Multi-Arm Bandits over Action Erasure Channels

Osama A. Hanna[†*], Merve Karakas[†*], Lin F. Yang[†] and Christina Fragouli[†]
[†]University of California, Los Angeles
Email:{ohanna, mervekarakas, linyang, christina.fragouli}@ucla.edu

***Abstract*—We consider a novel multi-arm bandit (MAB) setup, where a learner needs to communicate the actions to distributed agents over erasure channels, while the rewards for the actions are directly available to the learner through external sensors. In our model, while the distributed agents know if an action is erased, the central learner does not (there is no feedback), and thus does not know whether the observed reward resulted from the desired action or not. We propose a scheme that can work on top of any (existing or future) MAB algorithm and make it robust to action erasures. Our scheme results in a worst-case regret over action-erasure channels that is at most a factor of $O(1/\sqrt{1-\epsilon})$ away from the no-erasure worst-case regret of the underlying MAB algorithm, where ϵ is the erasure probability. We also propose a modification of the successive arm elimination algorithm and prove that its worst-case regret is $\tilde{O}(\sqrt{KT}+K/(1-\epsilon))$, which we prove is optimal by providing a matching lower bound.**

I. Introduction

Multi-armed bandit (MAB) problems have gained popularity in a wide range of applications that include recommendation systems, clinical trials, advertising, and distributed robotics [1], [2]. MABs are sequential decision problems where a learner, at each round, selects an action from a set of K available actions (arms) and receives an associated reward; the aim is to maximize the total reward over T rounds. In this paper, we develop a theoretical framework that explores a new setup, MAB systems with action erasures.

In particular, we consider a MAB system where the learner needs to communicate the actions to be taken to distributed agents over wireless channels subject to erasures with some probability ϵ, while the rewards for the actions taken are directly available to the learner through external sensors. For example, the learner may be regulating drone traffic in a slowly varying environment (weather conditions, obstacles) where passing by drones receive which path to take or which maneuvers to perform, and the learner monitors the outcome through video cameras, magnetic and other sensors. Similarly, since online learning has been used in medical micro-robots [3], [4], our setup can help with problems such as directing inside veins micro-robots on how to move or how much of a medical substance to release, and observing the results through patient medical imaging and vital sign measurements. In our model, while the agent knows if an action is erased, the learner does not (there is no feedback); therefore, the learner at each round receives a reward that may be a result of playing the requested action or another action (if an erasure occurs).

Note that at any given round, although the requested action may be erased and not received by the agent, the agent still needs to play an action - the micro-robot needs to release some amount (perhaps zero) of substance and the drone needs to continue moving in some way. We make the assumption that if at a round the agent does not receive an action, the agent continues to play the most recently received action, until a new action is successfully received to replace it. This is we believe a reasonable assumption. Indeed, during exploitation, we would like the agents to persistently play the identified optimal action, which this strategy achieves. In contrast, as we discuss in Section III playing a random or a fixed action results in suboptimal performance, while using more sophisticated strategies (e.g., keeping track of all previous actions played and accordingly optimizing) may not be possible in distributed systems, where multiple agents may be playing different actions and where agents may not even be able to observe the rewards.

To the best of our knowledge, this setup has not been systematically studied, although there exists extensive MAB literature. Indeed, existing work has examined MAB systems under a multitude of setups and constraints such as stochastic bandits [5]–[7], adversarial bandits [8]–[10], or contextual bandits [10], [11]; and over a number of distributed setups as well [12]–[15]. In particular, a number of works examine the case where rewards are corrupted adversarially or stochastically [16]–[19]. However, all the works we know of assume that the requested actions can be sent through perfect communication channels. This assumption makes sense for delay-tolerant applications, where we can leverage feedback and error-correcting codes to ensure perfect communication of the desired action; yet as the applicability of the MAB framework expands to agents, such as micro-robots, that are communication limited, we believe that our proposed model can be of interest, both in terms of theory and practice.

The main questions we ask (and answer) in this paper are: for our action erasure model (i) what is an optimal strategy? and (ii) if a MAB algorithm is already deployed, can we couple it, at low regret cost, with a generic layer, akin to erasure coding, that makes the MAB algorithm operation robust to erasures? Our main contributions include:

- We propose a scheme that can work on top of any MAB algorithm and make it robust to action erasures. Our scheme results in a worst-case regret over the erasure channel that is at most a factor of $O(1/\sqrt{1-\epsilon})$ away from

∗ Equal contribution.

This work is partially supported by NSF grants #2221871, #2007714, and #2221871, by Army Research Laboratory grant under Cooperative Agreement W911NF-17-2-0196, and by Amazon Faculty Award.

978-1-6654-7554-9/23/$31.00 ©2023 IEEE

the no-erasure worst-case regret of the underlying MAB algorithm, where $\epsilon \in [0, 1)$ is the erasure probability.
- We propose a modification of the successive arm elimination algorithm and prove that its worst-case regret is $\tilde{O}(\sqrt{KT} + K/(1-\epsilon))$.
- We prove a matching $\Omega(\frac{K}{1-\epsilon})$ regret lower bound.

A. Related Work

MAB Algorithms. Over the years, several stochastic MAB algorithms that achieve optimal or near-optimal regret bounds have been proposed under different assumptions for the environment or model parameters. For instance, over a horizon of length T, Thompson sampling [6] and UCB [5], [7], achieve a gap dependent regret bound of $\tilde{O}(\sum_{i:\Delta_i>0} \frac{1}{\Delta_i})$ and a worst-case regret bound of $O(\sqrt{KT \log T})$, where $\tilde{O}$ hides $\log$ factors. These algorithms achieve nearly optimal regret; a lower bound of $\Omega(\sum_{i:\Delta_i>0} \frac{1}{\Delta_i})$ on the gap-dependent regret was proved in [20] while a lower bound on the worst-case regret of $\Omega(\sqrt{KT})$ was proved in [21]. However, such algorithms are not robust to action erasures.

MAB with Adversarial Corruption. Recently, [16] introduced stochastic multi-armed bandits with adversarial corruption, where the rewards are corrupted by an adversary with a corruption budget C, an upper bound on the accumulated corruption magnitudes over T rounds. [16] proposes a simple elimination algorithm that achieves $\tilde{O}(C\sqrt{KT})$ regret bound in the worst-case. Later, [17] improved the dependency on C to be additive by achieving a worst-case regret bound of $\tilde{O}(\sqrt{KT} + KC)$ while also providing a regret lower bound of $\Omega(C)$. The work in [19] further improves the worst-case regret bound to $\tilde{O}(\sqrt{KT}+C)$ when the optimal arm is unique. The work in [18] studies a similar problem where rewards are corrupted in each round independently with a fixed probability. [18] achieves a regret of $\tilde{O}(\sqrt{KT}+CT)$, where C is an upper bound on the expected corruption, and proves a $\Omega(CT)$ lower bound for their model.

While our model can be reduced to MABs with reward corruption, the amount of corruption C amounts to $\Omega(\epsilon T)$. This causes the best-known algorithms for MABs with adversarial corruption to suffer a linear regret, given the increased exploration required for the large amount of corruption. In contrast, by exploiting the fact that corruptions are in actions, we achieve a gap-dependent regret bound of $\tilde{O}(\sum_{i:\Delta_i>0} \frac{1}{\Delta_i})$ and a worst-case regret bound of $\tilde{O}(\sqrt{KT} + \frac{K}{1-\epsilon})$.

B. Paper Organization

Section II introduces our system model and notation; Section III discusses some straightforward approaches and why they would fail; Section IV, and Section V describe our proposed algorithms and upper bounds; and Section VI provides our lower bound.

II. Problem Formulation

Standard MAB Setup. We consider a MAB problem over a horizon of length T, where a learner interacts with an environment by pulling arms from a set of K arms (or actions).

TABLE I
EXAMPLE OF ACTIONS PLAYED UNDER ACTION ERASURES.

Round (t)	**1**	**2**	**3**	**4**	**5**	**...**
Learner (a_t)	1	3	2	4	2	...
Agent ($\tilde{a}_t$)	1	3	3	3	2	...
Erasure			x	x		...

That is, we assume operation in T discrete times, where at each time $t \in [T]$, the learner sends the index of an arm a_t to an agent; the agent pulls the arm a_t resulting in a reward r_t that can be observed by the learner. The learner selects the arm a_t based on the history of pulled arms and previously seen rewards $a_1, r_1, ..., a_{t-1}, r_{t-1}$. The reward r_t is sampled from an unknown distribution with an unknown mean μ_{a_t}. The set of K distributions for all arm rewards is referred to as a bandit instance. We use Δ_i to denote the gap between the mean value of arm i and the best (highest mean) arm. For simplicity, we follow the standard assumption that rewards are supported on $[0, 1]$. The analysis directly extends to subGaussian reward distributions.

MAB with Action Erasures. We assume that the learner is connected to the agent over an erasure channel, where the action index sent by the learner to the agent at each time t can be erased with probability $\epsilon \in [0, 1)$. We follow the standard erasure channel model where erasures at different times are independent. We assume no feedback in the channel; in particular, while the agent knows when the action is erased (does not receive anything at time t), the learner does not. Moreover, we assume that the learner can directly observe the reward of the action played r_t.

Agent Operation. Recall that at each time t, the learner transmits an action index a_t. The agent plays the most recent action she received: in particular, at time t, the agent plays the action $\tilde{a}_t = a_t$ if no erasure occurs, while if an erasure occurs, $\tilde{a}_t = \tilde{a}_{t-1}$. We assume that $\tilde{a}_0$ is initialized uniformly at random, i.e., if the first action is erased, the agent chooses $\tilde{a}_1$ uniformly at random. In the example described in Table I, $\tilde{a}_4 = \tilde{a}_3 = \tilde{a}_2 = a_2$. Section III motivates why we select this particular agent operation, by arguing that alternative simple strategies can significantly deteriorate the performance.

Performance Metric. The objective is to minimize the regret

$$R_T = \sum_{t=1}^{T} \max_{i\in[K]} \mu_i - \sum_{t=1}^{T} \mu_{\tilde{a}_t}. \tag{1}$$

III. Motivation and Challenges

In this paper we consider a specific agent operation, namely, the agent simply plays the most recently received action. In this section, we argue that this is a reasonable choice, by considering alternate simple strategies and explaining why they do not work well. We also discuss the technical challenges of our approach.

Random Action. A very simple strategy could be, when an erasure occurs, for the agent to uniformly at random select one of the K actions to play. The rewards seen by the learner will

follow a new distribution with mean that is shifted from the pulled arm mean by a constant; in particular,

$$\mathbb{E}[r_t|a_t] = (1-\epsilon)\mu_{a_t} + \epsilon\mathbb{E}[r_t|\varepsilon_t] = (1-\epsilon)\mu_{a_t} + \frac{\epsilon}{K}\sum_{k=1}^{K}\mu_k,$$

where ε_t is the event that erasure occurs at time t. Hence, the best arm does not change, the gaps between arm means do not change, and the learner can identify the best arm and provide a good policy. However, as actions are erased in $\Omega(\epsilon T)$ iterations, the regret becomes $\Omega(\epsilon T)$.

Fixed Action. Very similar arguments hold if, when erasures occur, the agent always plays a fixed (predetermined) action i; unless i happens to be the optimal action, the experienced regret will be linear.

Last Received Action. To see why the previous strategies fail, observe that although the learner may have identified the best policy, this will not be consistently played. In this paper, we make the assumption that if an action is erased, the agent plays the last successfully received action. Thus if the optimal action is identified, it will be consistently played.

We note that our selected agent strategy introduces memory and a challenge in the analysis since it creates a more complicated dependency between the action and erasures. For example, any of the previously sent actions by the learner has a non-zero probability of being played at the current iteration, where the probability changes with time and the sequence of previous actions. In the previous two strategies, the learner still observes a reward with fixed mean and can be treated as a valid MAB instance (although the optimal action might be different from the ground truth). The last received action strategy unfortunately changes the reward distribution and standard MAB analysis no longer holds.

IV. Repeat-the-Instruction Module

In this section we propose a scheme that works on top of any MAB algorithm to make it robust to erasures. Our scheme adds a form of repetition coding layer on top of the MAB algorithm operation. In particular, if the underlying MAB algorithm selects an action to play, our scheme sends the action chosen by the algorithm α times to the agent, and only associates the last received reward with the chosen action (hence the name Repeat-the-Instruction). Since the agent plays the last received action, one successful reception within the α slots is sufficient to make the last reward sampled from the distribution of the chosen arm. Thus our algorithm, summarized in Agorithm 1, effectively partitions the T rounds into T/α groups of rounds, where each group of α rounds is treated by the underlying bandit algorithm as one time slot. The next theorem describes what is the resulting performance.

Theorem 1: Let ALG be a MAB algorithm with expected regret upper bounded by $R_T^{\text{ALG}}(\{\Delta_i\}_{i=1}^K)$ for any instance with gaps $\{\Delta_i\}_{i=1}^K$. For $\alpha = \lceil 2\log T/\log(\frac{1}{\epsilon})\rceil$, using Repeat-the-Instruction on top of ALG achieves an expected regret $\mathbb{E}[R_T]$

$$\mathbb{E}[R_T] \le 2\alpha R^{\text{ALG}}_{\lceil \frac{T}{\alpha}\rceil}(\{\Delta_i\}_{i=1}^K) + \alpha + 1, \tag{2}$$

where the expectation is over the randomness in the MAB instance, erasures, and algorithm.

Algorithm 1 Repeat-the-Instruction Module

Input: α, ALG
for $t \leftarrow 1, ..., T$ **do**
  Learner:
    1) $a_t = \text{ALG}(\frac{t-1}{\alpha}+1)$ **if** $(t \equiv 1 \mod \alpha)$
      else $a_t = a_{t-1}$
    3) observe r_t (corresponding to $\tilde{a}_t$) **if** $(t \equiv 0 \mod \alpha)$
  Agent:
    2) play $\tilde{a}_{t-1}$ **if** erasure **else** play a_t.
      $\tilde{a}_0$ is initialized uniformly at random.
end for

Proof. We use "run" to refer to the α rounds that repeat each action dictated by ALG. Let G be the event that all runs contain at least one round with no erasure. Conditioned on G, the maximum number of consecutive plays for an action due to erasures is $2\alpha - 1$ (across two runs, when the action transmitted at the first round of the first run is not erased, while the first $\alpha - 1$ rounds of the next run are erased). Hence, we have that

$$\mathbb{E}[R_T|G] \le 2\alpha\mathbb{E}[\sum_{i\equiv 1 \bmod \alpha,\ \alpha\le t\le T} \mu_{a^\star} - \mu_{a_t}|G] + \alpha, \tag{3}$$

where $a^\star$ is the optimal arm, the second term bounds the regret for the first α iterations. Note that, conditioned on G, if $t \equiv 0 \bmod \alpha$ with $t \ge \alpha$, we have that $\tilde{a}_t = a_{t-\alpha+1}$, that is, the reward r_t is sampled from arm $a_{t-\alpha+1}$. Indeed in Algorithm 1 r_t is the reward associated with arm $a_{t-\alpha+1}$ for $t \equiv 0 \bmod \alpha, t \ge \alpha$. We observe that as erasures occur independently of the bandit environment and learners actions, we have that for $t \equiv 0 \bmod \alpha, t \ge \alpha$, any set $A \subseteq \mathbb{R}$ and action $a_{t-\alpha+1}$, we have that $\mathbb{P}[r_t \in A|G, a_{t-\alpha+1}] = \mathbb{P}[r_t \in A|\tilde{a}_t = a_{t-\alpha+1}]$ (note that G only depends on erasures). In particular, conditioned on G, the algorithm ALG receives rewards generated from a bandit instance with the same reward distributions as the original instance, hence, the same gaps, and time horizon $\lceil \frac{T}{\alpha}\rceil$. Hence, its regret is upper bounded by $R^{\text{ALG}}_{\lceil \frac{T}{\alpha}\rceil}(\{\Delta_i\}_{i=1}^K)$. Substituting in (3) we get that

$$\mathbb{E}[R_T|G] \le 2\alpha R^{\text{ALG}}_{\lceil \frac{T}{\alpha}\rceil}(\{\Delta_i\}_{i=1}^K) + \alpha. \tag{4}$$

We finally upper bound the probability of the event G. Let G_i denote the event that run i contains at least one slot with no erasure. The probability of G^C can be bounded as

$$\mathbb{P}[G^C] \overset{(1)}{\le} \sum_{i=1}^{\lceil T/\alpha\rceil}\mathbb{P}[G_i^c] = \sum_{i=1}^{\lceil T/\alpha\rceil}\epsilon^\alpha \le \sum_{i=1}^{\lceil T/\alpha\rceil}\frac{1}{T^2} \le 1/T, \tag{5}$$

where (1) follows by the union bound. From (4), we get that

$$\begin{aligned}\mathbb{E}[R_T] &\le 2\alpha R^{\text{ALG}}_{\lceil \frac{T}{\alpha}\rceil}(\{\Delta_i\}_{i=1}^K) + \alpha + T\mathbb{P}[G^C] \\ &\le 2\alpha R^{\text{ALG}}_{\lceil \frac{T}{\alpha}\rceil}(\{\Delta_i\}_{i=1}^K) + \alpha + 1. \qquad \blacksquare\end{aligned}$$

Corollary 1: For $\alpha = \lceil 2\log T/\log(\frac{1}{\epsilon})\rceil$, if R_T^{ALG} is the worst-case expected regret of ALG, then

$$\mathbb{E}[R_T] \le R_T^{\text{ALG}}/\sqrt{1-\epsilon}. \tag{6}$$

This follows from Theorem 1 by observing that $\log(1/\epsilon) = \Omega(1-\epsilon)$, and that the worst-case regret for any algorithm is $\Omega(\sqrt{KT})$. The next corollary follows by substituting the regret bound of the UCB algorithm [5], [7] in Theorem 1.

Corollary 2: For $\alpha = \lceil 2\log T/\log(\frac{1}{\epsilon})\rceil$, the proposed algorithm with the UCB algorithm [5], [7] achieves a gap-dependent regret bounded by $\mathbb{E}[R_T] \leq c\alpha \sum_{i:\Delta_i>0} \frac{\log T}{\Delta_i}$, and a worst-case regret bounded by $\mathbb{E}[R_T] \leq c\sqrt{TK\log T/(1-\epsilon)}$, where c is a universal constant.

Observation. Note that our proposed module achieves the optimal dependency of the regret on T and K. However, we show next that the multiplicative dependency on ϵ is suboptimal by providing an algorithm with a regret bound that has additive dependency on ϵ. This effect is small for small values of ϵ but becomes significant for ϵ approaching 1.

V. The Lingering SAE (L-SAE) Algorithm

The intuition behind the repeat-the-instruction algorithm is that we do not frequently switch between arms - and thus playing the last successfully received action often coincides with playing the desired action. In this section, we propose "Lingering Successive Arm Elimination" (L-SAE), an algorithm that by design does not frequently switch arms, and evaluate its performance; in the next section, we prove a lower bound establishing L-SAE is order optimal.

L-SAE builds on the Successive Arm Elimination (SAE) algorithm [22]. SAE works in batches of exponentially growing length, where at the end of each batch the arms that appear to be suboptimal are eliminated. In batch i, each of the surviving arms is pulled 4^i times. We add two modifications to SAE to make it robust to erasures. First, we do not use the frequent arm switches of the first batches, when 4^i is small. Instead, in the first batch, the algorithm pulls each arm 4α times. Then, the number of pulls for surviving arms in a batch is 4 times that of the previous batch. Second, we ignore the first half of the samples for each arm. This ensures that all the chosen rewards are picked from the desired arm with high probability, as the probability of half the samples being erased is small. Given these modifications, we also update the arm elimination criterion to account for the higher variance in the mean estimates. In particular, the algorithm starts with $A = [K]$ as the good arms set and at the end of batch i, the algorithm eliminates arms with empirical mean that is away by more than $\sqrt{\log(KT)/(\alpha 4^{i-2})}$ from the empirical mean of the arm that appears to be best. The pseudo-code of the algorithm is provided in Algorithm 2.

Theorem 2: For $\alpha = \lceil 2\log T/\log(\frac{1}{\epsilon})\rceil$, Algorithm 2 achieves a regret that is bounded by

$$R_T \leq c\left(\frac{K\log T}{1-\epsilon} + \sum_{i:\Delta_i>0}\frac{\log T}{\Delta_i}\right)$$

with probability at least $1-1/T$, and for a constant c.

Proof. Let G be the good event that the second half of all arm pulls in all batches are from the correct (desired) arm. Note that G does not occur when there is a batch i and an arm j such that all half of arm j pulls in batch i coincide with an erasure. As since in any batch each arm is pulled at least 2α times we have that

$$\mathbb{P}[G] = 1 - \mathbb{P}[G^C] \geq 1 - K\log(T)\epsilon^{\alpha} \geq 1 - 0.25/T. \quad (7)$$

We notice that erasures are independent of the bandit environment, and learner actions; and G only depends on erasures. Hence, conditioned on G and the picked arm a, the second half of the rewards $\{r_j^a\}_{j=M_i/2+1}^{M_i}$ are picked from arm a and they follow the original reward distribution of arm a. Hence, as the rewards are only supported on $[0,1]$, we have that conditioned on G, a, the reward $r_j^a, j > M_i/2$ is $1/4$-subGaussian with mean μ_a. By concentration of sub-Gaussian random variables, conditioned on G, we also have that the following event

$$G' = \{|\mu_a^{(i)} - \mu_a| \leq 2\sqrt{\log(KT)/M_i} \forall a \in A_i \forall i \in [\log T]\},$$

where M_i is the number of pulls for surviving arms in batch i, occurs with probability at least $1 - 0.25/T$. Hence, we have that

$$\mathbb{P}[G \cap G'] \geq (1-0.25/T)^2 \geq 1 - 1/T.$$

In the remaining part of the proof we condition on the event $G \cap G'$. By the elimination criterion in Algorithm 2, and assuming $G \cap G'$, the best arm will not be eliminated. This is because the elimination criterion will not hold for the best arm as

$$\begin{aligned}\mu_a^{(i)} - \mu_{a^\star}^{(i)} &\leq \mu_a - \mu_{a^\star} + 4\sqrt{\log(KT)/M_i} \\ &\leq 4\sqrt{\log(KT)/M_i} \forall a \forall i. \end{aligned} \quad (8)$$

Now consider an arm with gap $\Delta_a > 0$ and let i be the smallest integer for which $4\sqrt{\log(KT)/M_i} < \frac{\Delta_a}{2}$. Then, we have that

$$\begin{aligned}\mu_{a^\star}^{(i)} - \mu_a^{(i)} &\geq \mu_{a^\star} - \mu_a - 4\sqrt{\log(KT)/M_i} > \Delta_a - \frac{\Delta_a}{2} \\ &> 4\sqrt{\log(KT)/M_i}. \end{aligned} \quad (9)$$

Hence, arm a will be eliminated before the start of batch $i+1$. We also notice that from $4\sqrt{\log(KT)/M_i} < \frac{\Delta_a}{2}$, the value of i can be bounded as

$$i \leq \max\{1, \log_4(\frac{65\log(KT)}{\alpha\Delta_i^2})\}, \quad (10)$$

By the exponential increase in the number of pulls of each arm, we get that until eliminated, arm a will be pulled by the learner at most

$$T_a(T) \leq \sum_{j=1}^{i} 4^j\alpha \leq 4^{i+1} \leq c(\alpha + \frac{\log(KT)}{\Delta_a^2}),$$

Algorithm 2 Lingering SAE Algorithm

- Initialize: set of good arms $A = [K]$, batch index $i = 1$.
- For batch i:
  - Pull each arm in A, $M_i = \alpha 4^i$ times to receive rewards $r_1^a, ..., r_{M_i}^a$ for arm $a \in A$.
  - Update means: $\mu_a^{(i)} = \sum_{j=M_i/2+1}^{M_i} r_j^a/(M_i/2)\ \forall a \in A$.
  - $A \leftarrow \{a \in A | \max_{\tilde{a}\in A} \mu_{\tilde{a}}^{(i)} - \mu_a^{(i)} \leq 4\sqrt{\log(KT)/M_i}\}$.
  - $i \leftarrow i+1$.

for some absolute constant $c > 0$. We also notice that on event G, the agent will pull arm i at most $4T_i(T)$ times. This results in a regret that is at most $c(\alpha + \log(KT)/\Delta_a)$. Summing the regret over all arms, we get that conditioned on $G \cap G'$ we have that

$$R_T \leq c\left(\frac{K\log T}{\log(1/\epsilon)} + \sum_{a:\Delta_a>0}\frac{\log(KT)}{\Delta_a}\right).$$

The proof is concluded by noticing that $\log(1/\epsilon) = O(1-\epsilon)$. ■

The previous theorem directly implies the following worst-case regret bound $R_T \leq c\Big(\frac{K\log T}{1-\epsilon} + \sqrt{KT}\Big)$.

VI. Lower Bound

We here prove a lower bound that matches the upper bound in Theorem 2 up to log factors. A lower bound of $\Omega(\sum_{i:\Delta_i>0}\frac{1}{\Delta_i})$ on the gap-dependent regret is already provided in [20] and a lower bound on the worst-case regret of $\Omega(\sqrt{KT})$ is provided in [21]. Thus it suffices to prove a lower bound of $\Omega(K/(1-\epsilon))$ which we provide next in Theorem 3.

Theorem 3: Let $T \geq \frac{K}{4\log(1/\epsilon)}$, and $\epsilon \geq 1/2$. Assuming the agent operation in Section II, for any policy π, there exists a K-armed bandit instance ν such that

$$\mathbb{E}[R_T(\pi,\nu)] \geq c\frac{K}{1-\epsilon}, \tag{11}$$

where $\mathbb{E}[R_T(\pi,\nu)]$ is the expected regret of the policy π over the instance ν and c is a universal constant.

Proof. We consider K bandit instances, each with K arms, where in instance ν_i the means of the reward distributions are

$$\mu_j^{(i)} = \mathbf{1}\{j=i\}, j = 1\dots K \tag{12}$$

with $\mathbf{1}$ the indicator function. We assume a noiseless setting where in instance ν_i picking arm j results in reward $\mu_j^{(i)}$ almost surely. We consider two events:

E_i indicates that the first $1/\log(1/\epsilon)$ pulls of arm i are erased;

$E_i' = \{\tilde{a}_0 \neq i\}$ indicates that the agent in the first round, if the first transmitted arm is erased, does not select to pull arm i (recall that if an erasure occurs at the first round the agent randomly selects an action according to some distribution).

We note that E_i, E_i' are independent events since erasures are independent of the agent operation.

We will show that there is no policy π that can make $\mathbb{E}[R_T(\pi,\nu_i)|E_i\cap E_i']$ to be small for all i. We first note that there are at least[1] $K/2$ arms with $\mathbb{P}[E_i'] \geq 1/2$. Pick a set $I \subseteq [K]$ with $|I| = K/2$, and $\mathbb{P}[E_i'] \geq 1/2\ \forall i \in I$. Now define the event $\mathcal{P}_i$:

$\mathcal{P}_i$: arm i is picked no more than $\frac{1}{\log(1/\epsilon)}$ times by the learner in the first $\frac{K}{4\log(1/\epsilon)}$ rounds.

We next consider the minimum worst case probability of $\mathcal{P}_i$ conditioned on $E_i\cap E_i'$ under the distribution induced by instance ν_i; namely,

$$\min_{\pi}\max_{i\in I}\mathbb{P}_{\nu_i}[\mathcal{P}_i|E_i\cap E_i']. \tag{13}$$

[1]We assume for simplicity that K is divisible by 4. The proof easily generalizes by taking the floor in divisions, which only affects the constants.

Let the set A denote the indices of arms for which $\mathcal{P}_i$ holds, i.e., $A = \{i \in I|\mathcal{P}_i\}$, and let $B = I/A$. We have that A, B are disjoint and their union is the set I. Moreover, the quantity in (13) can be rewritten as

$$\min_{\pi}\max_{i\in I}\mathbb{P}_{\nu_i}[i\in A|E_i\cap E_i']. \tag{14}$$

We make two observations: (i) after $\frac{K}{4\log(1/\epsilon)}$ rounds, there exist at least $\frac{|I|}{2} = \frac{K}{4}$ arms in I for which $\mathcal{P}_i$ holds (hence, $|A| \geq |I|/2$ and $|B| \leq |I|/2$); and (ii) if for instance ν_i, conditioned on $E_i\cap E_i'$, arm i is picked less than $1/\log(1/\epsilon)$ times by the learner, then all the rewards received by the learner will be zeros, thus no information will be provided to the policy during the first $\frac{K}{4\log(1/\epsilon)}$ rounds.

From observation (ii) and the fact that the result of whether $i \in A$ or not, does not change after the first reward feedback that is 1, it follows that the optimal value for (13) does not change if the learner does not observe rewards. Hence, the learner can proceed assuming that all previous rewards are zeros. As a result, the learner can decide on the arms to pull in the first $K/(4\log(1/\epsilon))$ slots ahead of the time (i.e., at $t = 1$; since no feedback is required); equivalently, the learner can divide the set I into two sets A and B (possibly in a random way) ahead of the time with $|B| \leq |I|/2$ (from observation (i)). As the learner decides on A ahead of the time, the probability of $i \in A$ does not depend on the instance, reward outcomes and $\tilde{a}_0$, hence, we can simply denote $\mathbb{P}_{\nu_i}[i\in A|E_i\cap E_i']$ as $\mathbb{P}[i\in A]$. This shows that, the problem of minimizing $\max_{i\in I}\mathbb{P}_{\nu_i}[\mathcal{P}_i|E_i\cap E_i']$ is equivalent to partitioning the set I into two sets A and B (with A containing the indices where $\mathcal{P}_i$ holds), $|A| \leq |I|/2$ and with the goal of minimizing $\max_{i\in I}\mathbb{P}[i\in A]$. It is easy to see that the minimum value for $\max_{i\in I}\mathbb{P}[i\in A]$, and similarly, $\max_{i\in I}\mathbb{P}_{\nu_i}[\mathcal{P}_i|E_i\cap E_i']$, is at least $1/2$ as $\sum_{i=1}^{|I|}\mathbb{P}[i\notin A] \leq |I|/2$.

Hence, for any policy π, there is an instance $\nu_i, i \in I$ such that conditioned on $E_i\cap E_i'$, arm i is picked no more than $1/\log(1/\epsilon)$ times by the learner in the first $K/2\log(1/\epsilon)$ rounds with probability at least $1/2$. But for instance ν_i, whenever arm i is not pulled we incur a regret of value 1, we get that there is $i \in I$ such that $\mathbb{P}\Big[R_T(\pi,\nu_i) \geq \frac{K}{4\log(1/\epsilon)}|E_i\cap E_i'\Big] \geq 1/2$. Then, we have that $\mathbb{E}[R_T(\pi,\nu_i)|E_i\cap E_i'] \geq c\frac{K}{\log(1/\epsilon)}$. By non-negativity of regret, we have that

$$\begin{aligned}\mathbb{E}[R_T(\pi,\nu_i)] &\geq c\frac{K}{\log(1/\epsilon)}\mathbb{P}[E_i\cap E_i']\\ &\overset{(i)}{=} c\frac{K}{\log(1/\epsilon)}\mathbb{P}[E_i]\mathbb{P}[E_i']\\ &\overset{(ii)}{\geq} c/2\frac{K}{\log(1/\epsilon)}\mathbb{P}[E_i],\end{aligned} \tag{15}$$

where (i) follows from the fact that E_i, E_i' are independent, and (ii) follows since $i \in I$ and thus $P(E_i') \geq \frac{1}{2}$. Moreover, $\mathbb{P}[E_i] = \epsilon^{1/\log(1/\epsilon)} = e^{-1}$ and thus $\mathbb{E}[R_T(\pi,\nu_i)] \geq c\frac{K}{\log(1/\epsilon)}$. The proof is concluded by observing that $\log(1/\epsilon) = O(1-\epsilon)$ for $\epsilon \geq 1/2$. ■

References

[1] P. Matikainen, P. M. Furlong, R. Sukthankar, and M. Hebert, "Multi-armed recommendation bandits for selecting state machine policies for robotic systems," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 4545–4551. DOI: 10.1109/ICRA.2013.6631223.

[2] D. Bouneffouf, I. Rish, and C. Aggarwal, "Survey on applications of multi-armed and contextual bandits," in *2020 IEEE Congress on Evolutionary Computation (CEC)*, 2020, pp. 1–8. DOI: 10.1109/CEC48606.2020.9185782.

[3] Y. Yang, M. A. Bevan, and B. Li, "Hierarchical planning with deep reinforcement learning for 3d navigation of microrobots in blood vessels," *Advanced Intelligent Systems*, vol. 4, no. 11, p. 2 200 168, 2022. DOI: https://doi.org/10.1002/aisy.202200168. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/aisy.202200168. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202200168.

[4] Z. Zou, Y. Liu, Y. Young, O. S. Pak, and A. C. H. Tsang, "Gait switching and targeted navigation of microswimmers via deep reinforcement learning," *Commun Phys*, vol. 5, no. 158, 2022. DOI: 10.1038/s42005-022-00935-x.

[5] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, pp. 235–256, May 2002. DOI: 10.1023/A:1013689704352.

[6] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, pp. 285–294, 1933.

[7] T. L. Lai, "Adaptive treatment allocation and the multi-armed bandit problem," *Annals of Statistics*, vol. 15, pp. 1091–1114, 1987.

[8] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. Schapire, "Gambling in a rigged casino: The adversarial multi-armed bandit problem," *Foundations of Computer Science, 1975., 16th Annual Symposium on*, Jul. 1998. DOI: 10.1109/SFCS.1995.492488.

[9] S. Bubeck and A. Slivkins, "The best of both worlds: Stochastic and adversarial bandits," in *Proceedings of the 25th Annual Conference on Learning Theory*, S. Mannor, N. Srebro, and R. C. Williamson, Eds., ser. Proceedings of Machine Learning Research, vol. 23, Edinburgh, Scotland: PMLR, 2012, pp. 42.1–42.23. [Online]. Available: https://proceedings.mlr.press/v23/bubeck12b.html.

[10] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The nonstochastic multiarmed bandit problem," *SIAM Journal on Computing*, vol. 32, no. 1, pp. 48–77, 2002. DOI: 10.1137/S0097539701398375. eprint: https://doi.org/10.1137/S0097539701398375. [Online]. Available: https://doi.org/10.1137/S0097539701398375.

[11] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proceedings of the 19th International Conference on World Wide Web*, ser. WWW '10, Raleigh, North Carolina, USA: Association for Computing Machinery, 2010, pp. 661–670, ISBN: 9781605587998. DOI: 10.1145/1772690.1772758. [Online]. Available: https://doi.org/10.1145/1772690.1772758.

[12] S. Shahrampour, A. Rakhlin, and A. Jadbabaie, "Multi-armed bandits in multi-agent networks," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2786–2790. DOI: 10.1109/ICASSP.2017.7952664.

[13] D. Kalathil, N. Nayyar, and R. Jain, "Decentralized learning for multiplayer multiarmed bandits," *IEEE Transactions on Information Theory*, vol. 60, no. 4, pp. 2331–2345, 2014. DOI: 10.1109/TIT.2014.2302471.

[14] P. Landgren, V. Srivastava, and N. E. Leonard, "Distributed cooperative decision making in multi-agent multi-armed bandits," *Automatica*, vol. 125, p. 109 445, 2021, ISSN: 0005-1098. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109820306439.

[15] P. Landgren, "Distributed multi-agent multi-armed bandits," 2019.

[16] T. Lykouris, V. Mirrokni, and R. Leme, "Stochastic bandits robust to adversarial corruptions," Jun. 2018, pp. 114–122. DOI: 10.1145/3188745.3188918.

[17] A. Gupta, T. Koren, and K. Talwar, "Better algorithms for stochastic bandits with adversarial corruptions," in *Proceedings of the Thirty-Second Conference on Learning Theory*, A. Beygelzimer and D. Hsu, Eds., ser. Proceedings of Machine Learning Research, vol. 99, PMLR, 2019, pp. 1562–1578. [Online]. Available: https://proceedings.mlr.press/v99/gupta19a.html.

[18] S. Kapoor, K. K. Patel, and P. Kar, "Corruption-tolerant bandit learning," *Mach. Learn.*, vol. 108, no. 4, pp. 687–715, 2019, ISSN: 0885-6125. DOI: 10.1007/s10994-018-5758-5. [Online]. Available: https://doi.org/10.1007/s10994-018-5758-5.

[19] I. Amir, I. Attias, T. Koren, Y. Mansour, and R. Livni, "Prediction with corrupted expert advice," *Advances in Neural Information Processing Systems*, vol. 33, pp. 14 315–14 325, 2020.

[20] T. L. Lai, H. Robbins, *et al.*, "Asymptotically efficient adaptive allocation rules," *Advances in applied mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[21] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "Gambling in a rigged casino: The adversarial multi-armed bandit problem," in *Proceedings of IEEE 36th annual foundations of computer science*, IEEE, 1995, pp. 322–331.

[22] P. Auer and R. Ortner, "Ucb revisited: Improved regret bounds for the stochastic multi-armed bandit problem," *Periodica Mathematica Hungarica*, vol. 61, no. 1-2, pp. 55–65, 2010.