Time-Varying Noise Perturbation and Power Control for Differential-Privacy-Preserving Wireless Federated Learning

Dang Qua Nguyen and Taejoon Kim
Department of Electrical Engineering and Computer Science
University of Kansas, Lawrence, KS 66045 USA
Email: {quand, taejoonkim}@ku.edu

***Abstract*—Wireless Federated Learning (FL) is a framework that enables a server to collaboratively train a learning model with distributed users via wireless channels without sharing users' training data. However, due to the engineering-inversion attack, users' data privacy leakage during the training process is a crucial concern. Differential privacy (DP) techniques are commonly applied to deal with this issue. This approach, however, can cause degradation in the learning utility. This paper proposes a privacy-preserving differentially private FL algorithm that applies time-varying noise variance perturbation. Taking advantage of the existing wireless channel noises, we jointly design DP noise variances and users' transmit power to address privacy and learning utility tradeoffs in the wireless FL. In addition, the number of FL iterations is optimized by minimizing the upper bound on the learning error. We conduct simulations to demonstrate the effectiveness of our approach in terms of DP guarantee and learning utility.**

***Index Terms*—Federated learning, differential privacy, time-varying noise variance, noise perturbation and power control, convergence analysis.**

I. Introduction

Federated learning (FL) is an emerging decentralized machine learning (ML) framework that allows multiple distributed users to collaboratively train a learning model [1]–[5]. It operates through a series of iterations with a central server distributing a global learning model and each user training the global model on its local data to obtain a local model update. Unlike conventional centralized learning framework that requires uploading an extensive amount of data to the central server, only local model updates are uploaded from the distributed users to the central server [1]. FL has been thought of as an efficient ML approach in terms of reducing communication overhead while providing data privacy [1].

Recent findings, however, have shown that model updates in FL can lead to severe privacy breaches [6], [7], motivating recent research endeavors [8], [9] in privacy-preserving wireless FL approaches. A common idea is adding artificial noises to the local model updates prior to sending them to the central server [9]–[12]. The artificially-added DP noise can ensure that no specific local data point can be reverse-engineered from the model updates, thus safeguarding the privacy of users's data. In particular, there is a trade-off between privacy guarantee and learning accuracy performance that depends on the artificial DP noise variances. The larger the DP noise variances are, the higher the privacy guarantee is, but they lower the learning utility.

To balance the trade-off between learning accuracy and privacy guarantee, a time-varying noise perturbation mechanism is proposed in [13]. Compared to the conventional differentially private FL approaches [9], [10], [12] that apply the same noise variance, the DP noise variance in [13] is adaptively changed at every FL iteration. As the FL proceeds, smaller DP noise variances are applied to maintain a DP guarantee without sacrificing learning performance. The approach in [13] made the ideal assumption that the local model updates are transmitted through noiseless communication channels to the central server.

Unlike [9]–[13], noisy wireless channels between the central server and distributed users were considered in [14]–[16]. The common methods in [14]–[16] are decreasing the signal-to-noise ratio (SNR) of local model update links, which have the equivalent effect as increasing the DP noise variances. This method, however, impairs transmission quality in model update links, leading to learning accuracy degradation.

This paper proposes a differentially private wireless FL method that applies time-varying noise perturbation to strike a tradeoff between privacy and communication quality. We consider the wireless channel noise in our proposed algorithm. We characterize its DP performance guarantee by jointly designing the DP noise variances and the users' transmit power, while maintaining the minimum SNR performances in the local model update links. We analyze the convergence of the proposed algorithm and derive an upper bound of the expected learning error. We optimize the number of FL iterations by minimizing the derived upper bound. Simulations are conducted to demonstrate the benefits of the proposed method.

The work of T. Kim was supported in part by the National Science Foundation (NSF) under Grant CNS2212565, CNS2225577, ITE2226447, the Office of Naval Research (ONR) under Grant N000142112472, and the NSF and Office of the Under Secretary of Defense (OUSD) – Research and Engineering, ITE2326898, as part of the NSF Convergence Accelerator Track G: Securely Operating Through 5G Infrastructure Program.

Notation: A bold lowercase letter $\mathbf{a}$ is a column vector. The $\mathbf{a}^T$, $\mathbf{a}(i)$, and $\|\mathbf{a}\|$ are, respectively, the transpose, ith entry, and 2-norm of $\mathbf{a}$. $|\mathcal{A}|$ denotes the cardinality of set $\mathcal{A}$. $\mathbf{I}$ denotes an identity matrix with appropriate dimensions. For $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{n\times 1}$, $\langle \mathbf{x}, \mathbf{y}\rangle$ denotes their inner product. $\mathbb{E}[\cdot]$ denotes the expectation operator.

II. System Model and Algorithm

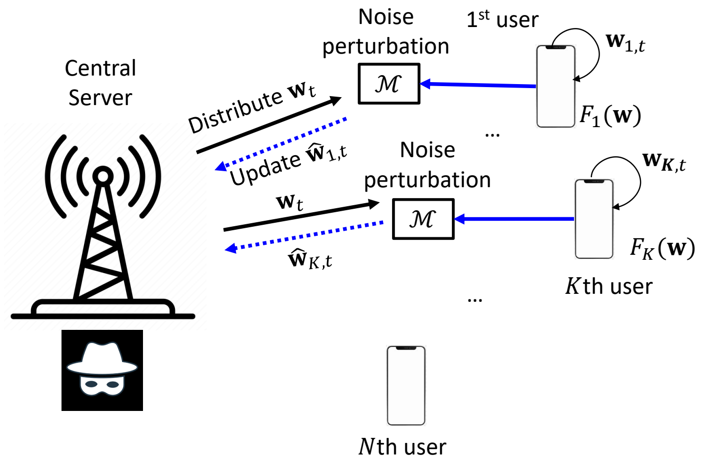

Fig. 1: Wireless differentially private FL system

We consider a wireless FL system with a central server and N distributed users, as illustrated in Fig 1, to collaboratively train a global learning model $\mathbf{w}^\star \in \mathbb{R}^{d\times 1}$ via wireless channels through T FL iterations. The $\mathbf{w}^\star$ is defined by

$$\mathbf{w}^\star = \underset{\mathbf{w}}{\operatorname{argmin}}\, F(\mathbf{w}), \tag{1}$$

where $F(\mathbf{w}) = \sum_{n=1}^{N} F_n(\mathbf{w})$ is the global loss function and $F_n(\mathbf{w})$ is the local loss function at the nth user, for $n \in \mathcal{N} = \{1, 2, \ldots, N\}$. The nth user utilizes its data set $\mathcal{D}_n$ to locally train the global model in each FL iteration, where $|\mathcal{D}_n| = D$, for $n \in \mathcal{N}$.

We assume that the server in Fig. 1 is exposed to a malicious entity, who may attempt to obtain the users's local data points in $\{\mathcal{D}_n\}$. To cope with the privacy threat due to the malicious entity at the server, we propose a differentially private FL algorithm, which is described in Algorithm 1. In Step 1, the central server randomly initializes a global model $\mathbf{w}_0$ when $t = 0$ to begin T FL iterations (Step 2). In Step 3, the server sends the global model $\mathbf{w}_t \in \mathbb{R}^{d\times 1}$, as shown in Fig. 1, to a random subset of users $\mathcal{K} \subseteq \mathcal{N}$, where $|\mathcal{K}| = K$ and $\mathcal{K}$ is uniformly realized, for $t = 1, 2, \cdots, T$. Next, in Step 5, the kth user updates the received model through E iterations of local stochastic gradient descent (localSGD) updates with data set $\mathcal{D}_k$ to attain $\mathbf{w}_{k,t}^{(E)}$, for $k \in \mathcal{K}$. Specifically, $\mathbf{w}_{k,t}^{(e)} = \mathbf{w}_{k,t}^{(e-1)} - \eta_t \widetilde{\nabla} F_k(\mathbf{w}_{k,t}^{(e-1)})$, for $e = 1, 2, \ldots, E$, where $\widetilde{\nabla} F_k(\cdot)$ denotes the stochastic gradient of $F_k(\cdot)$, η_t is the learning rate at the tth FL iteration, and $\mathbf{w}_{k,t}^{(0)} = \mathbf{w}_t$. Unlike the previous works [10], [13] that apply the same learning rate for all T FL iterations, the learning rate $\{\eta_t\}$ are adaptively changed in this work to accelerate the convergence of Algorithm 1. In Steps 6 and 7, the model difference $\mathbf{v}_{k,t} = \mathbf{w}_{k,t}^{(E)} - \mathbf{w}_t$ is computed and clipped by a constant C to obtain $\mathbf{u}_{k,t} = c_{k,t}\mathbf{v}_{k,t}$ so that $\|\mathbf{u}_{k,t}\| \le C$, where $c_{k,t} = \min\left\{1, \frac{C}{\|\mathbf{v}_{k,t}\|}\right\}$ [11], [13]. Herein, the clipping operator facilitates the training stability and DP privacy guarantee analysis [17]. Then, the a DP Gaussian noise $\mathbf{z}_t$ is added to $\sqrt{\rho}\mathbf{u}_{k,t}$ to get $\widehat{\mathbf{w}}_{k,t} = \sqrt{\rho}\mathbf{u}_{k,t} + \mathbf{z}_t$, as presented in Step 8, where ρ is the transmit power for the model update link and $\mathbf{z}_t$ has zero-mean and covariance $\sigma_t^2\mathbf{I}$. In Step 10, the perturbed model differences $\{\widehat{\mathbf{w}}_{k,t}\}$ are transferred to the server via noisy wireless channels to create the $(t+1)$th global model $\mathbf{w}_{t+1} = \mathbf{w}_t + \frac{1}{K}\sum_{k\in\mathcal{K}}(\widehat{\mathbf{w}}_{k,t} + \mathbf{n}_t)$, where $\mathbf{n}_t$ is the additive Gaussian noise in the channel with zero-mean and covariance $\xi_t^2\mathbf{I}$. To ensure an (ϵ, δ)-DP guarantee and communication quality in the model update links, the DP noise variance $\{\sigma_t^2\}$ and transmit power ρ are jointly designed in Section III.

Algorithm 1 Differentially private FL with time-varying noise variances

```
Require: $K$, $\{\eta_t\}$, $T$, $\epsilon$, $\delta$, and $C$
Output: $\mathbf{w}_T$
 1: Initialization: $\mathbf{w}_0$ and $t = 0$
 2: while $t < T$ do
 3:     Send $\mathbf{w}_t$ to users in a random subset of users $\mathcal{K} \subseteq \mathcal{N}$
 4:     while $k \in \mathcal{K}$ do
 5:         Local model update: $\mathbf{w}_{k,t}^{(E)} = \text{localSGD}(\mathbf{w}_t, \eta_t, E)$
 6:         Compute model difference: $\mathbf{v}_{k,t} = \mathbf{w}_{k,t}^{(E)} - \mathbf{w}_t$
 7:         Clipping: $\mathbf{u}_{k,t} = c_{k,t}\mathbf{v}_{k,t}$
 8:         Adding DP noise and update: $\widehat{\mathbf{w}}_{k,t} = \sqrt{\rho}\mathbf{u}_{k,t} + \mathbf{z}_t$
 9:     end while
10:     Model aggregation: $\mathbf{w}_{t+1} = \mathbf{w}_t + \frac{1}{K}\sum_{k\in\mathcal{K}}(\widehat{\mathbf{w}}_{k,t} + \mathbf{n}_t)$
11:     $t = t + 1$
12: end while
```

III. Joint DP Noise Variances and Transmit Power Optimization

In this section, we present a joint optimization approach to design the DP noise variances $\{\sigma_t^2\}$ and transmit power ρ. We first define the concept of neighbor datasets.

***Definition* 1.** Two data sets $\mathcal{A}$ and $\mathcal{A}'$ are neighbors if they differ by only one data point.

We now define (ϵ, δ)-DP [8], for $\epsilon \ge 0$ and $\delta \in (0, 1]$.

***Definition* 2.** ((ϵ, δ)-DP) A randomized mechanism $\mathcal{M} : \mathcal{X} \to \mathcal{R}$ satisfies (ϵ, δ)-DP if for any two neighbor data sets $\mathcal{A}, \mathcal{A}' \in \mathcal{X}$ and for any subset of output $\mathcal{S} \subseteq \mathcal{R}$ it holds that

$$\Pr[\mathcal{M}(\mathcal{A}) \in \mathcal{S}] \le e^\epsilon \Pr[\mathcal{M}(\mathcal{A}') \in \mathcal{S}] + \delta. \tag{2}$$

The (ϵ, δ)-DP definition provides a rigorous approach to quantify the privacy guarantee of a randomized mechanism $\mathcal{M}$. The condition in (2) implies that the smaller ϵ and δ, the more difficult to obtain the individual data point in $\mathcal{A}$ even though the malicious entity has the knowledge about its neighbor $\mathcal{A}'$ [8].

To facillate the (ϵ, δ)-DP analysis, we define the ℓ_2-sensitivity Δ_f of a function f [8]:

$$\Delta_f = \max_{\mathcal{A},\mathcal{A}'} \|f(\mathcal{A}) - f(\mathcal{A}')\|, \tag{3}$$

where $f(\cdot)$ is a function defined in $\mathcal{A}$ and $\mathcal{A}'$. Using [10, Lemma 1], the sensitivity of Algorithm 1 is given by

$$\Delta_f = \frac{\sqrt{\rho}C}{D}. \tag{4}$$

Next, the (ϵ, δ)-DP performance of Algorithm 1 is characterized in the following lemma.

***Lemma* 1.** For $\epsilon \geq 0$, the Algorithm 1 achieves the (ϵ, δ)-DP such that

$$1 \geq \delta \geq \tilde{\delta}, \tag{5}$$

where $\tilde{\delta} = \exp(h(\omega))$, $h(\omega) = -\frac{(2\omega\epsilon+1)^2}{8\omega}$, and $\omega = \frac{ND^2}{KC^2\rho\sum_{t=1}^{T}\frac{1}{\sigma_t^2+\xi_t^2}}$.

The proof of Lemma 1 can be found in the supplement document [18, Appendix A].

We note that the DP noise variances $\{\sigma_t^2\}$ and transmit power ρ are restricted by the total transmit power ν at each user, i.e., $\sigma_t^2 + \rho \leq \nu$, $\forall t$. It is desirable to design $\{\sigma_t^2\}$ and ρ to minimize $\tilde{\delta}$. At the same time, we also want to achieve a minimum required SNR value θ_t (e.g., required quality of service) at every model update link. These aspects are addressed in the following problem:

$$\begin{aligned} \min_{\rho,\{\sigma_t^2\}} \quad & \exp(h(\omega)), && \text{(6a)} \\ \text{subject to} \quad & 0 < \sigma_t^2 + \rho \leq \nu, \forall t, && \text{(6b)} \\ & \theta_t \leq \frac{\rho}{\sigma_t^2 + \xi_t^2}, \forall t, && \text{(6c)} \\ & 0 < \rho, 0 < \sigma_t, \forall t, && \text{(6d)} \end{aligned}$$

where the constraint (6b) indicates the power constraint per iteration and the constraint (6d) indicates the required SNR performance θ_t in the model update link.

Since $\exp(\cdot)$ in (6a) is a monotonically increasing function, the problem in (6) is equivalent to

$$\begin{aligned} \min_{\rho,\{\sigma_t^2\}} \quad & h(\omega), && \text{(7a)} \\ \text{subject to} \quad & \text{(6b)-(6d)}. && \text{(7b)} \end{aligned}$$

The problem in (7) is non-convex due to the coupling between ρ and $\{\sigma_t\}$, which is difficult to solve directly. A conventional approach to deal with this difficulty is to find a stationary solution by solving the Karush–Kuhn–Tucker (KKT) conditions [19], which is complicated when the number of constraints in (7) $(3T+1)$ becomes large. Instead of using the KKT conditions, we propose an approximate solution based on the shape of the function $h(\omega)$. It is not difficult to observe that $h(\omega) = -\frac{(2\omega\epsilon+1)^2}{8\omega}$ is a unimodal function with respect to $\omega > 0$. Therefore, it is sufficient to solve (7) by finding the maximum and minimum values of ω subject to (7b). We have the following proposition to characterize the solution of (7).

***Proposition* 1.** We define $\omega_1 = \min_{\rho,\{\sigma_t^2\}} \omega$ and $\omega_2 = \max_{\rho,\{\sigma_t^2\}} \omega$ subject to (7b), which are achieved by $\{\rho_1, \{\sigma_{t,1}^2\}\}$ and $\{\rho_2, \{\sigma_{t,2}^2\}\}$, respectively. Assume that $\omega_l = \operatorname{argmin}\{h(\omega_1), h(\omega_2)\}$, where $l = 1$ if $h(\omega_2) > h(\omega_1)$ and $l = 2$, otherwise. Then, the optimal solution $\{\rho^\star, \{\sigma_t^{2\star}\}\}$ to (7) is given by $\{\rho_l^\star, \{\sigma_{t,l}^2\}\}$.

The ω_1 and ω_2 in Proposition 1 can be obtained by applying numerical search approaches.

IV. Convergence Analysis

Given the optimal solution ρ and $\{\sigma_t^2\}$ found in Section III, we establish the convergence of Algorithm 1 by analyzing the upper bound of expected learning error $\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^\star\|^2]$ at the tth FL iteration.

We first present standard assumptions. in analyzing the convergence of FL [20].

Assumption 1. (L-smooth [21]) The local objective function $F_n(\cdot)$ is convex, differentiable, and L-smooth, $\forall n \in \mathcal{N}$, i.e., $\|\nabla F_n(\mathbf{x}) - \nabla F_n(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$, $\forall \mathbf{x}, \mathbf{y}$. We have a standard property of a L-smooth function, $F_n(\mathbf{y}) \leq F_n(\mathbf{x}) + \langle \nabla F_n(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|^2$.

Assumption 2. (μ-strong convex [21]) The loss function $F_n(\cdot)$ is μ-strong convex, $\langle \nabla F_n(\mathbf{x}) - \nabla F_n(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \mu\|\mathbf{x} - \mathbf{y}\|^2$, $\forall n \in \mathcal{N}$.

Assumption 3. The stochastic gradient at each user is unbiased $\mathbb{E}[\widetilde{\nabla} F_n(\mathbf{x})] = \nabla F_n(\mathbf{x})$ and has bounded variance $\mathbb{E}[\|\widetilde{\nabla} F_n(\mathbf{x})\|^2] < \sigma_f^2$, $\forall n \in \mathcal{N}$.

***Lemma* 2.** ([22, Lemma F.1]) Suppose $\widetilde{\mathbf{x}}$ is an unbiased estimator of $\mathbf{x}$. We define $b(\mathbf{x}, C)$ as the bias of clipping operation $\min\{1; \frac{C}{\|\widetilde{\mathbf{x}}\|}\}\widetilde{\mathbf{x}}$, which is given by $b(\widetilde{\mathbf{x}}, C) = \left\|\mathbf{x} - \min\{1; \frac{C}{\|\widetilde{\mathbf{x}}\|}\}\widetilde{\mathbf{x}}\right\|$. Then, we have $b(\widetilde{\mathbf{x}}, C) \leq \frac{\mathbb{E}[\|\widetilde{\mathbf{x}}\|^p]}{C^{p-1}}$, for $p > 1$.

Given the latter assumptions, we readily characterize the upper bound of learning error of Algorithm 1. We begin with establishing intermediate bounds, which are useful to our analysis. We define $\widehat{\mathbf{w}}_{t+1} = \mathbf{w}_t + \frac{1}{N}\sum_{n=1}^{N}\mathbf{u}_{n,t}$ and $\overline{\mathbf{w}}_{t+1} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{w}_{n,t}^{(E)}$, for $t = 0, \ldots, T-1$. Applying Cauchy-Schwarz inequality to $\mathbb{E}[\|\mathbf{w}_{t+1} - \mathbf{w}^\star\|^2] = \mathbb{E}[\|\mathbf{w}_{t+1} - \widehat{\mathbf{w}}_{t+1} + \widehat{\mathbf{w}}_{t+1} - \overline{\mathbf{w}}_{t+1} + \overline{\mathbf{w}}_{t+1} - \mathbf{w}^\star\|^2]$ yields

$$\begin{aligned} \mathbb{E}[\|\mathbf{w}_{t+1} - \mathbf{w}^\star\|^2] &\leq 3\mathbb{E}[\|\mathbf{w}_{t+1} - \widehat{\mathbf{w}}_{t+1}\|^2] \\ &+ 3\mathbb{E}[\|\widehat{\mathbf{w}}_{t+1} - \overline{\mathbf{w}}_{t+1}\|^2] + 3\mathbb{E}[\|\overline{\mathbf{w}}_{t+1} - \mathbf{w}^\star\|^2]. \end{aligned} \tag{8}$$

In the following, the goal is to upper bound the three terms on the right-hand side (r.h.s.) of (8), which are presented in Lemmas 3-5, respectively.

***Lemma* 3.** Suppose $\eta_t \leq \min\{\frac{1}{EL}, \frac{\mu}{L^2}\}$. The following upper bound holds,

$$\mathbb{E}[\|\widehat{\mathbf{w}}_{t+1} - \mathbf{w}^\star\|^2] \leq (1 + N\eta_t^2)(1 - \mu\eta_t)^E\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^\star\|^2] + \frac{E(E-1)^2L^2\sigma_f^2}{N}e\eta_t^2 + \frac{E^2\sigma^2\eta_t^2}{2} + E^2(E-1)L^2\sigma^2e\eta_t^4. \tag{9}$$

***Lemma* 4.** Suppose $\eta_t \leq \min\{\frac{1}{EL}, \frac{\mu}{L^2}\}$. The following upper bound holds,

$$\mathbb{E}[\|\widehat{\mathbf{w}}_{t+1} - \overline{\mathbf{w}}_{t+1}\|^2] \leq \frac{1}{C}\Big(2E^2L^2\eta_t^2\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^\star\|^2] + 2E^2\sigma_f^2\eta_t^2 + 2(E-1)E^2L^2\sigma_f^2 e\eta_t^4\Big). \quad (10)$$

***Lemma* 5.** Suppose $\eta_t \leq \min\{\frac{1}{EL}, \frac{\mu}{L^2}\}$. The following upper bound holds,

$$\mathbb{E}[\|\mathbf{w}_{t+1} - \widehat{\mathbf{w}}_{t+1}\|^2] \leq \alpha\Big(2E^2L^2\eta_t^2\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^\star\|^2] + 2E^2\sigma_f^2\eta_t^2 + 2(E-1)E^2L^2\sigma_f^2 e\eta_t^4\Big) + \max_t \frac{\rho}{\theta_t K}, \quad (11)$$

where $\alpha = \Big(\rho + 1 + \frac{\rho+1}{K}\Big)\Big(\frac{2}{C} + 2\Big)$.

The proofs of Lemmas 3-5 are relegated to the supplemental document [18, Appendices B-D].

Incorporating (9)-(11) into (8) yields

$$\mathbb{E}[\|\mathbf{w}_{t+1} - \mathbf{w}^\star\|^2] \leq m_{1,t}\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^\star\|^2] + m_{2,t}, \forall t, \quad (12)$$

where $m_{1,t} = 3(1 + N\eta_t^2)(1 - \mu\eta_t)^E + (\frac{6}{C} + 6\alpha)E^2L^2\eta_t^2$ and $m_{2,t} = 3E(E-1)^2L^2\frac{\sigma_f^2}{N}e\eta_t^2 + 3\frac{E^2\sigma^2\eta_t^2}{2} + 3E^2(E-1)L^2\sigma^2 e\eta_t^4 + \frac{6}{C}E^2\sigma_f^2\eta_t^2 + \frac{6}{C}(E-1)E^2L^2\sigma_f^2 e\eta_t^4 + 6\alpha E^2\sigma_f^2\eta_t^2 + 6\alpha(E-1)E^2L^2\sigma_f^2 e\eta_t^4 + 3(\max_t \frac{\rho}{\theta_t K})$, for $\eta_t \leq \min\{\frac{1}{EL}, \frac{\mu}{L^2}\}$. From the upper bound in (12), we have the following theorem that characterizes the upper bound of the expected learning error $\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^\star\|^2]$.

***Theorem* 1.** Assume t_0 such that $\eta_{t_0} = \frac{4}{\mu E t_0} \leq \min\left\{\frac{1}{EL}, \frac{\mu}{L^2}, \frac{1}{E\mu}, \frac{E\mu}{4N}, \frac{\mu}{4E((\frac{6}{C}+6\alpha)L^2+\mu^2)}\right\}$ and define $\eta_t = \frac{4}{\mu E t}$, for $t \geq t_0$. The expected learning error of Algorithm 1 at the tth learning iterations is upper bounded, $\forall t \geq t_0$, by

$$\mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^\star\|^2] \leq \frac{t_0}{t}\mathbb{E}[\|\mathbf{w}_{t_0} - \mathbf{w}^\star\|^2] + tM_0 + \frac{M_1}{E^2\mu^2 t} + \frac{M_2}{E^4\mu^4 t^2}, \quad (13)$$

where $M_0 = 3(\max_t \frac{\rho}{K\theta_t})$, $M_1 = 48E(E-1)^2L^2\frac{\sigma_f^2}{N}e + 24E^2\sigma^2 + \frac{96}{C}E^2\sigma_f^2 + 96\alpha E^2\sigma_f^2$, and $M_2 = 192E^2(E-1)L^2\sigma^2 e\eta_t^4 + \frac{384}{C}(E-1)E^2L^2\sigma_f^2 e + 384\alpha(E-1)E^2L^2\sigma_f^2 e$.

The proof of Theorem 1 can be found in the supplemental material [18, Appendix E].

On the r.h.s. of (13), $\frac{t_0}{t}\mathbb{E}[\|\mathbf{w}_{t_0} - \mathbf{w}^\star\|^2]$, $\frac{M_1}{E^2\mu^2 t}$, and $\frac{M_2}{E^4\mu^4 t^2}$ are independent of the DP and channel noises. We note that these values converge to 0 as t tends to infinity. However, the term tM_0 on the r.h.s. of (13) depends on the DP and wireless channel noises and it diverges as t increases. This observation implies that the Algorithm 1 may diverge due to the accumulating DP and communication noises as the FL proceeds. Therefore, it is necessary to stop Algorithm 1 after certain FL iterations $t^\star$. To characterize $t^\star$, we minimize the r.h.s. of (13) with respect to t

$$t^\star = \operatorname*{argmin}_t \frac{t_0}{t}\mathbb{E}[\|\mathbf{w}_{t_0} - \mathbf{w}^\star\|^2] + tM_0 + \frac{M_1}{E^2\mu^2 t} + \frac{M_2}{E^4\mu^4 t^2}, \quad (14)$$

It is not difficult to see that the problem (14) is convex. Hence, $t^\star$ can be characterized by equating the first-order derivative equation to be zero.

V. Simulation

We evaluate the learning accuracy of the proposed algorithm terminated at the optimal number of learning iterations $t^\star$ determined by (14).

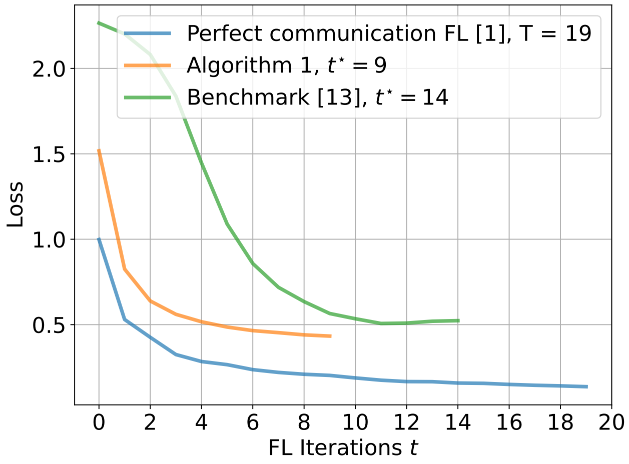

Fig. 2: Training loss of Algorithm 1 with $t^\star = 9$, benchmark [13] with $t^\star = 14$, and the perfect communication FL [1] with $T = 19$.

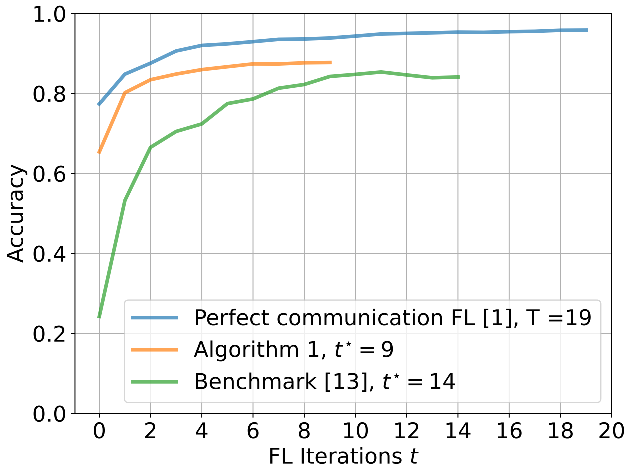

Fig. 3: Training accuracy of Algorithm 1 with $t^\star = 9$, benchmark [13] with $t^\star = 14$, and the perfect communication FL [1] with $T = 19$.

We set the total number of users $N = 100$, the number of selected users in each learning iteration $K = 10$, the number of local SGD epochs $E = 10$, and the clipping threshold $C = 10$. The (ϵ, δ)-DP requirements are $\epsilon = 10$ and $\delta = 0.001$. We use the MNIST dataset [23] for model training, containing 60000 training and 10000 testing data samples of handwritten digits from 0 to 9. We train a feed-forward neural network using Algorithm 1 that consists of three fully-connected networks.

Fig. 2 illustrates the training loss of Algorithm 1, benchmark [13], and perfect communication FL [1]. It should be noted that the perfect communication FL does not guarantee any data privacy in model updates and does not take into account communication noises $\{\mathbf{n}_t\}$ in the model updates. Compared to the benchmark [13], the proposed approach achieves a smaller training loss with $t^\star = 9$ determined by (14). On the other hand, the benchmark requires $t^\star = 14$ iterations to achieve a close training loss performance to that of Algorithm 1. This implies that the Algorithm 1 requires fewer communication overhead than the benchmark [13].

Fig. 3 illustrates the testing accuracy performance of the learning models in Fig. 2. It can be seen in Fig. 3 that the proposed approach achieves the 87% testing accuracy performance in $t^\star = 9$ iterations while the benchmark [13] requires $t^\star = 14$ to achieve the similar performance. The proposed approach attains a lower testing accuracy performance than the perfect communication due to the DP noise introduced into the model update.

VI. Conclusion

A differentially private FL algorithm has been proposed to jointly address the privacy-preserving and communication quality. To provide a privacy guarantee while maintaining the communication quality in the model update link, we jointly optimized the time-varying DP noise variances and transmit power subject to the minimum SNR requirements. The convergence behavior of the proposed algorithm was analyzed. Based on the convergence analysis, the number of FL iterations is optimized to minimize the learning error of the proposed algorithm. Through simulations, we show the advantages of the proposed algorithm.

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*. PMLR, 2017, pp. 1273–1282.

[2] G. Zhu, Y. Du, D. Gündüz, and K. Huang, "One-bit over-the-air aggregation for communication-efficient federated edge learning: Design and convergence analysis," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 2120–2135, 2021.

[3] Y. Wang, Y. Xu, Q. Shi, and T.-H. Chang, "Quantized federated learning under transmission delay and outage constraints," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 323–341, 2022.

[4] K. Omid and T. Kim, "Distributed quantized transmission and fusion for federated machine learning," in *VTC 2023 - IEEE Vehicular Technology Conference*, 2023.

[5] T. Gafni, N. Shlezinger, K. Cohen, Y. C. Eldar, and H. V. Poor, "Federated learning: A signal processing perspective," *IEEE Signal Processing Magazine*, vol. 39, no. 3, pp. 14–41, 2022.

[6] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[7] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, "Inverting gradients - how easy is it to break privacy in federated learning?" in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33, 2020, pp. 16 937–16 947.

[8] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3–4, p. 211–407, aug 2014.

[9] N. Agarwal, A. T. Suresh, F. X. X. Yu, S. Kumar, and B. McMahan, "cpSGD: Communication-efficient and differentially-private distributed SGD," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018.

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

[11] X. Zhang, X. Chen, M. Hong, S. Wu, and J. Yi, "Understanding clipping for federated learning: Convergence and client-level differential privacy," ser. Proceedings of Machine Learning Research, vol. 162. Proceedings of the 39th International Conference on Machine Learning, 17–23 Jul 2022, pp. 26 048–26 067.

[12] W.-N. Chen, P. Kairouz, and A. Özgür, "Breaking the communication-privacy-accuracy trilemma," *IEEE Transactions on Information Theory*, vol. 69, no. 2, pp. 1261–1281, 2023.

[13] X. Yuan, W. Ni, M. Ding, K. Wei, J. Li, and H. V. Poor, "Amplitude-varying perturbation for balancing privacy and utility in federated learning," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 1884–1897, 2023.

[14] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, "Differentially private aircomp federated learning with power adaptation harnessing receiver noise," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.

[15] D. Liu and O. Simeone, "Privacy for free: Wireless federated learning via uncoded transmission with adaptive power control," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 170–185, 2021.

[16] M. S. E. Mohamed, W.-T. Chang, and R. Tandon, "Privacy amplification for federated learning via user sampling and wireless aggregation," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3821–3835, 2021.

[17] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '16. Association for Computing Machinery, 2016, p. 308–318.

[18] D. Q. Nguyen and T. Kim, "Supplement to Time-Varying Noise Perturbation and Power Control for Differential-Privacy-Preserving Wireless Federated Learning," 2023. [Online]. Available: https://github.com/quandku/Supplement-to-Asilomar-2023

[19] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[20] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization," in *International Conference on Artificial Intelligence and Statistics*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:203593931

[21] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, 1st ed. Springer Publishing Company, Incorporated, 2014.

[22] R. Das, S. Kale, Z. Xu, T. Zhang, and S. Sanghavi, "Beyond uniform lipschitz condition in differentially private optimization," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23. JMLR.org, 2023.

[23] Y. LeCun and C. Cortes, "MNIST handwritten digit database," 2010. [Online]. Available: http://yann.lecun.com/exdb/mnist/