Byzantine-Robust Federated Linear Bandits

Ali Jadbabaie, Haochuan Li, Jian Qian and Yi Tian

Abstract—In this paper, we study a linear bandit optimization problem in a federated setting where a large collection of distributed agents collaboratively learn a common linear bandit model. Standard federated learning algorithms applied to this setting are vulnerable to Byzantine attacks on even a small fraction of agents. We propose a novel algorithm with a robust aggregation oracle that utilizes the geometric median. We prove that our proposed algorithm is robust to Byzantine attacks on fewer than half of agents and achieves a sublinear \( \tilde{O}(T^{3/4}) \) regret with \( O(\sqrt{T}) \) steps of communication in \( T \) steps. Moreover, we make our algorithm differentially private via a tree-based mechanism. Finally, if the level of corruption is known to be small, we show that using the geometric median of mean oracle for robust aggregation further improves the regret bound.

I. INTRODUCTION

Recommendation systems have been a workhorse of e-commerce [1] and operations management applications [2] for more than a decade. The explosion of interest in personalized recommendation systems, however, has raised critical ethics and privacy issues. These trends, together with recent advances in federated and distributed computation, have given rise to new challenges and opportunities for the design of new recommendation systems that are developed using a secure, private, and federated architecture.

A key ingredient of such a system, at its core would be a bandit optimization engine. To this end, the current paper is motivated by the consideration of data corruption in a federated recommendation system. The recommendation system is modeled by a linear bandit with time-varying decision sets [3, 4, 5]. The data corruption is modeled by the Byzantine attack [6], a famous error model in distributed systems where parts of the system fail and there is imperfect information about the occurrences of the failures. How does one design provably robust algorithms in such a scenario?

More specifically, consider the scenario where one has to make recommendations to many devices. It is natural to assume that a device is continually used by the same user and that the users at different devices share similarity (e.g., from the same user group) so that at time step \( t \), the decision sets \( D_i^t \) for the devices are drawn i.i.d. from a distribution where \( i \) denotes a device index. The distribution is unknown and can change over time, modeling the fact that the preferences of the user group may be influenced by certain events as time goes by. We take the linear bandit model [3], according to which after making recommendation \( x_i^t \in D_i^t \), the reward we receive satisfies \( \mathbb{E}[r_i^t|x_i^t] = x_i^t \cdot \theta^t \). Such a model is a special case of federated linear bandits [7, 8, 9].

We consider a centralized federated learning setup [10, 11, 12], where devices are distributed and communicate with a central controller. Either due to noncooperative user behaviors or due to hijacking of the device by some adversary, the communications from some devices to the controller may be corrupted. Hence, it is vital that the federated recommendation system is robust to such corruptions. Here we consider a rather general and classical data corruption scheme called the Byzantine attack [6], where the corrupted information is arbitrary and we have no knowledge about whether the corruption happens at a particular device. Such a scenario has been considered in federated optimization [13, 14], where the performance of an algorithm is measured by the convergence rate. However, it is unclear how federated recommendation systems can be made robust to such attacks, where the performance of an algorithm is measured by the notion of regret.

An immediate question is how we should define regret in such a scenario. Since corrupted devices may fail arbitrarily under the Byzantine attack, a reasonable way is to consider the regret defined on the uncorrupted devices, which we call robust regret. A robust algorithm is then one that achieves sublinear robust regret. Since the controller has no information about which device has failed, it is challenging to design algorithms robust to the Byzantine attack.

In this paper, we design an algorithm that is robust to such attacks under the above federated linear bandit model. Notably, we show that so long as more than half of the devices are consistently reliable, our algorithm, called Byzantine-UCB, achieves \( \tilde{O}(dNT^{3/4}) \) robust regret for \( N \) federated linear bandits of dimension \( d \) in \( T \) steps with \( O(\sqrt{T}) \) steps of communication. Essential to achieve robustness is the i.i.d. assumption on the decision sets for different devices. Under such an assumption, reliable information can be obtained via robust estimation [15, 16], specifically, by using the (geometric) median estimator in place of the mean estimator [17].

Although existing works show that geometric median provably robustifies the convergence of federated optimiza-
tion, things are very different for the federated bandit problem. Unlike optimization where the geometric median is used to robustly estimate the mean of gradients, the linear bandit problem does not involve gradients. Instead, the challenge of the bandit problem is the well-known exploration-exploitation dilemma, where the agent attempts to acquire new knowledge (called “exploration”) or to optimize its decisions based on existing knowledge (called “exploitation”). It becomes more challenging in a federated setting with Byzantine attacks.

The Byzantine-UCB algorithm has the advantage of being agnostic to the proportion of devices that are corrupted. If the corruption proportion \( \alpha \) is small and its upper bound is known, another robust estimation can be obtained by the median of mean approach [18] to further reduce the regret. Specifically, the algorithm Byzantine-UCB-MoM, with a different aggregation oracle based on median of mean, interpolates the robust regret between \( \tilde{O}(dNT^{1/2}) \) and \( \tilde{O}(dNT^{3/4}) \) for \( 0 \leq \alpha \leq 1/2 \).

Privacy preservation is another concern in federated learning [19]. Our algorithms, equipped with the tree-based mechanism [20, 21], simultaneously achieve differential privacy and a slightly worse regret with an extra factor of \( \sqrt{d} \). Our algorithms, equipped with the tree-based mechanism [19], simultaneously achieve differential privacy and a slightly worse regret with an extra factor of \( \sqrt{d} \).

Two recent papers [22, 23] also study fault-tolerant federated bandit or reinforcement learning problems. However, we focus on very different settings and aspects. First, Dubey and Pentland [22] study the multi-armed bandit problem instead of the linear bandit problem we consider and use a different corruption model. Fan et al. [23] study the more general federated reinforcement learning problem. However, they only consider convergence to stationary points which is weaker than the regret bounds considered in this paper.

II. PRELIMINARIES

Notation. For any integer \( n \in \mathbb{N} \), let \([n]\) be the set \( \{1, \ldots, n\} \). For a vector \( x \), we use \( x_i \) to denote its \( i \)-th coordinate and \( \|x\|_2 \) to denote its \( \ell_2 \) norm. Given a positive semi-definite matrix \( A \), we denote \( \|x\|_A = \sqrt{x^T A x} \). For a matrix \( A \), we denote its spectral norm and Frobenius norm by \( \|A\|_2 \) and \( \|A\|_F \) respectively. Given two symmetric matrices \( A \) and \( B \) with the same size, we write \( A < B \) or \( B > A \) if \( B - A \) is positive definite. We also write \( A \leq B \) or \( B \geq A \) if \( B - A \) is positive semi-definite. Finally, we use the standard \( O(\cdot) \), \( \Theta(\cdot) \) and \( \Omega(\cdot) \) notation, with \( \tilde{O}(\cdot) \), \( \Theta(\cdot) \), and \( \Omega(\cdot) \) further hiding logarithmic factors.

A. Problem setup

a) Federated learning under Byzantine attacks: We consider the federated environment where there is one central server and \( N \) distributed agents. We assume that communication happens only between the controller and each agent. Let \( \mathcal{N} \) be the set of all agents with \( |\mathcal{N}| = N \). At each time \( t \in [T] \), several agents may be subject to a Byzantine attack and try to send arbitrarily corrupted information to the controller. Let \( \mathcal{N}_0 \) and \( \mathcal{N}_1 \) denote the set of noncorrupted and corrupted agents at time \( t \) respectively. Here we say an agent is reliable if it does not get attacked and corrupted otherwise.

Let \( \mathcal{N}_0 = \bigcap_t \mathcal{N}_0^t \) be the set of consistently reliable agents and \( \mathcal{N}_0 = |\mathcal{N}_0| \). Also define \( \mathcal{N}_1 = \bigcup_t \mathcal{N}_1^t \) as the complement of \( \mathcal{N}_0 \) and \( \mathcal{N}_1 = |\mathcal{N}_1| \). Assume at least half of the agents are consistently reliable, i.e., the corruption level \( \alpha \triangleq \mathcal{N}_1/N \) satisfies \( \alpha < 1/2 \).

b) Federated linear bandits: At every time \( t \in [T] \), each noncorrupted agent \( i \in \mathcal{N}_0^t \) is presented with a decision set \( D_i^t \subseteq \mathbb{R}^d \). It selects an action \( x_i^t \) from \( D_i^t \) and receives a reward \( r_i^t = \langle x_i^t, \theta^* \rangle + \eta_i^t \), where \( \theta^* \in \mathbb{R}^d \) is some unknown parameter and \( \eta_i^t \) is a noise. We assume the decision set and the true parameter are bounded, i.e., \( \max_{x \in D_i^t} \|x\|_2 \leq 1 \), \( \|\theta^*\|_2 \leq \sqrt{d} \). To see why the assumption that \( \|\theta^*\|_2 \leq \sqrt{d} \) is reasonable, consider multi-armed bandits, a special case of linear bandits, where standard bounded average reward assumption implies that \( \|\theta^*\|_2 = \mathcal{O}(\sqrt{d}) \).

The randomness of the model comes from \( D_i^t \) and \( \eta_i^t \) on which we make the following assumptions: at every time step \( t \), the pairs \( \{(D_i^t, \eta_i^t)\}_{i \in \mathcal{N}_0^t} \) are i.i.d. sampled from the an unknown distribution \( P_t \) conditioned on previous \( \{(D_i^s, \eta_i^s)\}_{i \in \mathcal{N}_0^s} \) for all \( s < t \). Also, assume that \( D_i^t \) and \( \eta_i^t \) are independent for each \( t \in [T] \) and \( i \in \mathcal{N}_0^t \). Let \( P^t_\alpha \) be the marginal distribution of \( \eta_i^t \). We assume \( P^t_\alpha \) is \( R \)-subGaussian. Note that the independence between \( D_i^t \) and \( \eta_i^t \) is assumed for ease of exposition and can be relaxed as in [3].

Since we should not expect to pull the right arm on the corrupted steps, the objective of the agents is thus to minimize the cumulative pseudo-regret on the steps where they are not attacked. Formally, we define the regret as follows:

\[
R_T = \sum_{t=1}^T \sum_{i \in \mathcal{N}_0^t} \left( \max_{x \in D_i^t} \langle x, \theta^* \rangle - \langle x_i^t, \theta^* \rangle \right). \tag{1}
\]

Note that in the presence of corruptions, if each agent learns its own problem without collaborating with others, the regret will be linear in \( T \), as shown in Proposition 1.

Proposition 1: For a given set of agents \( \mathcal{N} \), corruption level \( \alpha > 0 \), there exists an instance of a federated linear bandit problem with corruptions under our assumptions such that without communication, \( R_T \geq c \alpha N T \) for some absolute constant \( c > 0 \).

Proof: Consider the setting that \( D_i^t = \{-1, 1\} \) for every \( t \in [T] \) and \( i \in \mathcal{N}_0^t \). Let \( |\theta^*| = 1 \). For any \( i \in \mathcal{N}_1^t \), at each time \( t \), agent \( i \) is attacked with probability \( 1/2 \) independently with other time steps or other agents. When it is attacked, it receives a reward according to a fake parameter \( \theta^\text{fake} = -\theta^* \).

Then any algorithm on agent \( i \) which does not communicate with other agents or the controller cannot distinguish between the two models \( \theta^* = 1 \) or \( \theta^* = -1 \). Then we can always choose one of them to make the algorithm perform no better than a random guess. Therefore the regret on agent \( i \) is at least \( T \). Noting that there can be \( \alpha N \) such agents, we complete the proof.

Therefore, it is necessary to learn in a federated way. In the next section, we propose a federated algorithm which achieves a regret of \( \tilde{O}(T^{3/4}) \).
B. Robust aggregation

Standard federated learning algorithms cannot be applied in the current setting because they are vulnerable to Byzantine attacks. For example, Dubey and Pentland [7] proposed an algorithm for federated linear bandits without corruptions. In their algorithm, the updates collected from agents are aggregated by a simple arithmetic mean, which is known to be vulnerable to Byzantine attacks on even a single agent.

To robustify the algorithm, we utilize the geometric median in our aggregation rule. For a collection of vectors \( z_1, \ldots, z_n \in \mathbb{R}^d \), let \( g(z) = \frac{1}{n} \sum_{i=1}^n \| z - z_i \|_2 \). We define \( \text{GM}_{[n]}(z_i) = \arg\min_{z \in \mathbb{R}^d} g(z) \) as their geometric median. In practice, this minimization problem is usually solved approximately. Hence, we further define the \( \epsilon \)-approximate geometric median as an approximate solution \( \hat{z} \) satisfying \( g(\hat{z}) \leq \min_{z \in \mathbb{R}^d} g(z) + \epsilon \) which we will denote by \( \text{GM}_{[n]}^\epsilon(z_i) \). Note that if some attacked \( z_i \) is even not a vector in \( \mathbb{R}^d \), we view it as 0 in \( \mathbb{R}^d \) when computing the geometric median. We can also define the geometric median of matrices by replacing the \( \ell_2 \) vector norm with the Frobenius norm. Recently, Pillutla et al. [13] proposed a robust oracle based on a smoothed Weiszfeld algorithm which returns an approximate geometric median with only a small number of calls to the average oracle. We will also adopt this robust oracle in our algorithm.

III. The Byzantine-UCB Algorithm and Its Robust Regret Bound

In this section, we present our algorithm Byzantine-UCB with the achieved regret bound. In general, our algorithm obtains a regret bound using a reasonable amount of communication between the agents and the controller while being robust to Byzantine attacks. Before introducing Byzantine-UCB, we first introduce our general algorithmic framework (Algorithm 1) for Byzantine-robust federated linear bandit optimization.

A. Algorithmic framework.

In Algorithm 1, to reduce the amount of communication, we divide the \( T \) steps into \( K \) episodes of length \( L \), where \( K, L \in \mathbb{N} \). Assume \( T = KL \) exactly holds for simplicity; in general we can round up \( K \) to the closest integer. At the start of each episode, the controller synchronizes the parameters \( \theta_k \) and \( \Lambda_k \) with all agents. Define \( T_{\text{sync}} = \{ 1, L+1, \ldots, (K-1)L+1 \} \) as the set of steps when communication happens. For each \( k \in [K] \), define \( T_k = \{ (k-1)L+1, \ldots, kL \} \) as the set of steps between the \( k \)-th and \( (k+1) \)-th communication.

During the \( k \)-th episode, agent \( i \) runs in the same fashion as the celebrated LinUCB algorithm [3]. The idea is to construct a confidence region which contains \( \theta^* \) with high probability, and then to follow the principle of optimism in the face of uncertainty. Specifically, the confidence region is constructed as \( \Theta_k = \{ \theta \in \mathbb{R}^d : \| \theta - \theta_k \|_{\Lambda_k} \leq \beta_k \} \), where \( \beta_k \) is stored locally and specified by the algorithmic instantiation. Then, on receiving the decision set \( D_i^k \) from the environment, the agent picks the most optimistic choice

\[
x_i^k = \arg\max_{x \in D_i^k} \sup_{\theta \in \Theta_k}(x, \theta) = \arg\max_{x \in D_i^k} \langle x, \theta_k \rangle + \beta_k \| x \|_{\Lambda_k}^{-1}.
\]

At the end of the \( k \)-th episode, the central controller receives all the Gram matrices \( \{U_i^k\}_{i \in N} \) and the weighted feature sums \( \{ \bar{v}_i^k \}_{i \in N} \) for the \( k \)-th episode from the agents. The controller first checks if \( U_i^k \) is symmetric (if differential privacy is required, it further checks \( \| U_i^k \|_F, \| U_i^k \|_2 \leq L \)). If they are clearly corrupted, set \( U_i^k = 0 \) and \( u_i^k = 0 \). Then the controller updates the existing Gram matrices \( \widehat{V}_i^{k+1} \) and feature sums \( \widehat{v}_i^{k+1} \). The key to achieve Byzantine-robustness is
the robust aggregation oracle (Aggregate) which computes $\Lambda_k$ and $b_k$ from the sets $\{\hat{V}^i_k + \lambda_k I\}_{i \in \mathbb{N}}$ and $\{\hat{v}^i_k\}_{i \in \mathbb{N}}$. Then $\Lambda_{k+1}$ and the latest estimation $\theta_{k+1}$ are broadcast to all agents at the beginning of the $(k+1)$-th episode.

In Algorithm 1, we use the function Privatize to guarantee differential privacy. For now, we assume differential privacy is not required and first focus on the regret analysis. We will talk about privacy issues in detail in Section VI.

To tightly characterize how much the decisions taken $x^i_t$ vary from its expectation, we make the following assumption.

**Assumption 1:** Assume for every $t \in [T]$ and $i \in \mathcal{N}_t$, we have $\|x^i_t(x^i_t) - E[x^i_t(x^i_t)]\|_2^2 \leq \sigma^2$, where the randomness of $x^i_t = \arg\max_{x \in \mathcal{D}^i_t}(\langle x, \theta^* \rangle + \beta_k \|x\|_{\mathcal{D}^i_{t-1}})$ comes from the randomness in $\mathcal{D}^i_t$.

All theorems henceforth hold under Assumption 1. Note that $\|x^i_t\|_2 \leq 1$ directly implies $\|x^i_t(x^i_t) - E[x^i_t(x^i_t)]\|_2^2 \leq 4$. Thus in the worst case, $\sigma \leq 2$. On the other hand, if for every fixed $t \in [T]$, the decision sets for different agents are the same, we have $\sigma = 0$.

### B. The Byzantine-UCB algorithm.

Byzantine-UCB instantiates the algorithmic framework (Algorithm 1) with Aggregate chosen to be an oracle that computes the $\epsilon$-approximate geometric median of the input set and without the requirement of differential privacy. Its regret bound is shown in the following theorem.

**Theorem 2 (Robust regret bound of Byzantine-UCB):**

Let $C_\alpha = \frac{2-2\alpha}{1-2\alpha}$. For any given $\delta \in (0,1)$, let $t_\alpha = \log\left(\frac{2\alpha N T}{\delta}\right)$. Choose $\lambda_k = 2C_\alpha \epsilon + \max\{\lambda_0, \Lambda_1\}$ where $\lambda_0 = L$ and $\lambda_1 = 8\sqrt{La_\alpha}$. Choose $\beta_k = 3\sqrt{\lambda_k d} + \frac{4\sqrt{(k-1)Lda_\alpha (\sigma + R) + C_\alpha \epsilon}}{\sqrt{k}} + 2R\sqrt{\frac{d \ln N}{k}}$. Then with probability at least $1 - \delta$, the regret of Byzantine-UCB is bounded by $O\left(R \sqrt{d N T + N d \ln T} \left(\sqrt{L + C_\alpha \epsilon + C_\alpha \sigma \sqrt{T} + \frac{\sqrt{T} \ln C_\alpha (\sigma + R)}{L + C_\alpha \epsilon + C_\alpha \sigma \sqrt{T}}\right)\right)$.

In particular, if choosing $L = C_\alpha (\sigma + R) \sqrt{T} / \epsilon$ and $\epsilon$ is sufficiently small, then we have $R_T = O\left(d N T^{3/4}\right)$.

The above theorem implies a sublinear rate in $T$ and also demonstrates the relationship between the number of communication rounds $K$, the size of the corruption $\alpha$, the variance of the decision set $\sigma^2$ and the upper bound of the regret achieved by Algorithm 1. First, note that although the regret bound depends on $\alpha$, the algorithm is completely agnostic to it. Next, it is easy to see that the larger the size of the corruption or the variance term is, the larger the regret bound is. However, the dependence of regret upper bound on the number of communication rounds $K$ is more intricate. As long as $K = \Omega(\sqrt{T})$, the regret bound stays at $O(T^{3/4})$. However, if we further reduce $K$, the regret bound will increase. Therefore, the best number of communication rounds without affecting the convergence rate is $\Theta(\sqrt{T})$.

**Comparison to previous works.** When there are no corruptions, several existing works (e.g. [7]) achieve an $O(\sqrt{T})$ which is nearly optimal. However, it is not clear whether the $O(T^{3/4})$ regret bound under Byzantine attacks is optimal and we leave answering the question as future work. Note that our regret bound does not reduce to the $O(\sqrt{T})$ bound when the level of corruption $\alpha$ goes to 0 because the algorithm is agnostic to $\alpha$. We will provide a corruption level aware algorithm in Section V which improves the regret when an upper bound of $\alpha$ is known. The improved rate does reduce to $O(\sqrt{T})$ when $\alpha \to 0$.

The number of communications rounds in Theorem 2 is $O(\sqrt{T})$, whereas in [7] is $O(N \log T)$. Our communication complexity does not depend on $N$ but has a worse dependence on $T$ compared to theirs. They are able to obtain a $\log T$ communication because in their algorithm, the agent adaptively decides when to communicate with the controller. However, if the agents can be arbitrarily attacked, we can not really allow the agent to decide the communication rounds. Otherwise, an attacked agent may choose to communicate every round which is highly communication-inefficient. Therefore, reducing the communication complexity becomes more challenging in presence of corruptions. We will see in Section VI that it becomes even more challenging with privacy constraints.

### IV. ANALYSES FOR THEOREM 2

In this section, we provide the analyses for Theorem 2. For convenience, we introduce $W_k = \sum_{t=1}^{(k-1)L} \sum_{i \in \mathcal{N}_0} x^i_t(x^i_t)^\top, \ s_k = \sum_{t=1}^{(k-1)L} \sum_{i \in \mathcal{N}_0} x^i_t$ where the summations are over consistently noncorrupted agents only. Then the least square estimate of $\theta^*$ can be written as $\theta^*_k = W_k^{-1} s_k$ which is widely used in previous noncorrupted linear bandit algorithms [3]. However, since we do not know $\mathcal{N}_0$, the least square estimate is not computable. Instead, we use another estimate $\theta_k = \Lambda_k^{-1} b_k$ where $\Lambda_k = \lambda_k I + GM_{\mathcal{N}_0} (\hat{V}^k) = \lambda_k I + W_k / N_0 + E_k$, $b_k = GM_{\mathcal{N}_0} (\hat{v}^k) = s_k / N_0 + e_k$,

where $\lambda_k > 0$ is a time-varying regularization parameter to ensure the positive definiteness and also to control the regret. $E_k$ and $e_k$ are the error terms of using geometric median instead of arithmetic mean:

$$E_k \equiv GM_{\mathcal{N}_0} (\hat{V}^k) - \frac{1}{N_0} \sum_{t=1}^{(k-1)L} \sum_{i \in \mathcal{N}_0} x^i_t(x^i_t)^\top, \ e_k \equiv GM_{\mathcal{N}_0} (\hat{v}^k) - \frac{1}{N_0} \sum_{t=1}^{(k-1)L} \sum_{i \in \mathcal{N}_0} x^i_t.$$

We can bound these two error terms as follows.

**Lemma 3:** Using the same parameter choices as in Theorem 2, with probability at least $1 - \delta/2$, for all $k \in [K]$,

$$\|E_k\|_2 \leq 4C_\alpha \sigma \sqrt{(k-1)L} + C_\alpha \epsilon,$$

$$\|e_k\|_2 \leq 4C_\alpha (\sigma + R) \sqrt{(k-1)Ld} + C_\alpha \epsilon.$$
Proof: We first bound \( \|E_k\|_2 \). We have with probability at least \( 1 - \delta / 4 \),
\[
\|E_k\|_F = \|\sum_{i \in \mathcal{N}_0} V_i^k \|_F = 2C_0 \sigma \sqrt{2(1 - \delta)} + C_0 \epsilon ,
\]
where (i) is due to Lemma 6, (ii) is by triangle inequality and Jensen’s inequality on the convex operator \( \|\cdot\|_F \).

Then by [24, Theorem 1.8], with probability at least \( 1 - \delta / 4 \),
\[
\|E_k\|_2 = \|\sum_{i \in \mathcal{N}_0} V_i^k \|_2 = 2C_0 \sigma \sqrt{2(1 - \delta)} + C_0 \epsilon ,
\]
where (i) is due to Lemma 6, (ii) is by triangle inequality and Jensen’s inequality on the convex operator \( \|\cdot\|_F \).

Then we bound the three terms separately. First, we choose \( \lambda_k = K \sqrt{\sigma} \sqrt{T} \), where \( \lambda_0 = L \) and \( \lambda_1 = K \sqrt{\sigma} \sqrt{T} \). Then Lemma 3 guarantees \( \lambda_k \geq 2 \|E_k\|_2 \). It is easy to verify \( \Lambda_k \) is symmetric. It is also positive definite since \( \lambda_k \geq 2 \|E_k\|_2 \). Therefore \( \|\Lambda_k^{-1}\| \) is well-defined. Then,
\[
\|E_k\|_2 \leq 2C_0 \sigma \sqrt{2(1 - \delta)} + C_0 \epsilon .
\]

Then we can similarly bound \( \|E_k\|_2 \) with probability at least \( 1 - \delta / 8 \). We complete the proof by noting that the total failure probability considered in this lemma is less than \( \delta / 2 \).

Then we can bound the difference between \( \theta_k \) and \( \theta_k^{\text{be}} \) and thus also bound the difference between \( \theta_k \) and \( \theta^* \).

Lemma 4 (Approximation error): Using the same parameter choices as in Theorem 2, with probability at least \( 1 - 3\delta / 4 \), for all \( x \in \mathbb{R}^d \) and \( k \in [K] \), we have \( |x^T (\theta_k - \theta^*)| \leq \beta_k \|x\|_{\Lambda_k^{-1}} \).

Proof: For every \( x \in \mathbb{R}^d \), we have
\[
x^T (\theta_k - \theta^*) = x^T \Lambda_k^{-1} b_k - x^T \Lambda_k^{-1} \lambda_k \theta^*
= x^T \Lambda_k^{-1} \frac{\lambda_k}{N_0} + x^T \Lambda_k^{-1} e_k - x^T \Lambda_k^{-1} (\lambda_k I + E_k + \frac{W_k}{N_0}) \theta^*
= -x^T \Lambda_k^{-1} (\lambda_k I + E_k) \theta^* + x^T \Lambda_k^{-1} e_k + x^T \Lambda_k^{-1} (s_k - W_k) \theta^*
\]
\[\triangleq R_1 + R_2 + R_3.\]

Then we bound the three terms separately. First, we choose \( \lambda_k = 2C_0 \sigma + \max \{\lambda_0, \lambda_1 \sqrt{K} \} \), where \( \lambda_0 = L \) and \( \lambda_1 = K \sqrt{\sigma} \sqrt{T} \). Then Lemma 3 guarantees \( \lambda_k \geq 2 \|E_k\|_2 \). Therefore \( \|\Lambda_k^{-1}\| \) is well-defined. Then,
\[
|R_1| = |x^T \Lambda_k^{-1} (\lambda_k I + E_k) \theta^*| \leq \sqrt{\|x\|_{\Lambda_k^{-1}}.}
\]

Similarly, we can bound the second term
\[
|R_2| \leq \frac{4C_0 (\sigma + R) \sqrt{(k - 1) / 4} + C_0 \sigma \sqrt{\|x\|_{\Lambda_k^{-1}}.}}{\sqrt{\lambda_k}}.
\]

To bound \( R_3 \), first note that
\[
s_k - W_k \theta^* = \sum_{i=1}^{(k-1)/k} \sum_{i \in \mathcal{N}_0} x_i^k (x_i^k - (x_i^k)^\top \theta^*)
= \sum_{i=1}^{(k-1)/k} \sum_{i \in \mathcal{N}_0} x_i^k e_i.
\]

By [3, Theorem 1], with probability at least \( 1 - \delta / 4 \),
\[
\|s_k - W_k \theta^*\|_{(W_k + \lambda_0 N_0 I)/2} \leq 2R^2 dt
\]
for any \( k \). Note that since \( \lambda_k \geq 2 \|E_k\|_2 \), we have
\[
N_0 \Lambda_k - W_k - N_0 (\lambda_k I + E_k) \geq \frac{N_0}{2} \lambda_k I \geq \frac{N_0}{2} \lambda_0 I.
\]

Then we have
\[
(W_k + \frac{N_0}{2} \lambda_0 I)^{-1} \geq \frac{1}{N_0} \Lambda_k^{-1}
\]
and
\[
|R_3| \leq \frac{1}{N_0} \|x\|_{\Lambda_k^{-1}} \|s_k - W_k \theta^*\|_{\Lambda_k^{-1}}
\]
\[\leq \frac{1}{N_0} \|x\|_{\Lambda_k^{-1}} \|s_k - W_k \theta^*\|_{(W_k + \lambda_0 N_0 I)/2} \leq 2R \sqrt{\frac{\lambda_k}{N_0}} \|x\|_{\Lambda_k^{-1}}.
\]

Thus we can show that for every \( x \in \mathbb{R}^d \),
\[
|x^T (\theta_k - \theta^*)| \leq |R_1| + |R_2| + |R_3| \leq \beta_k \|x\|_{\Lambda_k^{-1}}.
\]

In previous papers like [3], a constant \( \beta \) is used to obtain a regret of \( \mathcal{O}(\sqrt{T}) \). Here, Lemma 4 shows that \( \beta_k = \mathcal{O}(1) \).

Therefore, we can obtain an \( \mathcal{O}(T^{3/4}) \) regret bound at best.

A. Bounding the regret

With some standard analyses, we can bound the regret by
\[
R_T \leq 2\beta_{\max} \sqrt{NT} \sqrt{\sum_{t=1}^{T} \sum_{i \in \mathcal{N}_0} (x_i^t)^\top \Lambda_k^{-1} x_i^t},
\]
where \( \beta_{\max} = \max_{k \in [K]} \beta_k \). To get an \( \mathcal{O}(T^{3/4}) \) regret bound, we also need to show
\[
\sum_{t=1}^{T} \sum_{i \in \mathcal{N}_0} (x_i^t)^\top \Lambda_k^{-1} x_i^t = \mathcal{O}(1).
\]

Previously works like [3] also bounded some quantity like this. But we have some additional issues here to deal with.

First, \( \Lambda_k \) contains corrupted data. We need to use Lemma 4 to deal with it. In addition, the summation is taken over \( \mathcal{N}_0 \) which is time-varying. We use a concentration argument to bound the difference between the summation over \( \mathcal{N}_0 \) and that over \( \mathcal{N}_0^T \). Finally, communication does not happen every step, which results in an additional error term. Due to this, we need more careful analyses and to choose \( \lambda_0 = L \). The full proof of Theorem 2 is deferred to Appendix B.

V. CORRUPTION LEVEL AWARE ALGORITHM AND ITS IMPROVED REGRET

So far we use the geometric median oracle for robust aggregation (Aggregate). One advantage of this choice is that it is agnostic to the corruption level \( \alpha \). However, it may also turn into a disadvantage if we know \( \alpha \) is small and we have a good estimate of its bound. This prior knowledge, if utilized effectively, can improve the performance of an algorithm. In this section, we introduce the geometric median of mean oracle, which can be viewed as an interpolation between geometric median and arithmetic mean. We then show that the resulting Byzantine-UCB-MoM algorithm,
based on an approximate geometric median of mean oracle, has a better robust regret bound.

Suppose $\alpha \leq 1/4$ is known. Right after the algorithm starts, the oracle randomly splits the set of all agents $\mathcal{N}$ into $P = 3N_1$ groups, $\mathcal{P} = \{G_i\}_{i \in [P]}$, as equally as possible. Therefore we have $\left[\frac{1}{3}\right] \leq |G_i| \leq \left[\frac{2}{3}\right]$. At each time $t \in [T]$, we say a group is noncorrupted if all its agents are noncorrupted; say it is corrupted otherwise. Then at the group level, we can also define the set of consistently noncorrupted groups $\mathcal{P}_0$ with $\mathcal{P}_0 = |\mathcal{P}_0|$ and its complement $\mathcal{P}_1$ with $\mathcal{P}_1 = |\mathcal{P}_1|$. Let $\gamma = P_1/P$ be the fraction of possibly corrupted groups. We know $\gamma \leq 1/3$ and thus $C_\gamma \leq 4$.

At each synchronization step, the oracle first computes arithmetic means within each group and then uses the approximate geometric median oracle to aggregate these means. Formally, given a set of vectors or matrices $\{z_j\}_{j \in \mathcal{N}}$, the oracle returns

$$GM_{1 \leq i \leq P} \left( \frac{1}{|G_i|} \sum_{j \in G_i} z_j \right).$$

A. The Byzantine-UCB-MoM algorithm.

Aware of $\alpha$, Byzantine-UCB-MoM instantiates the algorithmic framework (Algorithm 1) by choosing $\text{Aggregate}$ to be an $\epsilon$-approximate geometric median of mean oracle (2) and without the requirement of differential privacy. Since geometric median of mean is an interpolation between geometric median and arithmetic mean, the error between the geometric median of mean and the mean is smaller. Therefore, we obtain the following tighter regret bound when $\alpha$ is small.

**Theorem 5 (Regret Bound of Byzantine-UCB-MoM):** Suppose $\alpha \leq 1/4$. For any given $\delta \in (0, 1)$, let $\epsilon = \log \left( \frac{128\alpha N_T}{\delta} \right)$. Choose $\lambda_k = 8\epsilon + \max \left\{ \lambda_0, \lambda_1 \sqrt{k} \right\}$ where $\lambda_0 = L$ and $\lambda_1 = 128\alpha \sqrt{\alpha L}$. Choose $\beta_k = 3\sqrt{\lambda_k d} + \frac{64(\sigma + R)\sqrt{\alpha(k-1) Ld + \epsilon} \epsilon}{\sqrt{\lambda_k}} + 2R \sqrt{\frac{d}{\beta_k}}$.

Then with probability at least $1 - \delta$, the regret of Byzantine-UCB-MoM is bounded by

$$R_T = \mathcal{O} \left( R_{D\epsilon} \sqrt{NT} + ND \sqrt{T} \left( \sqrt{L + \epsilon + \sigma} \sqrt{\alpha T} + \frac{\sqrt{\alpha T (\sigma + R)}}{\sqrt{L + \epsilon + \sigma} \sqrt{\alpha T}} \right) \right).$$

In particular, if we choose $\epsilon$ small enough and $L = \max \{ (\sigma + R) \sqrt{\alpha T}, 1 \}$, then we have

$$R_T = \mathcal{O} \left( d N \left( \alpha^{1/4} T^{3/4} + \sqrt{T} \right) \right).$$

This bound is much tighter than that in Theorem 2 when $\alpha$ is very small. For example, when $\alpha = 0$, the oracle reduces to the arithmetic mean oracle and the regret is bounded by $\mathcal{O}(\sqrt{T})$, matching the lower bound in $T$ dependence.

VI. Differential privacy guarantees

In this section, we talk about the differential privacy guarantees. Note that in Algorithm 1, the function $\text{Privatize}$ is executed by the controller other than by each agent locally as in [7]. This is because when an agent is Byzantine-attacked, even after privatizing the data it intends to send to the controller, the attacker can still deprivatize it or even send other private information to the controller. Since in our algorithm, the controller has access to the original data of agents without privatizing, we assume the controller is trustable for all agents.

To guarantee differential privacy, we need to further make the following standard bounded reward assumption.

**Assumption 2:** $|r_i| \leq 1$ for every $t \in [T]$ and $i \in \mathcal{N}_t$.

As each user only trusts the controller and the agent that it is interacting with, the decision sets $\{D_i\}_{i \in [T]}$ and rewards $\{r_i\}_{i \in [T]}$ of each agent $i$ must be made private to all other agents. Moreover, when an agent is attacked, it may also send private information to the controller, which also needs to be made private. No matter if agent $i$ is attacked or not, it suffices to make each update it sends to the server private to other agents.

To guarantee convergence, the function $\text{Privatize}$ in our algorithm effectively returns

$$\tilde{V}_i^k = V_i^k + H_i^k, \quad \tilde{v}_i^k = v_i^k + h_i^k,$$

where $H_i^k$ and $h_i^k$ are some Gaussian noise according to the tree-based mechanism. To make the algorithm differentially private, based on the tree-based mechanism, we have with probability $1 - \delta/4$,

$$\|H_i^k\|_2, \|h_i^k\|_2 \leq BL,$$

where $B = \mathcal{O}(\sqrt{d})$, for every $i \in \mathcal{N}$ and $k \in [K]$. Note that the above bound is linear in $L$, the number of rounds between two communications. Therefore, the less frequently the communication happens, the harder to guarantee privacy, which makes it more challenging to reduce the communication complexity. Besides, the $\sqrt{d}$ dependence in $B$ makes the regret worst by such a factor.

VII. Conclusion

In this paper, we proposed a novel setup for linear bandit algorithms for use in modern recommendation systems with the following properties 1) A federated learning architecture: the algorithm allows the data to be stored in local devices; 2) Robustness to Byzantine attacks: the algorithm has sublinear regret even if some agents send arbitrarily corrupted messages to the controller; and 3) Differential privacy: the agents do not want to reveal their identities to other agents. Furthermore, we propose an algorithm that meets all three objectives and takes advantage of a known small proportion of corrupted agents to obtain an improved robust regret. All our algorithms achieve an $\mathcal{O}(T^{3/4})$ regret which does not match the $\Omega(\sqrt{T})$ lower bound. It is an interesting open problem to close this gap.

REFERENCES

[1] J. Schafer, J. Konstan, and J. Riedl. Recommender systems in e-commerce. In EC ’99, 1999.

[2] Thomas Asikis and George Lekakos. Operations research and recommender systems. In Sakae Yamamoto,
editor, Human Interface and the Management of Information. Information and Knowledge in Applications and Services, pages 579–589, Cham, 2014. Springer International Publishing. ISBN 978-3-319-07863-2.

[3] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. Advances in Neural Information Processing Systems, 24:2312–2320, 2011.

[4] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In Proceedings of the 19th international conference on World wide web, pages 661–670, 2010.

[5] 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, 2011.

[6] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. ACM Transactions on Programming Languages and Systems, 4(3):382–401, 1982.

[7] Abhimanyu Dubey and Alex Pentland. Differentially-private federated linear bandits. arXiv preprint arXiv:2010.11425, 2020.

[8] Chuanhao Li and Hongning Wang. Communication efficient federated learning for generalized linear bandits. ArXiv, abs/2202.01087, 2022.

[9] Ruiquan Huang, WeiQiang Wu, Jing Yang, and Cong Shen. Federated linear contextual bandits. ArXiv, abs/2110.14177, 2021.

[10] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. arXiv preprint arXiv:1912.04977, 2019.

[11] Ali Jadbabaie, Anuran Makur, and Devavrat Shah. Federated optimization of smooth loss functions. 2022.

[12] Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, Ali Jadbabaie, and Ramtin Pedarsani. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. ArXiv, abs/1909.13014, 2020.

[13] Krishna Pillutla, Sham M Kakade, and Zaid Harchaoui. Robust aggregation for federated learning. arXiv preprint arXiv:1912.13445, 2019.

[14] Zhaoxian Wu, Qing Ling, Tianyi Chen, and Georgios B Giannakis. Federated variance-reduced stochastic gradient descent with robustness to byzantine attacks. IEEE Transactions on Signal Processing, 68:4583–4596, 2020.

[15] Peter J Huber. Robust statistics, volume 523. John Wiley & Sons, 2004.

[16] Guillaume Lecué and Matthieu Lerasle. Robust machine learning by median-of-means: Theory and practice. Annals of Statistics, 2020. ISSN 21688966. doi: 10.1214/19-AOS1828.

[17] Stanislav Minsker et al. Geometric median and robust estimation in banach spaces. Bernoulli, 21(4):2308–2335, 2015.

[18] John Darzentas, A. S. Nemirovsky, and D. B. Yudin. Problem Complexity and Method Efficiency in Optimization. The Journal of the Operational Research Society, 1984. ISSN 01605682. doi: 10.2307/2581380.

[19] Xuefei Yin, Yanming Zhu, and Jiankun Hu. A comprehensive survey of privacy-preserving federated learning. ACM Computing Surveys (CSUR), 54:1 – 36, 2021.

[20] Cynthia Dwork, Moni Naor, Toniann Pitassi, and Guy N Rothblum. Differential privacy under continual observation. In Proceedings of the forty-second ACM symposium on Theory of computing, pages 715–724, 2010.

[21] T-H Hubert Chan, Elaine Shi, and Dawn Song. Private and continual release of statistics. ACM Transactions on Information and System Security (TISSEC), 14(3): 1–24, 2011.

[22] Abhimanyu Dubey and Alex ’Sandy’ Pentland. Private and byzantine-proof cooperative decision-making. In AAMAS, 2020.

[23] Flint Xiaofeng Fan, Yining Ma, Zhongxiang Dai, Wei Jing, Cheston Tan, and Kian Hsiang Low. Fault-tolerant federated reinforcement learning with theoretical guarantee. ArXiv, abs/2110.14074, 2021.

[24] Thomas P Hayes. A large-deviation inequality for vector-valued martingales. Combinatorics, Probability and Computing, 2005.

[25] O. Shamir. A variant of azuma’s inequality for martingales with subgaussian tails. ArXiv, abs/1110.2392, 2011.

APPENDIX

A. Auxiliary lemmas

The following lemma modifies [14, Lemma 2] and can be used to bound the difference between geometric median and arithmetic mean if choosing $z_0$ as the arithmetic mean of noncorrupted vectors.

**Lemma 6:** Let $\{z_i\}_{i \in \mathcal{N}}$ be a set of vectors or matrices and $\tilde{z} = \text{GM}_{i \in \mathcal{N}}(z_i)$. Given $\mathcal{N}_1 \subseteq \mathcal{N}$ with $\alpha = \frac{|\mathcal{N}_1|}{|\mathcal{N}|} < 1/2$ and any fixed $z_0$, letting $C_\alpha = \frac{2 - 2\alpha}{1 - 2\alpha}$, then

$$\|\tilde{z} - z_0\| \leq C_\alpha \left( \frac{1}{|\mathcal{N}_1| - |\mathcal{N}_0|} \sum_{i \in \mathcal{N}_0} \|z_i - z_0\| + \epsilon \right).$$

**Proof:** The proof is similar to that of [14, Lemma 2]. $\blacksquare$

B. Proofs of Theorem 2

**Proof:** Let $x_{t, i}^* \triangleq \arg\max_{x \in \mathcal{D}_i} \langle x, \theta^* \rangle$ be the optimal action. According to the algorithm, $x_t^i = \arg\max_{x \in \mathcal{D}_i} \langle x, \theta_k \rangle + \beta_k \|x\|_{\Lambda_k}^{-1}$. Therefore,

$$\langle x_t^i, \theta_k \rangle + \beta_k \|x_t^i\|_{\Lambda_k}^{-1} \geq \langle x_{t, i}^*, \theta_k \rangle + \beta_k \|x_{t, i}^*\|_{\Lambda_k}^{-1}.$$
\[ |\langle x_i^t, \theta_k \rangle - \langle x_i^t, \theta^* \rangle| \leq \beta_k \|x_i^t\|_{\Lambda_k^{-1}}, \]
\[ |\langle x_i^{*t}, \theta_k \rangle - \langle x_i^{*t}, \theta^* \rangle| \leq \beta_k \|x_i^{*t}\|_{\Lambda_k^{-1}}. \]

Combining the above three inequalities, we obtain
\[ \langle x_i^{*t}, \theta^* \rangle - \langle x_i^{*t}, \theta^* \rangle \leq 2\beta_k \|x_i^{*t}\|_{\Lambda_k^{-1}} \leq 2\beta_{\max} \|x_i^{*t}\|_{\Lambda_k^{-1}}, \]
where we define \( \beta_{\max} \triangleq \max_{k \in [K]} \beta_k \). Therefore the regret is bounded by
\[ R_T \leq 2\beta_{\max} \sum_{k=1}^{K} \sum_{t \in T_k} \sum_{i \in N_0} \|x_i^t\|_{\Lambda_k^{-1}} \leq 2\beta_{\max} \sqrt{NT} \sum_{k=1}^{K} \sum_{t \in T_k} \sum_{i \in N_0} (x_i^t)^\top \Lambda_k^{-1} x_i^t. \]

Note that
\[ \sum_{i \in N_0} (x_i^t)^\top \Lambda_k^{-1} x_i^t \]
\[ = \frac{N_0}{N_0} \sum_{i \in N_0} (x_i^t)^\top \Lambda_k^{-1} x_i^t + N_0^2 \]
\[ \leq 2 \sum_{i \in N_0} (x_i^t)^\top \Lambda_k^{-1} x_i^t + \xi_t, \]
where
\[ \xi_t \triangleq N_0 \left( \frac{1}{N_0} \sum_{i \in N_0} (x_i^t)^\top \Lambda_k^{-1} x_i^t - \frac{1}{N_0} \sum_{i \in N_0} (x_i^t)^\top \Lambda_k^{-1} x_i^t \right) \]
\[ = N_0 \left( \frac{1}{N_0} \sum_{i \in N_0} \text{Trace} \left( \Lambda_k^{-1} \left[ (x_i^t)^\top (x_i^t)^\top - E[(x_i^t)^\top (x_i^t)^\top)] \right] \right) \]
\[ - \frac{1}{N_0} \sum_{i \in N_0} \text{Trace} \left( \Lambda_k^{-1} \left[ (x_i^t)^\top (x_i^t)^\top - E[(x_i^t)^\top (x_i^t)^\top)] \right] \right) \]

Note that by Assumption 1, we have
\[ |\text{Trace} \left( \Lambda_k^{-1} \left[ (x_i^t)^\top - E[(x_i^t)^\top)] \right] \right)| \leq \sigma d \|\Lambda_k^{-1}\|_2, \]
By our i.i.d. assumption on decision sets, it is straightforward to verify that \( \xi_t \) is a \( \left( 3\sigma^2 d^2 N \|\Lambda_k^{-1}\|_2^2 \right) - \text{subGaussian random variable} \). Then by Azuma’s inequality for martingales with subGaussian tails [25, Theorem 2], with probability at least \( 1 - \delta/4 \),
\[ |\sum_{k=1}^{K} \sum_{t \in T_k} \xi_t| \leq \sigma d \sqrt{6NL \sum_{k=1}^{K} \|\Lambda_k^{-1}\|_2^2} \log \frac{8}{\delta} \]
\[ \leq 6\sigma d \sqrt{NL} \sqrt{\sum_{k=1}^{K} \lambda_k^{-2}} \]
\[ \leq 6\sigma d \sqrt{NL} \sqrt{\sum_{k=1}^{K} \lambda_k^{-2}} \leq d \sqrt{NL} \leq N d \sqrt{\ell}. \]

Then we can obtain that
\[ R_T \leq 4\beta_{\max} \sqrt{NT} \sqrt{\sum_{k=1}^{K} \sum_{t \in T_k} \sum_{i \in N_0} (x_i^t)^\top \Lambda_k^{-1} x_i^t + N d \sqrt{\ell}} \]
\[ \leq 4\beta_{\max} \sqrt{NT} \sqrt{I_1 + I_2 + N d \sqrt{\ell}}, \]
where we define
\[ I_1 = \sum_{k=1}^{K} \sum_{t \in T_k} \sum_{i \in N_0} (x_i^t)^\top (\Lambda_k^{-1} - \Lambda_{k+1}^{-1}) x_i^t \]
\[ = \sum_{k=1}^{K} \sum_{t \in T_k} \sum_{i \in N_0} \|\Lambda_k^{-1} - \Lambda_{k+1}^{-1}\|_2 \]
\[ \leq \sum_{k=1}^{K} \sum_{t \in T_k} \sum_{i \in N_0} \text{Trace} \left( \Lambda_k^{-1} - \Lambda_{k+1}^{-1} \right) \]
\[ = L N_0 \text{Trace} \left( \Lambda_k^{-1} - \Lambda_{k+1}^{-1} \right) \leq \frac{L N_0}{\lambda_0}, \]
\[ I_2 = \sum_{k=1}^{K} \sum_{t \in T_k} \sum_{i \in N_0} (x_i^t)^\top \Lambda_{k+1}^{-1} x_i^t \]
\[ \leq N_0 \sum_{k=1}^{K} \sum_{t \in T_k} \sum_{i \in N_0} (x_i^t)^\top (W_{k+1} + \lambda_0 N_0/2)^{-1} x_i^t \]
\[ \leq 2 N_0 d \text{log} \left( 1 + 2T/\lambda_0 \right) \leq 2 N_0 d \ell, \]
where the second last inequality is due to [3, Lemma 11]. Choosing \( \lambda_0 = L \) and combining all the inequalities above, we get the desired regret bound with probability at least \( 1 - \delta \). \hfill \blacksquare

### C. Proof of Theorem 5

If we use the geometric median of mean oracle, the estimate is \( \theta_k = \Lambda_k^{-1} b_k \) where \( \Lambda_k \) and \( b_k \) are defined as
\[ \Lambda_k = \lambda_k I + GM_{1,\xi_{\leq}} \left( \frac{1}{|T|} \sum_{j \in G_k} (v_j^k) \right) \]
\[ = \lambda_k I + \frac{W_k}{N_0} + E_{\alpha}^k, \]
\[ b_k = GM_{1,\xi_{\leq}} \left( \frac{1}{|T|} \sum_{j \in G_k} (v_j^k) \right) = \frac{W_k}{N_0} + e_{\alpha}^k, \]
where \( E_{\alpha}^k \) and \( e_{\alpha}^k \) are the error terms of using geometric median of mean instead of arithmetic mean:
\[ E_{\alpha}^k \triangleq GM_{1,\xi_{\leq}} \left( \frac{1}{|T|} \sum_{j \in G_k} (v_j^k) \right) - \frac{1}{N_0} \sum_{i \in N_0} v_i^k, \]
\[ e_{\alpha}^k \triangleq GM_{1,\xi_{\leq}} \left( \frac{1}{|T|} \sum_{j \in G_k} (v_j^k) \right) - \frac{1}{N_0} \sum_{i \in N_0} v_i^k. \]
Similar to Lemma 3, we can bound these two error terms in the following lemma:
**Lemma 7**: Using the same parameter choices as in Theorem 5, with probability at least \( 1 - \delta/2 \), for all \( k \in [K] \),
\[ \|E_{\alpha}^k\|_2 \leq 64 \sigma / \sqrt{\alpha (k-1) L d + 4 \epsilon}, \]
\[ \|e_{\alpha}^k\|_2 \leq 64 \sigma (R) / \sqrt{\alpha (k-1) L d + 4 \epsilon}. \]
**Proof**: The proof is similar to that of Lemma 3 and is omitted. \hfill \blacksquare

With Lemma 7, we can bound the difference between \( \theta_k \) and \( \theta^* \) for Byzantine-UCB-MOM.

**Lemma 8 (Approximation error)**: Using the same parameter choices as in Theorem 5, with probability at least \( 1 - \delta/4 \), for all \( x \in \mathbb{R}^d \) and \( k \in [K] \), we have \( |x^\top (\theta_k - \theta^*)| \leq \beta_k \|x\|_{\Lambda_k^{-1}} \).

Lemma 8 and its proof are the same as Lemma 4 except that we choose different parameters \( \lambda_k \) and \( \beta_k \) here. Now we are ready to prove Theorem 5.

**Proof** [Proof of Theorem 5] The proof is the same as that of Theorem 2 except that we use different parameter choices and that we need to bound the following term more carefully:
\[ I_\alpha \triangleq \left| \sum_{k=1}^{K} \sum_{t \in T_k} \xi_t \right|. \]
First, when \( \alpha = 0 \), by the definition of \( \xi_t \), we know \( \xi_t = 0 \) for every \( t \in [T] \) and thus \( I_\alpha = 0 \). When \( \alpha > 0 \), by its definition, we must have \( \alpha \geq 1/N \). Then following the proof of Theorem 2, we can bound that with probability at least \( 1 - \delta/4 \),
\[ I_\alpha \leq 6 \sigma d \sqrt{NL} \sqrt{\sum_{k=1}^{K} \lambda_k^{-2}} \leq d \sqrt{NL} \sqrt{\sum_{k=1}^{K} \lambda_k^{-2}} \leq N d \sqrt{\ell}. \]
This bound is tight enough for the theorem. We can complete the proof following that of Theorem 5. \hfill \blacksquare