# Compressed Private Aggregation for Scalable and Robust Federated Learning over Massive Networks

Natalie Lang, *Graduate Student Member, IEEE,* Nir Shlezinger, *Senior Member, IEEE,*
Rafael G. L. D'Oliveira, *Member, IEEE,* and Salim El Rouayheb, *Senior Member, IEEE*

**Abstract**—Federated learning (FL) is an emerging paradigm that allows a central server to train machine learning models using remote users' data. Despite its growing popularity, FL faces challenges in preserving the privacy of local datasets, its sensitivity to poisoning attacks by malicious users, and its communication overhead, especially in large-scale networks. These limitations are often individually mitigated by local differential privacy (LDP) mechanisms, robust aggregation, compression, and user selection techniques, which typically come at the cost of accuracy. In this work, we present *compressed private aggregation (CPA)*, allowing massive deployments to simultaneously communicate at extremely low bit rates while achieving privacy, anonymity, and resilience to malicious users. CPA randomizes a codebook for compressing the data into a few bits using nested lattice quantizers, while ensuring anonymity and robustness, with a subsequent perturbation to hold LDP. CPA-aided FL is proven to converge in the same asymptotic rate as FL without privacy, compression, and robustness considerations, while satisfying both anonymity and LDP requirements. These analytical properties are empirically confirmed in a numerical study, where we demonstrate the performance gains of CPA compared with separate mechanisms for compression and privacy, as well as its robustness in mitigating the harmful effects of malicious users.

**Index Terms**—Federated learning, local differential privacy, anonymity, compression.

## 1 Introduction

The unprecedented success of deep learning highly relies on the availability of data, often gathered by edge devices such as mobile phones, sensors, and vehicles. As data may be private, there is a growing need to avoid leakage of private data while still being able to use it to train machine learning models. *Federated learning (FL)* [2], [3], [4], [5] is an emerging paradigm for training a model on multiple edge devices, exploiting their computational capabilities [6]. FL avoids directly sharing the users' data, as training is performed locally with periodic centralized aggregations of the models orchestrated by a server.

Learning in a federated manner is subject to several core challenges that are not encountered in traditional centralized machine learning [4], [5]. Despite the training being performed locally and not involving data sharing, it was recently shown that private information can be extracted and that the data can even be reconstructed from the exchanged models updates by model inversion attacks, if these are not properly protected [7], [8], [9], [10]. Furthermore, the fact that training is done on the users' side indicates that malicious users can affect the learned model by, for example, poisoning attacks [11], [12]. Another prominent challenge stems from the repeated exchange of highly parameterized models between the server and the devices during the FL procedure. As the communication links are possibly rate-limited channels, FL can notably load the communication infrastructure, which, in turn, often results in considerable delays and degraded convergence [13], [14]. These usually become more dominant in large-scale FL networks, causing significant overhead as well as yielding notable computational burden on the server side that recovers and aggregates the individual models, particularly when the number of users is huge, with possibly millions of participants.

Various methods have been proposed to tackle the above challenges: to preserve privacy, the local differential privacy (LDP) framework is commonly adopted. LDP quantifies privacy leakage of a single data sample when some function of the local datasets, e.g., a trained model, is publicly available [15]. LDP can be boosted by corrupting the model updates with privacy preserving noise (PPN) [16], via splitting/shuffling [17] or dimension selection [18]. An alternative privacy regime considered in FL is $k$-anonymity, which involves mechanisms that render certain features indistinguishable [19], [20]. Both LDP and $k$-anonymity mechanisms induce some level of perturbation that typically affects the learning procedure. Considering the difficulty of dealing with unreliable and malicious users, this issue is typically addressed by Byzantine robust methods [21], [22], [23]. Such techniques have the servers use non-affine aggregation which reduces the sensitivity to outliers and thus limits the harmful effect of corrupted model updates; yet typically degrade performance in the absence of malicious users.

The communication overhead of FL is often relaxed by

- *N. Lang and N. Shlezinger are with the School of ECE, Ben-Gurion University of the Negev, Be'er-Sheva, Israel (e-mails: langn@post.bgu.ac.il; nirshl@bgu.ac.il).*
- *R. G. L. D'Oliveira is with the School of Mathematical and Statistical Sciences, Clemson University, SC (e-mail: rdolive@clemson.edu).*
- *S. El Rouayheb is with the Department of ECE, Rutgers University, Piscataway, NJ (e-mail: salim.elrouayheb@rutgers.edu).*

*Parts of this work were presented in the 2023 IEEE International Conference on Acoustics Speech, and Signal Processing (ICASSP) as the paper [1].*

reducing the volume of model updates via lossy compression. This can be achieved by having each user transmit only part of its updates by sparsifying or sub-sampling [24], [25], [26], [27], [28], [29]. An alternative approach discretizes the updates of the model through quantization, so that it is conveyed using a small number of bits [30], [31], [32], [33], [34]. Scalability is typically enabled by limiting the number of participating devices through user selection [14]. These methods determine which of the users participate in each round of training, taking into account the individual constraints of computation and communication resources [35], as well as the magnitude of local updates [14].

Recent studies consider both the challenges of compression and privacy in FL. The works [36], [37] propose to quantize the local gradient with a differentially private 1-bit compressor; [38], [39], [40] employ probabilistic quantizers to achieve compression in a manner that also enhances privacy, so that the incorporation of a dedicated PPN can result in the compressed representation obeying established multivariate LDP mechanisms [39]. All these schemes have the server separately recover the model updates for each user and then aggregate via conventional averaging. Thus, they are neither inherently scalable to suit massive systems and tolerate large groups of colluding users, nor account for robustness considerations.

In this work, we present a novel privacy preserving scheme designed for robust large-scale FL. The method, coined *CPA*, dramatically reduces communications by conveying model updates via messages of only a few bits, while providing $k$-anonymity and LDP, as well as limiting the individual contribution of each user to increase robustness to malicious users. Unlike existing FL techniques, CPA jointly provides compression, proven privacy, inherent scalability, and empirically observed Byzantine robustness, without limiting learning capabilities, as summarized in Table 1. It is inspired by private multi-group aggregation [41] and geo-indistinguishability [42], which involve settings that fundamentally differ from FL in their task, yet inherently employ massive systems where scalability and robustness are key factors.

We design CPA by leveraging nested lattice quantizers [43] combined with random codebooks to encode the set of model updates into few bits at each user. The discretizing operation of the quantizers is exploited to provide anonymity, and is then further perturbed by incorporating an established randomized response (RR) mechanism. We analytically show that the resulting representation conveyed by each user rigorously holds both $k$-anonymity and LDP guarantees, and empirically demonstrate that it utterly limits each user's influence and leads to robustness from different forms of corrupted models.The conveyed few-bit representations are aggregated by the server via a decoding procedure, translating the received bits from all different users into an empirical discrete histogram over the model update values.

The aggregated mean of this histogram is shown to converge into the averaged global trained model, yielding the desired updated global model in each FL iteration. By doing so, the server does not reconstruct the individual model updates, which, when combined with the few-bit communication involved in CPA, notably facilitates the participation of numerous users and supports scalability. Furthermore, we systematically show that the distortion in the resulting aggregated model compared to vanilla FL (without communication, privacy, or security considerations) decreases as the number of users increases, and that the resulting model converges in the same asymptotic order as vanilla FL. These theoretical findings are numerically demonstrated in our experimental study. There, we evaluate CPA for learning several different image classification models, showing that its overall distortion is reduced compared to conventional methodologies for private compressed FL, and that this reduced distortion is translated into an improved performance of the learned model.

Our main contributions are summarized as follows:

- **Novel scalable aggregation technique:** CPA presents a joint design of probabilistic model quantization and users 'voting' for private aggregation. The perturbation introduced therein to meet LDP, is mitigated not by the conventional federated averaging (FedAvg) but rather by a unique reconstruction of discrete histograms, having the server avoids recovering the individual updates for each user. While this allows applicability over large-scale FL systems, it also guarantees $k$-anonymity by design.
- **Byzantine robustness following user's low-influence:** CPA exploits the high-dimensional structure of the model updates through (possibly high-rate) lattice quantization but still dramatically reduces the conventional FL communication overhead. This follows since the users transmit at most $B$ bits per sample, which inherently limits their influence on the final model and allows the training to be Byzantine robust against erroneous adversarial users and poisoning attacks.
- **Theoretical and experimental evaluation:** The ideas introduced in CPA draw inspiration from previous studies in different domains on model compression and private aggregation. The novelty and contribution of CPA relies on coupling and fusing these parallel domains in a noise-controllable manner. The ability to learn reliably in large scale networks systematically exemplified for CPA in both analytical and extensive numerical analysis.

The remainder of this paper is organized as follows: Section 2 briefly reviews the FL system model and the relevant preliminaries. CPA is presented in Section 3, while Section 4 theoretically analyzes its privacy guarantees and convergence profile. In Section 5 we numerically evaluate CPA. Finally, Section 6 provides concluding remarks.

Notations: throughout this paper, we use boldface lowercase letters for vectors, e.g., $\boldsymbol{x}$, boldface uppercase letters for matrices, e.g., $\boldsymbol{X}$, and calligraphic letters for sets, e.g., $\mathcal{X}$. The stochastic expectation, variance, and $\ell_2$ norm are denoted by $\mathbb{E}[\cdot]$, $\mathrm{Var}(\cdot)$, and $\|\cdot\|$, respectively, while $\mathbb{Z}$ and $\mathbb{R}$ are the sets of integer and real numbers, respectively.

## 2 System Model and Preliminaries

In this section we present the system model of bit-constrained and private FL. We begin by recalling some relevant basics in FL and quantization in Subsections 2.1-

TABLE 1
Comparison between compressed private aggregation (CPA) and existing FL studies

| | Compression | Proven Privacy | | Security | Scalability |
|---|---|---|---|---|---|
| | | LDP | Anonymity | | |
| Byzantine robust aggregation, e.g., [21], [22], [23] | ✗ | ✗ | | ✓ | ✗ |
| User selection, e.g., [14], [35] | ✗ | ✗ | | ✗ | ✓ |
| Model updates compression, e.g., [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34] | ✓ | ✗ | | ✗ | ✗ |
| Privacy enhancement, e.g., [15], [16], [17], [18] | ✗ | ✓ | ✗ | ✗ | ✗ |
| Joint compression & privacy, e.g., [36], [37], [38], [39], [40] | ✓ | ✓ | ✗ | ✗ | ✗ |
| CPA | ✓ | ✓ | ✓ | ✓ | ✓ |

2.2 respectively. We then review the privacy preliminaries in Subsection 2.3, and formulate our problem in Subsection 2.4.

### 2.1 Federated Learning

In FL, a server trains a machine learning model parameterized by $\boldsymbol{w} \in \mathbb{R}^d$ using data available at a group of $K$ users indexed by $1, \ldots, K$. These datasets, denoted $\mathcal{D}_1, \ldots, \mathcal{D}_K$, are assumed to be private. Thus, as opposed to conventional centralized learning where the server can use $\mathcal{D} = \bigcup_{r=1}^{K} \mathcal{D}_r$ to train $\boldsymbol{w}$, in FL the users cannot share their data with the server. Let $F_r(\boldsymbol{w})$ be the empirical risk of a model $\boldsymbol{w}$ evaluated over the dataset $\mathcal{D}_r$. The training goal is to recover the $d \times 1$ optimal weights vector $\boldsymbol{w}^{\rm opt}$ satisfying

$$\boldsymbol{w}^{\rm opt} = \mathop{\arg\min}_{\boldsymbol{w}} \left\{ F(\boldsymbol{w}) \triangleq \frac{1}{K} \sum_{r=1}^{K} F_r\left(\boldsymbol{w}\right) \right\}. \tag{1}$$

Generally speaking, FL involves the distribution of a global model to the users. Each user locally trains this model using its own data and sends back the model update [5]. Therefore, users do not directly expose their private data, as training is performed locally. The server then aggregates the models into an updated global model and the procedure repeats iteratively.

Arguably, the most common FL scheme is *FedAvg* [2], where the global model is updated by averaging the local models. Letting $\boldsymbol{w}_t$ denote the global parameters vector available at the server at time step $t$, the server shares $\boldsymbol{w}_t$ with the users, who each performs $\tau$ training iterations using its local $\mathcal{D}_r$ to update $\boldsymbol{w}_t$ into $\boldsymbol{w}^r_{t+\tau}$. Typically, the information conveyed from the users to the server is not the model weights, i.e., $\boldsymbol{w}^r_{t+\tau}$, but the updates to the model generated in the current round, i.e., $\boldsymbol{h}^r_{t+\tau} \triangleq \boldsymbol{w}^r_{t+\tau} - \boldsymbol{w}_t$. As the server knows $\boldsymbol{w}_t$, it recovers $\boldsymbol{w}_{t+\tau}$ from the difference $\boldsymbol{w}^r_{t+\tau} - \boldsymbol{w}_t$. The server in turn sets the global model to be

$$\boldsymbol{w}_{t+\tau} \triangleq \boldsymbol{w}_t + \frac{1}{K} \sum_{r=1}^{K} \boldsymbol{h}^r_{t+\tau} = \frac{1}{K} \sum_{r=1}^{K} \boldsymbol{w}^r_{t+\tau}, \tag{2}$$

where it is assumed for simplicity that all $K$ users participate in each FL round. The updated global model is again distributed to the users and the learning procedure continues.

When the local optimization at the users' side is carried out using stochastic gradient descent (SGD), then FedAvg applies the *local-SGD* method [44]. In this case, each user of index $r$ sets $\boldsymbol{w}^r_t = \boldsymbol{w}_t$, and updates its local model via

$$\boldsymbol{w}^r_{t+1} \leftarrow \boldsymbol{w}^r_t - \eta_t \nabla F_r^{j^r_t}\left(\boldsymbol{w}^r_t\right), \tag{3}$$

where $j^r_t$ is the sample index chosen uniformly from $\mathcal{D}_r$, $\eta_t$ is the learning rate, and $F_r^{j^r_t}(\cdot)$ is the empirical risk computed using the $j^r_t$-th sample in $\mathcal{D}_r$. As sharing $\boldsymbol{w}^r_{t+\tau}$ can possibly load the communication network and leak private information, it motivates the integration of quantization and privacy enhancement techniques, discussed below.

### 2.2 Quantization Preliminaries

Vector quantization is the encoding of a set of continuous-amplitude quantities into a finite-bit representation [45]. Vector quantizers which are invariant of the underlying distribution of the vector to be quantized are referred to as *universal vector quantizers*; a leading approach to implement such quantizers is based on lattice quantization [46]:

**Definition 2.1** (Lattice Quantizer). *A lattice quantizer of dimension $L \in \mathbb{Z}^+$ and generator matrix $\boldsymbol{G} \in \mathbb{R}^{L \times L}$ maps $\boldsymbol{x} \in \mathbb{R}^L$ into a discrete representation $Q_{\mathcal{L}}(\boldsymbol{x})$ by selecting the nearest point in the lattice $\mathcal{L} \triangleq \{\boldsymbol{G}\boldsymbol{l} : \boldsymbol{l} \in \mathbb{Z}^L\}$, i.e.,*

$$Q_{\mathcal{L}}(\boldsymbol{x}) = \mathop{\arg\min}_{\boldsymbol{z} \in \mathcal{L}} \|\boldsymbol{x} - \boldsymbol{z}\|. \tag{4}$$

To apply $Q_{\mathcal{L}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^{ML}$, it is divided into $\left[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M\right]^T$, and each sub-vector is quantized separately.

A lattice $\mathcal{L}$ partitions $\mathbb{R}^L$ into cells centered around the lattice points, where the basic cell is $\mathcal{P}_0 = \{\boldsymbol{x} : Q_{\mathcal{L}}(\boldsymbol{x}) = \boldsymbol{0}\}$. The number of lattice points in $\mathcal{L}$ is countable but infinite. Thus, to obtain a finite-bit representation, it is common to restrict $\mathcal{L}$ to include only points in a given sphere of radius $\gamma$, $\mathcal{L}_\gamma$, and the number of lattice points, $|\mathcal{L}_\gamma|$, dictates the number of bits per sample – $R \triangleq \frac{1}{L}\log_2(|\mathcal{L}_\gamma|)$. An event in which the input to the lattice quantizer does not reside in this sphere is referred to as *overloading*, from which quantizers are typically designed to avoid [45]. In the special case of $L = 1$ with $\boldsymbol{G} = \Delta_{\rm Q} > 0$, $Q_{\mathcal{L}}(\cdot)$ specializes conventional scalar uniform quantization $Q(\cdot)$.

**Definition 2.2** (Uniform Quantizer). *A mid-tread scalar uniform quantizer with support $\gamma$ and spacing $\Delta_{\rm Q}$ is defined as*

$$Q(x) = \begin{cases} \Delta_{\rm Q} \left\lfloor \frac{x}{\Delta_{\rm Q}} + \frac{1}{2} \right\rfloor & \text{if } x < |\gamma|, \\ \operatorname{sign}(x) \cdot \left(\gamma - \frac{1}{2}\Delta_{\rm Q}\right) & \text{otherwise} \end{cases} \tag{5}$$

*where $R = \log_2\left(2\gamma/\Delta_{\rm Q}\right)$ bits are used to represent $x$.*

The formulation of lattice and uniform quantizers in Defs. 2.1-2.2 gives rise to two extensions, which are adopted in the sequel. The first is *Probabilistic quantization*, which converts the quantizers to implement stochastic mapping. A conventional probabilistic quantization technique uses dithered quantization (DQ), which applies $Q_{\mathcal{L}}$ to a noisy

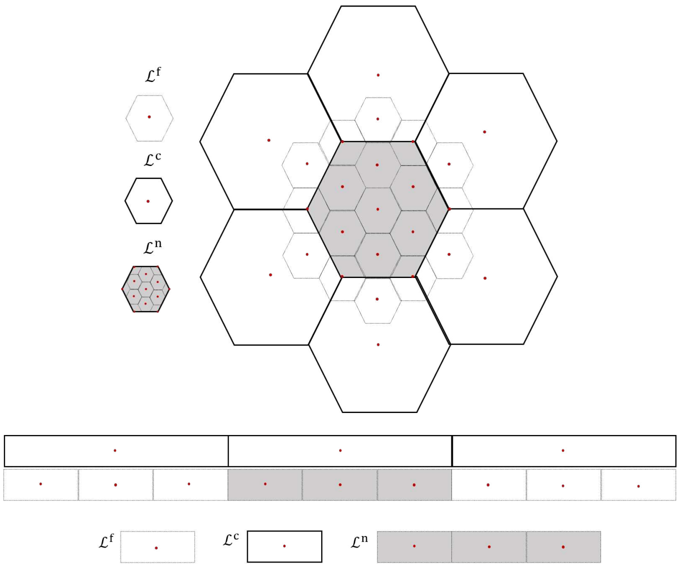
$\mathcal{L}^{\mathrm{f}}$
$\mathcal{L}^{\mathrm{c}}$
$\mathcal{L}^{\mathrm{n}}$
$\mathcal{L}^{\mathrm{f}}$
$\mathcal{L}^{\mathrm{c}}$
$\mathcal{L}^{\mathrm{n}}$

Fig. 1. Nested self-similar lattice quantizers for $L = 2$ (top) and $L = 1$ (bottom).

version of the input [47], [48]. When the added noise is uniformly distributed over $\mathcal{P}_0$ and the quantizer is not overloaded, the resulting distortion becomes an i.i.d. stochastic process [48], [49].

The second extension of lattice quantizers is nested quantization, which implements increased resolution quantization using multiple low-resolution quantizers. For simplicity, we now focus on nested quantization with two quantizers, as visualized in Fig. 1. The formal definition of a two-stage nested lattice quantizer is as follows [50].

**Definition 2.3** (Nested Lattice Codebook [51])**.** *A lattice $\mathcal{L}^{\mathrm{c}}$ is said to be nested in $\mathcal{L}^{\mathrm{f}}$ if $\mathcal{L}^{\mathrm{c}} \subset \mathcal{L}^{\mathrm{f}}$. Let $\mathcal{P}_0^{\mathrm{c}}$ denote the basic lattice cell of $\mathcal{L}^{\mathrm{c}}$, then a nested lattice codebook $\mathcal{L}^{\mathrm{n}}$ based on the nested lattice pair $\mathcal{L}^{\mathrm{c}} \subset \mathcal{L}^{\mathrm{f}}$ is defined as*

$$\mathcal{L}^{\mathrm{n}} \triangleq \mathcal{L}^{\mathrm{f}} \cap \mathcal{P}_0^{\mathrm{c}}. \tag{6}$$

A nested formulation allows quantizing with the fine lattice quantizer ($\mathcal{L}^{\mathrm{f}}$) using the nested ($\mathcal{L}^{\mathrm{n}}$) and the coarse ($\mathcal{L}^{\mathrm{c}}$) ones. In particular, one can quantize $\boldsymbol{x} \in \mathbb{R}^L$ by computing $Q_{\mathcal{L}^{\mathrm{f}}}(\boldsymbol{x}) = Q_{\mathcal{L}^{\mathrm{c}}}(\boldsymbol{x}) + Q_{\mathcal{L}^{\mathrm{n}}}\big(\boldsymbol{x} - Q_{\mathcal{L}^{\mathrm{c}}}(\boldsymbol{x})\big)$. Nested lattice quantizers naturally specialize multi-bit scalar uniform quantization [43], where the nesting condition implies that the quantization spacing of the coarse quantizer must be an integer multiple of the corresponding spacing of the fine quantizer. While Def. 2.3 is given for a two-stage quantizer, i.e., $\mathcal{L}^{\mathrm{f}}$ is implemented using two quantizers, it can be recursively extended into multiple stages by quantizing over $\mathcal{L}^{\mathrm{n}}$ in a nested fashion.

## 2.3 Privacy Preliminaries

Privacy in settings involving queries between users and a server is commonly quantified in terms of differential privacy (DP) [52], [53] and LDP [54], [55]. While both provide users with privacy guarantees from untruthful adversaries, the latter does not assume a trusted third-party server, and is thus commonly adopted in FL [15], [17], [18], [39], [55], [56], [57]. Therefore, we consider LDP in this work, defined below.

**Definition 2.4** ($\varepsilon$-LDP [58])**.** *A randomized mechanism $\mathcal{M}$ satisfies $\varepsilon$-LDP if for any pairs of input values $v, v'$ in the domain of $\mathcal{M}$ and for any possible output $y$, it holds that*

$$\Pr[\mathcal{M}(v) = y] \leq e^{\varepsilon} \Pr[\mathcal{M}(v') = y]. \tag{7}$$

We note that a smaller $\varepsilon$ means greater protection of privacy. Def. 2.4 implies that privacy can be achieved by stochasticity: if two different inputs are probable (up to some privacy budget) to be associated with the same algorithm output, then privacy is preserved, as each sample is not uniquely distinguishable.

For continuous quantities, common mechanisms that achieve $\varepsilon$-LDP are widely based on perturbation with privacy preserving noise (PPN), e.g., Laplacian or multivariate $t$ [59]. The PPN distribution parameters set to meet the LDP privacy level $\varepsilon$. For private binary queries, a principle method for achieving $\varepsilon$-LDP is the *RR mechanism* [60]. In RR, a user who possesses a private bit transmits it correctly with probability $p > 1/2$. By (7), it can be shown that RR satisfies $\log\left(\frac{p}{1-p}\right)$-LDP [58] and can be viewed as a PPN mechanism.

Although LDP is a preferable privacy measure, it is often guaranteed by the introduction of a dominant PPN perturbations. Alternative privacy measures, which are not inherently bundled with stochasticity, are based on anonymization [58] such as $k$-anonymity [61]:

**Definition 2.5** ($k$-anonymity [61])**.** *A deterministic mechanism $\mathcal{M}$ holds $k$-anonymity if for every input $v$ in the domain of $\mathcal{M}$ there are at least $k-1$ different inputs $\{v_i'\}_{i=1}^{k-1}$ satisfying*

$$\mathcal{M}(v) = \mathcal{M}(v_i'), \qquad \forall i \in \{1, \ldots, k-1\}. \tag{8}$$

If $\mathcal{M}$ satisfies $k$-anonymity, any observer of $\mathcal{M}$'s output is unable to discriminate between at least $k$ possible inputs.

## 2.4 Problem Description

### 2.4.1 Threat Model

FL was shown to be exploitable by adversaries, with various possible attacks and threat models [62]. Here, we focus on two types of threats. The first is privacy attacks, i.e., algorithms that reconstruct the raw original private data, based on unintentional information leakage regarding the data or the machine learning model, being a unique characteristic of FL. Inspired by [63], we investigate an *honest-but-curious server* with the goal of uncovering the users' data. The attacker is allowed to separately store and process updates transmitted by individual users, but may not interfere with the learning algorithm. The attacker may not modify the model architecture nor may it send malicious global parameters that do not represent the actual global learned model.

An additional threat considered is that of *adversarial participants*, often assumed in Byzantine robust FL [22]. Under this model, an unknown subset of the participating users may convey corrupted model updates, via poisoning attacks [11]. The identity of the unreliable users is not known to the server nor to the remaining reliable participants.

### 2.4.2 Problem Formulation

Our goal is to design a global aggregation mechanism [5] for FL that provides privacy guarantees, compression, robustness, and is scalable. In particular, we are interested in obtaining a mapping $\boldsymbol{h}_t^r \mapsto \boldsymbol{w}_t$ of the local updates at the $r$-th user into the global model available at the server. The scheme must be:

*R1* *Private*: holding $k$-anonymity and $\varepsilon$-LDP with respect to the private datasets $\{\mathcal{D}_r\}$, for a given anonymity degree $k$ and privacy budget $\varepsilon$, respectively.
*R2* *Compressed*: communications to the server should involve at most $B$ bits per sample.
*R3* *Universal*: invariant to the distribution of $\boldsymbol{h}_t^r$.
*R4* *Robust*: resilient to adversarial participants and tolerate a large group of colluding users.
*R5* *Scalable*: operable with possibly millions of participants.

In *R1* we focus on achieving LDP in each round of communication. One can use a per-round privacy level to formulate an overall privacy guarantee after a given amount of rounds via the composition theorem [64, Thm. III.1.], as the overall privacy level after $T$ rounds is at most $T \cdot \varepsilon$. Nevertheless, recent work has shown that by additional processing, a per-round privacy level can be translated into a multi-round one, obtaining an overall privacy budget depending on $\varepsilon$ that does not linearly grow with the number of rounds [17]. For these reasons, we formulate our privacy budget as in *R1*.

Evidently, requirements *R1-R3* can be satisfied by first perturbing the data to meet *R1*, followed by universal quantization to satisfy *R2-R3*, as both techniques are invariant to the distribution of $\boldsymbol{h}_t^r$. However, the server decoding in these separate schemes requires individual reconstruction, which may result in violating *R5* while not accounting for *R4*. Furthermore, both privacy and quantization can be modeled as corrupting the model updates, and thus using separate mechanisms may result in an overall noise which degrades the accuracy of the trained model beyond that needed to meet *R1-R3*. These observations motivate a joint design tailored for FL, studied next.

## 3 Compressed Private Aggregation

In this section we introduce CPA, deriving its basic steps for 1-bit messages in Subsection 3.1, and its extension to multi-bit messages via nested quantization in Subsection 3.2. Then, we provide a discussion in Subsection 3.3.

### 3.1 1-Bit CPA

We design CPA based on *R1-R5* by extending the recent schemes of [41] and [42] to FL settings. Broadly speaking, CPA leverages the repeated communications of FL to generate a random codebook and encode the data with the help of an $L$ dimension lattice quantizer (holding *R3*). The generated code enables the transmission of a set of $L$ model update entries with a single bit, i.e., $B = \frac{1}{L}$ (satisfying *R2*), which guarantees $k$-anonymity of the data. We then support LDP by applying RR to the transferred bits (satisfying *R1*). In the decoding procedure, the received bits are translated into an empirical histogram over the model update values, rather than recovering each model update separately (holding *R5*). The aggregated mean over this histogram converges into the FedAvg trained model, inherently limiting the influence of potential malicious participating users, as they can, at most, flip one bit (assuring *R4*). These steps, illustrated in Fig. 2 and summarized as Algorithm 1, are described below in detail.

#### 3.1.1 Initialization

To initialize CPA, the privacy parameters $k$ and $\varepsilon$ are set, and the compression lattice $\mathcal{L}$ is determined, i.e., fixing the dimension of the lattice $L$, its generator matrix $\boldsymbol{G}$, radius $\gamma$, and rate $R$ [65, Ch. 2]. We allow the lattice to change over the FL rounds, and thus denote it by $\mathcal{L}_t$. The motivation to do so is to allow the quantizer to adapt its mapping along the FL procedure, and particularly by gradually decreasing the dynamic range over time to better represent the model updates whose magnitude typically decreases as FL approaches converges. Additionally, a common seed $s_r$ is shared between each user and the server. This can be provided by the user along with the initial sharing of updates in the FL procedure, as done in, e.g., [33].

#### 3.1.2 Encoding

The CPA procedure is carried out on each FL global aggregation round. Therefore, we describe it for a given time step $t$, in which the users have updated their local models. Since the encoding is identical for all users, we focus on the $r$-th user, who is ready to transmit $\boldsymbol{h}_t^r$. In the encoding step, the model updates are compressed into a single bit using a quantizer and a random binary codebook, and then perturbed to enhance privacy. These steps are formulated as follows.

**Quantization:** To begin, $\boldsymbol{h}_t^r \in \mathbb{R}^d$ is decomposed into distinct vectors $\{\boldsymbol{h}_{t,i}^r\}_{i=1}^M$ such that

$$\boldsymbol{h}_{t,i}^r \in \mathbb{R}^L, \quad M \triangleq \left\lceil \frac{d}{L} \right\rceil; \tag{9}$$

and being quantized by applying an $L$-dimensional lattice quantizer (Def. 2.1) to each, i.e., $\boldsymbol{h}_{t,i}^r$ is mapped into $Q_{\mathcal{L}_t}(\boldsymbol{h}_{t,i}^r)$.

**1-Bit Compression:** Next, the discrete codeword is compressed into a single bit according to the index of the assigned lattice point. To this end, the seed $s_r$ is used to randomize a codeword $\boldsymbol{v}_{t,i}^r$, which is uniformly distributed over all words in $\{-1,1\}^{2^{LR}}$ having an equal amount of 1's and $-1$'s. Then, as illustrated in Fig. 2, the lattice point $Q_{\mathcal{L}_t}(\boldsymbol{h}_{t,i}^r)$ is represented by its index in the lattice, denoted $l$, and, in turn, a single bit $\bar{b}_{t,i}^r$ is set according to the $l$-th entry of the vector $\boldsymbol{v}_{t,i}^r$.

Formally, we write $Q_{\mathcal{L}_t}(\boldsymbol{h}_{t,i}^r) = \boldsymbol{q}^l$ where $\boldsymbol{q}^l \in \mathbb{R}^L$ is the $l$-th lattice point in $\mathcal{L}_t$. The bit that the user conveys to the server is selected based on $\left[\boldsymbol{v}_{t,i}^r\right]_l$, where

$$\bar{b}_{t,i}^r \triangleq \begin{cases} 1 & \text{if } \left[\boldsymbol{v}_{t,i}^r\right]_l = 1, \\ -1 & \text{otherwise.} \end{cases} \tag{10}$$

The resulting processing converts the $L$ dimensional model updates vector $\boldsymbol{h}_{t,i}^r$ into a single bit representation $\bar{b}_{t,i}^r$. This procedure is exemplified using a scalar quantizer in Fig. 3.

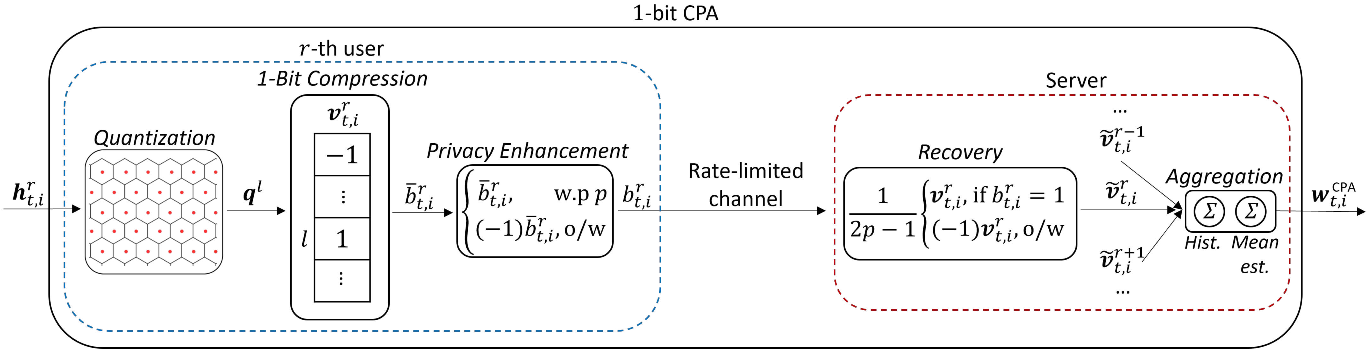
1-bit CPA
$r$-th user
1-Bit Compression
Quantization
$\boldsymbol{h}^r_{t,i}$
$\boldsymbol{q}^l$
$\boldsymbol{v}^r_{t,i}$
$-1$
$\vdots$
$l$
$1$
$\vdots$
$\bar{b}^r_{t,i}$
Privacy Enhancement
$\begin{cases}\bar{b}^r_{t,i}, & \text{w.p } p\\ (-1)\bar{b}^r_{t,i}, & \text{o/w}\end{cases}$
$b^r_{t,i}$
Rate-limited channel
Server
Recovery
$\frac{1}{2p-1}\begin{cases}\boldsymbol{v}^r_{t,i}, \text{if } b^r_{t,i}=1\\ (-1)\boldsymbol{v}^r_{t,i}, \text{o/w}\end{cases}$
$\tilde{\boldsymbol{v}}^{r-1}_{t,i}$
$\tilde{\boldsymbol{v}}^r_{t,i}$
$\tilde{\boldsymbol{v}}^{r+1}_{t,i}$
Aggregation
$\Sigma$ $\Sigma$
Hist. Mean est.
$\boldsymbol{w}^{\text{CPA}}_{t,i}$

Fig. 2. Overview of CPA. The left dashed box represents the $r$-th user encoding while the right describes the server decoding.

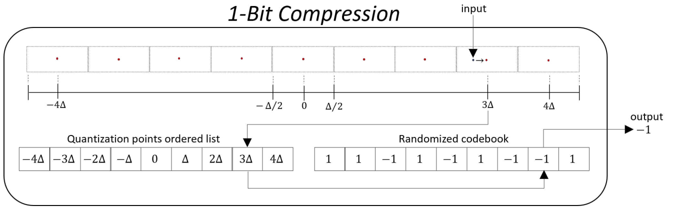

Fig. 3. Example: a scalar input is mapped into a point that corresponds to the continues vale of $3\Delta$, which is $9$-th quantization point. Accordingly, as the $9$-th entry of the randomized vector is $-1$, so does the output.

**Privacy Enhancement:** As we show in Section 4, $k$-anonymity (Def. 2.5) directly follows from the design of $\boldsymbol{v}^r_{t,i}$. To also maintain $\varepsilon$-LDP, RR is applied to $\bar{b}^r_{t,i}$. RR guarantees $\varepsilon$-LDP by having the true value of $\bar{b}^r_{t,i}$ conveyed with probability $p = \frac{e^\varepsilon}{1+e^\varepsilon}$ and its complement with $1-p$. Consequently, the bit which the sever receives as a representation of $\boldsymbol{h}^r_{t,i}$ is

$$b^r_{t,i} = \begin{cases} \bar{b}^r_{t,i} & \text{w.p. } p,\\ (-1)\cdot \bar{b}^r_{t,i} & \text{w.p. } 1-p. \end{cases} \tag{11}$$

### 3.1.3 Decoding

The decoding procedure avoids having the server reconstruct each individual model. Instead, the server uses the bits it receives corresponding to the $i$-th sub-vector of the model parameters to directly compute the desired aggregated model. This is achieved in two stages: first, the bits $\{b^r_{t,i}\}_{r=1}^K$ are recovered into unbiased estimates of their codewords $\{\boldsymbol{v}^r_{t,i}\}_{r=1}^K$, which are directly aggregated into an empirical *histogram*, used to update the global model.

**Recovery:** Due to the shared seed $s_r$, the server knows $\boldsymbol{v}^r_{t,i}$ and can thus associate each bit with its corresponding codeword. However, since the bits are perturbed by the RR mechanism, the server can only recover an estimate of the codeword. This is achieved by setting

$$\tilde{\boldsymbol{v}}^r_{t,i} = \frac{1}{2p-1}\begin{cases} \boldsymbol{v}^r_{t,i} & \text{if } b^r_{t,i}=1,\\ (-1)\cdot \boldsymbol{v}^r_{t,i} & \text{otherwise;} \end{cases} \tag{12}$$

where the weighting factor $\frac{1}{2p-1}$ assures that $\tilde{\boldsymbol{v}}^r_{t,i}$ is an unbiased estimator of $\boldsymbol{v}^r_{t,i}$.

**Aggregation:** The server then constructs with an aggregated mean of all $\{\tilde{\boldsymbol{v}}^r_{t,i}\}_{r=1}^K$, i.e.,

$$\tilde{\boldsymbol{v}}_{t,i} \triangleq \frac{1}{K}\sum_{r=1}^{K} \tilde{\boldsymbol{v}}^r_{t,i}. \tag{13}$$

Practically, $\tilde{\boldsymbol{v}}_{t,i}$ is a discrete normalized histogram. Aggregation through (13) can, in principle, yield negative histogram values, which can be kept or mitigated by thresholding [42].

The estimated histogram is utilized for updating the global model, replacing the conventional FedAvg update in (2) by

$$\boldsymbol{w}^{\text{CPA}}_{t,i} = \boldsymbol{w}^{\text{CPA}}_{t-\tau,i} + \sum_{l=1}^{2^R} [\tilde{\boldsymbol{v}}_{t,i}]_l \cdot \boldsymbol{q}^l. \tag{14}$$

The global model $\boldsymbol{w}^{\text{CPA}}_t$ is then obtained by stacking the sub-vectors $\{\boldsymbol{w}^{\text{CPA}}_{t,i}\}_{i=1}^M$.

**Algorithm 1:** 1-bit CPA at time step $t$

```
Initialization:
    Shared seed $s_r$, degree of anonymity $k$, privacy
     budget $\varepsilon$, and lattice $\mathcal{L}_t$;
Encode (at the $r$-th user side, for each $i$):
    Quantize $\boldsymbol{h}^r_{t,i}$ into $\boldsymbol{q}^l$, $l \in \{1,\dots,2^R\}$, using $Q_{\mathcal{L}_t}$;
    Set $\bar{b}^r_{t,i}$ using (10);
    Augment $\bar{b}^r_{t,i}$ into $b^r_{t,i}$ via (11), convey to server;
Decode (at the server side, for each $i$):
    Recover $\{\tilde{\boldsymbol{v}}^r_{t,i}\}_{r=1}^K$ via (12) ;
    Obtain an empirical histogram via (13);
    Update the global model $\boldsymbol{w}^{\text{CPA}}_{t,i}$ using (14);
Result: Updated $i$-th global model sub-vector, $\boldsymbol{w}_{t,i}$.
```

## 3.2 Nested CPA

Algorithm 1 is particularly designed to operate with a large number of users. Instead of recovering the individual model updates sub-vector of each user $\{\boldsymbol{h}^r_{t,i}\}$, the aggregation of $\tilde{\boldsymbol{v}}_{t,i}$ in (13) forms an estimate of the *distribution* of the quantized updates over the lattice $\mathcal{L}_t$. As shown in the proof of Theorem 4.1, in the horizon of an asymptotically large

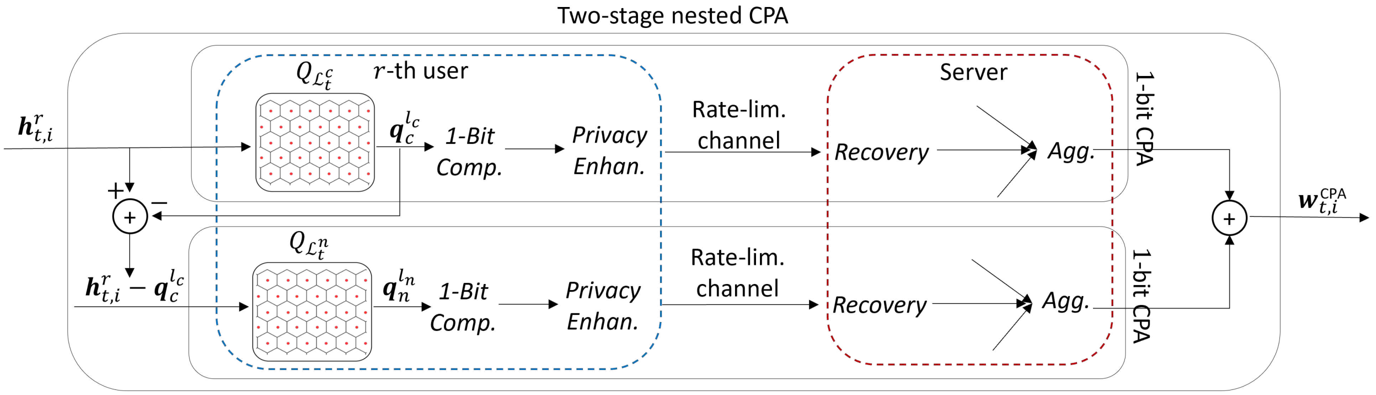

Fig. 4. Overview of nested CPA. The upper solid box represents the $1$-bit CPA with a coarse quantizer while the bottom describes that with the nested one.

number of users $K$, the mean value taken over $\tilde{\boldsymbol{v}}_{t,i}$ converges to the federated average of $\{Q_{\mathcal{L}_t}(\boldsymbol{h}_{t,i}^r)\}_{r=1}^K$ for any given lattice quantizer employed. This motivates the usage of quantizers with high rate $R$, for which the distortion in $Q_{\mathcal{L}_t}(\boldsymbol{h}_{t,i}^r)$ compared to $\boldsymbol{h}_{t,i}^r$ is small. However, for a finite number of users, a large number of lattice points typically results in a less accurate estimation of the probability over the lattice via $\tilde{\boldsymbol{v}}_{t,i}$, giving rise to a tradeoff between the number of users $K$ and the quantization rate $R$.

In order to alleviate this tradeoff, enabling CPA to operate reliably with high resolution lattice quantizers (large $R$), we propose to implement fine quantization using multiple low-rate quantizers via *nested quantization* (Def. 2.3). This allows constructing a separate histogram for each low-rate lattice quantizer, such that the mean over the fine lattice, i.e., the desired low-distorted averaged model, can be computed from these histograms. However, this comes at the cost of additional bits conveyed by the users, as each quantized value is no longer conveyed using a single bit as in Algorithm 1. We next formulate this form of nested CPA, focusing on a two-stage nested operation (i.e., with two bits $B = \frac{2}{L}$), which can be extended to multiple stages (and multiple bits) as discussed in Subsection 2.2. The general procedure is illustrated in Fig. 4.

### 3.2.1 Initialization

In addition to the initial steps of Algorithm 1, nested CPA divides the fine lattice $\mathcal{L}_t$ into a nested lattice $\mathcal{L}_t^{\text{n}}$ and a coarse lattice $\mathcal{L}_t^{\text{c}}$ (see Def. 2.3), with rates $R_{\text{n}}$ and $R_{\text{c}}$, respectively.

### 3.2.2 Encoding

As in Algorithm 1, the model updates are divided into $\{\boldsymbol{h}_{t,i}^r\}_{i=1}^M$. Here, the sub-vectors are quantized using the low-rate lattice quantizers, yielding

$$\begin{aligned} Q_{\mathcal{L}_t^{\text{c}}}(\boldsymbol{h}_{t,i}^r) &= \boldsymbol{q}_{\text{c}}^{l_{\text{c}}}, \\ Q_{\mathcal{L}_t^{\text{n}}}(\boldsymbol{h}_{t,i}^r - Q_{\mathcal{L}_t^{\text{c}}}(\boldsymbol{h}_{t,i}^r)) &= \boldsymbol{q}_{\text{n}}^{l_{\text{n}}}. \end{aligned} \tag{15}$$

The fact that $Q_{\mathcal{L}_t^{\text{c}}}(\cdot)$ and $Q_{\mathcal{L}_t^{\text{n}}}(\cdot)$ form a nested representation of the fine quantizer $Q_{\mathcal{L}_t^{\text{f}}}(\cdot)$ indicates that

$$Q_{\mathcal{L}_t^{\text{f}}}(\boldsymbol{h}_{t,i}^r) = \boldsymbol{q}_{\text{c}}^{l_{\text{c}}} + \boldsymbol{q}_{\text{n}}^{l_{\text{n}}}. \tag{16}$$

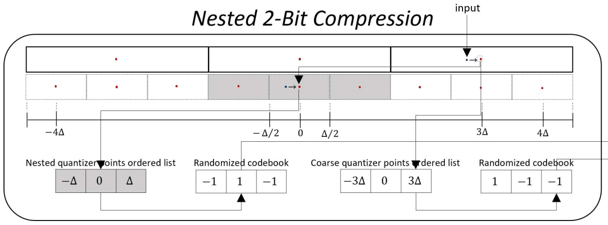

Fig. 5. Example: the input is mapped to $3\Delta$ and $0$ by the coarse and nested quantizers respectively; that are the third and second quantization points in each, correspond to $1$ and $-1$ in the randomized vectors, as the outputs.

Each of the discrete representations in (15) is then separately mapped into a single bit through the 1-bit compression and privacy enhancement mechanisms detailed in Subsection 3.1. For clarity, the nested extension of the 1-bit compression example given in Fig. 3 is depicted in Fig. 5.

### 3.2.3 Decoding

Since the server has access to two bits for $\boldsymbol{h}_{t,i}^r$, representing its coarse and nested quantization, it reconstructs two separate histograms for each lattice. By repeating the steps in (12) and (13), the server obtains two histogram estimates: one over $\mathcal{L}_t^{\text{c}}$, denoted $\tilde{\boldsymbol{v}}_{t,i}^{\text{c}} \in \mathbb{R}^{2^{R_{\text{c}}}}$, and another over $\mathcal{L}_t^{\text{n}}$ denoted $\tilde{\boldsymbol{v}}_{t,i}^{\text{n}} \in \mathbb{R}^{2^{R_{\text{n}}}}$. By (16), the averaging of $\{Q_{\mathcal{L}_t^{\text{f}}}(\boldsymbol{h}_{t,i}^r)\}_{r=1}^K$ is estimated as the sum of the empirical means over $\mathcal{L}_t^{\text{c}}$ and $\mathcal{L}_t^{\text{n}}$, i.e., the aggregation in (14) is replaced with

$$\boldsymbol{w}_{t,i}^{\text{CPA}} = \boldsymbol{w}_{t-\tau,i}^{\text{CPA}} + \sum_{l_c=1}^{2^{R_{\text{c}}}} \left[\tilde{\boldsymbol{v}}_{t,i}^{\text{c}}\right]_{l_{\text{c}}} \cdot \boldsymbol{q}_{\text{c}}^{l_{\text{c}}} + \sum_{l_n=1}^{2^{R_{\text{n}}}} \left[\tilde{\boldsymbol{v}}_{t,i}^{\text{n}}\right]_{l_{\text{n}}} \cdot \boldsymbol{q}_{\text{n}}^{l_{\text{n}}}. \tag{17}$$

The aggregation rule in (17) is designed to approach the corresponding rule obtained using 1-bit CPA in Algorithm 1 with the fine lattice $\mathcal{L}_t^{\text{f}}$, while using relatively easy-to-estimate histograms over smaller dictionaries. These nested extensions of CPA allows to facilitate the learning of an improved model in a federated manner when the number of users $K$ is limited, as numerically evidenced in Section 5. There, notable improvements of the nested operation over 1-bit CPA are observed for $K = 10$ users, which vanish for $K = 1000$ users. This capability comes at the cost of additional bits being exchanged, though such increased communication is likely to be more tolerable when learning with tens of users compared with learning with thousands and more users.

### 3.3 Discussion

The proposed CPA is a dual-function mechanism for enhancing privacy while compressing the model updates and aggregating in a robust fashion over large-scale FL systems. It is inspired by the coding scheme of private multi-group aggregation [41], with the incorporation of lattice quantization and LDP enhancement. The perturbation introduced in RR, originating from the need to meet *R1*, is mitigated here not by the conventional FedAvg as in, e.g., [33], but rather by a unique reconstruction of discrete histograms, which in turn guarantees $k$-anonymity, as we show in Section 4. While CPA exploits the high-dimensional structure of the model updates through (possibly high-rate) lattice quantization, it still dramatically reduces the conventional FL communication overhead. This follows since the users transmit at most $B$ bits per sample, and the server avoids recovering the individual updates for each user. The resulting operation is thus scalable and applicable without limiting the number of participants in each FL round. In fact, its associated distortion decreases with the number of users, as shown in Section 4.

Moreover, because CPA inherently limits the influence of a user in the final model, it allows for the training to be Byzantine robust against erroneous adversarial users and poisoning attacks [22], as was also observed for one bit SGD with majority vote aggregation in [66] . This is numerically demonstrated in Section 5. We note that CPA is expected to enhance privacy also against external adversaries, due to LDP post-processing property [57]. However, since FL is motivated by the need to avoid sharing local data with a centralized server, characterizing LDP guarantees from external adversaries is left for future work. Altogether, CPA satisfies *R1-R5* without notably affecting the utility of the learned model, compared to using separate privacy enhancement and quantization, as numerically demonstrated in Section 5.

CPA is designed assuming that all users share the same privacy requirement $\varepsilon$ by *R1*. However, since the encoding of CPA is done separately by each user, one can extend its operation to user-specific privacy budgets. The direct approach simply has all users set their RR parameter $p$ to be the lowest non-flipping probability among all utilized LDP mechanisms, i.e., the one corresponding to the user with the most strict privacy requirements. Yet, it can possibly be that each user uses a different value of $p$ by modifying the aggregation rule at the server, though this extension and its analysis are left for future work. Furthermore, the privacy requirement considered in CPA is imposed on each communication round by *R1*. Traditionally, this requirement can be related to an upper bound on the privacy budget in FL over a fixed number of global training rounds by the composition theorem [64, Thm. III.1.]. Alternatively, additional mechanisms can be implemented to avoid accumulating privacy leakage over multiple rounds, as characterized by the composition theorem [17]. We leave the combination of CPA with such mechanisms for future study.

The nested implementation of CPA allows users to convey multiple bits per sample. By doing so, the server is able to construct smaller and therefore more accurate histograms, improving the global model update design in each FL round, particularly when $K$ is relatively small. However, with each bit added, the influence of each individual user on the constructed global model, being accordingly updated, grows and increases CPA's sensitivity to malicious users. This gives rise to the existence of a tradeoff between model accuracy, compression, and robustness, whose analysis is left for future work.

## 4 Performance Analysis

CPA is designed to jointly support compression and privacy in FL over large-scale networks. Compression directly follows as each user conveys merely $B$ bits per sample. For 1-bit CPA, this boils down to merely $M$ bits, i.e., the number of bits is not larger than the number of weights. Consequently, we dedicate this section to theoretically analyze the *privacy* and *learning* implications of CPA, focusing on its 1-bit implementation for simplicity. We characterize its privacy guarantees (Subsection 4.1), distortion in its recovered global model (Subsection 4.2), and convergence profile (Subsection 4.3).

### 4.1 Privacy Analysis

In accordance with Requirement *R1*, we are considering two privacy measures: LDP and $k$-anonymity. As conventionally done in the private FL literature [16], [39], [40], [67], [68], we characterize the privacy by observing its leakage with respect to the weights for each FL round. This stems from the sequential data processing nature of the local learning procedure, which implies that assuring privacy with respect to the model weights guarantees privacy with regard to the datasets.

More specifically, for a given dataset $\mathcal{D}_r$, each bit produced by CPA can be generally viewed as encompassing two subsequent transformations: the first is the training of the model and the quantization of its weights into that corresponding bit, represented by the mapping $f : \mathcal{D}_r \to \{0, 1\}$; and the second is the incorporation of further perturbation via the RR mechanism $R(\cdot)$, i.e., $R(f(\mathcal{D}_r))$. This sequential data-processing form is related to a conventional result in the literature on differential privacy, stating that $\varepsilon$-LDP is not only guaranteed for $f(\mathcal{D}_r)$, but also for $\mathcal{D}_r$ [69]. This is proven in the following claim:

**Claim 1.** *Let $\mathcal{A}$ be a finite set, and $f : \mathcal{A} \to \{0, 1\}$. Let $R : \{0, 1\} \to \{0, 1\}$ be the RR mechanism with differential privacy budget $\varepsilon$. Then, the mechanism $M : \mathcal{A} \to \{0, 1\}$ defined as $M(\boldsymbol{x}) = R(f(\boldsymbol{x}))$ is an $\varepsilon$-LDP mechanism (i.e. any two elements of $\mathcal{A}$ are $\varepsilon$-indistinguishable).*

*Proof:* Let $\boldsymbol{x} \neq \boldsymbol{y} \in \mathcal{A}$ and $i \in \{0, 1\}$. If it holds that $f(\boldsymbol{x}) = f(\boldsymbol{y})$, then

$$\begin{aligned}\mathbb{P}[M(\boldsymbol{x}) = i] = \mathbb{P}[R\left(f(\boldsymbol{x})\right) = i] &= \mathbb{P}[R\left(f(\boldsymbol{y})\right) = i] \\ &= \mathbb{P}[M(\boldsymbol{y}) = i].\end{aligned}$$

That is, since $\mathbb{P}[M(\boldsymbol{x}) = i] = \mathbb{P}[M(\boldsymbol{y}) = i]$; we actually get that $\boldsymbol{x}, \boldsymbol{y}$ are 0-indistinguishable. Otherwise, the two bits satisfy $f(\boldsymbol{x}) \neq f(\boldsymbol{y})$, and since $R(\cdot)$ is the RR mechanism with differential privacy budget $\varepsilon$, it satisfies

$$\begin{aligned}\mathbb{P}[M(\boldsymbol{x})=i] = \mathbb{P}[R\left(f(\boldsymbol{x})\right)=i] &\leq e^{\varepsilon}\mathbb{P}[R\left(f(\boldsymbol{y})\right)=i] \\ &= e^{\varepsilon}\mathbb{P}[M(\boldsymbol{y})=i].\end{aligned}$$

Thus, $\mathbb{P}[M(\boldsymbol{x})=i] \leq e^{\varepsilon}\mathbb{P}[M(\boldsymbol{y})=i]$, and $M$ is an $\varepsilon$ locally differentially private mechanism on $\mathcal{A}$. □

We note that the setting where $f(\boldsymbol{x}) = f(\boldsymbol{y})$, i.e., two different datasets lead to having an identical mapping (due to quantization), is actually where the $k$-anonymity of CPA appears.

The encoding steps of CPA are directly derived to meet the definitions of both privacy measures, LDP and $k$-anonymity, as formally stated in the following propositions.

**Proposition 1.** *CPA is $\varepsilon$-LDP with respect to $\mathcal{D}_r$, per communication round.*

The LDP guarantee in Proposition 1 follows solely from the usage of RR, and is invariant of the preceding processing of CPA. While in this work we further utilize probabilistic quantizers only for CPA's distortion analysis (Subsection 4.2), one can possibly additionally enhance privacy in the algorithm's quantization stage by a proper design of this stochastic compression technique, as shown in [39], though this extension is left for future study. Nevertheless, the quantization stage plays a key role in achieving $k$-anonymity, as stated next.

**Proposition 2.** *CPA preserves $k$-anonymity with respect to the lattice quantization of $\boldsymbol{h}_{t,i}^{r}$.*

*Proof:* $k$-anonymity (Def. 2.5) follows by-construction from CPA's design in Algorithm 1. There, a user's update $\boldsymbol{h}_t^r$ is decomposed into sub-vectors $\{\boldsymbol{h}_{t,i}^r\}_{i=1}^M$. Each $\boldsymbol{h}_{t,i}^r \in \mathbb{R}^L$ is quantized into one codeword out of all possible codewords, say of index $l$, $\boldsymbol{q}_l = Q_{\mathcal{L}_t}(\boldsymbol{h}_{t,i}^r)$; which is in turn mapped into a binary vector $\boldsymbol{v}_{t,i}^r$ of length equals to the total number of codewords. $\boldsymbol{v}_{t,i}^r$ is comprised of equal amount of 1's and $-1$'s in expectation and assured to have 1 in its $l$th entry (see Fig. 2).

Therefore, as half of $\boldsymbol{v}_{t,i}^r$'s entries are expected to be 1, it is implied that $k$ values - being half of codewords - are valid candidates mappings of $\boldsymbol{h}_{t,i}^r$. Since a vector quantizer of dimension $L$ and rate $R$ has $2^{RL}$ codewords, 1-bit CPA with such a quantizer has the server not able to distinguish between expected $k \triangleq 2^{LR}/2 = 2^{LR-1}$ adequate values. □

The $k$-anonymity of CPA guarantees the indistinguishability between $k$ possible values of the lattice quantization of $\boldsymbol{h}_{t,i}^r$, where $k = 2^{LR-1}$ is solely determined by the quantization dimension and rate parameters. Both can be tuned to fix a desirable value of $k$ in parallel to trade-offing privacy and compression considerations.

While Proposition 2 formulates the anonymity degree of each sub-vector, Corollary 1 reveals the higher degree of anonymity achieved with respect to the complete model.

**Corollary 1.** *CPA preserves $k^M$ anonymity with respect to the lattice quantization of $\boldsymbol{h}_t^r$.*

*Proof:* The corollary follows directly from Proposition 2. The server cannot distinguish between $k$ different possibilities for each $\boldsymbol{h}_{t,i}^r$ and $\boldsymbol{h}_t^r$ is a concatenation of $\{\boldsymbol{h}_{t,i}^r\}_{i=1}^M$. Thus, each set of bits can represent $k^M$ different $\boldsymbol{h}_t^r$ settings. □

Proposition 2 in fact follows from the nature of the $k$-anonymity measure, which is defined over finite sets (see Def. 2.5). However, by the operation of the quantization procedure, it collapses all the continuous values within a certain cell into the same decision, i.e., lattice point. This can be interpreted as a kind of "continuous" $k$-anonymity, as, in addition to the indistinguishability between $k$ different lattice points guaranteed by Proposition 2, a quantized private value can potentially originate from any item in the uncountable set that covers the decision area mapped into this lattice point.

### 4.2 Weights Distortion

CPA incorporates lossy compression, RR, and a unique aggregation formulation that deviates from the conventional FedAvg. All of these steps inherently induce some distortion on the model updates, compared to the desired average of the model updates. To quantify this distortion, we characterize the difference between the model aggregated via CPA denoted $\boldsymbol{w}_{t+\tau}^{\text{CPA}}$ obtained by stacking $\{\boldsymbol{w}_{t,i}^{\text{CPA}}\}_{i=1}^M$ in (14), with the desired average (whose direct computation gives rise to communication and privacy considerations) obtained from the same global model at time $t$, given by

$$\boldsymbol{w}_{t+\tau}^{\text{FA}} \triangleq \boldsymbol{w}_t^{\text{CPA}} + \frac{1}{K}\sum_{r=1}^{K}\boldsymbol{h}_{t+\tau}^r. \tag{18}$$

Next, we show that the effect of the excessive distortion induced by CPA can be mitigated while recovering the desired $\boldsymbol{w}_{t+\tau}^{\text{FA}}$ as $\boldsymbol{w}_{t+\tau}^{\text{CPA}}$. Thus, the accuracy of the global learned model is maintained despite the incorporation of the scalable compression and privacy mechanisms of CPA. In our analysis we adopt the following assumption

*AS1* CPA uses *probabilistic quantization*, i.e., $Q_{\mathcal{L}}(\cdot)$ implements *DQ*.

Probabilistic quantizers, as assumed in *AS1*, are often employed in FL, due to the stochastic nature of their associated distortion [25], [30], [31], [33], [39]. For a given probabilistic lattice quantizer defined using the lattice $\mathcal{L}$ with lattice points $\{\boldsymbol{q}^l\}_{l=1}^{2^R}$, we define $\bar{\sigma}_{\mathcal{L}}^2$ to be the normalized second-order moment of this lattice, or alternatively, the variance of the distortion induced by DQ with lattice $\mathcal{L}$, as

$$\bar{\sigma}_{\mathcal{L}}^2 \triangleq \frac{\frac{1}{L}\int_{\mathcal{P}_0}\|\boldsymbol{x}\|^2 d\boldsymbol{x}}{\text{vol}(\mathcal{P}_0)^{2/L}}; \tag{19}$$

where $L$ is the lattice dimension, 'vol' stands for 'volume', and $\mathcal{P}_0 = \{\boldsymbol{x} : Q_{\mathcal{L}}(\boldsymbol{x}) = \boldsymbol{0}\}$ is the basic lattice cell (see further details in Section 2.2). For CPA employing this lattice quantizer, the distance between the recovered model $\boldsymbol{w}_{t+\tau}^{\text{CPA}}$ and the desired one $\boldsymbol{w}_{t+\tau}^{\text{FA}}$ satisfies:

**Theorem 4.1.** *When Assumption* AS1 *holds, the mean-squared distance between $\boldsymbol{w}_{t+\tau}^{\text{CPA}}$ and $\boldsymbol{w}_{t+\tau}^{\text{FA}}$ is bounded by*

$$\mathbb{E}\left[\left\|\boldsymbol{w}_{t+\tau}^{\text{CPA}} - \boldsymbol{w}_{t+\tau}^{\text{FA}}\right\|^2\right] \leq \frac{M}{K}\left(\sum_{l=1}^{2^R}\frac{\left\|\boldsymbol{q}^l\right\|^2}{(2p-1)^2} + \bar{\sigma}_{\mathcal{L}}^2\right). \tag{20}$$

*Proof:* To prove Theorem 4.1, we separate the distortion induced by CPA into two terms and bound each of them separately. The first term is the error caused by the discrete histogram estimation in (13); the latter is the compression distortion.

As in Subsection 3.1.2, we denote by $\{\boldsymbol{v}_i\}_{i=1}^M$ the decomposition of a vector $\boldsymbol{v}$ into $M$ distinct $L\times 1$ sub-vectors. Moreover, we write $\{c^l\}$ as the quantizer words' rates, given by $c^l \triangleq \frac{1}{K}\sum_{r=1}^K \delta\left[Q_{\mathcal{L}_t}(\boldsymbol{h}^r_{t+\tau,i})-\boldsymbol{q}^l\right]$, where $\delta[\cdot]$ is the Kronecker delta. By (18) and (14) we have that

$$\mathbb{E}\left[\left\|\boldsymbol{w}^{\rm CPA}_{t+\tau,i}-\boldsymbol{w}^{\rm FA}_{t+\tau,i}\right\|^2\right]=\mathbb{E}\left[\left\|\frac{1}{K}\sum_{r=1}^{K}\boldsymbol{h}^r_{t+\tau,i}-\sum_{l=1}^{2^R}\left[\tilde{\boldsymbol{v}}_{t+\tau,i}\right]_l\boldsymbol{q}^l\right\|^2\right]$$
$$\overset{(a)}{=}\mathbb{E}\left[\left\|\left(\frac{1}{K}\sum_{r=1}^{K}\boldsymbol{h}^r_{t+\tau,i}-Q_{\mathcal{L}_t}(\boldsymbol{h}^r_{t+\tau,i})\right)-\left(\sum_{l=1}^{2^R}\left(c^l-\left[\tilde{\boldsymbol{v}}_{t+\tau,i}\right]_l\right)\boldsymbol{q}^l\right)\right\|^2\right], \qquad (4.2.21)$$

where $(a)$ is obtained by adding and subtracting $\frac{1}{K}\sum_{r=1}^K Q_{\mathcal{L}}\left(\boldsymbol{h}^r_{t+\tau,i}\right)=\sum_{l=1}^{2^R} c^l\cdot\boldsymbol{q}^l$.

To proceed, we define the compression distortion $\boldsymbol{e}^r_{t+\tau,i}\triangleq Q_{\mathcal{L}_t}(\boldsymbol{h}^r_{t+\tau,i})-\boldsymbol{h}^r_{t+\tau,i}$ and the histogram estimation error $\eta^l_{t+\tau,i}\triangleq c^l-[\tilde{\boldsymbol{v}}_{t+\tau,i}]_l$. The joint distribution of $\{\boldsymbol{e}^r_{t+\tau,i}\},\{\eta^l_{t+\tau,i}\}$ is characterized in the following lemma:

**Lemma 4.2.1.** *For any $\{\boldsymbol{h}^r_{t+\tau,i}\}$ it holds that the sequences $\{\boldsymbol{e}^r_{t+\tau,i}\}$ and $\{\eta^l_{t+\tau,i}\}$ are uncorrelated (over r and l, receptively), zero-mean, and mutually uncorrelated. The variance of $\boldsymbol{e}^r_{t+\tau,i}$ equals $\bar{\sigma}^2_{\mathcal{L}}$ while that of $\eta^l_{t+\tau,i}$ is bounded by $\frac{1}{K\cdot(2p-1)^2}$.*

*Proof:* By Assumption *AS1*, $Q_{\mathcal{L}_t}$ realizes DQ, and thus its distortion is zero-mean i.i.d. with variance $\bar{\sigma}^2_{\mathcal{L}}$ [48], [49], [70]. Combining this with the independence of the dither, the codewords, and the RR implies that $\boldsymbol{e}^r_{t+\tau,i}$ and $\eta^l_{t+\tau,i}$ are uncorrelated.

For $\eta^l_{t,i}$, we define $\tilde{\eta}^{l,r}_{t,i}=\delta[Q_{\mathcal{L}}(\boldsymbol{h}^r_{t,i})-\boldsymbol{q}_l]-\left[\tilde{\boldsymbol{v}}^r_{t,i}\right]_l$. By the definitions of the codeword and RR, it holds that for any $\{\boldsymbol{h}^r_{t,i}\}$,

$$\tilde{\eta}^{l,r}_{t,i}=\begin{cases}\begin{cases}1-\frac{1}{2p-1}, \text{ w.p } p\\ 1+\frac{1}{2p-1}, \text{ w.p } 1-p\end{cases} & \text{if } \delta[Q_{\mathcal{L}}(\boldsymbol{h}^r_{t,i})-\boldsymbol{q}_l]=1,\\ \pm\frac{1}{2p-1} \text{ w.p } 0.5 & \text{otherwise;}\end{cases}$$

and thus $\eta^l_{t+\tau,i}=\frac{1}{K}\sum_{r=1}^K\tilde{\eta}^{l,r}_{t+\tau,i}$ are i.i.d. zero-mean with variance not larger than $\frac{1}{K\cdot(2p-1)^2}$. □

Altogether, combining (4.2.21) with Lemma 4.2.1 yields

$$\mathbb{E}\left[\left\|\boldsymbol{w}^{\rm CPA}_{t+\tau}-\boldsymbol{w}^{\rm FA}_{t+\tau}\right\|^2\right]=\sum_{i=1}^{M}\mathbb{E}\left[\left\|\boldsymbol{w}^{\rm CPA}_{t+\tau,i}-\boldsymbol{w}^{\rm FA}_{t+\tau,i}\right\|^2\right]$$
$$=\sum_{i=1}^{M}\mathbb{E}\left[\left\|\frac{1}{K}\sum_{r=1}^{K}\boldsymbol{e}^r_{t+\tau,i}\right\|^2\right]+\mathbb{E}\left[\left\|\sum_{l=1}^{2^R}\eta^l_{t+\tau,i}\boldsymbol{q}^l\right\|^2\right]$$
$$\le\frac{M}{K}\bar{\sigma}^2_{\mathcal{L}}+\frac{M}{K(2p-1)^2}\sum_{l=1}^{2^R}\left\|\boldsymbol{q}^l\right\|^2,$$

thus proving (20). □

Theorem 4.1 implies that the recovered model can be made arbitrarily close to the desired one by increasing the number of edge users participating in the FL training procedure, as (20) decreases as $1/K$. This holds as there, the histogram estimation error term, i.e., $\frac{1}{K}\sum_{l=1}^{2^R}\frac{\left\|\boldsymbol{q}^l\right\|^2}{(2p-1)^2}$ accounts for the distance between the histogram's empirical evaluation to its true value; which is the distance between the empirical mean of the i.i.d-randomized codewords $\{\boldsymbol{v}^r_{t,i}\}_{r=1}^K$ and its expected value. By the law of large numbers, this distance approaches zero with growing $K$ while its associated variance decreases at a rate of $1/K$.

The exact same arguments also accounts for the compression distortion term $\frac{1}{K}\bar{\sigma}^2_{\mathcal{L}}$; due to the unique framework of FL, where the users quantized quantities are only taken in *average*. In CPA, the employed compression is a probabilistic one, for which the output can be modeled as the input plus an i.i.d additive-noise of mean zero and bounded variance. As a result, adding more users does not induce more quantization errors, but is actually a contributing factor that improves the empirical estimations of CPA. This indicates the suitability of CPA for FL over large networks.

However, while the difference decaying rate is of order $\mathcal{O}(1/K)$, it is still affected by the need to compress the model updates and enhance their privacy. This is revealed in Theorem 4.1 via the presence of $M,\bar{\sigma}^2_{\mathcal{L}}$ and $p$, arising from each consideration. In particular, $M\triangleq\lceil\frac{d}{L}\rceil$ stands for the number of distinct $L\times 1$ sub-vectors in $\boldsymbol{h}^r_t\in\mathbb{R}^d$, i.e., $\{\boldsymbol{h}^r_{t,i}\}_{i=1}^M$, where each is being quantized by applying an $L$-dimensional lattice quantizer. Consequently, lower $M$, or equivalently, higher quantization dimension $(L)$, particularly (physically) implies that more entries of $\boldsymbol{h}^r_t$ are being quantized together. In vector quantitation theory [51, Part V], this is known to improve compression performance; what further explains why (20) linearly depends on $M$. The moment $\bar{\sigma}^2_{\mathcal{L}}$ follows from the distortion induced by lattice quantization, while $p$ accounts for the distortion induced by the RR mechanism. Specifically, $p$ is dictated by the privacy level $\varepsilon$, which implies that, as expected, stricter privacy constraints lead to additional distortion in the recovered global model.

## 4.3 Federated Learning Convergence

In the previous subsection we bounded the distortion induced by CPA in each communication round to achieve privacy and compression. Next, we show that this property is translated into FL convergence guarantees. To that aim, we further introduce the following assumptions, that are commonly employed in FL convergence studies in, e.g., [33], [44], [71], on the local datasets, stochastic gradients, and objectives:

*AS2* Each dataset $\mathcal{D}_r$ is comprised of i.i.d samples. However, different datasets can be statistically heterogeneous, i.e., arise from different distributions.

*AS3* The expected squared $\ell_2$-norm of the vector $\nabla F_r^{j_t^r}(\boldsymbol{w})$ in (3) is bounded by some $\xi_r^2>0$ for all $\boldsymbol{w}\in\mathbb{R}^d$.

*AS4* The objective functions $\{F_r(\cdot)\}_{r=1}^K$ are $\rho_s$-smooth and $\rho_c$-strongly convex, i.e., for all $\boldsymbol{w}_1,\boldsymbol{w}_2\in\mathbb{R}^d$ we have

$$(\boldsymbol{w}_1-\boldsymbol{w}_2)^T\nabla F_r(\boldsymbol{w}_2)+\frac{1}{2}\rho_c\|\boldsymbol{w}_1-\boldsymbol{w}_2\|^2$$
$$\le F_r(\boldsymbol{w}_1)-F_r(\boldsymbol{w}_2)\le$$
$$(\boldsymbol{w}_1-\boldsymbol{w}_2)^T\nabla F_r(\boldsymbol{w}_2)+\frac{1}{2}\rho_s\|\boldsymbol{w}_1-\boldsymbol{w}_2\|^2.$$

Statistical heterogeneity as in *AS2* is a common characteristic of FL [3], [4], [5]. It is consistent with Requirement *R3*, which does not impose any specific distribution on the data. Statistical heterogeneity implies that the local objectives differ between users, hence the dependence on $r$ in *AS3*, often employed in distributed learning studies [33], [44], [71]. Following *AS2* and [33], [39], [71], [72], we define the heterogeneity gap,

$$\psi \triangleq F(\boldsymbol{w}^{\rm opt}) - \frac{1}{K}\sum_{r=1}^{K}\min_{\boldsymbol{w}} F_r(\boldsymbol{w}), \tag{4.3.1}$$

where $\boldsymbol{w}^{\rm opt}$ is defined in (1), and $F(\boldsymbol{w}^{\rm opt})$, $\min_{\boldsymbol{w}} F_r(\boldsymbol{w})$ are the minimum values of $F,\ F_r$, respectively. Eq. (4.3.1) quantifies the degree of non-i.i.d: for i.i.d. data, $\psi$ approaches zero as the number of samples grows; and otherwise, non-i.i.d. data results with nonzero $\psi$, and its magnitude reflects the heterogeneity of the data distribution. Finally, assumption *AS4* holds for a range of objective functions used in FL, including $\ell_2$-norm regularized linear regression and logistic regression [33].

It is emphasized, though, that assumptions *AS1-AS4* are only introduced for having a tractable analysis, and we further empirically demonstrate the usefulness of CPA, whose derivation is invariant to these assumptions, in settings where they do not necessarily hold as exemplified in Section 5.

The following theorem characterizes the convergence of FL employing CPA with local-SGD training:

**Theorem 4.2.** *Let $\mathcal{L}$ be a lattice with generator matrix $\boldsymbol{G}$, moment $\bar{\sigma}^2_{\mathcal{L}}$, and points $\{\boldsymbol{q}^l\}_{l=1}^{2^R}$. Set $\varphi \triangleq \tau\max(1, 4\rho_s/\rho_c)$. Then consider CPA-aided FL satisfying* AS1-AS4 *while using, at each round $t$, a lattice quantizer $\mathcal{L}_t$ with generator matrix $\boldsymbol{G}_t = \zeta_t\cdot\boldsymbol{G}$, where $\zeta_t$ is a positive sequence holding $\zeta_t^2 \leq C\cdot\eta_t^2$ for some fixed $C>0$. Under this setting, local-SGD with step-size $\eta_t = \frac{\tau}{\rho_c(t+\varphi)}$ for each $t\in\mathbb{N}$ satisfies*

$$\mathbb{E}\left[F(\boldsymbol{w}_t^{\rm CPA})\right] - F(\boldsymbol{w}^{\rm opt}) \leq \frac{\rho_s}{2(t+\varphi)}\max\left(\frac{\rho_c^2+\tau^2 b}{\tau\rho_c^2}, \varphi\|\boldsymbol{w}_0 - \boldsymbol{w}^{\rm opt}\|^2\right), \tag{4.3.2}$$

*where*

$$b \triangleq \frac{1}{K}\Bigg\{M\cdot C\cdot\Big(\sum_{l=1}^{2^R}\frac{\left\|\boldsymbol{q}^l\right\|^2}{(2p-1)^2} + \bar{\sigma}^2_{\mathcal{L}}\Big) + \frac{1}{K}\sum_{r=1}^{K}\xi_r^2 + 8(\tau-1)^2\sum_{r=1}^{K}\xi_r^2\Bigg\} + 6\rho_s\psi. \tag{4.3.3}$$

*Proof:* To prove the theorem, we derive a recursive bound on the weights error, from which the FL convergence bound is then concluded. This outline follows the steps used in [33], which did not consider privacy or anonymity guarantees. We next briefly describe the main steps for completeness, deferring the proofs of some of the intermediate lemmas to [33].

### *Recursive Bound on Weights Error*

Denote by $\mathcal{I}_\tau \triangleq \{n\tau | n = 1, 2, \dots\}$ the set of global synchronization steps, i.e., $t+1\in\mathcal{I}_\tau$ is a time step indicates communication of all devices. Aiding these notations, CPA induces excessive distortion (compared to vanilla FedAvg) in each time instance in $\mathcal{I}_\tau$. This can be formally written as

$$\boldsymbol{w}_{t+1}^r = \begin{cases}\boldsymbol{w}_t^r - \eta_t\nabla F_r^{j_t^r}\left(\boldsymbol{w}_t^r\right) & t+1\notin\mathcal{I}_\tau,\\ \boldsymbol{w}_t^{\rm CPA} & t+1\in\mathcal{I}_\tau;\end{cases}$$
$$\overset{(a)}{\triangleq}\begin{cases}\boldsymbol{w}_t^r - \eta_t\nabla F_r^{j_t^r}\left(\boldsymbol{w}_t^r\right) + \underbrace{\boldsymbol{d}_{t+1}}_{=0} & t+1\notin\mathcal{I}_\tau,\\ \frac{1}{K}\sum\limits_{r'=1}^{K}\left(\boldsymbol{w}_t^{r'} - \eta_t\nabla F_{r'}^{j_t^{r'}}\left(\boldsymbol{w}_t^{r'}\right) + \boldsymbol{d}_{t+1}\right) & t+1\in\mathcal{I}_\tau;\end{cases} \tag{4.3.4}$$

where $(a)$ follows from the subtraction and addition of $\boldsymbol{w}_t^{\rm FA}$ defined in (18), and setting $\boldsymbol{d}_{t+1}\triangleq\boldsymbol{w}_t^{\rm CPA} - \boldsymbol{w}_t^{\rm FA}$.

We next define a virtual sequence $\{\boldsymbol{z}_t\}$ from $\{\boldsymbol{w}_t^r\}$ that coincides with $\boldsymbol{w}_t^{\rm CPA}$ for $t\in\mathcal{I}_\tau$. Specifically,

$$\boldsymbol{z}_{t+1}\triangleq\frac{1}{K}\sum_{r=1}^{K}\boldsymbol{w}_{t+1}^r \overset{(4.3.4)}{=} \frac{1}{K}\sum_{r=1}^{K}\left(\boldsymbol{w}_t^r - \eta_t\nabla F_r^{j_t^r}\left(\boldsymbol{w}_t^r\right) + \boldsymbol{d}_{t+1}\right)$$
$$= \boldsymbol{z}_t - \eta_t\underbrace{\frac{1}{K}\sum_{r=1}^{K}\left(\nabla F_r^{j_t^r}\left(\boldsymbol{w}_t^r\right) - \frac{1}{\eta_t}\boldsymbol{d}_{t+1}\right)}_{\triangleq\tilde{\boldsymbol{g}}_t}. \tag{4.3.5}$$

In (4.3.5), $\tilde{\boldsymbol{g}}_t$ is the averaged noisy stochastic gradient, where the averaged full gradient are $\boldsymbol{g}_t\triangleq\frac{1}{K}\sum_{r=1}^{K}\nabla F_r\left(\boldsymbol{w}_t^r\right)$.

The resulting model is thus equivalent to that used in [33, App. C]. Thus, by *AS4* it follows that if $\eta_t\leq\frac{1}{4\rho_s}$ then

$$\mathbb{E}\left[\left\|\boldsymbol{z}_{t+1} - \boldsymbol{w}^{\rm opt}\right\|^2\right] \leq (1-\eta_t\rho_c)\mathbb{E}\left[\left\|\boldsymbol{z}_t - \boldsymbol{w}^{\rm opt}\right\|^2\right] + 6\rho_s\eta_t^2\psi + \eta_t^2\mathbb{E}\left[\left\|\tilde{\boldsymbol{g}}_t - \boldsymbol{g}_t\right\|^2\right] + 2\mathbb{E}\left[\frac{1}{K}\sum_{r=1}^{K}\left\|\boldsymbol{z}_t - \boldsymbol{w}_t^r\right\|^2\right]. \tag{4.3.6}$$

Equation (4.3.6) bounds the expected distance between the virtual sequence $\{\boldsymbol{z}_t\}$ and the optimal weights $\boldsymbol{w}^{\rm opt}$ in a recursive manner. We further bound the summands in (4.3.6):

**Lemma 4.3.1.** *If the step-size $\eta_t$ is non-increasing and satisfies $\eta_t\leq 2\eta_{t+\tau}$ for each $t\geq 0$, then, when* AS3 *holds, we have*

$$\eta_t^2\mathbb{E}\left[\left\|\tilde{\boldsymbol{g}}_t - \boldsymbol{g}_t\right\|^2\right] \leq \frac{\eta_t^2}{K}\left(M\cdot C\left(\sum_{l=1}^{2^R}\frac{\left\|\boldsymbol{q}^l\right\|^2}{(2p-1)^2} + \bar{\sigma}^2_{\mathcal{L}}\right) + \frac{1}{K}\sum_{r=1}^{K}\xi_r^2\right). \tag{4.3.7}$$

*Proof:* To prove (4.3.7), we separate it into two independent terms and bound each separately. Specifically,

$$\eta_t^2\mathbb{E}\left[\left\|\tilde{\boldsymbol{g}}_t - \boldsymbol{g}_t\right\|^2\right] = \mathbb{E}\left[\left\|\boldsymbol{d}_{t+1}\right\|^2\right] \tag{4.3.8}$$
$$+ \eta_t^2\mathbb{E}\left[\left\|\frac{1}{K}\sum_{r=1}^{K}\left(\nabla F_r^{j_t^r}\left(\boldsymbol{w}_t^r\right) - \nabla F_r\left(\boldsymbol{w}_t^r\right)\right)\right\|^2\right] \tag{4.3.9}$$
$$- \frac{\eta_t}{K}\sum_{r=1}^{K}\mathbb{E}\left[\boldsymbol{d}_{t+1}^T\left(\nabla F_r^{j_t^r}\left(\boldsymbol{w}_t^r\right) - \nabla F_r\left(\boldsymbol{w}_t^r\right)\right)\right]. \tag{4.3.10}$$

We first note that (4.3.10) $= 0$ as the stochastic gradients are unbiased estimates of the true gradients and are independent of the distortion. For the first term, it holds that

$$\text{(4.3.8)} \overset{(a)}{\leq} \frac{M}{K}\left(\sum_{l=1}^{2^R}\frac{\left\|\boldsymbol{q}^l\right\|^2}{(2p-1)^2}+\bar{\sigma}_{\mathcal{L}}^2\right)$$

$$\overset{(b)}{\leq} \frac{\eta_t^2 M\cdot C}{K}\left(\sum_{l=1}^{2^R}\frac{\left\|\boldsymbol{q}^l\right\|^2}{(2p-1)^2}+\bar{\sigma}_{\mathcal{L}}^2\right), \qquad (4.3.11)$$

where $(a)$ follows by Theorem 4.1; $(b)$ holds as CPA here realizes a time-variant lattice quantizer $\mathcal{L}_t$ with a bounded scaled generator matrix. For the second term we have

$$\text{(4.3.9)} \overset{(a)}{=} \frac{\eta_t^2}{K^2}\sum_{r=1}^{K}\mathbb{E}\left[\left\|\nabla F_r^{j_t^r}\left(\boldsymbol{w}_t^r\right)-\nabla F_r\left(\boldsymbol{w}_t^r\right)\right\|^2\right]$$

$$\overset{(b)}{\leq} \frac{\eta_t^2}{K^2}\sum_{r=1}^{K}\xi_r^2, \qquad (4.3.12)$$

where (a) follows form the uniform distribution of the random index $j_t^r$; and $(b)$ stems from *AS3*. Combining (4.3.11) and (4.3.12) concludes the proof. □

**Lemma 4.3.2.** *If the step-size $\eta_t$ is non-increasing and satisfies $\eta_t \leq 2\eta_{t+\tau}$ for each $t \geq 0$, then when* AS3 *holds, we have*

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{z}_t-\boldsymbol{w}_t^r\|^2\right] \leq \frac{4(\tau-1)^2\eta_t^2}{K}\sum_{r=1}^{K}\xi_r^2, \qquad (4.3.13)$$

*Proof:* The proof of Lemma 4.3.2 is given in [33, App. C]. □

Next, we define $\delta_t \triangleq \mathbb{E}\left[\|\boldsymbol{z}_t-\boldsymbol{w}^{\rm opt}\|^2\right]$, which represents the $\ell_2$-norm of the error in the weights of the global model for $t \in \mathcal{I}_\tau$. Using Lemmas 4.3.1-4.3.2, while integrating (4.3.7) and (4.3.13) into (4.3.6), we obtain the following recursive relationship:

$$\delta_{t+1} \leq (1-\eta_t\rho_c)\delta_t + \eta_t^2 b, \qquad (4.3.14)$$

where $b$ is defined in (4.3.3). The relationship in (4.3.14) is used in the sequel to prove the FL convergence bound in (4.3.2).

### FL Convergence Bound

We next obtain the convergence bound by setting the step-size and the FL parameters in (4.3.14) to bound $\delta_t$; and combine the resulting bound with *AS4* to prove (4.3.2). In particular, we set $\eta_t$ to take the form $\eta_t = \frac{\beta}{t+\varphi}$ for some $\beta > 0$ and $\varphi \geq \max(4\rho_s\beta, \tau)$, for which $\eta_t \leq \frac{1}{4\rho_s}$ and $\eta_t \leq 2\eta_{t+\tau}$, implying that (4.3.6), (4.3.7), and (4.3.13) hold. Under such settings, in [33, App. C] is it proved that for $\lambda \geq \max\left(\frac{1+\beta^2 b}{\beta\rho_c}, \varphi\delta_0\right)$ is holds that $\delta_t \leq \frac{\lambda}{t+\varphi}$ for all integer $t \geq 0$. Finally, the smoothness of the objective *AS4* implies that

$$\mathbb{E}\left[F(\boldsymbol{w}_t^{\rm CPA})\right] - F(\boldsymbol{w}^{\rm opt}) \leq \frac{\rho_s}{2}\delta_t \leq \frac{\rho_s\lambda}{2(t+\varphi)}. \qquad (4.3.15)$$

Setting $\beta = \frac{\tau}{\rho_c}$ results in $\varphi \geq \tau\max(1, 4\rho_s/\rho_c)$ and $\lambda \geq \max\left(\frac{\rho_c^2+\tau^2 b}{\tau\rho_c^2}, \varphi\delta_0\right)$; once substituted into (4.3.15), proves (4.3.2). □

Theorem 4.2 rigorously bounds the difference in the objective value of the optimal model $\boldsymbol{w}^{\rm opt}$ and the one learned by CPA over $t$ learning rounds with local-SGD; i.e., the users' batch size is set to 1. By taking $t$ to be asymptotically large in (4.3.2), we obtain the asymptotic convergence profile, indicating that CPA with local-SGD converges at a rate of $\mathcal{O}(1/t)$. This reverse dependence on $t$ was formed due to the specific design of the step size $\eta_t$ to gradually decreases, which is also known to contribute to the convergence of FL [44], [71], and the usage of a lattice with a gradually decaying dynamic range to fit the quantizer to the decaying magnitude of the model updates expected for converging FL. The asymptotic rate of $\mathcal{O}(1/t)$ is of the same order of convergence as FL with neither privacy nor compression constraints [44], [71], indicating the ability of CPA to satisfy these requirements while mitigating their harmful effects on the learning procedure.

In the non-asymptotic regime, the integration of compression and privacy techniques does influence model convergence, as revealed in Theorem 4.2 by the coefficient $b$. The scalability of CPA is reflected in the first summand of (4.3.3), which vanishes as the number of users $K$ grows. The terms which do not vanish as $K \to \infty$, i.e., the last two summands in (4.3.3), stem from the usage of multiple local iterations per round and from the presence of statistical heterogeneity, respectively [71]; both are common properties of FL that are not targeted in our design of CPA.

# 5 Experimental Study

In this section we numerically evaluate CPA and compare it to alternative approaches for compression and privacy in FL. We consider the federated training of different model architectures for handwritten digit identification with the MNIST dataset as well as image classification based on CIFAR-10. We quantify the distortion induced by CPA, the accuracy of the learned models, and the robustness to malicious users[1].

## 5.1 Setup

We consider FL using local-SGD with the number of edge users varying from as small as $K = 10$ to massive networks with $K = 1000$, each studying different aspects in the design of CPA.

### Baselines

We numerically evaluate the following schemes:

*vanilla FL*: assuring neither privacy nor compression.

*CPA*: 1-bit CPA with a scalar quantizer;

*CPA w/o RR*: 1-bit CPA with a scalar quantizer without LDP constraints, i.e., $\varepsilon \to \infty$;

*nested CPA*: two-stage nested CPA (see Subsection 3.2) with $R_{\rm c} = 1$, and $R_{\rm n} = 3$;

*Laplace*: local updates perturbated by a Laplacian PPN, realize the Laplace mechanism [73] and satisfy only privacy.

*signSGD & RR*: the common *signSGD* [32], which also utilizes 1-bit representations, followed by *RR*, realizing a straightforward separated design satisfying *R1-R5*.

[1] The source code used in our experimental study, including all the hyper-parameters, is available online at https://github.com/langnatalie/CPA.

*JoPEQ*: the scheme of [39], which transforms randomized lattice quantization distortion into PPN, tackling *R1-R3*.

*MVU*: the scheme of [40], which introduces discrete-valued LDP-preserving perturbation to the quantized representation of the model update.

Unless stated otherwise, all benchmarks holding $\varepsilon$-LDP set $\varepsilon = 0.5$. As for the ones involving compression, they utilize a mid-tread uniform scalar quantizer, i.e., $L = 1$ with bit-rate $R = 1$. Note that by Proposition 2, the CPA schemes satisfy $k$-anonymity with $k = 2^{LR-1}$, e.g., $k = 4$ for nested CPA with $R_{\rm n} = 3$.

### Evaluation Metrics

We aim to numerically validating that CPA indeed minimizes the excess distortion compared to individual compression and privacy enhancement operating with the same *R1-R3*. To this end, we evaluate the observed signal-to-noise ratio (SNR) [dB] of the weights obtained by CPA compared to the desired FedAvg, which we compute as the estimated variance of the model weights and divide it by the estimated variance of the distortion, namely,

$$\text{SNR} \triangleq \text{Var}(\boldsymbol{w}^{\text{FA}}) / \text{Var}(\boldsymbol{w}^{\text{FA}} - \boldsymbol{w}^{\text{CPA}}). \qquad (5.1.1)$$

Then, we compute performance scoring in terms of both validation set and test set accuracy [%].

### Architectures

We consider the following models in training:

*Linear*: the model comprised of a tunable weight matrix and a bias vector of corresponding dimensions as those of the data;

*MLP*: a multi-layer perceptron (MLP) with two hidden layers and intermediate ReLU activations;

*CNN*2/3: a convolutional neural network (CNN) composed of two or three convolutional layers, respectively, followed by fully connected ones, with intermediate ReLU activations, max-pooling and dropout layers.

*ResNet*-18: the 18 layers deep CNN of [74]. Here, we set the initial weights to be the pre-trained version of the network trained on more than a million images from the ImageNet database, having the network able to classify images into 1000 object categories. Therefore, we extend the model to constitute another final linear layer that maps the output into the required number of labels in accordance with the used dataset (e.g, 100 for CIFAR-100).

### Datasets

The above architectures are trained using the following datasets:

*MNIST*: the dataset resembles a handwritten digit identification task, is comprised of $28 \times 28$ gray-scale images divided into $60,000$ training examples and $10,000$ test examples; where each edge user possesses $5$ uniformly drawn samples.

*CIFAR*-10: a natural image classification dataset, comprised of $32 \times 32$ RGB images divided into $50,000$ training examples and $10,000$ test examples, uniformly distributed among $K$ users.

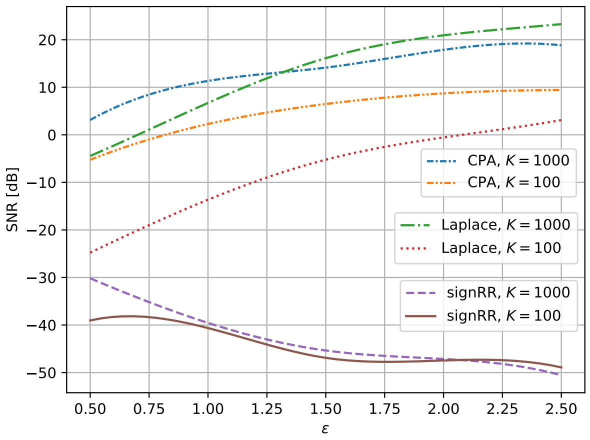

Fig. 6. SNR versus $\varepsilon$ in the received models training a linear regression model using the MNIST dataset for $K \in \{100, 1000\}$ edge users.

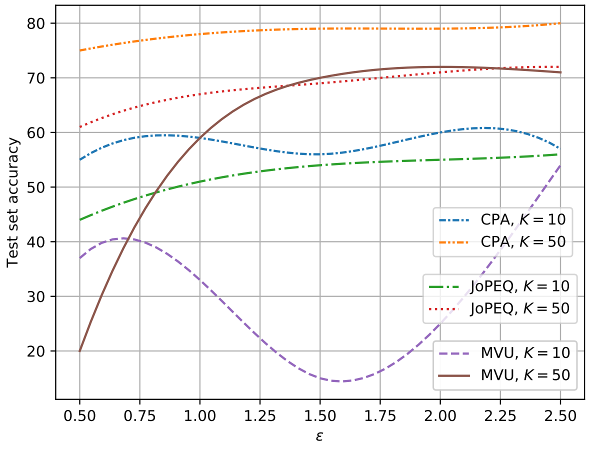

Fig. 7. Test set accuracy versus $\varepsilon$ in the received models training a linear regression model using the MNIST dataset for $K \in \{10, 50\}$ edge users.

*CIFAR*-100: a natural image classification dataset, consisting of $60,000$ color images partitioned into $100$ classes, with each class holding $600$ images. The dataset is further divided into $50,000$ training images and $10,000$ testing image, uniformly distributed among $K$ users.

## 5.2 Results

### Performance

We begin by considering the FL training of a linear regression model using MNIST, with $K = \{100, 1000\}$ participating edge users, for different privacy budgets, i.e., $\varepsilon$ values. Accordingly, Fig. 6 reports the SNR (5.1.1) values as a function of the privacy budget $\epsilon$. Evidently, CPA attains fairly equivalent performance compared to the alternative of Laplace, which only meets *R1* and *R3*, while satisfying *R1-R5* altogether. The SNR of both schemes grows with looser privacy constraints and/or more users participating; while signSGD & RR demonstrates neither. This can be attributed to the coarse sign operation, whose distortion is so dominant such that it is sometimes reduced by privacy, and is barely influenced by the number of edge users taking part in the FL training.

TABLE 2
Empirical evaluation for different number of FL participants

| | Number of edge participants $K$ | | | | |
|---|---|---|---|---|---|
| | 50 | 100 | 400 | 700 | 1000 |
| FedAvg, test set acc. | 94 | 94 | 94 | 95 | 95 |
| CPA, test set acc. | 93 | 96 | 96 | 96 | 96 |
| MSE | 0.011 | 0.007 | 0.0036 | 0.0029 | 0.0031 |

TABLE 3
Empirical evaluation of for different privacy budgets

| | LDP budget $\varepsilon$ $\left(p = \frac{e^{\varepsilon}}{1+e^{\varepsilon}}\right)$ | | | | |
|---|---|---|---|---|---|
| | 0.1 | 0.15 | 0.2 | 0.25 | 0.3 |
| FedAvg, test set acc. | 94 | | | | |
| CPA, test set acc. | 87 | 89 | 92 | 92 | 93 |
| MSE | 0.0068 | 0.0065 | 0.0063 | 0.0062 | 0.0062 |

Following Fig. 6, we continue with measuring and monitoring the performance for variable privacy budgets not only in terms of SNR but also for test set accuracy values. For that aim, we trained a linear model on the MNIST dataset and depict in Fig. 7 the performance of the counterparts baselines: CPA, joint privacy enhancement and quantization (JoPEQ) [39], and minimum variance unbiased (MVU) [40]. It in noted that both JoPEQ and MVU are not tailored for a massive number of FL participants, and we therefore experimented $K \in \{10, 50\}$. The expected behavior is a monotonically increasing one with incrementing $\varepsilon$, as higher budgets are attributed with lower added noise to the learning process. This is observed for all baselines for $K = 50$, with lesser stability for $K = 10$ (which is most notable for MVU). For either of the values of $K$ and $\varepsilon$, CPA performs the best, which adds to its inherent ability to benefit from operating with a massive number of users.

We proceed to numerically evaluate the effect of the parameters of CPA, as identified in Theorems 4.1-4.2. For that, we inspect the behavior of the mean squared error (MSE) $\frac{1}{M}\left\|\boldsymbol{w}_{t+\tau}^{\text{CPA}} - \boldsymbol{w}_{t+\tau}^{\text{FA}}\right\|^2$ bounded in Theorem 4.1, in terms of both parameters dependence and relation to model accuracy; where the latter is highly correlated with the convergence bound captured in Theorem 4.2. For that aim, we trained a CNN2 model on the MNIST dataset along 150 global rounds, and evaluated the performance of both vanilla FL and CPA. Specifically, the latter is the 1-bit CPA scheme which uses a mid-tread uniform scalar quantizer, i.e., $L = 1$, with bit-rate $R = 1$.

As the relation between different bit-rates and the converged model accuracy would be monitored in the sequel in Fig. 10, we focus here on the impact of $K$ and $\varepsilon$. At first, we varied the number of edge devices $K$ and fixed the privacy budget $\varepsilon = 0.5$, and computed the test accuracy of the baselines compared to the MSE; as summarized in Table 2. Secondly, we repeated this procedure for fixed $K = 100$ and varying $\varepsilon$ $\left(p = \frac{e^{\varepsilon}}{1+e^{\varepsilon}}\right)$ in Table 3.

It is revealed that the MSE indeed deceases for either of the parameters $K$ or $p$, in line with Theorem 4.1. Additionally, as desired, the MSE values corresponds to the model accuracy score. In practice, this quantitatively shows that convergence in the weights guarantees convergence in model performance. It follows since we first show that despite incorporating, using the algorithm of CPA, privacy and compression into FL, we obtain close resemblance in weights (via the MSE metric); and then we show that this indeed leads to similar performance in terms of the task (via the test accuracy). Thus, Tables 2-3 further support the relevance of Theorems 4.1-4.2 in the analytical analysis of CPA.

TABLE 4
Baselines test set accuracy results

| | MNIST | | CIFAR-10 | |
|---|---|---|---|---|
| | Linear | MLP | CNN2 | CNN3 |
| vanilla FL | 87 | 90 | 48 | 60 |
| Laplace | 86 | 88 | 45 | 60 |
| signSGD & RR | 79 | 10 | 10 | 10 |
| JoPEQ | 82 | 78 | 47 | 66 |
| CPA w/o RR | 87 | 80 | 43 | 47 |
| CPA | 85 | 86 | 50 | 67 |

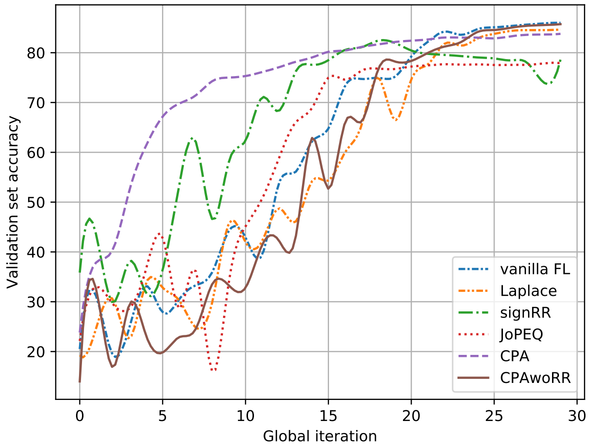

Fig. 8. Convergence profile of different FL schemes training a linear regression model using the MNIST dataset with $K = 1000$ edge users.

### *Convergence*

Next, we evaluate how the reduced excess distortion of CPA is translated into an improved learning. We depict in Fig. 8 the validation set learning curves of all referenced methods. Fig. 8 indicates that CPA performs similarly to vanilla FL which satisfies neither privacy (*R1*) nor compression (*R2*), while simultaneously assuring both. We further observe that the straightforward signSGD & RR suffers from excessive distortion which deteriorates its learned model accuracy due to the usage of distinct mechanisms for quantization and privacy, as illustrated Fig. 6. A similar observation (though of a less notable gain) is noted in comparison to the joint design via JoPEQ operating with the same rate of one bit per sample.

We continue with showing that CPA is beneficial regardless of the model specific design. We report in Table 4 the baselines' converged models test accuracy results also for an MLP model, showing in line findings with the linear model. Table 4 also reports reports the baselines' converged models test accuracy results for two CNNs trained for the CIFAR-10 dataset using $K = 1000$ users; while Fig. 9 describes the validation set learning curves of all referenced methods.

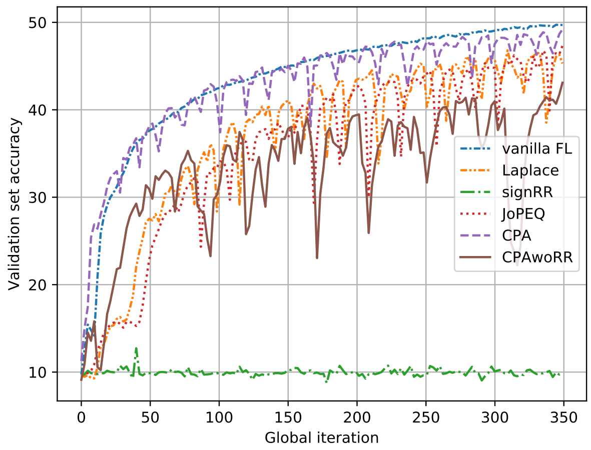

Fig. 9. Convergence profile of different FL schemes training a 2-layered CNN model using the CIFAR-10 dataset with $K = 1000$ edge users.

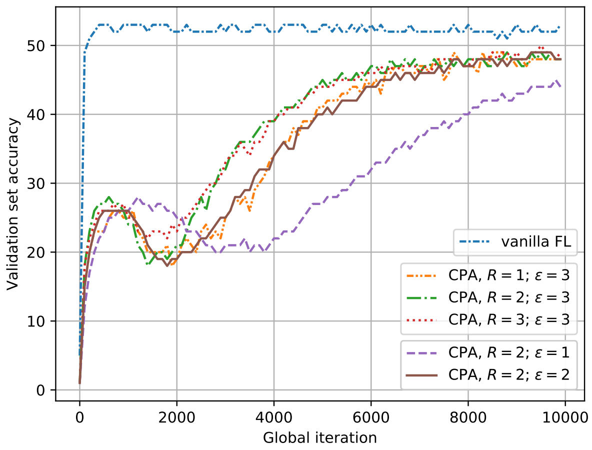

Fig. 10. Convergence profile of CPA for different $R$, $\varepsilon$ values; training a ResNet-18 model using the CIFAR-100 dataset with $K = 10$ edge users.

Unlike the handwritten digit classification, in the current task it is harder for the models to converge, particularly for signSGD & RR, which utterly fails to converge as revealed by Table 4. For the CNN3 model, we can see that the performance of both CPA and JoPEQ is alike, yet the former holds *R1-R5* while the latter only does so for *R1-R3*.

It is noted that when training deep models, adding a minor level of distortion can sometimes improve the final model performance, see, e.g., [75], [76]. Hence, CPA without RR does not necessarily outperform CPA with LDP consideration; as evidenced in Fig 9 and Table 4, having CPA without RR outperforms its nosier CPA counterpart. Nevertheless, the opposite holds in Fig. 8 and Table 5, which consider a simpler (and shallower) linear model. There, the perturbation induced by RR have a consistent harmful effect on the trained model.

To further support the utility of 1-bit CPA regardless of the chosen dataset and/or model architecture, even for small-scale deployments; we depict in Fig. 10 the convergence profile of CPA training ResNet-18 on CIFAR-100 using merely $K = 10$ clients. There, different combinations of the bit-rate $(R)$ and the privacy budget $(\varepsilon)$ are tested and referenced to the performance achieved with vanilla FL, constrained with neither privacy nor compression. The training performed over $10K$ global rounds, each for 3 local iterations, using the *Adam* optimizer [77].

It can be observed that, as expected, vanilla FL performs best and converges fast, having no noise being added to its learning process. CPA, on the other hand, takes longer to converge while it attains a slightly lower accuracy score, which is also attributed to the fact that only few users participate rather than hundreds and thousands of them. Furthermore, Fig. 10 reveals the trade-off between $R$, $\varepsilon$, and accuracy; having the privacy being the more dominant factor to deteriorate the accuracy.

### *Nested Operation*

We next numerically validate the gains of nested CPA. As detailed in Subsection 3.2, nested CPA alleviates the tradeoff between the number of users $K$ and the quantization rate $R$. This is because for a limited number of users, a large number of lattice points typically results in a less accurate estimation of the probability over the lattice points, which in turn translates into degraded models. In Table 5 we can empirically view this behavior and the ability of the nested design to mitigate its harmful effects.

TABLE 5
Test set accuracy and SNR results for 1-bit and nested CPA

| | $K = 10$ | | $K = 100$ | | $K = 1000$ | |
|---|---|---|---|---|---|---|
| | Acc. | SNR | Acc. | SNR | Acc. | SNR |
| CPA | 49 | -17.57 | 81 | -0.07 | 85 | 0.09 |
| nested CPA | 59 | -6.16 | 83 | 0.08 | 86 | 0.17 |
| JoPEQ | 46 | -31.45 | 64 | -19.35 | 70 | -4.45 |
| CPA w/o RR | 60 | -5.09 | 84 | 10.10 | 87 | 15.51 |
| nested CPA w/o RR | 63 | 3.23 | 85 | 12.92 | 87 | 24.18 |

Table 5 summarizes the test accuracy and SNR values obtained for a linear model trained on MNIST for different number of participating clients $K$; for the baselines 1-bit CPA and two-stage nested CPA, broadcasting the server 1 and 2 bits per sample, respectively. These are contrasted with JoPEQ, which utilizes a conventional multi-bit quantizer of rate $R = 2$. There, it is indicated that the higher the number of users is, the lessen the improvement of the nested operation over the 1-bit scheme for both accuracy and SNR metrics. JoPEQ is comparable to nested CPA in terms of bits per sample $(R)$, yet demonstrating an inferior performance, equivalent to that of single-bit CPA for $K = 10$; while the latter is far better in the large-scale scenario, what further supports its benefits in massive deployments.

### *Byzantine Robustness*

We conclude by verifying CPA's toleration under colluding malicious participants. Table 6 reports the test accuracy of the converged models of the datasets and architectures considered. We simulate manipulations of a subset of the users, where it is either the scenario that a user is sending its 1-bit data constantly as '1'; or randomly flipping it; referenced to the result achieved with None. We observe that CNN3 'None' does not necessarily outperform its nosier '1' or 'Flip' counterparts. This phenomenon is similar to the one evidenced in Fig. 9 and Table 4 for CPA with and without RR. CPA's immunity is observed regardless of the

TABLE 6
CPA's test set accuracy with a subset of malicious users K=1000

| Malicious subset | MNIST | | | | | |
|---|---|---|---|---|---|---|
| | Linear | | | MLP | | |
| | None | ’1’s | Flip | None | ’1’s | Flip |
| 20% | 85 | 85 | 85 | 86 | 85 | 84 |
| 30% | 85 | 84 | 84 | 86 | 84 | 84 |
| | CIFAR-10 | | | | | |
| | CNN2 | | | CNN3 | | |
| | None | ’1’s | Flip | None | ’1’s | Flip |
| 20% | 50 | 48 | 48 | 67 | 68 | 69 |
| 30% | 50 | 47 | 48 | 67 | 69 | 66 |

model and/or data chosen as Table 6 indicates a degrade of single percents in accuracy under the simulated attacks. This further ensures that CPA, in addition to being a joint compression and privacy mechanism for massive deployments, also provides robustness to Byzantine adversaries.

## 6 Conclusions

We proposed CPA, which realizes quantization and privacy in scalable and robust FL. CPA combines nested lattice quantization and encoding via a random codebook, with a dedicated RR mechanism and discrete histogram aggregations to yield provable desired privacy and anonymity levels in a manner that is scalable to large networks and resilient to malicious manipulations. Our analysis characterizes the excess distortion induced by CPA and its convergence, showing that it achieves similar asymptotic convergence profile as FL without privacy or compression considerations. We demonstrated that CPA results with less distorted and more reliable models compared to alternative compression and privacy FL methods, while approaching the performance achieved without these constraints and demolishing poisoning attacks.

## References

[1] N. Lang, E. Sofer, N. Shlezinger, R. G. L. D'Oliveira, and S. El Rouayheb, "CPA: Compressed private aggregation for scalable federated learning over massive networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[2] 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.

[3] P. Kairouz *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[4] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, 2020.

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

[6] J. Chen and X. Ran, "Deep learning with edge computing: A review." *Proc. IEEE*, vol. 107, no. 8, pp. 1655–1674, 2019.

[7] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," *Advances in neural information processing systems*, vol. 32, 2019.

[8] B. Zhao, K. R. Mopuri, and H. Bilen, "idlg: Improved deep leakage from gradients," *CoRR*, vol. abs/2001.02610, 2020. [Online]. Available: http://arxiv.org/abs/2001.02610

[9] Y. Huang, S. Gupta, Z. Song, K. Li, and S. Arora, "Evaluating gradient inversion attacks and defenses in federated learning," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[10] H. Yin *et al.*, "See through gradients: Image batch recovery via gradinversion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 337–16 346.

[11] M. Fang, X. Cao, J. Jia, and N. Gong, "Local model poisoning attacks to Byzantine-robust federated learning," in *USENIX Security Symposium*, 2020, pp. 1605–1622.

[12] V. Mothukuri *et al.*, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, 2021.

[13] S. P. Karimireddy *et al.*, "Scaffold: Stochastic controlled averaging for federated learning," in *International conference on machine learning*. PMLR, 2020, pp. 5132–5143.

[14] M. Chen, N. Shlezinger, H. V. Poor, Y. C. Eldar, and S. Cui, "Communication-efficient federated learning," *Proceedings of the National Academy of Sciences*, vol. 118, no. 17, 2021.

[15] M. Kim, O. Günlü, and R. F. Schaefer, "Federated learning with local differential privacy: Trade-offs between privacy, utility, and communication," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 2650–2654.

[16] K. Wei *et al.*, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3454–3469, 2020.

[17] L. Sun, J. Qian, and X. Chen, "Ldp-fl: Practical private aggregation in federated learning with local differential privacy," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 2021.

[18] R. Liu, Y. Cao, M. Yoshikawa, and H. Chen, "Fedsel: Federated SGD under local differential privacy with top-k dimension selection," in *International Conference on Database Systems for Advanced Applications*. Springer, 2020, pp. 485–501.

[19] X. Yin, Y. Zhu, and J. Hu, "A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–36, 2021.

[20] G. A. Kaissis, M. R. Makowski, D. Rückert, and R. F. Braren, "Secure, privacy-preserving and federated machine learning in medical imaging," *Nature Machine Intelligence*, vol. 2, no. 6, pp. 305–311, 2020.

[21] A. Vempaty, L. Tong, and P. K. Varshney, "Distributed inference with Byzantine data: State-of-the-art review on data falsification attacks," *IEEE Signal Process. Mag.*, vol. 30, no. 5, pp. 65–75, 2013.

[22] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *International Conference on Machine Learning*. PMLR, 2018, pp. 5650–5659.

[23] L. Li, W. Xu, T. Chen, G. B. Giannakis, and Q. Ling, "RSA: Byzantine-robust stochastic aggregation methods for distributed learning from heterogeneous datasets," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 1544–1551.

[24] P. Han, S. Wang, and K. K. Leung, "Adaptive gradient sparsification for efficient federated learning: An online learning approach," in *IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2020, pp. 300–310.

[25] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *CoRR*, vol. abs/1610.05492, 2016. [Online]. Available: http://arxiv.org/abs/1610.05492

[26] C. Hardy, E. Le Merrer, and B. Sericola, "Distributed deep learning on edge-devices in the parameter server model," in *Workshop on Decentralized Machine Learning, Optimization and Privacy*, 2017.

[27] A. Aji and K. Heafield, "Sparse communication for distributed gradient descent," in *EMNLP 2017: Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics (ACL), 2017, pp. 440–445.

[28] D. Alistarh *et al.*, "The convergence of sparsified gradient methods," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[29] Y. Lin, S. Han, H. Mao, Y. Wang, and B. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," in *International Conference on Learning Representations*, 2018.

[30] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-efficient SGD via gradient quantization and encoding," *Advances in Neural Information Processing Systems*, vol. 30, pp. 1709–1720, 2017.

[31] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "Fedpaq: A communication-efficient federated learn-

ing method with periodic averaging and quantization," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2021–2031.

[32] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, "signSGD: Compressed optimisation for non-convex problems," in *International Conference on Machine Learning*. PMLR, 2018, pp. 560–569.

[33] N. Shlezinger, M. Chen, Y. C. Eldar, H. V. Poor, and S. Cui, "UVeQFed: Universal vector quantization for federated learning," *IEEE Trans. Signal Process.*, vol. 69, pp. 500–514, 2020.

[34] S. Horvóth, C.-Y. Ho, L. Horvath, A. N. Sahu, M. Canini, and P. Richtárik, "Natural compression for distributed deep learning," in *Mathematical and Scientific Machine Learning*. PMLR, 2022, pp. 129–141.

[35] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *IEEE international conference on communications (ICC)*, 2019.

[36] L. Lyu, "DP-SIGNSGD: When efficiency meets privacy and robustness," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 3070–3074.

[37] Y. Zhang, D. Liu, and O. Simeone, "Leveraging channel noise for sampling and privacy via quantized federated langevin monte carlo," in *IEEE International Workshop on Signal Processing Advances in Wireless Communication (SPAWC)*, 2022.

[38] S. Amiri, A. Belloum, S. Klous, and L. Gommans, "Compressive differentially private federated learning through universal vector quantization," in *AAAI Workshop on Privacy-Preserving Artificial Intelligence*, 2021.

[39] N. Lang, E. Sofer, T. Shaked, and N. Shlezinger, "Joint privacy enhancement and quantization in federated learning," *IEEE Trans. Signal Process.*, vol. 71, pp. 295–310, 2023.

[40] K. Chaudhuri, C. Guo, and M. Rabbat, "Privacy-aware compression for federated data analysis," in *Conference on Uncertainty in Artificial Intelligence*, 2022.

[41] C. Naim, R. G. L. D'Oliveira, and S. El Rouayheb, "Private multigroup aggregation," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 3, pp. 800–814, 2022.

[42] R. Thesmar, J. Thesmar, R. G. L. D'Oliveira, and M. Medard, "Cabdriver: Concentration to accurate boundaries while distorting randomly input variables to elude recognition," in *International ITG Workshop on Smart Antennas*, 2021.

[43] A. Abdi and F. Fekri, "Nested dithered quantization for communication reduction in distributed training," *CoRR*, vol. abs/1904.01197, 2019. [Online]. Available: http://arxiv.org/abs/1904.01197

[44] S. U. Stich, "Local SGD converges fast and communicates little," in *International Conference on Learning Representations*, 2019.

[45] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2325–2383, 1998.

[46] R. Zamir and M. Feder, "On universal quantization by randomized uniform/lattice quantizers," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 428–436, 1992.

[47] S. P. Lipshitz, R. A. Wannamaker, and J. Vanderkooy, "Quantization and dither: A theoretical survey," *Journal of the audio engineering society*, vol. 40, no. 5, pp. 355–375, 1992.

[48] R. M. Gray and T. G. Stockham, "Dithered quantizers," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 805–812, 1993.

[49] R. Zamir and M. Feder, "On lattice quantization noise," *IEEE Trans. Inf. Theory*, vol. 42, no. 4, pp. 1152–1159, 1996.

[50] R. Zamir, S. Shamai, and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1250–1276, 2002.

[51] Y. Polyanskiy and Y. Wu, "Lecture notes on information theory," *Lecture Notes for 6.441 (MIT), ECE563 (University of Illinois Urbana-Champaign), and STAT 664 (Yale)*, 2012-2017.

[52] X. Yang, T. Wang, X. Ren, and W. Yu, "Survey on improving data utility in differentially private sequential data publishing," *IEEE Trans. Big Data*, vol. 7, no. 4, pp. 729–749, 2017.

[53] J. M. Abowd, "The US census bureau adopts differential privacy," in *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2867–2867.

[54] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith, "What can we learn privately?" *SIAM Journal on Computing*, vol. 40, no. 3, pp. 793–826, 2011.

[55] Y. Wang, Y. Tong, and D. Shi, "Federated latent dirichlet allocation: A local differential privacy based framework," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 6283–6290, 2020.

[56] Y. Zhao *et al.*, "Local differential privacy-based federated learning for internet of things," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 8836–8853, 2020.

[57] X. Xiong, S. Liu, D. Li, Z. Cai, and X. Niu, "A comprehensive survey on local differential privacy," *Security and Communication Networks*, vol. 2020, 2020.

[58] T. Wang, X. Zhang, J. Feng, and X. Yang, "A comprehensive survey on local differential privacy toward data statistics and analysis," *Sensors*, vol. 20, no. 24, p. 7030, 2020.

[59] M. Reimherr and J. Awan, "Elliptical perturbations for differential privacy," in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[60] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, 1965.

[61] L. Sweeney, "k-anonymity: A model for protecting privacy," *International journal of uncertainty, fuzziness and knowledge-based systems*, vol. 10, no. 05, pp. 557–570, 2002.

[62] D. Usynin *et al.*, "Adversarial interference and its mitigations in privacy-preserving collaborative machine learning," *Nature Machine Intelligence*, vol. 3, no. 9, pp. 749–758, 2021.

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

[64] C. Dwork, G. N. Rothblum, and S. Vadhan, "Boosting and differential privacy," in *IEEE Annual Symposium on Foundations of Computer Science*, 2010, pp. 51–60.

[65] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*. Springer Science & Business Media, 2013, vol. 290.

[66] J. Bernstein, J. Zhao, K. Azizzadenesheli, and A. Anandkumar, "signSGD with majority vote is communication efficient and fault tolerant," in *International Conference on Learning Representations*, 2019.

[67] P. Kairouz, Z. Liu, and T. Steinke, "The distributed discrete gaussian mechanism for federated learning with secure aggregation," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5201–5212.

[68] N. Agarwal, P. Kairouz, and Z. Liu, "The skellam mechanism for differentially private federated learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 5052–5064, 2021.

[69] 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.

[70] N. Shlezinger, Y. C. Eldar, and M. R. Rodrigues, "Hardware-limited task-based quantization," *IEEE Trans. Signal Process.*, vol. 67, no. 20, pp. 5223–5238, 2019.

[71] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on non-IID data," in *International Conference on Learning Representations*, 2020.

[72] N. Lang, A. Cohen, and N. Shlezinger, "Stragglers-aware low-latency synchronous federated learning via layer-wise model updates," *IEEE Trans. Commun.*, 2025.

[73] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," *Journal of Privacy and Confidentiality*, vol. 7, no. 3, pp. 17–51, 2016.

[74] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[75] G. An, "The effects of adding noise during backpropagation training on a generalization performance," *Neural computation*, vol. 8, no. 3, pp. 643–674, 1996.

[76] T. Sery, N. Shlezinger, K. Cohen, and Y. C. Eldar, "Over-the-air federated learning from heterogeneous data," *IEEE Trans. Signal Process.*, vol. 69, pp. 3796–3811, 2021.

[77] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, 2015.

**Natalie Lang** received her B.Sc. and M.Sc. degrees in 2019 and 2021, respectively, from Ben-Gurion University, Israel, all in Electrical and Computer Engineering. She is currently pursuing a Ph.D. degree in Electrical and Computer Engineering at Ben-Gurion University. Her research interests include data privacy, compression, and the joint design of both for federated learning and signal processing.

**Nir Shlezinger** (M'17-SM'23) is an assistant professor in the School of Electrical and Computer Engineering at Ben-Gurion University, Israel. He received his B.Sc., M.Sc., and Ph.D. degrees in 2011, 2013, and 2017, respectively, from Ben-Gurion University, Israel, all in electrical and computer engineering. From 2017 to 2019, he was a postdoctoral researcher at the Technion, and from 2019 to 2020, he was a postdoctoral researcher at the Weizmann Institute of Science, where he was awarded the FGS Prize for his research achievements. He is the recipient of the 2024 IEEE ComSoc Fred W. Ellersick Award, and the 2024 Krill Prize for outstanding young researchers. His research interests include communications, information theory, signal processing, and machine learning.

**Rafael G. L. D'Oliveira** is an Assistant Professor at the School of Mathematical and Statistical Sciences at Clemson University. He received a B.S. and an M.S. degree in mathematics and a Ph.D. degree in applied mathematics from the University of Campinas in Brazil in 2009, 2012, and 2017. He was a postdoctoral research associate with the Research Laboratory of Electronics at the Massachusetts Institute of Technology from 2020 to 2022, with Rutgers University from 2018 to 2019, and with the Illinois Institute of Technology in 2017. He did a research internship at Telecom Paristech from 2015 to 2016. His research interests include Privacy and Security, Distributed Computing, Coding Theory, and Information Theory.

**Salim El Rouayheb** (Senior Member, IEEE) received the Diploma degree in electrical engineering from the Faculty of Engineering, Lebanese University, Roumieh, Lebanon, in 2002, the M.S. degree from the American University of Beirut, Lebanon, in 2004, and the Ph.D. degree in electrical engineering from Texas A&M University, College Station, in 2009. He was a Post-Doctoral Research Fellow with UC Berkeley from 2010 to 2011 and a Research Scholar with Princeton University from 2012 to 2013. He was an Assistant Professor with the ECE Department, Illinois Institute of Technology, from 2013 to 2017. In 2019, he held the Walter Tyson Junior Faculty Chair at Rutgers University, where he is now an Associate Professor in the ECE Department. He received the Google Faculty Award in 2018 and the NSF CAREER Award in 2016. His research focuses on information-theoretic security and data privacy, with an emphasis on distributed learning.