Corruption Robust Thompson Sampling for Gaussian Bandits

Yinglun Xu[1,*] **Zhiwei Wang**[2,*], **Gagandeep Singh**[1]
[1]University of Illinois Urbana-Champaign, [2]Tsinghua University
[1]{yinglun6,ggnds}@illinois.edu, [2]wangzhiw21@mails.tsinghua.edu.cn

Abstract

Thompson sampling is one of the most popular learning algorithms for online sequential decision-making problems and has rich real-world applications. However, traditional Thompson sampling algorithms are limited by the assumption that the rewards received are uncorrupted, which may not hold in real-world applications where adversarial reward poisoning exists. To make Thompson sampling more reliable, our goal is to make it robust against adversarial reward poisoning. Particularly, we consider a strong attack threat model where an adversary applies corruption after observing the agent's actions. The main challenge is that one can no longer compute the actual posteriors for the true reward, as the agent can only observe the rewards after corruption. In this work, we solve this problem by computing pseudo-posteriors that are less likely to be manipulated by the attack. Particularly, we focus on two popular settings: stochastic bandits and contextual linear bandits with priors as Gaussian distributions. **We are the first** to propose robust algorithms based on Thompson sampling for the two bandit settings in both cases where the agent is aware or unaware of the attacker's budget. We theoretically show that our algorithms guarantee near-optimal regret under any attack strategy.

*Equal Contribution

Proceedings of the 29[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

1 INTRODUCTION

The multi-armed bandit (MAB) setting is a popular learning paradigm for sequential decision-making problems (Slivkins et al., 2019). Due to its simplicity, many industrial applications, such as recommendation systems, frame their problems as MAB (Brodén et al., 2018; Chu et al., 2011). As one of the most famous bandit learning strategies, Thompson sampling has been widely applied in these applications and achieves promising performance both empirically (Chapelle and Li, 2011; Scott, 2010) and theoretically (Agrawal and Goyal, 2013, 2017). Compared to another popular learning strategy known as optimality in the face of uncertainty (OFUL/UCB), Thompson sampling has several advantages, which we discuss in the appendix.

Despite the success, Thompson sampling faces the problem of low efficacy under adversarial reward poisoning attacks (Jun et al., 2018; Xu et al., 2021; Liu and Shroff, 2019). Existing algorithms assume that the reward signals corresponding to selecting an arm are drawn stochastically from a fixed distribution depending on the arm. However, this assumption does not always hold in practice. For example, a malicious user can provide false feedback for items suggested by a recommendation system. Even under small corruption, Thompson sampling algorithms suffer from significant regret under attacks (Xu et al., 2021). While robust versions of the learning algorithms following other fundamental exploration strategies such as optimality in the face of uncertainty (OFUL) and ϵ-greedy were developed (Lykouris et al., 2018; Neu and Olkhovskaya, 2020; Ding et al., 2022; He et al., 2022; Xu et al., 2023), there has been no prior investigation of robust Thompson sampling algorithms. The main challenge is that under the reward poisoning attacks, it becomes impossible to compute the actual posteriors based on the true reward, which is essentially required by the algorithm. Naively computing the posteriors based on the corrupted reward allows the attacker to manipulate the learning agent arbitrarily (Xu et al., 2021).

This work. We are the first to show the feasibility

of making Thompson sampling algorithms provably robust against reward poisoning. We focus on two popular bandit settings: stochastic bandits and contextual linear bandits. We consider a typical scenario where the prior distributions of each arm are Gaussian distributions. We develop robust Thompson sampling algorithms for the two bandit settings. We consider both cases where the corruption budget of the attack is known or unknown to the learning agent. We prove that our algorithm in the stochastic bandit case is near-optimal up to logarithmic factors. In the contextual linear bandit case, our algorithm is near-optimal up to logarithmic and a $\sqrt{d}$ factor, where d is the dimension of the context space. The guarantee is similar to the case of applying Thompson sampling to linear contextual bandits with no corruption (Agrawal and Goyal, 2013). We adopt two ideas to achieve robustness against reward poisoning attacks in the two MAB settings. The first idea is 'optimality in the face of corruption,' a general idea similar to the popular exploration strategy, 'optimality in the face of uncertainty.' Here, the agent optimistically estimates the best possible reward for an arm, considering the uncertainty and the influence of data corruption on its estimation. In the stochastic MAB setting, we show that the Thompson sampling algorithm can ensure sufficient exploration on arms and identify the optimal arm by relying on optimistic posteriors, considering potential attacks. The second idea is to adopt a weighted estimator (He et al., 2023) that is less susceptible to the attack. In the linear contextual MAB setting, we show that with such an estimator, the influence of the attack on the estimation of the posteriors is limited, and the Thompson sampling algorithm can identify the optimal arm with high probability at each round. We empirically test our algorithms under attacks and show that our algorithms are much more robust than other fundamental bandit algorithms, such as UCB, in practice. Our algorithm has similar performance compared to the state-of-the-art robust algorithm CW-OFUL (He et al., 2022) for the linear contextual bandit setting, and significantly better performance than the robust algorithms SE-WR and PE-WR (Wang et al., 2024) for the stochastic bandit setting. While achieving high performance in practice, our algorithm also inherits the advantages of using the Thompson sampling exploration strategy, as mentioned earlier.

2 PRELIMINARIES

2.1 Stochastic Bandit Setting

For the stochastic multi-armed bandit setting, an environment consists of N arms with fixed support in $[0, 1]$ reward distributions centered at $\{\mu_1, \ldots, \mu_N\}$, and an agent interacts with the environment for T rounds. Note that the reward distributions here do not have to be Gaussian distributions. At each round t, the agent selects an arm $i(t) \in [N]$ and receives reward r_t^o drawn from the reward distribution associated with the arm $i(t)$. The performance of the bandit algorithm is measured by its expected regret $R_T = \mathbb{E}\left[\sum_{t=1}^{T}\left(\mu_{i^*} - \mu_{i(t)}\right)\right]$, where i^* is the best arm at hindsight: $i^* = \arg\max_{i\in[N]} \mu_i$. Without loss of generality, we assume arm $i^* = 1$ is the optimal arm. We denote $\Delta_i = \mu_1 - \mu_i$ as the gap between arm i and the optimal arm. The time horizon T is predetermined, but the reward distribution of each arm is unknown to the agent. The agent's goal is to minimize its expected regret R_T.

2.2 Linear Contextual Bandit Setting

For the linear contextual bandit setting, an environment consists of N arms and a context space with d dimensions $\mathbb{R}^d$. An agent interacts with the environment for T rounds. At each round t, N contexts $\{x_i(t) \in \mathbb{R}^d\}, i = 1, \ldots, N$ are revealed by the environment for the N arms. We denote $\mathbf{x}(t) = (x_1(t), \ldots, x_N(t))$. The agent draws an arm $i(t)$ and receives a reward $r_i^o(t)$. The reward is drawn from a distribution depending on the arm $i(t)$ and the context $x_{i(t)}(t)$. The expectation of reward is a linear function depending on the context: $\mathbb{E}[r(t)|x_{i(t)}(t)] = x_{i(t)}(t)^T\mu$, where $\mu \in \mathbb{R}^d$ is the reward parameter. Without loss of generality, we assume the contexts and the reward parameters are bounded $\|x_i(t)\|_2 \leq 1, \|\mu\|_2 \leq 1$. The regret of the agent in this case is defined as $R_T = \sum_{t=1}^{T} x_{i^*(t)}(t)^T\mu - x_{i(t)}(t)^T\mu$ where $i^*(t) = \arg\max_i x_i(t)^T\mu$ is the optimal arm at time t. The time horizon T is predetermined, but the agent's reward parameter μ is unknown. The agent's goal is to minimize the regret.

To make the regret bounds scale-free, we adopt a standard assumption (Agrawal and Goyal, 2013) that $\epsilon_t = r^o(t) - x_{i(t)}(t)^T\mu$ is conditionally σ-sub-Gaussian for a constant $\sigma \geq 0$, i.e.,

$$\forall\lambda \in \mathbb{R}, \mathbb{E}\left[e^{\lambda\epsilon_t} \mid \{x_i(t)\}_{i=1}^{N}, \mathcal{H}_{t-1}\right] \leq \exp\left(\frac{\lambda^2\sigma^2}{2}\right),$$

where $\mathcal{H}_{t-1} = \left\{i(s), r(s), x_{i(s)}(s), s = 1, \ldots, t-1\right\}$.

2.3 Strong Reward Poisoning Attack against Bandits

This work considers the strong reward poisoning attack model (Wei et al., 2022), where an attacker sits between the environment and the agent. At each round t, the attacker observes the arm pulled by the agent $i(t)$ and the reward $r^o(t)$. In the contextual bandit setting, it also

observes the context $x_{i(t)}(t)$. Then the attacker can inject a perturbation $c(t)$ to the reward, and the agent will receive the corrupted reward $r(t) = r^o(t) + c(t)$ in the end. The attacker has full knowledge of the environment and the learner, including the algorithm it uses and the actions it takes each time. We denote C as the budget of the attack $C : \sum_{t=1}^{T} |c(t)| \leq C$, representing the maximum amount of total perturbation it can apply during the training process. We also refer to it as 'corruption level' since it indicates the level of corruption the learning agent faces.

The weak attack model has been considered in previous works (Lykouris et al., 2018; Liu and Shroff, 2019). Unlike the strong attack model, the weak attack decides on the perturbation of the reward before observing the actions taken by the agent. This work only considers the strong attack model for the following reasons: 1. The strong attack is more realistic. For example, in a recommendation system, the attacker, which is a malicious user, observes the recommendation first before deciding on the malicious feedback 2. Wei et al. (2022) shows that a robust algorithm against strong attack can be used to construct robust algorithms against weak attacks. In section 3 and 4, we discuss the case where the corruption level or an upper bound on it is known or unknown to the learning agent.

2.4 Thompson Sampling Algorithms

Thompson sampling is a heuristic exploration strategy that belongs to the family of randomized probability matching algorithms (Thompson, 1933). At each time step, a general Thompson sampling algorithm takes an arm based on a randomly drawn belief about the rewards of the arms. More specifically, the algorithm maintains a posterior distribution related to the expected reward of each arm. At each time, the algorithm samples a parameter from each arm's posterior to formulate a belief on the reward of an arm, and then it takes the arm with the maximal reward belief. After observing the reward of the taken arm, the algorithm updates the posteriors. The distribution of each arm's prior will influence the exact format of the algorithm.

In this work, we always assume that the priors for the rewards of the arms and the reward parameter are Gaussian distributions. This is a typical choice representative of Thompson sampling algorithms (Agrawal and Goyal, 2013, 2017). Although our algorithms assume Gaussian priors, in principle, it is not hard to extend them to other kinds of priors following the same idea, and many of our theoretical results are not dependent on the format of priors. In the appendix, we show the Thompson sampling algorithms in Alg 3 and 4 for the stochastic and linear contextual MAB settings, respectively, with Gaussian distributions as priors.

3 ROBUST THOMPSON SAMPLING FOR STOCHASTIC MULTI-ARMED BANDITS

In this section, we present our robust Thompson sampling algorithm for the stochastic MAB setting and the theoretical guarantee of its learning efficiency. We discuss both cases, whether the corruption level C is known or unknown to the learning agent. To understand why the original Thompson sampling algorithm (Alg 3) is vulnerable under the poisoning attack, we note that the actual posteriors of arms given the uncorrupted reward drawn from the environments can no longer be acquired. When computing the posterior as if the data are uncorrupted, the resulting posteriors can be biased. Therefore, the attacker can make the learner underestimate the posteriors of the optimal arms or overestimate those of the sub-optimal arms. As a result, the learner believes that the optimal arm has a low reward and rarely selects it.

To make the algorithm robust against attack, we utilize the idea of optimism. Instead of computing the posteriors as if the data are uncorrupted, the algorithm computes the optimistic posteriors with respect to corruption for each of the arms. For each arm, the algorithm finds the posterior that maximizes the expected reward for any possible true rewards before corruption. In the stochastic MAB settings with Gaussian priors, the mean of the posterior is $\frac{\sum_{s=1,i(s)=i}^{t-1} r(s)}{k_i(t)+1}$ where $k_i(t)$ is the number of times arm i be pulled before time t. Suppose the algorithm expects the corruption level to be no greater than $\bar{C}$, then the possible maximal true reward sum of an arm is $\sum_{s=1,i(s)=i}^{t-1} r(s)+\bar{C}$, so the mean of the optimistic posterior should be $\frac{\sum_{s=1,i(s)=i}^{t-1} r(s)+\bar{C}}{k_i(t)+1}$.

The robustness against the bias of posteriors induced by the poisoning attack is achieved by optimism. By using the optimistic posteriors for each arm, the algorithm never underestimates the posteriors of any arm. Even if a sub-optimal arm becomes the empirically optimal arm, after being selected a few times, its optimal posterior will be close to its actual posterior, which is inferior to the optimistic posterior of the optimal arm. As a result, the optimal arm will eventually be selected. Together with the Thompson sampling strategy to deal with the stochastic rewards from the environment, the algorithm can be robust and efficient in the stochastic MAB setting under poisoning attacks.

The empirical post-attack mean $\hat{\mu}_i(t)$ for arm i at time t is defined by $\hat{\mu}_i(t) := \frac{\sum_{s=1,i(s)=i}^{t-1} r(s)}{k_i(t)+1}$(note that $\hat{\mu}_i(t) = 0$ when $k_i(t) = 0$) and the empirical pre-attack mean $\hat{\mu}_i^o(t)$ for arm i at time t is defined by

$\hat{\mu}_i^o(t) := \frac{\sum_{s=1,i(s)=i}^{t-1} r^o(s)}{k_i(t)+1}$(note that $\hat{\mu}_i^o(t) = 0$ when $k_i(t) = 0$). Let $\theta_i(t)$ denote a sample generated independently for each arm i from the posterior distribution at time t. This is generated from posterior distribution $\mathcal{N}\left(\hat{\mu}_i(t) + \frac{\overline{C}}{k_i+1}, \frac{1}{k_i(t)+1}\right)$, where $\overline{C}$ is a hyper-parameter of the algorithm for tuning robustness against different level of corruption. In Alg 1, we formally show our robust Thompson sampling algorithm for the stochastic MAB setting.

Algorithm 1 Robust Thompson Sampling for Stochastic Bandits

1: **Params**: robustness level $\overline{C}$
2: For all $i \in [N]$, set $k_i = 0, \hat{\mu}_i = 0$
3: **for** $t = 1, 2, \ldots,$ **do**
4: For each arm $i = 1, \ldots, N$, sample $\theta_i(t)$ from the $\mathcal{N}\left(\hat{\mu}_i + \frac{\overline{C}}{k_i(t)+1}, \frac{1}{k_i(t)+1}\right)$ distribution.
5: Play arm $i(t) := \arg\max_i\{\theta_i(t)\}$ and observe reward r_t
6: Set $\hat{\mu}_{i(t)} := \frac{\hat{\mu}_{i(t)}k_i(t)+r_t}{k_{i(t)}+1}, k_i(t) := k_i(t) + 1$
7: **end for**

At each round t, Alg 1 samples a parameter $\theta_i(t)$ from a Gaussian distribution for arm i with the compensation term $\frac{\overline{C}}{k_i(t)+1}$ to make it the optimistic posterior. This enables the algorithm to explore the optimal arm even if the attack injects a negative bias. Notice that when $\overline{C} = 0$, it degenerates into original Thompson Sampling using Gaussian priors. The regret of the algorithm under the attack is guaranteed in Theorem 3.1 as below.

Theorem 3.1. *For the N-armed stochastic bandit problem under any reward poisoning attack with corruption level C, the expected regret of the Robust Thompson Sampling Alg 1 with $\overline{C} \geq C$ is bounded by:*

$$\mathbb{E}[\mathcal{R}(T)] \leq O(\sqrt{NT\ln N} + N\overline{C} + N).$$

The big-Oh notation hides absolute constants.

Proof Sketch: A detailed proof can be found in the appendix, and we also highlight the unique challenges we encounter and solve for developing and proving corruption-robust algorithms based on the Thompson sampling strategy. Here, we provide a sketch for the proof. First, we define two good events:

Definition 3.2 (Good Events)**.** For $i \neq 1$, define $E_i^{\mu}(t)$ is the event $\hat{\mu}_i(t) \leq \mu_i + \frac{\Delta_i}{3}$, and $E_i^{\theta}(t)$ is the event $\theta_i(t) \leq \mu_1 - \frac{\Delta_i}{3}$.

$E_i^{\mu}(t)$ represents the case where the empirical means of sub-optimal arms are not much larger than their true mean. $E_i^{\theta}(t)$ represents cases where the sampled rewards from sub-optimal arms' posteriors are not much larger than their true mean.

Next, we can prove that when both good events are true, the probability that the agent selects the optimal arm is high, so the regret in this case is limited. Then, we can show that the probability of either or both good events being false is low, so even if the worst scenario happens when the good events are false, the regret is still limited due to the low probability of this case. The key reason why both good events are true with a high probability is because of the bonus term $\frac{\overline{C}}{k_i(t)s+1}$ we add for the posterior distributions of each arm compensates for the bias induced by the attack in the worst case. As a result, the agent is very unlikely to underestimate the performance of each arm under any poisoning attack within the budget limit. Therefore, the explorations of each arm are likely to be sufficient. Finally, by combining all the cases, we can show that the total regret is limited.

Corruption level C known to the learner: In this case, we set $\overline{C} = C$ in Alg 1. Then, the dependency of regret on C is linear according to Theorem 3.1, which is near-optimal (Gupta et al., 2019).

Corruption level C unknown to the learner: In this case, we set $\overline{C} = \sqrt{T\ln N/N}$ in Alg 1. Theorem 3.1 shows that if $C \leq \sqrt{T\ln N/N}$, the regret can be upper bounded by $O(\sqrt{NT\ln N})$ when $T \geq N$, and if $C > \sqrt{T\ln N/N}$ the regret can be trivially bounded by $O(T)$. This upper bound is near-optimal when $C \leq \sqrt{T\ln N/N}$ due to the lower bound in Theorem 1.4 from Agrawal and Goyal (2017). The multi-armed bandit case of Theorem 4.12 in He et al. (2023) shows it's also near-optimal when $C > \sqrt{T\ln N/N}$.

Note that in the case of an unknown corruption level, our regret is not adaptive to the corruption level C. It is possible to design robust algorithms whose regret is adaptive to the corruption level (Lykouris et al., 2018; Wang et al., 2024), which is also the typical case in the weak attack scenario. However, in the appendix, we theoretically show that our regret is the best one can hope for under the strong attacks. The regret bound our algorithm achieves is tighter than that of an algorithm with regret adaptive to C.

4 ROBUST THOMPSON SAMPLING FOR CONTEXTUAL LINEAR BANDITS

This section presents our robust Thompson sampling algorithm for the contextual linear MAB setting and its theoretical guarantee. Similar to the stochastic bandit setting, the posterior on the reward parameter based on the corrupted reward can be biased compared to the actual posterior, resulting in poor decisions on action selection. Even worse, the bias induced by the reward

corruption can be large when computing the posterior as in Alg 4. Consequently, Zhao et al. (2021) follows the UCB exploration strategy using such an estimator, and the resulting algorithm is still not robust enough under the poisoning attacks. Therefore, we are not using the original estimator for our robust algorithm.

Based on He et al. (2023), we use a weighted ridge regression estimator as described in Alg 2 line 6 to compute the Gaussian posterior. Such an estimator can effectively reduce the bias induced by reward poisoning. The key is that it assigns less weight w_t to the data with a 'large' context $w_t = \min\left\{1, \gamma/\left\|x_{i(t)}(t)\right\|_{\boldsymbol{B(t)}^{-1}}\right\}$ so that its estimation is less sensitive to data corruption in these cases. We formally show the algorithm in Alg 2, where $v_t = \sigma\sqrt{9d\ln\left(\frac{t+1}{\delta}\right)}$ and $\gamma > 0$ is a hyper-parameter representing the degree of robustness against different corruption.

Algorithm 2 Robust Thompson Sampling for Contextual Linear Bandits

1: **Params**: robustness level γ
2: Set $B = I_d, \hat{\mu} = 0_d, f = 0_d$.
3: **for** $t = 1, 2, \ldots,$ **do**
4: Sample $\tilde{\mu}(t)$ from distribution $\mathcal{N}\left(\hat{\mu}, v_t^2 B(t)^{-1}\right)$.
5: Play arm $i(t) := \arg\max_i x_i(t)^T \tilde{\mu}(t)$, and observe reward r_t.
6: Set $w_t = \min\left\{1, \gamma/\left\|x_{i(t)}(t)\right\|_{\boldsymbol{B(t)}^{-1}}\right\}$
7: Update $B(t+1) = B(t) + w_t x_{i(t)}(t) x_{i(t)}(t)^T, f = f + w_t x_{i(t)}(t) r_t, \hat{\mu} = B(t)^{-1} f$.
8: **end for**

In general, Alg 2 is very similar to the original TS algorithm in Alg 4 except for using the ridge regression estimator. This change ensures that the posterior calculated in line 3 is not far from the actual posterior under poisoning attacks. In Theorem 4.1, we provide a high probability bound on the regret for Alg 2.

Theorem 4.1. *For the stochastic contextual bandit problem with linear payoff functions, with probability $1-\delta$, the total regret in time T for Robust Thompson Sampling for Contextual Linear Bandits (Algorithm 2) is bounded by*

$$
\begin{aligned}
&O\Bigg(de^{(1+\frac{C\gamma}{\sqrt{d}})^2}\sqrt{dT\ln T\ln\left(\frac{T}{\delta}\right)} \\
&+ C\gamma e^{(1+\frac{C\gamma}{\sqrt{d}})^2}\sqrt{dT\ln T} \\
&+ \frac{d^2 e^{(1+\frac{C\gamma}{\sqrt{d}})^2}}{\gamma}\ln T\sqrt{\ln\left(\frac{T}{\delta}\right)} \\
&+ Cde^{(1+\frac{C\gamma}{\sqrt{d}})^2}\ln T\Bigg)
\end{aligned}
$$

Proof Sketch: Here we provide a proof sketch for Theorem 4.1. The full proof can be found in the appendix. Similar to the proof for Theorem 3.1, first, we define two good events:

Definition 4.2 (Good Events)**.** Define $E^{\mu}(t)$ as the event

$$
\begin{aligned}
&\forall i : \left|x_i(t)^T\hat{\mu}(t) - x_i(t)^T\mu\right| \\
&\leq \left(\sigma\sqrt{d\ln\left(\frac{t^3}{\delta}\right)} + 1 + C\gamma\right)\cdot\|x_i(t)\|_{B(t)^{-1}}
\end{aligned}
$$

Define $E^{\theta}(t)$ as the event that

$$
\forall i : \left|\theta_i(t) - x_i(t)^T\hat{\mu}(t)\right| \leq \sqrt{4d\ln(t)}v_t\|x_i(t)\|_{B(t)^{-1}}.
$$

$E^{\mu}(t)$ represents the case where the mean of the posterior of each arm is close to its actual reward parameter. $E^{\theta}(t)$ represents the case where the sampled reward for each arm is close to its expected value.

Next, we prove that both good events hold with a high probability. The key reason is that the posterior distribution computed by the weighted estimator is less sensitive to any change in the rewards. Therefore, the agent is more robust against data corruption. As a result, the evaluation for each arm is more accurate, and the exploration will be sufficient correspondingly. For each arm that is sub-optimal at a round, we define an arm as saturated if it has been selected more than a specific number of times. The value for this particular number is also limited due to the weighted estimator. Next, we prove that if an arm is saturated and both good events are true, then it is very unlikely for the algorithm to select the arm. In other words, for a sub-optimal arm at a round, if it has already been selected a limited number of times, it is very unlikely to be chosen further. Therefore, the cumulative times that a sub-optimal arm is selected at a time is limited, so the regret of the algorithm is bounded.

Corruption level C known to the learner: In this case, we set $\gamma = \sqrt{d}/C$. By Theorem 4.1, the regret is upper bounded by

$$
\begin{aligned}
\mathcal{R}(T) &= O\Bigg(d\sqrt{dT\ln T\ln\left(\frac{T}{\delta}\right)} \\
&\quad + Cd\sqrt{d}\ln T\sqrt{\ln\left(\frac{T}{\delta}\right)}\Bigg) \\
&= \widetilde{O}(d\sqrt{dT} + d\sqrt{d}C)
\end{aligned}
$$

The algorithm becomes the same as the Linear Thompson sampling algorithm in Alg 2 when there is no corruption $C = 0$. The guarantee is near-optimal up to

a $\sqrt{d}$ factor according to the results in He et al. (2022), which is the same as the case of the Linear Thompson sampling algorithm.

Corruption level C unknown to the learner: In this case, we set $\gamma = \sqrt{d}/\sqrt{T}$ in Alg 2. Based on Theorem 4.1, the algorithm's regret is upper bounded by $\widetilde{O}(d\sqrt{dT})$ when $C \leq \sqrt{T}$, and by $O(T)$ when $C > \sqrt{T}$. Compared to the optimal guarantees given in He et al. (2022), our algorithm's guarantee is near-optimal up to a $\sqrt{d}$ factor.

In the next section, we empirically test the robustness of our algorithm and compare it against other state-of-the-art robust algorithms. In the appendix, we directly show the comparison between the theoretical guarantees of our robust Thompson sampling (RTS) algorithms and other robust algorithms in different bandit settings in Table 1.

5 SIMULATION RESULTS

In this section, we show the simulation results of running our algorithm on general bandit environments under attacks commonly used in other literature.

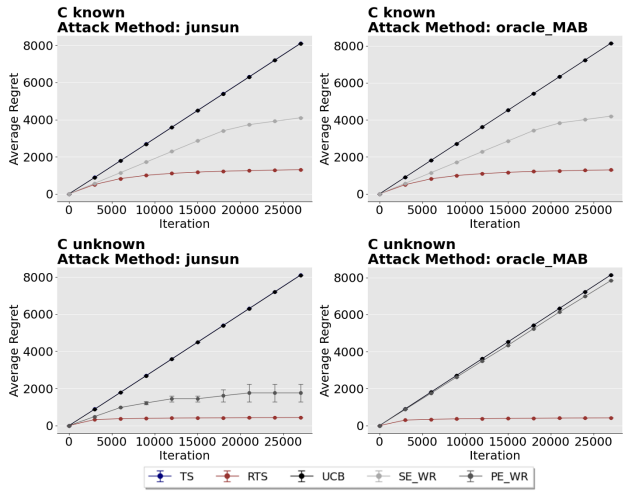

Figure 1: The cumulative regret of learning algorithms during training under different attacks in the stochastic bandit setting. The SE_WR algorithm is only applied to the cases when C is known to the learner, and the PE_WR algorithm is applied only when C is unknown.

5.1 Stochastic MAB Setting

Experiment setup: We consider a stochastic MAB environment consisting of $N = 5$ arms and a total number of timesteps of $T = 30000$. For comparison, we choose the UCB and Thompson sampling algorithms as the baseline vulnerable algorithms. For the baseline robust algorithm, we choose 'SE_WR' and 'PE_WR' algorithms from Wang et al. (2024). Here, SE-WR is designed for the case where the corruption level C is known to the learner, and PE_WR is designed for the case where C is unknown. Particularly, the PE_WR algorithm has a regret that is adaptive to C. The comparison against this algorithm also empirically examines if an algorithm with a regret not adaptive to the unknown corruption level is more robust in the case of strong attacks.

We adopt two attack strategies: (1) Junsun's attack. (2) oracle MAB attack from Jun et al. (2018). The choice of corruption level varies in our experiments, which are specified in the experimental results. Each experiment is repeated 10 times, and we show the empirical mean and standard deviation in the experimental results.

Cumulative regret during training under attack: Here, we intuitively show the behavior of our robust Thompson sampling algorithms under the attack. The corruption level for the attack is set as $C = 300$. Such a corruption level is sufficient to reveal the vulnerability of the non-robust learning algorithms. We show the results of other corruption levels later.

In Figure 1, we show the cases of corruption level C known and unknown to the learning agent. The bars in the plots are the variance of the data. For any non-robust algorithm, the regret rapidly increases with time, indicating that they rarely take the optimal arm during the whole learning process. We notice that the regret of the two vulnerable algorithms under the attack are similar. The reason is that the attacker highlights a target arm. For the vulnerable algorithms, they will believe that the target arm is optimal and almost always take that arm during training. So, their behavior and regret are very similar under the attack. For our algorithm, in all scenarios, after the first few timesteps, the regret increases slowly with time. This suggests that after some explorations, our algorithm identifies the optimal arm and takes it for most of the time. This observation is the same for the SE_WR algorithm. However, the PE_WR algorithm has a high regret under the oracle_MAB attack, indicating that its performance is low under some attacks. We will discuss the reason in the next part where we provide a more comprehensive empirical evaluation on algorithms under various corruption level.

Robustness evaluation and comparison under different corruption levels : Here, we show the total regret of the algorithms under attacks at different corruption levels. For both attack strategies, Figure 2 shows that for the vulnerable algorithms, the regret becomes large quickly as the corruption level

increases, indicating that these algorithms cannot find the optimal arm with even a low corruption level. In the known corruption level case, the regret almost increases linearly with the corruption level for our robust algorithm, which agrees with our theoretical result. In the unknown corruption level case, for our algorithm, we observe a threshold in the corruption level such that if the corruption level exceeds the threshold, the regret of our algorithm increases rapidly with the corruption level, which is aligned with our theoretical analysis. We notice that when the corruption level is small in the unknown C case, the regret of our algorithm decreases slightly as the corruption level increases. This is not surprising as, in the case of a low corruption level, the regret bound is $\tilde{O}(d\sqrt{dT})$, which is independent of the corruption level C.

We also observe that the performance of our robust Thompson sampling is better than that of the robust baseline algorithms in all scenarios. Particularly, the PE_WR algorithm has a low performance under the oracle attack. We believe this is a result of having a regret adaptive to the corruption level, which leads to worse theoretical guarantees on regret as we discuss in the appendix.

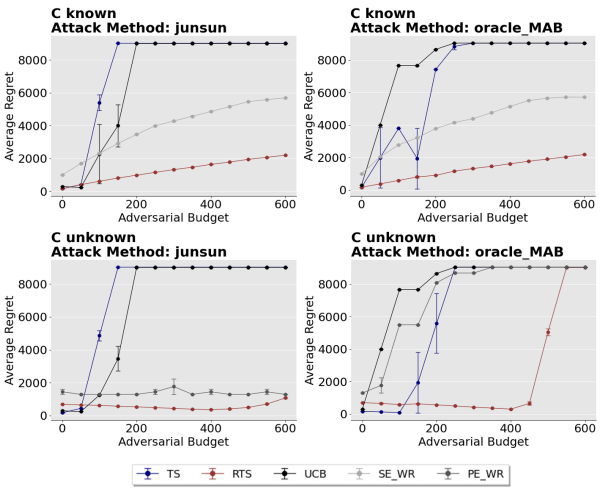

Figure 2: The total regret of different algorithms under attacks at different corruption levels in the stochastic bandit setting.

5.2 Contextual Linear Bandit Setting

Experiment setup: We consider a linear contextual bandit environment with $N = 5$ arms and $T = 10000$ timesteps. The dimension of the context space is $d = 5$. For the baseline algorithms, we choose two standard algorithms, LinUCB and LinTS, to represent the vulnerable algorithms. They are the extensions of the UCB and Thompson sampling algorithm to the linear contextual case. We choose the state-of-the-art CW-OFUL algorithm (He et al., 2022) as the robust baseline. We adopt two attack strategies: (1) Garcelon's attack proposed in Garcelon et al. (2020); (2) oracle MAB attack, which is also adopted in Garcelon et al. (2020).

Cumulative regret during training under attack: Here, we show the behavior of different learning algorithms under different attacks during the learning process. The corruption level for the attack is set as $C = 300$. In Figure 3, we show the cases of corruption level C known and unknown to the learning agent. Similar to the stochastic bandit case, the regret of the vulnerable algorithms rapidly increases with time, suggesting that they almost can never find the optimal arm. For the robust algorithms, after the first few timesteps, it learns to estimate the reward parameter accurately and can almost always find the optimal arm.

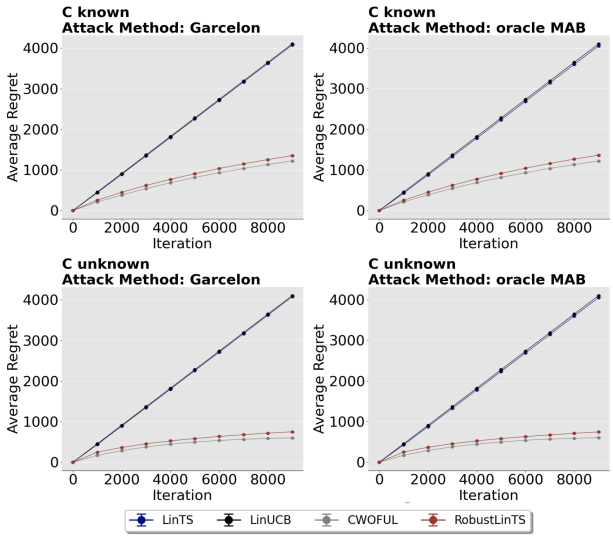

Figure 3: The cumulative regret of different learning algorithms during training under attacks in the linear contextual bandit setting.

Robustness evaluation and comparison under different corruption level: Here, we show the regret of different learning algorithms under attacks at different corruption levels. For both attack strategies we test with, the results in Fig 4 show that for the vulnerable algorithms, the regret increases quickly as the corruption level increases and converges to a large value in the end, indicating that these algorithms can no longer find the optimal arm for even a relatively low corruption level. For our robust algorithm, when C is known to the agent, the regret increases linearly with the corruption level; when C is unknown to the agent, there exists a threshold in the corruption level such

that at one point, the regret rapidly increases with the corruption level. This observation agrees with our theoretical result. Figure 3 and 4 also show that our algorithm performs similarly to the CW-OFUL algorithm. In our setup, our algorithm performs slightly worse in the known corruption level C case and significantly better in the unknown C case, especially when C is large. Our robust algorithms not only inherit the idea of Thompson sampling exploration but also achieve state-of-the-art performance in practice.

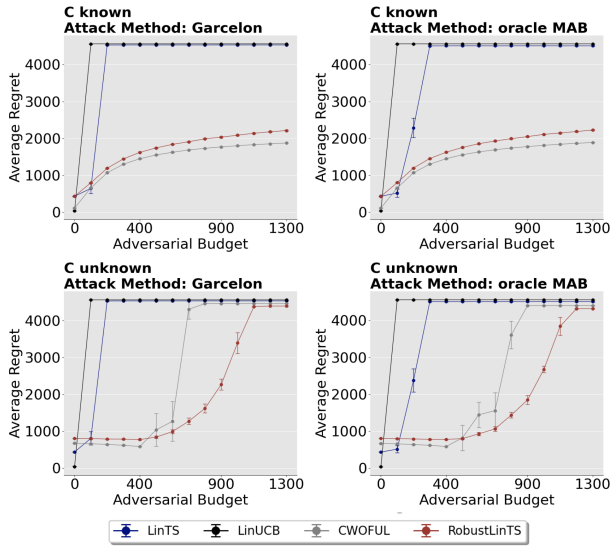

Figure 4: The regret of different algorithms under attacks at different corruption levels in the linear contextual bandit setting.

6 RELATED WORK

Thompson Sampling Algorithms: Algorithms based on Thompson sampling have been widely applied in sequential decision-making problems (Agrawal et al., 2017; Bouneffouf et al., 2014; Ouyang et al., 2017), including MAB settings with different constraints and reinforcement learning. Agrawal and Goyal (2013, 2017) develop insightful theoretical understandings of the learning efficiency of Thompson sampling. In the stochastic MAB and linear contextual MAB settings, algorithms based on Thompson sampling achieve near-optimal performance in that the upper bounds on regret match the lower bound of the setting (Agrawal and Goyal, 2012, 2013). However, these algorithms are designed for a setting without poisoning attacks, which may not be true in real-world scenarios.

Reward Poisoning Attack against Bandits: Reward poisoning attacks against bandits have been considered a practical threat against MAB algorithms (Lykouris et al., 2018). A majority of studies on poisoning attacks adopt a strong attack model where the attacker decides its attack strategy after the agent takes an arm at each timestep (Jun et al., 2018; Liu and Shroff, 2019; Garcelon et al., 2020). This attack scenario is argued to be more practical (Zhang et al., 2021). Jun et al. (2018) proposes attack strategies that can work for specific learning algorithms. It has been well understood that the most famous bandit algorithms are vulnerable to poisoning attacks. The other attack model is called weak attack, where the attacker decides its attack strategy before the agent takes an arm (Lykouris et al., 2018). Xu et al. (2021) shows that a family of algorithms, including the most famous ones, are vulnerable even under weak attacks.

Robust Bandit Algorithms: Finding bandit algorithms robust against poisoning attacks is a popular topic. Robust algorithms against weak or strong attack models have been developed in both stochastic bandit and linear contextual bandit settings (Lykouris et al., 2018; Neu and Olkhovskaya, 2020; Ding et al., 2022; He et al., 2022; Wang et al., 2024). Wei et al. (2022) shows that with an algorithm robust against the strong attack model, one can extend it to a robust algorithm against the weak attack model. Our work focuses on robustness against the strong attack model, as (1) the strong attack model is more practical, and (2) one can develop robust algorithms against weak attacks based on our algorithms.

Differentially Private Bandits: Differentially-private bandit setting is closely related to the poisoning attack (Mishra and Thakurta, 2015; Hu et al., 2021), and efficient learning algorithms have been achieved through Thompson sampling (Hu and Hegde, 2022). The differentially private setting can be considered a different robustness against poisoning attack setting. Here, the attacker can modify a certain number of data points, and a differentially private algorithm must ensure its behavior is similar under any possible attacks. However, there are two main differences between the two settings: 1. The constraint on the attack. In our reward poisoning setting, the attacker can poison as much data as they want, as long as the total amount of perturbation is limited. In contrast, in the differentially private setting, the attacker can only poison a limited number of data points. 2. The goal of the learning algorithm is different. In the reward poisoning setting, the agent only needs to guarantee that the total regret is limited under the corruption. In contrast, in the differentially private setting, the agent should behave almost identically when some data are corrupted. As a result, a differentially private algorithm is not necessarily a robust algorithm against reward poisoning attacks. Even under a limited corruption budget, the observed

data can completely differ from the original data at every data point during training. So, the guarantees on differential privacy cannot directly lead to guarantees on a tight bound of regret under the reward poisoning attack. In fact, Ou et al. (2024) proves that the Thompson sampling algorithm is already differentially private in the stochastic MAB setting, but it is not adversarially robust as aforementioned.

7 CONCLUSION AND LIMITATION

This work proposes two robust Thompson sampling algorithms for stochastic and linear contextual MAB settings, with a theoretical guarantee of near-optimal regret. However, our work is limited to the case where the posteriors of the arms are Gaussian distributions, and we do not cover more complicated settings like MDP. In the future, we aim to extend our techniques to other online learning settings.

References

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on learning theory*, pages 39–1. JMLR Workshop and Conference Proceedings, 2012.

Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *International conference on machine learning*, pages 127–135. PMLR, 2013.

Shipra Agrawal and Navin Goyal. Near-optimal regret bounds for thompson sampling. *Journal of the ACM (JACM)*, 64(5):1–24, 2017.

Shipra Agrawal, Vashist Avadhanula, Vineet Goyal, and Assaf Zeevi. Thompson sampling for the mnl-bandit. In *Conference on learning theory*, pages 76–78. PMLR, 2017.

Djallel Bouneffouf, Romain Laroche, Tanguy Urvoy, Raphael Féraud, and Robin Allesiardo. Contextual bandit for active learning: Active thompson sampling. In *Neural Information Processing: 21st International Conference, ICONIP 2014, Kuching, Malaysia, November 3-6, 2014. Proceedings, Part I 21*, pages 405–412. Springer, 2014.

Björn Brodén, Mikael Hammar, Bengt J Nilsson, and Dimitris Paraschakis. Ensemble recommendations via thompson sampling: an experimental study within e-commerce. In *23rd international conference on intelligent user interfaces*, pages 19–29, 2018.

Olivier Chapelle and Lihong Li. An empirical evaluation of thompson sampling. *Advances in neural information processing systems*, 24, 2011.

Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214. JMLR Workshop and Conference Proceedings, 2011.

Qin Ding, Cho-Jui Hsieh, and James Sharpnack. Robust stochastic linear contextual bandits under adversarial attacks. In *International Conference on Artificial Intelligence and Statistics*, pages 7111–7123. PMLR, 2022.

Evrard Garcelon, Baptiste Roziere, Laurent Meunier, Jean Tarbouriech, Olivier Teytaud, Alessandro Lazaric, and Matteo Pirotta. Adversarial attacks on linear contextual bandits. *Advances in Neural Information Processing Systems*, 33, 2020.

Anupam Gupta, Tomer Koren, and Kunal Talwar. Better algorithms for stochastic bandits with adversarial corruptions. In *Conference on Learning Theory*, pages 1562–1578. PMLR, 2019.

Jiafan He, Dongruo Zhou, Tong Zhang, and Quanquan Gu. Nearly optimal algorithms for linear contextual bandits with adversarial corruptions. *Advances in Neural Information Processing Systems*, 35:34614–34625, 2022.

Jiafan He, Heyang Zhao, Dongruo Zhou, and Quanquan Gu. Nearly minimax optimal reinforcement learning for linear markov decision processes. In *International Conference on Machine Learning*, pages 12790–12822. PMLR, 2023.

Bingshan Hu and Nidhi Hegde. Near-optimal thompson sampling-based algorithms for differentially private stochastic bandits. In *Uncertainty in Artificial Intelligence*, pages 844–852. PMLR, 2022.

Bingshan Hu, Zhiming Huang, and Nishant A Mehta. Optimal algorithms for private online learning in a stochastic environment. *arXiv e-prints*, pages arXiv–2102, 2021.

Kwang-Sung Jun, Lihong Li, Yuzhe Ma, and Jerry Zhu. Adversarial attacks on stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 3640–3649, 2018.

Tal Lancewicki, Shahar Segal, Tomer Koren, and Yishay Mansour. Stochastic multi-armed bandits with unrestricted delay distributions. In *International Conference on Machine Learning*, pages 5969–5978. PMLR, 2021.

Fang Liu and Ness Shroff. Data poisoning attacks on stochastic bandits. In *International Conference on Machine Learning*, pages 4042–4050. PMLR, 2019.

Thodoris Lykouris, Vahab Mirrokni, and Renato Paes Leme. Stochastic bandits robust to adversarial corruptions. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 114–122, 2018.

Nikita Mishra and Abhradeep Thakurta. (nearly) optimal differentially private stochastic multi-arm bandits. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pages 592–601, 2015.

Gergely Neu and Julia Olkhovskaya. Efficient and robust algorithms for adversarial linear contextual bandits. In *Conference on Learning Theory*, pages 3049–3068. PMLR, 2020.

Tingting Ou, Rachel Cummings, and Marco Avella. Thompson sampling itself is differentially private. In *International Conference on Artificial Intelligence and Statistics*, pages 1576–1584. PMLR, 2024.

Yi Ouyang, Mukul Gagrani, Ashutosh Nayyar, and Rahul Jain. Learning unknown markov decision processes: A thompson sampling approach. *Advances in neural information processing systems*, 30, 2017.

Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018.

Steven L Scott. A modern bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658, 2010.

Aleksandrs Slivkins et al. Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning*, 12(1-2):1–286, 2019.

William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933.

Xuchuang Wang, Jinhang Zuo, Xutong Liu, John Lui, and Mohammad Hajiesmaili. Stochastic bandits robust to adversarial attacks. *arXiv preprint arXiv:2408.08859*, 2024.

Chen-Yu Wei, Christoph Dann, and Julian Zimmert. A model selection approach for corruption robust reinforcement learning. In *International Conference on Algorithmic Learning Theory*, pages 1043–1096. PMLR, 2022.

Yinglun Xu, Bhuvesh Kumar, and Jacob D Abernethy. Observation-free attacks on stochastic bandits. *Advances in Neural Information Processing Systems*, 34:22550–22561, 2021.

Yinglun Xu, Bhuvesh Kumar, and Jacob Abernethy. On the robustness of epoch-greedy in multi-agent contextual bandit mechanisms. *arXiv preprint arXiv:2307.07675*, 2023.

Xuezhou Zhang, Yiding Chen, Xiaojin Zhu, and Wen Sun. Robust policy gradient against strong data corruption. In *International Conference on Machine Learning*, pages 12391–12401. PMLR, 2021.

Heyang Zhao, Dongruo Zhou, and Quanquan Gu. Linear contextual bandits with adversarial corruptions. *arXiv preprint arXiv:2110.12615*, 2021.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Not Applicable]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary Materials

Bandit Setting	Algorithm	Knowledge of C	Adversary	Regret
Stochastic	RTS	Known	Strong	$\tilde{O}(\sqrt{NT}+NC)$
Stochastic	RTS	Unknown	Strong	$\tilde{O}(\sqrt{NT}), C \leq \sqrt{T\ln N/N}$ $\tilde{O}(T), C > \sqrt{T\ln N/N}$
Stochastic	SE-WR	Known	strong	$\tilde{O}(\sqrt{NT}+NC)$
Stochastic	PE-WR	Unknown	strong	$\tilde{O}(\sqrt{NT}+NC^2)$
Stochastic	MAAER	Unknown	Weak	$\tilde{O}(\sqrt{NT}+N\sqrt{CT})$
Stochastic	BARBAR	Unknown	Weak	$\tilde{O}(\sqrt{NT}+NC)$
Contextual	RTS	Known	Strong	$\tilde{O}(d\sqrt{dT}+d\sqrt{d}C)$
Contextual	CW-OFUL	Known	Strong	$\tilde{O}(d\sqrt{T}+dC)$
Contextual	RTS	Unknown	Strong	$\widetilde{O}(d\sqrt{dT}), C \leq \sqrt{T}$ $\widetilde{O}(T), C > \sqrt{T}$
Contextual	CW-OFUL	Unknown	Strong	$\widetilde{O}(d\sqrt{T}), C \leq \sqrt{T}$ $\widetilde{O}(T), C > \sqrt{T}$

Table 1: Comparison between different robust bandit algorithms. RTS is our robust Thompson sampling algorithm; MAAER is the multi-layer active arm elimination race algorithm from Lykouris et al. (2018); BARBAR is the robust algorithm from Gupta et al. (2019). CW-OFUL is the robust algorithm from He et al. (2022). SE-WR and PE-WR are the robust algorithms from Wang et al. (2024)

A Proof for Section 3

A.1 Proof of Theorem 3.1

To begin with, we prove the case where $\overline{C} = C$ in Alg 1. Following the proof in Agrawal and Goyal (2017), we define two good events such that the agent is likely to pull the optimal arm when the events are true.

Definition A.1 (Good Events)**.** For $i \neq 1$, define $E_i^{\mu}(t)$ is the event $\hat{\mu}_i(t) \leq \mu_i + \frac{\Delta_i}{3}$, and $E_i^{\theta}(t)$ is the event $\theta_i(t) \leq \mu_1 - \frac{\Delta_i}{3}$. μ_i, Δ_i are defined in Section 2.1.

$E_i^{\mu}(t)$ holds mean that the empirical post-attack mean of any sub-optimal arm is not much greater than its true mean, and $E_i^{\theta}(t)$ holds means that the sampled value of any sub-optimal arm is not much greater than its true mean. Intuitively, under such situations, the regret should be low.

Next, we define a random variable $p_{i,t}$ determined by $\mathcal{H}_{t-1}$. $p_{i,t}$ represents that for a history $\mathcal{H}_{t-1}$, the probability of the sample from the optimal arm's distribution being much higher than the means of other arms.

Definition A.2. Define, $p_{i,t}$ as the probability $p_{i,t} := \Pr\left(\theta_1(t) > \mu_1 - \frac{\Delta_i}{3} \mid \mathcal{H}_{t-1}\right)$.

We decompose the regret into different cases based on whether the good events are true or not. The following lemma bounds the expected number of arm pulls for arm i when both the good events are true.

Lemma A.3.

$$\sum_{t=1}^{T} \Pr\left(i(t)=i, E_i^{\mu}(t), E_i^{\theta}(t)\right) \leq 72\left(e^{64}+4\right)\frac{\ln\left(T\Delta_i^2\right)}{\Delta_i^2} + \frac{4}{\Delta_i^2}$$

Sampling from the optimistic posterior, the reward belief of the optimal arm is likely to be large even under poisoning attacks. Therefore, the value of $p_{i,t}$ is more likely to be large, and the probability of pulling any

sub-optimal arm i in this case will be small. We notice that the constant term here is a big value. In Section 5 we empirically show that in practice the constant term is small.

Next, we consider the case when only $E_i^{\mu}(t)$ is true. The key insight is that for a sub-optimal arm i, when the empirical post-attack mean $\hat{\mu}_i$ is close to the true mean μ_i and the arm has already been pulled many times, the probability that sampled $\theta_i(t)$ is large is low. As a result, the total number of times when the sub-optimal arm is pulled in this case is limited. The formal result is shown in A.4

Lemma A.4.

$$\sum_{t=1}^{T} \Pr\left(i(t)=i, \overline{E_i^{\theta}(t)}, E_i^{\mu}(t)\right)$$
$$\leq \sum_{t=1}^{T} \Pr\left(i(t)=i, \overline{E_i^{\theta}(t)},\ E_i^{\mu}(t), k_i(t) \leq \max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2}, \frac{12C}{\Delta_i}\}\right) + \frac{1}{\Delta_i^2}$$

At last, we consider the case when neither good event is true. The key insight is that when a sub-optimal arm i has already been pulled many times, the probability that the empirical post-attack mean $\hat{\mu}_i$ is far from the true mean μ_i is low, and the total number of times it being pulled in this case is limited as shown in Lemma A.5.

Lemma A.5. *For $i \neq 1$,*

$$\sum_{t=1}^{T} \Pr\left(i(t)=i, \overline{E_i^{\mu}(t)}\right) \leq \sum_{t=1}^{T} \Pr\left(i(t)=i, \overline{E_i^{\mu}(t)}, k_i(t) \leq \max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2}, \frac{12C}{\Delta_i}\}\right) + 1 + \frac{8}{\Delta_i^2}$$

Next, we can prove Theorem 3.1 by combining the lemmas.

Proof of Theorem 3.1. We decompose the expected number of plays of a suboptimal arm $i \neq 1$ depending on whether the good events hold or not.

$$\mathbb{E}\left[k_i(T)\right] = \sum_{t=1}^{T} \Pr(i(t)=i) = \sum_{t=1}^{T} \Pr\left(i(t)=i, E_i^{\mu}(t), E_i^{\theta}(t)\right) + \sum_{t=1}^{T} \Pr\left(i(t)=i, E_i^{\mu}(t), \overline{E_i^{\theta}(t)}\right) +$$
$$\sum_{t=1}^{T} \Pr\left(i(t)=i, \overline{E_i^{\mu}(t)}\right).$$

For each sub-optimal arm i, combine the bounds from Lemmas A.3 to A.5, we obtain

$$\mathbb{E}\left[k_i(T)\right] \leq 72\left(e^{64}+4\right)\frac{\ln\left(T\Delta_i^2\right)}{\Delta_i^2} + \max\{\frac{12C}{\Delta_i}, \frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2}\} + \frac{13}{\Delta_i^2} + 1$$

$\Rightarrow$ The total regret can be upper bouded by:

$$\mathbb{E}[\mathcal{R}(T)] = \sum_{i=1}^{N} \Delta_i \mathbb{E}\left[k_i(T)\right] \leq \sum_{i=1}^{N} [72\left(e^{64}+5\right)\frac{\ln\left(T\Delta_i^2\right)}{\Delta_i} + 12C + \frac{13}{\Delta_i} + \Delta_i]$$

Then for every arm i with $\Delta_i \geq e\sqrt{\frac{N\ln N}{T}}$, the expected regret is bounded by $O\left(\sqrt{NT\ln N} + NC + N\right)$. And for arms with $\Delta_i \leq e\sqrt{\frac{N\ln N}{T}}$, the total regret is bounded by $O(\sqrt{NT\ln N} + NC)$. By combining these results we prove the Theorem 3.1.

□

We notice that the constant term in the regret guarantee is large, which is similar to the case in Agrawal and Goyal (2017). Our empirical results in Section 5 suggest that the actual regret is much smaller than the constant here. In the future, we will improve the tightness of the bound.

A.2 Proof of Lemma A.3

Lemma A.6 (Lemma 2.8 in Agrawal and Goyal (2017)). *For all $t, i \neq 1$ and all instantiations H_{t-1} of $\mathcal{H}_{t-1}$ we have*

$$\Pr\left(i(t)=i, E_i^{\mu}(t), E_i^{\theta}(t) \mid H_{t-1}\right) \leq \frac{(1-p_{i,t})}{p_{i,t}} \Pr\left(i(t)=1, E_i^{\mu}(t), E_i^{\theta}(t) \mid H_{t-1}\right)$$

From the proof of Lemma 2.8 in Agrawal and Goyal (2017), it's not hard to find that it is the event $E_i^{\theta}(t)$ that holds the above inequality. Therefore, although the distribution of θ_i has been changed to make the algorithm robust against adversarial attack, their proof can still be applied directly to the lemma A.6.

Lemma A.7. *Let s_j denote the time of the j^{th} play of the first arm. Then,*

$$\mathbb{E}\left[\frac{1}{p_{i,s_j+1}}-1\right] \leq \begin{cases} e^{64}+5 & \\ \frac{4}{T\Delta_i^2} & j > \frac{72\ln\left(T\Delta_i^2\right)}{\Delta_i^2} \end{cases}$$

Proof. Given $\mathcal{H}_{s_j}$, let Θ_j denote a random variable sampled from $\mathcal{N}\left(\hat{\mu}_1\left(s_j+1\right)+\frac{C}{j+1}, \frac{1}{j+1}\right)$, and Θ_j^o denote a random variable sampled from $\mathcal{N}\left(\hat{\mu}_1^o\left(s_j+1\right), \frac{1}{j+1}\right)$. We abbreviate $\hat{\mu}_1\left(s_j+1\right)$ to $\hat{\mu}_1$ and $\hat{\mu}_1^o\left(s_j+1\right)$ to $\hat{\mu}_1^o$ in the following. Let G_j and G_j^o be the geometric random variable denoting the number of consecutive independent trials until a sample of Θ_j or Θ_j^o becomes greater than $\mu_1-\frac{\Delta_i}{3}$, respectively. Notice that $\hat{\mu}_1\left(s_j+1\right)+\frac{C}{j+1} \geq \hat{\mu}_1^o\left(s_j+1\right)$, then we have

$$p_{i,s_j+1}=\Pr\left(\Theta_j > \mu_1-\frac{\Delta_i}{3} \mid \mathcal{H}_{s_j}\right) \geq \Pr\left(\Theta_j^o > \mu_1-\frac{\Delta_i}{3} \mid \mathcal{H}_{s_j}\right)$$

$\Rightarrow$

$$\mathbb{E}\left[\frac{1}{p_{i,s_j+1}}-1\right] \leq \mathbb{E}\left[\frac{1}{\Pr\left(\Theta_j^o > \mu_1-\frac{\Delta_i}{3} \mid \mathcal{H}_{s_j}\right)}-1\right]$$

From the result in Agrawal and Goyal (2017),

$$\mathbb{E}\left[\frac{1}{\Pr\left(\Theta_j^o > \mu_1-\frac{\Delta_i}{3} \mid \mathcal{H}_{s_j}\right)}-1\right] \leq \begin{cases} e^{64}+5 & \\ \frac{4}{T\Delta_i^2} & j > \frac{72\ln\left(T\Delta_i^2\right)}{\Delta_i^2} \end{cases}$$

Combining the above two inequalities, we derive the needed result. □

Proof. Let s_k denote the time at which arm i is pulled k times. Set $s_0=0$. Now applying Lemma A.6 and Lemma A.7, we have

$$\begin{aligned}
\sum_{t=1}^{T} \Pr\left(i(t)=i, E_i^{\mu}(t), E_i^{\theta}(t)\right) &= \sum_{t=1}^{T} \mathbb{E}\left[\Pr\left(i(t)=i, E_i^{\mu}(t), E_i^{\theta}(t) \mid \mathcal{H}_{t-1}\right)\right] \\
&\leq \sum_{t=1}^{T} \mathbb{E}\left[\frac{(1-p_{i,t})}{p_{i,t}} \Pr\left(i(t)=1, E_i^{\theta}(t), E_i^{\mu}(t) \mid \mathcal{H}_{t-1}\right)\right] \\
&= \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{E}\left[\frac{(1-p_{i,t})}{p_{i,t}} I\left(i(t)=1, E_i^{\theta}(t), E_i^{\mu}(t)\right) \mid \mathcal{H}_{t-1}\right]\right] \\
&= \sum_{t=1}^{T} \mathbb{E}\left[\frac{(1-p_{i,t})}{p_{i,t}} I\left(i(t)=1, E_i^{\theta}(t), E_i^{\mu}(t)\right)\right] \\
&= \sum_{k=0}^{T-1} \mathbb{E}\left[\frac{(1-p_{i,s_k+1})}{p_{i,s_k+1}} \sum_{t=s_k+1}^{s_{k+1}} I\left(i(t)=1, E_i^{\theta}(t), E_i^{\mu}(t)\right)\right] \\
&\leq \sum_{k=0}^{T-1} \mathbb{E}\left[\frac{(1-p_{i,s_k+1})}{p_{i,s_k+1}}\right] \\
&\leq 72\left(e^{64}+4\right)\frac{\ln\left(T\Delta_i^2\right)}{\Delta_i^2}+\frac{4}{\Delta_i^2}
\end{aligned}$$

Then we obtain the bound in Lemma A.3. □

A.3 Proof of Lemma A.4

Proof. We have

$$\sum_{t=1}^{T}\Pr\left(i(t)=i,\overline{E_i^{\theta}(t)},E_i^{\mu}(t)\right)=\sum_{t=1}^{T}\Pr\left(i(t)=i,k_i(t)\le\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\},\overline{E_i^{\theta}(t)},E_i^{\mu}(t)\right)+$$
$$\sum_{t=1}^{T}\Pr\left(i(t)=i,k_i(t)>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\},\overline{E_i^{\theta}(t)},E_i^{\mu}(t)\right)$$

Next, we prove that the probability that the event $E_i^{\theta}(t)$ is violated is small when $k_i(t)$ is large enough and $E_i^{\mu}(t)$ holds. Notice that

$$\begin{aligned}
&\sum_{t=1}^{T}\Pr\left(i(t)=i,k_i(t)>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\},\overline{E_i^{\theta}(t)},E_i^{\mu}(t)\right)\\
&\le\mathbb{E}\Big[\sum_{t=1}^{T}\Pr\left(i(t)=i,\overline{E_i^{\theta}(t)}\mid k_i(t)>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\},E_i^{\mu}(t),\mathcal{H}_{t-1}\right)\Big]\\
&\le\mathbb{E}\Big[\sum_{t=1}^{T}\Pr\left(\theta_i(t)>\mu_1-\frac{\Delta_i}{3}\mid k_i(t)>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\},E_i^{\mu}(t),\mathcal{H}_{t-1}\right)\Big]
\end{aligned}$$

Recall that $\theta_i(t)$ is sampled from $\mathcal{N}\left(\hat{\mu}_i(t)+\frac{C}{k_i(t)+1},\frac{1}{k_i(t)+1}\right)$. Since $\hat{\mu}_i(t)\le\mu_i+\frac{\Delta_i}{3}$, we have

$$\begin{aligned}
&\Pr\left(\theta_i(t)>\mu_1-\frac{\Delta_i}{3}\mid k_i(t)>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\},\hat{\mu}_i(t)\le\mu_i+\frac{\Delta_i}{3},\mathcal{H}_{t-1}\right)\\
&\le\Pr\left(\mathcal{N}\left(\mu_i+\frac{\Delta_i}{3}+\frac{C}{k_i(t)+1},\frac{1}{k_i(t)+1}\right)>\mu_1-\frac{\Delta_i}{3}\mid\mathcal{H}_{t-1},k_i(t)>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\}\right)
\end{aligned}$$

Since $k_i(t)>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\}$, from the property of Gaussian distribution we obtain that

$$\begin{aligned}
\Pr\left(\mathcal{N}\left(\mu_i+\frac{\Delta_i}{3}+\frac{C}{k_i(t)+1},\frac{1}{k_i(t)+1}\right)>\mu_1-\frac{\Delta_i}{3}\right)&\le\frac{1}{2}e^{-\frac{\left(k_i(t)+1\right)\left(\frac{\Delta_i}{3}-\frac{C}{k_i(t)+1}\right)^2}{2}}\\
&\le\frac{1}{2}e^{-\frac{\left(k_i(t)+1\right)\left(\frac{\Delta_i}{3}\right)^2}{2}}\\
&\le\frac{1}{T\Delta_i^2}
\end{aligned}$$

the second inequality holds because $k_i(t)\ge\frac{12C}{\Delta_i}$ and the last inequality holds because $k_i(t)\ge\frac{32\ln\left(T\Delta_i^2\right)}{\left(\Delta_i\right)^2}$. Therefore,

$$\Pr\left(\theta_i(t)>\mu_1-\frac{\Delta_i}{3}-\frac{\Delta_i}{12}\mid k_i(t)>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\},\hat{\mu}_i(t)\le\mu_i+\frac{\Delta_i}{3},\mathcal{H}_{t-1}\right)\le\frac{1}{T\Delta_i^2}$$

$\Rightarrow$

$$\sum_{t=1}^{T}\Pr\left(i(t)=i,k_i(t)>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2},\frac{12C}{\Delta_i}\},\overline{E_i^{\theta}(t)},E_i^{\mu}(t)\right)\le\frac{1}{\Delta_i^2}$$

Then we finish the proof. □

A.4 Proof of Lemma A.5

Proof. Let s_k denote the time at which arm i is pulled k times. Set $s_0 = 0$. We have

$$\sum_{t=1}^{T} \Pr\left(i(t)=i, \overline{E_i^{\mu}(t)}\right) \leq \sum_{t=1}^{T} \Pr\left(i(t)=i, \overline{E_i^{\mu}(t)}, k_i(t) \leq \max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2}, \frac{12C}{\Delta_i}\}\right) + \sum_{k>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2}, \frac{12C}{\Delta_i}\}}^{T-1} \Pr\left(\overline{E_i^{\mu}\left(s_{k+1}\right)}\right)$$

At time s_{k+1} for $k \geq 1$, we have $\hat{\mu}_i\left(s_{k+1}\right) = \frac{\sum_{t=1,i(t)=i}^{s_{k+1}} r_i(t)}{k+1} \leq \frac{\sum_{t=1,i(t)=i}^{s_{k+1}} r_i^o(t)}{k+1} + \frac{C}{k+1}$. By Chernoff-Hoeffding inequality, when $k > \max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2}, \frac{12C}{\Delta_i}\}$,

$$\begin{aligned}
&\Pr\left(\hat{\mu}_i\left(s_{k+1}\right) > \mu_i + \frac{\Delta_i}{3}\right) \\
&\leq \Pr\left(\frac{\sum_{t=1,i(t)=i}^{s_{k+1}} r_i^o(t)}{k+1} + \frac{C}{k+1} > \mu_i + \frac{\Delta_i}{3}\right) \leq e^{-2(k+1)\left(\frac{\Delta_i}{3} - \frac{C}{k+1}\right)^2} \leq e^{\frac{-(k+1)\Delta_i^2}{8}}
\end{aligned}$$

we then obtain that

$$\begin{aligned}
&\sum_{k>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2}, \frac{12C}{\Delta_i}\}}^{T-1} \Pr\left(\overline{E_i^{\mu}\left(s_{k+1}\right)}\right) = \sum_{k>\max\{\frac{32\ln\left(T\Delta_i^2\right)}{\Delta_i^2}, \frac{12C}{\Delta_i}\}}^{T-1} \Pr\left(\hat{\mu}_i\left(s_{k+1}\right) > \mu_i + \frac{\Delta_i}{3}\right) \\
&\leq 1 + \sum_{k=1}^{T-1} \exp\left(-\frac{(k+1)\Delta_i^2}{8}\right) \leq 1 + \frac{8}{\Delta_i^2}
\end{aligned}$$

Combining the above results we finish the proof. □

B Proof for Section 4

B.1 Proof of Theorem 4.1

Follow the proof of Agrawal and Goyal (2013), to prove Theorem 4.1, we begin with some basic definitions. The sample $\tilde{\mu}(t)$ from the posterior is a belief in the reward parameter. Therefore, we denote the actual sample for the belief in the reward of an arm as $\theta_i(t) := x_i(t)^T \tilde{\mu}(t)$. By definition of $\tilde{\mu}(t)$ in Algorithm 2, marginal distribution of each $\theta_i(t)$ is Gaussian with mean $x_i(t)^T \hat{\mu}(t)$ and standard deviation $v_t \|x_i(t)\|_{B(t)^{-1}}$, where $v_t = \sigma\sqrt{9d\ln\left(\frac{t+1}{\delta}\right)}$. Similar to before, we denote $\Delta_i(t) := x_{i^*(t)}(t)^T\mu - x_i(t)^T\mu$ as the gap between the mean reward of optimal arm and arm i at time t, and we define two good events as Definition 4.2

Next, we define a notion called saturated arm to indicate whether an arm has been taken for enough time such that the variance in its reward estimation is less than the gap in reward compared to the optimal arm at a timestep.

Definition B.1 (Saturated Arm)**.** Denote $g_t = \sqrt{4d\ln(t)}v_t + \sigma\sqrt{d\ln\left(\frac{t^3}{\delta}\right)} + 1 + C\gamma$. An arm i is called saturated at time t if $\Delta_i(t) > g_t\|x_i(t)\|_{B(t)^{-1}}$, and unsaturated otherwise. Let $C(t)$ be the set of saturated arms at time t.

First, in Lemma B.2 we show that good events hold with a high probability at each round. The reason why Lemma B.2 holds is because of the weighted ridge estimator we use for computing the posteriors. With such an estimator, the attack has less influence on the estimations, and therefore the difference between the estimation and the true value can be bounded.

Lemma B.2. *For all* $t, 0 < \delta < 1, \Pr\left(E^{\mu}(t)\right) \geq 1 - \frac{\delta}{t^2}$. *For all possible filtrations* $\mathcal{H}_{t-1}, \Pr\left(E^{\theta}(t) \mid \mathcal{H}_{t-1}\right) \geq 1 - \frac{1}{t^2}$.

Next, Lemma B.3 shows that when the good events are true, with a high probability, the sampled reward for the optimal arm at a timestep is likely to be larger than its actual expected reward.

Lemma B.3. *Denote* $p = \frac{1}{4e^{(1+\frac{C\gamma}{\sqrt{d}})^2}\sqrt{\pi}}$. *For any filtration* $\mathcal{H}_{t-1}$ *such that* $E^{\mu}(t)$ *is true,*

$$\Pr\left(\theta_{i^*(t)}(t) > x_{i^*(t)}(t)^T\mu \mid \mathcal{H}_{t-1}\right) \geq p$$

Lemma B.3 suggests that the algorithm is unlikely to underestimate the reward of the optimal arm.

Based on Lemma B.2 and Lemma B.3, we can further show that when the good events are true, since the optimal arm and the unsaturated arm will be pulled with at least a certain probability, the algorithm can perform effective exploration. Therefore, the regret will be upper bounded with a high probability. We establish a super-martingale process that will form the basis of our proof of the high-probability regret bound. This result shows that expected regret will be $O(\frac{g_t}{p} * \sum_t \|x_{i(t)}(t)\|_{B(t)^{-1}})$.

Definition B.4. Recall that regret (t) was defined as, regret $(t) = \Delta_{i(t)}(t) = x_{i^*(t)}(t)^T\mu - x_{i(t)}(t)^T\mu$. Define regret $'(t) = \text{regret}(t) \cdot I\left(E^{\mu}(t)\right)$.

Definition B.5. Let

$$X_t = \text{regret}'(t) - \min\{\frac{3g_t}{p}\|x_{i(t)}(t)\|_{B(t)^{-1}} + \frac{2g_t}{pt^2}, 1\}$$
$$Y_t = \sum_{w=1}^{t} X_w.$$

Lemma B.6. $(Y_t; t = 0, \ldots, T)$ *is a super-martingale process with respect to filtration* $\mathcal{H}_t$.

Proof of Theorem 4.1. Note that X_t is bounded, $|X_t| \leq 1 + \frac{3}{p}g_t + \frac{2}{pt^2}g_t \leq \frac{6}{p}g_t$. Thus, we can apply the Azuma-Hoeffding inequality, to obtain that with probability $1 - \frac{\delta}{2}$,

$$\sum_{t=1}^{T}\text{regret}'(t) \leq \sum_{t=1}^{T}\min\{\frac{3g_t}{p}s_{a(t)}(t), 1\} + \sum_{t=1}^{T}\frac{2g_t}{pt^2} + \sqrt{2\left(\sum_t \frac{36g_t^2}{p^2}\right)\ln\left(\frac{2}{\delta}\right)}$$

Note that p is a constant. Also, by definition, $g_t \leq g_T$. Therefore, from above equation, with probability $1 - \frac{\delta}{2}$,

$$\sum_{t=1}^{T}\text{regret}'(t) \leq \sum_{t=1}^{T}\min\{\frac{3g_t}{p}s_{i(t)}(t), 1\} + \frac{2g_T}{p}\sum_{t=1}^{T}\frac{1}{t^2} + \frac{6g_T}{p}\sqrt{2T\ln\left(\frac{2}{\delta}\right)}$$

Now, we have

$$\begin{aligned}
&\sum_{t=1}^{T}\min\{1, \frac{3g_t}{p}s_{t,i(t)}\} \\
&= \underbrace{\sum_{t:w_t=1}\min\left(1, \frac{3g_t}{p}\sqrt{x_{i(t)}(t)^\top B_{i(t)}^{-1}x_{i(t)}(t)}\right)}_{I_1} + \underbrace{\sum_{t:w_t<1}\min\left(1, \frac{3g_t}{p}\sqrt{x_{i(t)}(t)^\top B_{i(t)}^{-1}x_{i(t)}(t)}\right)}_{I_2},
\end{aligned}$$

For the term I_1, we consider all rounds $t \in [T]$ with $w_t = 1$ and we assume these rounds can be listed as $\{t_1, \cdots, t_m\}$ for simplicity. For each $i \leq m$, we can construct matrix $A_i = I + \sum_{j=1}^{i-1} x_{i(t_j)}(t_j)x_{i(t_j)}(t_j)^\top$. Due to the definition of B_t, we have

$$B_{t_i} \geq I + \sum_{j=1}^{i-1} w_{t_j}x_{i(t_j)}(t_j)x_{i(t_j)}(t_j)^\top = A_i$$

It further implies that for vector $x_{i(t)}(t)$, we have

$$x_{i(t)}(t)^\top B_{i(t)}^{-1}x_{i(t)}(t) \leq x_{i(t)}(t)^\top A_i^{-1}x_{i(t)}(t)$$

The term I_1 can be bounded by

$$\begin{aligned}
I_1 &= \sum_{t:w_t=1} \min\left(1, \frac{3g_t}{p}\sqrt{x_{i(t)}(t)^\top B_{i(t)}^{-1} x_{i(t)}(t)}\right) \\
&\le \sum_{i=1}^{m} \frac{3g_t}{p} \min\left(1, \sqrt{x_{i(t_i)}(t_i)^\top A_i^{-1} x_{i(t_i)}(t_i)}\right) \\
&\le \frac{3g_t}{p}\sqrt{\sum_{i=1}^{m} 1 \times \sum_{i=1}^{m} \min\left(1, x_{i(t_i)}(t_i)^\top A_i^{-1} x_{i(t_i)}(t_i)\right)} \\
&\le \frac{3g_t}{p}\sqrt{2dT\ln(1+T)},
\end{aligned}$$

For the second term I_2, according to the definition for weight $w_t < 1$ in Algorithm 1 , we have $w_t = \gamma/\sqrt{x_{i(t)}(t)^\top B_t^{-1} x_{i(t)}(t)}$, which implies that

$$\begin{aligned}
I_2 &= \sum_{t:w_t<1} \min\left(1, \frac{3g_t}{p}\sqrt{x_{i(t)}(t)^\top B_{i(t)}^{-1} x_{i(t)}(t)}\right) \\
&= \sum_{t:w_t<1} \min\left(1, \frac{3g_t}{p} w_t x_{i(t)}(t)^\top B_{i(t)}^{-1} x_{i(t)}(t)/\gamma\right) \\
&\le \sum_{t:w_t<1} \min\left((1+\frac{3g_t}{p\gamma}), (1+\frac{3g_t}{p\gamma}) w_t x_{i(t)}(t)^\top B_{i(t)}^{-1} x_{i(t)}(t)\right) \\
&= \sum_{t:w_t<1} (1+\frac{3g_t}{p\gamma}) \min\left(1, w_t x_{i(t)}(t)^\top B_{i(t)}^{-1} x_{i(t)}(t)\right)
\end{aligned}$$

where the first equation holds due to the definition of weight w_t. Now, we assume the rounds with weight $w_t < 1$ can be listed as $\{t_1, \cdots, t_m\}$ for simplicity. In addition, we introduce the auxiliary vector x_i' as $x_i' = \sqrt{w_{t_i}} x_{i(t_i)}(t_i)$ and matrix B_i' as

$$B_i' = I + \sum_{j=1}^{i-1} w_{t_j} x_{i(t_j)}(t_j) x_{i(t_j)}(t_j)^\top = I + \sum_{j=1}^{i-1} x_j' \left(x_j'\right)^\top.$$

We have $\left(B_i'\right)^{-1} \succeq B_i^{-1}$. Therefore, for each $i \in [m]$, we have

$$x_{i(t_i)}(t_i)^\top \left(B_i'\right)^{-1} x_{i(t_i)}(t_i) \ge x_{i(t_i)}(t_i)^\top B_i^{-1} x_{i(t_i)}(t_i)$$

Now we have

$$\begin{aligned}
&\sum_{i=1}^{m} \min\left(1, w_{t_i} x_{i(t_i)}(t_i)^\top B_{i(t_i)}^{-1} x_{i(t_i)}(t_i)\right) \\
&\le \sum_{i=1}^{m} \min\left(1, w_{t_i} x_{i(t_i)}(t_i)^\top (B_i')^{-1} x_{i(t_i)}(t_i)\right) \\
&= \sum_{i=1}^{m} \min\left(1, (x_i')^\top \left(B_i'\right)^{-1} x_i'\right) \\
&\le 2d\ln(1+T),
\end{aligned}$$

Then we have

$$\begin{aligned}
I_2 &\le \sum_{t:w_t<1} (2+\frac{3g_t}{p\gamma}) \min\left(1, w_t x_{i(t_i)}(t_i)^\top B_{i(t_i)}^{-1} x_{i(t_i)}(t_i)\right) \\
&\le 2d(1+\frac{3g_t}{p\gamma})\ln(1+T).
\end{aligned}$$

Recalling the definitions of p and g_T, by definition $g_T = O\left(d\sqrt{\ln\left(\frac{T}{\delta}\right)} + C\gamma\right)$. Substituting the above, we get

$$\begin{aligned}
&\sum_{t=1}^{T} \text{regret}'(t) \\
&= O\left(e^{(1+\frac{C\gamma}{\sqrt{d}})^2}\left(d\sqrt{\ln\left(\frac{T}{\delta}\right)} + C\gamma\right)\cdot\sqrt{dT\ln T} + e^{(1+\frac{C\gamma}{\sqrt{d}})^2}\left(d\sqrt{\ln\left(\frac{T}{\delta}\right)} + C\gamma\right)\frac{d}{\gamma}\ln T + 2d\ln T\right) \\
&= O\left(de^{(1+\frac{C\gamma}{\sqrt{d}})^2}\sqrt{dT\ln T\ln\left(\frac{T}{\delta}\right)} + C\gamma e^{(1+\frac{C\gamma}{\sqrt{d}})^2}\sqrt{dT\ln T} + \frac{d^2 e^{(1+\frac{C\gamma}{\sqrt{d}})^2}}{\gamma}\ln T\sqrt{\ln\left(\frac{T}{\delta}\right)} + \right. \\
&\left. Cde^{(1+\frac{C\gamma}{\sqrt{d}})^2}\ln T\right)
\end{aligned}$$

Also, because $E^{\mu}(t)$ holds for all t with probability at least $1-\frac{\delta}{2}$ (see Lemma B.2), $\text{regret}'(t) = \text{regret}(t)$ for all t with probability at least $1-\frac{\delta}{2}$. Hence, with probability $1-\delta$,

$$\begin{aligned}
&\mathcal{R}(T) = \sum_{t=1}^{T}\text{regret}(t) = \sum_{t=1}^{T}\text{regret}'(t) \\
&= O\left(de^{(1+\frac{C\gamma}{\sqrt{d}})^2}\sqrt{dT\ln T\ln\left(\frac{T}{\delta}\right)} + C\gamma e^{(1+\frac{C\gamma}{\sqrt{d}})^2}\sqrt{dT\ln T} + \frac{d^2 e^{(1+\frac{C\gamma}{\sqrt{d}})^2}}{\gamma}\ln T\sqrt{\ln\left(\frac{T}{\delta}\right)}\right. \\
&\left. + Cde^{(1+\frac{C\gamma}{\sqrt{d}})^2}\ln T\right).
\end{aligned}$$

Choose $\gamma = \sqrt{d}/C$, its regret can be upper bounded by $\mathcal{R}(T) = O\left(d\sqrt{dT\ln T\ln\left(\frac{T}{\delta}\right)} + Cd\sqrt{d}\ln T\sqrt{\ln\left(\frac{T}{\delta}\right)}\right)$.

□

B.2 Proof of Lemma B.2

Proof. First, the probability bound for $E^{\mu}(t)$ can be directly obtained from Lemma 1 in Agrawal and Goyal (2013).

Now we bound the probability of event $E^{\mu}(t)$. We use Lemma C.3 with $m_t = \sqrt{w_t}x_{i(t)}(t), \epsilon_t = \sqrt{w_t}(r^0_{i(t)}(t) - x_{i(t)}(t)^T\mu), \mathcal{H}'_t = (a(s+1), m_{s+1}, \epsilon_s : s \le t)$. By the definition of $\mathcal{H}'_t, m_t$ is $\mathcal{H}'_{t-1}$-measurable, and ϵ_t is $\mathcal{H}'_t$-measurable. ϵ_t is conditionally σ-sub-Gaussian due to $\sqrt{w_t} \le 1$ and the problem setting, and is a martingale difference process:

$$\mathbb{E}\left[\sqrt{w_t}\epsilon_t \mid \mathcal{H}'_{t-1}\right] = \mathbb{E}\left[\sqrt{w_t}r^0_{i(t)}(t) \mid x_{i(t)}(t), i(t)\right] - \sqrt{w_t}x_{i(t)}(t)^T\mu = 0$$

We denote

$$\begin{aligned}
M_t &= I_d + \sum_{s=1}^{t} m_s m_s^T = I_d + \sum_{s=1}^{t} w_t x_{i(s)}(s)x_{i(s)}(s)^T \\
\xi_t &= \sum_{s=1}^{t} m_s\epsilon_s = \sum_{s=1}^{t} w_t x_{i(s)}(s)\left(r^0_{i(s)} - x_{i(s)}(s)^T\mu\right)
\end{aligned}$$

Note that $B(t) = M_{t-1}$, and $\hat{\mu}(t) - \mu = M_{t-1}^{-1}\left(\xi_{t-1} - \mu + \sum_{s=1}^{t} w_s x_{i(s)}(s)c(s)\right)$. Let for any vector $y \in \mathbb{R}$ and

matrix $A \in \mathbb{R}^{d\times d}$, $\|y\|_A$ denote $\sqrt{y^T A y}$. Then, for all i,

$$\begin{aligned}\left|x_i(t)^T\hat{\mu}(t) - x_i(t)^T\mu\right| &= \left|x_i(t)^T M_{t-1}^{-1}\left(\xi_{t-1} - \mu + \sum_{s=1}^{t} w_s x_{i(s)}(s)c(s)\right)\right| \\ &\le \|x_i(t)\|_{M_{t-1}^{-1}} \|\xi_{t-1} - \mu + \sum_{s=1}^{t} w_s x_{i(s)}(s)c(s)\|_{M_{t-1}^{-1}} \\ &\le (\|\xi_{t-1} - \mu\|_{M_{t-1}^{-1}} + \sum_{s=1}^{t} |w_s||c(s)|\|x_{i(s)}(s)\|_{B(t)^{-1}}) \\ &\|x_i(t)\|_{B(t)^{-1}}\end{aligned}$$

The inequality holds because M_{t-1}^{-1} is a positive definite matrix. Using Lemma C.3 , for any $\delta' > 0, t \ge 1$, with probability at least $1-\delta'$,

$$\|\xi_{t-1}\|_{M_{t-1}^{-1}} \le \sigma\sqrt{d\ln\left(\frac{t}{\delta'}\right)}$$

Therefore, $\|\xi_{t-1} - \mu\|_{M_{t-1}^{-1}} \le R\sqrt{d\ln\left(\frac{t}{\delta'}\right)} + \|\mu\|_{M_{t-1}^{-1}} \le R\sqrt{d\ln\left(\frac{t}{\delta'}\right)} + 1$. By the definition of w_k, we also note that $\sum_{s=1}^{t} |w_s||c(s)|\|x_{i(s)}(s)\|_{B_t^{-1}} \le \gamma\sum_{s=1}^{t}|c(s)| \le C\gamma$. Substituting $\delta' = \frac{\delta}{t^2}$, we get that with probability $1-\frac{\delta}{t^2}$, for all i,

$$\begin{aligned}\left|x_i(t)^T\hat{\mu}(t) - x_i(t)^T\mu\right| &\le \|x_i(t)\|_{B(t)^{-1}} \cdot \left(R\sqrt{d\ln\left(\frac{t}{\delta'}\right)} + 1 + C\gamma\right) \\ &\le \|x_i(t)\|_{B(t)^{-1}} \cdot \left(R\sqrt{d\ln\left(t^3\right)\ln\left(\frac{1}{\delta}\right)} + 1 + C\gamma\right) \\ &= g_t s_i(t).\end{aligned}$$

This proves the bound on the probability of $E^{\mu}(t)$. □

B.3 Proof of Lemma B.3

Proof. Given event $E^{\mu}(t)$, $\left|x_{i^*(t)}(t)^T\hat{\mu}(t) - x_{i^*(t)}(t)^T\mu\right| \le g_t s_{t,i^*(t)}(t)$. And, since Gaussian random variable $\theta_{i^*(t)}(t)$ has mean $x_{i^*(t)}(t)^T\hat{\mu}(t)$ and standard deviation $v_t s_{i^*(t)}(t)$, using Lemma C.1,

$$\begin{aligned}&\Pr\left(\theta_{i^*(t)}(t) \ge x_{i^*(t)}(t)^T\mu \mid \mathcal{H}_{t-1}\right) \\ =&\Pr\left(\frac{\theta_{i^*(t)}(t) - x_{i^*(t)}(t)^T\hat{\mu}(t)}{v_t s_{t,i^*(t)}} \ge \frac{x_{i^*(t)}(t)^T\mu - x_{i^*(t)}(t)^T\hat{\mu}(t)}{v_t s_{t,i^*(t)}} \mid \mathcal{H}_{t-1}\right) \\ \ge&\frac{1}{4\sqrt{\pi}}e^{-Z_t^2}\end{aligned}$$

where

$$\begin{aligned}|Z_t| &= \left|\frac{x_{i^*(t)}(t)^T\mu - x_{i^*(t)}(t)^T\hat{\mu}(t)}{v_t s_{i^*(t)}(t)}\right| \\ &\le \frac{g_t s_{i^*(t)}(t)}{v_t s_{i^*(t)}(t)} \\ &= \frac{\left(\sigma\sqrt{d\ln\left(\frac{t^3}{\delta}\right)} + 1 + C\gamma\right)}{\sigma\sqrt{9d\ln\left(\frac{t}{\delta}\right)}} \\ &\le 1 + \frac{C\gamma}{\sqrt{d}}\end{aligned}$$

Therefore

$$\Pr\left(\theta_{i^*(t)}(t) \geq x_{i^*(t)}(t)^T\mu \mid \mathcal{H}_{t-1}\right) \geq \frac{1}{4e^{(1+\frac{C\gamma}{\sqrt{d}})^2}\sqrt{\pi}}$$

□

B.4 Proof of Lemma B.6

Lemma B.7. *For any filtration $\mathcal{H}_{t-1}$ such that $E^\mu(t)$ is true,*

$$\Pr\left(i(t) \notin C(t) \mid \mathcal{H}_{t-1}\right) \geq p - \frac{1}{t^2}.$$

This Lemma is from Lemma 4 in Agrawal and Goyal (2013). By using the Lemma B.2, we get that when both events $E^\mu(t)$ and $E^\theta(t)$ hold, for all $j \in C(t), \theta_j(t) \leq b_j(t)^T\mu + g_t s_{t,j}$. Also, by Lemma B.3, we have that if $E^\mu(t)$ is true,$\Pr\left(\theta_{i^*(t)}(t) > x_{i^*(t)}(t)^T\mu \mid \mathcal{H}_{t-1}\right) \geq p$. Then directly following the proof of Lemma 3 in Agrawal and Goyal (2013) we can obtain our result.

Lemma B.8. *For any filtration $\mathcal{H}_{t-1}$ such that $E^\mu(t)$ is true,*

$$\mathbb{E}\left[\Delta_{i(t)}(t) \mid \mathcal{H}_{t-1}\right] \leq \min\{\frac{3g_t}{p}\mathbb{E}\left[s_{i(t)}(t) \mid \mathcal{H}_{t-1}\right] + \frac{2g_t}{pt^2}, 1\}$$

This Lemma is from Lemma 4 in Agrawal and Goyal (2013). Using Lemma B.7, for any $\mathcal{H}_{t-1}$ such that $E^\mu(t)$ is true, we have $\Pr\left(i(t) \notin C(t) \mid \mathcal{H}_{t-1}\right) \geq p - \frac{1}{t^2} = \frac{1}{4e^{(1+\frac{C\gamma}{\sqrt{d}})^2}\sqrt{\pi}} - \frac{1}{t^2}$. Also, by Lemma B.2 we have that on the events $E^\mu(t)$ and $E^\theta(t)$, $\theta_i(t) \leq x_i(t)^T\mu + g_t\|x_i(t)\|_{B(t)^{-1}}$. Using these two facts, by directly following the proof in Lemma 4 of Agrawal and Goyal (2013) we immediately obtain our needed result.

Proof. We need to prove that for all $t \in [1, T]$, and any $\mathcal{H}_{t-1}, \mathbb{E}\left[Y_t - Y_{t-1} \mid \mathcal{H}_{t-1}\right] \leq 0$, i.e.

$$\mathbb{E}\left[\text{regret}'(t) \mid \mathcal{H}_{t-1}\right] \leq \min\{\frac{3g_t}{p}\mathbb{E}\left[s_{i(t)}(t) \mid \mathcal{H}_{t-1}\right] + \frac{2g_t}{pt^2}, 1\}$$

Note that whether $E^\mu(t)$ is true or not is completely determined by $\mathcal{H}_{t-1}$. If $\mathcal{H}_{t-1}$ is such that $E^\mu(t)$ is not true, then $\text{regret}'(t) = \text{regret}(t) \cdot I\left(E^\mu(t)\right) = 0$, and the above inequality holds trivially. And, for $\mathcal{H}_{t-1}$ such that $E^\mu(t)$ holds, the inequality follows from Lemma B.8. □

C Inequalities

Lemma C.1. *For a Gaussian distributed random variable Z with mean m and variance σ^2, for any $z \geq 1$,*

$$\frac{1}{2\sqrt{\pi}z}e^{-z^2/2} \leq \Pr(|Z - m| > z\sigma) \leq \frac{1}{\sqrt{\pi}z}e^{-z^2/2}.$$

Lemma C.2 (Azuma-Hoeffding inequality)**.** *If a super-martingale $(Y_t; t \geq 0)$, corresponding to filtration $\mathcal{H}_t$, satisfies $|Y_t - Y_{t-1}| \leq C_t$ for some constant C_t, for all $t = 1, \ldots, T$, then for any $a \geq 0$,*

$$\Pr\left(Y_T - Y_0 \geq a\right) \leq e^{-\frac{a^2}{2\sum_{t=1}^T C_t^2}}$$

Lemma C.3 (Abbasi-Yadkori et al. (2011))**.** *Let $(\mathcal{H}'_t; t \geq 0)$ be a filtration, $(m_t; t \geq 1)$ be an $\mathbb{R}^d$-valued stochastic process such that m_t is $\left(\mathcal{H}'_{t-1}\right)$-measurable, $(\eta_t; t \geq 1)$ be a real-valued martingale difference process such that η_t is $(\mathcal{H}'_t)$-measurable. For $t \geq 0$, define $\xi_t = \sum_{\tau=1}^t m_\tau\eta_\tau$ and $M_t = I_d + \sum_{\tau=1}^t m_\tau m_\tau^T$, where I_d is the d-dimensional identity matrix. Assume η_t is conditionally R-sub-Gaussian. Then, for any $\delta' > 0, t \geq 0$, with probability at least $1 - \delta'$,*

$$\|\xi_t\|_{M_t^{-1}} \leq R\sqrt{d\ln\left(\frac{t+1}{\delta'}\right)},$$

where $\|\xi_t\|_{M_t^{-1}} = \sqrt{\xi_t^T M_t^{-1}\xi_t}$.

D Discussion on Technical Challenge in Developing Robust TS

We clarify that we solve unique challenges in the theoretical analysis when applying robust UCB-based techniques to TS algorithms. We use the stochastic MAB as an example to show the challenges. First of all, the UCB exploration strategy is fundamentally different from that of TS. The analysis on UCB is centered on the properties of confidence intervals, while the analysis on TS is centered on the posteriors. It is infeasible to directly extend the proof on UCB to TS. In addition, the confidence intervals have the same meaning with and without an attack. However, the concept of posteriors is challenged by the poisoning attack because it is only defined on the true rewards that are no longer available under the attack. This makes the understanding and proof for robust TS algorithms non-trivial. Next, we explain the challenges we encounter in the proof.

We adopt the proof framework from Agrawal and Goyal (2012). The framework defines a good event $E_i^{\mu}(t)$ to be true when the empirical mean does not significantly exceed the true mean, and another good event $E_i^{\theta}(t)$ to be true where the sampling θ_i remains reasonably close to the empirical mean. When the good events hold, the chance of pulling the optimal arm is high. When these good events do not hold, it is probably the case when a suboptimal arm has been pulled for sufficient times, which also leads to limited regret. This framework faces new challenges when being applied to the learning scenario under the data poisoning attack.

Due to the presence of an adversary, the empirical mean no longer serves as an estimator derived from a distribution centered on the true mean. Under our design, the sampled (θ_i) no longer adheres to a distribution centered on the empirical mean. Consequently, the good events we construct have a different meaning. Since the empirical mean $\hat{\mu}$ is influenced by the attacker, the probability distribution of $\hat{\mu}$ depends on the attack strategy, which can be arbitrary and unknown to the agent. We need to lower bound the probability of the good event $\hat{\mu_i} < \mu_i + \Delta_i/3$ in this case. Under the attack, the posteriors of the arms become unknown to the agent. The sampling (θ_i) is sampled from a pseudo posterior designed by our algorithm, and its relation to the true posterior is uncertain due to the attack. We need to lower bound the probability that the sampling (θ_i) in this case is bounded by $\theta_i \leq \mu_1 - \Delta_i/3$.

For the case where these events do not hold on a sub-optimal, the idea is to prove that with a high probability p, this case happens when the sub-optimal arm is not pulled for sufficient times N. However, the relation between p and N becomes very different under the attack. We carefully design a modified value of N based on the knowledge of corruption level C, so that we can prove the value of p is still high while the value of N is bounded. This allows our analysis to achieve a near-optimal guarantee on the regret bound.

In conclusion, the approach in our work results from careful design and theoretical analysis, and it cannot be straightforwardly derived from the original TS algorithm or alternative analyses of the robust OFU algorithm.

E Thompsom Sampling Algorithms

We provide the formal definitions of the standard Thompson sampling algorithms for the stochastic bandit setting and linear contextual bandit setting in Algorithm 3 and 4. We refer the interested readers to Agrawal and Goyal (2013, 2017) for a more comprehensive explanation of the two algorithms. In addition, we list several advantages of the Thompson sampling strategy compared to OFUL as follows.

- **Utilizing prior information**: Thompson sampling algorithms utilize and benefit from the prior information about the arms.

- **Easy to implement**: While the regret of a UCB algorithm depends critically on the specific choice of upper-confidence bound, Thompson sampling depends only on the best possible choice. This becomes an advantage when there are complicated dependencies among actions, as designing and computing with appropriate upper confidence bounds present significant challenges Russo et al. (2018). In practice, Thompson sampling is usually easier to implement Chapelle and Li (2011).

- **Stochastic exploration**: Thompson sampling is a random exploration strategy, which could be more resilient under some bandit settings Lancewicki et al. (2021).

Algorithm 3 Thompson Sampling for Stochastic Bandits

For each arm $i = 1, \ldots, N$ set $k_i = 0, \hat{\mu}_i = 0$
for $t = 1, 2, \ldots,$ **do**
  For each arm $i = 1, \ldots, N$, sample $\theta_i(t)$ from the $\mathcal{N}\left(\hat{\mu}_i, \frac{1}{k_i+1}\right)$ distribution.
  Play arm $i(t) := \arg\max_i\{\theta_i(t)\}$ and observe reward r_t
  Set $\hat{\mu}_{i(t)} := \frac{\hat{\mu}_{i(t)}k_{i(t)}+r_t}{k_{i(t)}+1}, k_{i(t)} := k_{i(t)} + 1$
end for

Algorithm 4 Thompson Sampling for Linear Contextual Bandits

Set $B(1) = I_d, \hat{\mu} = 0_d, f = 0_d$.
for $t = 1, 2, \ldots,$ **do**
  Sample $\tilde{\mu}(t)$ from distribution $\mathcal{N}\left(\hat{\mu}, B(t)^{-1}\right)$.
  Play arm $i(t) := \arg\max_i x_i(t)^T\tilde{\mu}(t)$, and observe reward r_t.
  Update $B(t+1) = B(t) + x_i(t)x_i(t)^T, f = f + x_i(t)r_t, \hat{\mu} = B(t)^{-1}f$.
end for

F Comparison to The Algorithm With Regret Adaptive to Corruption Level

Here, taking the stochastic MAB setting as an example, we discuss and compare our algorithm against the algorithms whose regret is adaptive to the corruption level C under the attack. He et al. (2022) proves that if an algorithm's regret bound when there is no attack is R_0, its regret is $\Omega(T)$ when at corruption level $C \geq R_0/K$. For our algorithm, we choose $R_0 = O(\sqrt{NT})$ so that the regret bound is of the same order as the original Thompson sampling. Our algorithm guarantees that the regret bound is $O(\sqrt{NT})$ when $C = O(\sqrt{T/N})$, and $O(T)$ when $C = \Omega(\sqrt{T/N})$, which is tight. Suppose an algorithm has a regret bound adaptive to C, which can be denoted as a strictly monotonically increasing function $f(C)$. Consider the function achieves the tight regret bound at $f(0) = R_0$ and $f(C) = \Omega(T)$ when $C \geq R_0/K$ for some R_0, which are the same as our algorithm. However, in this case, the regret bound of the algorithm at the region in the middle $0 < C < R_0/K$ satisfies $f(C) > R_0$, which is worse than the guarantee of our algorithm. As a concrete example, we compare our algorithm against the PE-WR algorithm proposed by Wang et al. (2024). The bound on regret of PE-WR is adaptive to C. When $C = O(T^\alpha)$ with $1/4 < \alpha < 1/2$, the regret upper bound of PE-WR becomes $O(T^{2\alpha})$, which is greater than that of RTS $O(\sqrt{T})$. When $C = O(T^\alpha)$ with $\alpha \leq 1/4$ or $\alpha > 1/2$, the regret upper bounds of both algorithms become the same: $O(\sqrt{T})$ or $O(T)$.

As an empirical evidence, we present the empirical comparison between our algorithm and PE-WR again in Fig. 5. We find that under the oracle attack we test with, our algorithm is more robust than the PE-WR algorithms. In the case when the corruption level C is unknown, we observe that the regret of PE-WR increases much faster than our algorithm as the corruption level increases. Particularly, under the oracle-MAB attack, the PE-WR algorithm only achieves better performance than the vanilla Thompson sampling when the corruption level is high. When the corruption level is small, its regret is even higher than vanilla Thompson sampling, suggesting that the algorithm is not practical. These observations agree with our discussion above that having a regret adaptive to the unknown corruption level decreases the performance of an algorithm.

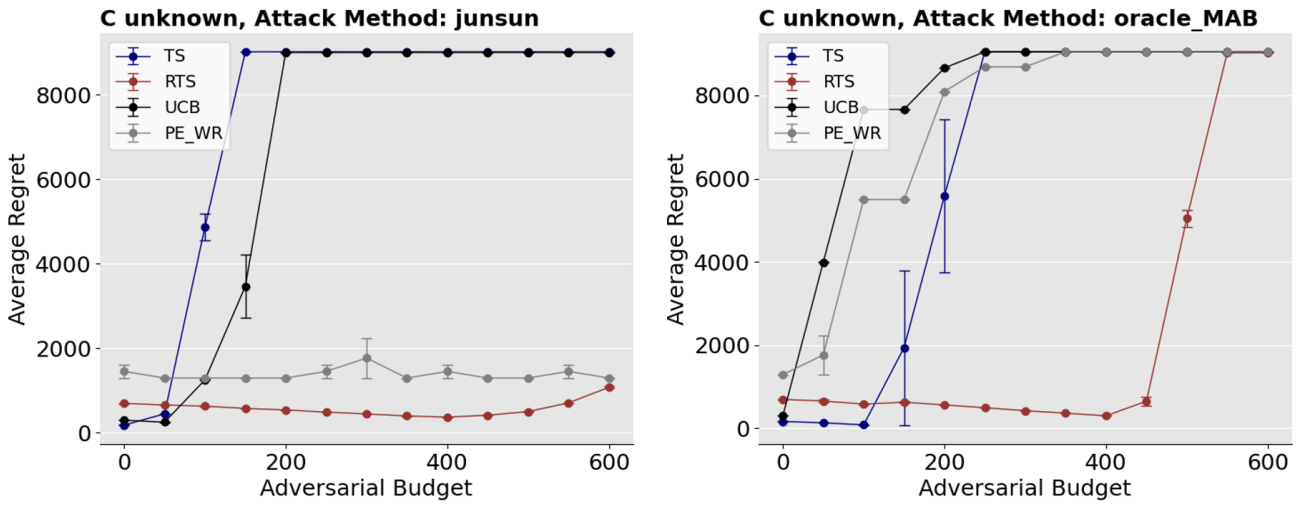

Figure 5: The total regret of different algorithms under attacks at different corruption levels in the stochastic bandit setting. The regret bound of the PE-WR algorithm is adaptive to the corruption level C.