned}
\text{First term in (23)} = \sum_{j=1}^{J}\Bigg(&\sum_{\substack{t=t_j^n+K \\ \Delta_k^j > 2\epsilon}}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^j, \\
&N_k^j(t-1) < u_{1k}^j, S_k(t-1) < u_{2k}^j\} + \\
&\sum_{\substack{t=t_j^n+K \\ \Delta_k^j \leq\ 2\epsilon}}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^j, N_k^j(t-1) < u_{1k}^j\}\Bigg) \\
= \sum_{j=1}^{J}\Bigg(&\sum_{\substack{t=t_j^n+K \\ \Delta_k^j > 2\epsilon}}^{jn} \min\Big\{\mathbb{1}\{I_t = k, k \neq k_*^t, N_k^j(t-1) < \\
&u_{1k}^j\}, \mathbb{1}\{I_t = k, k \neq k_*^t, S_k(t-1) < u_{2k}^j\}\Big\} \\
&+ \sum_{\substack{t=t_j^n+K \\ \Delta_k^j \leq 2\epsilon}}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^j, N_k^j(t-1) < u_{1k}^j\}\Bigg)
\end{aligned}$$

$$\leq \min\Bigg\{ \sum_{j=1}^{J} \sum_{\substack{t=t_j^n+K \\ \Delta_k^j > 2\epsilon}}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^t, N_k^j(t-1) < u_{1k}^j\},$$

$$\sum_{j=1}^{J} \sum_{\substack{t=t_j^n+K \\ \Delta_k^j > 2\epsilon}}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^t, S_k(t-1) < u_{2k}^j\}\Bigg\} +$$

$$\sum_{j=1}^{J} \sum_{\substack{t=t_j^n+K \\ \Delta_k^j \leq 2\epsilon}}^{jn} \mathbb{1}\{I_t = k, k \neq k_*^j, N_k^j(t-1) < u_{1k}^j\}$$

$$\leq \min\Bigg\{ \sum_{\substack{j=1 \\ \Delta_k^j > 2\epsilon}}^{J} \frac{2\alpha \log(n)}{(\Delta_k^j)^2},\ \sum_{t=1}^{Jn} \mathbb{1}\Bigg\{ I_t = k, k \neq k_*^t,$$

$$S_k(t-1) < \max_{\{j \in [J]: \Delta_k^j > 2\epsilon\}} \left\{ \frac{2\alpha \log(Jn^2)}{(\Delta_k^j - 2\epsilon)^2} \right\} \Bigg\} \Bigg\} +$$

$$\sum_{\substack{j=1 \\ \Delta_k^j \leq 2\epsilon}}^{J} \frac{2\alpha \log(n)}{(\Delta_k^j)^2}$$

$$\leq \underbrace{\min\left\{ \sum_{\substack{j=1 \\ \Delta_k^j > 2\epsilon}}^{J} \frac{2\alpha \log(n)}{(\Delta_k^j)^2}, \frac{2\alpha \log(Jn^2)}{(\Delta_k^{\min} - 2\epsilon)^2} \right\} + \sum_{\substack{j=1 \\ \Delta_k^j \leq 2\epsilon}}^{J} \frac{2\alpha \log(n)}{(\Delta_k^j)^2}}_{\triangleq W_k^J} \tag{24}$$

$$\text{Second term of (23)} \leq \sum_{j=1}^{J} \Bigg(1 + \sum_{t=t_j^n+K}^{jn} \mathbb{1}\{\text{(14) or (15) or (16) or (17) is True}\} \Bigg). \tag{25}$$

Using (23), (24), (25) and taking expectation, we get

$$\mathbb{E}[\tilde{S}_k(Jn)] \leq W_k^J + \sum_{j=1}^{J} \Bigg(1 + \sum_{t=t_j^n+K}^{jn} \Pr\{\text{(14) or (15) or (16) or (17) is True}\} \Bigg). \tag{26}$$

Next, we bound the probability of the event that at least one of (14) or (15) or (16) or (17) is true. We use the union bound, followed by the application of one-sided Hoeffding's inequality (steps are similar to the proof of Lemma 1 and 3) to get,

$$\begin{aligned} &\Pr\{\text{(14) or (15) or (16) or (17) is True}\} \\ &\leq \Pr\{\text{(14) is True}\} + \Pr\{\text{(15) is True}\} + \Pr\{\text{(16) is True}\} \\ &\quad + \Pr\{\text{(17) is True}\} \\ &\leq \frac{2}{(t-t_j^n)^{\alpha-1}} + \frac{2}{(t-t_j^n)^{2(\alpha-1)}}. \end{aligned} \tag{27}$$

Using (26) and (27), we have

$$\mathbb{E}[\tilde{S}_k(Jn)] \leq W_k^J + \sum_{j=1}^{J} \Bigg(1 + \sum_{t=t_j^n+K}^{jn} \frac{2}{(t-t_j^n)^{\alpha-1}} + \frac{2}{(t-t_j^n)^{2(\alpha-1)}} \Bigg)$$

$$\leq W_k^J + \sum_{j=1}^{J} \Bigg(1 + \int_{s=(j-1)n+K}^{\infty} \Bigg(\frac{2}{(s-t_j^n)^{\alpha-1}} + \frac{2}{(s-t_j^n)^{2(\alpha-1)}} \Bigg) ds \Bigg)$$

$$\leq W_k^J + J\Bigg(\frac{\alpha}{\alpha-2} + \frac{2}{2\alpha-3} \Bigg).$$

Hence, the theorem follows.

REFERENCES

[1] S. Bubeck, N. Cesa-Bianchi, *et al.*, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Foundations and Trends® in Machine Learning*, vol. 5, no. 1, pp. 1–122, 2012.

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

[3] H. Robbins, "Some aspects of the sequential design of experiments," 1952.

[4] D. Bouneffouf, I. Rish, and C. Aggarwal, "Survey on applications of multi-armed and contextual bandits," in *2020 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–8, 2020.

[5] N. Silva, H. Werneck, T. Silva, A. C. Pereira, and L. Rocha, "Multi-armed bandits in recommendation systems: A survey of the state-of-the-art and future directions," *Expert Systems with Applications*, vol. 197, p. 116669, 2022.

[6] A. Lazaric, E. Brunskill, *et al.*, "Sequential transfer in multi-armed bandit with finite set of models," *Advances in Neural Information Processing Systems*, vol. 26, 2013.

[7] A. Shilton, S. Gupta, S. Rana, and S. Venkatesh, "Regret bounds for transfer learning in bayesian optimisation," in *Artificial Intelligence and Statistics*, pp. 307–315, PMLR, 2017.

[8] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, pp. 235–256, 2002.

[9] M. Soare, O. Alsharif, A. Lazaric, and J. Pineau, "Multi-task linear bandits," in *NIPS2014 workshop on transfer and multi-task learning: theory meets practice*, 2014.

[10] Z. Wang, C. Zhang, M. K. Singh, L. Riek, and K. Chaudhuri, "Multitask bandit learning through heterogeneous feedback aggregation," in *International Conference on Artificial Intelligence and Statistics*, pp. 1531–1539, PMLR, 2021.

[11] Y. Qin, T. Menara, S. Oymak, S. Ching, and F. Pasqualetti, "Non-stationary representation learning in sequential multi-armed bandits," in *ICML Workshop on Reinforcement Learning Theory*, 2021.

[12] L. Cella, A. Lazaric, and M. Pontil, "Meta-learning with stochastic linear bandits," in *International Conference on Machine Learning*, pp. 1360–1370, PMLR, 2020.

[13] L. Cella and M. Pontil, "Multi-task and meta-learning with sparse linear bandits," in *Uncertainty in Artificial Intelligence*, pp. 1692–1702, PMLR, 2021.

[14] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *The collected works of Wassily Hoeffding*, pp. 409–426, 1994.

[15] C. McDiarmid *et al.*, "On the method of bounded differences," *Surveys in combinatorics*, vol. 141, no. 1, pp. 148–188, 1989.