

---

# Differentially-Private Federated Linear Bandits

---

Abhimanyu Dubey and Alex Pentland  
Media Lab and Institute for Data, Systems and Society  
Massachusetts Institute of Technology  
{dubeya, pentland}@mit.edu

## Abstract

The rapid proliferation of decentralized learning systems mandates the need for differentially-private cooperative learning. In this paper, we study this in context of the contextual linear bandit: we consider a collection of agents cooperating to solve a common contextual bandit, while ensuring that their communication remains private. For this problem, we devise FEDUCB, a multiagent private algorithm for both centralized and decentralized (peer-to-peer) federated learning. We provide a rigorous technical analysis of its utility in terms of regret, improving several results in cooperative bandit learning, and provide rigorous privacy guarantees as well. Our algorithms provide competitive performance both in terms of pseudoregret bounds and empirical benchmark performance in various multi-agent settings.

**Note:** This paper appears in full as a spotlight presentation at NeurIPS 2020.

## 1 Introduction

The objective of the federated paradigm is to allow cooperative estimation with larger amounts of data (from multiple clients, devices, etc.) while keeping the data decentralized [22]. There has been a surge of interest in this problem from both academia and industry, owing to its overall applicability. Most research on provably private algorithms in the federated setting has been on distributed supervised learning [23] and optimization [17]. The contextual bandit problem, however, is a very interesting candidate for private methods, since the involved contexts and rewards both typically contain sensitive user information [33]. There is an increasing body of work on online learning and multi-armed bandits in cooperative settings [9, 26, 34], and private single-agent learning [36, 33], but methods for private *federated* bandit learning are still elusive, despite their immediate applicability.

**Contributions.** In this paper, we study the *federated* contextual bandit problem under constraints of differential privacy. We consider two popular paradigms: (a) *centralized* learning, where a central controller coordinates different clients [22, 43] and (b) *decentralized* peer-to-peer learning *with delays*, where agents communicate directly with each other, without a controller [12, 34, 26, 25, 27]. We provide a rigorous formulation of  $(\epsilon, \delta)$ -differential privacy in the *federated* contextual bandit, and present two variants of FEDUCB, the first federated algorithm ensuring that each agent is private with respect to the data from all other agents, and provides a parameter to control communication.

Next, we prove rigorous bounds on the cumulative group pseudoregret obtained by FEDUCB. In the centralized setting, we prove a high probability regret bound of  $\tilde{O}(d^{3/4}\sqrt{MT/\epsilon})$  which matches the non-private bound in terms of its dependence on  $T$  and  $M$ . In the decentralized case, we prove a corresponding regret bound of  $\tilde{O}(d^{3/4}\sqrt{(\text{diameter}(\mathcal{G}))MT/\epsilon})$ , where  $\mathcal{G}$  is the communication network between agents. In addition to the regret bounds, we present a novel analysis of communication complexity, and its connections with the privacy budget and regret.

**All proofs have been deferred to the appendix for brevity.**

## 2 Related Work

**Multi-Agent and Distributed Bandits.** Bandit learning in multi-agent distributed settings has received attention from several academic communities. Channel selection in distributed radio networks consider the (context-free) multi-armed bandit with collisions [30, 32, 31] and cooperative estimation over a network with delays [26, 25, 27]. For the contextual case, recent work has considered non-private estimation in networks with delays [11, 9, 42, 24]. A closely-related problem is that of bandits with side information [6, 4], where the single learner obtains multiple observations every round, similar to the multi-agent communicative setting. Our work builds on the remarkable work of Abbasi-Yadkori *et al.*[1], which in turn improves the LinUCB algorithm introduced in [28].

**Differential Privacy.** Our work utilizes *differential privacy*, a mathematically rigorous framework introduced by Dwork [13, 15] that requires the behavior of an algorithm to fluctuate only slightly (in probability) with any change in its inputs. A technique to maintain differential privacy for the continual release of statistics was introduced in [8, 14], known as the *tree-based* algorithm that privatizes the partial sums of  $n$  entries by adding at most  $\log n$  noisy terms. This method has been used to preserve privacy across several online learning problems, including convex optimization [21, 20], online data analysis [19], collaborative filtering [5] and data aggregation [7]. In the single-agent bandit setting, differential privacy using tree-based algorithms have been explored in the multi-armed case [38, 35, 40] and the contextual case [36]. In particular, our work uses several elements from [36], extending their single-agent results to the federated multi-agent setting. For the multi-agent multi-armed (i.e., context-free) bandit problem, differentially private algorithms have been devised for the centralized [39] and decentralized [10] settings. Empirically, the advantages of privacy-preserving contextual bandits has been demonstrated in the work of Malekzadeh *et al.*[33], and Hannun *et al.*[18] consider a centralized multi-agent contextual bandit algorithm that use secure multi-party computations to provide privacy guarantees (both works do not have any regret guarantees).

## 3 Background and Preliminaries

We use boldface to represent vectors, e.g.,  $\mathbf{x}$ , and matrices, e.g.,  $\mathbf{X}$ . For any matrix  $\mathbf{X}$ , its Gram matrix is given by  $\mathbf{X}^\top \mathbf{X}$ . Any symmetric matrix  $\mathbf{X}$  is positive semi-definite (PSD) if  $\mathbf{y}^\top \mathbf{X} \mathbf{y} \geq 0$  for any vector  $\mathbf{y}$ , and we denote PSD by  $\mathbf{X} \succcurlyeq \mathbf{0}$ . For any PSD matrix  $\mathbf{X}$  we denote the ellipsoid  $\mathbf{X}$ -norm of vector  $\mathbf{y}$  as  $\|\mathbf{y}\|_{\mathbf{X}} = \sqrt{\mathbf{y}^\top \mathbf{X} \mathbf{y}}$ . For two matrices  $\mathbf{X}$  and  $\mathbf{Y}$ ,  $\mathbf{X} \succcurlyeq \mathbf{Y}$  implies that  $\mathbf{X} - \mathbf{Y} \succcurlyeq \mathbf{0}$ . For any PSD matrix  $\mathbf{X}$  and vectors  $\mathbf{u}, \mathbf{v}$ , we denote the  $\mathbf{X}$ -ellipsoid inner product as  $\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{X}} = \mathbf{u}^\top \mathbf{X} \mathbf{v}$ , and drop the subscript when  $\mathbf{X} = \mathbf{I}$ . We denote the set  $\{a, a + 1, \dots, b - 1, b\}$  by the shorthand  $[a, b]$ , and if  $a = 1$  we refer to it as  $[b]$ . We denote the  $\gamma^{\text{th}}$  power of graph  $\mathcal{G}$  as  $\mathcal{G}_\gamma$  ( $\mathcal{G}_\gamma$  has edge  $(i, j)$  if the shortest distance between  $i$  and  $j$  in  $\mathcal{G}$  is  $\leq \gamma$ ).  $N_\gamma(v)$  denotes the set of nodes in  $\mathcal{G}$  at a distance of at most  $\gamma$  (including itself) from node  $v \in \mathcal{G}$ .

**Federated Contextual Bandit.** This is an extension of the linear contextual bandit [28, 1] involving a set of  $M$  agents. At every trial  $t \in [T]$ , each agent  $i \in [M]$  is presented with a *decision set*  $\mathcal{D}_{i,t} \subset \mathbb{R}^d$  from which it selects an action  $\mathbf{x}_{i,t} \in \mathbb{R}^d$ . It then obtains a reward  $y_{i,t} = \mathbf{x}_{i,t}^\top \boldsymbol{\theta}^* + \eta_{i,t}$  where  $\boldsymbol{\theta}^* \in \mathbb{R}^d$  is an unknown (but fixed) parameter and  $\eta_{i,t}$  is a noise parameter sampled i.i.d. every trial for every agent. The objective of the agents is to minimize the cumulative *group pseudoregret*:<sup>1</sup>

$$\mathcal{R}_M(T) = \sum_{i=1, t=1}^{M, T} \langle \mathbf{x}_{i,t}^* - \mathbf{x}_{i,t}, \boldsymbol{\theta}^* \rangle, \text{ where } \mathbf{x}_{i,t}^* = \arg \max_{\mathbf{x} \in \mathcal{D}_{i,t}} \langle \mathbf{x}, \boldsymbol{\theta}^* \rangle \text{ is optimal.}$$

In *non-private* distributed contextual bandits, a group pseudoregret of  $\tilde{O}(\sqrt{dMT})$  has been achieved [42, 43], which is the order-optimality we seek in our algorithms as well.

*Assumptions.* We assume: (a) bounded action set:  $\forall i, t, \|\mathbf{x}_{i,t}\| \leq L$ , (b) bounded mean reward:  $\forall \mathbf{x}, \langle \boldsymbol{\theta}^*, \mathbf{x} \rangle \leq 1$ , (c) bounded target parameter:  $\|\boldsymbol{\theta}^*\| \leq S$ , (d) sub-Gaussian rewards:  $\forall i, t, \eta_{i,t}$  is  $\sigma$ -sub-Gaussian, (e) bounded decision set:  $\forall i, t, \mathcal{D}_{i,t}$  is compact, (f) bounded reward:  $\forall i, t, |y_{i,t}| \leq B$ .<sup>2</sup>

**Differential Privacy.** The contextual bandit problem involves two sets of variables that any agent must private to the other participating agents – the available decision sets  $(\mathcal{D}_{i,t})_{t \in [T]}$  and observed

<sup>1</sup>The *pseudoregret* is an expectation (over the randomness of  $\eta_{i,t}$ ) of the stochastic quantity *regret*, and is more amenable to high-probability bounds. However, a bound over the pseudoregret can also bound the regret with high probability, e.g., by a Hoeffding concentration (see, e.g., [41]).

<sup>2</sup>Assuming bounded rewards is necessary for the privacy mechanism, and is standard in the literature [3, 2].

rewards  $(y_{i,t})_{t \in [T]}$ . However, the communication between agents only depends on the chosen actions  $(\mathbf{x}_{i,t})_{t \in [T]}$  (and not all of  $\mathcal{D}_{i,t}$ ), and hence making our technique differentially private with respect to  $((\mathbf{x}_{i,t}, y_{i,t}))_{t \in [T]}$  will suffice. We first denote two sequences  $\mathbf{S}_i = ((\mathbf{x}_{i,t}, y_{i,t}))_{t \in [T]}$  and  $\mathbf{S}'_i = ((\mathbf{x}'_{i,t}, y'_{i,t}))_{t \in [T]}$  as  $t$ -neighbors if for each  $t' \neq t$ ,  $(\mathbf{x}_{i,t}, y_{i,t}) = (\mathbf{x}'_{i,t}, y'_{i,t})$ . We can now provide the formal definition for federated differential privacy:

**Definition 1** (Federated Differential Privacy for Contextual Bandits). *In a federated learning setting with  $M \geq 2$  agents, a randomized multiagent contextual bandit algorithm  $A = (A_i)_{i=1}^M$  is  $(\varepsilon, \delta, M)$ -federated differentially private under continual multi-agent observation if for any  $i, j$  s.t.  $i \neq j$ , any  $t$  and set of sequences  $\mathbf{S}_i = (S_k)_{k=1}^M$  and  $\mathbf{S}'_i = (S_k)_{k=1, k \neq i}^M \cup S'_i$  such that  $\mathbf{S}_i$  and  $\mathbf{S}'_i$  are  $t$ -neighboring, and any subset of actions  $\mathcal{S}_{j, \geq t} \subset \mathcal{D}_{j,t} \times \mathcal{D}_{j,t+1} \times \dots \times \mathcal{D}_{j,T}$  of actions from rounds  $t$  to  $T$ , it holds that:*

$$\mathbb{P}(A_j(\mathbf{S}_i) \in \mathcal{S}_{j, \geq t}) \leq e^\varepsilon \cdot \mathbb{P}(A_j(\mathbf{S}'_i) \in \mathcal{S}_{j, \geq t}) + \delta.$$

## 4 Federated LinUCB with Differential Privacy

In this section, we introduce our algorithm for federated learning with differential privacy. For the remainder of this section, for exposition, we consider the single-agent setting and drop an additional index subscript we use in the actual algorithm (e.g., we refer to the action at time  $t$  as  $\mathbf{x}_t$  and not  $\mathbf{x}_{i,t}$  for agent  $i$ ). We build on the celebrated LinUCB algorithm, an application of the optimism heuristic to the linear bandit case [28, 1], designed for the single-agent problem. The central idea of the algorithm is, at every round  $t$ , to construct a *confidence set*  $\mathcal{E}_t$  that contains  $\theta^*$  with high probability, followed by computing an upper confidence bound on the reward of each action within the decision set  $\mathcal{D}_t$ , and finally selecting the action with the largest UCB, i.e.,  $\mathbf{x}_t = \arg \max_{\mathbf{x} \in \mathcal{D}_t} (\max_{\theta \in \mathcal{E}_t} \langle \mathbf{x}, \theta \rangle)$ . The confidence set is an ellipsoid centered on the regularized linear regression estimate (for  $\mathbf{X}_{<t} = [\mathbf{x}_1^\top \ \mathbf{x}_2^\top \ \dots \ \mathbf{x}_{t-1}^\top]^\top$  and  $\mathbf{y}_{<t} = [y_1 \ y_2 \ \dots \ y_{t-1}]^\top$ ):

$$\mathcal{E}_t := \left\{ \theta \in \mathbb{R}^d : \|\theta - \hat{\theta}_t\|_{\mathbf{V}_t} \leq \beta_t \right\}, \text{ where } \hat{\theta}_t := \arg \min_{\theta \in \mathbb{R}^d} [\|\mathbf{X}_{<t} \theta - \mathbf{y}_{<t}\|_2^2 + \|\theta\|_{\mathbf{H}_t}^2].$$

The regression solution can be given by  $\hat{\theta}_t := (\mathbf{G}_t + \mathbf{H}_t)^{-1} \mathbf{X}_{<t}^\top \mathbf{y}_{<t}$ , where  $\mathbf{G}_t = \mathbf{X}_{<t}^\top \mathbf{X}_{<t}$  is the Gram matrix of actions,  $\mathbf{H}_t$  is a (time-varying) regularizer, and  $\beta_t$  is an appropriately chosen confidence parameter. Typically in non-private settings, the regularizer is constant, i.e.,  $\mathbf{H}_t = \lambda \mathbf{I} \ \forall t, \lambda > 0$  [1, 28], however, in our case, we will carefully select  $\mathbf{H}_t$  to introduce privacy, using a strategy similar to [36]. Given  $\mathbf{V}_t = \mathbf{G}_t + \mathbf{H}_t$ , let  $\text{UCB}_t(\mathbf{x}; \theta) = \langle \theta, \mathbf{x} \rangle + \beta_t \|\mathbf{x}\|_{\mathbf{V}_t^{-1}}$ .

In the federated setting, since there are  $M$  learners that have distinct actions, the communication protocol is a key component of algorithm design: communication often creates *heterogeneity* between agents, e.g., for any two agents, their estimators  $(\hat{\theta}_t)$  at any instant are certainly distinct, and the algorithm must provide a control over this heterogeneity, to bound the group regret. We additionally require that communication between agents is  $(\varepsilon, \delta)$ -private, making the problem more challenging.

### 4.1 Centralized Environment with a Controller

We first consider the centralized communication environment where there exists a controller that coordinates communication between different agents, as is typical in large-scale distributed learning []. We consider a set of  $M$  agents that each are interacting with the contextual bandit, and periodically communicate with the controller, that synchronizes them with other agents. We present the algorithm CENTRALIZED FEDUCB in Algorithm 1 that details our approach.

The central idea of the algorithm is to carefully perturb the Gram matrices  $\mathbf{V}_{i,t}$  and the reward vector  $\mathbf{u}_{i,t}$  with random noise based on the sensitivity of these elements and the level of privacy required, as is typical in the literature [36]. First, each agent updates its local (unsynchronized) estimates. These are used to construct the UCB in a manner identical to the standard OFUL algorithm [1]. If, for any agent  $i$ , the log-determinant of the local Gram matrix exceeds the synchronized Gram matrix ( $\mathbf{S}_{i,t}$ ) by an amount  $D/\Delta t_i$  (where  $\Delta t_i$  is the time since the last synchronization), then it sends a signal to the controller, that synchronizes *all* agents with their latest action/reward pairs. The synchronization is done using a *privatized* version of the Gram matrix and rewards, carried out by the subroutine PRIVATIZER (Section 4.3, Alg. 2). This synchronization ensures that the *heterogeneity* between the agents is controlled, allowing us to control the overall regret and limit communication as well:

---

**Algorithm 1** CENTRALIZED FEDUCB( $D, M, T, \rho_{\min}, \rho_{\max}$ )

---

```
1: Initialization:  $\forall i$ , set  $\mathbf{S}_{i,1} \leftarrow \mathbf{0}$ ,  $\mathbf{s}_{i,1} \leftarrow \mathbf{0}$ ,  $\widehat{\mathbf{Q}}_{i,0} \leftarrow \mathbf{0}$ ,  $\mathbf{U}_{i,1} \leftarrow \rho_{\min} \mathbf{I}$ ,  $\bar{\mathbf{u}}_{i,1} \leftarrow \mathbf{0}$ .
2: for For each iteration  $t \in [T]$  do
3:   for For each agent  $i \in [M]$  do
4:     Set  $\mathbf{V}_{i,t} \leftarrow \mathbf{S}_{i,t} + \mathbf{U}_{i,t}$ ,  $\tilde{\mathbf{u}}_{i,t} \leftarrow \mathbf{s}_{i,t} + \bar{\mathbf{u}}_{i,t}$ .
5:     Receive  $\mathcal{D}_{i,t}$  from environment.
6:     Compute regressor  $\tilde{\boldsymbol{\theta}}_{i,t} \leftarrow \mathbf{V}_{i,t}^{-1} \tilde{\mathbf{u}}_{i,t}$ .
7:     Compute  $\beta_{i,t}$  following Proposition 2.
8:     Select  $\mathbf{x}_{i,t} \leftarrow \arg \max_{\mathbf{x} \in \mathcal{D}_{i,t}} \langle \mathbf{x}, \tilde{\boldsymbol{\theta}}_{i,t} \rangle + \beta_{i,t} \|\mathbf{x}\|_{\mathbf{V}_{i,t}^{-1}}$ .
9:     Obtain  $y_{i,t}$  from environment.
10:    Update  $\mathbf{U}_{i,t+1} \leftarrow \mathbf{U}_{i,t} + \mathbf{x}_{i,t} \mathbf{x}_{i,t}^\top$ ,  $\mathbf{u}_{i,t+1} \leftarrow \mathbf{u}_{i,t} + \mathbf{x}_{i,t} y_{i,t}$ .
11:    Update  $\widehat{\mathbf{Q}}_{i,t} \leftarrow \widehat{\mathbf{Q}}_{i,t-1} + [\mathbf{x}_{i,t}^\top y_{i,t}]^\top [\mathbf{x}_{i,t}^\top y_{i,t}]$ 
12:    if  $\log \det(\mathbf{V}_{i,t} + \mathbf{x}_{i,t} \mathbf{x}_{i,t}^\top) + M(\rho_{\max} - \rho_{\min}) \mathbf{I} - \log \det(\mathbf{S}_{i,t}) \geq D/\Delta t_i$  then
13:      SYNCHRONIZE  $\leftarrow$  TRUE.
14:    end if
15:    if SYNCHRONIZE then
16:       $[\forall \text{ AGENTS}]$  Agent sends  $\widehat{\mathbf{Q}}_{i,t} \rightarrow$  PRIVATIZER and gets  $\widehat{\mathbf{U}}_{i,t+1}, \widehat{\mathbf{u}}_{i,t+1} \leftarrow$  PRIVATIZER.
17:       $[\forall \text{ AGENTS}]$  Agent communicates  $\widehat{\mathbf{U}}_{i,t+1}, \widehat{\mathbf{u}}_{i,t+1}$  to controller.
18:       $[\text{CONTROLLER}]$  Compute  $\mathbf{S}_{t+1} \leftarrow \mathbf{S}_t + \sum_{i=1}^M \widehat{\mathbf{U}}_{i,t+1}$ ,  $\mathbf{s}_{t+1} \leftarrow \mathbf{s}_t + \sum_{i=1}^M \widehat{\mathbf{u}}_{i,t+1}$ .
19:       $[\text{CONTROLLER}]$  Communicate  $\mathbf{S}_{t+1}, \mathbf{s}_{t+1}$  back to agent.
20:       $[\forall \text{ AGENTS}]$   $\mathbf{S}_{i,t+1} \leftarrow \mathbf{S}_{t+1}$ ,  $\mathbf{s}_{i,t+1} \leftarrow \mathbf{s}_{t+1}$ ,  $\Delta t_i \leftarrow 0$ ,  $\mathbf{U}_{i,t+1} \leftarrow \rho_{\min} \mathbf{I}$ ,  $\bar{\mathbf{u}}_{i,t+1} \leftarrow \mathbf{0}$ .
21:       $[\forall \text{ AGENTS}]$   $\widehat{\mathbf{Q}}_{i,t+1} \leftarrow \mathbf{0}$ .
22:    else
23:       $\mathbf{S}_{i,t+1} \leftarrow \mathbf{S}_{i,t}$ ,  $\mathbf{s}_{i,t+1} \leftarrow \mathbf{s}_{i,t}$ ,  $\Delta t_i \leftarrow \Delta t_i + 1$ .
24:    end if
25:  end for
26: end for
```

---

**Proposition 1** (Communication Complexity). *If Algorithm 1 is run with threshold  $D$ , then total rounds of communication  $n$  is upper bounded by  $2\sqrt{(dT/D)} \cdot \log(\rho_{\max}/\rho_{\min} + TL^2/d\rho_{\min}) + 4$ .*

The PRIVATIZER subroutine provides suitable perturbations based on the privacy budget ( $\varepsilon$  and  $\delta$ ), and is inspired by the single-agent private algorithm of [36]. In this paper, we assume these to be identical for all agents (however, the algorithm and analysis hold for unique privacy budgets as well, as long as a lower bound on the budgets is known). In turn, the quantities  $\varepsilon$  and  $\delta$  affect the algorithm (and regret) via the quantities  $\rho_{\min}$ ,  $\rho_{\max}$ , and  $\kappa$  which bound the norm of perturbations at any round:

**Definition 2** (Sparsely-accurate  $\rho_{\min}$ ,  $\rho_{\max}$  and  $\kappa$ ). *Consider a subsequence  $\bar{\sigma}$  of  $[T] = 1, \dots, T$  of size  $n$ . The bounds  $0 \leq \rho_{\min} \leq \rho_{\max}$  and  $\kappa$  are  $(\alpha/2nM, \bar{\sigma})$ -accurate for  $(\mathbf{H}_{i,t})_{i \in [M], t \in \bar{\sigma}}$  and  $(\mathbf{h}_{i,t})_{i \in [M], t \in \bar{\sigma}}$ , if, for each round  $t \in \bar{\sigma}$  and agent  $i$ :*

$$\|\mathbf{H}_{i,t}\| \leq \rho_{\max}, \quad \|\mathbf{H}_{i,t}^{-1}\| \leq 1/\rho_{\min}, \quad \|\mathbf{h}_{i,t}\|_{\mathbf{H}_{i,t}^{-1}} \leq \kappa; \quad \text{with probability at least } (1 - \alpha/2nM).$$

The motivation for obtaining accurate bounds  $\rho_{\min}$ ,  $\rho_{\max}$  and  $\kappa$  stems from the fact that in the non-private case, the quantities that determine regret are not stochastic conditioned on the obtained sequence  $(\mathbf{x}_t, y_t)_{t \in [T]}$ , whereas the addition of stochastic regularizers in the private case requires us to have control over their perturbations (as described above). To form the UCB, we additionally require a suitable confidence threshold  $\beta_{i,t}$  for each agent, which is defined as follows:

**Definition 3** (Accurate  $(\beta_{i,t})_{i \in [M], t \in [T]}$ ). *A sequence  $(\beta_{i,t})_{i \in [M], t \in [T]}$  is  $(\alpha, M, T)$ -accurate for  $(\mathbf{H}_{i,t})_{i \in [M], t \in [T]}$  and  $(\mathbf{h}_{i,t})_{i \in [M], t \in [T]}$ , if it satisfies  $\|\tilde{\boldsymbol{\theta}}_{i,t} - \boldsymbol{\theta}^*\|_{\mathbf{V}_{i,t}} \leq \beta_{i,t}$  with probability at least  $1 - \alpha$  for all rounds  $t = 1, \dots, T$  and agents  $i = 1, \dots, M$  simultaneously.*

**Proposition 2.** *Consider an instance of the problem where synchronization occurs exactly  $n$  times on instances  $\bar{\sigma}$ , up to and including  $T$  trials, and  $\rho_{\min}$ ,  $\rho_{\max}$  and  $\kappa$  are  $(\alpha/2nM)$ -accurate. Then, for Algorithm 1, the sequence  $(\beta_{i,t})_{i \in [M], t \in [T]}$  is  $(\alpha, M, T)$ -accurate where:*

$$\beta_{i,t} := \sigma \sqrt{2 \log(2/\alpha) + d \log(\det(\mathbf{V}_{i,t})) - d \log(M\rho_{\min})} + MS\sqrt{\rho_{\max}} + M\gamma.$$

The key point here is that for the desired levels of privacy ( $\varepsilon, \delta$ ) and synchronization rounds ( $n$ ), we can calculate appropriate  $\rho_{\min}$ ,  $\rho_{\max}$  and  $\kappa$ , which in turn provide us a UCB algorithm with

guarantees. Our basic techniques are similar to [1, 36], however, the multi-agent setting introduces several new aspects in the analysis, such as the control of heterogeneity.

**Theorem 1** (Private Group Regret with Centralized Controller). *Assuming Proposition 2 holds, and synchronization occurs in at least  $n = \Omega(d \log(\rho_{\max}/\rho_{\min} + TL^2/d\rho_{\min}))$  rounds, Algorithm 1 obtains the following group pseudoregret with probability at least  $1 - \alpha$ :*

$$\mathcal{R}_M(T) = O\left(\sigma\sqrt{MTd}\left(\log(\rho_{\max}/\rho_{\min} + TL^2/d\rho_{\min}) + \sqrt{\log 2/\alpha} + MS\sqrt{\rho_{\max}} + M\kappa\right)\right).$$

Furthermore, if the mean reward of the best arm and second best arm are separated by a gap  $\geq \Delta$  for all  $i \in [M], t \in [T]$ , then Alg. 1 obtains the following gap-dependent pseudoregret w.p.  $\geq (1 - \alpha)$ :

$$\mathcal{R}_M(T) = O\left((\sigma d/\Delta)\left(\log(\rho_{\max}/\rho_{\min} + TL^2/d\rho_{\min}) + \sqrt{\log 2/\alpha}\right)^{3/2}\right).$$

This regret bound is obtained by setting  $D = 2Td(\log(\rho_{\max}/\rho_{\min} + TL^2/d\rho_{\min}) + 1)^{-1}$ , which therefore ensures  $O(M \log T)$  total communication (by Proposition 1). Theorem 1 demonstrates the relationship between communication complexity (i.e., number of synchronization rounds) and the regret bound for a fixed privacy budget, via the dependence on the bounds  $\rho_{\min}, \rho_{\max}$  and  $\kappa$ . We now present similar results (for a fixed privacy budget) on group regret for the decentralized setting, which is more involved, as the delays and lack of a centralized controller make it difficult to control the heterogeneity of information between agents. Subsequently, in the next section, we will present results on the privacy guarantees of our algorithms.

## 4.2 Decentralized Peer-to-Peer Environment

In this environment, we assume that the collection of agents communicate by directly sending messages to each other. The communication network is denoted by an undirected (connected) graph  $\mathcal{G} = (V, E)$ , where, edge  $e_{ij} \in E$  if agents  $i$  and  $j$  can communicate directly. The protocol operates as follows: every trial, each agent interacts with their respective bandit, and obtains a reward. At any trial  $t$ , after receiving the reward, each agent  $v$  sends the message  $\mathbf{m}_{v,t}$  to all its neighbors in  $\mathcal{G}$ . This message is forwarded from agent to agent  $\gamma$  times (taking one trial of the bandit problem each between forwards), after which it is dropped. This communication protocol, based on the *time-to-live* (delay) parameter  $\gamma \leq \text{diam}(\mathcal{G})$  is a common technique to control communication complexity, known as the LOCAL protocol [16, 29, 37]. Each agent  $v \in V$  therefore also receives messages  $\mathbf{m}_{v',t-d(v,v')}$  from all the agents  $v'$  such that  $d(v, v') \leq \gamma$ , i.e., from all agents in  $N_\gamma(v)$ . We defer the details for this algorithm to the full paper, and provide the associated regret bound for brevity.

**Theorem 2** (Decentralized Private Group Regret). *Assuming Proposition 2 holds, and synchronization occurs in at least  $n = \Omega(d(\bar{\chi}(\mathcal{G}_\gamma) \cdot \gamma)(1 + L^2)^{-1} \log(\rho_{\max}/\rho_{\min} + TL^2/d\rho_{\min}))$  rounds, decentralized FEDUCB obtains the following group pseudoregret with probability at least  $1 - \alpha$ :*

$$\mathcal{R}_M(T) = O\left(\sigma\sqrt{M(\bar{\chi}(\mathcal{G}_\gamma) \cdot \gamma)Td}\left(\log\left(\frac{\rho_{\max}}{\rho_{\min}} + \frac{TL^2}{\gamma d\rho_{\min}}\right) + \sqrt{\log \frac{2}{\alpha}} + MS\sqrt{\rho_{\max}} + M\kappa\right)\right).$$

**Remark 1** (Decentralized Group Regret). *Decentralized FEDUCB obtains an identical dependence on the privacy bounds  $\rho_{\min}, \rho_{\max}$  and  $\kappa$  and horizon  $T$  as Algorithm 1, since the underlying bandit subroutines are identical for both. The key difference is in additional the leading factor of  $\sqrt{\bar{\chi}(\mathcal{G}_\gamma) \cdot \gamma}$ , which arises from the delayed spread of information: if  $\mathcal{G}$  is dense, e.g., complete, then  $\gamma = 1$  and  $\bar{\chi}(\mathcal{G}_\gamma) = 1$ , since there is only one clique of  $\mathcal{G}$ . In the worst case, if  $\mathcal{G}$  is a line graph, then  $\bar{\chi}(\mathcal{G}_\gamma) = M/\gamma$ , giving an additional factor of  $M$  (i.e., it is as good as each agent acting individually). In more practical scenarios, we expect  $\mathcal{G}$  to be hierarchical, and expect a delay overhead of  $o(1)$ . Note that since each agent only communicates within its clique, the factor  $\bar{\chi}(\mathcal{G}_\gamma)$  is optimal, see [11].*

## 4.3 Privacy Guarantees

We now discuss the privacy guarantees for both algorithms. Here we present results for the centralized algorithm, but our results hold (almost identically) for the decentralized case as well (see appendix). Note that each agent interacts with data from other agents only via the cumulative parameters  $\mathbf{S}_t$  and  $\mathbf{s}_t$ . These, in turn, depend on  $\mathbf{Z}_{i,t} = \mathbf{U}_{i,t} + \mathbf{H}_{i,t}$  and  $\mathbf{z}_{i,t} = \bar{\mathbf{u}}_{i,t} + \mathbf{h}_{i,t}$  for each agent  $i$ , on the instances  $t$  that synchronization occurs.

---

**Algorithm 2** PRIVATIZER( $\varepsilon, \delta, M, T$ ) for any agent  $i$ 


---

- 1: **Initialization:**
  - 2: If communication rounds  $n$  are fixed *a priori*, set  $m \leftarrow 1 + \lceil \log n \rceil$ , else  $m \leftarrow 1 + \lceil \log T \rceil$ .
  - 3: Create binary tree  $\mathcal{T}$  of depth  $m$ .
  - 4: **for** node  $n$  in  $\mathcal{T}$  **do**
  - 5:   Sample noise  $\widehat{\mathbf{N}} \in \mathbb{R}^{(d+1) \times (d+1)}$ , where  $\widehat{\mathbf{N}}_{ij} \sim \mathcal{N}(0, 16m(L^2 + 1)^2 \log(2/\delta)^2 / \varepsilon^2)$ .
  - 6:   Store  $\mathbf{N} = (\widehat{\mathbf{N}} + \widehat{\mathbf{N}}^\top) / \sqrt{2}$  at node  $n$ .
  - 7: **end for**
  - 8: **Runtime:**
  - 9: **for** each communication round  $t \leq n$  **do**
  - 10:   Receive  $\widehat{\mathbf{Q}}_{i,t}$  from agent, and update corresponding nodes in  $\mathcal{T}$  (see [21], Alg. 5).
  - 11:   Compute  $\mathbf{M}_{i,t+1}$  using the least nodes of  $\mathcal{T}$  (see [21], Alg. 5).
  - 12:   Set  $\widehat{\mathbf{U}}_{i,t+1} = \mathbf{U}_{i,t+1} + \mathbf{H}_{i,t}$  as top-left  $d \times d$  submatrix of  $\mathbf{M}_{i,t+1}$ .
  - 13:   Set  $\widehat{\mathbf{u}}_{i,t+1} = \mathbf{u}_{i,t+1} + \mathbf{h}_{i,t}$  as first  $d$  entries of last column of  $\mathbf{M}_{i,t+1}$ .
  - 14:   Return  $\widehat{\mathbf{U}}_{i,t+1}, \widehat{\mathbf{u}}_{i,t+1}$  to agent.
  - 15: **end for**
- 

**Proposition 3** (see [13, 36]). *Consider  $n \leq T$  synchronization rounds occurring on trials  $\bar{\sigma} \subseteq [T]$ . If the sequence  $(\mathbf{Z}_{i,t}, \mathbf{z}_{i,t})_{t \in \bar{\sigma}}$  is  $(\varepsilon, \delta)$ -differentially private with respect to  $(\mathbf{x}_{i,t}, \mathbf{y}_{i,t})_{t \in [T]}$ , for each agent  $i \in [M]$ , then all agents are  $(\varepsilon, \delta, M)$ -federated differentially private.*

The implementation of the private release of  $(\mathbf{U}_{i,t}, \bar{\mathbf{u}}_{i,t})$  is done by the ubiquitous tree-based mechanism for partial sums. Following [36], we also aggregate both into a single matrix  $\mathbf{M}_{i,t} \in \mathbb{R}^{(d+1) \times (d+1)}$ , by first concatenating:  $\mathbf{A}_{i,t} := [\mathbf{X}_{i,1:t}, \mathbf{y}_{i,1:t}] \in \mathbb{R}^{t \times (d+1)}$ , and then computing  $\mathbf{M}_{i,t} = \mathbf{A}_{i,t}^\top \mathbf{A}_{i,t}$ . Furthermore, the update is straightforward:  $\mathbf{M}_{i,t+1} = \mathbf{M}_{i,t} + [\mathbf{x}_{i,t} \ \mathbf{y}_{i,t}]^\top [\mathbf{x}_{i,t} \ \mathbf{y}_{i,t}]$ . The tree mechanism requires at most  $m = 1 + \lceil \log_2 T \rceil$  counters, therefore noise that ensures each node guarantees  $(\varepsilon / \sqrt{8m \ln(2/\delta)}, \delta/2m)$ -privacy is sufficient to make the sequence  $(\varepsilon, \delta)$ -private.

**Tree-Based Mechanism.** The tree-based mechanism [14] for differential privacy maintains a binary tree  $\mathcal{T}$ , where the leaf nodes are the variables needed to be summed (i.e., in our case  $(\mathbf{M}_{i,t})_t$ ), and each parent node maintains the (matrix) sum of its children. To make the partial sums private, a noise matrix is also added to each node in  $\mathcal{T}$ . We utilize additive Gaussian noise: at each node, we sample  $\widehat{\mathbf{N}} \in \mathbb{R}^{(d+1) \times (d+1)}$ , where each  $\widehat{\mathbf{N}}_{i,j} \sim \mathcal{N}(0, \sigma_N^2)$ , and  $\sigma_N^2 = 16m(L^2 + 1)^2 \log(2/\delta)^2 / \varepsilon^2$ , and symmetrize it (see step 6 of Alg. 2). The total noise  $\mathbf{H}_{i,t}$  is the sum of at most  $m$  such terms, hence the variance of each element in  $\mathbf{H}_{i,t}$  is  $\leq m\sigma_N^2$ . We can bound the operator norm of the top-left  $(d \times d)$ -submatrix of each noise term. With probability at least  $1 - \alpha/nM$ :

$$\|\mathbf{H}_{i,t}\|_{\text{op}} \leq \Lambda = \sqrt{32}m(L^2 + 1) \log(4/\delta)(4\sqrt{d} + 2 \ln(2nM/\alpha)) / \varepsilon.$$

However, as noted by [36], this  $\mathbf{H}_{i,t}$  would not always be PSD. To ensure that it is always PSD, we can shift each  $\mathbf{H}_{i,t}$  by  $2\Lambda \mathbf{I}$ , giving a bound on  $\rho_{\max}$ . Similarly, we can obtain bounds on  $\rho_{\min}$  and  $\kappa$ :

**Proposition 4.** *Fix  $\alpha > 0$ . If each agent  $i$  samples noise parameters  $\mathbf{H}_{i,t}$  and  $\mathbf{h}_{i,t}$  using the tree-based Gaussian mechanism mentioned above for all  $n$  trials of  $\bar{\sigma}$  in which communication occurs, then the following  $\rho_{\min}, \rho_{\max}$  and  $\kappa$  are  $(\alpha/2nM, \bar{\sigma})$ -accurate bounds:*

$$\rho_{\min} = \Lambda, \quad \rho_{\max} = 3\Lambda, \quad \kappa \leq \sqrt{m(L^2 + 1) \left( \sqrt{d} + 2 \log(2nM/\alpha) \right)} / (\sqrt{2}\varepsilon).$$

**Corollary 1** ( $(\varepsilon, \delta)$ -dependent Regret). *FEDUCB with the PRIVATIZER subroutine in Alg. 2, obtains  $\tilde{O}(d^{3/4} \sqrt{MT/\varepsilon})$  centralized regret and  $\tilde{O}(d^{3/4} \sqrt{(\bar{\chi}(\mathcal{G}_\gamma) \cdot \gamma) MT/\varepsilon})$  decentralized regret.*

## 5 Discussion and Conclusion

We make improvements along several fronts – while there has been prior work on multi-agent private linear bandits [18, 33] our work is the first to provide rigorous guarantees on private linear bandits in the multi-agent setting. Our work additionally provides the first algorithm with regret guarantees for contextual bandits in decentralized networks, extending the work of many on multi-armed bandits [34, 26, 25, 10, 11].

## References

- [1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 2312–2320, 2011.
- [2] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, pages 39–1, 2012.
- [3] Shipra Agrawal and Navin Goyal. Further optimal regret bounds for thompson sampling. In *Artificial Intelligence and Statistics*, pages 99–107, 2013.
- [4] Swapna Bucapatnam, Atilla Eryilmaz, and Ness B Shroff. Stochastic bandits with side observations on networks. In *The 2014 ACM international conference on Measurement and modeling of computer systems*, pages 289–300, 2014.
- [5] Joseph A Calandrino, Ann Kilzer, Arvind Narayanan, Edward W Felten, and Vitaly Shmatikov. "you might also like:" privacy risks of collaborative filtering. In *2011 IEEE symposium on security and privacy*, pages 231–246. IEEE, 2011.
- [6] Nicolo Cesa-Bianchi, Claudio Gentile, and Giovanni Zappella. A gang of bandits. In *Advances in Neural Information Processing Systems*, pages 737–745, 2013.
- [7] T-H Hubert Chan, Elaine Shi, and Dawn Song. Privacy-preserving stream aggregation with fault tolerance. In *International Conference on Financial Cryptography and Data Security*, pages 200–214. Springer, 2012.
- [8] TH Hubert Chan, Elaine Shi, and Dawn Song. Private and continual release of statistics. In *International Colloquium on Automata, Languages, and Programming*, pages 405–417. Springer, 2010.
- [9] Abhimanyu Dubey and Alex Pentland. Kernel methods for cooperative contextual bandits. *International Conference on Machine Learning*, 2020.
- [10] Abhimanyu Dubey and Alex Pentland. Private and byzantine-proof cooperative decision-making. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pages 357–365, 2020.
- [11] Abhimanyu Dubey and Alex Pentland. Robust algorithms for multiagent bandits with heavy tails. *International Conference on Machine Learning*, 2020.
- [12] Abhimanyu Dubey and Alex ‘Sandy’ Pentland. Thompson sampling on symmetric alpha-stable bandits. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5715–5721. International Joint Conferences on Artificial Intelligence Organization, 7 2019.
- [13] Cynthia Dwork. Differential privacy. *Encyclopedia of Cryptography and Security*, pages 338–340, 2011.
- [14] 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.
- [15] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.
- [16] Pierre Fraigniaud. Locality in distributed graph algorithms, 2016.
- [17] Robin C Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.
- [18] Awni Hannun, Brian Knott, Shubho Sengupta, and Laurens van der Maaten. Privacy-preserving contextual bandits. *arXiv preprint arXiv:1910.05299*, 2019.

- [19] Moritz Hardt and Guy N Rothblum. A multiplicative weights mechanism for privacy-preserving data analysis. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 61–70. IEEE, 2010.
- [20] Roger Iyengar, Joseph P Near, Dawn Song, Om Thakkar, Abhradeep Thakurta, and Lun Wang. Towards practical differentially private convex optimization. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 299–316. IEEE, 2019.
- [21] Prateek Jain, Pravesh Kothari, and Abhradeep Thakurta. Differentially private online learning. In *Conference on Learning Theory*, pages 24–1, 2012.
- [22] 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.
- [23] Jakub Konečný, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.
- [24] Nathan Korda, Balázs Szörényi, and Li Shuai. Distributed clustering of linear bandits in peer to peer networks. In *Journal of machine learning research workshop and conference proceedings*, volume 48, pages 1301–1309. International Machine Learning Society, 2016.
- [25] Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. Distributed cooperative decision-making in multiarmed bandits: Frequentist and bayesian algorithms. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 167–172. IEEE, 2016.
- [26] Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. On distributed cooperative decision-making in multiarmed bandits. In *2016 European Control Conference (ECC)*, pages 243–248. IEEE, 2016.
- [27] Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. Social imitation in cooperative multiarmed bandits: Partition-based algorithms with strictly local information. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 5239–5244. IEEE, 2018.
- [28] 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.
- [29] Nathan Linial. Locality in distributed graph algorithms. *SIAM Journal on computing*, 21(1):193–201, 1992.
- [30] Keqin Liu and Qing Zhao. Decentralized multi-armed bandit with multiple distributed players. In *2010 Information Theory and Applications Workshop (ITA)*, pages 1–10. IEEE, 2010.
- [31] Keqin Liu and Qing Zhao. Distributed learning in cognitive radio networks: Multi-armed bandit with distributed multiple players. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3010–3013. IEEE, 2010.
- [32] Keqin Liu and Qing Zhao. Distributed learning in multi-armed bandit with multiple players. *IEEE Transactions on Signal Processing*, 58(11):5667–5681, 2010.
- [33] Mohammad Malekzadeh, Dimitrios Athanasakis, Hamed Haddadi, and Ben Livshits. Privacy-preserving bandits. *arXiv preprint arXiv:1909.04421*, 2019.
- [34] David Martínez-Rubio, Varun Kanade, and Patrick Rebeschini. Decentralized cooperative stochastic multi-armed bandits. In *Advances in Neural Information Processing Systems*, 2019.
- [35] 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. AUAI Press, 2015.
- [36] Roshan Shariff and Or Sheffet. Differentially private contextual linear bandits. In *Advances in Neural Information Processing Systems*, pages 4296–4306, 2018.

- [37] Jukka Suomela. Survey of local algorithms. 45(2):24, 2013.
- [38] Abhradeep Guha Thakurta and Adam Smith. (nearly) optimal algorithms for private online learning in full-information and bandit settings. In *Advances in Neural Information Processing Systems*, pages 2733–2741, 2013.
- [39] Aristide CY Tossou and Christos Dimitrakakis. Differentially private, multi-agent multi-armed bandits. In *European Workshop on Reinforcement Learning (EWRL)*, 2015.
- [40] Aristide CY Tossou and Christos Dimitrakakis. Algorithms for differentially private multi-armed bandits. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.
- [41] Michal Valko, Nathaniel Korda, Rémi Munos, Ilias Flaounas, and Nelo Cristianini. Finite-time analysis of kernelised contextual bandits. *arXiv preprint arXiv:1309.6869*, 2013.
- [42] Yuanhao Wang, Jiachen Hu, Xiaoyu Chen, and Liwei Wang. Distributed bandit learning: How much communication is needed to achieve (near) optimal regret. *arXiv preprint arXiv:1904.06309*, 2019.
- [43] Yuanhao Wang, Jiachen Hu, Xiaoyu Chen, and Liwei Wang. Distributed bandit learning: Near-optimal regret with efficient communication. In *International Conference on Learning Representations*, 2020.