When Are Linear Stochastic Bandits Attackable?

Huazheng Wang [*][1] Haifeng Xu [2] Hongning Wang [2]

Abstract

We study adversarial attacks on linear stochastic bandits: by manipulating the rewards, an adversary aims to control the behaviour of the bandit algorithm. Perhaps surprisingly, we first show that some attack goals can never be achieved. This is in a sharp contrast to context-free stochastic bandits, and is intrinsically due to the correlation among arms in linear stochastic bandits. Motivated by this finding, this paper studies the *attackability* of a k-armed linear bandit environment. We first provide a complete necessity and sufficiency characterization of attackability based on the geometry of the arms' context vectors. We then propose a two-stage attack method against LinUCB and Robust Phase Elimination. The method first asserts whether the given environment is attackable; and if yes, it poisons the rewards to force the algorithm to pull a target arm linear times using only a sublinear cost. Numerical experiments further validate the effectiveness and cost-efficiency of the proposed attack method.

1. Introduction

In a contextual bandit problem, a learner takes sequential actions to interact with an environment to maximize its received cumulative reward. As a natural and important variant, linear stochastic bandits (Auer, 2002; Li et al., 2010; Abbasi-yadkori et al., 2011) assume the expected reward of an arm a is a linear function of its feature vector x_a and an unknown bandit parameter $\boldsymbol{\theta}^*$. A linear bandit algorithm thus adaptively improves its estimation of $\boldsymbol{\theta}^*$ based on the reward feedback on its pulled arms. Thanks to their sound theoretical guarantees and promising empirical performance, linear stochastic bandits have become a reference solution to many real-world problems, such as content recommendation and online advertisement (Li et al., 2010; Chapelle & Li, 2011; Durand et al., 2018).

Since bandit algorithms adapt their behavior according to its received feedback, such algorithms are susceptible to adversarial attacks, especially poisoning attacks. Under such an attack, a malicious adversary observes the pulled arm and its reward feedback, and then modifies the reward to misguide the bandit algorithm to pull a target arm, which is of the adversary's interest. Due to the wide applicability of bandit algorithms in practice as mentioned before, understanding the robustness of such algorithms under poisoning attacks becomes increasingly important (Jun et al., 2018; Liu & Shroff, 2019; Garcelon et al., 2020).

Most existing studies on adversarial attacks in bandits focused on the context-free stochastic multi-armed bandit (MAB) settings. Jun et al. (2018) and Liu & Shroff (2019) showed that an adversary can force any MAB algorithm to pull a target arm linear times only using a logarithmic cost. Garcelon et al. (2020) showed the vulnerability of k-armed linear contextual bandits under poisoning attacks. Linear stochastic bandits are related to context-free stochastic bandits and linear contextual bandits. Interestingly, however, there is no known result about attacks on linear stochastic bandit until now. This paper shall provide a formal explanation for this gap — the analysis of attacks to linear stochastic bandits turns out to be significantly more challenging due to the correlation among arms; in fact, *some learning environment is provably unattackable*.

Specifically, we fill the aforementioned gap by studying poisoning attacks on k-armed linear stochastic bandits, where an adversary modifies the reward using a sublinear attack cost to misguide the bandit algorithm to pull a target arm $\tilde{x}$ linear times. We first show that a linear stochastic bandit environment is *not always efficiently attackable*[1], and its attackability is governed by the feasibility of finding a parameter vector $\tilde{\boldsymbol{\theta}}$, under which the rewards of all non-target arms are smaller than the reward of target arm $\tilde{x}$ and the reward of $\tilde{x}$ remains the same as that in the original environment specified by $\boldsymbol{\theta}^*$. Intuitively, to promote the target arm $\tilde{x}$, an adversary needs to lower the rewards of

[*] Most work was done while with University of Virginia. [1] Princeton University [2] University of Virginia. Correspondence to: Huazheng Wang <huazhengwang@gmail.com>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

[1] Throughout this paper, "efficient attack" means fooling the bandit algorithm to pull the target arm for linear times with a sublinear attack cost. We will use *attackable* and *efficiently attackable* interchangeably, as the adversary normally only has a limited budget and needs to design a cost-efficient strategy.

non-target arms in the *null space* of $\tilde{x}$ by $\tilde{\boldsymbol{\theta}}$, which might not be always feasible. We prove the feasibility of the resulting convex quadratic program is both *sufficient* and *necessary* for attacking a linear stochastic bandit environment.

Inspired by our attackability analysis, we propose a two-stage attack framework against linear stochastic bandit algorithms and demonstrate its application against LinUCB (Li et al., 2010) and Robust Phase Elimination (Bogunovic et al., 2021): the former is one of the most widely used linear contextual bandit algorithms, and the latter is a robust version designed for settings with adversarial corruptions. In the first stage, our method collects a carefully calibrated amount of rewards on the target arm to assess whether the given environment is attackable. The decision is based on an "empirical" version of our feasibility characterization. If attackable, i.e., there exists a feasible solution $\tilde{\boldsymbol{\theta}}$, in the second stage the method depresses the rewards the bandit algorithm receives on non-target arms based on $\tilde{\boldsymbol{\theta}}$, to fool the bandit algorithm to recognize the target arm as optimal. We prove that in an attackable environment, both algorithms can be successfully manipulated with only a sublinear cost.

Our main contributions can be summarized as follows:

- We characterize the sufficient and necessary conditions about when a stochastic linear bandit environment is attackable as the feasibility of a convex quadratic program. En route to proving the sufficiency, we also provide an oracle attack method that can attack *any* no-regret learning algorithm given the knowledge of ground-truth bandit parameter $\boldsymbol{\theta}^*$. If the environment is unattackable, i.e., the program is infeasible, our necessity proof implies that even the vanilla LinUCB algorithm cannot be efficiently attacked. A direct corollary of our characterization is that context-free stochastic MAB is always attackable, resonating the findings in (Jun et al., 2018; Liu & Shroff, 2019).
- We propose a two-stage attack method that works *without* the knowledge of ground-truth bandit parameter. In the first stage, the algorithm detects the attackability of the environment and estimates the model parameter. In the second stage, it solves for a working solution $\tilde{\boldsymbol{\theta}}$ and attacks accordingly. Our theoretical analysis shows this attack method is effective against LinUCB (Li et al., 2010) and Robust Phase Elimination (Bogunovic et al., 2021), i.e., pulling the target arm $T - o(T)$ times using $o(T)$ cost when the environment is attackable.

2. Preliminaries

Linear stochastic bandit. We study poisoning attacks to the fundamental k-armed linear stochastic bandit problem (Auer, 2002), where a bandit algorithm sequentially interacts with an environment for T rounds. In each round t, the algorithm pulls an arm $a_t \in [k] = \{1, \cdots, k\}$ from a set $\mathcal{A} = \{x_i\}_{i=1}^k$ with k arms, and receives reward r_{a_t} from the environment. Each arm a is associated with a d-dimensional context vector $x_a \in \mathbb{R}^d$; and the observed reward follows a linear mapping $r_{a_t} = x_{a_t}^\mathsf{T}\boldsymbol{\theta}^* + \eta_t$, where $\boldsymbol{\theta}^* \in \mathbb{R}^d$ is a common unknown bandit parameter vector and η_t is an R-sub-Gaussian noise term. We assume context vectors and parameters are all bounded; and for convenience and without loss of generality, we assume $||x_i||_2 \leq 1$ and $\|\boldsymbol{\theta}^*\|_2 \leq 1$. The performance of a bandit algorithm is evaluated by its pseudo-regret, which is defined as $R_T(\boldsymbol{\theta}^*) = \sum_{t=1}^T (x_{a^*}^\mathsf{T}\boldsymbol{\theta}^* - x_{a_t}^\mathsf{T}\boldsymbol{\theta}^*)$, where a^* is the best arm according to $\boldsymbol{\theta}^*$, i.e., $a^* = \arg\max_{a\in[k]}[x_a^\mathsf{T}\boldsymbol{\theta}^*]$.

LinUCB. LinUCB (Li et al., 2010; Abbasi-yadkori et al., 2011) is a classical algorithm for linear stochastic bandit. It estimates a bandit model parameter $\hat{\boldsymbol{\theta}}$ using ridge regression, i.e., $\hat{\boldsymbol{\theta}}_t = \mathbf{A}_t^{-1}\sum_{i=1}^t x_{a_i} r_i$, where $\mathbf{A}_t = \sum_{i=1}^t x_{a_i}x_{a_i}^\mathsf{T} + \lambda\mathbf{I}$ and λ is the L2-regularization coefficient. We use $\|x\|_\mathbf{A} = \sqrt{x^\mathsf{T}\mathbf{A}x}$ to denote the matrix norm of vector x. The confidence bound about reward estimation on arm x is defined as $\text{CB}_t(x) = \alpha_t\|x\|_{\mathbf{A}_t^{-1}}$, where α_t is a high probability bound of $\|\boldsymbol{\theta}^* - \hat{\boldsymbol{\theta}}_t\|_{\mathbf{A}_t}$. In each round t, LinUCB pulls an arm with the highest upper confidence bound, i.e., $a_t = \arg\max_{a\in[k]}[x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_t + \text{CB}_t(x_a)]$ to balance the exploration-exploitation trade-off. LinUCB algorithm achieves a sublinear upper regret bound, i.e., $R_T = \tilde{O}(\sqrt{T})$ ignoring the logarithmic term.

Threat model. The goal of an adversary is to fool a linear stochastic bandit algorithm to pull the target arm $\tilde{x} \in \mathcal{A}$ for $T - o(T)$ times. Like most recent works in this space (Jun et al., 2018; Liu & Shroff, 2019; Garcelon et al., 2020; Zhang et al., 2020), we also consider the widely studied poisoning attack on the rewards: after arm a_t is pulled by the bandit algorithm, the adversary modifies the realized reward r_{a_t} from the environment by Δr_t into $\tilde{r}_{a_t}$, i.e., $\tilde{r}_{a_t} = r_{a_t} + \Delta r_t$, and feeds the manipulated reward $\tilde{r}_{a_t}$ to the algorithm. Naturally, the adversary aims to achieve its attack goal with minimum attack cost, defined as $C(T) = \sum_{t=1}^T |\Delta r_t|$. By convention, an attack strategy is said to be *efficient* if it uses only a sublinear total cost, i.e., $C(T) = o(T)$.

We conclude the preliminaries with an important remark about a key difference between attacking linear stochastic bandit studied in this paper and attacking k-armed linear contextual bandit setting recently studied in (Garcelon et al., 2020). In linear contextual bandits, all arms share a context vector at each round but each arm has its own (to-be-estimated) bandit parameter. Therefore, the reward manipulation at a round t will only affect the parameter estimation of the pulled arm a_t, but not any other arms'. This "isolates" the attack's impact in different arms. In contrast, in linear stochastic bandit, all arms share the same bandit parameter but have different context vectors. And thus manipulating the reward of any arm will alter the shared bandit parameter estimation, which will then affect the reward estimation of all arms due to the correlation among their context vectors. Such coupled effect of adversarial manipulation from the pulled arm a_t to all other arms is unique in linear stochastic bandits, and makes its analysis of attack much more challenging. This is also the fundamental reason that some linear stochastic bandit environment may not be attackable (see our illustration in Example 1).

3. The Attackability of A Linear Stochastic Bandit Environment

We study the attackability of a linear stochastic bandit *environment*. At the first glance, one might wonder whether *attackability* is the property of a bandit *algorithm* rather than a property of the environment, since if an algorithm can be attacked, we should have "blamed" the algorithm for not being robust. A key finding of this work is attackability is also a property of the learning environment; and in other words, *not* all environments are attackable.

Definition 1 (Attackability of a k-Armed Linear Stochastic Bandit Environment)**.** *A k-armed linear stochastic bandit environment $\langle \mathcal{A} = \{x_i\}_{i=1}^k, \boldsymbol{\theta}^* \rangle$ is attackable with respect to the target arm $\tilde{x} \in \mathcal{A}$ if for* any *no-regret learning algorithm, there exists an attack method that uses $o(T)$ attack cost and fools the algorithm to pull arm $\tilde{x}$ at least $T - o(T)$ times with high probability*[2] *for any T large enough, i.e., T larger than a constant T_0.*

We make a few remarks about the above definition of attackability. First, this definition is all about the bandit environment $\langle \mathcal{A}, \boldsymbol{\theta}^* \rangle$ and the target arm $\tilde{x}$, but independent of any specific bandit algorithm. Second, if an attack method can only fool a bandit algorithm to pull the target arm $\tilde{x}$ under a few different time horizons T, it does not count as a successful attack – it has to succeed for infinitely many time horizons. Third, by reversing the order of quantifiers, we obtain the assertion that a bandit environment is not attackable w.r.t. the target arm $\tilde{x}$ if *there exists some no-regret learning algorithm* such that no attack method can use $o(T)$ attack cost to fool the algorithm to pull arm $\tilde{x}$ at least $T - o(T)$ times with high probability for any T large enough.

The following simple yet insightful example illustrates that there are indeed linear stochastic bandit environment setups where some no-regret learning algorithm *cannot* be attacked.

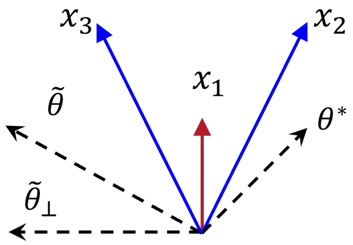

Figure 1. Illustration of attackability of a linear stochastic bandit environment.

Example 1 (An unattackable environment)**.** *Figure 1 shows a three-arm environment with $\mathcal{A} = \{x_1 = (0,1), x_2 = (1,2), x_3 = (-1,2)\}$. Suppose the target arm $\tilde{x} = x_1$ and the ground-truth bandit parameter $\boldsymbol{\theta}^* = (1,1)$*[3]*. The expected true rewards of the arms are $r_1^* = 1, r_2^* = 3, r_3^* = 1$ and x_2 is the best arm in this environment. Based on Definition 1, we will need to identify a no-regret learning algorithm that cannot be attacked in this environment, and we argue that LinUCB is such an algorithm. Suppose, for the sake of contradiction, that there exists an efficient attack which fools LinUCB to pull x_1 $T - o(T)$ times. LinUCB then must estimate $\boldsymbol{\theta}^*$ in the x_1's direction almost accurately as T becomes large, since the $\Omega(T)$ amount of true reward feedback in x_1 direction will ultimately dominate the $o(T)$ adversarial manipulations. This suggests that the estimated parameter $\hat{\boldsymbol{\theta}}_t$ will be close to $(\alpha, 1)$ for some α. Since $(\alpha, 1)^\mathsf{T}(x_2 + x_3) = 4$, implying that either x_2 or x_3 will have its estimated reward larger than 2 (i.e., strictly larger than x_1's estimated reward) for any α. This shows that for large T, x_1 cannot be the arm with the highest UCB during the execution of LinUCB, which causes a contradiction. Therefore, this environment cannot be efficiently attacked with $o(T)$ cost. Here we give an intuitive argument about this environment with target arm $\tilde{x}$ is not attackable, while its formal proof is an instantiation of our Theorem 1.*

Note that when $\mathcal{A} = \{x_1, x_2\}$, the environment becomes attackable: as shown in Figure 1, a feasible attack strategy is to perturb reward according to $\tilde{\boldsymbol{\theta}} = (-2, 1)$. The key idea is that in the null space of x_1, $\tilde{\boldsymbol{\theta}}_\perp$ reduces the reward of x_2 to make x_1 the best arm without changing the actual reward of x_1 from the environment. The presence of arm x_3 prevents the existence of such a $\tilde{\boldsymbol{\theta}}_\perp$ (and therefore $\tilde{\boldsymbol{\theta}}$) and makes the environment unattackable.

The above example motivates us to study when a linear stochastic bandit environment is attackable. After all, only when facing an unattackable environment, we can ensure the existence of a no-regret algorithm that will be robust to any $o(T)$ poisoning attacks.

Next we show that there indeed exists a complete character-

[2] Typically, the high probability refers to $1 - \delta$ probability for an arbitrarily small δ. Please see theorems later for rigorous statements.

[3] One can re-scale all vectors with norms smaller than 1, e.g., divide each dimension by 10, without changing the conclusion that the environment is unattackable.

ization about when a linear stochastic bandit environment is attackable. As Example 1 shows, the attackability of a bandit environment depends on the arm set $\mathcal{A} = \{x_i\}_{i=1}^k$, the target arm $\tilde{x}$, and the underlying bandit parameter $\boldsymbol{\theta}^*$. For clarity of presentation, in this section, we shall always assume that the adversary knows exactly the ground-truth bandit parameter $\boldsymbol{\theta}^*$ and thus the true expected reward of each arm. This is also called the *oracle attack* in previous works (Jun et al., 2018; Rakhsha et al., 2020). But in the next section, we will show that this assumption is not necessary: when the bandit environment is indeed attackable, we can design provably successful attacks even if the adversary does not know the underlying bandit parameter $\boldsymbol{\theta}^*$.

We need the following convenient notation to state our results. Let

$$\boldsymbol{\theta}_{\parallel}^* = \frac{\tilde{x}^{\mathsf{T}}\boldsymbol{\theta}^*}{\|\tilde{x}\|_2^2}\tilde{x} \tag{1}$$

denote the projection of ground-truth bandit parameter $\boldsymbol{\theta}^*$ onto the targeted $\tilde{x}$ direction. Since the attackability depends on the target arm $\tilde{x}$ as well, we shall include the target arm $\tilde{x}$ as part of the bandit environment. The following theorem provides a clean characterization about the attackability of a linear stochastic bandit environment.

Theorem 1 (Characterization of Attackability). *A bandit environment $\langle \mathcal{A} = \{x_i\}_{i=1}^k, \boldsymbol{\theta}^*, \tilde{x}\rangle$ is attackable* if and only if *the optimal objective ϵ^* of the following convex quadratic program (CQP) satisfies $\epsilon^* > 0$.*

$$\begin{aligned}
\textit{maximize} \quad & \epsilon \\
\textit{subject to} \quad & \tilde{x}^{\mathsf{T}}\boldsymbol{\theta}_{\parallel}^* \geq \epsilon + x_a^{\mathsf{T}}(\boldsymbol{\theta}_{\parallel}^* + \tilde{\boldsymbol{\theta}}_{\perp}), \quad \textit{for } x_a \neq \tilde{x}. \\
& \tilde{x}^{\mathsf{T}}\tilde{\boldsymbol{\theta}}_{\perp} = 0 \\
& \|\boldsymbol{\theta}_{\parallel}^* + \tilde{\boldsymbol{\theta}}_{\perp}\|_2 \leq 1
\end{aligned} \tag{2}$$

where $\epsilon \in \mathbb{R}$ and $\tilde{\boldsymbol{\theta}}_{\perp} \in \mathbb{R}^d$ are variables.

Since the conceptual message of Theorem 1 significantly differs from previous studies on adversarial attacks in bandit algorithms, we would like to elaborate on its implications.

First of all, we, for the first time, point out some learning environment is intrinsically robust. Even the vanilla LinUCB algorithm, as we will analyze in the proof of Theorem 1, cannot be efficiently attacked when CQP (2) is not satisfied. Notably, although almost all previous works have focused on the vulnerability of bandit algorithms, e.g., by designing attacks against UCB, ϵ-Greedy (Jun et al., 2018), LinUCB (Garcelon et al., 2020), it just so happens that they were already studied under an attackable environment (see our Corollary 2). To our best knowledge, the problem about the intrinsic robustness of a linear bandit environment has not been studied before and can be viewed as a complement to these previous works. Second, as we will show next, our proof techniques are also significantly different from existing ones, since what is central to our proof is to demonstrate that when CQP (2) is not satisfied, there will exist a robust algorithm that cannot be efficiently attacked by *any* adversary. This can be viewed as analyzing the robustness of certain bandit algorithms when $\epsilon^* \leq 0$ in CQP (2).

Since CQP (2) and its solutions will show up very often in our later analysis, we provide the following definition for reference convenience.

Definition 2 (Attackability Index and Certificate)**.** *The optimal objective ϵ^* of CQP (2) is called the* attackability index *and the optimal solution $\tilde{\boldsymbol{\theta}}_{\perp}$ to CQP (2) is called the* attackability certificate.[4]

We should note both the index ϵ^* and certificate $\tilde{\boldsymbol{\theta}}_{\perp}$ are intrinsic to the bandit environment $\langle \mathcal{A} = \{x_i\}_{i=1}^k, \boldsymbol{\theta}^*, \tilde{x}\rangle$, and are irrelevant to any bandit algorithms used. As we will see in the next section when designing attack algorithms *without* the knowledge of $\boldsymbol{\theta}^*$, the index ϵ^* will determine how difficult it is to attack the environment.

Proof Sketch of Theorem 1. **Proof of sufficiency.** This direction is relatively straightforward. Suppose the attackability index $\epsilon^* > 0$ and let $\tilde{\boldsymbol{\theta}}_{\perp}$ be a certificate. We design the **oracle null space attack** based on the knowledge of bandit parameter $\boldsymbol{\theta}^*$. Let target parameter $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta}_{\parallel}^* + \tilde{\boldsymbol{\theta}}_{\perp}$ where $\boldsymbol{\theta}_{\parallel}^*$ is defined in Eq (1). The adversary perturbs the reward of any non-target arm $x_a \neq \tilde{x}$ by $\tilde{r}_{a,t} = x_a^{\mathsf{T}}\tilde{\boldsymbol{\theta}} + \tilde{\eta}_t$, whereas the reward of the target arm $\tilde{x}$ is *not* touched. To make attack appear less "suspicious", a sub-Gaussian noise $\tilde{\eta}_t$ is added to the perturbed reward to make it stochastic. The first constraint in CQP (2) ensures the non-target arms' rewards are smaller than the target arm's, and thus any no-regret bandit algorithm will only pull the non-target arms $o(T)$ times. The resulting cost is also $o(T)$ since the expected cost on each attack is bounded by a constant. Hence, such an attack is successful. Importantly, we note that this argument only relies on the definition of "no regret" but does not depend on what the algorithm is. This is crucial for proving the sufficiency of attackability.

Proof of necessity. This is the more difficult direction. We shall prove that if $\epsilon^* \leq 0$, the bandit environment is not attackable. To do so, we need to identify at least one no-regret bandit algorithm such that no attack strategy can successfully attack it. We argue that even the vanilla LinUCB is already robust to any attack strategy with $o(T)$ cost when $\epsilon^* \leq 0$. Recall that LinUCB maintains an estimate $\hat{\boldsymbol{\theta}}_t$ at round t using the attacked rewards $\{\tilde{r}_{1:t}\}$. We consider LinUCB with the choice of L2-regularization parameter λ that guarantees $\|\hat{\theta}_t\|_2 < 1$ in order to satisfy the last constraint in CQP (2). Consider the decomposition $\hat{\boldsymbol{\theta}}_t = \hat{\boldsymbol{\theta}}_{t,\parallel} + \hat{\boldsymbol{\theta}}_{t,\perp}$,

[4] We sometimes omit "attackability" when it is clear from the context, and simply mention *index* and *certificate*.

where $\tilde{x} \perp \hat{\boldsymbol{\theta}}_{t,\perp}$ and $\tilde{x} \parallel \hat{\boldsymbol{\theta}}_{t,\parallel}$.

Suppose, for the sake of contradiction, that LinUCB is attackable when $\epsilon^* \leq 0$. According to Definition 1, the target arm $\tilde{x}$ will be pulled $T - o(T)$ times with high probability for infinitely many different time horizons T. Fix any large T; we know that $\tilde{x}$ must have the largest UCB score whenever it is pulled at some round $t \in [T]$, or formally, for any $x_a \neq \tilde{x}$ we must have the following:

$$\tilde{x}^\mathsf{T}\hat{\boldsymbol{\theta}}_{t,\parallel} + \mathrm{CB}_t(\tilde{x}) \geq x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{t,\parallel} + x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{t,\perp} + \mathrm{CB}_t(x_a). \quad (3)$$

By attackability, we know that the above inequality will hold for infinitely many ts. Our main idea to construct the proof is that as $t \to \infty$, we have $\mathrm{CB}_t(\tilde{x}) \to 0$ and $\mathrm{CB}_t(x_a) > 0$. Moreover, the estimation of $\hat{\boldsymbol{\theta}}_{t,\parallel}$ will converge to $\boldsymbol{\theta}^*_\parallel$, since $\tilde{x}$ will be pulled for $t - o(t)$ times. The key challenge is to show $\mathrm{CB}_t(x_a) - \mathrm{CB}_t(\tilde{x})$, due to Inequality (3), is *strictly* greater than 0 for all large t. To do so, we prove a $\Theta\left(\sqrt{\frac{\log(t/\delta)}{o(t)}}\right)$ *lower* bound for $\mathrm{CB}_t(x_a)$ (Lemma 3 in Appendix B) and an $O(\sqrt{\frac{\log(t/\delta)}{t}})$ *upper* bound for $\mathrm{CB}_t(\tilde{x})$ (Lemma 2). The main technical barrier we overcome is the lower bound proof for the confidence bound term, which employs non-standard arguments since most (if not all) of the bandit algorithm analysis only needs the upper bound of the confidence bound terms. Due to this reason, we believe this technical proof is of independent interest, particularly for the analysis of robust properties of linear bandit algorithms.

By letting $t \to \infty$, we obtain the following condition:

$$\tilde{x}^\mathsf{T}\boldsymbol{\theta}^*_\parallel > x_a^\mathsf{T}\boldsymbol{\theta}^*_\parallel + x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{t,\perp}, \forall x_a \neq \tilde{x}. \quad (4)$$

This implies that for any sufficiently large t, there must exist a $\hat{\boldsymbol{\theta}}_{t,\perp}$ that and makes the optimal objective of CQP (2) ϵ^* positive. But this contradicts the starting assumption of $\epsilon^* \leq 0$; hence, the bandit environment is not attackable. □

We now provide an intuitive explanation about Theorem 1. CQP (2) is to find $\tilde{\boldsymbol{\theta}}_\perp$ such that: 1) it is orthogonal to $\tilde{x}$ (hence its subscript); and 2) it maximizes the gap ϵ between $\tilde{x}^\mathsf{T}\boldsymbol{\theta}^*_\parallel$ and the largest $x_a^\mathsf{T}(\boldsymbol{\theta}^*_\parallel + \tilde{\boldsymbol{\theta}}_\perp)$ among all $x_a \neq \tilde{x}$. Theorem 1 states that the bandit environment is attackable *if and only if* such a gap is strictly larger than 0, i.e., when the *geometry of context vectors* allows the adversary to lower non-target arms' rewards in the null space of $\tilde{x}$. The constraint $\|\boldsymbol{\theta}^*_\parallel + \tilde{\boldsymbol{\theta}}_\perp\|_2 \leq 1$ ensures the attacked rewards are in the same scale as the unattacked rewards.

Recent works have shown that any no-regret algorithm for the context-free k-armed setting (where arm set $\mathcal{A}$ is orthonormal) can always be attacked (Liu & Shroff, 2019). This finding turns out to be a corollary of Theorem 1.

Corollary 2. *For standard stochastic multi-armed bandit setting where arm set $\mathcal{A}$ is orthonormal, the environment $\langle \mathcal{A} = \{x_a\}, \boldsymbol{\theta}^*, \tilde{x}\rangle$ is attackable for any target arm $\tilde{x}$.*

Proof. Since arms are orthogonal to each other, it is easy to see that $\tilde{\boldsymbol{\theta}}_\perp = -C[\sum_{x_a : x_a \neq \tilde{x}} x_a]$ achieves objective value $C - \tilde{x}^T\boldsymbol{\theta}^*_\parallel$ in CQP (2). Let C be a large enough positive constant such that the objective value is positive gives us a feasible $\tilde{\boldsymbol{\theta}}_\perp$ to CQP (2), which yields the corollary. □

The intuition behind this corollary is that arms in context-free stochastic multi-armed bandits are independent, and an adversary can arbitrarily lower the rewards of non-target arms to make the target arm optimal. This is also the attack strategy in (Jun et al., 2018; Liu & Shroff, 2019). Garcelon et al. (2020) showed that similar idea works for k-armed linear contextual bandits where each arm is associated with an unknown bandit parameter and the reward estimations are independent among different arms.

We should point an important distinction between poisoning attacks to k-armed bandits and another line of research on *stochastic bandits under adversarial corruption* initiated by Lykouris et al. (2018). For poisoning attacks considered in this paper, the adversary manipulates the realized rewards *after* the algorithm selects an action, whereas in (Lykouris et al., 2018), the adversary manipulates the entire reward vector *before* the algorithm selects any action. Obviously, the later threat model is strictly weaker and has led to various bandit algorithms that can have sublinear regret so long as the total manipulation is sublinear in T (Lykouris et al., 2018; Zimmert & Seldin, 2021).

4. Effective Attacks without Knowledge of True Model Parameters

In the previous section, we characterized the attackability of a linear stochastic bandit environment by assuming the knowledge of ground-truth bandit parameter $\boldsymbol{\theta}^*$. We now show that such prior knowledge is not needed when designing practical attacks. Towards this end, we propose provably effective attacks against two representative bandit algorithms: the most well-known LinUCB and a state-of-the-art robust linear stochastic bandit algorithm, Robust Phase Elimination (Bogunovic et al., 2021). We remark that the optimal attacks to these algorithms depend on the characteristics of algorithms themselves and are generally different, due to their different levels of robustness. This also resonates the important message mentioned in the introduction, i.e., the attackability analysis often goes hand-in-hand with the understanding of robustness of different algorithms, as reflected in various parts of our analysis. However, we point out that it is an intriguing open question to understand whether there is a single attack strategy that can manipulate any no-regret algorithm in an attackable environment.

Two-stage Null Space Attack. Our proposed attack method is presented in Algorithm 1. The method spends the first T_1 rounds as the first stage to attack rewards on all arms by imitating a bandit environment $\boldsymbol{\theta}_0$, in which $\tilde{x}$ is the best arm such that arm $\tilde{x}$ will be pulled most often by the bandit algorithm. This stage is for the adversary to observe the true rewards of $\tilde{x}$ from the environment, which helps it estimate the parameter $\boldsymbol{\theta}^*_{\parallel}$. At round T_1, the method uses the estimate of $\boldsymbol{\theta}^*_{\parallel}$, denoted as $\tilde{\boldsymbol{\theta}}_{\parallel}$, to perform the "attackability test" by solving CQP (2) using $\tilde{\boldsymbol{\theta}}_{\parallel}$ to obtain an estimated index $\tilde{\epsilon}^*$ and certificate $\tilde{\boldsymbol{\theta}}_{\perp}$. If $\tilde{\epsilon}^* > 0$, the method asserts the environment is attackable and starts the second stage of attack. From T_1 to T, the method perturbs the reward of *non-target arms* by $\tilde{r}_t = x_{a_t}^{\mathsf{T}}(\tilde{\boldsymbol{\theta}}_{\parallel} + \tilde{\boldsymbol{\theta}}_{\perp}) + \tilde{\eta}_t$ (just like the oracle attack but using the estimated $\tilde{\boldsymbol{\theta}}_{\parallel}$). When the bandit algorithm pulls the target arm $\tilde{x}$ for the first time in the second stage, the method will compensate the reward of $\tilde{x}$ as shown in line 20, where $n(\tilde{x})$ is the number of times target arm is pulled before T_1. The goal is to correct the rewards on $\tilde{x}$ collected in the first stage to follow $\tilde{\boldsymbol{\theta}}$ instead of $\boldsymbol{\theta}_0$. Note that a larger T_1 brings in more observations on $\tilde{x}$ to improve the estimate of $\boldsymbol{\theta}^*_{\parallel}$; but it also means a higher attack cost. The optimal choice of T_1 depends on the "robustness" of the bandit algorithm to be attacked. Consequently, it also leads to different attack cost for different algorithms. For example, as we will show next, the attack to Robust Phase Elimination will be more costly than the attack to LinUCB.

Note that our attackability test might make both false positive and false negative assertions due to the estimation error in $\tilde{\boldsymbol{\theta}}_{\parallel}$. But as T becomes larger, the estimate gets more accurate and the assertion is correct with high probability.

Remark 1. *We acknowledge that the rewards from the two stages follow different reward distributions and could be detected, e.g., using some homogeneity test (Li et al., 2021). Thus a bandit player could realize the attack if equipped with some change detector. However, attacking such a cautious bandit algorithm is beyond the scope of this paper. Moreover, it is very difficult (if not impossible) to attack with a stationary reward distribution or undetectable perturbations (e.g., using a fixed target parameter $\tilde{\theta}$). We could easily find cases where the adversary's parameter $\tilde{\theta}$ is limited to extremely few choices and it is almost impossible to directly start the attack with a valid $\tilde{\theta}$ without knowing θ^*. For example, if we change the third arm in Example 1 to $x_3 = (-1+\epsilon, 0)$ with a small ϵ, we can see that the valid parameters are only in a small range around $\tilde{\theta} = (-1-\epsilon, 1)$. Therefore, in order to attack with a stationary reward distribution, the adversary needs to start from somewhere very close to $\tilde{\theta} = (-1-\epsilon, 1)$, which we believe is extremely difficult without knowing θ^*. Overall, we think designing an attack strategy against a bandit algorithm with reward change detector or showing the inability to attack such cautious algorithms is an important future work of ours.*

Algorithm 1 Two-stage Null Space Attack

```
1: **Inputs:** $T, T_1$
2: $\boldsymbol{\theta}_0 = \arg\max_{\|\boldsymbol{\theta}\|_2 \le 1} \left[\tilde{x}^{\mathsf{T}}\boldsymbol{\theta} - \max_{x_a \neq \tilde{x}} x_a^{\mathsf{T}}\boldsymbol{\theta}\right]$, let $\epsilon_0^*$ be its optimal objective
3: **if** $\epsilon_0^* \le 0$ **then**                // Initial attackability test
4:     **return** Not attackable
5: **end if**
6: **for** $t = 1$ to $T_1$ **do**
7:     Set $\tilde{r}_t = x_{a_t}^{\mathsf{T}}\boldsymbol{\theta}_0 + \tilde{\eta}_t$        // Attack as if $\tilde{x}$ is the best
8:     Bandit algorithm observes modified reward $\tilde{r}_t$
9: **end for**
10: Estimate $\tilde{\boldsymbol{\theta}}_{\parallel} = \frac{\sum_{i=1}^{n(\tilde{x})} r_i(\tilde{x})}{n(\tilde{x})\|\tilde{x}\|_2^2}\tilde{x}$
11: Solve CQP (2) using $\tilde{\boldsymbol{\theta}}_{\parallel}$ to obtain the estimated attackability index $\tilde{\epsilon}^*$ and certificate $\tilde{\boldsymbol{\theta}}_{\perp}$
12: **if** $\tilde{\epsilon}^* \le 0$ **then**                // Attackability test
13:     **return** Not attackable
14: **else**                                   // Attack stage
15:     Set $\tilde{\boldsymbol{\theta}} = \tilde{\boldsymbol{\theta}}_{\parallel} + \tilde{\boldsymbol{\theta}}_{\perp}$
16:     **for** $t = T_1 + 1$ to $T$ **do**
17:         **if** $x_{a_t} \neq \tilde{x}$ **then**
18:             Set $\tilde{r}_t = x_{a_t}^{\mathsf{T}}\tilde{\boldsymbol{\theta}} + \tilde{\eta}_t$
19:         **else if** $x_{a_t} = \tilde{x}$ for the first time **then**
20:             Set $\tilde{r}_t = n(\tilde{x}) \times \tilde{x}^{\mathsf{T}}(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}_0) + \tilde{x}^{\mathsf{T}}\tilde{\boldsymbol{\theta}} + \tilde{\eta}_t$
21:         **else**
22:             Set $\tilde{r}_t = r_t$
23:         **end if**
24:         Bandit algorithm observes modified reward $\tilde{r}_t$
25:     **end for**
26: **end if**
```

Attack against LinUCB. We now show how LinUCB algorithm can be attacked by Algorithm 1.

Theorem 3. *For large enough T_1, the attack strategy in Algorithm 1 will correctly assert the attackability with probability at least $1-\delta$. Moreover, when the environment is attackable, with probability at least $1-3\delta$ the attack strategy will fool LinUCB to pull non-target arms at most*

$$O\Big(d\big(\sqrt{\log(T/\delta)} + \sqrt{T_1}\log(T_1/\delta) + \sqrt{T\log(1/\delta)}/\sqrt{T_1}\big)\sqrt{T\log(T/\delta)}/\epsilon^*\Big)$$

rounds. And with probability at least $1-4\delta$, the adversary's cost is at most

$$O\Big(T_1 + d\big(\sqrt{\log(T/\delta)} + \sqrt{T_1}\log(T_1/\delta) + \sqrt{T\log(1/\delta)}/\sqrt{T_1}\big)\sqrt{T\log(T/\delta)}/\epsilon^*\Big).$$

Specifically, when $T_1 = T^{1/2}$, the strategy gives the minimum attack cost in the order of $\tilde{O}(T^{3/4})$, and the non-target arms are pulled at most $\tilde{O}(T^{3/4})$ rounds.

Proof Sketch. To prove the the assertion is correct with high probability, the key idea is that the estimated $\tilde{\boldsymbol{\theta}}_{\parallel}$ is close to the true parameter $\boldsymbol{\theta}^*_{\parallel}$. We first note that in the first stage, the bandit algorithm will pull the target arm $\tilde{x}$ $T_1 - O(\sqrt{T_1})$ times, since $\tilde{x}$ is the best arm according to $\boldsymbol{\theta}_0$. According to the Hoeffding's inequality, the estimation error $\|\tilde{\boldsymbol{\theta}}_{\parallel} - \boldsymbol{\theta}^*_{\parallel}\|_2 \leq \sqrt{\frac{2\log(2/\delta)}{T_1 - O(\sqrt{T_1})}}$. Therefore, with a large enough T_1, the error on $\tilde{x}$'s reward estimation is smaller than ϵ^*. Thus solving CQP (2) with $\tilde{\boldsymbol{\theta}}_{\parallel}$ and we can correctly assert attackability with positive estimated index $\tilde{\epsilon}^*$ when the environment is attackable with index ϵ^*.

To prove the success and the cost of the attack, we need to analyze the behavior of LinUCB under the reward discrepancy between the two stages. Our proof crucially hinges on the following robustness result of LinUCB.

Lemma 1. *Consider LinUCB with ridge regression under poisoning attack. Let $S'_t = \sum_{\tau \in \{1\ldots t\}, x_{a_\tau} \neq \tilde{x}} |\Delta_\tau|$ be the total corruption on non-target arms until time t and assume every corruption on target arm is bounded by γ. For any $t = 1 \ldots T$, with probability at least $1 - \delta$ we have*

$$\|\tilde{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_t\|_{A_t} \leq \alpha_t + S'_t/\sqrt{\lambda} + \gamma\sqrt{t} \tag{5}$$

where $\alpha_t = \sqrt{d \log\left(\frac{1+t/\lambda}{\delta}\right)} + \sqrt{\lambda}$.

Based on this lemma, we can derive the regret $R_T(\tilde{\boldsymbol{\theta}})$ of LinUCB with $\tilde{\boldsymbol{\theta}}$ as the true parameter. The total corruption on non-target arms is $O\big(d\sqrt{T_1}\log(T_1/\delta)\big)$ given the rewards are generated by $\boldsymbol{\theta}_0$ (the rewards of target arm in the first stage are compensated in line 20). Because the target arm's rewards are not attacked in the second stage and follows $\boldsymbol{\theta}^*$, we have $\gamma = \tilde{O}(1/\sqrt{T_1})$. Since the non-target arms are pulled at most $R_T(\tilde{\boldsymbol{\theta}})/\epsilon^*$ rounds, substitute the total corruption back and we have the resulting bound.

The attack cost has two sources: attacks in the first stage for T_1 times, and attacks on the non-target arms in the second stage. The second term has the same order as the number of rounds where the non-target arms are pulled by LinUCB. Each attack cost can be decomposed as 1) the change of mean reward $|x_a^\mathsf{T}(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*)|$, and 2) the sub-Gaussian noise $|\tilde{\eta}_t|$, the sum of which increases linearly with high probability. By setting $T_1 = T^{1/2}$, the total cost achieves $\tilde{O}(T^{3/4})$. □

Remark 2. *Lemma 1 shows that LinUCB still enjoys sub-linear regret for any corruption amount $S = o(\sqrt{T})$. This tolerance of $o(\sqrt{T})$ attack turns out to be the same as the recently proposed robust linear contextual bandit algorithm based on phase elimination in (Bogunovic et al., 2021) (which we examine next). However, the regret term $S\sqrt{T}$ in LinUCB has a worse dependence on S within the $S = o(\sqrt{T})$ regime compared to the $O(S^2)$ regret dependence of the robust algorithm in (Bogunovic et al., 2021).*

Attack against Robust Phase Elimination. We now show that Robust Phase Elimination (RobustPhE) (Bogunovic et al., 2021) can also be attacked by Algorithm 1. Comparing to attacking LinUCB, the robustness of RobustPhE brings challenge to the first stage attack, as the attack cost is more sensitive to the length of this stage.

Corollary 4. *For any large enough T_1, the attack strategy in Algorithm 1 will correctly assert the attackability with high probability. Moreover, when the environment is attackable, with probability at least $1 - 2\delta$ the attack strategy will fool RobustPhE to pull non-target arms at most*

$$O\Big(\big(d\sqrt{T}\log(T/\delta) + \sqrt{d}T\log(T)\log(1/\delta)/\sqrt{T_1} + T_1^2\big)/\epsilon^*\Big)$$

rounds. And with probability at least $1 - 3\delta$, the adversary's cost is at most

$$O\Big(T_1 + \big(d\sqrt{T}\log(T/\delta) + \sqrt{d}T\log(T)\log(1/\delta)/\sqrt{T_1} + T_1^2\big)/\epsilon^*\Big)$$

Specifically, setting $T_1 = T^{2/5}$ gives the minimum attack cost order $\tilde{O}(T^{4/5})$ and the non-target arms are pulled at most $\tilde{O}(T^{4/5})$ rounds.

RobustPhE has an additional regret term $O(S^2)$ for total corruption S (assuming S is unknown to the bandit algorithm). If we view the second stage attack model $\tilde{\boldsymbol{\theta}}$ as the underlying environment bandit model, rewards generated by $\boldsymbol{\theta}_0$ in the first stage should be considered as corrupted rewards. The T_1 amount of rewards from the first stage means T_1 amount of corruption, which leads to the additional T_1^2 term compared with the results in Theorem 3. Hence, the adversary can only run fewer iterations in the first stage but spend more attack cost there. On the other hand, this also facilitates the design of attack such that line 19-20 in Algorithm 1 is not necessary: the corruption in the first stage can be handled by the robustness of RobustPhE. The unattacked rewards in second stage are viewed as misspecification from $\tilde{\boldsymbol{\theta}}$ with error γ, which leads to the $\tilde{O}(\gamma T)$ term (the second term) in the bound. Our success in attacking RobustPhE does not violate the robustness claim in the original paper (Bogunovic et al., 2021): RobustPhE could tolerate $o(\sqrt{T})$ corruption and our attack cost is $\tilde{O}(T^{4/5})$.

5. Experiments

We use simulation-based experiments to validate the effectiveness and cost-efficiency of our proposed attack methods. In our simulations, we generate a size-k arm pool $\mathcal{A}$, in which each arm a is associated with a context vector $x_a \in \mathbb{R}^d$. Each dimension of x_a is drawn from a set of zero-mean Gaussian distributions with variances sampled from a uniform distribution $U(0, 1)$. Each x_a is then normalized

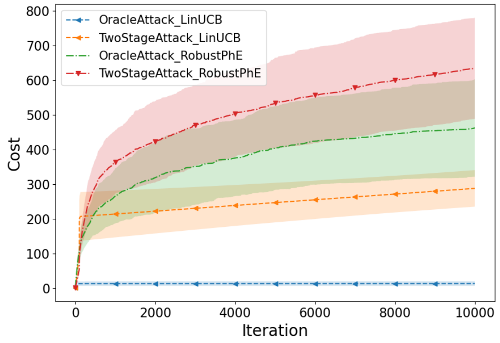

Figure 2. Total cost of attack under different attack methods. We report averaged cost and its variance of 10 runs.

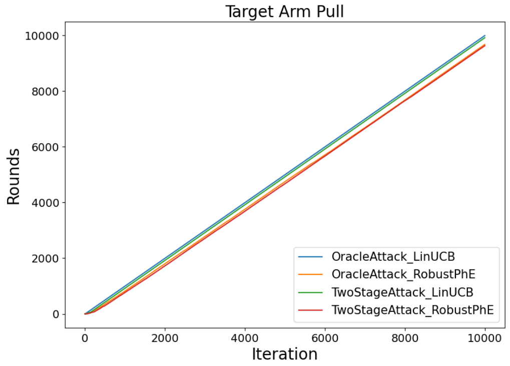

Figure 3. Target arm pulls under different attack methods.

to $\|x_a\|_2 = 1$. The bandit model parameter $\boldsymbol{\theta}^*$ is sampled from $N(0, 1)$ and normalized to $\|\boldsymbol{\theta}^*\|_2 = 1$. We set d to 10, the standard derivation σ of Gaussian noise η_t and $\tilde{\eta}_t$ to 0.1, and the arm pool size k to 30 in our simulations. We run the experiment for $T = 10,000$ iterations. To create an attackable environment, we will re-sample the environment $\langle \mathcal{A}, \boldsymbol{\theta}^*, \tilde{x} \rangle$ until it is attackable, following Theorem 1.

We compared the two proposed attack methods, oracle null space attack and two-stage null space attack, against LinUCB (Li et al., 2010) and Robust Phase Elimination (RobustPhE) (Bogunovic et al., 2021). We report average results of 10 runs where in each run we randomly sample an attackable environment. Both oracle attack and two-stage attack can effectively fool the two bandit algorithms to pull the target arm linear times as shown in Figure 3, and the total cost of the attack is shown in Figure 2. We observe that both attack methods are cost-efficient with a sublinear total cost, while the two-stage attack requires higher attack cost when attacking the same bandit algorithm. Specifically, we notice that the adversary spends almost *linear* cost in the first stage. This is because in the first stage the adversary attacks according to parameter $\boldsymbol{\theta}_0$, which leads to an almost constant cost at every round. This is to help the adversary to estimate bandit model parameter in order to construct target parameter $\tilde{\boldsymbol{\theta}}$. In the second stage, the cost gets much smaller since the adversary only attacks the non-target arms. We also notice that for the same attack method, attacking RobustPhE requires a higher cost and the number of target arm pull is also smaller comparing with attacking LinUCB, due to the robustness of the algorithm.

6. Related Work

Adversarial attacks to bandit algorithms was first studied in the stochastic multi-armed bandit setting (Jun et al., 2018; Liu & Shroff, 2019) and recently in linear contextual bandits (Garcelon et al., 2020). These works share a similar attack idea: lowering the rewards of non-target arms while not modifying the reward of target arm. However, as our attackability analysis revealed, this idea can fail in a linear stochastic bandit environment where one cannot lower the rewards of non-target arms without modifying the reward of target arm, due to their correlation. This insight is a key reason that gives rise to unattackable environments. Ma et al. (2018) also considered the attackability issue of linear bandits, but under the setting of *offline* data poisoning attack where the adversary has the power to modify the rewards in history. There are also several recent works on reward poisoning attacks against reinforcement learning (Yu & Sra, 2019; Zhang et al., 2020; Rakhsha et al., 2021; 2020), but with quite different focus as ours. Besides reward poisoning attacks, recent works also studied other threat model such as action poisoning attacks (Liu & Lai, 2020; 2021).

A parallel line of works focused on improving the robustness of bandit algorithms. Lykouris et al. (2018) proposed a robust MAB algorithm and Gupta et al. (2019) further improved the solution with additive regret dependency on attack cost. Zimmert & Seldin (2021); Masoudian & Seldin (2021) proposed best-of-both-world solutions for both stochastic and adversarial bandits which also solved stochastic bandits with adversarial corruption. Ito (2021) further proposed optimal robust algorithm to adversarial corruption. These work assumed a weaker oblivious adversary who determines the manipulation before the bandit algorithm pulls an arm. Hajiesmaili et al. (2020) studied robust adversarial bandit algorithm. Guan et al. (2020) proposed a median-based robust bandit algorithm for probabilistic unbounded attack. Bogunovic et al. (2021) proposed robust phase elimination algorithm for linear stochastic bandits under a stronger adversary (same as ours), which could tolerate $o(\sqrt{T})$ corruption when the total corruption is unknown to the algorithm. We showed that our two-stage null space attack could effectively attack this algorithm with $\tilde{O}(T^{4/5})$ cost. Recently, Zhao et al. (2021) proposed an OFUL style robust algorithm that can handle infinite action set, but only tolerates $o(T^{1/4})$ corruption.

7. Conclusion

In this paper, we studied the problem of poisoning attacks in k-armed linear stochastic bandits. Different from context-free stochastic bandits and k-armed linear contextual bandits where the environment is always attackable, we showed that some linear stochastic bandit environments are *not* attackable due to the correlation among arms. We characterized the attackability condition as the feasibility of a CQP based on the geometry of the arms' context vectors. Our key insight is that given the ground-truth parameter $\boldsymbol{\theta}^*$, the adversary should perform oracle attack that lowers the reward of non-target arms in the null space of the target arm's convex vector $\tilde{x}$. Based on this insight, we proposed a two-stage null space attack without the knowledge of $\boldsymbol{\theta}^*$ and applied it against LinUCB and Robust Phase Elimination. We showed that the proposed attack methods are effective and cost-efficient, both theoretically and empirically.

As our future work, it is interesting to study the lower bound of attack cost in linear stochastic bandits and also design cost-optimal attack method with a matching upper bound. One idea is to design a multi-stage attack method following the doubling trick idea, which was brief discussed in Appendix C.3. Although the analysis could be much more challenging than our two-stage attack, it may lead to a lower attack cost as well as handling unknown time horizon T. Another intriguing direction is to study algorithm-oblivious choice of the length of the first stage T_1 — or more generally, algorithm-oblivious attack strategies — that can achieve sublinear cost for *arbitrary* no-regret algorithm without the knowledge of $\boldsymbol{\theta}^*$.

Acknowledgements

This work is based upon work supported by National Science Foundation under grant IIS-2128019, IIS-2007492 and IIS-1838615, Bloomberg Data Science Ph.D. Fellowship, and a Google Faculty Research Award.

References

Abbasi-yadkori, Y., Pál, D., and Szepesvári, C. Improved algorithms for linear stochastic bandits. In *NIPS*, pp. 2312–2320. 2011.

Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3:397–422, 2002.

Bogunovic, I., Losalka, A., Krause, A., and Scarlett, J. Stochastic linear bandits robust to adversarial attacks. In *International Conference on Artificial Intelligence and Statistics*, pp. 991–999. PMLR, 2021.

Chapelle, O. and Li, L. An empirical evaluation of thompson sampling. In *Advances in neural information processing systems*, pp. 2249–2257, 2011.

Durand, A., Achilleos, C., Iacovides, D., Strati, K., Mitsis, G. D., and Pineau, J. Contextual bandits for adapting treatment in a mouse model of de novo carcinogenesis. In *Machine Learning for Healthcare Conference*, pp. 67–82, 2018.

Garcelon, E., Roziere, B., Meunier, L., Tarbouriech, J., Teytaud, O., Lazaric, A., and Pirotta, M. Adversarial attacks on linear contextual bandits. *Advances in Neural Information Processing Systems*, 33, 2020.

Guan, Z., Ji, K., Bucci Jr, D. J., Hu, T. Y., Palombo, J., Liston, M., and Liang, Y. Robust stochastic bandit algorithms under probabilistic unbounded adversarial attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 4036–4043, 2020.

Gupta, A., Koren, T., and Talwar, K. Better algorithms for stochastic bandits with adversarial corruptions. In *Conference on Learning Theory*, pp. 1562–1578. PMLR, 2019.

Hajiesmaili, M., Talebi, M. S., Lui, J., Wong, W. S., et al. Adversarial bandits with corruptions: Regret lower bound and no-regret algorithm. *Advances in Neural Information Processing Systems*, 33, 2020.

Ito, S. On optimal robustness to adversarial corruption in online decision problems. *Advances in Neural Information Processing Systems*, 34:7409–7420, 2021.

Jun, K.-S., Li, L., Ma, Y., and Zhu, J. Adversarial attacks on stochastic bandits. In *Advances in Neural Information Processing Systems*, pp. 3640–3649, 2018.

Lattimore, T., Szepesvari, C., and Weisz, G. Learning with good feature representations in bandits and in rl with a generative model. In *International Conference on Machine Learning*, pp. 5662–5670. PMLR, 2020.

Li, C., Wu, Q., and Wang, H. Unifying clustered and non-stationary bandits. In *International Conference on Artificial Intelligence and Statistics*, pp. 1063–1071. PMLR, 2021.

Li, L., Chu, W., Langford, J., and Schapire, R. E. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pp. 661–670. ACM, 2010.

Liu, F. and Shroff, N. Data poisoning attacks on stochastic bandits. In *International Conference on Machine Learning*, pp. 4042–4050, 2019.

Liu, G. and Lai, L. Action-manipulation attacks on stochastic bandits. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3112–3116. IEEE, 2020.

Liu, G. and Lai, L. Provably efficient black-box action poisoning attacks against reinforcement learning. *Advances in Neural Information Processing Systems*, 34, 2021.

Lykouris, T., Mirrokni, V., and Paes Leme, R. Stochastic bandits robust to adversarial corruptions. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 114–122. ACM, 2018.

Ma, Y., Jun, K.-S., Li, L., and Zhu, X. Data poisoning attacks in contextual bandits. In *International Conference on Decision and Game Theory for Security*, pp. 186–204. Springer, 2018.

Masoudian, S. and Seldin, Y. Improved analysis of the tsallis-inf algorithm in stochastically constrained adversarial bandits and stochastic bandits with adversarial corruptions. In *Conference on Learning Theory*, pp. 3330–3350. PMLR, 2021.

Rakhsha, A., Radanovic, G., Devidze, R., Zhu, X., and Singla, A. Policy teaching via environment poisoning: Training-time adversarial attacks against reinforcement learning. In *International Conference on Machine Learning*, pp. 7974–7984. PMLR, 2020.

Rakhsha, A., Zhang, X., Zhu, X., and Singla, A. Reward poisoning in reinforcement learning: Attacks against unknown learners in unknown environments. *arXiv preprint arXiv:2102.08492*, 2021.

Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Yu, T. and Sra, S. Efficient policy learning for non-stationary mdps under adversarial manipulation. *arXiv preprint arXiv:1907.09350*, 2019.

Zhang, X., Ma, Y., Singla, A., and Zhu, X. Adaptive reward-poisoning attacks against reinforcement learning. *arXiv preprint arXiv:2003.12613*, 2020.

Zhao, H., Zhou, D., and Gu, Q. Linear contextual bandits with adversarial corruptions. *arXiv preprint arXiv:2110.12615*, 2021.

Zimmert, J. and Seldin, Y. Tsallis-inf: An optimal algorithm for stochastic and adversarial bandits. *J. Mach. Learn. Res.*, 22:28–1, 2021.

A. Notations

For clarity, we collect the notations used in the paper below.

$\tilde{x}$	Context vector of target arm
x_a	Context vector of arm a
$\boldsymbol{\theta}^*$	Unknown bandit model parameter of the environment
$\boldsymbol{\theta}^*_{\parallel}$	Projection of $\boldsymbol{\theta}^*$ on $\tilde{x}$
r_t	Unattacked reward feedback at time t
η_t	Sub-Gaussian noise in reward, i.e., $r_t = x_t^\mathsf{T}\boldsymbol{\theta}^* + \eta_t$.
$\tilde{r}_t$	Attacked reward
$\hat{\boldsymbol{\theta}}_t$	Parameter estimated by the victim bandit algorithm with attacked rewards $\{\tilde{r}_{1:t}\}$
$\tilde{\boldsymbol{\theta}}_{\parallel}$	Parameter parallel to $\tilde{x}$, estimated by adversary with unattacked rewards
$\tilde{\boldsymbol{\theta}}$	Paramter of adversary's attack strategy
$\tilde{\boldsymbol{\theta}}_{\perp}$	Attackability certificate, the parameter orthogonal to $\tilde{x}$ solved by CQP (2)
ϵ^*	Attackability index, optimal objective of CQP (2)
$\tilde{\epsilon}^*$	Estimated index, optimal objective of CQP (2) with $\tilde{\boldsymbol{\theta}}_{\parallel}$ replacing $\boldsymbol{\theta}^*_{\parallel}$

B. Details on Attackability of Linear Stochastic Bandits

We illustrate the details of oracle null space attack in Algorithm 2, which is constructed for the sufficiency proof of Theorem 1.

Algorithm 2 Oracle Null Space Attack

```
**Inputs:** $T, \boldsymbol{\theta}^*$
**Initialize:**
**if** Optimal objective $\epsilon^*$ of CQP (2) $> 0$ **then**                // Attackability Test
    Find the optimal solution $\tilde{\boldsymbol{\theta}}_{\perp}$
    Set $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta}^*_{\parallel} + \tilde{\boldsymbol{\theta}}_{\perp}$
**else**
    **return** Not attackable
**end if**
**for** $t = 1$ to $T$ **do**
    Bandit algorithm pulls arm $a_t$
    Attacker observes the corresponding reward $r_t = x_{a_t}^\mathsf{T}\boldsymbol{\theta}^* + \eta_t$ from the environment
    **if** $x_{a_t} \neq \tilde{x}$ **then**
        Set $\tilde{r}_t = x_{a_t}^\mathsf{T}\tilde{\boldsymbol{\theta}} + \tilde{\eta}_t$                // Attack
    **else**
        Set $\tilde{r}_t = r_t$
    **end if**
    Bandit algorithm observes modified reward $\tilde{r}_t$
**end for**
```

B.1. Sufficiency Proof of Theorem 1

Proof. For sufficiency proof, we show that there exists an efficient attack strategy if CQP (2) holds.

Suppose the attackability index $\epsilon^* > 0$ and let $\tilde{\boldsymbol{\theta}}_{\perp}$ be a certificate. In Algorithm 2, we design the **oracle null space attack** based on the knowledge of bandit parameter $\boldsymbol{\theta}^*$. Let $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta}^*_{\parallel} + \tilde{\boldsymbol{\theta}}_{\perp}$ where $\boldsymbol{\theta}^*_{\parallel}$ is defined in Eq (1). The adversary perturbs the reward of any non-target arm $x_a \neq \tilde{x}$ by $\tilde{r}_{a,t} = x_a^\mathsf{T}\tilde{\boldsymbol{\theta}} + \tilde{\eta}_t$, whereas the reward of the target arm $\tilde{x}$ is *not* touched. In other words, the adversary misleads the algorithm to believe that $\tilde{\boldsymbol{\theta}}$ is the ground-truth parameter We should note both $\tilde{\boldsymbol{\theta}}$ and $\boldsymbol{\theta}^*$ generate the same expected reward on $\tilde{x}$, i.e., $\tilde{x}^\mathsf{T}\tilde{\boldsymbol{\theta}} = \tilde{x}^\mathsf{T}\boldsymbol{\theta}^*_{\parallel} = \tilde{x}^\mathsf{T}\boldsymbol{\theta}^*$. To make the attack appear less "suspicious", a sub-Gaussian noise term $\tilde{\eta}_t$ is added to the perturbed reward to make it stochastic. The key idea is that the attacker does not need to perturb the reward of the target arm because the original reward is the same as perturbed reward in expectation.

Instead, the attacker only perturbs the reward in the *null space* of $\tilde{x}$ according to $\tilde{\boldsymbol{\theta}}_{\perp}$, which guarantees the cost-efficiency of the attack.

Since the perturbed rewards observed by the bandit algorithm are generated by $\tilde{\boldsymbol{\theta}}$, the target arm $\tilde{x}$ is the optimal arm in this environment due to the attackability index ϵ^* being strictly positive. According to the definition, any no-regret bandit algorithm will only pull suboptimal arms, i.e., the non-target arms, $o(T)$ times and pull target arm $T - o(T)$ times with high probability. Thus the attack is successful. Moreover, the cost of oracle attack is $o(T)$ because the attacker only perturbs rewards on the non-target arms for $o(T)$ times, and the cost on each attack is bounded by a constant (because of the finite norm of x_a and $\boldsymbol{\theta}^*$). □

B.2. Necessity Proof of Theorem 1

To prove its necessity, we will rely on the following results.

Lemma 2. *Suppose arm x is pulled n times till round t by LinUCB. Its confidence bound $CB_t(x)$ in LinUCB satisfies*

$$CB_t(x) \le \frac{\alpha_t}{\sqrt{n}}. \tag{6}$$

with probability at least $1-\delta$, where $\alpha_t = \sqrt{d \log\left(\frac{1+t/\lambda}{\delta}\right)} + \sqrt{\lambda}$. Furthermore, we have

$$CB_t(x) \le O\left(\sqrt{\frac{\log(t/\delta)}{n}}\right) \tag{7}$$

with probability at least $1-\delta$.

Proof. In (Abbasi-yadkori et al., 2011), the exploration bonus term is computed as $\text{CB}_t(x) = \alpha_t \|x\|_{\mathbf{A}_t^{-1}}$. Denote $\mathbf{A}'_t = n \times xx^\mathsf{T} + \lambda\mathbf{I}$. Since $\mathbf{A}_t = \sum_{i=1}^t x_{a_i} x_{a_i}^\mathsf{T} + \lambda\mathbf{I}$, we have $\mathbf{A}_t \succ \mathbf{A}'_t$. We can thus bound $\|x\|_{\mathbf{A}_t^{-1}}$ by

$$\|x\|_{\mathbf{A}_t^{-1}} \le \|x\|_{\mathbf{A}_t'^{-1}} \le \frac{1}{\sqrt{n}} \tag{8}$$

According to Theorem 2 in (Abbasi-yadkori et al., 2011),

$$\alpha_t = \sqrt{d \log\left(\frac{1+t/\lambda}{\delta}\right)} + \sqrt{\lambda} = \Theta(\sqrt{\log(t/\delta)}). \tag{9}$$

Combining α_t and (8) finishes the proof. □

Claim 1. *Target arm $\tilde{x}$ is pulled $n = T - o(T)$ times till round T. According to Lemma 2, we have*

$$CB_t(\tilde{x}) \le O\left(\sqrt{\frac{\log(T/\delta)}{T - o(T)}}\right) \tag{10}$$

Lemma 3. *Suppose arm x is pulled $t-m$ times till round t by LinUCB, and other arms are pulled m times in total. Confidence bound $CB_t(x_a)$ of any arm x_a that is not parallel to x satisfies*

$$CB_t(x_a) \ge \alpha_t \left(\frac{1}{\sqrt{m+\lambda}} - \frac{b}{\sqrt{t-m+\lambda}}\right) \tag{11}$$

with probability at least $1-\delta$, where $\alpha_t = \sqrt{d \log\left(\frac{1+t/\lambda}{\delta}\right)} + \sqrt{\lambda}S$ and constant $b = \frac{x_a^\mathsf{T} x}{x^\mathsf{T} x}$. Furthermore, we have

$$CB_t(x_a) \ge \Theta\left(\sqrt{\log(t/\delta)}\left(\frac{1}{\sqrt{m}} - \frac{1}{\sqrt{t-m}}\right)\right) \tag{12}$$

with probability at least $1-\delta$

Proof. Since $\text{CB}_t(x) = \alpha_t \|x\|_{\mathbf{A}_t^{-1}}$, we need to show a lower bound of $\|x_a\|_{\mathbf{A}_t^{-1}}$. Since $x_a \nparallel x$, we decompose $x_a = x_a^{\|} + x_a^{\perp}$, where $x_a^{\|} \parallel x$. By the reverse triangle inequality we have

$$\|x_a\|_{\mathbf{A}_t^{-1}} \geq \|x_a^{\perp}\|_{\mathbf{A}_t^{-1}} - \|x_a^{\|}\|_{\mathbf{A}_t^{-1}} \tag{13}$$

First we analyze the term $\|x_a^{\perp}\|_{\mathbf{A}_t^{-1}}$. Decompose $\mathbf{A}_t = (t-m) \times xx^\mathsf{T} + \sum_{i, x_{a_i} \neq x} x_{a_i} x_{a_i}^\mathsf{T} + \lambda \mathbf{I}$ and let $\mathbf{A}'_t = \sum_{i, x_{a_i} \neq x} x_{a_i} x_{a_i}^\mathsf{T} + \lambda \mathbf{I}$. Since $x_a^{\perp} \perp x$, we have

$$x_a^{\perp\mathsf{T}} \mathbf{A}_t x_a^{\perp} = x_a^{\perp\mathsf{T}} \mathbf{A}'_t x_a^{\perp}.$$

There are at most m terms in the summation of $\mathbf{A}'_t = \sum_{i, x_{a_i} \neq x} x_{a_i} x_{a_i}^\mathsf{T} + \lambda I$, thus

$$x_a^{\perp\mathsf{T}} \mathbf{A}'_t x_a^{\perp} \leq x_a^{\perp\mathsf{T}} \left(\frac{m}{\|x_a^{\perp}\|_2^2} \times x_a^{\perp} x_a^{\perp\mathsf{T}} + \lambda \mathbf{I} \right) x_a^{\perp} \leq m + \lambda$$

Then we have

$$\|x_a^{\perp}\|_{\mathbf{A}_t^{-1}} = \sqrt{x_a^{\perp\mathsf{T}} \mathbf{A}_t^{-1} x_a^{\perp}} = \sqrt{x_a^{\perp\mathsf{T}} \mathbf{A}_t'^{-1} x_a^{\perp}} = \frac{1}{\sqrt{x_a^{\perp\mathsf{T}} \mathbf{A}'_t x_a^{\perp}}} \geq \frac{1}{\sqrt{m+\lambda}} \tag{14}$$

Similar to Eq (8), we have

$$\|x_a^{\|}\|_{\mathbf{A}_t^{-1}} \leq \frac{\|x_a^{\|}\|_2}{\|x\|_2} \frac{1}{\sqrt{t-m+\lambda}} \tag{15}$$

Let constant $b = \frac{\|x_a^{\|}\|_2}{\|x\|_2} = \frac{x_a^\mathsf{T} x}{x^\mathsf{T} x}$. Substitute the terms and we have

$$\text{CB}_t(x) = \alpha_t \|x\|_{\mathbf{A}_t^{-1}} \geq \alpha_t \left(\|x_a^{\perp}\|_{\mathbf{A}_t^{-1}} - \|x_a^{\|}\|_{\mathbf{A}_t^{-1}} \right) \geq \alpha_t \left(\frac{1}{\sqrt{m+\lambda}} - \frac{b}{\sqrt{t-m+\lambda}} \right). \tag{16}$$

□

Claim 2. *Non-target arms are pulled $m = o(T)$ times till round T. According to Lemma 3, any arm $x_a \nparallel \tilde{x}$ satisfies*

$$\textit{CB}_t(x_a) \geq \Theta \left(\sqrt{\log(T/\delta)} \left(\frac{1}{\sqrt{o(T)}} - \frac{1}{\sqrt{T - o(T)}} \right) \right) \tag{17}$$

with probability at least $1 - \delta$.

Lemma 4. *Suppose the non-target arms $\{x_a \neq \tilde{x}\}$ are pulled $o(T)$ times, the arm $\tilde{x}$ is pulled $T - o(T)$ times, and the total manipulation is C_T. With probability at least $1 - \delta$, reward estimation error satisfies*

$$\left| x^\mathsf{T} \hat{\boldsymbol{\theta}}_{T,\|} - x^\mathsf{T} \boldsymbol{\theta}^*_{\|} \right| \leq \frac{C_T}{T - o(T)} + \frac{\alpha_t}{\sqrt{T - o(T)}}, \qquad \forall x \in \mathcal{A}. \tag{18}$$

Proof.

$$
\begin{aligned}
\|\hat{\boldsymbol{\theta}}_{T,\|} - \boldsymbol{\theta}^*_{\|}\|_2 &= \left\| \frac{\tilde{x}^\mathsf{T}(\hat{\boldsymbol{\theta}}_T - \boldsymbol{\theta}^*)}{\|\tilde{x}\|_2^2} \tilde{x} \right\|_2 \\
&= \frac{1}{\|\tilde{x}\|_2^2} \left\| \tilde{x}^\mathsf{T} \mathbf{A}_t^{-1} \left(\sum_{t=1}^T x_t \tilde{r}_t(x_t) - \mathbf{A}_t \boldsymbol{\theta}^* \right) \tilde{x} \right\|_2 \\
&= \frac{1}{\|\tilde{x}\|_2^2} \left\| \tilde{x}^\mathsf{T} \mathbf{A}_t^{-1} \left(\sum_{t=1}^T x_t (\tilde{r}_t(x_t) - x_t^\mathsf{T} \boldsymbol{\theta}^*) - \lambda \boldsymbol{\theta}^* \right) \tilde{x} \right\|_2 \\
&\le \frac{1}{\|\tilde{x}\|_2^2} \left\| \tilde{x}^\mathsf{T} \mathbf{A}_t^{-1} \left(\sum_{t=1}^T x_t \Delta_t + \sum_{t=1}^T x_t \eta_t - \lambda \boldsymbol{\theta}^* \right) \tilde{x} \right\|_2 \\
&\le \frac{1}{\|\tilde{x}\|_2^2} \left\| \tilde{x}^\mathsf{T} \mathbf{A}_t^{-1} \left(\sum_{t=1}^T x_t \Delta_t \right) \tilde{x} \right\|_2 + \frac{1}{\|\tilde{x}\|_2^2} \left\| \tilde{x}^\mathsf{T} \mathbf{A}_t^{-1/2} \mathbf{A}_t^{-1/2} \left(\sum_{t=1}^T x_t \eta_t - \lambda \boldsymbol{\theta}^* \right) \tilde{x} \right\|_2 \\
&\le \frac{C_T}{T - o(T)} + \frac{\alpha_t}{\sqrt{T - o(T)}}
\end{aligned}
$$

Note that $\|\tilde{x}\|_2 \le 1$. In the last step, the first term is because there are $T - o(T)$ number of $\tilde{x}\tilde{x}^\mathsf{T}$ in A_t and $\|\tilde{x}^\mathsf{T}\mathbf{A}_t^{-1}\|_2 \le \frac{1}{T-o(T)}$, and $\|\sum_{t=1}^T x_t \Delta_t\|_2$ is bounded by total manipulation C_T. Similarly, in the second term we have $\|\tilde{x}^\mathsf{T}\mathbf{A}_t^{-1/2}\|_2 \le \frac{1}{\sqrt{T-o(T)}}$, and $\|\mathbf{A}_t^{-1/2}\left(\sum_{t=1}^T x_t\eta_t - \lambda\boldsymbol{\theta}^*\right)\|_2 \le \|\mathbf{A}_t^{-1/2}\left(\sum_{t=1}^T x_t\eta_t\right)\|_2 + \|\mathbf{A}_t^{-1/2}\lambda\boldsymbol{\theta}^*\|_2 = \|\sum_{t=1}^T x_t\eta_t\|_{\mathbf{A}_t^{-1}} + \|\lambda\boldsymbol{\theta}^*\|_{\mathbf{A}_t^{-1}} \le \alpha_t$ is the self-normalized bound for vector-valued martingales following Theorem 1 in (Abbasi-yadkori et al., 2011). □

Now we are ready to prove that the index ϵ^* in CQP (2) being positive is the necessary condition of an attackable environment.

Proof of Necessity of Theorem 1. We prove if $\epsilon^* \le 0$, the bandit environment is not attackable. To prove this, we show that there exists some no-regret bandit algorithm (LinUCB in particular) such that no attacking strategy can succeed. In particular, we will show that LinUCB (with a specific choice of its L2-regularization parameter λ) is robust under any attacking strategy with $o(T)$ cost when $\epsilon^* \le 0$. We prove it by contradiction: assume LinUCB is attackable with $o(T)$ cost when $\epsilon^* \le 0$. According to Definition 1, the target arm $\tilde{x}$ will be pulled $T - o(T)$ times for infinitely many different time horizons T, and the following inequalities hold when arm $\tilde{x}$ is pulled by LinUCB:

$$\tilde{x}^\mathsf{T}\hat{\boldsymbol{\theta}}_{T,\|} + \mathrm{CB}_T(\tilde{x}) > x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{T,\|} + x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{T,\perp} + \mathrm{CB}_T(x_a), \forall x_a \ne \tilde{x} \tag{19}$$

where $\hat{\boldsymbol{\theta}}_t$ is LinUCB's estimated parameter at round t based on the attacked rewards. We decompose $\hat{\boldsymbol{\theta}}_T = \hat{\boldsymbol{\theta}}_{T,\|} + \hat{\boldsymbol{\theta}}_{T,\perp}$, where $\tilde{x} \perp \hat{\boldsymbol{\theta}}_{t,\perp}$ and $\tilde{x} \parallel \hat{\boldsymbol{\theta}}_{T,\|}$. We will now show that the above inequalities lead to

$$\tilde{x}^\mathsf{T}\boldsymbol{\theta}^*_{\|} > x_a^\mathsf{T}\boldsymbol{\theta}^*_{\|} + x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{T,\perp}, \forall x_a \ne \tilde{x}$$

when $T \to \infty$.

By Lemma 4 we have

$$
\begin{aligned}
x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{T,\|} &\ge x_a^\mathsf{T}\boldsymbol{\theta}^*_{\|} - \frac{C_T}{T - o(T)} - \frac{\alpha_T}{\sqrt{T - o(T)}} \\
\tilde{x}^\mathsf{T}\hat{\boldsymbol{\theta}}_{T,\|} &\le \tilde{x}^\mathsf{T}\boldsymbol{\theta}^*_{\|} + \frac{C_T}{T - o(T)} + \frac{\alpha_T}{\sqrt{T - o(T)}}
\end{aligned}
$$

Substitute them back and we have that with probability at least $1 - 2\delta$,

$$\tilde{x}^\mathsf{T}\boldsymbol{\theta}^*_{\|} > x_a^\mathsf{T}\boldsymbol{\theta}^*_{\|} + x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{T,\perp} + \mathrm{CB}_T(x_a) - \mathrm{CB}_T(\tilde{x}) - \frac{2C_T}{T - o(T)} - \frac{2\alpha_T}{\sqrt{T - o(T)}}, \forall x_a \ne \tilde{x} \tag{20}$$

Let us first consider the case of $x_a \nparallel \tilde{x}$. Substitute Claim 1 and Claim 2 back and we have with probability at least $1 - 4\delta$

$$\begin{aligned}\tilde{x}^\mathsf{T}\boldsymbol{\theta}^*_\parallel > & x_a^\mathsf{T}\boldsymbol{\theta}^*_\parallel + x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{T,\perp} + \Theta\left(\sqrt{\log(T/\delta)}\left(\frac{1}{\sqrt{o(T)}} - \frac{1}{\sqrt{T-o(T)}}\right)\right) \\ & - O\left(\sqrt{\frac{\log(T/\delta)}{T-o(T)}}\right) - \frac{2C_T}{T-o(T)} - \frac{2\alpha_T}{\sqrt{T-o(T)}}, \forall x_a \nparallel \tilde{x}\end{aligned}$$

Taking $T \to \infty$ and noticing that the last three terms diminish to 0 faster than the third term, there must exists a T_0 such that for any $T > T_0$,

$$\tilde{x}^\mathsf{T}\boldsymbol{\theta}^*_\parallel > x_a^\mathsf{T}\boldsymbol{\theta}^*_\parallel + x_a^\mathsf{T}\hat{\boldsymbol{\theta}}_{T,\perp}, \forall x_a \neq \tilde{x} \tag{21}$$

satisfies when $x_a \nparallel \tilde{x}$.

Now we consider the special case that some $x_a \parallel \tilde{x}$ and show that the above inequality is still strict. Let $x_a = c\tilde{x}$. If $|c| > 1$, we have $\text{CB}_T(x_a) - \text{CB}_T(\tilde{x}) = (c-1)\text{CB}_T(\tilde{x}) > 0$. If $|c| < 1$, since $\tilde{x}$ is pulled linear times for any large t with sublinear cost, then $\tilde{x}^\mathsf{T}\hat{\theta}_{t,\parallel} > 0$; otherwise the cost would be linear. We directly have $\tilde{x}^\mathsf{T}\hat{\theta}_{t,\parallel} = x_a^\mathsf{T}\hat{\theta}_{t,\parallel} + (1-c)\tilde{x}^\mathsf{T}\hat{\theta}_{t,\parallel} > x_a^\mathsf{T}\hat{\theta}_{t,\parallel}$. This leads to $\tilde{x}^\mathsf{T}\theta^*_\parallel > x_a^\mathsf{T}\theta^*_\parallel$ (inequality (21)) since $x_a \perp \hat{\boldsymbol{\theta}}_{T,\perp}$.

Combining the two cases, we know there must exist a $\hat{\boldsymbol{\theta}}_{T,\perp}$ that satisfies inequality (21) (the first constraint of CQP (2)), $\tilde{x} \perp \hat{\boldsymbol{\theta}}_{T,\perp}$ (the second constraint of CQP (2)), and makes the objective of CQP (2) larger than 0. To satisfy the last constraint, we consider LinUCB with the choice of L2-regularization parameter λ that guarantees $\|\hat{\boldsymbol{\theta}}_t\|_2 < 1$ given the data for large enough T and any $t < T$. Note that ridge regression is equivalent to minimizing square loss under some constraint $\|\hat{\boldsymbol{\theta}}_t\|_2 \leq c$, and there always exists a one-to-one correspondence between λ and c (one simple way to show the correspondence is using KKT conditions). Therefore, we are guaranteed to find a λ that gives us $c = 1 - \zeta$ where ζ is an arbitrarily small constant. Then we know that $\hat{\boldsymbol{\theta}}_{T,\perp}$ satisfies $\|\hat{\boldsymbol{\theta}}_T = \hat{\boldsymbol{\theta}}_{T,\parallel} + \hat{\boldsymbol{\theta}}_{T,\perp}\|_2 \leq c < 1$. We prove the last constraint $\|\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta}^*_\parallel + \hat{\boldsymbol{\theta}}_{T,\perp}\|_2 \leq 1$ by the fact that $\|\hat{\boldsymbol{\theta}}_{T,\parallel} + \hat{\boldsymbol{\theta}}_{T,\perp}\|_2 < 1$ and $\|\boldsymbol{\theta}^*_\parallel - \hat{\boldsymbol{\theta}}_{T,\parallel}\|_2$ is arbitrarily small for large enough T according to Lemma 4.

Now we proved that there exists a $\hat{\boldsymbol{\theta}}_{T,\perp}$ that satisfies all the constraints in CQP (2) with positive objective, which means the optimal objective ϵ^* must also be positive. This however contradicts our assumption $\epsilon^* \leq 0$, implying that such LinUCB is not attackable by any attack strategy. □

C. Details on Effective Attacks Without Knowledge of Model Parameters

We now prove the theorems of using Two-stage Null Space Attack (Algorithm 1) against LinUCB and Robust Phase Elimination.

C.1. Proof of Theorem 3

Proof of Theorem 3. We first prove that for a large enough T, Algorithm 1 will correctly assert the attackability with probability at least $1 - \delta$. We rely on the following lemma to show $\tilde{\boldsymbol{\theta}}_\parallel$ estimated in step 11 of Algorithm 1 is close to the true parameter $\boldsymbol{\theta}^*_\parallel$.

Lemma 5 (Estimation error of $\tilde{\boldsymbol{\theta}}_\parallel$)**.** *Algorithm 1 estimates $\boldsymbol{\theta}^*_\parallel$ by*

$$\tilde{\boldsymbol{\theta}}_\parallel = \frac{\sum_{i=1}^{n(\tilde{x})} r_i(\tilde{x})}{n(\tilde{x})\|\tilde{x}\|_2^2}\tilde{x}. \tag{22}$$

With probability at least $1 - \delta$, the estimation error is bounded by

$$\|\tilde{\boldsymbol{\theta}}_\parallel - \boldsymbol{\theta}^*_\parallel\|_2 \leq \sqrt{\frac{2R^2\log(1/\delta)}{n}} \tag{23}$$

where the rewards have R-sub-Gaussian noise.

Proof. $\boldsymbol{\theta}_{\parallel}^{*}$ is the projected vector of $\boldsymbol{\theta}^{*}$ onto $\tilde{x}$, which is

$$\boldsymbol{\theta}_{\parallel}^{*} = \frac{\tilde{x}^{\mathsf{T}}\boldsymbol{\theta}^{*}}{\|\tilde{x}\|_2^2}\tilde{x}$$

as defined in Eq (1). Though we need to estimate the vector $\tilde{\boldsymbol{\theta}}_{\parallel} \in \mathbb{R}^d$, we actually only need to estimate the scale of it by $\hat{l} = \frac{\sum_{i=1}^{n(\tilde{x})} r_i(\tilde{x})}{n(\tilde{x})\|\tilde{x}\|_2^2}$, since the direction is known to be $\tilde{x}$. Based on Hoeffding's inequality, the estimation error of averaged rewards on $\tilde{x}$ is bounded by

$$P\left(\left|\frac{\sum_{i=1}^{n(\tilde{x})} r_i(\tilde{x})}{n(\tilde{x})} - r^*(\tilde{x})\right| \geq \sqrt{\frac{2R^2\log(1/\delta)}{n(\tilde{x})}}\right) \leq \delta \tag{24}$$

where $r^*(\tilde{x}) = \tilde{x}^{\mathsf{T}}\boldsymbol{\theta}^*$. Considering $\|\tilde{x}\|_2^2 = 1$ and we finish the proof. □

In the first stage, the bandit algorithm will pull the target arm $\tilde{x}$ for $T_1 - O(\sqrt{T_1})$ times, since $\tilde{x}$ is the best arm according to $\boldsymbol{\theta}_0$. According to Lemma 5, with probability at least $1-\delta$ the estimation error is bounded as

$$\|\tilde{\boldsymbol{\theta}}_{\parallel} - \boldsymbol{\theta}_{\parallel}^*\|_2 \leq \sqrt{\frac{2R^2\log(1/\delta)}{T_1 - O(\sqrt{T_1})}}.$$

As a result, with a large enough T_1, the error of $\tilde{x}$'s reward estimation satisfies

$$|\tilde{x}^{\mathsf{T}}\tilde{\boldsymbol{\theta}}_{\parallel} - \tilde{x}^{\mathsf{T}}\boldsymbol{\theta}_{\parallel}^*| \leq \|\tilde{x}\|_2\|\tilde{\boldsymbol{\theta}}_{\parallel} - \boldsymbol{\theta}_{\parallel}^*\|_2 \leq \sqrt{\frac{2R^2\log(1/\delta)}{T_1 - O(\sqrt{T_1})}} \leq \epsilon^*.$$

Thus solving CQP (2) with $\tilde{\boldsymbol{\theta}}_{\parallel}$ replacing $\boldsymbol{\theta}_{\parallel}^*$ and we could correctly assert attackability with an estimated positive index $\tilde{\epsilon}^*$ when the environment is attackable with index ϵ^*.

Remark 3. *From the analysis above, we notice that the adversary requires sufficiently large T_1 to collect enough rewards on the target arm, in order to make the correct attackability assertion. When T_1 is not large enough, the attackability test may have false positive or false negative conclusions. We empirically test the error rate and report the results in Additional Experiments section.*

We now prove the success and total cost of the proposed attack. The analysis relies on the "robustnes" property of LinUCB stated in Lemma 1, which is restated and proved below.

Lemma 6 (Robustness of ridge regression)**.** *Consider LinUCB with ridge regression under poisoning attack. Let $S_t' = \sum_{\tau\in\{1\ldots t\}, x_{a_\tau}\neq\tilde{x}} |\Delta_\tau|$ be the total corruption on non-target arms until time t and assume every corruption on target arm is bounded by γ. Then for any $t = 1\ldots T$, with probability at least $1-\delta$ we have*

$$\|\tilde{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_t\|_{A_t} \leq \alpha_t + S_t'/\sqrt{\lambda} + \gamma\sqrt{t} \tag{25}$$

where $\alpha_t = \sqrt{d\log\left(\frac{1+t/\lambda}{\delta}\right)} + \sqrt{\lambda}$.

Proof. Based on the closed form solution of ridge regression, we have

$$\hat{\boldsymbol{\theta}}_t = \tilde{\boldsymbol{\theta}} - \lambda\mathbf{A}_t^{-1}\tilde{\boldsymbol{\theta}} + \mathbf{A}_t^{-1}\sum_{\tau=1}^{t} x_{a_\tau}[\eta_\tau + \Delta_\tau]$$

Therefore, using ideas from (Abbasi-yadkori et al., 2011), we can have

$$\begin{aligned}
\|\hat{\boldsymbol{\theta}}_t - \tilde{\boldsymbol{\theta}}\|_{\mathbf{A}_t} &\le \lambda^{1/2}\|\boldsymbol{\theta}^*\|_2 + \|\sum_{\tau=1}^{t} x_{a_\tau}\eta_\tau\|_{\mathbf{A}_t^{-1}} + \|\sum_{\tau=1}^{t} x_{a_\tau}\Delta_\tau\|_{\mathbf{A}_t^{-1}} \\
&\le \alpha_t + \|\sum_{\tau=1}^{t} x_{a_\tau}\Delta_\tau\|_{\mathbf{A}_t^{-1}} \\
&\le \alpha_t + \|\sum_{\tau\in\{1\ldots t\}, x_{a_\tau}\neq\tilde{x}} x_{a_\tau}\Delta_\tau\|_{\mathbf{A}_t^{-1}} + \|\sum_{\tau\in\{1\ldots t\}, x_{a_\tau}=\tilde{x}} x_{a_\tau}\Delta_\tau\|_{\mathbf{A}_t^{-1}} \\
&\le \alpha_t + \|\sum_{\tau\in\{1\ldots t\}, x_{a_\tau}\neq\tilde{x}} x_{a_\tau}\Delta_\tau\|_{\mathbf{A}_t^{-1}} + \|\gamma n(\tilde{x})\tilde{x}\|_{\mathbf{A}_t^{-1}} \\
&\le \alpha_t + \|\sum_{\tau\in\{1\ldots t\}, x_{a_\tau}\neq\tilde{x}} x_{a_\tau}\Delta_\tau\|_2/\sqrt{\lambda} + \|\gamma n(\tilde{x})\tilde{x}\|_{\mathbf{A}_t^{-1}} \\
&\le \alpha_t + S_t'/\sqrt{\lambda} + \|\gamma n(\tilde{x})\tilde{x}\|_{\mathbf{A}_t^{-1}} \\
&\le \alpha_t + S_t'/\sqrt{\lambda} + \frac{\gamma n(\tilde{x})}{\sqrt{n(\tilde{x})}} \\
&\le \alpha_t + S_t'/\sqrt{\lambda} + \gamma\sqrt{t}
\end{aligned}$$

with probability at least $1-\delta$, where $n(\tilde{x})$ is the times target arm has been pulled. The second step is based on the definition of α_t and introduces the high probability bound, the fourth step is because we have $|\Delta_\tau| < \gamma$ if $x_{a_\tau} = \tilde{x}$; the fifth step is because of $\mathbf{A}_t \succeq \lambda\mathbf{I}$; and the second last inequality follows Eq (8). Finally, notice that $n(\tilde{x}) \le t$ and we finish the proof. □

Let us first analyze the attack in the first stage. Denote $R_T(\boldsymbol{\theta})$ as the regret of LinUCB until round T, where $\boldsymbol{\theta}$ is the ground-truth parameter. We know from (Abbasi-yadkori et al., 2011) that if the rewards are all generated by $\boldsymbol{\theta}$ then with probability at least $1-\delta$ we have

$$R_T(\boldsymbol{\theta}) = \alpha_T\sqrt{dT\log\left(\frac{1+T/\lambda}{\delta}\right)} = O(d\sqrt{T}\log(T/\delta)) \tag{26}$$

where $\alpha_t = \sqrt{d\log\left(\frac{1+t/\lambda}{\delta}\right)} + \sqrt{\lambda}$. Then the attack in the first T_1 rounds based on $\boldsymbol{\theta}_0$ should make the bandit algorithm pull $\tilde{x}$ at least $T_1 - R_{T_1}(\boldsymbol{\theta}_0)/\epsilon_0^*$ times. According to Lemma 5, with probability at least $1-\delta$ parameter estimation error is bounded by

$$\|\tilde{\boldsymbol{\theta}}_{\parallel,T_1} - \boldsymbol{\theta}_{\parallel}^*\|_2 \le \sqrt{2\log(1/\delta)}/\sqrt{T_1 - R_{T_1}(\boldsymbol{\theta}_0)/\epsilon_0^*} \le 2\sqrt{2\log(1/\delta)}/\sqrt{T_1} \tag{27}$$

for large enough T_1. This means we have

$$\gamma = \|\tilde{x}^\mathsf{T}\big(\tilde{\boldsymbol{\theta}}_{\parallel,T_1} - \boldsymbol{\theta}_{\parallel}^*\big)\| \le 2\sqrt{2\log(1/\delta)}/\sqrt{T_1} \tag{28}$$

Now we prove the attack is successful with high probability. Consider the regret of the LinUCB against $\tilde{\boldsymbol{\theta}}$ as the ground-truth parameter. The estimation error in $\hat{\boldsymbol{\theta}}_t - \tilde{\boldsymbol{\theta}}$ has three sources: the sub-Gaussian noise, the rewards on non-target arms in the first stage generated by $\boldsymbol{\theta}_0$ (the rewards on the target arm are corrected to $\tilde{\boldsymbol{\theta}}$ in step 19-20 in Algorithm 1), and the unattacked rewards on target arm in the second stage generated by $\boldsymbol{\theta}^*$. According to Lemma 1, with probability at least $1-3\delta$, we have

$$\|\hat{\boldsymbol{\theta}}_t - \tilde{\boldsymbol{\theta}}\|_{\mathbf{A}_t} \le \alpha_t + R_{T_1}(\boldsymbol{\theta}_0)/\sqrt{\lambda} + 2\sqrt{2t\log(1/\delta)}/\sqrt{T_1}, t > T_1.$$

To show the number of rounds pulling non-target arms, we first look at the regret against $\tilde{\boldsymbol{\theta}}$, i.e., $R_T(\tilde{\boldsymbol{\theta}})$.

$$
\begin{aligned}
R_T(\tilde{\boldsymbol{\theta}}) &\leq \sum_{t=1}^{T} \left(\tilde{x}^{\mathsf{T}} \tilde{\boldsymbol{\theta}} - x_{a_t}^{\mathsf{T}} \tilde{\boldsymbol{\theta}} \right) \\
&\leq \sum_{t=1}^{T} \left(\tilde{x}^{\mathsf{T}} \hat{\boldsymbol{\theta}}_t + \mathrm{CB}_t(\tilde{x}) - x_{a_t}^{\mathsf{T}} \tilde{\boldsymbol{\theta}} \right) \\
&\leq \sum_{t=1}^{T} \left(x_{a_t}^{\mathsf{T}} \hat{\boldsymbol{\theta}}_t + \mathrm{CB}_t(x_{a_t}) - x_{a_t}^{\mathsf{T}} \tilde{\boldsymbol{\theta}} \right) \\
&\leq \sum_{t=1}^{T} 2\mathrm{CB}_t(x_{a_t}) \\
&\leq 2\sqrt{T \sum_{t=1}^{T} \mathrm{CB}_t^2(x_{a_t})} \\
&\leq 2\|\hat{\boldsymbol{\theta}}_T - \tilde{\boldsymbol{\theta}}\|_{\mathbf{A}_T} \sqrt{T \sum_{t=1}^{T} \|x\|_{\mathbf{A}_t^{-1}}^2} \\
&\leq 2\big(\alpha_T + R_{T_1}(\boldsymbol{\theta}_0)/\sqrt{\lambda} + 2\sqrt{2T\log(1/\delta)}/\sqrt{T_1}\big) \sqrt{dT \log\left(\frac{1 + T/\lambda}{\delta}\right)}
\end{aligned}
$$

holds with probability at least $1 - 3\delta$. And LinUCB will pull non-target arms at most $R_T(\tilde{\boldsymbol{\theta}})/\epsilon^*$ times, which can be bounded by

$$
\begin{aligned}
R_T(\tilde{\boldsymbol{\theta}})/\epsilon^* &\leq 2\left(\alpha_T + R_{T_1}(\boldsymbol{\theta}_0)/\sqrt{\lambda} + 2\sqrt{2T\log(1/\delta)}/\sqrt{T_1}\right) \sqrt{dT \log\left(\frac{1 + T/\lambda}{\delta}\right)}/\epsilon^* \\
&\leq 2\left(\sqrt{dT \log\left(\frac{1 + T/\lambda}{\delta}\right)} + \sqrt{\lambda} + R_{T_1}(\boldsymbol{\theta}_0)/\sqrt{\lambda} + 2\sqrt{2T\log(1/\delta)}/\sqrt{T_1}\right) \sqrt{dT \log\left(\frac{1 + T/\lambda}{\delta}\right)}/\epsilon^*
\end{aligned}
$$

and is in the order of

$$
O\left(d\left(\sqrt{\log(T/\delta)} + \sqrt{T_1}\log\left(T_1/\delta\right) + \sqrt{T\log(1/\delta)}/\sqrt{T_1}\right)\sqrt{T\log(T/\delta)}/\epsilon^*\right) \tag{29}
$$

The $\sqrt{T_1}\log\left(T_1/\delta\right)$ term is due to the "corrupted" rewards of non-target arms observed in the first stage. Setting $T_1 = T^{1/2}$ gives us the minimum number of rounds pulling non-target arms in $\tilde{O}(T^{3/4})$ according to Eq (29).

Now we prove the total cost $C(T)$. Note that in order to make the attack "stealthy", we inject sub-Gaussian noise $\tilde{\eta}_t$ on perturbed reward to make it stochastic. We separate the total cost by the cost on changing the mean reward and the cost of sub-Gaussian noise as

$$
C(T) = \sum_{t=\{1..T\}, \tilde{r}_t \neq r_t} |\Delta r_t| \leq \sum_{i=1}^{N} |x_{a_i}^{\mathsf{T}}(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*)| + \sum_{i=1}^{N} |\tilde{\eta}_i| \tag{30}
$$

where $i \in \{t = \{1..T\} : \tilde{r}_t \neq r_t\}$. Let $N = |\{i\}|$ be the total rounds of attack. Since we know the times attacking non-target arms is bounded by Eq 29 and attack target arm at most T_1 times, we have with probablity $1 - \delta$,

$$
N = T_1 + O\left(d\left(\sqrt{\log(T/\delta)} + \sqrt{T_1}\log\left(T_1/\delta\right) + \sqrt{T\log(1/\delta)}/\sqrt{T_1}\right)\sqrt{T\log(T/\delta)}/\epsilon^*\right) = \tilde{O}(T^{3/4}) \tag{31}
$$

when setting $T_1 = T^{1/2}$.

Notice that since $\tilde{\eta}_t$ is R-sub-Gaussian, its absolute value $|\tilde{\eta}_t|$ is also R-sub-Gaussian, and $\mathbb{E}[|\tilde{\eta}_t|] < L$ for some constant L following Proposition 2.5.2 in (Vershynin, 2018). According to the general Hoeffding's inequality, with probability at least $1-\delta$,

$$\sum_{i=1}^{N} |\tilde{\eta}_i| \leq NL + \sqrt{NL\log(2/\delta)} = O(N + \sqrt{N\log(1/\delta)}) \tag{32}$$

Thus the second term of Eq (30) has the order of $O(N) = \tilde{O}(T^{3/4})$. The first term of Eq (30) is bounded by $2N + 2T_1$ because each attack changes the mean reward at most 2 except the compensation step in line 20, and the reward compensation on target arm can be bounded by $2T_1$ because target arm is pulled at most T_1 times in the first stage. Overall, with probability at least $1-4\delta$

$$C(T) \leq \sum_{i=1}^{N} |x_{a_i}^{\mathsf{T}}(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*)| + \sum_{i=1}^{N} |\tilde{\eta}_i| \leq 2N + 2T_1 + NL + \sqrt{NL\log(2/\delta)} = \tilde{O}(N) = \tilde{O}(T^{3/4}) \tag{33}$$

when setting $T_1 = T^{1/2}$.

□

C.2. Proof Sketch of Corollary 4

The proof is similar to the proof of Theorem 3, and thus we only explain the difference here.

Instead of using Lemma 1 to analyze the impact of perturbed rewards generated by $\boldsymbol{\theta}_0$ (against $\tilde{\boldsymbol{\theta}}$) collected in the first stage, we know RobustPhE has an additional regret term $O(S^2)$ for corruption S (assuming S is unknown to the bandit algorithm). Since the bandit algorithm observes T_1 rewards in the first stage, $S \leq 2T_1$ and the additional regret due to rewards from first stage is $\tilde{O}(T_1^2)$. For the unattacked rewards on target arm in the second stage generated by $\boldsymbol{\theta}^*$, we view them as rewards generated by $\tilde{\boldsymbol{\theta}}$ with misspecification error γ, i.e., $|\tilde{x}^{\mathsf{T}}(\boldsymbol{\theta}^* - \tilde{\boldsymbol{\theta}})| \leq \gamma$. Proposition 5.1 in (Lattimore et al., 2020) showed that the phase elimination algorithm with misspecification γ has additional regret in $O(\gamma\sqrt{d}T\log(T))$. With $\gamma \leq 2\sqrt{2\log(1/\delta)}/\sqrt{T_1}$ by Eq (28), the total regret is

$$R_T(\tilde{\boldsymbol{\theta}}) = O\Big(d\sqrt{T}\log(T/\delta) + \sqrt{d}T\log(T)\log(1/\delta)/\sqrt{T_1} + T_1^2\Big)$$

with probability at least $1-2\delta$. Therefore, we have with probability at least $1-2\delta$, the attack strategy will fool RobustPhE to pull non-target arms at most $O\Big((d\sqrt{T}\log(T/\delta) + \sqrt{d}T\log(T)\log(1/\delta)/\sqrt{T_1} + T_1^2)/\epsilon^*\Big)$ rounds.

Similar to Eq (31), we bound the total rounds of attacking RobustPhE by

$$N = T_1 + O\Big(\big(d\sqrt{T}\log(T/\delta) + \sqrt{d}T\log(T)\log(1/\delta)/\sqrt{T_1} + T_1^2\big)/\epsilon^*\Big) \tag{34}$$

From Eq (33), we know the total cost is in the same order as the rounds of attack. So with probability at least $1-3\delta$ the total cost is

$$O\Big(T_1 + (d\sqrt{T}\log(T/\delta) + \sqrt{d}T\log(T)\log(1/\delta)/\sqrt{T_1} + T_1^2)/\epsilon^*\Big).$$

Setting $T_1 = T^{2/5}$ gives us the minimum attack cost $\tilde{O}(T^{4/5})$, and the non-target arms are pulled at most $\tilde{O}(T^{4/5})$ rounds.

Remark 4. *Note that we bound the total corruption by T_1, which means the adversary does not need to compensate the rewards on the target arm as shown in line 20 in Algorithm 1. The robustness of RobustPhE allows us to carry over the rewards in the first stage while LinUCB does not.*

C.3. Attack under unknown T

Our two-stage null space attack algorithm requires that T is known for the convenience of analysis. Here we discuss a promising idea that leveraging the *doubling trick* to extend the attack to the case of unknown T, which will lead to a multi-stage attack that repeatedly adjusts the target parameter $\tilde{\theta}$ at each stage. More concretely, to attack LinUCB without knowing T, the adversary can start with a pre-specified horizon T_0 and run the two-stage attack. Once the actual horizon reaches T_0, the adversary will expand the horizon to T_0^2 (i.e., the "doubling" trick), and views previous T_0 rounds as the new first stage, re-calculates target parameter $\tilde{\theta}$ using rewards from the T_0 rounds and runs the new attack strategy until T_0^2. Using this exponential horizon sequence $\{T_i = T_0^{2^i}, i \in \mathbb{N}\}$, we have a multi-stage attack on LinUCB with unknown time horizon. Similarly, to attack RobustPhE, we will need to adjust the horizon sequence to be $\{T_i = T_0^{(5/2)^i}, i \in \mathbb{N}\}$, which will lead to a similar multi-stage attack on RobustPhE. We believe this direction is feasible based on our current analysis, and more thorough and complete proof should be the target of our next work.

D. Additional Experiments

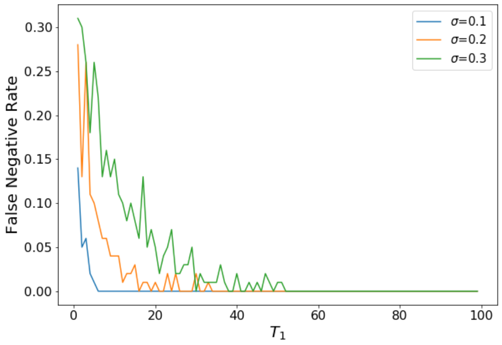

Figure 4. False negative rate of attackability test

In Figure 4, we study the false negative rate of the attackability test in Algorithm 1, i.e., how often the adversary mistakenly asserts that an attackable environment is not attackable. As we explained in Proof of Theorem 1, the wrong assertion is because of using estimated $\tilde{\boldsymbol{\theta}}_{\|}$ instead of the ground-truth bandit parameter. In this experiment, we consider a challenging attackable three-arm environment with $\mathcal{A} = \{x_1 = (0,1), x_2 = (0.11, 1.1), x_3 = (-2, 0)\}$, $\tilde{x} = x_1$ and $\boldsymbol{\theta}^* = (0, 0.5)$. By solving CQP (2), we have attackability index $\epsilon^* = 0.005$ and certificate $\tilde{\boldsymbol{\theta}}_{\perp} = (-0.5, 0)$[5]. We test two-stage null space attack against LinUCB with $T = 10,000$ and the adversary will test the attackability after the first $T_1 = T^{1/2} = 100$ rounds. We vary T_1 from 1 to 100 to see how many iterations is sufficient for attackability test. We report averaged results of 100 runs. We also vary the standard derivation σ of Gaussian noise from 0.1 to 0.3. In Figure 4, we can see that the false negative rate is almost zero when $T_1 > 50$, suggesting $T_1 = 100$ is sufficient. When $\sigma = 0.1$ the adversary only needs around 10 rounds to make a correct assertion. We also notice the false negative rate becomes higher under a larger noise scale. As suggested in Lemma 5, the error in $\tilde{\boldsymbol{\theta}}_{\|}$ estimation is larger if noise scale is larger or the number of target arm's rewards $n(\tilde{x})$ is smaller, which highly depends on T_1. Larger error means CQP (2) with $\tilde{\boldsymbol{\theta}}_{\|}$ is more likely to be unfeasible and gives false negative assertion. However, $T_1 = 100$ is still enough for the attackability test when $\epsilon^* = 0.005$.

[5] We introduce arm x_3 to guarantee the first dimension of $\tilde{\boldsymbol{\theta}}_{\perp}$ cannot be smaller than -0.5. Comparing $\tilde{x}$ and x_2 and we can see the optimal solution is $\epsilon^* = 0.005$.