Differentially Private Federated Learning with Drift Control

Wei-Ting Chang Mohamed Seif Ravi Tandon
Department of Electrical and Computer Engineering
University of Arizona, Tucson, AZ, 85721
Email: {*wchang, mseif, tandonr*}@email.arizona.edu

2022 56th Annual Conference on Information Sciences and Systems (CISS) | 978-1-6654-1796-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/CISS53076.2022.9751200

***Abstract*—In this paper, we consider the problem of differentially private federated learning with statistical data heterogeneity. More specifically, users collaborate with the parameter server (PS) to jointly train a machine learning model using their local datasets that are non-i.i.d. across users. The PS is assumed to be honest-but-curious so that the data at users need to be kept private from the PS. More specifically, interactions between the PS and users must satisfy differential privacy (DP) for each user. In this work, we propose a differentially private mechanism that simultaneously deals with user-drift caused by non-i.i.d. data and the randomized user participation in the training process. Specifically, we study SCAFFOLD, a popular federated learning algorithm, that has shown better performance on dealing with non-i.i.d. data than previous federated averaging algorithms. We study the convergence rate of SCAFFOLD under differential privacy constraint. Our convergence results take into account time-varying perturbation noises used by the users, and data and user sampling. We propose two time-varying noise allocation schemes in order to achieve better convergence rate and satisfy a total DP privacy budget. We also conduct experiments to confirm our theoretical findings on real world dataset.**

***Index Terms*—Federated learning, Rényi Differential Privacy, Sampling, Stochastic Gradient Descent.**

I. Introduction

Federated learning (FL) [1] is a framework that enables multiple users to jointly train a machine learning model with the help of a parameter server (PS). In the training of FL, the PS interacts with multiple users to train a ML model in an iterative manner. Several variations of FL have been proposed, depending on the information exchanged between the PS and users. Specifically, there are two broad approaches to FL: (a) federated stochastic gradient descent (FedSGD), and (b) federated averaging (FedAvg) [1]. In FedSGD, users transmit the gradients computed using global model and local datasets back to the PS for gradient aggregation and global model updates. In FedAvg, users perform model updates locally and send the updated model back to the PS for model aggregation.

There are several motivating factors behind the surging popularity of FL: (a) centralized approaches can be inefficient in terms of storage/computation, and FL provides natural parallelization for training, and can leverage increasing computational power of devices and (b) local data at each user is never shared, but only gradient computations from each user are collected. Despite the fact that in FL, local data is never shared by a user, as shown in recent works [2]–[4], even exchanging gradients or models in a raw form can leak information. In addition, exchanging gradients or models incurs significant communication overhead in terms of the cost and latency, and it is often the bottleneck of the training process. Therefore, it is crucial to design training protocols that are both communication efficient and private. There are key challenges in federated optimization: (1) straggler problem where some users are slow in terms of their computation and communication capabilities, (2) statistical data heterogeneity across users. It has been shown that FedAvg performance degrades severely under non-iid data distribution. Specifically, the data heterogeneity introduces a drift in the local model updates from the global model. In order to tackle the problem of user drift, solutions such as FedProx [5] and SCAFFOLD [6] were proposed. In FedProx, the local loss function is modified by adding a penality term that penalizes the drift of the local model from the global one. However, one main drawback of this framework is that it fails to converge to a global optimum solution in contrast to the recent FL algorithm, SCAFFOLD [6]. In SCAFFOLD , the goal of the algorithm is to estimate the update direction for the server model, and estimate the update direction of each user. Furthermore, the difference between the two estimates is used to correct the local updates at the users.

There is a large body of recent work focusing on the design of differentially private FL (see a compherensive survery [7] and references therein). Differential privacy (DP) [8] has been adopted as a *de facto* standard notion for private data analysis and aggregation. Within the context of FL, the notion of local differential privacy (LDP) is more suitable in which a user can locally perturb and disclose the data (gradients/local models) to an *untrusted* data curator/aggregator. In the literature, there have been several research efforts to design FL algorithms satisfying LDP [9], [10], which require significant amount of perturbation noise to ensure privacy guarantees. However, the amount of noise can be reduced when employing user sampling [11], where users are sampled by the PS to participate in the training in each iteration. More specifically, users have the choice to decide whether or not to participate in the training process, and when to participate during the training process. It is worth noting there is a lack of understanding of the utility-priacy tradeff in FL under the non-iid setting. Recently, the authors of [12] have proposed DP FedProx under

This work has been supported in part by NSF Grants CAREER 1651492, CNS 1715947, CCF 2100013 and the 2018 Keysight Early Career Professor Award.

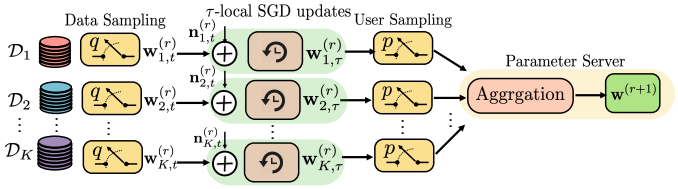

Fig. 1. Illustration of the private FL framework: Users collaborate with the PS to jointly train an ML model. In order to ensure some level of privacy, each user k adds a random noise $n_{k,t}^{(t)}$ to perturb the local model update $\mathbf{w}_{k,t}^{(r)}$ at each local iteration.

statistical data heterogeneity (i.e., non-iid) assumption, and analyzed the convergence rate of the FL algorithm under DP constraints. However, as we mentioned earlier the drawback of this framework is not achieving global optimality even without privacy constraints.

Main Contributions: The contribution of this paper is summarized as follows. We study the problem of differentially private federated learning with non-iid assumption. More specifically, we propose differentially private mechanisms for SCAFFOLD [1], a popular FL algorithm that performs reasonably well compared to FedAvg and its recent variation FedProx under data heterogenity scenarios. In our mechanism, we perform dynamic privacy budget allocation, where we reduce the amount of perturbation noise across communication rounds. We also take into account the impact of randomized users participation and data sampling on the central privacy and the convergence rate of the federated learning algorithm. We also present experimental results to show the advantage of our proposed mechanisms compared to conventional mechanisms which use fixed amount of noise.

II. System Model & Problem Statement

In a federated learning system, there are K users who jointly train a machine learning model $\mathbf{w} \in \mathbb{R}^d$ by minimizing the global loss function $F(\mathbf{w})$, i.e., $\mathbf{w}^* = \arg\min_{\mathbf{w}} F(\mathbf{w}) \triangleq \frac{1}{\sum_{k=1}^K D_k} \sum_{k=1}^K D_k f_k(\mathbf{w})$, where, instead of direct minimization of $F(\mathbf{w})$, each user k minimizes its local loss function $f_k(\mathbf{w}) = (1/D_k) \sum_{i=1}^{D_k} f_k(\mathbf{w}; \mathbf{u}_i^{(k)}, v_i^{(k)})$ using its local dataset $\mathcal{D}_k = \{(\mathbf{u}_i^{(k)}, v_i^{(k)})\}_{i=1}^{D_k}$, $|\mathcal{D}_k| = D_k$, and $\mathbf{u}_i^{(k)}$ is the i-th data point and $v_i^{(k)}$ is the corresponding label. Typically, $\mathcal{D}_k$'s are drawn from unknown probability distributions, hence, the data across users can potentially be non-i.i.d.

In practice, the minimization of $F(\mathbf{w})$ is done by using iterative gradient-based algorithms such as distributed stochastic gradient descent (SGD) algorithm. More specifically, in this work, we focus on FedAvg with drift control, i.e., SCAFFOLD [6]. We describe the training algorithm next (also see Algorithm 1). The algorithm consists of R (communication) rounds, where each round is comprised of a total of τ local iterations. At the beginning of the training, the PS initializes the global model $\mathbf{w}^{(0)}$ randomly, and initializes the global control variate $\mathbf{c}^{(0)}$ randomly or sets it to $\mathbf{0}$. Each user k initializes its local control variate to $\mathbf{c}_k^{(0)}$ similarly. In the r-th round, the PS samples a subset of users $\mathcal{S}^{(r)} \subseteq [1:K]$, $|\mathcal{S}^{(r)}| = S$, where we use $p = S/K$ to denote the fraction of sampled users. The PS then broadcasts the global parameter vector $\mathbf{w}^{(r-1)}$ to participating users, who then set the initial model for that round by $\mathbf{w}_{k,0}^{(r)} = \mathbf{w}^{(r-1)}$, $k \in \mathcal{S}^{(r)}$. Each user $k \in \mathcal{S}^{(r)}$ computes its local gradient using stochastic mini batch SGD for τ iterations, where the mini batch used is denoted by $\mathcal{B}_k \subseteq \mathcal{D}_k$, with size B_k (i.e., $|\mathcal{B}_k| = B_k$). For simplicity, we assume that all $D_k = D$ and $B_k = B$. Thus, the mini batch stochastic gradient estimate of user k at local iteration t within r-th round is as follows,

$$\mathbf{g}_k(\mathbf{w}_{k,t-1}^{(r)}) = \frac{1}{B} \sum_{i \in \mathcal{B}_k} \nabla f_k(\mathbf{w}_{t-1}; (\mathbf{u}_i^{(k)}, v_i^{(k)})), \qquad (1)$$

where the true gradient of user $k \in \mathcal{S}^{(r)}$ is defined as, $\nabla f_k(\mathbf{w}_{k,t-1}^{(r)}) = (1/D) \sum_{i=1}^{D} \nabla f_k(\mathbf{w}_{k,t-1}^{(r)}; (\mathbf{u}_i^{(k)}, v_i^{(k)}))$. At each local iteration t, the model is updated using the following update rule,

$$\mathbf{w}_{k,t}^{(r)} = \mathbf{w}_{k,t-1}^{(r)} - \eta_\ell (\mathbf{g}_k(\mathbf{w}_{k,t-1}^{(r)}) + \mathbf{n}_{k,t}^{(r)} - \mathbf{c}_k^{(r-1)} + \mathbf{c}^{(r-1)}),$$

where η_ℓ denotes the local learning rate, and $\mathbf{n}_{k,t}^{(r)}$ denotes the perturbation added for ensuring privacy with zero mean and variance $\mathbb{E}[\|\mathbf{n}_{k,t}^{(r)}\|^2] = d(\sigma_{k,t}^{(r)})^2$. The perturbed stochastic mini-batch gradient of user k at r-th round and t-th local iteration is denoted as $\tilde{\mathbf{g}}_k(\mathbf{w}_{k,t-1}^{(r)}) = \mathbf{g}_k(\mathbf{w}_{k,t-1}^{(r)}) + \mathbf{n}_{k,t}^{(r)}$. Once the participating user performs τ mini-batch SGD, user k updates its local control variate as follows,

$$\mathbf{c}_k^{(r)} = \mathbf{c}_k^{(r-1)} - \mathbf{c}^{(r-1)} + \frac{1}{\tau \eta_\ell} (\mathbf{w}_{k,0}^{(r)} - \mathbf{w}_{k,\tau}^{(r)}), \qquad (2)$$

and sends $\Delta \mathbf{w}_k^{(r)} = \mathbf{w}_{k,\tau}^{(r)} - \mathbf{w}_{k,0}^{(r)}$, and $\Delta \mathbf{c}_k^{(r)} = \mathbf{c}_k^{(r)} - \mathbf{c}_k^{(r-1)}$ back to the PS. The PS aggregates the differential model and local control variate updates by computing $\Delta \mathbf{w}^{(r)} = \frac{1}{|\mathcal{S}^{(r)}|} \sum_{k \in \mathcal{S}^{(r)}} \Delta \mathbf{w}_k^{(r)}$ and $\Delta \mathbf{c}^{(r)} = \frac{1}{|\mathcal{S}^{(r)}|} \sum_{k \in \mathcal{S}^{(r)}} \Delta \mathbf{c}_k^{(r)}$. Finally, the PS updates the global model and global control variate $\mathbf{w}^{(r)} = \mathbf{w}^{(r-1)} + \eta_g \Delta \mathbf{w}^{(r)}$ and $\mathbf{c}^{(r)} = \mathbf{c}^{(r-1)} + \frac{|\mathcal{S}^{(r)}|}{K} \Delta \mathbf{c}^{(r)}$, where η_g is the global learning rate. The training process continues until convergence or until a preset stopping criteria is met (such as when the privacy budget is exhausted). In this work, we assume that the local loss functions f_k's are L-smooth (hence, the gradients are locally L-Lipschitz). We also assume that the variance of the stochastic gradient of one data point is bounded by σ^2, and that the variance of mini-batch stochastic gradient is bounded by σ^2/B, i.e., $\mathbb{E}[\|\mathbf{g}_k(\mathbf{w}_{k,t}^{(r)}) - \nabla f_k(\mathbf{w}_{k,t-1}^{(r)})\|^2] \leq \sigma^2/B$.

In this work, we assume that the PS is honest but curious, where the PS follows the algorithm faithfully, but is interested in learning about users' data. We also assume that the final model $\mathbf{w}^{(R)}$ will be released to untrustworthy third party after the training is completed. Informally, an algorithm is considered to be differentially private when the outputs of the algorithm on two slightly different inputs (in terms of a

[1] Concurrently, the authors in [13] studied the problem of DP SCAFFOLD where the amount of noise is the same for every user and fixed across the training. In our work, our convergence results are general that take into account the amount of noise per user at each local iteration.

pre-defined distance) are indistinguishable. Formally, the LDP guarantee can be described as follows.

Definition 1. *((ϵ, δ)-LDP [8]) Let $\mathcal{D}_k$ be the local dataset of user k. For user k, a randomized mechanism $\mathcal{M}_k : \mathcal{D}_k \to \mathbb{R}^d$ is (ϵ, δ)-LDP if for any two neighboring datasets $\mathcal{D}_k, \mathcal{D}'_k$ that differ by at most one element, any any measurable subset $\mathcal{O}_k \subseteq Range(\mathcal{K}_k)$, we have*

$$\Pr(\mathcal{M}_k(\mathcal{D}_k) \in \mathcal{O}_k) \leq e^{\epsilon} \Pr(\mathcal{M}_k(\mathcal{D}'_k) \in \mathcal{O}_k) + \delta. \quad (3)$$

However, the standard LDP defined above is known to have a loose composition bound [8] when the data is accessed more than once within the algorithm. Hence, a relaxed alternative definition, Rényi differential privacy (RDP), that provides tighter composition bound (on standard LDP) was proposed.

Definition 2. *(Rényi Divergence) For two probability distributions P and Q, the Rényi of divergence of $\alpha > 1$ is*

$$D_\alpha(P||Q) \triangleq \frac{1}{\alpha - 1} \log \mathbb{E}_{x \sim Q} \left[\left(\frac{P(x)}{Q(x)} \right)^{\alpha} \right].$$

Definition 3. *((α, ϵ)-Rényi DP [14]) For user k, a random mechanism $\mathcal{M}_k : \mathcal{D}_k \to \mathbb{R}^d$ is (α, ϵ)-RDP if for any neighboring datasets $\mathcal{D}_k, \mathcal{D}'_k$, any any measurable subset $\mathcal{O}$ Range$(\mathcal{K}_k)$, we have*

$$D_\alpha(\Pr(\mathcal{M}_k(\mathcal{D}_k) \in \mathcal{O}_k) || \Pr(\mathcal{M}_k(\mathcal{D}'_k) \in \mathcal{O}_k)) \leq \epsilon.$$

Since FL optimization is an iterative algorithm, the priv guarantees degrades gracefully with the number of iterati It is worth noting that the Rényi DP based composition a T iterations gives a tighter bound for (ϵ', δ) DP guarantee. next present the privacy guarantees of T sequential mec nisms, each acting on the same dataset.

Definition 4. *(Composition of RDP) For a given α, composition of T mechanisms $\mathcal{M}_i$s, each satisfying $(\alpha$, RDP gives $(\alpha, \sum_{i=1}^{T} \epsilon_i)$-RDP.*

Definition 5. *(Conversion from RDP to DP [14]) If a pri mechanism $\mathcal{M}$ satisfies $(\alpha, \sum_{i=1}^{T} \epsilon_i)$-RDP, it also satisfies $\left(\sum_{i=1}^{T} \epsilon_i + \frac{\log(1/\delta))}{(\alpha-1)}, \delta\right)$-DP.*

Definition 6. *(Gaussian mechanism [14]) Suppose a releases a function $f(\mathcal{D}_k)$ of a local dataset $\mathcal{D}_k$ subjec (α, ϵ)-RDP. The Gaussian mechanism is defined as:*

$$\mathcal{M}_k(\mathcal{D}_k) \triangleq f(\mathcal{D}_k) + \mathcal{N}(0, \sigma^2 \mathbf{I}).$$

If the sensitivity of the function is bounded by Δf, i.e., $\|f(\mathcal{D}_k) - f(\mathcal{D}'_k)\| \leq \Delta f$, $\forall \mathcal{D}_k, \mathcal{D}'_k$, then for a given σ, the Gaussian mechanism satisfies (α, ϵ)-RDP, where $\epsilon = \alpha(\Delta f)^2 / 2\sigma^2$.

III. Main Results & Discussions

In this section, we present our results on DP SCAFFOLD. We first present our proposed scheme and show how we allocate perturbation noise across communication rounds. We next analyze the privacy leakage of the proposed algorithm, followed by convergence rates.

Algorithm 1 Differentially Private SCAFFOLD

Initialize $\mathbf{w}^{(0)}$, and $\mathbf{c}^{(0)} = \mathbf{0}$ at the PS;
Initialize $\mathbf{c}_k^{(0)} = \mathbf{0}$ at all user k;
for round $r = 1, ..., R$ **do**
  sample users $\mathcal{S}^{(r)} \subseteq [K]$
  PS sends $\mathbf{w}^{(r-1)}$ and $\mathbf{c}^{(r-1)}$ to user $k \in \mathcal{S}^{(r)}$;
  for each user $k \in \mathcal{S}^{(r)}$ in parallel **do**
    initialize $\mathbf{w}_{k,0}^{(r)} = \mathbf{w}^{(r-1)}$;
    for local iteration $t = 1, ..., \tau$ **do**
      Compute $\mathbf{g}_k(\mathbf{w}_{k,t-1}^{(r)})$ according to;
      $\mathbf{w}_{k,t}^{(r)} = \mathbf{w}_{k,t-1}^{(r)} - \eta_\ell(\mathbf{g}_k(\mathbf{w}_{k,t-1}^{(r)}) + \mathbf{n}_{k,t}^{(r)} - \mathbf{c}_k^{(r-1)} + \mathbf{c}^{(r-1)})$
    end for
    $\mathbf{c}_k^{(r)} = \mathbf{c}_k^{(r-1)} - \mathbf{c}^{(r-1)} + \frac{1}{\tau\eta_\ell}(\mathbf{w}_{k,0}^{(r)} - \mathbf{w}_{k,\tau}^{(r)})$
    Send $(\Delta\mathbf{w}_k^{(r)}, \Delta\mathbf{c}_k^{(r)}) = (\mathbf{w}_{k,\tau}^{(r)} - \mathbf{w}_{k,0}^{(r)}, \mathbf{c}_k^{(r)} - \mathbf{c}_k^{(r-1)})$
  end for

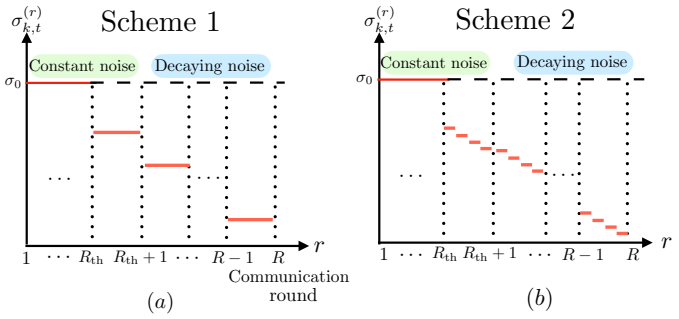

Fig. 2. Illustration of noise decaying Gaussian mechanisms where the amount of noise decays across communications rounds: (a) Scheme 1, time varying noise across communication rounds but fixed noise across local iterations, and (b) Scheme 2, time varying noise across local iterations.

Dynamic privacy budget allocation: Given a central privacy target, our goal is to perform privacy budget allocation to optimize the the model accuracy. Furthermore, we use the composition property of Rényi DP to adaptively allocate the amount of noise across communication rounds. We next present the privacy analysis of the proposed scheme followed by convergence rate analysis for convex functions.

At each round r, S sampled users participate each with probability p in the training of the global model. Each user k performs mini-batch SGD for τ local iterations, where for each local iteration t, B points of the local dataset are sampled independently with probability q.

Scheme 1: In this scheme, users perturb their local gradients with the same amount of noise across the local iterations and across communication rounds till round R_{th}. At the beginning

of round $R_{\text{th}}+1$, users reduces their amount of noise as follows:

$$(\sigma_{k,t}^{(r)})^2 = \begin{cases} \sigma_0^2, & 1 \le r \le R_{\text{th}}, \\ \beta^{r-R_{\text{th}}}\sigma_0^2, & R_{\text{th}} < r \le R, \end{cases} \tag{6}$$

where $\beta \in (0,1]$. Note that in this scheme, the same amount of perturbation noise is added across local iterations.

Scheme 2: In this scheme, users perturb their local gradients with the same amount of noise across the local iterations and across communication rounds till round R_{th}. At the beginning of round $R_{\text{th}}+1$, users reduces their amount of noise as follows:

$$(\sigma_{k,t}^{(r)})^2 = \begin{cases} \sigma_0^2, & 1 \le r \le R_{\text{th}}, \\ \beta^{\tau(r-R_{\text{th}}-1)+t-1}\sigma_0^2, & 1 \le t \le \tau, R_{\text{th}} < r \le R, \end{cases} \tag{7}$$

where $\beta \in (0,1]$. The main difference here is that the amount of perturbation noise is reduced across local iterations unlike Scheme 1. We next present the total central privacy leakage of DP SCAFFOLD. The intuition behind reducing noise is that as the training process continues, we expect the gradients to "shrink" (i.e., converge to zero) and thus it may be sufficient to add lower amount of noise in the later part of the training.

A. Privacy Analysis

We use the composition property of RDP (Definition 4) to analyze our proposed FL algorithms. At each communication round r, each user k performs mini-batch SGD for τ local iterations. At each local iteration, the mini-batches are randomly sampled (without replacement) from the local datasets, which which in turn amplifies the privacy level [15]. It is worth noting that the privacy amplification results by sub-sampling were developed for the centralized setting. In order to use these results, the data points must be sampled independently at random with probability pq for the distributed setting. After computing the local gradient $\mathbf{g}_k(\mathbf{w}_{k,t-1}^{(r)})$, each user injects a random Gaussian noise $\mathbf{n}_{k,t}^{(r)}$ for privacy. For the scope of this paper, we assume that the users perturbs their local gradients with amount of noise at each local iteration t, i.e., $\sigma_{k,t}^{(r)} = \sigma_t^{(r)}$. We summarize our proposed scheme in Algorithm 1. We next analyze the privacy leakage of each scheme as follows.

Theorem 1. *Scheme* 1 *satisfies* (α, ϵ_c)*-RDP after* R *communication rounds, where*

$$\epsilon_c^{(1)} = \frac{24p^2q^2\alpha L^2\tau}{B^2\sigma_0^2} \times \left[R_{th} + \frac{1-(1/\beta)^{R-R_{th}+1}}{1-(1/\beta)} - 1\right],$$

where $\beta \in [(5/\sigma_0^2)^{\frac{1}{R-R_{th}}}, 1]$, $\sigma_0 \ge \sqrt{5}$, $pq \le 0.1$, L *is the Lipschitz constant, and* $\alpha \le \beta^{R-R_{th}}\sigma_0^2 \log(1/pq)$.

Theorem 2. *Scheme* 2 *satisfies* (α, ϵ_c)*-RDP after* R *communication rounds, where*

$$\epsilon_c^{(2)} = \frac{24p^2q^2\alpha L^2\tau}{B^2\sigma_0^2} \times \left[R_{th} + \frac{1}{\tau} \times \frac{1-(1/\beta)^{\tau(R-R_{th})}}{1-(1/\beta)}\right],$$

where $\beta \in [(5/\sigma_0^2)^{1/(\tau(R-R_{th})-1)}, 1]$, $\sigma_0 \ge \sqrt{5}$ *and* $pq \le 0.1$, *and* $\alpha \le \beta^{\tau(R-R_{th})-1}\sigma_0^2 \log(1/pq)$.

The proofs of these theorems are presented in Appendix A of the full version of this paper [16]. From the above expressions we can observe that the central privacy leakage recovers the case when each user perturbs with the same amount of noise across time, i.e., σ_0^2. For a given σ_0^2, we can observe that Scheme 1 leaks less compared to Scheme 2, the reason is that we in Scheme 2 we further keep reducing the amount of noise across local iterations which results in privacy degradation unlike Scheme 1 where the amount of noise is fixed across local iterations.

B. Convergence Analysis

We next present the general convergence result for the case when the local loss function f_k's are convex and L-smooth. We then tailor the bound to specific noise allocation schemes, and show the impact of noise decaying threshold R_{th}, starting noise variance $(\sigma_{k,0}^{(0)})^2$, and the noise decaying factor β.

Theorem 3. *Suppose the local loss functions,* f_k*'s, are convex and* L*-smooth, then for any* $S = pK$, $B = qD$ *where* $p, q \in (0,1]$*, a number of communication round* R *and local iteration* τ*, and any effective learning rate* $\tilde{\eta} = \tau\eta_g\eta_\ell$*, where* $\eta_g \ge 1$ *and* $\eta_\ell \le \frac{1}{81L\tau\eta_g}$*, we have,*

$$\mathbb{E}[f(\bar{\mathbf{w}}^{(R)})] - f(\mathbf{w}^*) \le \underbrace{\frac{16\tilde{\eta}}{\tau SR}\sum_{r=1}^{R}\sum_{k,t}\left(\tilde{\sigma}_{k,t}^{(r)}\right)^2}_{\text{1:Impact of DP}}$$
$$\underbrace{+\frac{\tilde{\eta}}{25\eta_g^2 S\tau^3 R}\sum_{r=1}^{R}\sum_{t}\sum_{i=0}^{t-1}\left(1+\frac{1}{\tau-1}\right)^i\sum_{k}\left(\tilde{\sigma}_{k,t-i}^{(r)}\right)^2}_{\text{1:Impact of DP}}$$
$$\underbrace{+\frac{1}{\tilde{\eta}R}\|\mathbf{w}^{(0)}-\mathbf{w}^*\|^2 + \frac{11\tilde{\eta}K\mathcal{C}_0}{SR} + \frac{15\tilde{\eta}}{\tau SB}\left(1+\frac{\tau}{\eta_g^2}\right)\sigma^2}_{\text{2:Standard Convergence}}, \tag{8}$$

where $\mathcal{C}_0 = (1/K)\sum_k \mathbb{E}[\|\mathbb{E}[\mathbf{c}_k^{(0)}] - \nabla f_k(\mathbf{w}^*)\|^2]$ *and* $\mathbb{E}[f(\bar{\mathbf{w}}^{(R)})] = (1/R)\sum_r \mathbb{E}[f(\mathbf{w}^{(r-1)})]$

The proof of Theorem 3 can be found in Appendix B of [16]. Our proof is adapted from [6], where the key idea of the proof in [6] is to keep track of various sources of error. In the original SCAFFOLD, there are two main sources of error: a) user-drift and b) control-lag. User-drift occurs due to non-i.i.d. data used in the training across users, and the fact that local loss functions are not the same function. Whereas, the control-lag comes from the fact that not every user updates their local control variate at every round by design. The main distinction from our proof to the one in [6] is that the noise used for privacy needs to be taken into account. Since the gradients are perturbed, the control variates are also perturbed. Hence, the noise not only shows up when we bound the user-drift, but also appears when we bound the variance of the global and local control variates.

The bound in (8) is consists of two parts. The first part comes from the standard analysis of convergence rate. The second part shows the impact of making the algorithm differentially

private. At first glance, the right-hand side does not appear to approach zero as R goes to infinity. However, one can carefully choose effective learning rate $\tilde{\eta}$ to ensure convergence. One can readily check that if all noise variance $(\tilde{\sigma}_{k,t}^{(r)})^2 = 0,\ \forall k,t,r$, only the standard convergence part remains, and the bound of original SCAFFOLD can be recovered with a small difference in the constant coefficient by choosing appropriate learning rates. By using Lagrange Multiplier method, we can obtain the optimal effective learning rate, i.e.,

$$\tilde{\eta}^* = \frac{\|\mathbf{w}^{(0)} - \mathbf{w}^*\|}{\sqrt{R}} \Bigg(\frac{16}{\tau SR} \sum_{r=1}^{R} \sum_{k,t} \left(\tilde{\sigma}_{k,t}^{(r)}\right)^2 + \frac{1}{25\eta_g^2 S\tau^3 R} \sum_{r=1}^{R} \sum_{t} \sum_{i=0}^{t-1} \left(1 + \frac{1}{\tau - 1}\right)^i \sum_{k} \left(\tilde{\sigma}_{k,t-i}^{(r)}\right)^2 + \frac{11K\mathcal{C}_0}{SR} + \frac{15}{\tau SB}\left(1 + \frac{\tau}{\eta_g^2}\right)\sigma^2 \Bigg)^{\frac{-1}{2}} \tag{9}$$

Therefore, the bound in (8) becomes,

$$\mathbb{E}[f(\bar{\mathbf{w}}^{(R)})] - f(\mathbf{w}^*) \leq \frac{2\|\mathbf{w}^{(0)} - \mathbf{w}^*\|}{\sqrt{R}} \Bigg(\frac{16}{\tau SR} \sum_{r=1}^{R} \sum_{k,t} \left(\tilde{\sigma}_{k,t}^{(r)}\right)^2 + \frac{1}{25\eta_g^2 S\tau^3 R} \sum_{r=1}^{R} \sum_{t} \sum_{i=0}^{t-1} \left(1 + \frac{1}{\tau - 1}\right)^i \sum_{k} \left(\tilde{\sigma}_{k,t-i}^{(r)}\right)^2 + \frac{11K\mathcal{C}_0}{SR} + \frac{15}{\tau SB}\left(1 + \frac{\tau}{\eta_g^2}\right)\sigma^2 \Bigg)^{\frac{1}{2}} \tag{10}$$

Now we tailor the result to specific noise allocation schemes.

Corollary 1. *Under the same assumptions as in Theorem 3, for the case when constant noise is added across all rounds, local iterations and users, i.e.,* $\left(\tilde{\sigma}_{k,t}^{(r)}\right)^2 = \sigma_0^2$*, the convergence rate can be further upper bounded and simplified to the following,*

$$\mathbb{E}[f(\bar{\mathbf{w}}^{(R)})] - f(\mathbf{w}^*) \leq \frac{2\|\mathbf{w}^{(0)} - \mathbf{w}^*\|}{\sqrt{R}} \Bigg(\frac{11K\mathcal{C}_0}{SR} + \frac{15}{\tau SB}\left(1 + \frac{\tau}{\eta_g^2}\right)\sigma^2 + \frac{K}{S}\left(16 + \frac{3}{25\eta_g^2\tau}\right)\sigma_0^2 \Bigg)^{\frac{1}{2}} \tag{11}$$

Corollary 2. *Under the same assumptions as in Theorem 3, by applying Scheme 1, where perturbation noise is allocated according to* (6)*, the convergence rate can be expressed as,*

$$\mathbb{E}[f(\bar{\mathbf{w}}^{(R)})] - f(\mathbf{w}^*) \leq \frac{2\|\mathbf{w}^{(0)} - \mathbf{w}^*\|}{\sqrt{R}} \Bigg(\frac{11K\mathcal{C}_0}{SR} + \frac{15}{\tau SB}\left(1 + \frac{\tau}{\eta_g^2}\right)\sigma^2 + \frac{K}{SR}\left(R_{th} + \frac{1 - \beta^{R - R_{th} + 1}}{1 - \beta} - 1\right)\left(16 + \frac{3}{25\eta_g^2\tau}\right)\sigma_0^2 \Bigg)^{\frac{1}{2}} \tag{12}$$

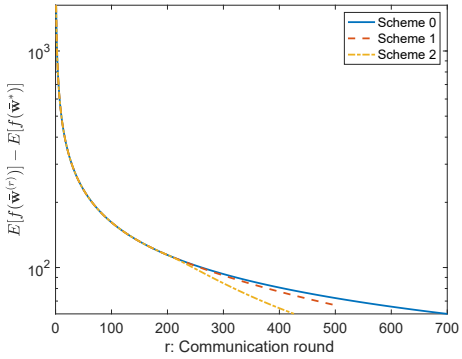

Fig. 3. Comparisons of convergence bounds for different schemes, where $\sigma_0^2 = 8$, $R_{\text{th}} = 200$, $R = 700$, $\|\mathbf{w}^{(0)} - \mathbf{w}^*\| = 1$, $\mathcal{C}_0 = 1$, $D = 2500$, $B = 500$, $\eta_g = 2$, $\tau = 200$, $S = 8$, $K = 40$, $L = 2$, $\beta^{(1)} = 0.998$ and $\beta^{(2)} = 0.99998$.

Corollary 3. *Under the same assumptions as in Theorem 3, by applying Scheme 2, where perturbation noise is allocated according to* (7)*, the convergence rate can be expressed as,*

$$\mathbb{E}[f(\bar{\mathbf{w}}^{(R)})] - f(\mathbf{w}^*) \leq \frac{2\|\mathbf{w}^{(0)} - \mathbf{w}^*\|}{\sqrt{R}} \Bigg(\frac{11K\mathcal{C}_0}{SR} + \frac{15}{\tau SB}\left(1 + \frac{\tau}{\eta_g^2}\right)\sigma^2 + \frac{R_{th}}{R}\frac{K}{S}\left(16 + \frac{3}{25\eta_g^2\tau}\right)\sigma_0^2 + \frac{K}{\tau SR}\left(16 + \frac{3}{25\eta_g^2}\right)\left(\frac{1 - \beta^{\tau(R - R_{th})}}{1 - \beta}\right)\sigma_0^2 \Bigg)^{\frac{1}{2}} \tag{13}$$

Clearly, all three bounds behave as $O(1/\sqrt{R})$ asymptotically and converge as $R \to \infty$. However, the effects of data sampling, user sampling, perturbation noise and R_{th} start to show when R is finite. By selecting, $R_{\text{th}} = R$, both (12) and (13) revert back to (11). Next, we can see that user sampling ($p = S/K$) affects all the terms within (11), (12) and (13). To achieve faster convergence, p should be as large as possible. Intuitively, larger p means more information about local data is provided for training, which naturally leads to larger privacy leakage. Similarly, data sampling provides the same effect but to only the term with the bound on the variance of gradient σ^2. For fixed parameters, by directly comparing the rates in (12) and (13), we can see that (13) has faster convergence. It is also clear that both (12) and (13) converge the fastest when selecting $R_{\text{th}} = 1$ (and fixing all other parameters).

In Fig. 3, we numerically compare the proposed schemes in terms of their convergence rate along with the baseline scheme (Scheme 0 in Fig. 3) where the amount of perturbation of noise is the same across users and time. All the schemes satisfy total privacy budget $(11.5147, 10^{-5})$-DP. From the figure, we can see the advantage of the noise decaying schemes in terms of convergence. However, scheme 1 and 2 exhausted the privacy budget at around $R = 500$ and 400, respectively. While scheme 0 can potentially continue training before running out of privacy budget, it may take significantly longer time for the model to converge.

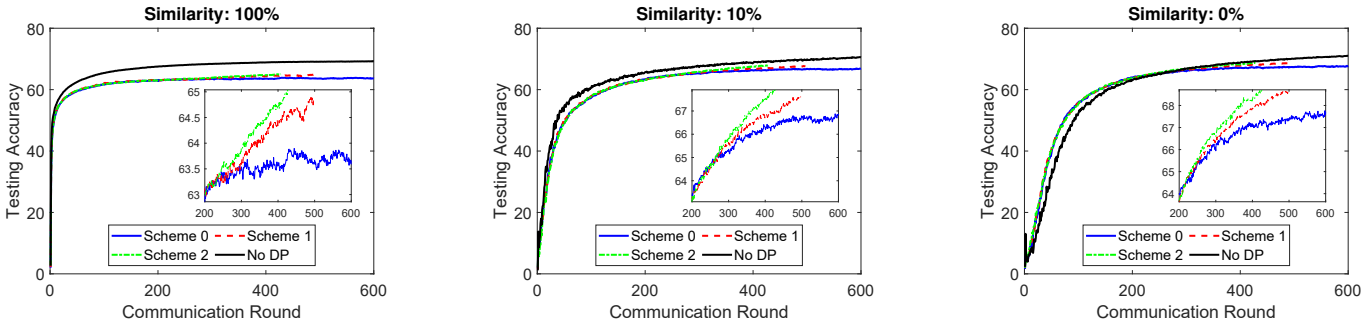

Fig. 4. Comparison between the proposed noise allocation schemes, where all schemes satisfy $(7.6769, 10^{-5})$-DP. Scheme 2 is shown to have the best accuracy with the least amount of communication round needed among all private schemes.

IV. Experiments

In this section, we evaluate the performance of the proposed DP SCAFFOLD schemes through experiments. We consider image classification task on EMNIST-balanced dataset using logistic regression with negative log-likelihood loss. The EMNIST-balanced dataset consists of $131,600$ handwritten digits and letters that can be classified into 47 classes. For a dataset with $\zeta\%$-similarity, $\zeta\%$ of the local dataset at each user is filled with data drawn in i.i.d. fashion, and the rest of the local dataset is filled using the sort-and-partition method in [17]. This simulates a non-i.i.d. data distribution across the users. We assume there are $K = 40$ users, and set user sampling probability as $p = 0.2$. A total of 2500 data points are allocated to each users, and each mini-batch is of size 500, hence, $q = 0.2$. The rest of the data is used as testing data. The training continues until the privacy budget is exhausted by each scheme, therefore, the number of total rounds R depends on the noise allocation scheme and the parameters used. There are a total of $\tau = 50$ local iterations within each round. We assume all three noise allocation schemes start with $\sigma_0^2 = 8$, and $\beta = 0.998$ and 0.99992 for scheme 1 and 2, respectively. Scheme 1 and 2 start decaying at $R_{\text{th}} = 200$. The corresponding privacy leakage after $R = 600$ rounds are $(7.6769, 10^{-5})$-DP for scheme 0. We then use this as the privacy budget for scheme 1 and 2. As a result, scheme 1 and 2 exhausted the privacy budget at Round 497 and 428, respectively. From Fig. 4, both scheme 1 and 2 at their stopping round already outperform scheme 0 at Round 600 for all three data similarity levels considered. Therefore, with the same privacy budget, it is beneficial to reduce noise added, which saves upto 28% of computation and communication.

V. Conclusion

In this paper, we proposed DP mechanism for SCAFFOLD under statistical data heterogeneity scenario. Specifically, we introduced the idea of using time-varying noise to SCAFFOLD based on the fact that gradients become smaller as training progresses. We derived convergence and privacy leakage results, and showed that the dynamic noise allocation schemes achieve faster convergence. Through experiments, we showed that, under the same privacy budget, the proposed schemes require less computation and communication to achieve higher testing accuracy. An interesting direction is to see what benefits we can gain in the medium and low privacy regimes.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*, 2017, pp. 1273–1282.

[2] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE Symposium on Security and Privacy (S & P)*, May 2017, pp. 3–18.

[3] J. Hayes, L. Melis, G. Danezis, and E. De Cristofaro, "LOGAN: Membership inference attacks against generative models," *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 1, pp. 133–152, 2019.

[4] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE Symposium on Security and Privacy (S&P)*, May 2019, pp. 691–706.

[5] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *arXiv preprint arXiv:1812.06127*, 2018.

[6] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *International Conference on Machine Learning (ICML)*. PMLR, 2020, pp. 5132–5143.

[7] S. Ulukus, S. Avestimehr, M. Gastpar, S. Jafar, R. Tandon, and C. Tian, "Private retrieval, computing and learning: Recent progress and future challenges," *arXiv preprint arXiv:2108.00026*, 2021.

[8] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.

[9] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," *arXiv preprint arXiv:1712.07557*, 2017.

[10] A. Lowy and M. Razaviyayn, "Locally differentially private federated learning: Efficient algorithms with tight risk bounds," *arXiv preprint arXiv:2106.09779*, 2021.

[11] B. Balle, G. Barthe, and M. Gaboardi, "Privacy amplification by subsampling: Tight analyses via couplings and divergences," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, pp. 6277–6287, 2018.

[12] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, and H. V. Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3454–3469, 2020.

[13] M. Noble, A. Bellet, and A. Dieuleveut, "Differentially private federated learning on heterogeneous data," *arXiv preprint arXiv:2111.09278*, 2021.

[14] I. Mironov, "Rényi differential privacy," in *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, 2017, pp. 263–275.

[15] M. Bun, C. Dwork, G. N. Rothblum, and T. Steinke, "Composable and versatile privacy via truncated CDP," in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, 2018, pp. 74–86.

[16] W.-T. Chang, M. Seif, and R. Tandon, "Differentially Private Federated Learning with Drift Control," 2021. [Online]. Available: https://www.dropbox.com/s/qhkow2thd90ks14/CISS2022_FL.pdf?dl=0

[17] T. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," *arXiv preprint arXiv:1909.06335*, 2019.