Privacy-Preserving Wireless Federated Learning Exploiting Inherent Hardware Impairments

Privacy-Preserving Wireless Federated Learning Exploiting Inherent Hardware Impairments

Sina Rezaei Aghdam[1], Ehsan Amid[2], Marija Furdek[1], and Alexandre Graell i Amat[1]

[1]*Chalmers University of Technology, Gothenburg, Sweden* {sinar, furdek, alexandre.graell}@chalmers.se
[2]*Google Research, Mountain View, CA* eamid@google.com

***Abstract*—We consider a wireless federated learning system where multiple data holder edge devices collaborate to train a global model via sharing their parameter updates with an honest-but-curious parameter server. We demonstrate that the inherent hardware-induced distortion perturbing the model updates of the edge devices can be exploited as a privacy-preserving mechanism. In particular, we model the distortion as power-dependent additive Gaussian noise and present a power allocation strategy that provides privacy guarantees within the framework of differential privacy. We conduct numerical experiments to evaluate the performance of the proposed power allocation scheme under different levels of hardware impairments.**

***Index Terms*—Over-the-air computation, wireless federated learning, differential privacy, hardware impairments.**

I. Introduction

Federated learning has recently emerged as a promising technique for distributed machine learning in a variety of applications, including mobile edge computing [1]. Unlike conventional centralized machine learning techniques that require datasets to reside on a single server, federated learning leverages the local processing capabilities of edge devices and requires only a limited amount of communication between the devices and the server. In particular, instead of offloading the entire data, each user uses its own dataset to train a local model and sends a small update to the global model maintained by the server. This offers significant gains in terms of privacy and communication efficiency [2].

When deployed over wireless networks, the superposition nature of wireless channels can be used as a natural data aggregator for federated learning. This phenomenon is referred to as *over-the-air aggregation*, where simultaneously transmitted analog waves from different edge devices are weighted by channel coefficients and superposed at the server [3]. Accurate and fair computation of the global model requires that the gradient estimates received from different devices have the same power at the parameter server. One way to address this issue is to perform channel-inversion power control at the edge devices [4], [5]. Alternatively, the parameter server can mitigate channel fading by using multiple receiver antennas [6], [7].

Despite its various attractive features, wireless federated learning in its primary form does not provide sufficiently strong privacy guarantees. Although edge devices do not explicitly share their data in its original format, several recent works have shown that model parameters can themselves be informative to an honest-but-curious server (see, e.g., [8]–[12]), leading to privacy leakage. Accordingly, the study of privacy for wireless federated learning has been the subject of much recent research [13]–[16]. These works adopt the notion of differential privacy as a formal model for quantifying information disclosure about individuals and introduce strategies for limiting it. Specifically, in [13], [14], the authors show that by adding artificial noise to the model parameters on the edge device and properly adjusting the variance of this noise, different levels of differential privacy protection can be achieved. Moreover, [15], [16] show that the inherent channel noise can be harnessed via proper power allocation to achieve privacy for free.

In this paper, we demonstrate that in addition to channel noise, the distortion introduced by the imperfect hardware of edge devices can also be used as a privacy-preserving mechanism. Motivated by the analytical analysis and experimental validations in [17]–[19], we model distortion as an additive Gaussian noise whose variance is proportional to the signal power. Within the differential privacy framework, we derive an upper-bound on the privacy violation probability and use it for formulating a privacy preservation condition. We then propose a distortion-aware power allocation scheme to satisfy this condition. We conduct numerical experiments to evaluate the performance of the proposed scheme under different levels of hardware impairments. Our numerical results reveal that in realistic scenarios with imperfect hardware, taking into account the impact of distortion in power allocation can considerably reduce the performance loss caused by enforcing differential privacy constraints. Exploiting distortion in realistic hardware can even result in improved performance compared to ideal distortion-free hardware.

The remainder of this paper is organized as follows. In Section II, we introduce our system model and the required technical background, including the definition of differential privacy. The derivation of the privacy preservation criterion is described in Section III. In Section IV, we present a power allocation scheme that satisfies the differential privacy criterion. In Section V, we evaluate the performance of the proposed solution using numerical examples. Finally, we conclude the paper in Section VI.

The research in this paper has been supported by Chalmers Artificial Intelligence Research Centre (CHAIR).

Notation: Throughout the paper, $\mathbb{R}$ and $\mathbb{C}$ represent the sets of real and complex values, respectively. We denote a zero-mean normal distribution with variance σ^2 by $\mathcal{N}(0,\sigma^2)$. Zero-mean complex Gaussian random variables are represented by $\mathcal{CN}(0,\sigma^2)$. The cardinality of a set $\mathcal{A}$ is denoted by $|\mathcal{A}|$. Furthermore, the difference between two sets is defined as $\mathcal{A}'-\mathcal{A}''=\{x \mid x\in\mathcal{A}', x\notin\mathcal{A}''\}$ and $\|\mathcal{A}'-\mathcal{A}''\|_1$ denotes its cardinality. The union of sets $\mathcal{A}_1,\ldots,\mathcal{A}_K$ is represented by $\mathcal{A}=\cup_{k=1}^{K}\mathcal{A}_k$. Vectors are denoted by boldfaced letters. The transpose and the Euclidean norm of a vector $\boldsymbol{a}$ are denoted by $\boldsymbol{a}^{\mathsf{T}}$ and $\|\boldsymbol{a}\|$, respectively.

II. System Model

We consider a wireless federated learning system (as shown in Fig. 1) consisting of K single-antenna edge devices that attempt to collaboratively learn a shared model with the aid of a single-antenna parameter server. Each device k stores a local dataset, denoted by $\mathcal{B}_k$, which consists of

$$\mathcal{B}_k=\left\{\left(\boldsymbol{u}_{k,i},\upsilon_{k,i}\right)\right\}_{i=1}^{|\mathcal{B}_k|}, \tag{1}$$

where $\boldsymbol{u}_{k,i}$ denotes the i-th training sample at the k-th device and $\upsilon_{k,i}\in\mathbb{R}$ is its corresponding label. The federated learning process finds the optimal model parameters $\boldsymbol{\theta}\in\mathbb{R}^d$ that minimize the global loss function $F(\boldsymbol{\theta})$:

$$\underset{\boldsymbol{\theta}}{\text{minimize}}\quad F(\boldsymbol{\theta})\triangleq\frac{1}{K}\sum_{k=1}^{K}F_k(\boldsymbol{\theta}). \tag{2}$$

Here, $F_k(\boldsymbol{\theta})$ is the local loss function given by

$$F_k(\boldsymbol{\theta})=\frac{1}{|\mathcal{B}_k|}\sum_{(\boldsymbol{u}_{k,i},\upsilon_{k,i})\in\mathcal{B}_k}f(\boldsymbol{\theta},\boldsymbol{u}_{k,i},\upsilon_{k,i}), \tag{3}$$

and $f(\boldsymbol{\theta},\boldsymbol{u}_{k,i},\upsilon_{k,i})$ is the sample loss that quantifies the prediction error of the model $\boldsymbol{\theta}$ on the training samples $\boldsymbol{u}_{k,i}$ with respect to the labels $\upsilon_{k,i}$.

A. Learning Protocol

In order to solve (2), a distributed stochastic gradient descent (SGD) is adopted by the parameter server and the edge devices via the following procedure:

- **Step 1:** The parameter server broadcasts an initial model, i.e., $\boldsymbol{\theta}^{(1)}$, to the edge devices allowing them to initialize their learning model.
- **Step 2:** Each device performs local training on its own dataset and transmits the trained learning model parameters to the server. More specifically, at each communication round $t=1,\ldots,T$, each device computes the local gradient

$$\nabla F_k(\boldsymbol{\theta}^{(t)})=\frac{1}{|\mathcal{B}_k|}\sum_{(\boldsymbol{u}_{k,i},\upsilon_{k,i})\in\mathcal{B}_k}\nabla f(\boldsymbol{\theta}^{(t)},\boldsymbol{u}_{k,i},\upsilon_{k,i}), \tag{4}$$

  and sends it to the server. Here, $F_k(\boldsymbol{\theta}^{(t)})$ represents the local loss function.

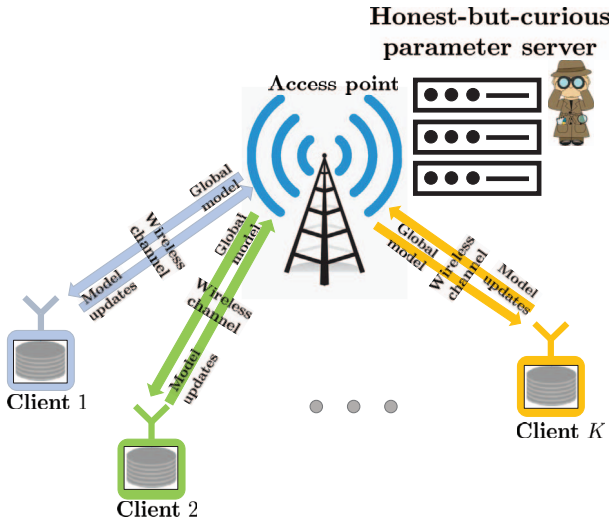

Fig. 1. Wireless federated learning with K edge devices and an honest-but-curious parameter server.

- **Step 3:** The transmitted updates are aggregated over the air and received by the server. In particular the received aggregated signal at the parameter server, i.e., $\widehat{\nabla F}(\boldsymbol{\theta}^{(t)})$, yields an estimation of the global gradient. This allows the server to update the global model using gradient descent

$$\boldsymbol{\theta}^{(t+1)}=\boldsymbol{\theta}^{(t)}-\eta\widehat{\nabla F}(\boldsymbol{\theta}^{(t)}), \tag{5}$$

  where η denotes the learning rate. The updated model is then broadcasted back to the edge devices.
- **Step 4:** Steps 2 and 3 are repeated until a convergence criterion is met.

B. Communication Model

All edge devices communicate with the edge server over a shared wireless channel. We assume an ideal downlink channel where the server broadcasts the initial and the aggregated models (Steps 1 and 3) through a distortion-free channel. This assumption can be justified by the fact that the edge server can communicate through a base station, which usually has less stringent constraints in terms of power and hardware complexity compared to edge devices. In the uplink transmission (Step 2), in order to facilitate the over-the-air aggregation of model updates, the local gradients are amplitude-modulated at the devices and sent to the server using an analog transmission scheme [20]. The non-ideal hardware in the edge devices distorts the transmit signal. We model this distortion as power-dependent additive Gaussian noise. At communication round $t=1,\ldots,T$, the transmitted signal from the k-th device can be written as

$$\boldsymbol{x}_k^{(t)}=\sqrt{\frac{\rho_k^{(t)}}{\|\boldsymbol{g}_k^{(t)}\|^2}}\boldsymbol{g}_k^{(t)}+\boldsymbol{e}_k^{(t)}, \tag{6}$$

where $\rho_k^{(t)}$ denotes the transmit power of device k at communication round t, and $\boldsymbol{g}_k^{(t)} = |\mathcal{B}_k| \nabla F_k\big(\boldsymbol{\theta}^{(t)}\big)$ is the scaled version of the local gradient in (4). The additive distortion noise $\boldsymbol{e}_k^{(t)}$ is distributed according to $\mathcal{N}\big(\mathbf{0}, \kappa_k \rho_k^{(t)} \mathbf{I}\big)$[1] where $\kappa_k \in [0, 1)$ is the proportionality coefficient. The rationale behind this model is that, in many practical cases, a fixed portion of the signal is turned into distortion [17]. This can happen, for example, due to amplitude-to-amplitude (AM-AM) nonlinearities in the power amplifier [18].

We further assume that the edge devices have perfect channel state information and accordingly, they compensate for the phases of their channels in the course of transmission. The received signal at the server can therefore be written as

$$\boldsymbol{y}^{(t)} = \sum_{k=1}^{K} \big|h_k^{(t)}\big| \boldsymbol{x}_k^{(t)} + \boldsymbol{w}^{(t)}, \tag{7}$$

where $h_k^{(t)}$ denotes the fading channel coefficient for the k-th device at the t-th communication round, and $\boldsymbol{w}^{(t)} \sim \mathcal{N}(0, N_0)$ is independent and identically distributed (i.i.d.) additive Gaussian noise. We consider the case of block fading channels where the channel coefficients remain unchanged within each time slot, and change independently from one communication round to the other.

C. Threat Model and Privacy Mechanism

We consider an honest-but-curious server that follows the protocol instructions correctly, but it may attempt to break privacy through observing the signals received throughout the learning process. Specifically, the server observes

$$\boldsymbol{y}^{(t)} = \sum_{k=1}^{K} b_k^{(t)} \big|h_k^{(t)}\big| \boldsymbol{g}_k^{(t)} + \boldsymbol{w}_{\text{eff}}^{(t)}, \tag{8}$$

over $t = 1, \ldots, T$, and can infer information about the edge devices' private data. Here, $b_k^{(t)}$ is given by

$$b_k^{(t)} = \sqrt{\frac{\rho_k^{(t)}}{\|\boldsymbol{g}_k^{(t)}\|^2}}, \tag{9}$$

and $\boldsymbol{w}_{\text{eff}}^{(t)} = \boldsymbol{w}^{(t)} + \sum_{k=1}^{K} \big|h_k^{(t)}\big| \boldsymbol{e}_k^{(t)}$ is the effective noise distributed according to $\mathcal{N}\big(0, (\sigma^{(t)})^2\big)$, with

$$\sigma^{(t)} = \sqrt{N_0 + \sum_{k=1}^{K} \kappa_k \rho_k^{(t)} |h_k^{(t)}|^2}. \tag{10}$$

The effective noise $\boldsymbol{w}_{\text{eff}}^{(t)}$ will be exploited as a privacy-preserving mechanism in the next section.

D. Differential Privacy

Differential privacy is a strong and provable privacy guarantee for an individual's input to a randomized function, i.e., a privacy mechanism. Informally, this guarantee implies that the behavior of the mechanism is essentially unchanged over inputs from any pair of adjacent datasets. In our case, adjacent datasets are the same, except for an example associated with a single edge device. Mathematically, this can be expressed as two global datasets $\mathcal{B}' = \cup_{k=1}^{K} \mathcal{B}_k'$ and $\mathcal{B}'' = \cup_{k=1}^{K} \mathcal{B}_k''$ where $\|\mathcal{B}_i' - \mathcal{B}_i''\|_1 = 1$ for some device i and $\|\mathcal{B}_k' - \mathcal{B}_k''\|_1 = 0$ for all $k = 1, \ldots, K$ except for $k = i$ [16]. Using this definition, we can now formally define the notion of (ϵ, δ) differential privacy.

Definition 1 (Differential Privacy [16], [21])**.** Let $\epsilon > 0$ and $0 \leq \delta \leq 1$. For any two adjacent datasets $\mathcal{B}'$ and $\mathcal{B}''$, (ϵ, δ) differential privacy requires that

$$\mathrm{P}(\boldsymbol{y}|\mathcal{B}') \leq \exp(\epsilon) \mathrm{P}(\boldsymbol{y}|\mathcal{B}'') + \delta, \tag{11}$$

where $\mathrm{P}(\boldsymbol{y}|\mathcal{B}')$ and $\mathrm{P}(\boldsymbol{y}|\mathcal{B}'')$ stand for the distributions of received signals $\boldsymbol{y} = \big\{\boldsymbol{y}^{(t)}\big\}_{t=1}^{T}$ conditioned on the uses of either of the adjacent datasets.

The (ϵ, δ) differential privacy condition in (11) can be rewritten as [21, Lemma 3.17]

$$\Pr(|\mathcal{L}_{\mathcal{B}', \mathcal{B}''}(\boldsymbol{y})| \leq \epsilon) \geq 1 - \delta, \tag{12}$$

where

$$\mathcal{L}_{\mathcal{B}', \mathcal{B}''}(\boldsymbol{y}) = \ln\left(\frac{\mathrm{P}(\boldsymbol{y}|\mathcal{B}')}{\mathrm{P}(\boldsymbol{y}|\mathcal{B}'')}\right) \tag{13}$$

is referred to as differential privacy loss. According to (12), (ϵ, δ) differential privacy ensures that the absolute value of the privacy loss is bounded by ϵ with probability at least $1 - \delta$. Smaller values of ϵ and δ provide therefore stronger privacy guarantees. More precisely, if (12) is satisfied for sufficiently small ϵ and δ, this implies that it is statistically impossible for the parameter server to detect whether a particular training sample is included in the training dataset, even in the extreme case when all other training samples are known.

III. Privacy Preservation Condition

In this section, we derive a privacy preservation condition by applying the notion of differential privacy to our problem. To this end, we first derive an expression for the privacy loss due to the disclosure of the received signal $\boldsymbol{y} = \big[\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(T)}\big]$ at the parameter server by exploiting inherent distortion as the privacy-preserving mechanism. This can be written as

$$\begin{aligned}\mathcal{L}_{\mathcal{B}', \mathcal{B}''}(\boldsymbol{y}) &= \ln\left(\prod_{t=1}^{T} \frac{\mathrm{P}(\boldsymbol{y}^{(t)}|\boldsymbol{y}^{(t-1)}, \ldots, \boldsymbol{y}^{(1)}, \mathcal{B}')}{\mathrm{P}(\boldsymbol{y}^{(t)}|\boldsymbol{y}^{(t-1)}, \ldots, \boldsymbol{y}^{(1)}, \mathcal{B}'')}\right) \\ &= \sum_{t=1}^{T} \ln\left(\frac{\mathrm{P}(\boldsymbol{y}^{(t)}|\boldsymbol{y}^{(t-1)}, \ldots, \boldsymbol{y}^{(1)}, \mathcal{B}')}{\mathrm{P}(\boldsymbol{y}^{(t)}|\boldsymbol{y}^{(t-1)}, \ldots, \boldsymbol{y}^{(1)}, \mathcal{B}'')}\right). \end{aligned} \tag{14}$$

[1] Gaussianity can be justified analytically by the central limit theorem, since $\boldsymbol{e}_k^{(t)}$ describes the aggregate effect of different residual hardware impairments [19].

From (8) and for a given dataset $\mathcal{B}=\cup_{k=1}^{K}\mathcal{B}_k$, we have

$$\mathrm{P}\big(\boldsymbol{y}^{(t)}|\boldsymbol{y}^{(t-1)},\ldots,\boldsymbol{y}^{(1)},\mathcal{B}\big)=\frac{1}{\sigma^{(t)}\sqrt{2\pi}} \cdot\exp\left(-\frac{\left\|\boldsymbol{y}^{(t)}-\sum_{k=1}^{K}b_k^{(t)}(\mathcal{B}_k)\big|h_k^{(t)}\big|\boldsymbol{g}_k^{(t)}(\mathcal{B}_k)\right\|^2}{2(\sigma^{(t)})^2}\right), \quad (15)$$

where $b_k^{(t)}(\mathcal{B}_k)$ can be calculated by replacing $\boldsymbol{g}_k^{(t)}(\mathcal{B}_k)$ in (9). Now, by using (15) in (14) we obtain

$$\begin{aligned}
&\mathcal{L}_{\mathcal{B}',\mathcal{B}''}(\boldsymbol{y})\\
&=\sum_{t=1}^{T}\ln\left(\frac{\exp\left(-\frac{\left\|\boldsymbol{y}^{(t)}-\sum_{k=1}^{K}b_k^{(t)}(\mathcal{B}'_k)\big|h_k^{(t)}\big|\boldsymbol{g}_k^{(t)}(\mathcal{B}'_k)\right\|^2}{2(\sigma^{(t)})^2}\right)}{\exp\left(-\frac{\left\|\boldsymbol{y}^{(t)}-\sum_{k=1}^{K}b_k^{(t)}(\mathcal{B}''_k)\big|h_k^{(t)}\big|\boldsymbol{g}_k^{(t)}(\mathcal{B}''_k)\right\|^2}{2(\sigma^{(t)})^2}\right)}\right)\\
&=\sum_{t=1}^{T}\ln\left(\frac{\exp\left(-\frac{\|\boldsymbol{w}_{\text{eff}}^{(t)}\|^2}{2(\sigma^{(t)})^2}\right)}{\exp\left(-\frac{\|\boldsymbol{w}_{\text{eff}}^{(t)}+\boldsymbol{v}^{(t)}\|^2}{2(\sigma^{(t)})^2}\right)}\right)\\
&=\sum_{t=1}^{T}\frac{\|\boldsymbol{v}^{(t)}\|^2+2(\boldsymbol{w}_{\text{eff}}^{(t)})^{\mathsf{T}}\boldsymbol{v}^{(t)}}{2(\sigma^{(t)})^2}\triangleq\Gamma, \quad (16)
\end{aligned}$$

where

$$\boldsymbol{v}^{(t)}=\sum_{k=1}^{K}\big|h_k^{(t)}\big|\Big(b_k^{(t)}(\mathcal{B}''_k)\boldsymbol{g}_k^{(t)}(\mathcal{B}''_k)-b_k^{(t)}(\mathcal{B}'_k)\boldsymbol{g}_k^{(t)}(\mathcal{B}'_k)\Big). \quad (17)$$

By applying the triangle inequality, the norm of $\boldsymbol{v}^{(t)}$ can be bounded as

$$\|\boldsymbol{v}^{(t)}\|\le 2\max_k\sqrt{\rho_k^{(t)}}\big|h_k^{(t)}\big|\triangleq\Delta^{(t)}. \quad (18)$$

Following similar steps as in [21, Appendix A] and [16, Eq. (52)], the following upper-bound is obtained on the privacy violation probability

$$\begin{aligned}
\Pr\left(|\Gamma|>\epsilon\right)&\overset{\text{(i)}}{\le}\Pr\left(\left|\sum_{t=1}^{T}\frac{(\boldsymbol{w}_{\text{eff}}^{(t)})^{\mathsf{T}}\boldsymbol{v}^{(t)}}{(\sigma^{(t)})^2}\right|>\epsilon-\sum_{t=1}^{T}\frac{\|\boldsymbol{v}^{(t)}\|^2}{2(\sigma^{(t)})^2}\right)\\
&=2\Pr\left(\sum_{t=1}^{T}\frac{(\boldsymbol{w}_{\text{eff}}^{(t)})^{\mathsf{T}}\boldsymbol{v}^{(t)}}{(\sigma^{(t)})^2}>\epsilon-\sum_{t=1}^{T}\frac{\|\boldsymbol{v}^{(t)}\|^2}{2(\sigma^{(t)})^2}\right)\\
&\overset{\text{(ii)}}{\le}2\Pr\left(\Lambda>\epsilon-\sum_{t=1}^{T}\frac{1}{2}\left(\frac{\Delta^{(t)}}{\sigma^{(t)}}\right)^2\right), \quad (19)
\end{aligned}$$

where (i) comes from the inequality $\Pr(|X+c|>\epsilon)\le\Pr(|X|+c>\epsilon)$ for any arbitrary $c\ge 0$, and (ii) is obtained using (18) where

$$\Lambda\sim\mathcal{CN}\left(0,\sum_{t=1}^{T}\left(\frac{\Delta^{(t)}}{\sigma^{(t)}}\right)^2\right). \quad (20)$$

The upper-bound in (19) can therefore be written as

$$\Pr\left(|\Gamma|>\epsilon\right)\le\frac{2}{\sqrt{2\pi\sum_{t=1}^{T}\left(\frac{\Delta^{(t)}}{\sigma^{(t)}}\right)^2}}\int_{s}^{\infty}\exp\left(-\frac{z^2}{2\sum_{t=1}^{T}\left(\frac{\Delta^{(t)}}{\sigma^{(t)}}\right)^2}\right)\mathrm{d}z, \quad (21)$$

where

$$s=\epsilon-\sum_{t=1}^{T}\frac{1}{2}\left(\frac{\Delta^{(t)}}{\sigma^{(t)}}\right)^2>0. \quad (22)$$

Finally, by using (19)-(22) in (12), the privacy preservation condition is expressed as

$$\frac{2}{\sqrt{2\pi\nu}}\int_{\epsilon-\frac{\nu}{2}}^{\infty}\exp\left(-\frac{z^2}{2\nu}\right)\mathrm{d}z<\delta, \quad (23)$$

where

$$\nu=\sum_{t=1}^{T}\left(\frac{\Delta^{(t)}}{\sigma^{(t)}}\right)^2=\sum_{t=1}^{T}\frac{\left(2\max_k\sqrt{\rho_k^{(t)}}\big|h_k^{(t)}\big|\right)^2}{N_0+\sum_{k=1}^{K}\kappa_k\rho_k^{(t)}|h_k^{(t)}|^2}. \quad (24)$$

IV. Privacy-Preserving Power Allocation Scheme

In this section, we present an offline-optimized privacy-preserving power allocation scheme. In particular, we assume that the parameters $\{h_k^{(t)},\kappa_k,N_0\}$ are known and we seek the values $\rho_k^{(t)}$ for $k=1,\ldots,K$ and $t=1,\ldots,T$ to minimize the distortion of the recovered gradient from the received signal in (8) with respect to the ground-truth global gradient, i.e.,

$$\begin{aligned}
\mathrm{MSE}^{(t)}&=\mathbb{E}\left[\left\|\widehat{\nabla F}(\boldsymbol{\theta}^{(t)})-\nabla F(\boldsymbol{\theta}^{(t)})\right\|^2\right]\\
&=\frac{1}{K^2}\mathbb{E}\left[\left\|\sum_{k=1}^{K}\frac{\sqrt{\rho_k^{(t)}}h_k^{(t)}\boldsymbol{g}_k^{(t)}}{\xi^{(t)}\|\boldsymbol{g}_k^{(t)}\|}+\frac{\boldsymbol{w}_{\text{eff}}^{(t)}}{\xi^{(t)}}-\sum_{k=1}^{K}\frac{\boldsymbol{g}_k^{(t)}}{\|\boldsymbol{g}_k^{(t)}\|}\right\|^2\right], \quad (25)
\end{aligned}$$

while satisfying the differential privacy condition in (23). In (25), the expectation is over the distribution of the effective noise $\boldsymbol{w}_{\text{eff}}$ and $\xi^{(t)}$ is the normalization factor applied by the parameter server for recovering the gradient of interest. We assume that the edge devices are subject to a peak power constraint, i.e.,

$$(1+\kappa_k)\rho_k^{(t)}\le\rho_{\max}. \quad (26)$$

Accordingly, the relevant optimization problem can be defined as

$$\begin{aligned}
&\underset{\rho_k^{(t)}}{\text{minimize}} \quad \mathrm{MSE}^{(t)} \quad (27)\\
&\text{subject to} \quad (23) \text{ and } (26).
\end{aligned}$$

To guarantee a perfectly unbiased global model, the power scaling factors should be set such that the following gradient

alignment condition is satisfied,

$$\sqrt{\rho_k^{(t)}}\left|h_k^{(t)}\right| = \lambda^{(t)}. \tag{28}$$

In this case, by setting the normalization factor $\xi^{(t)} = \lambda^{(t)}$, the mean squared error in (25) simplifies to

$$\text{MSE}^{(t)} = \mathbb{E}\left[\left\|\frac{\boldsymbol{w}_{\text{eff}}^{(t)}}{\lambda^{(t)}}\right\|^2\right] = \frac{N_0 + \sum_{k=1}^{K}\kappa_k\lambda^{(t)}}{K^2(\lambda^{(t)})^2}. \tag{29}$$

When no privacy constraint is in place, the optimal solution for minimizing (29) under the power constraint in (26) is given by [22]

$$\tilde{\rho}_k^{(t)} = \frac{\rho_{\max}\left|h_j^{(t)}\right|^2}{(1+\kappa_j)\left|h_k^{(t)}\right|^2}, \tag{30}$$

where j is the index of the weakest channel, i.e.,

$$j = \arg\min_k \left|h_k^{(t)}\right|. \tag{31}$$

In words, the user with the weakest channel performs a full power transmission while the others apply power control with channel inversion to satisfy (28) using the resulting $\tilde{\lambda}^{(t)} = \sqrt{\rho_{\max}\left|h_j^{(t)}\right|^2/(1+\kappa_j)}$. Note that when differential privacy is enforced, due to the constraint (23), $\tilde{\lambda}^{(t)}$ may no longer be a feasible solution. In this case, noting that the privacy violation probability is an increasing function of $\lambda^{(t)}$, the value of $\tilde{\lambda}^{(t)}$ should be decreased until the constraint (23) is satisfied. Denote the maximum values of $\lambda^{(t)}$ satisfying (23) by $\lambda_{\text{p}}^{(t)}$. The solution to the optimization problem in (27) can then be stated as

$$\breve{\rho}_k^{(t)} = \frac{\left|\breve{\lambda}^{(t)}\right|^2}{\left|h_k^{(t)}\right|^2}, \tag{32}$$

where

$$\breve{\lambda}^{(t)} = \min\left\{\tilde{\lambda}^{(t)}, \lambda_{\text{p}}^{(t)}\right\}, \tag{33}$$

where $\lambda_{\text{p}}^{(t)}$ can be obtained using a simple line search.

V. Numerical Results

In this section, we evaluate the performance of the proposed privacy-preserving power allocation scheme under different hardware impairment levels. We consider the learning task of handwritten digit recognition using the well-known MNIST dataset, which consists of 60000 training samples and 10000 test samples. Each example is a 28×28 pixel gray scale image. The training samples are evenly distributed among the K devices. As a baseline classification model, we consider a feedforward neural network with a single hidden layer containing 100 hidden units followed by a softmax output layer. We use Adam as the optimizer with an initial learning rate of $\eta = 0.001$; the optimizer automatically adjusts the learning rate as the number of learning iterations increases to achieve faster convergence. We assume that the number of local steps an edge device performs before transmitting an update is equal to 30. The batch size is set to 128.

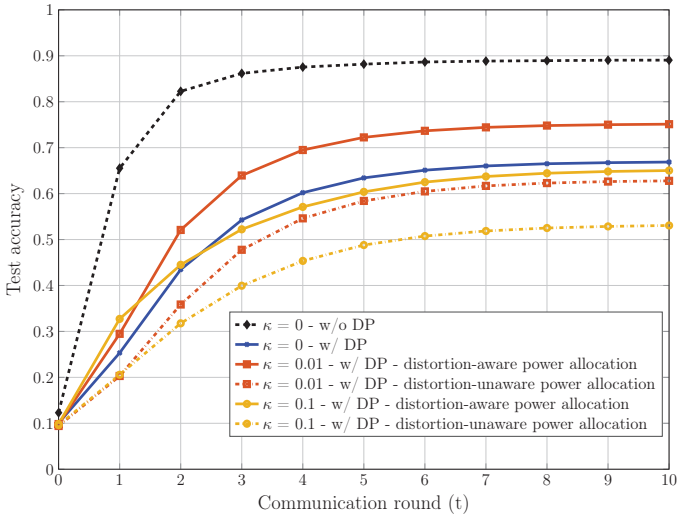

Fig. 2. Test accuracy for different power allocation strategies under several hardware impairment levels.

We consider a scenario with $K = 50$ edge devices and set the number of communication rounds to $T = 10$. The channel coefficients $h_k^{(t)}$ are drawn from a complex circularly-symmetric Gaussian distribution. The peak power constraint is set to $\rho_{\max} = 10\,\text{dBm}$ for all devices. The receiver noise variance is $N_0 = -20\,\text{dBm}$ and the differential privacy parameters are set to $\epsilon = 25$ and $\delta = 0.05$. Finally, we assume that the proportionality coefficient is equal for all edge devices, i.e., $\kappa_k = \kappa$ for $k = 1, \ldots, K$. We consider three different hardware impairment levels, $\kappa \in \{0, 0.01, 0.1\}$. Note that $\kappa = 0$ corresponds to a scenario with ideal hardware, whereas $\kappa = 0.01$ and $\kappa = 0.1$ represent cases with low-to-moderate and high levels of hardware impairment, respectively.[2] In each of these scenarios, we evaluate the learning performance versus the communication round. As benchmarks, we also include the results for the cases with distortion-unaware power allocation (where the effect of hardware impairments is ignored and the power allocation is carried out considering $\kappa = 0$ regardless of its actual value) and the case with ideal hardware and no privacy constraint.

Fig. 2 depicts the test accuracy, averaged over 50 independent trials with random splitting of the dataset in each trial, for the cases mentioned above. From the figure, it can be seen that for ideal hardware, enforcing the privacy constraint leads to a significant degradation in learning performance. This is because in this case, receiver noise is the only privacy-

[2]The proportionality coefficient κ is related to the transmit error vector magnitude (EVM) as EVM $= \sqrt{\kappa}$ [17]. Accordingly, $\kappa = 0.01$ and $\kappa = 0.1$ correspond to 10% and 31.62% EVM values, respectively. Typical EVM values can be derived from the minimum requirements specified in established standards.

preserving mechanism, and in order to satisfy the differential privacy constraint, devices must scale down their transmit power. However, by adopting the proposed power allocation schemes in realistic scenarios with imperfect hardware, the hardware-induced distortion at the devices can be used together with the receiver noise to satisfy the privacy requirements with a lower penalty in terms of learning performance. The proposed power allocation strategy can, therefore, achieve considerable performance gains compared to the conventional strategies that ignore the effect of hardware-induced distortion. In certain cases, the imperfect hardware devices can even achieve better performance than the ideal hardware scenario. An example of such gain attained by utilizing hardware-induced distortion can be seen for the case with $\kappa = 0.01$ (red curves with square markers). In particular, with $\kappa = 0.01$, the devices can transmit more power and yet guarantee the privacy preserving condition in (23), resulting in improved learning performance despite the slightly increased effective noise variance.

It should be noted that whether the imperfect hardware yields an improved performance with respect to the ideal hardware depends on the level of impairments as well as the stringency of the privacy requirements. For instance, in the example given in Fig. 2, it can be seen that the case with $\kappa = 0.1$ (yellow curves with circle markers) yields a degraded performance with respect to the ideal hardware. The reason behind this is that the amount of distortion injected into the transmitted updates is more than enough for satisfying the differential privacy requirements. As a result, the amount of transmit power at the devices is determined by the peak power constraint in (26) and the excessive distortion degrades the learning performance. Adopting a truncation-based approach by excluding the edge devices that experience deep fading channels similar to the solution in [4], can reduce the amount of excessive distortion. A detailed investigation of this approach is left for future research.

VI. Conclusion

We studied differentially-private wireless federated learning in realistic scenarios where edge devices are equipped with imperfect hardware. By modeling the hardware-induced distortion as power-dependent additive white Gaussian noise, we derived an expression for the privacy preservation condition and proposed a power allocation scheme to satisfy it. Our numerical results indicate that exploiting the inherent distortion considerably reduces the performance loss caused by the enforcement of differential privacy constraints. As a result, our proposed power allocation scheme achieves significant gains in terms of learning performance compared to conventional approaches that ignore the effect of hardware-induced distortion.

References

[1] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Commun. Surv. Tut.*, vol. 22, no. 3, pp. 2031–2063, third quarter 2020.

[2] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *arXiv preprint arXiv:1912.04977*, 2019.

[3] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, Jan. 2020.

[4] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 491–506, Jan. 2020.

[5] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, May 2020.

[6] M. M. Amiri, T. M. Duman, D. Gündüz, S. R. Kulkarni, and H. V. Poor, "Blind federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5129–5143, Aug. 2021.

[7] B. Tegin and T. M. Duman, "Machine learning at wireless edge with OFDM and low resolution ADC and DAC," *arXiv preprint arXiv:2010.00350*, 2020.

[8] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, Oct. 2015, pp. 1322–1333.

[9] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2019, pp. 691–706.

[10] H. Yin, A. Mallya, A. Vahdat, J. M. Alvarez, J. Kautz, and P. Molchanov, "See through gradients: Image batch recovery via gradinversion," in *Proc. IEEE/CVF Conf. Computer Vision Pattern Recognition*, Jun. 2021, pp. 16 337–16 346.

[11] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, "Inverting gradients–how easy is it to break privacy in federated learning?" *arXiv preprint arXiv:2003.14053*, 2020.

[12] F. Mo, A. Borovykh, M. Malekzadeh, H. Haddadi, and S. Demetriou, "Quantifying information leakage from gradients," *arXiv preprint arXiv:2105.13929*, 2021.

[13] M. Seif, R. Tandon, and M. Li, "Wireless federated learning with local differential privacy," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Los Angeles, CA, USA, Jun. 2020, pp. 2604–2609.

[14] 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 Trans. Inf. Forensics Security*, vol. 15, pp. 3454–3469, Apr. 2020.

[15] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, "Differentially private AirComp federated learning with power adaptation harnessing receiver noise," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Taipei, Taiwan, Dec. 2020, pp. 1–6.

[16] D. Liu and O. Simeone, "Privacy for free: Wireless federated learning via uncoded transmission with adaptive power control," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 170–185, Jan. 2021.

[17] E. Björnson, J. Hoydis, M. Kountouris, and M. Debbah, "Massive MIMO systems with non-ideal hardware: Energy efficiency, estimation, and capacity limits," *IEEE Trans. Inf. Theory*, vol. 11, no. 60, pp. 7112–7139, Nov. 2014.

[18] T. Schenk, *RF imperfections in high-rate wireless systems: Impact and digital compensation*. Springer Science & Business Media, 2008.

[19] C. Studer, M. Wenk, and A. Burg, "MIMO transmission with residual transmit-RF impairments," in *Proc. Int. ITG Workshop on Smart Antennas (WSA)*, Bremen, Germany, Feb. 2010, pp. 189–196.

[20] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," *IEEE Trans. Signal Process.*, vol. 68, pp. 2155–2169, Mar. 2020.

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

[22] X. Cao, G. Zhu, J. Xu, and K. Huang, "Optimized power control for over-the-air computation in fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7498–7513, Aug. 2020.