A Further Discussions

Here we provide further discussions on our results. We present a detailed comparison with Li and Wang (2022a) in Appendix A.1, where we first discuss the difference in the algorithmic design, and then elaborate on the concentration issue under the asynchronous setting with a simple example. In Appendix A.2, we give an alternative form of our algorithm, where we rewrite Algorithm 1 in an 'episodic' fashion. The purpose is to make it easier for readers to compare our algorithm with existing algorithms for federated linear bandits that are usually expressed in the 'episodic' form.

A.1 Comparison with Li and Wang (2022a)

Difference in algorithmic design. The Async-LinUCB algorithm proposed by Li and Wang (2022a) is not fully asynchronous since in their algorithm, if some agent uploads data to the server, the server will decide if each of the M agents needs to download the data. If the server decides that an agent needs to download the data, this agent has to first download the data from the server and then update its local policy before further interaction with the environment (i.e., taking the next action). In other words, if an agent is offline when the server requests a download, the agent cannot take any further action until it goes online and completes the required download and local model update. In contrast, under the communication protocol in our Algorithm 1, any offline agent can still take action until the trigger of the upload protocol. It is evident that their asynchronous communication protocol is very restricted.

Concentration issue. Next, we discuss the concentration issue, and we first illustrate the problem using a multi-arm bandit instance. Unlike the synchronous case, the reward estimator based on the server-end data can be biased in asynchronous federated linear bandits. To see so, let us consider the following simple example: The decision set contains two arms, A and B, and suppose for pulling arm A, the agent receives a reward equal to either 1 or -1 with equal probability. We assume that there are M agents, and each agent is active for two consecutive rounds. For each agent $m \in [M]$, if the agent has selected the arm A in the first round, then the agent will select again the arm A in the second round only if the agent receives a reward of 1 when pulling arm A in the first round. In this case, it is easy to show that with probability 0.5, an agent selects arm A one time with reward -1, and with probability 0.25, an agent selects arm A twice with total reward 2. Similarly, with probability 0.25, an agent selects arm A twice with a total reward of 0.

In the synchronous setting, all agent will upload their local data to the server at the end of each round. Thus, taking an average for all data at the server, the expected reward of arm A is 0, which equals the actual expected reward of arm A. However, in the asynchronous setting, things become more complicated. Suppose that for each agent, only selecting arm A twice will trigger the upload protocal. Then after two active rounds, an agent will upload its data to the server if and only if the agent receives reward 1 in the first round. Thus among the agents that upload the data, half of them receive a total reward of 2 and the other half receive a total reward of 0. In this case, taking an average for all data at the server, the expected reward of arm A is 0.5, which is a biased estimator compared with the actual expected reward.

Indeed, the above issue could happen in federated linear bandits with the Async-LinUCB algorithm (Li and Wang, 2022a). Specifically, let us consider a linear bandit instance with dimension $d = 2$, and we assume that arm A has context vector $\mathbf{x}_A = (3, 0)^\top$, arm B has context vector $\mathbf{x}_B = (0, 1/\sqrt{10})^\top$, the true model is $\boldsymbol{\theta}^* = \mathbf{0}$, the noise η is a Rademacher random variable, and the parameter λ is set to be 1. Therefore, the rewards for both arm A and B equal to 1 or -1 with 0.5 probability. In this case, based on the principle of optimism in the face of uncertainty, at the beginning, the optimistic estimators for the two arms A, B are 3β and $\beta/\sqrt{10}$ respectively. Thus, all agents will always choose arm A in the first round, so $\mathbf{x}_1 = \mathbf{x}_A$. After choosing arm A at the first round, the optimistic estimator for the two arms A, B in each agent's second round will be $9r/10 + 3\beta/\sqrt{10}$ and $\beta/\sqrt{10}$ respectively, where r is the reward received in the first round. Therefore, with confidence radius $\beta < 1$, each agent will select the arm A (i.e., $\mathbf{x}_2 = \mathbf{x}_A$) in the second round only if the agent receives a reward of $r = 1$ in the first round. Finally, only choosing arm A twice will increase the determinant of the covariance matrix enough to trigger the upload protocol (e.g., $\det(\lambda\mathbf{I} + \mathbf{x}_1\mathbf{x}_1^\top + \mathbf{x}_2\mathbf{x}_2^\top)/\det(\lambda\mathbf{I}) \geq 19$).

As demonstrated above, in the asynchronous setting, the reward estimator based on the server-end data can be biased, which leads to the issue that previous concentration results (e.g., Abbasi-Yadkori

Algorithm 2 Federated linear UCB (Alternative)

1: Initialize $\mathbf{\Sigma}_{m,1} = \mathbf{\Sigma}_1^{\text{ser}} = \lambda \mathbf{I}$, $\widehat{\boldsymbol{\theta}}_{m,1} = 0$, $\mathbf{b}_{m,0}^{\text{loc}} = 0$ and $\mathbf{\Sigma}_{m,0}^{\text{loc}} = 0$ for all $m \in [M]$
2: **for** $k = 1, 2, \ldots, K$ **do**
3: Participation set $P_k \subseteq [M]$ of arbitrary order
4: **for** each active agent $m \in P_k$ **do**
5: Receive $D_{m,k}$ from the environment
6: Select $\mathbf{x}_{m,k} \leftarrow \text{argmax}_{\mathbf{x} \in D_{m,k}} \langle \widehat{\boldsymbol{\theta}}_{m,k}, \mathbf{x}_k \rangle + \beta \|\mathbf{x}\|_{\mathbf{\Sigma}_{m,k}^{-1}}$ /* Optimistic decision */
7: Receive $r_{m,k}$ from environment
8: $\mathbf{\Sigma}_{m,k}^{\text{loc}} \leftarrow \mathbf{\Sigma}_{m,k-1}^{\text{loc}} + \mathbf{x}_{m,k}\mathbf{x}_{m,k}^\top$, $\mathbf{b}_{m,k}^{\text{loc}} \leftarrow \mathbf{b}_{m,k-1}^{\text{loc}} + r_{m,k}\mathbf{x}_{m,k}$ /*Local update*/
9: **if** $\det(\mathbf{\Sigma}_{m,k} + \mathbf{\Sigma}_{m,k}^{\text{loc}}) > (1+\alpha)\det(\mathbf{\Sigma}_{m,k})$ **then**
10: Agent m sends $\mathbf{\Sigma}_{m,k}^{\text{loc}}$ and $\mathbf{b}_{m,k}^{\text{loc}}$ to server /* Upload */
11: $\mathbf{\Sigma}_k^{\text{ser}} \leftarrow \mathbf{\Sigma}_k^{\text{ser}} + \mathbf{\Sigma}_{m,k}^{\text{loc}}$, $\mathbf{b}_k^{\text{ser}} \leftarrow \mathbf{b}_k^{\text{ser}} + \mathbf{b}_{m,k}^{\text{loc}}$ /* Global update */
12: $\mathbf{\Sigma}_{m,k}^{\text{loc}} \leftarrow 0$, $\mathbf{b}_{m,k}^{\text{loc}} \leftarrow 0$
13: Server sends $\mathbf{\Sigma}_k^{\text{ser}}$ and $\mathbf{b}_k^{\text{ser}}$ back to agent m /* Download */
14: $\mathbf{\Sigma}_{m,k+1} \leftarrow \mathbf{\Sigma}_k^{\text{ser}}$, $\mathbf{b}_{m,k} \leftarrow \mathbf{b}_k^{\text{ser}}$
15: $\widehat{\boldsymbol{\theta}}_{m,k+1} \leftarrow \mathbf{\Sigma}_{m,k+1}^{-1}\mathbf{b}_{m,k+1}$ /* Compute estimate */
16: **else**
17: $\mathbf{\Sigma}_{m,k+1} \leftarrow \mathbf{\Sigma}_{m,k}$, $\mathbf{b}_{m,k+1} \leftarrow \mathbf{b}_{m,k}$, $\widehat{\boldsymbol{\theta}}_{m,k+1} \leftarrow \widehat{\boldsymbol{\theta}}_{m,k}$
18: **end if**
19: **end for**
20: **for** other inactive agents $m \in [M] \setminus P_k$ **do**
21: $\mathbf{\Sigma}_{m,k+1} \leftarrow \mathbf{\Sigma}_{m,k}$, $\mathbf{b}_{m,k+1} \leftarrow \mathbf{b}_{m,k}$, $\widehat{\boldsymbol{\theta}}_{m,k+1} \leftarrow \widehat{\boldsymbol{\theta}}_{m,k}$
22: **end for**
23: **end for**

et al. (2011)) cannot be directly used for the server's data. This is why we need a more dedicated analysis to control this biased error (see Lemma 6.6 for more details).

A.2 An Alternative Form of Algorithm 1

We introduce an alternative form of Algorithm 1, which is displayed in Algorithm 2. Algorithm 2 can be viewed as the episodic[7] version of Algorithm 1, and its form aligns with those of the existing algorithms for federated linear bandits (Wang et al., 2019; Dubey and Pentland, 2020; Huang et al., 2021; Li and Wang, 2022a).

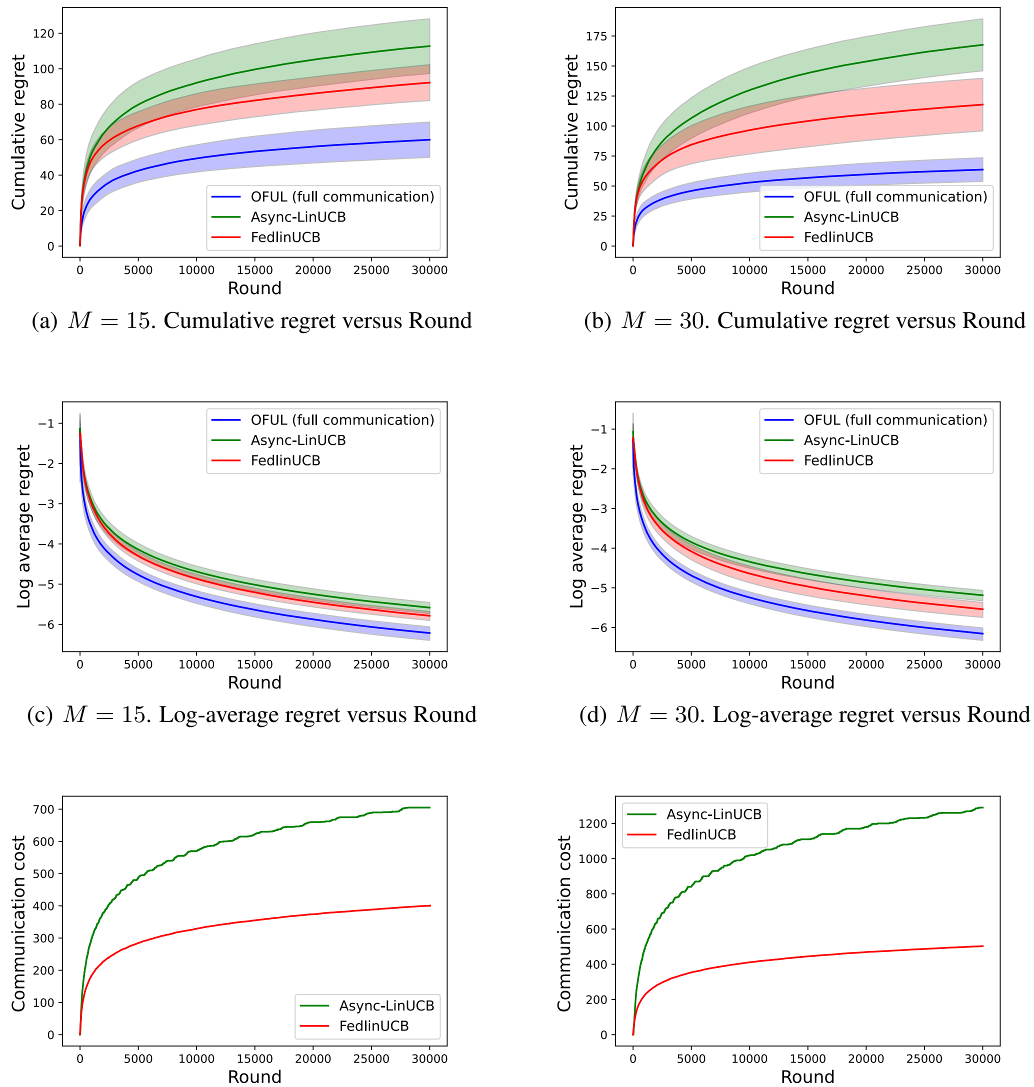
Specifically, in Algorithm 2, for each round (episode) $k \in [K]$, the set of active agents is given by P_k, where the order of agents in P_k can be arbitrary (Line 3). Then the agents in the set P_k participate according to the prefixed order (Line 4). The operations in the inner loop of Algorithm 2 (i.e., decision rule, upload/download, local/global update, and model estimates) are all identical to those in Algorithm 1. Therefore, Algorithm 2 is indeed equivalent to Algorithm 1 up to relabeling of the participation of the agents, and hence all the theoretical results for Algorithm 1 also hold for Algorithm 2.

B Experiments

In this section, we provide the remaining details on numerical simulations.

Experiment setup. We construct two linear bandit instances with dimension $d = 25$. In the first instance, the true model parameter $\boldsymbol{\theta}^*$ is $[1/\sqrt{d}, .., 1/\sqrt{d}] \in \mathbb{R}^d$. In the second instance, the true model parameter $\boldsymbol{\theta}^*$ is generated by uniform random sampling over the space $[-1/\sqrt{d}, 1/\sqrt{d}]^d$ with normalization. For each round $t \in [T]$, the active agent m_t is uniform sampled from all M agents

[7]Here 'episode' means a collection of every agent's interaction with the environment for one round, which is different from the usual term in online learning that refers to a sequential interaction lasting for a certain time horizon. We only use this term to differentiate Algorithm 2 from Algorithm 1.



(a) $M = 15$. Cumulative regret versus Round

(b) $M = 30$. Cumulative regret versus Round

(c) $M = 15$. Log-average regret versus Round

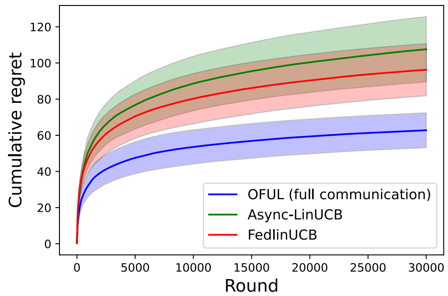
(d) $M = 30$. Log-average regret versus Round

(e) $M = 15$. Communication cost versus Round

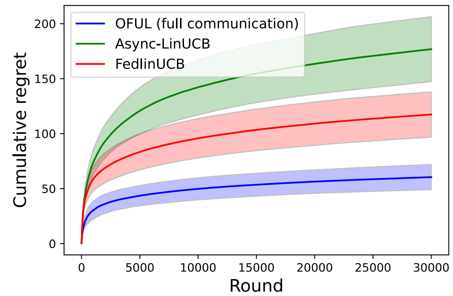
(f) $M = 30$. Communication cost versus Round

Figure 3: Comparison of FedlinUCB (ours), Async-LinUCB (Li and Wang, 2022a) and OFUL (full communication) (Abbasi-Yadkori et al., 2011) with parameter $\boldsymbol{\theta}^* = [1/\sqrt{d}, .., 1/\sqrt{d}] \in \mathbb{R}^d$. Experiments are run for $M = 15$ and $M = 30$, and results are averaged over 20 runs. Figures 3(a) and 3(b) present the cumulative regret; Figures 3(c) and 3(d) show the averaged regret (in log scale); Figures 3(e) and 3(f) compare the communication cost versus number of rounds. Note that the communication cost of OFUL with full communication is linear with the number of rounds T and is far greater than those of both FedLinUCB and Async-LinUCB. Therefore, in order to make a clearer comparison between the communication cost of FedLinUCB and Async-LinUCB, OFUL is omitted in Figure 3(e) and 3(f).

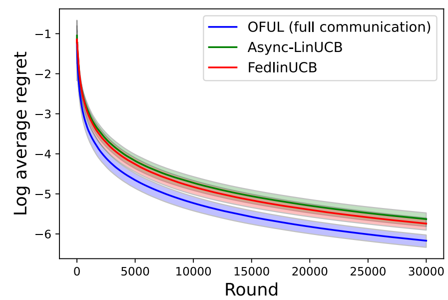
and the decision set $\mathcal{D}_t$ consists of 25 different actions uniformly randomly sampled from the space $[-1/\sqrt{d}, 1/\sqrt{d}]^d$. After the active agent m_t chooses an action $\mathbf{x}_t$, the agent m_t receives a reward given by $r_t = \langle \mathbf{x}_t, \boldsymbol{\theta}^* \rangle + \eta_t$, where η_t is a 0.3-Gaussian noise. We run simulation on the above linear bandit instance with the total number of rounds $T = 30000$ (repeating 20 times and taking the average) and the number of agents is set to be 15 or 30. We implement our `FedLinUCB` algorithm and compare its performance with Async-LinUCB (Li and Wang, 2022a) and OFUL Abbasi-Yadkori et al. (2011) with full communication (i.e., the active agent communicates with the server in each



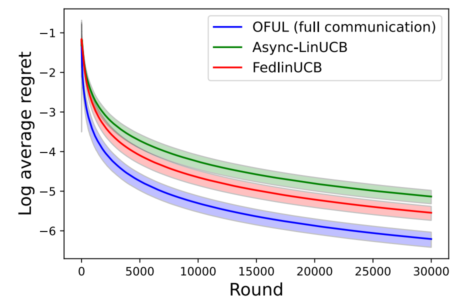
(a) $M = 15$. Cumulative regret versus Round



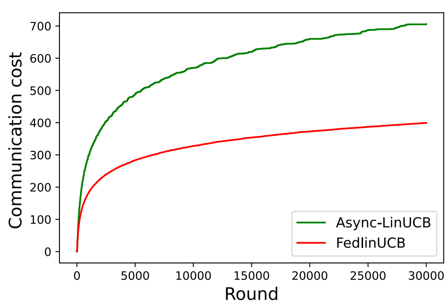
(b) $M = 30$. Cumulative regret versus Round



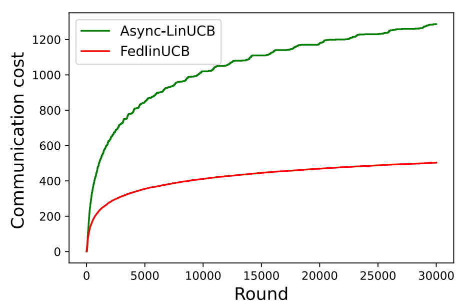
(c) $M = 15$. Log-average regret versus Round



(d) $M = 30$. Log-average regret versus Round



(e) $M = 15$. Communication cost versus Round



(f) $M = 30$. Communication cost versus Round

Figure 4: Comparison of FedlinUCB (ours), Async-LinUCB (Li and Wang, 2022a) and OFUL (full communication) (Abbasi-Yadkori et al., 2011) with random select $\boldsymbol{\theta}^*$. Experiments are run for $M = 15$ and $M = 30$, and results are averaged over 20 runs. Figures 4(a) and 4(b) present the cumulative regret; Figures 4(c) and 4(d) show the averaged regret (in log scale); Figures 4(e) and 4(f) compare the communication cost versus number of rounds. Note that the communication cost of OFUL with full communication is linear with the number of rounds T and is far greater than those of both FedLinUCB and Async-LinUCB. Therefore, in order to make a clearer comparison between the communication cost of FedLinUCB and Async-LinUCB, OFUL is omitted in Figure 4(e) and 4(f).

round). We set the parameter $\alpha = 1$ for `FedLinUCB` and $\gamma_U = \gamma_D = 5$ for Async-LinUCB to ensure that the communication costs of `FedLinUCB` and Async-LinUCB have similar magnitudes.

Results. The results are presented in Figure 3 and 4, suggesting that our algorithm significantly outperforms Async-LinUCB as our algorithm achieves smaller regret while spending less communication cost.

Specifically, Figure 3(a) and 3(b) displays the cumulative regret of our `FedLinUCB` algorithm, Async-LinUCB (Li and Wang, 2022a) and OFUL (Abbasi-Yadkori et al., 2011) with full communication.

It can be apparently seen that `FedLinUCB` outperforms Async-LinUCB in terms of regret. Next, Figure 3(c) and 3(d) show the average regret in log scale. These two plots show that the average regret of our algorithm has a rate very close to the optimal rate of OFUL. Finally, we plot communication cost versus number of rounds in Figure 3(e) and 3(f), which indicates that the communication cost of `FedLinUCB` is significantly lower than that of Async-LinUCB. Together with Figure 3(a) and 3(b), we see that our algorithm achieves lower regret with lower communication cost, compared to Async-LinUCB.

The experiment for Figure 4 adopts a different $\boldsymbol{\theta}^*$ (which is randomly selected sampled the space $[-1/\sqrt{d}, 1/\sqrt{d}]^d$ with normalization) from that of Figure 3. Figure 4 shows almost the same results as Figure 3, indicating our algorithms works in general.

Overall, the simulation corroborates our theoretical results. It also shows that our algorithm indeed outperforms Async-LinUCB (Li and Wang, 2022a).

For all the experiments, the results are averaged over 20 runs with the error bars chosen as the empirical one standard deviation. All experiments are conducted on a Macbook with 8-core CPU and 16 GB of memory.

C Missing Proofs in Section 6

Here we present the proof of the results in Section 6.

C.1 Communication complexity within each epoch

We first present the proof for the bound on the communication complexity within each epoch given in Lemma 6.2.

Proof of Lemma 6.2. For each agent $m \in [M]$, let n_m be the number of communications agent m has made during this epoch, and we denote the communication rounds as $t_1, \dots, t_{n_m}$ for simplicity. Now we consider the data uploaded to the server, and it can be denoted by the value of covariance matrix $\boldsymbol{\Sigma}^{\text{loc}}_{m,t_j}$ before communicating with the server. For each $j = 2, \dots, n_m$, according to the determinant-based criterion (Line 9) in Algorithm 1, we have

$$\det(\boldsymbol{\Sigma}_{m,t_j} + \boldsymbol{\Sigma}^{\text{loc}}_{m,t_j}) - \det(\boldsymbol{\Sigma}_{m,t_j}) > \alpha \cdot \det(\boldsymbol{\Sigma}_{m,t_j}),$$

which further implies that

$$\alpha \cdot \det(\boldsymbol{\Sigma}^{\text{ser}}_{T_i}) < \det(\boldsymbol{\Sigma}^{\text{ser}}_{T_i} + \boldsymbol{\Sigma}^{\text{loc}}_{m,t_j}) - \det(\boldsymbol{\Sigma}^{\text{ser}}_{T_i}), \tag{C.1}$$

where the inequality holds due to Lemma E.2 together with the fact that the communication in round t_1 updates the covariance matrix so that $\boldsymbol{\Sigma}_{m,t_j} \succeq \boldsymbol{\Sigma}^{\text{ser}}_{T_i}$. In addition, we define the sequence of all communications from T_i to $T_{i+1} - 1$ as $t'_1, \dots, t'_L$. For each round t'_j, if the agent $m_{t'_j}$ have already communicated with the server earlier in this epoch, we have

$$\begin{aligned}
\det(\boldsymbol{\Sigma}^{\text{ser}}_{t'_j}) - \det(\boldsymbol{\Sigma}^{\text{ser}}_{t'_{j-1}}) &= \det(\boldsymbol{\Sigma}^{\text{ser}}_{t'_{j-1}} + \boldsymbol{\Sigma}^{\text{loc}}_{m_{t'_j},t'_j}) - \det(\boldsymbol{\Sigma}^{\text{ser}}_{t'_{j-1}}) \\
&\geq \det(\boldsymbol{\Sigma}^{\text{ser}}_{T_i} + \boldsymbol{\Sigma}^{\text{loc}}_{m,t_j}) - \det(\boldsymbol{\Sigma}^{\text{ser}}_{T_i}) \\
&> \alpha \cdot \det(\boldsymbol{\Sigma}^{\text{ser}}_{T_i}),
\end{aligned} \tag{C.2}$$

where the first inequality holds due to Lemma E.1 together with the fact that $\boldsymbol{\Sigma}^{\text{ser}}_{t'_{j-1}} \succeq \boldsymbol{\Sigma}^{\text{ser}}_{T_i}$, and the second inequality follows from (C.1). Now, taking the sum of (C.2) over all round t'_j, we obtain

$$\det(\boldsymbol{\Sigma}^{\text{ser}}_{T_{i+1}-1}) - \det(\boldsymbol{\Sigma}^{\text{ser}}_{T_i}) = \sum_{1 \leq j \leq L} \det(\boldsymbol{\Sigma}^{\text{ser}}_{t'_j}) - \det(\boldsymbol{\Sigma}^{\text{ser}}_{t'_{j-1}}) \geq \sum_{m=1}^{M} (n_m - 1)\alpha \cdot \det(\boldsymbol{\Sigma}^{\text{ser}}_{T_i}).$$

Since $\det(\boldsymbol{\Sigma}_{\text{ser},T_{i+1}-1}) \leq 2\det(\boldsymbol{\Sigma}_{\text{ser},T_i})$, we further have

$$\sum_{j \in M} n_j \leq M + 1/\alpha.$$

Each communication between one agent and the server includes one upload and one download, so the communication complexity within one epoch is bounded by $2(M + 1/\alpha)$. This finishes the proof. $\square$

C.2 Proof for the covariance comparison

Next, we prove the comparison between the covariance matrices given in Lemma 6.5.

Proof of Lemma 6.5. Fix any round $t \in [T]$. Let $t_1 \le t$ be the last round such that agent m is active at round t_1. If agent m communicated with the server at this round, then we have

$$\lambda \mathbf{I} + \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t}^{\text{up}} \succeq \mathbf{0} = \frac{1}{\alpha} \boldsymbol{\Sigma}_{m,t}^{\text{loc}}.$$

Otherwise, according to determinant-based criterion (Line 9) in Algorithm 1, at the end of each round t_1, we have

$$\det\left(\boldsymbol{\Sigma}_{m,t_1} + \boldsymbol{\Sigma}_{m,t_1}^{\text{loc}}\right) \le (1+\alpha) \det(\boldsymbol{\Sigma}_{m,t_1}).$$

By Lemma E.4, for any non-zero vector $\mathbf{x} \in \mathbb{R}^d$, we have

$$\frac{\mathbf{x}^\top (\boldsymbol{\Sigma}_{m,t_1} + \boldsymbol{\Sigma}_{m,t_1}^{\text{loc}}) \mathbf{x}}{\mathbf{x}^\top \boldsymbol{\Sigma}_{m,t_1} \mathbf{x}} \le \frac{\det(\boldsymbol{\Sigma}_{m,t_1} + \boldsymbol{\Sigma}_{m,t_1}^{\text{loc}})}{\det(\boldsymbol{\Sigma}_{m,t_1})} \le 1 + \alpha.$$

Rearranging the above yields $\mathbf{x}^\top \boldsymbol{\Sigma}_{m,t_1}^{\text{loc}} \mathbf{x} \le \alpha \mathbf{x}^\top \boldsymbol{\Sigma}_{m,t_1} \mathbf{x}$, which then implies that

$$\boldsymbol{\Sigma}_{m,t_1} \succeq \frac{1}{\alpha} \boldsymbol{\Sigma}_{m,t_1}^{\text{loc}}$$

Note that $\boldsymbol{\Sigma}_{m,t_1}$ is the downloaded covariance matrix from last communication before round t_1, so it must satisfy $\boldsymbol{\Sigma}_{m,t_1} \preceq \boldsymbol{\Sigma}_{t_1}^{\text{ser}}$. Therefore, we have

$$\lambda \mathbf{I} + \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t_1}^{\text{up}} = \boldsymbol{\Sigma}_{t_1}^{\text{ser}} \succeq \boldsymbol{\Sigma}_{m,t_1} \succeq \frac{1}{\alpha} \boldsymbol{\Sigma}_{m,t_1}^{\text{loc}}.$$

Now, for round t, since agent m is inactive from round t_1 to t, then we have

$$\lambda \mathbf{I} + \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t}^{\text{up}} \succeq \lambda \mathbf{I} + \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t_1}^{\text{up}} \succeq \frac{1}{\alpha} \boldsymbol{\Sigma}_{m,t_1}^{\text{loc}} = \frac{1}{\alpha} \boldsymbol{\Sigma}_{m,t}^{\text{loc}},$$

which yields the first claim in Lemma 6.5.

Next, suppose agent m is the only active agent from round t_1 to $t_2 - 1$ and agent m only communicates with the server at round t_1. Further average the above inequality over all agents $m \in [M]$, and we get

$$\lambda \mathbf{I} + \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t}^{\text{up}} \succeq \frac{1}{M\alpha} \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t}^{\text{loc}}, \tag{C.3}$$

which implies that for $t_1 + 1 \le t \le t_2 - 1$, we have

$$\begin{aligned} \boldsymbol{\Sigma}_{m,t} &= \lambda \mathbf{I} + \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t_1}^{\text{up}} = \lambda \mathbf{I} + \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t}^{\text{up}} \\ &\succeq \frac{1}{1+M\alpha} \left(\lambda \mathbf{I} + \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t}^{\text{up}} + \sum_{m'=1}^{M} \boldsymbol{\Sigma}_{m',t}^{\text{loc}} \right) = \frac{1}{1+M\alpha} \boldsymbol{\Sigma}_t^{\text{all}}, \end{aligned}$$

where the second equation holds due to the fact that no agent communicate with server from round $t_1 + 1$ to $t_2 - 1$, and the inequality follows from (C.3). This yields the second claim in Lemma 6.5 and finishes the proof. $\square$

C.3 Proof of the local concentration for agents

Recall that the global concentration and corresponding global confidence bound have been shown in Lemma 6.3. Next, we establish the concentration properties of the local data on the agents' side.

Proof of Lemma 6.4. For each agent $m \in [M]$ and any rounds $1 \le t_1 \le t_2 \le T$, consider

$$\boldsymbol{\Sigma}_{m,t_1,t_2} = \alpha\lambda\mathbf{I} + \sum_{i=t_1+1, m_i=m}^{t_2} \mathbf{x}_i\mathbf{x}_i^\top \quad \text{and} \quad \mathbf{u}_{m,t_1,t_2} = \sum_{i=t_1+1, m_i=m}^{t_2} \mathbf{x}_i\eta_i.$$

By Theorem 2 in Abbasi-Yadkori et al. (2011), with probability at least $1-\delta/(T^2M)$, we have

$$\|\boldsymbol{\Sigma}_{m,t_1,t_2}^{-1}\mathbf{u}_{m,t_1,t_2}\|_{\boldsymbol{\Sigma}_{m,t_1,t_2}} \le R\sqrt{d\log\Big(\big(1+TL^2/(\alpha\lambda)\big)/\delta\Big)} + \sqrt{\lambda}S.$$

Then taking an union bound over all agent $m \in [M]$ and rounds $1 \le t_1 \le t_2 \le T$ and applying to $t_1 = N_m(t)$ and $t_2 = t$ for each $t \in [T]$, we obtain the desired concentration. This finishes the proof. $\square$

For clarity, we break Lemma 6.6 into two lemmas, Lemma C.1 for local confidence bound and Lemma C.2 for per-round regret, and prove them separately.

Lemma C.1 (Local confidence bound)**.** Under the setting of Theorem 5.1, with probability at least $1-\delta$, for each $t \in [T]$, the estimate $\widehat{\boldsymbol{\theta}}_{m,t+1}$ satisfies that $\|\boldsymbol{\theta}^* - \widehat{\boldsymbol{\theta}}_{m,t+1}\|_{\boldsymbol{\Sigma}_{m,t+1}} \le \beta$.

Proof of Lemma C.1. Since the estimated vector $\widehat{\boldsymbol{\theta}}_{m,t+1}$ and covariance matrix $\boldsymbol{\Sigma}_{m,t+1}$ will keep the same value as in the previous round if the agent m do not communicate with the server, we only need to consider for those round t where agent m communicates with the server. By the determinant-based criterion (Line 9) in Algorithm 1, if the agent m communicates with the server in round t, then at the end of this round, the covariance matrix $\boldsymbol{\Sigma}_{m,t+1}$ and vector $\mathbf{b}_{m,t+1}$ are given by

$$\boldsymbol{\Sigma}_{m,t+1} = \lambda\mathbf{I} + \sum_{m'=1}^{M}\boldsymbol{\Sigma}_{m',N_{m'}(t)}^{\text{up}} = \lambda\mathbf{I} + \sum_{m'=1}^{M}\boldsymbol{\Sigma}_{m',t}^{\text{up}}, \qquad \mathbf{b}_{m,t+1} = \sum_{m'=1}^{M}\mathbf{b}_{m',t}^{\text{up}}. \tag{C.4}$$

Therefore, the estimated vector $\widehat{\boldsymbol{\theta}}_{m,t+1}$ is

$$\begin{aligned}
\widehat{\boldsymbol{\theta}}_{m,t+1} &= \bigg(\lambda\mathbf{I} + \sum_{m'=1}^{M}\boldsymbol{\Sigma}_{m',t}^{\text{up}}\bigg)^{-1}\sum_{m'=1}^{M}\mathbf{b}_{m',t}^{\text{up}} \\
&= \bigg(\lambda\mathbf{I} + \sum_{m'=1}^{M}\boldsymbol{\Sigma}_{m',t}^{\text{up}}\bigg)^{-1}\sum_{m'=1}^{M}\big(\boldsymbol{\Sigma}_{m',t}^{\text{up}}\boldsymbol{\theta}^* + \mathbf{u}_{m',t}^{\text{up}}\big) \\
&= \boldsymbol{\theta}^* - \lambda\bigg(\lambda\mathbf{I} + \sum_{m'=1}^{M}\boldsymbol{\Sigma}_{m',t}^{\text{up}}\bigg)^{-1}\boldsymbol{\theta}^* + \bigg(\lambda\mathbf{I} + \sum_{m'=1}^{M}\boldsymbol{\Sigma}_{m',k}^{\text{up}}\bigg)^{-1}\sum_{m'=1}^{M}\mathbf{u}_{m',t}^{\text{up}} \\
&= \boldsymbol{\theta}^* - \lambda(\boldsymbol{\Sigma}_{m,t+1})^{-1}\boldsymbol{\theta}^* + \sum_{m'=1}^{M}(\boldsymbol{\Sigma}_{m,t+1})^{-1}\mathbf{u}_{m',t}^{\text{up}}.
\end{aligned}$$

Thus, the difference between $\widehat{\boldsymbol{\theta}}_{m,t+1}$ and the underlying truth $\boldsymbol{\theta}^*$ can be decomposed as

$$\begin{aligned}
\big\|\boldsymbol{\theta}^* - \widehat{\boldsymbol{\theta}}_{m,t+1}\big\|_{\boldsymbol{\Sigma}_{m,t+1}} &\le \big\|\lambda(\boldsymbol{\Sigma}_{m,t+1})^{-1}\boldsymbol{\theta}^*\big\|_{\boldsymbol{\Sigma}_{m,t+1}} + \bigg\|\sum_{m'=1}^{M}(\boldsymbol{\Sigma}_{m,t+1})^{-1}\mathbf{u}_{m',t}^{\text{up}}\bigg\|_{\boldsymbol{\Sigma}_{m,t+1}} \\
&\le \sqrt{\lambda}\|\boldsymbol{\theta}^*\|_2 + \bigg\|\sum_{m'=1}^{M}(\boldsymbol{\Sigma}_{m,t+1})^{-1}\mathbf{u}_{m',t}^{\text{up}}\bigg\|_{\boldsymbol{\Sigma}_{m,t+1}},
\end{aligned} \tag{C.5}$$

where the first inequality holds due to that fact that $\|\mathbf{a}+\mathbf{b}\|_{\boldsymbol{\Sigma}} \le \|\mathbf{a}\|_{\boldsymbol{\Sigma}} + \|\mathbf{b}\|_{\boldsymbol{\Sigma}}$ and the second inequality follows from $\boldsymbol{\Sigma}_{m,t+1} \ge \lambda\mathbf{I}$. By the assumption that $\|\boldsymbol{\theta}^*\|_2 \le S$, the first term can be controlled by $\sqrt{\lambda}S$. For the second term in (C.5), consider the following decomposition:

$$\begin{aligned}
\sum_{m'=1}^{M}(\boldsymbol{\Sigma}_{m,t+1})^{-1}\mathbf{u}_{m',t}^{\text{up}} &= \sum_{m'=1}^{M}(\boldsymbol{\Sigma}_{m,t+1})^{-1}\big(\mathbf{u}_{m',t}^{\text{up}} + \mathbf{u}_{m',t}^{\text{loc}}\big) - \sum_{m'=1}^{M}(\boldsymbol{\Sigma}_{m,t+1})^{-1}\mathbf{u}_{m',t}^{\text{loc}} \\
&= \underbrace{(\boldsymbol{\Sigma}_{m,t+1})^{-1}\mathbf{u}_{t}^{\text{all}}}_{\mathcal{A}} - \sum_{m'=1}^{M}\underbrace{(\boldsymbol{\Sigma}_{m,t+1})^{-1}\mathbf{u}_{m',t}^{\text{loc}}}_{B_{m'}}.
\end{aligned} \tag{C.6}$$

For the term $\mathcal{A}$, it follows from (6.5) in Lemma 6.5 that

$$\begin{aligned}\big\|(\mathbf{\Sigma}_{m,t+1})^{-1}\mathbf{u}_t^{\text{all}}\big\|_{\mathbf{\Sigma}_{m,t+1}} &= \big\|(\mathbf{\Sigma}_{m,t+1})^{-1/2}\mathbf{u}_t^{\text{all}}\big\|_2 \\ &\le \sqrt{1+M\alpha}\cdot\big\|(\mathbf{\Sigma}_t^{\text{all}})^{-1/2}\mathbf{u}_t^{\text{all}}\big\|_2 \\ &\le \sqrt{1+M\alpha}\cdot\Big(R\sqrt{d\log\big((1+TL^2/\lambda)/\delta\big)}+\sqrt{\lambda}S\Big),\end{aligned} \tag{C.7}$$

where the second inequality holds due to Lemma 6.3. Next, for each term $B_{m'}$ in (C.6), by (6.4) in Lemma 6.5, we have

$$\lambda\mathbf{I}+\sum_{j=1}^{M}\mathbf{\Sigma}_{j,t}^{\text{up}}\succeq\frac{1}{\alpha}\mathbf{\Sigma}_{m',t}^{\text{loc}},$$

which further implies that

$$\lambda\mathbf{I}+\sum_{j=1}^{M}\mathbf{\Sigma}_{j,t}^{\text{up}}\succeq\frac{1}{2\alpha}(\alpha\lambda\mathbf{I}+\mathbf{\Sigma}_{m',t}^{\text{loc}}). \tag{C.8}$$

Thus, the norm of each term $B_{m'}$ can be bounded as

$$\begin{aligned}\big\|(\mathbf{\Sigma}_{m,t+1})^{-1}\mathbf{u}_{m',t}^{\text{loc}}\big\|_{\mathbf{\Sigma}_{m,t+1}} &= \big\|(\mathbf{\Sigma}_{m,t+1})^{-1/2}\mathbf{u}_{m',t}^{\text{loc}}\big\|_2 \\ &\le \sqrt{2\alpha}\cdot\Big\|\big(\alpha\lambda\mathbf{I}+\mathbf{\Sigma}_{m',t}^{\text{loc}}\big)^{-1/2}\mathbf{u}_{m',t}^{\text{loc}}\Big\|_2 \\ &\le \sqrt{2\alpha}\cdot\bigg(R\sqrt{d\log\frac{\alpha\lambda+TL^2}{\alpha\lambda\delta}}+\sqrt{\lambda}S\bigg),\end{aligned} \tag{C.9}$$

where the first inequality holds due to (C.8) and the second inequality follows from Lemma 6.4.

Finally, combining (C.5), (C.6), (C.7) and (C.9), we obtain

$$\big\|\boldsymbol{\theta}^*-\widehat{\boldsymbol{\theta}}_{m,t+1}\big\|_{\mathbf{\Sigma}_{m,t+1}}\le\sqrt{\lambda}S+\big(\sqrt{1+M\alpha}+M\sqrt{2\alpha}\big)\bigg(R\sqrt{d\log\frac{\min(\alpha,1)\cdot\lambda+TL^2}{\min(\alpha,1)\cdot\lambda\delta}}+\sqrt{\lambda}S\bigg).$$

Thus we finish the proof of Lemma C.1. □

Lemma C.2 (Per-round regret)**.** Under the setting of Theorem 5.1, with probability at least $1-\delta$, for each $t\in[T]$, the regret in round t satisfies

$$\Delta_t=\max_{\mathbf{x}\in\mathcal{D}_t}\langle\boldsymbol{\theta}^*,\mathbf{x}\rangle-\langle\boldsymbol{\theta}^*,\mathbf{x}_t\rangle\le 2\beta\sqrt{\mathbf{x}_t^\top\mathbf{\Sigma}_{m_t,t}^{-1}\mathbf{x}_t}.$$

Proof of Lemma C.2. First, by Lemma C.1, with probability at least $1-\delta$, for each round $t\in[T]$ and each action $\mathbf{x}\in\mathcal{D}_t$, we have

$$\begin{aligned}\widehat{\boldsymbol{\theta}}_{m,t}^\top\mathbf{x}+\beta\sqrt{\mathbf{x}^\top\mathbf{\Sigma}_{m_t,t}^{-1}\mathbf{x}}-(\boldsymbol{\theta}^*)^\top\mathbf{x} &= (\widehat{\boldsymbol{\theta}}_{m,t}-\boldsymbol{\theta}^*)^\top\mathbf{x}+\beta\sqrt{\mathbf{x}^\top\mathbf{\Sigma}_{m_t,t}^{-1}\mathbf{x}} \\ &\ge -\|\widehat{\boldsymbol{\theta}}_{m,t}-\boldsymbol{\theta}^*\|_{\mathbf{\Sigma}_{m_t,t}}\cdot\|\mathbf{x}\|_{\mathbf{\Sigma}_{m_t,t}^{-1}}+\beta\sqrt{\mathbf{x}^\top\mathbf{\Sigma}_{m_t,t}^{-1}\mathbf{x}} \\ &\ge -\beta\|\mathbf{x}\|_{\mathbf{\Sigma}_{m_t,t}^{-1}}+\beta\sqrt{\mathbf{x}^\top\mathbf{\Sigma}_{m_t,t}^{-1}\mathbf{x}} \\ &= 0,\end{aligned} \tag{C.10}$$

where the first inequality holds due to the Cauchy-Schwarz inequality and the last inequality follows from Lemma C.1. (C.10) shows that the estimator for agent m_t is always optimistic. For simplicity, we denote the optimal action at round t as $\mathbf{x}^* = \arg\max_{\mathbf{x}\in\mathcal{D}_t}(\boldsymbol{\theta}^*)^\top\mathbf{x}$, and (C.10) further implies

$$\begin{aligned}\Delta_t &= (\boldsymbol{\theta}^*)^\top\mathbf{x}^* - (\boldsymbol{\theta}^*)^\top\mathbf{x}_t\\ &\le \widehat{\boldsymbol{\theta}}_{m,t}^\top\mathbf{x}^* + \beta\sqrt{(\mathbf{x}^*)^\top\boldsymbol{\Sigma}_{m_t,t}^{-1}\mathbf{x}^*} - (\boldsymbol{\theta}^*)^\top\mathbf{x}_t\\ &\le \widehat{\boldsymbol{\theta}}_{m,t}^\top\mathbf{x}_t + \beta\sqrt{\mathbf{x}_t^\top\boldsymbol{\Sigma}_{m_t,t}^{-1}\mathbf{x}_t} - (\boldsymbol{\theta}^*)^\top\mathbf{x}_t\\ &= (\widehat{\boldsymbol{\theta}}_{m,t} - \boldsymbol{\theta}^*)^\top\mathbf{x}_t + \beta\sqrt{\mathbf{x}_t^\top\boldsymbol{\Sigma}_{m_t,t}^{-1}\mathbf{x}_t}\\ &\le \|\widehat{\boldsymbol{\theta}}_{m,t} - \boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_{m_t,t}}\cdot\|\mathbf{x}_k\|_{\boldsymbol{\Sigma}_{m_t,t}^{-1}} + \beta\sqrt{\mathbf{x}_t^\top\boldsymbol{\Sigma}_{m_t,t}^{-1}\mathbf{x}_t}\\ &\le 2\beta\sqrt{\mathbf{x}_t^\top\boldsymbol{\Sigma}_{m_t,t}^{-1}\mathbf{x}_t},\end{aligned}$$

where the first inequality holds due to (C.10), the second inequality follows from the definition of action $\mathbf{x}_t$ in Algorithm 1, the third inequality applies the Cauchy-Schwarz inequality, and the last inequality is by Lemma C.1. Thus, we finish the proof of Lemma C.2. □

Combining Lemmas C.1 and C.2 yields Lemma 6.6

D Proof for Lower Bound

Lemma D.1 (Theorem 3 in Abbasi-Yadkori et al. 2011)**.** There exists a constant $C > 0$, such that for any normalized linear bandit instance with $R = L = S = 1$, the expectation of the regret for OFUL algorithm is upper bounded by $\mathbb{E}[\text{Regret}(T)] \le Cd\sqrt{T}\log T$.

Lemma D.2 (Theorem 24.1 in Lattimore and Szepesvári 2020)**.** There exists a set of hard-to-learn normalized linear bandit instances with $R = L = S = 1$, such that for any algorithm **Alg** and $T \ge d$, for a uniformly random instance in the set, the regret is lower bounded by $\mathbb{E}[\text{Regret}(T)] \ge cd\sqrt{T}$ for some constant $c > 0$.

Theorem 5.3 is an extension of the lower bound result in Wang et al. (2019, Theorem 2) from multi-arm bandits to linear bandits.

Proof of Theorem 5.3. For any algorithm **Alg** for federated bandits, we construct the auxiliary **Alg1** as follows: For each agent $m \in [M]$, it performs **Alg** until there is a communication between the agent m and the server (upload or download data). After the communication, the agent m remove all previous information and perform the `OFUL` Algorithm in Abbasi-Yadkori et al. (2011). In this case, for each agent $m \in [M]$, **Alg1** do not utilize any information from other agents and it will reduce to a single agent bandit algorithm.

Now, we uniformly randomly select a hard-to-learn instance from the set given in Lemma D.2, and let each agent $m \in [M]$ be active for T/M different rounds (where we assume T/M is an integer for simplicity). Since **Alg1** reduces to a single agent bandit algorithm, Lemma D.2 implies that the expected regret for agent m with **Alg1** is lower bounded by

$$\mathbb{E}[\text{Regret}_{m,\textbf{Alg1}}] \ge cd\sqrt{T/M}. \tag{D.1}$$

Taking the sum of (D.1) over all agents $m \in [M]$, we obtain

$$\mathbb{E}[\text{Regret}_{\textbf{Alg1}}] = \sum_{m=1}^{M}\mathbb{E}[\text{Regret}_{m,\textbf{Alg1}}] \ge cd\sqrt{MT}. \tag{D.2}$$

For each agent $m \in [M]$, let δ_m denote the probability that agent m will communicate with the server. Notice that before the communication, **Alg1** has the same performance as **Alg**, while for the rounds after the communication, **Alg1** executes the `OFUL` algorithm and Lemma D.1 suggests an $O(d\sqrt{T/M}\log(T/M))$ upper bounded for the expected regret. Therefore, the expected regret for agent m with **Alg1** is upper bounded by

$$\mathbb{E}[\text{Regret}_{m,\textbf{Alg1}}] \le \mathbb{E}[\text{Regret}_{m,\textbf{Alg}}] + \delta_m Cd\sqrt{T/M}\log(T/M). \tag{D.3}$$

Taking the sum of (D.3) over all agents $m \in [M]$, we obtain

$$\begin{aligned}
\mathbb{E}[\text{Regret}_{\textbf{Alg}}(T)] &= \sum_{m=1}^{M} \mathbb{E}[\text{Regret}_{m,\textbf{Alg1}}] \\
&\le \sum_{m=1}^{M} \mathbb{E}[\text{Regret}_{m,\textbf{Alg}}] + \bigg(\sum_{m=1}^{M} \delta_m\bigg) Cd\sqrt{T/M}\log(T/M) \\
&= \mathbb{E}[\text{Regret}_{\textbf{Alg}}] + \delta Cd\sqrt{T/M}\log(T/M), \qquad \text{(D.4)}
\end{aligned}$$

where $\delta = \sum_{m=1}^{M} \delta_m$ is the expected communication complexity. Combining (D.2) and (D.4), for any algorithm **Alg** with communication complexity $\delta \le c/(2C) \cdot M/\log(T/M) = O(M/\log(T/M))$, we have

$$\mathbb{E}[\text{Regret}_{\textbf{Alg}}] \ge cd\sqrt{MT} - \delta Cd\sqrt{T/M}\log(T/M) \ge cd\sqrt{MT}/2 = \Omega(d\sqrt{MT}).$$

This finishes the proof of Theorem 5.3. □

E Auxiliary Lemmas

To make the analysis self-contained in this paper, here we include the auxiliary lemmas that have been previously used.

Lemma E.1 (Lemma 2.2 in Tie et al. 2011)**.** For any positive semi-definite matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$, it holds that $\det(\mathbf{A}+\mathbf{B}+\mathbf{C}) + \det(\mathbf{A}) \ge \det(\mathbf{A}+\mathbf{B}) + \det(\mathbf{A}+\mathbf{C})$.

Lemma E.2 (Lemma 2.3 in Tie et al. 2011)**.** For any positive semi-definite matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$, it holds that $\det(\mathbf{A}+\mathbf{B}+\mathbf{C})\det(\mathbf{A}) \le \det(\mathbf{A}+\mathbf{B})\det(\mathbf{A}+\mathbf{C})$.

Theorem E.3 (Theorem 2 in Abbasi-Yadkori et al. 2011)**.** Suppose $\{\mathcal{F}_t\}_{t=0}^{\infty}$ is a filtration. Let $\{\eta_t\}_{t=1}^{\mathbb{R}}$ be a stochastic process in $\mathbb{R}$ such that η_t is $\mathcal{F}_t$-measurable and R-sub-Gaussian conditioning on $\mathcal{F}_{t-1}$, i.e, for any $c > 0$,

$$\mathbb{E}\left[\exp\left(c\eta_t\right)|\mathcal{F}_{t-1}\right] \le \exp\left(\frac{c^2R^2}{2}\right).$$

Let $\{\mathbf{x}_t\}_{t=1}^{\infty}$ be a stochastic process in $\mathbb{R}^d$ such that $\mathbf{x}_t$ is $\mathcal{F}_{t-1}$-measurable and $\|\mathbf{x}_t\|_2 \le L$. Let $y_t = \langle \mathbf{x}_t, \boldsymbol{\theta}^*\rangle + \eta_t$ for some $\boldsymbol{\theta}^* \in \mathbb{R}^d$ s.t. $\|\boldsymbol{\theta}^*\|_2 \le S$. For any $t \ge 1$, define

$$\boldsymbol{\Sigma}_t = \lambda\mathbf{I} + \sum_{i=1}^{t} \mathbf{x}_t\mathbf{x}_t^\top, \quad \text{and} \quad \widehat{\boldsymbol{\theta}}_t = \boldsymbol{\Sigma}_t^{-1}\sum_{i=1}^{t}\mathbf{x}_i y_i,$$

for some $\lambda > 0$. Then for any $\delta > 0$, with probability at least $1-\delta$, for all t, we have

$$\|\widehat{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_t} \le R\sqrt{d\log\left(\frac{1+tL^2/\lambda}{\delta}\right)} + \sqrt{\lambda}S.$$

Lemma E.4 (Lemma 12 in Abbasi-Yadkori et al. (2011))**.** Suppose $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{d\times d}$ are two positive definite matrices satisfying that $\mathbf{A} \succeq \mathbf{B}$, then for any $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{x}\|_{\mathbf{A}} \le \|\mathbf{x}\|_{\mathbf{B}} \cdot \sqrt{\det(\mathbf{A})/\det(\mathbf{B})}$.