DELAYED ADVERSARIAL ATTACKS ON STOCHASTIC BANDITS

Anonymous authors
Paper under double-blind review

ABSTRACT

We study adversarial attacks on stochastic bandits when, differently from previous works, the attack of the malicious attacker is delayed and starts after the learning process begins. We focus on strong attacks and capture the setting in which the malicious attacker lacks information about the beginning of the learning process, a limitation that can dramatically affect the effectiveness of the attack. We introduce a more general framework to study adversarial attacks on stochastic bandit algorithms, providing new definitions of success and profitability of an attack, that account for variable corruption start time. We then analyze success and profitability for different families of algorithms such as UCB and ϵ-greedy against an omniscient attacker. In particular, we derive upper and lower bounds on the number of target arm pulls, showing that our bounds are tight up to a sublinear factor. Finally, we identify an intuitive condition that characterizes when an attack can succeed as a function of its starting time and evaluate the tightness of our theoretical bounds on synthetic instances.

1 INTRODUCTION

The adoption of machine learning applications accelerates at an unprecedented pace, impacting industry and many aspects of humanity. Evaluating how these systems can be damaged is crucial. Many works address this issue in deep learning and reinforcement learning, trying to understand how a malicious entity could "attack" machine learning systems to alter their behavior Goodfellow et al. (2014); Sun et al. (2020); Inkawhich et al. (2019). Other works instead investigate defense strategies, designing robust techniques Chen et al. (2019); Zhang et al. (2020); Pattanaik et al. (2017) In optimal decision-making scenarios, the concept of "attack" can be translated as a way to alter the learner's behavior, fooling the algorithm into selecting specific actions or dramatically reducing the performance. Multi-Armed Bandits (MAB) Auer et al. (2002) are popular online decision-making algorithms whose theoretical guarantees are widely explored. Being a simple yet powerful framework, they are used in various real-world scenarios Bouneffouf et al. (2020) such as medical trials (Durand et al., 2018), recommendation systems (Zhou et al., 2017), advertising (Castiglioni et al., 2022), and finance (Shen et al., 2015; Huo and Fu, 2017).

Adversarial attack framework on multi-armed bandits (MAB) comprises three entities: a learner, an environment, and an attacker. The learner aims to optimize its policy by interacting with the environment. The attacker tries to alter the behavior of the learner by corrupting its reward feedback. The attacker aims to fool the learner into selecting a specific *target arm* which is sub-optimal. An attack is considered *successful* if the learner, upon receiving corrupted observations, selects the target arm $T - o(T)$ times. The attacker decides the amount of corruption to inject observing the arm played by the learner and the reward generated by the environment. This framework, denoted as *strong attack* model, became a standard in adversarial attacks on MAB Jun et al. (2018). The strong attack framework has been shown to be *unrecoverable* for most of the classical bandit algorithms Jun et al. (2018); Liu and Shroff (2019) meaning that a learner under attack will always experience linear regret. To the best of our knowledge, all work about adversarial attacks in stochastic bandits assumes that the attack starts at $t^A = 1$, at the beginning of the learning process. Previous works have proved that, under this attack model, every known technique is unrecoverable from attacks, meaning that an attack is always successful, suggesting that state-of-the-art techniques are not secure. Assuming that the start of the attack and the beginning of the learning process always coincide is a strong assumption, which rarely happens in practice, as it implies that the attacker knows when a

private entity (such as a company) initializes the learning algorithm. Motivated by this observation, we define a novel attack framework that relaxes this assumption and takes into account a generic time t^A for the attacker to start injecting corruption. We call this new framework *delayed attack model* and provide theoretical guarantees about the success of an attack when corruption injection is *delayed* with respect to the beginning of the learning process.

1.1 Original Contributions

Our contributions can be formalized as follows.

- We define the *delayed attack model*, a generalization of the classic adversarial attack framework, where an attack can start at a generic time $t^A > 1$ providing renewed definition of a *successfulness* and *profitability* of an attack. In our model, the previous definition of success may be too restrictive, since, for large enough t^A, the learner may already have built strong estimates about arms. In our definition, even if the attacker fools the learner to select the target arm does not imply that the target arm will be selected a linear number of time. To see this, consider a t^A sufficiently close to T, and the learner may be fooled into believing that the target arm is optimal, but has already exploited the true optimal arm a linear number of times. For this reason, we introduce a more granular metric, the *profitability*, which quantifies the effectiveness of an attack.
- With a fixed oracle attack algorithm, we analyze the success of different families of algorithms for stochastic bandits under our new attack model. For each technique, we derive an upper and lower bound on the number of pulls of the target arm as a function of the start of the attack t^A. In particular, we show that for UCB and ϵ-greedy, under an optimal attack, the number of pulls of the target arm is approximately $(T - t^A) - \frac{\Delta_{o,\tau}}{\epsilon} t^A$, where $\Delta_{o,\tau}$ is the mean reward gap between the optimal arm and the target arm. Our bounds are tight up to a sublinear factor. Finally, we discuss arm elimination techniques, which surprisingly exhibit a natural robustness to attacks under our framework, providing a proof sketch for active arm elimination (AAE).
- Finally, we provide a condition to characterize whether an attack can be successful depending on the start time t^A. In particular, we identify the threshold value $\alpha^*(\Delta_{o,\tau}, \epsilon) = \frac{\epsilon}{\epsilon + \Delta_{o,\tau}}$ that depends on the gap $\Delta_{o,\tau}$ and the parameter ϵ. We show that if the attack starts at time $t^A = \alpha T$ with $\alpha < \alpha^*$, then under our attack, the learner plays the target arm a linear number of times. On the other hand, if the attack starts at time $t^A = \alpha T$ with $\alpha > \alpha^*$, then under any attack the learner plays the target arm a sublinear number of times. We validate our analysis empirically via numerical experiments on synthetic instances.

Related Works Concerning attack strategies, the original adversarial attack model for stochastic bandits was introduced by Jun et al. (2018), proposing different corruption techniques such as oracle attack and attacks for specific specific techniques such as ϵ-greedy and UCB. Liu and Shroff (2019) propose the Adaptive attack by Constant Estimation (ACE), which generalizes previous techniques being not tied to a particular learner's algorithm. Later, Zuo (2020) improved the cost of the attack for UCB in the stochastic setting. All these works operate in the strong attack scenario where the attacker can observe both the played action and the corresponding reward. Works that explore robust techniques tend to assume a weak attack setting, where the attacker can only observe the reward vector generated by the environment. In this setting, Xu et al. (2021) propose an attack technique in which the attacker does not need observations and provides a criterion to characterize families of bandits that are naturally vulnerable to adversarial attacks. Lykouris et al. (2018) propose a robust variation of the Active Arm Elimination (AAE) algorithm agnostic to the amount of corruption injected. Similarly Gupta et al. (2019) proposes a robust algorithm agnostic to corruption. Rangi et al. (2022) study a defensive strategy against a weak attacker in a stochastic setting where the learner can access a limited number of samples free of corruption. Guan et al. (2020) propose a robust algorithm for a different attack model where the attacker can deal with an unbounded attack with a certain probability. Zhong et al. (2021) proposes Probabilistic Sequential Shrinking (PSS), a robust technique for best arm identification problem under adversarial corruption. Aside from the stochastic bandit setting, several works analyze the adversarial attack framework also for adversarial bandits (Ma and Zhou, 2023; Yang et al., 2021), for Gaussian process bandits (Bogunovic et al., 2020a; Han

and Scarlett, 2022), contextual bandits (Garcelon et al., 2020; Bogunovic et al., 2020b; Wang et al., 2022) and combinatorial bandits (Balasubramanian et al., 2024; Dong et al., 2022).

2 PRELIMINARIES

In a Multi-Armed Bandit (MAB) problem Auer et al. (2002) a learner interacts with an environment for T rounds. The learner has K available arms or actions. Each arm $i \in [K]$ [1] is associated with a σ^2-sub-Gaussian reward distribution γ_i with mean μ_i unknown to the learner. At each time $t \in [T]$, the learner selects an arm $i_t \in [K]$ and observes the corresponding reward $r_i(t) \sim \gamma_{i_t}$ generated by the environment. We denote the optimal arm as $o = \arg\max_{i \in [K]} \mu_i$. Let $\mathbb{I}\{\cdot\}$ be the indicator function. Then, we denote by $N_j(t) = \sum_{k=1}^{t} \mathbb{I}\{i_k = j\}$ the number of times an arm $j \in [K]$ has been pulled until time $t \in [T]$, and with $N_j(t_1 \to t_2) = \sum_{k=t_1}^{t_2} \mathbb{I}\{i_k = j\}$ the number of times an arm $j \in [K]$ has been pulled in the interval $[t_1, t_2]$ with $t_1 < t_2$ and $t_1, t_2 \in [T]$. Moreover, we denote by $\Delta_{i,j} \coloneqq \mu_i - \mu_j$ the gap between the means of two different arms $i, j \in [K]$. Finally, we define as $n_i(t) = \{t' \leq t : i_t = i\}$ the set of rounds in which the arm i is selected up to round t, and with $\hat{\mu}_i(t) = \sum_{t' \in n_i(t)} r(t')/N_i(t)$ the average reward of arm i up to round t. The objective is to minimize the regret over the time horizon, where the regret is defined as:

$$R(T) = \mu_o T - \sum_{t=1}^{T} \mu_{i_t}. \tag{1}$$

2.1 RECAP ON STANDARD ADVERSARIAL ATTACKS MODEL

In the classical adversarial attack framework Jun et al. (2018); Liu and Shroff (2019), an additional entity, called the attacker, sits between the learner and the environment. The attacker, in each round $t \in [T]$, upon observing the arm i played by the learner and the reward generated $r_i(t)$ may craft a corruption c_t to alter the reward observed by the learner $\tilde{r}_i(t) = r_i(t) - c_t$. The attacker aims to fool the learner into selecting a target sub-optimal arm τ. We assume that the target arm τ is such that $\mu_\tau < \mu_o$, otherwise, the learner converges to play the target arm even without the attack. [2] The aim of the attacker is to craft the minimal amount of corruption c_t such that the learner, receiving corrupted observations, believes the target arm τ optimal. The attacker is evaluated in terms of successfulness and cost of the attack. An attack is *successful* if the learner selects the target arm τ for $N_\tau(T) = T - o(T)$ rounds in expectation or high-probablity while the attacker pays a sublinear cost (Jun et al., 2018; Liu and Shroff, 2019). The cost is defined as the total corruption injected in the time horizon $C(T) = \sum_{t=1}^{T} |c_t|$. In general, there is no fixed budget for the attack. However, in designing attack techniques we prefer to be *stealth*, that is, we aim for attacks that minimize the cost c_t inflicted at each round, as dealing too much corruption at once can be suspicious in a realistic scenario.

2.2 ORACLE ATTACK

An attack strategy is an online algorithm that, upon observing the arm i played by the learner and the generated reward $r_i(t)$, returns a corruption c_t. The oracle attack model proposed by Jun et al. (2018) is an ideal attack model in which the attacker is omniscient, i.e., knows the true means μ_i for all $i \in [K]$. Although unrealistic in practice, the oracle attack model is useful for a worst-case analysis. In the following, we introduce the main components of the attack in (Jun et al., 2018). Suppose that the attacker knows the true mean of each arm. When the learner selects an arm i different from the target τ the attacker crafts an attack c_t such that:

$$c_t = \mathbb{I}\{i \neq \tau\} \left[\Delta_{\tau,i} + \epsilon\right]_+ \tag{2}$$

Where, $[k]_+ = \max(0, k)$ and ϵ is an arbitrary constant strictly greater than 0. Several attack strategies provide attacks that do not assume an omniscient attacker, such as the attack on UCB in

[1] In this work, we refer as $[A]$, $A \in \mathbb{N}$, to the set $\{1, \ldots, A\}$.

[2] We assume that τ is the arm with the lowest average reward i.e., $\mu_\tau = \min_{i \in [K]} \mu_i$. This is w.l.o.g. because all the arms with mean reward lower than the target arm can be eliminated since they are played a sub-linear number of times even without the attack.

(Jun et al., 2018) or the more generic adaptive attack by constant estimation (ACE) (Liu and Shroff, 2019). The standard framework of adversarial attacks on stochastic bandits that we describe assumes that the attacker injects corruption from $t^A = 1$, that is, when the bandit algorithm is instantiated and has no prior observations. Each known no-regret technique has proven vulnerable under this attack model. However, such a negative result is based heavily on the $t^A = 1$ assumption.

3 Delayed-Attack Framework

In the *delayed adversarial attack* framework we assume a variable attack start time $t^A \geq 1$, and study the success of techniques under adversarial corruption, which as we will discuss later is highly influenced by t^A. Our framework is a generalization of the standard attack model proposed by Jun et al. (2018); Liu and Shroff (2019). In particular, the standard framework corresponds to the specific case where $t^A = 1$. For each time $t \leq t^A$, the learner acts in a corruption-free scenario, as if the attacker is absent. For $t > t^A$ the attacker starts to inject corruption c_t at each round until the end of the horizon T to fool the learner into selecting a sub-optimal target arm τ in place of the optimal arm. Figure 1 provides a graphical representation of the setting. In the paper, we divide the horizon T into two phases, specifically, we refer to the pre-corruption phase indicating $T_1 = t^A$ rounds and the post-corruption phase of T_2 rounds such that $T = T_1 + T_2$. In T_1 the learner acts as in a classic bandit problem without corruption. In T_2 the problem shifts towards an adversarial attack model. The challenge for the attacker is that the learner after T_1 rounds may already have built good estimates of each arm.

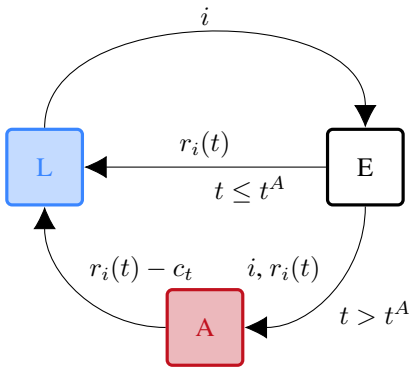


Figure 1: Delayed adversarial attack framework. L represent the learner, E the Environment, and A the attacker.

3.1 Successfulness and Profitability Analysis

In a given attack model, attack strategies are evaluated in terms of successfulness. In particular, an attack is successful if due to corruption, the learner selects the target arm τ a linear number of times $N_\tau(t) = T - o(T)$ while the attacker pays sublinear cost $C(T) = O(\log T)$. This condition is too restrictive for our framework and strongly related to the scenario where $t^A = 1$. Leaving the attack to start any time t^A, we need a weaker notion of successfulness, we define it as the condition in which the attack forces the learner to pull the target arm a linear number of times. Formally:

Definition 3.1 (Successfulness)**.** An Adversarial Attack starting at time t^A to a no-regret learner with σ^2-sub-Gaussian rewards acting over a horizon of T rounds is successful if the target arm τ is selected at least: $N_\tau(T) = \Omega(T)$ times in high probability or expectation.

Intuitively, this condition is intimately related to fool the learner into "believing" that the target arm is optimal. Similarly, if the attack is unsuccessful, the learner will not "believe" the target arm is optimal and injecting corruption may be useless. Starting to attack at time $t^A = 1$, the feasibility of the attack is almost always possible since, from the very first rounds, the learner receives corrupted observations. However, in our scenario, every non-corrupted sample fortifies the learner's estimates, increasing the corruption required to convince the learner to bet on the target arm. Depending on the starting time t^A, we can have situations where the attack is impossible, i.e., the learner will never believe the target arm τ optimal.

Successfulness is a binary condition that distinguishes between successful and non-successful attacks depending on the number of times the target arm is selected. However, this condition is verified in very different situations, as it only requires a linear number of pulls of the target arm. For this reason, we will also use a more fine-grained metric of successfulness called *profitability*. The rationale behind this metric is to quantify the number of times the attacker induces the learner to pull the target arm τ given that the attack is successful. Profitability responds to the question: *"Given that the attack is successful, how many times will the learner select the target arm?"* Profitability corresponds to the quantification of how many times the target arm has been pulled. Formally, it is defined as:

Definition 3.2 (Profitability)**.** The profitability of a successful attack is defined as the number of times the target arm $N_\tau(T)$ is pulled.

In other words, when an attack is successful, the attacker succeeded in making the learner believe that the target arm τ is optimal, fooling the learner into selecting τ a linear number of times $\Omega(T)$. Instead, profitability is the actual measure of the number of times the target arm has been pulled. These two definitions provide a more fine-grained way to quantify an adversarial attack in the proposed setting.

4 Theoretical Analysis

In this section, we consider an *oracle attacker* which inject corruption according to Eq. (2) after a variable time t^A, and analyzes guarantees about the success and profitability of the attack for different families, such as UCB and ϵ-greedy. In the end, we discuss why successive elimination techniques are naturally robust in our setting. We remark that the choice of an oracle attacker not only makes the analysis more intuitive but also offers a worst-case perspective, being the strongest attack in this setting. Before moving on to the analysis, we derive the confidence radius bound for the empirical means of an arm i, for round t. Consider the event:

$$E = \{|\hat{\mu}_i(t) - \mu_i| \leq \beta(N_i(t)) \quad \forall i, \forall t\}, \tag{3}$$

where given a probability $\delta > 0$ we define the confidence radius $\beta(N)$ as a decreasing function in the number of pulls, formally:

$$\beta(N_i(t)) = \sqrt{\frac{\log(\frac{2KT}{\delta})2\sigma^2}{N_i(t)}}. \tag{4}$$

Our radius differs from the one proposed by Jun et al. (2018) and Liu and Shroff (2019) because we consider a fixed horizon T. In the following lemma we prove that the event E holds in high-probability:

Lemma 4.1. *Consider event E, for any* $\delta \in (0, 1)$, $\mathbb{P}(E) > 1 - \delta$

Although the proof is standard, it is reported in Appendix A for completeness. Thanks to Lemma 4.1, with probability at least $1 - \delta$, we can bound the mean μ_i of arm i in the interval $[\hat{\mu}_i(t) - \beta(N_i(t)), \hat{\mu}_i(t) + \beta(N_i(t))]$.

4.1 Successfulness and Profitability Analysis for UCB learner

Consider an oracle attacker and an UCB learner, acting according to the following arm selection rule:

$$i_t = \begin{cases} t, & \text{if } t \leq K \\ \text{argmax}_i \left\{ \hat{\mu}_i(t-1) + 3\sigma\sqrt{\frac{\log T}{N_i(t-1)}} \right\}, & \text{o.w.} \end{cases} \tag{5}$$

To analyze successfulness and profitability, we study the number of pulls of the target arm taking into account delayed corruption. To this extent, we derive a (i) lower and (ii) upper bound on the number of pulls of the target arm $N_\tau(T)$, given that the attacker starts injecting corruption at time $t^A \geq 1$. Intuitively, to prove (i) we derive an upper bound on the number of pulls of the optimal arm in Lemma 4.2, and similarly in Lemma 4.3, an upper bound on a generic sub-optimal arm $i \notin \{o, \tau\}$. Then, we use these results to prove a lower bound for $N_\tau(t^A \to t)$ in Theorem 4.4. In the delayed scenario, we must distinguish between the optimal o and a generic arm i. Since the learner experience $O(\log(T))$ regret and has acted free of corruption for T_1 rounds, it may already have a robust estimate of the optimal arm which may be selected a linear number of time in T_1. Instead, sub-optimal arms have been played a logarithmic number of times and have less consolidated estimates. This scenario is harder for the attacker, since when the attack starts at time $t^A = 1$, the target arm is immediately recognized as optimal, since corruption will fake the observation from the beginning. In our scenario, the learner constructs a corruption-free estimate of each arm for $t < t^A$ rounds, and upon reaching round t^A the UCB learner will select the optimal arm o approximately $N_o(t^A) \approx t^A$ times, while every other arm i will be selected $N_i(t^A) \approx \log(t^A)$. After t^A, corruption starts. We now provide an upper bound for the quantities $N_o(t^A \to t)$ and $N_i(t^A \to t)$. These bounds represent the number of pulls of the optimal arm o and generic sub-optimal arm $i \neq \tau$ before corruption fools the learner

believing the target arm to be optimal. We proceed considering the following inequality, which determines a sufficient condition to ensure that an arm i is not pulled at a given time t:

$$\hat{\mu}_i^c(t) + \beta_{UCB}(N_i(t)) \leq \hat{\mu}_\tau(t) + \beta_{UCB}(N_\tau(t)), \tag{6}$$

where $\hat{\mu}_i^c(t)$ represents the partial corrupt estimator for arm i (corrupt after t^A) defined as:

$$\hat{\mu}_i^c(t) \coloneqq \hat{\mu}_i(t) - \frac{\sum_{k=t^A}^{t} c_k}{N_i(t^A \to t)}, \tag{7}$$

with $t^A < t$ and with c_k being the corruption crafted by the attacker during the corruption interval. Analyzing Inequality (6), we obtain the following lemma, which formally states the bound for the optimal arm pulls in the corruption phase:

Lemma 4.2. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1-\delta$, the learner will select the optimal arm in the corruption phase at most:*

$$N_o(t^A \to T) \leq \frac{(\eta_o + \Delta_{o,\tau})\, T_1}{\epsilon - \eta_o}. \tag{8}$$

For ease of presentation, we use the term $\eta_o \coloneqq \sqrt{\epsilon\sigma^2(2\log(2KT/\delta)+9\log T)/\Delta_{o,\tau}T_1}$ to incorporate the confidence radius β defined in Eq. (4), and the confidence radius of UCB for the optimal arm o, the complete proof can be found in Appendix A. Similarly, in the following lemma we provide an upper bound on the number of pulls for a generic sub-optimal arm $i \notin \{o, \tau\}$. Let $\eta_i \coloneqq \sqrt{\epsilon\sigma^2(2\log(2KT/\delta)+9\log T)/\Delta_{i,\tau}\log(T_1)}$. Equivalently, we use the term η_i for the confidence radius β and UCB for the specific arm i.

Lemma 4.3. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1-\delta$, the learner will select a generic non-optimal arm i in the corruption phase at most:*

$$N_i(t^A \to T) \leq \frac{(\eta_i + \Delta_{i,\tau}) \log T_1}{\epsilon - \eta_i}. \tag{9}$$

Thanks to Lemma 4.2 and Lemma 4.3, we can state the core theorem to lower bound the number of target arm pulls of UCB in the corrupted horizon T_2.

Theorem 4.4. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption with, $\epsilon > 0$ for the remaining T_2 rounds. Then, with probability at least $1-\delta$, the learner will select the target arm τ in the corruption phase at least:*

$$N_\tau(t^A \to T) \geq T_2 - \frac{\eta_o + \Delta_{o,\tau}}{\epsilon - \eta_o} T_1 - \sum_{i \in [K] \setminus \{\tau, o\}} \frac{\eta_i + \Delta_{i,\tau}}{\epsilon - \eta_i} \log T_1.$$

In the following corollary, we derive an asymptotic definition of Theorem 4.4

Corollary 4.5. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1-\delta$, the learner will select the target arm τ in the corruption phase asymptotically at least:*

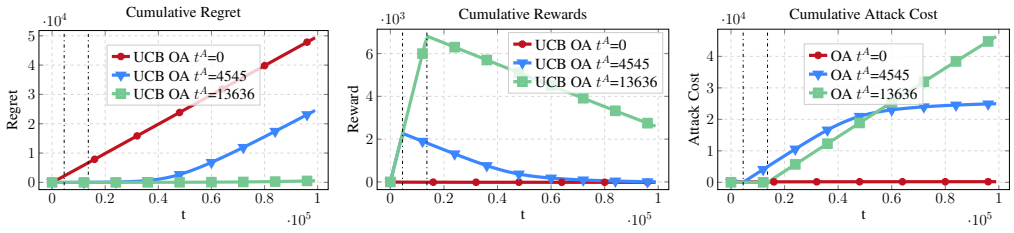
$$N_\tau(t^A \to T) \geq T_2 - \frac{\Delta_{o,\tau}}{\epsilon} T_1 - o(T). \tag{10}$$

Thanks to Corollary 4.5 we can easily recover an asymptotic estimate of the profitability of the attack, since it expresses the lower bound of $N_\tau(t^A \to T)$ in the corruption phase in terms of the instance-dependent gap $\Delta_{o,\tau}$ and the corruption parameter of the attacker ϵ.

4.2 Analysis for ϵ-Greedy

Suppose now that the oracle attacker faces a ϵ-greedy learner, a non-adaptive exploration strategy where the learner has a non-zero probability to explore random action, the rule is as follows:

$$i_t = \begin{cases} \operatorname{argmax}_{i \in [K]} \hat{\mu}_i(t), & \text{w.p. } 1 - \epsilon'(t) \\ \mathcal{U}(K), & \text{o.w.} \end{cases} \tag{11}$$



(a) (b) (c)

Figure 2: Each figure compares three identical UCB learners, each attacked at a different start time t^A. In particular, we show a comparison of the cumulative regrets in Figure 2a, cumulative rewards in Figure 2b and the attack cost in Figure 2c with a 95% confidence interval over 10 experiments. Each learner is attacked at three different times: (i) from the start time ($t^A = 1$) when the learning process begins, (ii) before the successfulness threshold $t^A < \alpha^* T$ and after the successfulness threshold $t^A > \alpha^* T$. These last two attacks are marked with the two dotted vertical lines, to highlight the magnitude of changes in the correspondent regret.

Where $\mathcal{U}(K)$ is the uniform distribution over $[K]$ and $\epsilon'(t)$ is the learning schedule[3] of ϵ-greedy, chosen to be $\epsilon'(t) = 1/t$. To derive a lower bound for $N_\tau(t^A \to T)$ when the learner is ϵ-greedy, we follow a procedure similar to the one used in UCB. First, we derive upper bounds for optimal $N_o(t^A \to T)$ and the generic arm number of pulls $N_o(t^A \to T)$. We consider the following inequality, which determines a sufficient condition to guarantee that an arm i is not pulled at a given time t when the learner is ϵ-greedy:

$$\hat{\mu}_i^c(t) \leq \hat{\mu}_\tau(t). \tag{12}$$

From the above inequality, we can derive the following lemmas:

Lemma 4.6. *Suppose a ϵ-greedy learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1 - \delta$, the learner will select the optimal arm o in the corruption phase at most:*

$$N_o(t^A \to T) \leq \frac{(\gamma_o + \Delta_{i,\tau}) \log T_1}{\epsilon - \gamma_o}. \tag{13}$$

Lemma 4.7. *Suppose a ϵ-greedy learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1 - \delta$, the learner will select a generic non-optimal arm i in the corruption phase at most:*

$$N_i(t^A \to T) \leq \frac{(\gamma_i + \Delta_{i,\tau}) \log T_1}{\epsilon - \gamma_i}. \tag{14}$$

Similarly to UCB analysis γ_o and γ_i refers to sub-constant terms hiding the confidence radius contribution for the ease of visualization. Finally, thanks to Lemma 4.6 and Lemma 4.7 we derive the following theorem

Theorem 4.8. *Suppose a ϵ-greedy learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption with, $\epsilon > 0$ for the remaining T_2 rounds. Then, with probability at least $1 - \delta$, the learner will select the target arm τ in the corruption phase at least:*

$$N_\tau(t^A \to T) \geq T_2 - \frac{\gamma_o + \Delta_{o,\tau}}{\epsilon - \gamma_o} T_1 - \sum_{i \in [K] \setminus \{\tau, o\}} \frac{\gamma_i + \Delta_{i,\tau}}{\epsilon - \gamma_i} \log T_1.$$

4.3 Discussion on Arm Elimination Techniques

With arm elimination algorithms, we refer to techniques that discard sub-optimal arms. Examples are Explore-Then-Commit (ETC) and Active Arm Elimination (AAE) Slivkins (2024). For instance, in AAE an arm i is eliminated at t if $\exists j \in [K]$ s.t. $\hat{\mu}_i(t) + \beta_{UCB}(N_i(t)) < \hat{\mu}_j(t) - \beta_{UCB}(N_j(t))$. Surprisingly, this family of techniques is naturally robust to adversarial attacks in the delayed attack model. Intuitively, for sufficiently large T_1, the target arm might be eliminated, when this happens, the attack cannot be successful.

[3]We use ϵ' to disambiguate between the oracle attack corruption and ϵ-greedy linear schedule.

5 SUCCESSFULNESS THRESHOLD

In this section, we show the implications of previous results on the successfulness of an attack. For simplicity, we will consider an UCB learner, but the results easily extend to ϵ-greedy. To this extent, note that Lemma 4.2 provides the minimum number of rounds $N_o(t^A \to T)$, in the corruption phase, required to change the belief of a UCB learner subject to an oracle attack, to select the target arm. In other words, before the learner believes that τ is optimal, there will still be many $N_o(t^A \to T)$ rounds in which the learner will select the optimal arm o. For some start attack time t^A, even by introducing corruption, the learner may never believe the target arm to be optimal, resulting in an unsuccessful attack. As a trivial example, if the attack starts near the horizon's end, the attack cannot select the target arm a linear number of times. A natural question that arises from reasoning about the above fact is whether there exists a threshold α^* that identifies the break-even point in the horizon $\alpha^* T$ where any attack starting after $t^A > \alpha^* T$ cannot make the learner pull the target arm τ a linear number of times. Intuitively, the threshold α^*, given $\Delta_{o,\tau}$ and fixed a value for ϵ can discriminate for any starting time t^A if the attack is successful. This condition can be derived from Theorem 4.4 and can be expressed in closed form as a parameter $\alpha^*(\Delta_{o,\tau}, \epsilon)$ that depends on the optimal gap $\Delta_{o,\tau}$ and the parameter ϵ. Formally:

Corollary 5.1. *Fixed a constant corruption $\epsilon > 0$. If the attack starts at αT with $\alpha < \alpha^*(\Delta_{o,\tau}, \epsilon)$, a UCB learner will select the target arm τ at least $\Omega(T)$ times with high probability.*

The proof follows from the derivation of the lower bound on $N_\tau(t^A \to T)$. Finally, we prove that our bounds are tight. In particular, with the following theorem, we provide a tight upper bound on the number of target arm pulls.

Theorem 5.2. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, from time αT, with $\alpha < \alpha^*$. Then, with probability at least $-\delta$, the learner will select the target arm τ in the corruption phase at most:*

$$N_\tau(t^A \to T) \leq T_2 - \frac{\Delta_{o,\tau}}{\epsilon} T_1 + o(T).$$

Similarly, as a corollary, we can show that if the attack starts from αT with $\alpha > \alpha^*(\Delta_{o,\tau}, \epsilon)$, it is not successful in high probability.

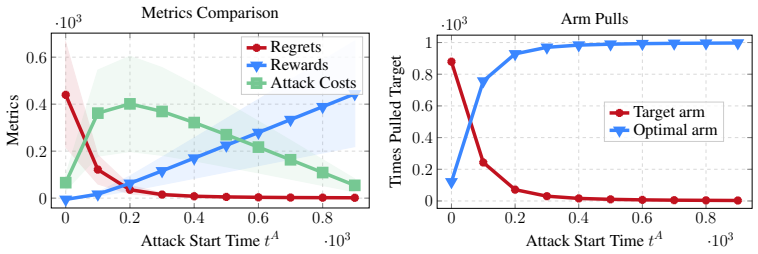
Corollary 5.3. *Fixed a constant corruption $\epsilon > 0$, if the attack starts at αT with $\alpha > \alpha^*(\Delta_{o,\tau}, \epsilon)$, an UCB learner will select the target arm τ at most $o(T)$ times with high probability.*

6 NUMERICAL EXPERIMENTS

We conduct two experiments, one comparing three specific attack times t^A before and after the successfulness threshold defined in Corollary 5.1 and Corollary 5.3. In the second experiment, we show the learner's metrics when the attack can start in every possible round of the horizon T. We use a UCB learner defined in Eq. (5) and an oracle attacker as in Eq. (2).

6.1 COMPARISON BETWEEN SPECIFIC STARTING TIMES

Given the successfulness threshold $\alpha^* = \frac{\epsilon}{\epsilon + \Delta_{o,\tau}}$, we show how the learner behaves when the attack starts at three specific times: (i) from time $t^A = 1$ as in the previous attack model Jun et al. (2018), (ii) from time $t^A < \alpha^* T$ where we prove the attack is still successful, and (iii) from time $t^A > \alpha^* T$ where we proved that the attack is not successful. For (ii) and (iii), we set the attack start time to be $t^A = \frac{1}{2}\alpha^* T$ and $t^A = \frac{3}{2}\alpha^* T$ respectively. Consider a two-arm instance, where the optimal and target arms have a mean reward $\mu_o = \Delta$ and $\mu_\tau = 0$, with $\Delta = 0.5$. The error tolerance δ is set to 0.05, and σ is set to 0.1. The Oracle attacker has the parameter ϵ set to 0.05. As environment parameters, the rewards for each arm i are i.i.d. sampled from a Gaussian distribution $\mathcal{N}(\mu_i, \sigma^2)$. We perform $E = 10$ trials with a horizon $T = 10^5$ and seed $1234 + k,\ k \in \{0, \ldots, E\}$. The results of the experiments shown in Figure 2 highlight that starting the attack at different times drastically changes the outcome. In particular, focusing on the cumulative regrets of Figure 2a we notice how the attack at time $t^A = 1$ and $t^A < \alpha^* T$, although slightly different, results in a linear regret for the learner, in contrast to the UCB instance attacked at $t^A > \alpha^* T$ which is clearly nonlinear. From



(a) (b)

Figure 3: This figure shows the behavior of a UCB learner, victim of an oracle attack, for variable attack start time t^A, with a 95% confidence interval on 15 experiments. On the left, Figure 3a shows the value of total *regrets*, total *rewards* and total *cost of the attack* for different times t^A. While on the right, Figure 3b shows the average number of pulls of the target arm τ and the optimal arm o depending on the starting time t^A.

a practical perspective, these results highlight how much the framework of adversarial attacks on stochastic bandits is sensitive to the start of the attack a parameter that is hardly known by an attacker. Although theoretically these techniques are proven to be weak against adversarial attacks, from a practical perspective, considering the successfulness threshold in the design phase can be helpful in making them robust. Intuitively, the threshold α^* can be used in the design of practical applications to mitigate attacks, for example, by securely training the algorithm until the threshold is reached or using verification techniques in initial samples. Furthermore, from the attacker's point of view, starting the attack after the successfulness threshold implies the highest cost on the horizon T, as shown in Figure 2c. For this reason, any attack performed after $\alpha^* T$ is not worth it in terms of cost, as the attacker pays the highest price without being capable of fooling the learner into selecting the target arm τ.

6.2 Comparison between each possible attack times

In the second experiment, for any time $t^A \in [T]$ we run an instance of UCB algorithm attacked by an oracle attack starting at t^A. For each t^A, we save the sum of the metrics obtained (regrets, rewards, and attack cost). The instance and parameter values, such as the mean reward of arms, δ, ϵ, σ are identical to the previous experiment. We perform $E = 15$ trials with a horizon $T = 10^3$ for each $t^A \in [1, \dots, 10^3]$ setting the seed at $1234 + k,\ k \in \{0 \dots E\}$. Figure 3 shows the results. Given a particular t^A, Figure 3a shows the total regret and the total rewards obtained by the learner, as well as the total attack of the attacker. Figure 3b shows how the number of pulls for the optimal o and target arm τ changes depending on the beginning of corruption. As expected, the target arm pulls down gracefully as the beginning of corruption is delayed.

7 Conclusions

Current state-of-the-art analysis of adversarial attacks in the multi-armed bandit framework assumes that the attacker starts to inject corruption at time $t^A = 1$. This assumption is often unrealistic, and current results show that techniques are unrecoverable from an attack. We provide the delayed attack model, a more fine-grained generalization of the previous attack model where an attack can start at any time $t^A \geq 1$. After characterizing the success of an attack within our framework, we prove for different families of algorithms that the success of an attack strongly depends on the starting time t^A. In particular, we provide lower and upper bounds on the number of times the target arm has been selected and define a threshold to discriminate when an attack can be successful depending on t^A. This model offers a new perspective to study adversarial attacks, closer to a realistic scenario, and opens new possibilities for defending, for instance, by verifying the rewards until reaching the successfulness threshold.

REFERENCES

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235–256, 2002.

Rishab Balasubramanian, Jiawei Li, Prasad Tadepalli, Huazheng Wang, Qingyun Wu, and Haoyu Zhao. Adversarial attacks on combinatorial multi-armed bandits, 2024. URL https://openreview.net/forum?id=vEEWhGjx0M.

Ilija Bogunovic, Andreas Krause, and Jonathan Scarlett. Corruption-Tolerant Gaussian Process Bandit Optimization, March 2020a.

Ilija Bogunovic, Arpan Losalka, Andreas Krause, and Jonathan Scarlett. Stochastic Linear Bandits Robust to Adversarial Attacks, October 2020b.

Djallel Bouneffouf, Irina Rish, and Charu C. Aggarwal. Survey on applications of multi-armed and contextual bandits. *2020 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, 2020.

Matteo Castiglioni, Andrea Celli, and Christian Kroer. Online learning with knapsacks: the best of both worlds. In *International Conference on Machine Learning*, pages 2767–2783. PMLR, 2022.

Tong Chen, Jiqiang Liu, Yingxiao Xiang, Wenjia Niu, Endong Tong, and Zhen Han. Adversarial attack and defense in reinforcement learning-from ai security view. *Cybersecurity*, 2:1–22, 2019.

Jing Dong, Ke Li, Shuai Li, and Baoxiang Wang. Combinatorial bandits under strategic manipulations. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM '22, page 219–229, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450391320. doi: 10.1145/3488560.3498413. URL https://doi.org/10.1145/3488560.3498413.

Audrey Durand, Charis Achilleos, Demetris Iacovides, Katerina Strati, Georgios D. Mitsis, and Joelle Pineau. Contextual bandits for adapting treatment in a mouse model of de novo carcinogenesis. In *Machine Learning in Health Care*, 2018.

Evrard Garcelon, Baptiste Roziere, Laurent Meunier, Jean Tarbouriech, Olivier Teytaud, Alessandro Lazaric, and Matteo Pirotta. Adversarial Attacks on Linear Contextual Bandits, October 2020.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *CoRR*, abs/1412.6572, 2014.

Ziwei Guan, Kaiyi Ji, Donald J. Bucci Jr, Timothy Y. Hu, Joseph Palombo, Michael Liston, and Yingbin Liang. Robust Stochastic Bandit Algorithms under Probabilistic Unbounded Adversarial Attack, February 2020.

Anupam Gupta, Tomer Koren, and Kunal Talwar. Better Algorithms for Stochastic Bandits with Adversarial Corruptions, March 2019.

Eric Han and Jonathan Scarlett. Adversarial Attacks on Gaussian Process Bandits, June 2022.

Xiaoguang Huo and Feng Fu. Risk-aware multi-armed bandit problem with application to portfolio selection. *Royal Society Open Science*, 4, 2017.

Matthew J. Inkawhich, Yiran Chen, and Hai Helen Li. Snooping attacks on deep reinforcement learning. In *Adaptive Agents and Multi-Agent Systems*, 2019.

Kwang-Sung Jun, Lihong Li, Yuzhe Ma, and Jerry Zhu. Adversarial attacks on stochastic bandits. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Fang Liu and Ness Shroff. Data poisoning attacks on stochastic bandits. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4042–4050. PMLR, 09–15 Jun 2019.

Thodoris Lykouris, Vahab Mirrokni, and Renato Paes Leme. Stochastic bandits robust to adversarial corruptions, March 2018.

Yuzhe Ma and Zhijin Zhou. Adversarial Attacks on Adversarial Bandits, January 2023.

Anay Pattanaik, Zhenyi Tang, Shuijing Liu, Gautham Bommannan, and Girish V. Chowdhary. Robust deep reinforcement learning with adversarial attacks. In *Adaptive Agents and Multi-Agent Systems*, 2017.

Anshuka Rangi, Long Tran-Thanh, Haifeng Xu, and Massimo Franceschetti. Saving stochastic bandits from poisoning attacks via limited data verification. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022*, pages 8054–8061. AAAI Press, 2022.

Weiwei Shen, Jun Wang, Yu-Gang Jiang, and Hongyuan Zha. Portfolio choices with orthogonal bandit learning. In *International Joint Conference on Artificial Intelligence*, 2015.

Aleksandrs Slivkins. Introduction to multi-armed bandits, 2024. URL `https://arxiv.org/abs/1904.07272`.

Jianwen Sun, Tianwei Zhang, Xiaofei Xie, L. Ma, Yan Zheng, Kangjie Chen, and Yang Liu. Stealthy and efficient adversarial attacks against deep reinforcement learning. *ArXiv*, abs/2005.07099, 2020.

Huazheng Wang, Haifeng Xu, and Hongning Wang. When Are Linear Stochastic Bandits Attackable?, July 2022.

Yinglun Xu, Bhuvesh Kumar, and Jacob D Abernethy. Observation-free attacks on stochastic bandits. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 22550–22561. Curran Associates, Inc., 2021.

Lin Yang, Mohammad Hassan Hajiesmaili, Mohammad Sadegh Talebi, John C.S. Lui, and Wing S. Wong. Adversarial bandits with corruptions. 2021.

Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Mingyan D. Liu, Duane S. Boning, and Cho-Jui Hsieh. Robust deep reinforcement learning against adversarial perturbations on state observations. *arXiv: Learning*, 2020.

Zixin Zhong, Wang Chi Cheung, and Vincent Y. F. Tan. Probabilistic Sequential Shrinking: A Best Arm Identification Algorithm for Stochastic Bandits with Corruptions, June 2021.

Qiang Zhou, Xiaofang Zhang, Jin Xu, and Bin Liang. Large-scale bandit approaches for recommender systems. In *International Conference on Neural Information Processing*, 2017.

Shiliang Zuo. Near optimal adversarial attacks on stochastic bandits and defenses with smoothed responses. *arXiv preprint arXiv:2008.09312*, 2020.

A Omitted Proofs

Lemma 4.1. *Consider event E, for any $\delta \in (0,1)$, $\mathbb{P}(E) > 1-\delta$*

Proof. Proving that $P(E) \geq 1-\delta$ is equivalent to prove that $P(E^c) \leq \delta$ where E^c is the complementary event. Now let

$$E_{i,t}^c = \{|\hat{\mu}_i(t) - \mu_i| \leq \beta(N_i(t))\},$$

then we have that:

$$\begin{aligned} P(E^c) &= P\left(\bigcup_{i=1}^{K}\bigcup_{t=1}^{T} E_{i,t}^c\right) \\ &\leq \sum_{i=1}^{K}\sum_{t=1}^{T} P\left(E_{i,t}^c\right) && (15) \\ &\leq \sum_{i=1}^{K}\sum_{t=1}^{T} 2\exp\left\{-\frac{N_i(t)\beta(N_i(t))^2}{2\sigma^2}\right\} && (16) \\ &\leq \delta, && (17) \end{aligned}$$

where in Inequality (15) we applied the Union Bound, in Inequality (16) the Hoeffding Bound and Inequality (17) follows by substituting $\beta(N_i(t))$ defined in Equation (4). □

A.1 UCB

Lemma 4.2. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1-\delta$, the learner will select the optimal arm in the corruption phase at most:*

$$N_o(t^A \to T) \leq \frac{(\eta_o + \Delta_{o,\tau})\, T_1}{\epsilon - \eta_o}. \tag{8}$$

Proof. Consider a UCB learner, experiencing $O(\log(t))$ regret. Consider an omniscient attacker, meaning that at each round, given that the optimal arm has been selected, she corrupts the amount $c_t = \Delta_{o,\tau} + \epsilon$. Let $t > t^A$ any round after the corruption has began. If

$$\hat{\mu}_o^c(t) + \beta_{UCB}(N_o(t)) \leq \hat{\mu}_\tau(t) + \beta_{UCB}(N_\tau(t)), \tag{18}$$

where $\mu_o^c(t)$ is a partial corrupted estimator where the corruption only happens in the interval (t^A, t), holds for the optimal arm o, the learner believes that target arm τ is optimal after a corruption phase (we distinguish between optimal arm o and a generic arm i with $i \neq \tau$). Now, the left hand side of Inequality (18) can be upper bounded by:

$$\hat{\mu}_o^c(t) + \beta_{UCB}(N_o(t)) \leq \hat{\mu}_o - \frac{cN_o(t^A \to t)}{N_o(t)} + \beta_{UCB}(N_o(t)),$$

where we have extracted the corruption from the partial corrupted estimator $\hat{\mu}_o^c(t)$. The extraction is possible since in the oracle attack, $\forall t \in [T]$ computes a constant, fixed attack $c_t = c = \Delta_{o,\tau} + \epsilon$. Then we can further upper bounding using the fact that event E holds:

$$\begin{aligned} &\hat{\mu}_o(t) - \frac{cN_o(t^A \to t)}{N_o(t)} + \beta_{UCB}(N_o(t)) \\ &\qquad \leq \mu_o + \beta(N_o(t)) - \frac{cN_o(t^A \to t)}{N_o(t)} + \beta_{UCB}(N_o(t)) \\ &\qquad = \mu_\tau + \Delta_{o,\tau} + \beta(N_o(t)) - \frac{cN_o(t^A \to t)}{N_o(t)} + \beta_{UCB}(N_o(t)) \\ &\qquad \leq \hat{\mu}_\tau(t) + \beta_{UCB}(N_\tau(t)) + \Delta_{o,\tau} + \beta(N_o(t)) - \frac{cN_o(t^A \to t)}{N_o(t)} + \beta_{UCB}(N_o(t)). \end{aligned} \tag{19}$$

Then, if we plug Equation (19) in the Inequality (18) we obtain:

$$\Delta_{o,\tau} + \beta\left(N_o(t)\right) + \beta_{UCB}\left(N_o(t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \leq 0. \tag{20}$$

Now notice that $N_o(t)$, with $t > t^A$ can be rewritten as $N_o(T_1) + N_o(t^A \to t)$. Moreover, since $\beta(N)$ is decreasing in the number of arm pulls we can further upper bound Inequality (20) as:

$$\begin{aligned}
&\Delta_{o,\tau} + \beta\left(N_o(t)\right) + \beta_{UCB}\left(N_o(t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \\
&\qquad \leq \Delta_{o,\tau} + \beta\left(N_o(t^A \to t)\right) + \beta_{UCB}\left(N_o(t^A \to t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \\
&\qquad = \Delta_{o,\tau} + \sqrt{\frac{\log(\frac{2KT}{\delta})2\sigma^2}{N_o(t^A \to t)}} + 3\sigma\sqrt{\frac{\ln(t)}{N_o(t^A \to t)}} - \frac{cN_o(t^A \to t)}{N_o(t)} \\
&\qquad \leq \Delta_{o,\tau} + \sqrt{\frac{\sigma^2\left(2\log\left(\frac{2KT}{\delta}\right) + 9\log t\right)}{N_o(t^A \to t)}} - \frac{cN_o(t^A \to t)}{N_o(t)}.
\end{aligned}$$

Now assume that:

$$N_o(t^A \to t) \geq \frac{\Delta_{o,\tau}}{\epsilon} T_1 \tag{21}$$

Exploiting Equation (21), we have that:

$$\Delta_{o,\tau} + \sqrt{\frac{\sigma^2\left(2\log\left(\frac{2KT}{\delta}\right) + 9\log t\right)}{N_o(t^A \to t)}} - \frac{cN_o(t^A \to t)}{N_o(t)} \leq 0,$$

if

$$\Delta_{o,\tau} + \sqrt{\frac{\epsilon\sigma^2\left(2\log\left(\frac{2KT}{\delta}\right) + 9\log T\right)}{\Delta_{o,\tau} T_1}} - \frac{cN_o(t^A \to t)}{N_o(t)} \leq 0, \tag{22}$$

where we use $t \leq T$. Recalling that

$$\eta \coloneqq \sqrt{\frac{\epsilon\sigma^2\left(2\log\left(\frac{2KT}{\delta}\right) + 9\log T\right)}{\Delta_{o,\tau} T_1}},$$

and plugging it in Inequality (22) we obtain:

$$\begin{aligned}
\Delta_{o,\tau} + \eta - \frac{cN_o(t^A \to t)}{N_o(t)} &= \Delta_{o,\tau} + \eta - \frac{cN_o(t^A \to t)}{N_o(T_1) + N_o(t^A \to t)} \\
&\leq \Delta_{o,\tau} + \eta - \frac{cN_o(t^A \to t)}{T_1 + N_o(t^A \to t)} \\
&= \Delta_{o,\tau} + \eta - \frac{\left(\Delta_{o,\tau} + \epsilon\right) N_o(t^A \to t)}{T_1 + N_o(t^A \to t)},
\end{aligned} \tag{23}$$

where in Inequality (23), we upper bound the number of optimal arm pulls to $N_o(T_1) \approx T_1$. Finally, solving for $N_o(t^A \to t)$ the following inequality:

$$\Delta_{o,\tau} + \eta - \frac{\left(\Delta_{o,\tau} + \epsilon\right) N_o(t^A \to t)}{T_1 + N_o(t^A \to t)} \leq 0,$$

we obtain the following result:

$$N_o(t^A \to t) \geq \frac{(\eta + \Delta_{o,\tau})T_1}{\epsilon - \eta}. \tag{24}$$

However, considering the condition expressed in Inequality (18), if Inequality (24) is true then learner would not act optimally. Implying that:

$$N_o(t^A \to t) \leq \frac{(\eta + \Delta_{o,\tau})T_1}{\epsilon - \eta}. \tag{25}$$

□

Lemma 4.3. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1-\delta$, the learner will select a generic non-optimal arm i in the corruption phase at most:*

$$N_i(t^A \to T) \leq \frac{(\eta_i + \Delta_{i,\tau}) \log T_1}{\epsilon - \eta_i}. \tag{9}$$

Proof. This proof follows similar steps of the proof for the Lemma 4.2. However, there are different assumptions. Again we consider a UCB learner, experiencing $O(\log(t))$ regret. Consider an omniscient attacker, meaning that at each round, given that a generic arm i has been pulled, she corrupts the amount $c_t = \Delta_{i,\tau} + \epsilon$. Let $t > t^A$ any round after the corruption has began. If the following Inequality (26) holds for the generic arm i, the learner believes that target arm τ is better than generic arm i after a corruption phase (here we distinguish between optimal arm o and a generic arm i with $i \notin \{o, \tau\}$).

$$\hat{\mu}_i^c(t) + \beta_{UCB}(N_i(t)) \leq \hat{\mu}_\tau(t) + \beta_{UCB}(N_\tau(t)), \tag{26}$$

where $\mu_i^c(t)$ is a partial corrupted estimator where the corruption only happens in the interval (t^A, t). Now, the left hand side of Inequality (26) can be upper bounded by:

$$\hat{\mu}_i^c(t) + \beta_{UCB}(N_i(t)) \leq \hat{\mu}_i - \frac{cN_i(t^A \to t)}{N_i(t)} + \beta_{UCB}(N_i(t)),$$

where we have extracted the corruption from the partial corrupted estimator $\hat{\mu}_i^c(t)$. The extraction is possible since in the oracle attack, $\forall t \in [T]$ computes a constant, fixed attack $c_t = c = \Delta_{i,\tau} + \epsilon$. Then we can further upper bounding using the fact that event E holds:

$$\begin{aligned}
&\hat{\mu}_i(t) - \frac{cN_i(t^A \to t)}{N_i(t)} + \beta_{UCB}(N_i(t)) \\
&\qquad \leq \mu_i + \beta(N_i(t)) - \frac{cN_i(t^A \to t)}{N_i(t)} + \beta_{UCB}(N_i(t)) \\
&\qquad = \mu_\tau + \Delta_{i,\tau} + \beta(N_i(t)) - \frac{cN_i(t^A \to t)}{N_i(t)} + \beta_{UCB}(N_i(t)) \\
&\qquad \leq \hat{\mu}_\tau(t) + \beta_{UCB}(N_\tau(t)) + \Delta_{i,\tau} + \beta(N_i(t)) - \frac{cN_i(t^A \to t)}{N_i(t)} + \beta_{UCB}(N_i(t)).
\end{aligned} \tag{27}$$

Then, if we plug back Equation (27) in the Inequality (26) we obtain:

$$\Delta_{i,\tau} + \beta(N_i(t)) + \beta_{UCB}(N_i(t)) - \frac{cN_i(t^A \to t)}{N_i(t)} \leq 0 \tag{28}$$

Now notice that $N_i(t)$, with $t > t^A$ can be rewritten as $N_i(T_1) + N_i(t^A \to t)$. Moreover, since $\beta(N)$ is decreasing in the number of arm pulls we can further upper bound Inequality (28) as:

$$\begin{aligned}
&\Delta_{i,\tau} + \beta(N_i(t)) + \beta_{UCB}(N_i(t)) - \frac{cN_i(t^A \to t)}{N_i(t)} \\
&\qquad \leq \Delta_{i,\tau} + \beta\left(N_i(t^A \to t)\right) + \beta_{UCB}\left(N_i(t^A \to t)\right) - \frac{cN_i(t^A \to t)}{N_i(t)} \\
&\qquad = \Delta_{i,\tau} + \sqrt{\frac{\log(\frac{2KT}{\delta})2\sigma^2}{N_i(t^A \to t)}} + 3\sigma\sqrt{\frac{\ln(t)}{N_i(t^A \to t)}} - \frac{cN_i(t^A \to t)}{N_i(t)} \\
&\qquad \leq \Delta_{i,\tau} + \sqrt{\frac{\sigma^2\left(2\log\left(\frac{2KT}{\delta}\right) + 9\log t\right)}{N_i(t^A \to t)}} - \frac{cN_i(t^A \to t)}{N_i(t)},
\end{aligned}$$

Now assume that:

$$N_i(t^A \to t) \geq \frac{\Delta_{i,\tau}}{\epsilon} \log(T_1) \tag{29}$$

Exploiting Equation (29), we have that

$$\Delta_{i,\tau} + \sqrt{\frac{\sigma^2 \left(2 \log\left(\frac{2KT}{\delta}\right) + 9 \log t\right)}{N_i(t^A \to t)}} - \frac{c N_i(t^A \to t)}{N_i(t)} \leq 0$$

if

$$\Delta_{i,\tau} + \sqrt{\frac{\epsilon\sigma^2 \left(2 \log\left(\frac{2KT}{\delta}\right) + 9 \log T\right)}{\Delta_{i,\tau} \log\left(T_1\right)}} - \frac{c N_i(t^A \to t)}{N_i(t)} \leq 0, \tag{30}$$

where we use $t \leq T$. Recalling that

$$\eta_i \coloneqq \sqrt{\frac{\epsilon\sigma^2 \left(2 \log\left(\frac{2KT}{\delta}\right) + 9 \log T\right)}{\Delta_{i,\tau} \log\left(T_1\right)}},$$

and plug it in Inequality (30) we obtain

$$\begin{aligned}
\Delta_{i,\tau} + \eta_i - \frac{c N_i(t^A \to t)}{N_i(t)} &= \Delta_{i,\tau} + \eta_i - \frac{c N_i(t^A \to t)}{N_i(T_1) + N_i(t^A \to t)} \\
&\leq \Delta_{i,\tau} + \eta_i - \frac{c N_i(t^A \to t)}{\log T_1 + N_i(t^A \to t)} \\
&= \Delta_{i,\tau} + \eta_i - \frac{\left(\Delta_{i,\tau} + \epsilon\right) N_i(t^A \to t)}{\log T_1 + N_i(t^A \to t)},
\end{aligned} \tag{31}$$

where in Inequality (31), we upper bound the number of a generic arm pulls to $N_i(T_1) \approx \log\left(T_1\right)$. Finally, solving the following Inequality for $N_i(t^A \to t)$

$$\Delta_{i,\tau} + \psi - \frac{\left(\Delta_{i,\tau} + \epsilon\right) N_i(t^A \to t)}{\log T_1 + N_i(t^A \to t)} \leq 0,$$

we obtain the following result

$$N_i(t^A \to t) \geq \frac{\left(\eta_i + \Delta_{i,\tau}\right)}{\epsilon - \eta_i} \log T_1. \tag{32}$$

However, considering the condition expressed in Inequality (26), if Inequality (32) is true then learner would not act optimally. Implying that:

$$N_i(t^A \to t) \leq \frac{\left(\eta_i + \Delta_{i,\tau}\right)}{\epsilon - \eta_i} \log T_1. \tag{33}$$

□

Theorem 4.4. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption with, $\epsilon > 0$ for the remaining T_2 rounds. Then, with probability at least $1 - \delta$, the learner will select the target arm τ in the corruption phase at least:*

$$N_\tau(t^A \to T) \geq T_2 - \frac{\eta_o + \Delta_{o,\tau}}{\epsilon - \eta_o} T_1 - \sum_{i \in [K] \setminus \{\tau, o\}} \frac{\eta_i + \Delta_{i,\tau}}{\epsilon - \eta_i} \log T_1.$$

Proof. The proof follows from the application of Lemma 4.2 and Lemma 4.3. The number of pulls of the target arm τ in the corruption phase can be defined as:

$$\begin{aligned}
N_\tau(t^A \to T) &= T_2 - \sum_{i \in [K] \setminus \{\tau\}} N_i(t^A \to T) \\
&= T_2 - N_o(t^A \to T) - \sum_{i \in [K] \setminus \{\tau, o\}} N_i(t^A \to T) \\
&\geq T_2 - \frac{\eta + \Delta_{o,\tau}}{\epsilon - \eta} T_1 - \sum_{i \in [K] \setminus \{\tau, o\}} \frac{\eta_i + \Delta_{i,\tau}}{\epsilon - \eta_i} \log T_1
\end{aligned} \tag{34}$$

□

Corollary 4.5. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1 - \delta$, the learner will select the target arm τ in the corruption phase asymptotically at least:*

$$N_\tau(t^A \to T) \geq T_2 - \frac{\Delta_{o,\tau}}{\epsilon} T_1 - o(T). \tag{10}$$

Proof. Consider the result from Theorem 4.4

$$\begin{aligned} N_\tau(t^A \to T) &\geq T_2 - \frac{\eta + \Delta_{o,\tau}}{\epsilon - \eta} T_1 - \sum_{i \in [K] \setminus \{\tau, o\}} \frac{\eta_i + \Delta_{i,\tau}}{\epsilon - \eta_i} \log T_1 \\ &\geq T_2 - \frac{\eta + \Delta_{o,\tau}}{\epsilon - \eta} T_1 - o(T). \end{aligned} \tag{35}$$

Inequality (35) follows from the fact that the term regarding a generic arm $i \in [K] \setminus \{o, \tau\}$, will be selected at most a sub-linear number of times both in pre-corruption and in the corruption phase. This is due to the UCB learner experiencing $O(\log T)$ regret. Furthermore, the term $\frac{\eta + \Delta_{o,\tau}}{\epsilon - \eta}$ in Inequality (35) can be divided in $\frac{\Delta_{o,\tau}}{\epsilon} + \frac{\eta(\epsilon - \Delta_{o,\tau})}{\epsilon(\epsilon + \eta)}$ to obtain:

$$\begin{aligned} T_2 - \frac{\eta + \Delta_{o,\tau}}{\epsilon - \eta} T_1 - o(T) &= T_2 - \left(\frac{\Delta_{o,\tau}}{\epsilon} + \frac{\eta(\epsilon - \Delta_{o,\tau})}{\epsilon(\epsilon + \eta)} \right) T_1 - o(T) \\ &= T_2 - \frac{\Delta_{o,\tau}}{\epsilon} T_1 - \frac{\eta(\epsilon - \Delta_{o,\tau})}{\epsilon(\epsilon + \eta)} T_1 - o(T) \\ &\geq T_2 - \frac{\Delta_{o,\tau}}{\epsilon} T_1 - o(T) \end{aligned} \tag{36}$$

□

Corollary 5.1. *Fixed a constant corruption $\epsilon > 0$. If the attack starts at αT with $\alpha < \alpha^*(\Delta_{o,\tau}, \epsilon)$, a UCB learner will select the target arm τ at least $\Omega(T)$ times with high probability.*

Proof. From results obtained by Theorem 4.4 and Corollary 4.5 we know that:

$$\begin{aligned} N_\tau(t^A \to T) &\geq T_2 - \frac{\Delta_{o,\tau}}{\epsilon} T_1 - o(T) \\ &= (1 - \alpha)T - \frac{\Delta_{o,\tau}}{\epsilon} \alpha T - o(T), \end{aligned} \tag{37}$$

where Equation (37) derives from $T_1 + T_2 = \alpha T + (1 - \alpha)T = T$. Now let $\alpha = \alpha^* - \delta$ where $\alpha^* = \frac{\epsilon}{\epsilon + \Delta_{o,\tau}}$ we can rewrite Equation (37) in:

$$\begin{aligned} (1 - \alpha)T - \frac{\Delta_{o,\tau}}{\epsilon} \alpha T + o(T) &= (1 - \alpha^* + \delta)T - \frac{\Delta_{o,\tau}}{\epsilon} (\alpha^* - \delta) T - o(T) \\ &= \delta T + (1 - \alpha^*)T - \frac{\Delta_{o,\tau}}{\epsilon} (\alpha^* - \delta) T - o(T) \\ &= \delta T + (1 - \alpha^* - \frac{\Delta_{o,\tau}}{\epsilon} \alpha^*) T + \frac{\Delta_{o,\tau}}{\epsilon} \delta T - o(T) \qquad (38) \\ &= \delta T + \frac{\Delta_{o,\tau}}{\epsilon} \delta T - o(T) \\ &= \frac{\epsilon + \Delta_{o,\tau}}{\epsilon} \delta T - o(T) \qquad (39) \end{aligned}$$

The middle term in Equation (38) is exactly 0 thus we obtain Equation (39) as final result. Finally:

$$\begin{aligned} N_\tau(t^A \to T) &\geq \frac{C}{\epsilon} \delta T - o(T) \\ &\geq \Omega(T), \end{aligned}$$

which concludes the proof. □

Theorem 5.2. *Suppose a* UCB *learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, from time αT, with $\alpha < \alpha^*$. Then, with probability at least $-\delta$, the learner will select the target arm τ in the corruption phase at most:*

$$N_\tau(t^A \to T) \leq T_2 - \frac{\Delta_{o,\tau}}{\epsilon} T_1 + o(T).$$

Proof. Consider a bandit instance in which we have only two arms, an optimal arm o and the target arm τ, with true means $\mu_o = 1$ and $\mu_\tau = 1 - \Delta$ respectively. Let $t > t^A$ a generic round t after corruption has began. We want to prove that:

$$N_\tau(t^A \to T) \leq T_2 - \frac{\Delta_{o,\tau} T_1}{\epsilon} + \gamma, \tag{40}$$

where γ is a sub-linear term. To prove Inequality (40), we proceed by contradiction. Consider Inequality (40) false, that is

$$N_\tau(t^A \to T) > T_2 - \frac{\Delta_{o,\tau} T_1}{\epsilon} + \gamma. \tag{41}$$

If Inequality (41) is true, it means that exist a round $t^A < t' < t$ where

$$N_\tau(t^A \to t') \geq T_2 - \frac{\Delta_{o,\tau} T_1}{\epsilon} + \gamma - 1, \tag{42}$$

and the learner selected the arm τ, formally:

$$\mu_o^c + \beta_{UCB}(N_o(t')) \leq \mu_\tau + \beta_{UCB}(N_\tau(t')). \tag{43}$$

Now, to prove that Inequality (41) is a contradiction we need to prove that Inequality (43) is false. Proving Inequality (43) false is equivalent to prove true its contrary, formally:

$$\mu_o^c + \beta_{UCB}(N_o(t')) \geq \mu_\tau + \beta_{UCB}(N_\tau(t')). \tag{44}$$

Finally, to prove Inequality (40) we now reduced to prove Inequality (44) true. Since the instance is defined with only two arms:

$$\begin{aligned} N_o(t^A \to t') &= T_2 - N_\tau(t^A \to t') \\ &\leq \frac{\Delta_{o,\tau} T_1}{\epsilon} - \gamma + 1. \end{aligned}$$

Then, we proceed by lower bounding the left hand side of Inequality (44) obtaining:

$$\begin{aligned} \mu_o^c + \beta_{UCB}(N_o(t)) &\geq \mu_o^c \\ &= \mu_o - \frac{(\epsilon + \Delta_{o,\tau}) N_o(t^A \to t')}{N_o(t')} \\ &= \mu_o - \frac{(\epsilon + \Delta_{o,\tau}) \left(\frac{\Delta_{o,\tau} T_1}{\epsilon} - \gamma + 1 \right)}{T_1 + \frac{\Delta_{o,\tau} T_1}{\epsilon} - \gamma + 1} \\ &\geq \mu_o - \frac{(\epsilon + \Delta_{o,\tau}) \left(\frac{\Delta_{o,\tau} T_1}{\epsilon} + 1 \right)}{T_1 + \frac{\Delta_{o,\tau} T_1}{\epsilon} - \gamma} \end{aligned} \tag{45}$$

Now the right most term in Inequality (45) can be rewritten as:

$$\frac{(\epsilon + \Delta_{o,\tau}) \left(\frac{\Delta_{o,\tau} T_1}{\epsilon} + 1 \right)}{T_1 + \frac{\Delta_{o,\tau} T_1}{\epsilon} - \gamma} = \Delta_{o,\tau} + \frac{\epsilon + \Delta_{o,\tau} + \Delta_{o,\tau}\gamma}{T_1 + \frac{\Delta_{o,\tau} T_1}{\epsilon}},$$

from which we obtain:

$$\mu_o - \Delta_{o,\tau} + \frac{\epsilon + \Delta_{o,\tau} + \Delta_{o,\tau}\gamma}{T_1 + \frac{\Delta_{o,\tau} T_1}{\epsilon}}. \tag{46}$$

Then we upper bounding the right hand side of Inequality (41) obtaining:

$$\begin{aligned}\mu_\tau + \beta_{UCB}(N_\tau(t)) &\leq \mu_\tau + 3\sigma\sqrt{\frac{\log T}{N_\tau(T_1) + N_\tau(t^A \to t')}} \\ &\leq \mu_\tau + 3\sigma\sqrt{\frac{\log T}{N_\tau(t^A \to t')}} \\ &\leq \mu_\tau + 3\sigma\sqrt{\frac{\log T}{T_2 - \frac{\Delta_{o,\tau}T_1}{\epsilon} + \gamma - 1}} \\ &\leq \mu_\tau + 3\sigma\sqrt{\frac{\log T}{T_2 - \frac{\Delta_{o,\tau}T_1}{\epsilon} - 1}},\end{aligned}$$

Finally, we obtain:

$$\mu_o - \Delta_{o,\tau} + \frac{\epsilon + \Delta_{o,\tau} + \Delta_{o,\tau}\gamma}{T_1 + \frac{\Delta_{o,\tau}T_1}{\epsilon}} \geq \mu_\tau + 3\sigma\sqrt{\frac{\log T}{T_2 - \frac{\Delta_{o,\tau}T_1}{\epsilon} - 1}} \tag{47}$$

Thus, Inequality (44) is true for $\gamma \geq \frac{T_1}{\Delta_{o,\tau}}\left(1 - \frac{\Delta}{\epsilon}\right) 3\sigma\sqrt{\frac{\log T}{T_2 - \frac{\Delta_{o,\tau}T_1}{\epsilon} - 1}} - \frac{\epsilon}{\Delta_{o,\tau}}$ resulting in a contradiction. □

A.2 ϵ-GREEDY

Lemma 4.6. *Suppose a ϵ-greedy learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1 - \delta$, the learner will select the optimal arm o in the corruption phase at most:*

$$N_o(t^A \to T) \leq \frac{(\gamma_o + \Delta_{i,\tau}) \log T_1}{\epsilon - \gamma_o}. \tag{13}$$

Proof. Consider a ϵ-greedy learner, experiencing $O(\log(t))$ regret. Consider an omniscient attacker, meaning that at each round, given that the optimal arm has been selected, she corrupts the amount $c_t = \Delta_{o,\tau} + \epsilon$. Let $t > t^A$ any round after the corruption has began. If

$$\hat{\mu}_o^c(t) \leq \hat{\mu}_\tau(t), \tag{48}$$

where $\mu_o^c(t)$ is a partial corrupted estimator where the corruption only happens in the interval (t^A, t), holds for the optimal arm o, the learner believes that target arm τ is optimal after a corruption phase (we distinguish between optimal arm o and a generic arm i with $i \neq \tau$). Now, the left hand side of Inequality (48) can be upper bounded by:

$$\hat{\mu}_o^c(t) \leq \hat{\mu}_o(t) - \frac{cN_o(t^A \to t)}{N_o(t)},$$

where we have extracted the corruption from the partial corrupted estimator $\hat{\mu}_o^c(t)$. The extraction is possible since in the oracle attack, $\forall t \in [T]$ computes a constant, fixed attack $c_t = c = \Delta_{o,\tau} + \epsilon$. Then we can further upper bounding using the fact that event E holds:

$$\begin{aligned}\hat{\mu}_o(t) - \frac{cN_o(t^A \to t)}{N_o(t)} &\leq \mu_o + \beta\left(N_o(t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \\ &= \mu_\tau + \Delta_{o,\tau} + \beta\left(N_o(t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \\ &\leq \hat{\mu}_\tau(t) + \beta(N_\tau(t)) + \Delta_{o,\tau} + \beta\left(N_o(t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)}. \end{aligned} \tag{49}$$

Then, if we plug Equation (49) in the Inequality (48) we obtain:

$$\beta\left(N_\tau(t)\right) + \Delta_{o,\tau} + \beta\left(N_o(t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \leq 0. \tag{50}$$

Now notice that terms $N_x(t)$, with $t > t^A$ can be rewritten as $N_x(T_1) + N_x(t^A \to t)$. We can further reduce Inequality (50), by lower bounding the term $\beta(N_\tau(t))$ as follows:

$$\begin{aligned} N_\tau(t) &= N_\tau(T_1) + N_\tau(t^A \to t) \\ &\approx \log(T_1) + N_\tau(t^A \to t) \qquad (51) \\ &\geq \log(T_1) + \sum_{s=t^A+1}^{t} \frac{\epsilon(t)'}{K} \qquad (52) \\ &\geq \log(T_1) + c(\log(t) - \log(t^A))\frac{1}{K} \end{aligned}$$

Where, in Eq. (51) since the agent is a no-regret learner, in the non-corrupted phase we assume sublinear pulls $N_x(T_1) \approx \log(T_1)$ for each non optimal arm $x \neq o$. Then, in Inequality (52), we lower bound number of target pulls $N_\tau(t^A \to t)$ in corruption phase to be at least the contribution of ϵ-greedy exploration. Using this lower bound we can further upper bound Inequality (50) as follows:

$$\begin{aligned} &\beta\left(N_\tau(t)\right) + \Delta_{o,\tau} + \beta\left(N_o(t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \\ &= \beta\left(N_\tau(T_1) + N_\tau(t^A \to t)\right) + \Delta_{o,\tau} + \beta\left(N_o(T_1) + N_o(t^A \to t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \\ &\leq \beta\left(N_\tau(t^A \to t)\right) + \Delta_{o,\tau} + \beta\left(N_o(t^A \to t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \\ &\leq \beta\left((t^A - t)\frac{\epsilon'(t)}{K}\right) + \Delta_{o,\tau} + \beta\left(N_o(t^A \to t)\right) - \frac{cN_o(t^A \to t)}{N_o(t)} \\ &\leq \Delta_{o,\tau} + \sqrt{\log\left(\frac{2KT}{\delta}\right) 2\sigma^2 \left(\frac{1}{\log(\frac{t}{t^A})\frac{c}{K}} + \frac{1}{N_o(t^A \to t)}\right)} - \frac{cN_o(t^A \to t)}{N_o(t)} \end{aligned} \qquad (53)$$

Where we assume $\log(\frac{2KT}{\delta} = o(\log(t))$. Notice that it is sufficient to take β and δ large enough. Now assume that:

$$N_o(t^A \to t) \geq \frac{\Delta_{o,\tau}}{\epsilon} T_1 \qquad (54)$$

Exploiting Equation (54), we have that:

$$\Delta_{o,\tau} + \sqrt{\log\left(\frac{2KT}{\delta}\right) 2\sigma^2 \left(\frac{1}{\log(\frac{t}{t^A})\frac{c}{K}} + \frac{1}{N_o(t^A \to t)}\right)} - \frac{cN_o(t^A \to t)}{N_o(t)} \leq 0,$$

if

$$\Delta_{o,\tau} + \sqrt{\log\left(\frac{2KT}{\delta}\right) 2\sigma^2 \left(\frac{1}{\log(\frac{T}{t^A})\frac{c}{K}} + \frac{\epsilon}{\Delta_{o,\tau} T_1}\right)} - \frac{cN_o(t^A \to t)}{N_o(t)} \leq 0, \qquad (55)$$

where we use $t \leq T$, and the ϵ-greedy exploration to be the linear schedule $\epsilon'(t) = \frac{1}{t}$. Now we denote the constant term to be:

$$\gamma_o \coloneqq \sqrt{\log\left(\frac{2KT}{\delta}\right) 2\sigma^2 \left(\frac{1}{\log(\frac{T}{t^A})\frac{c}{K}} + \frac{\epsilon}{\Delta_{o,\tau} T_1}\right)},$$

and plugging it in Inequality (55) we obtain:

$$\begin{aligned} \Delta_{o,\tau} + \gamma_o - \frac{cN_o(t^A \to t)}{N_o(t)} &= \Delta_{o,\tau} + \gamma_o - \frac{cN_o(t^A \to t)}{N_o(T_1) + N_o(t^A \to t)} \\ &\leq \Delta_{o,\tau} + \gamma_o - \frac{cN_o(t^A \to t)}{T_1 + N_o(t^A \to t)} \qquad (56) \\ &= \Delta_{o,\tau} + \gamma_o - \frac{(\Delta_{o,\tau} + \epsilon)\, N_o(t^A \to t)}{T_1 + N_o(t^A \to t)}, \end{aligned}$$

where in Inequality (56), we upper bound the number of optimal arm pulls to $N_o(T_1) \approx T_1$. Finally, solving for $N_o(t^A \to t)$ the following inequality:

$$\Delta_{o,\tau} + \gamma_o - \frac{(\Delta_{o,\tau} + \epsilon)\, N_o(t^A \to t)}{T_1 + N_o(t^A \to t)} \leq 0,$$

we obtain the following result:

$$N_o(t^A \to t) \geq \frac{(\gamma_o + \Delta_{o,\tau})T_1}{\epsilon - \gamma_o}. \tag{57}$$

However, considering the condition expressed in Inequality (48), if Inequality (57) is true then learner would not act optimally. Implying that:

$$N_o(t^A \to t) \leq \frac{(\gamma_o + \Delta_{o,\tau})T_1}{\epsilon - \gamma_o}. \tag{58}$$

□

Lemma 4.7. *Suppose a ϵ-greedy learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption, with $\epsilon > 0$, for the remaining T_2 rounds. Then, with probability at least $1 - \delta$, the learner will select a generic non-optimal arm i in the corruption phase at most:*

$$N_i(t^A \to T) \leq \frac{(\gamma_i + \Delta_{i,\tau}) \log T_1}{\epsilon - \gamma_i}. \tag{14}$$

Proof. This proof follows the same steps of the proof for the Lemma 4.6. With the only exception that the term that incorporates the confidence radius is defined as follows:

$$\gamma_i \coloneqq \sqrt{\log\left(\frac{2KT}{\delta}\right) 2\sigma^2 \left(\frac{1}{\log(\frac{T}{t^A})\frac{c}{K}} + \frac{\epsilon}{\Delta_{o,\tau}\log(T_1)}\right)},$$

□

Theorem 4.8. *Suppose a ϵ-greedy learner acts free of corruption for T_1 rounds. An oracle attacker injects corruption with, $\epsilon > 0$ for the remaining T_2 rounds. Then, with probability at least $1 - \delta$, the learner will select the target arm τ in the corruption phase at least:*

$$N_\tau(t^A \to T) \geq T_2 - \frac{\gamma_o + \Delta_{o,\tau}}{\epsilon - \gamma_o} T_1 - \sum_{i \in [K] \setminus \{\tau, o\}} \frac{\gamma_i + \Delta_{i,\tau}}{\epsilon - \gamma_i} \log T_1.$$

Proof. The proof follows from the application of Lemma 4.6 and Lemma 4.7. The proof structure follows the same steps as Theorem 4.4 for UCB. □

B EXPERIMENTS

In this section, we provide minor details about the experiments omitted in the main paper.

Experiments details

- Experiment were conducted using `python 3.11.6`
- CPU: `Apple M1`
- RAM: `16 GB`
- Operating System: `macOS 14.2.1`
- System Type: `64 bit`