Digital Over-the-Air Federated Learning in Multi-Antenna Systems

Sihua Wang, *Student Member, IEEE*, Mingzhe Chen, *Senior Member, IEEE*, Cong Shen, *Senior Member, IEEE*, Changchuan Yin, *Senior Member, IEEE*, and Christopher G. Brinton, *Senior Member, IEEE*

***Abstract*— In this paper, the performance optimization of federated learning (FL), when deployed over a realistic wireless multiple-input multiple-output (MIMO) communication system with digital modulation and over-the-air computation (AirComp) is studied. In particular, a MIMO system is considered in which edge devices transmit their local FL models (trained using their locally collected data) to a parameter server (PS) using beamforming to maximize the number of devices scheduled for transmission. The PS, acting as a central controller, generates a global FL model using the received local FL models and broadcasts it back to all devices. Due to the limited bandwidth in a wireless network, AirComp is adopted to enable efficient wireless data aggregation. However, fading of wireless channels can produce aggregate distortions in an AirComp-based FL scheme. To tackle this challenge, we propose a modified federated averaging (FedAvg) algorithm that combines digital modulation with AirComp to mitigate wireless fading while ensuring the communication efficiency. This is achieved by a joint transmit and receive beamforming design, which is formulated as an optimization problem to dynamically adjust the beamforming matrices based on current FL model parameters so as to minimize the transmitting error and ensure the FL performance. To achieve this goal, we first analytically characterize how the beamforming matrices affect the performance of the FedAvg in different iterations. Based on this relationship, an artificial neural network (ANN) is used to estimate the local FL models of all devices and adjust the beamforming matrices at the PS for future model transmission. The algorithmic advantages and improved performance of the proposed methodologies are demonstrated through extensive numerical experiments.**

***Index Terms*— Federated learning, MIMO, AirComp, digital modulation.**

Manuscript received 3 February 2023; revised 29 August 2023, 19 January 2024, and 24 April 2024; accepted 28 June 2024. Date of publication 19 July 2024; date of current version 11 October 2024. This work was supported in part by Beijing Natural Science Foundation under Grant L223027, in part by the National Natural Science Foundation of China under Grant 61629101, in part by the China 973 Program under Grant 2009CB320407, and in part by under Grant NSF CNS-2146171 and Grant CPS-2313109. An earlier version of this paper was presented at the 2023 IEEE Global Communications Conference (GLOBECOM) [DOI: 10.1109/GLOBECOM54140.2023.10436780]. The associate editor coordinating the review of this article and approving it for publication was H. Yang. *(Corresponding author: Changchuan Yin.)*

Sihua Wang and Changchuan Yin are with Beijing Laboratory of Advanced Information Network and Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: sihuawang@bupt.edu.cn; ccyin@ieee.org).

Mingzhe Chen is with the Department of Electrical and Computer Engineering and the Institute for Data Science and Computing, University of Miami, Coral Gables, FL 33146 USA (e-mail: mingzhe.chen@miami.edu).

Cong Shen is with the Charles L. Brown Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22903 USA (e-mail: cong@virginia.edu).

Christopher G. Brinton is with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47906 USA (e-mail: cgb@purdue.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2024.3425732.

Digital Object Identifier 10.1109/TWC.2024.3425732

I. Introduction

FEDERATED learning (FL) has been extensively studied as a distributed machine learning approach with data privacy [1], [2], [3], [4], [5], [6], [7]. During the FL training process, edge devices are required to train a local learning model using its collected data and transmit the trained learning model to a parameter server (PS) for global model aggregation. The PS, acting as a central center, can coordinate the process across edge devices and broadcast the global model to all devices. This procedure is repeated across several rounds until achieving an acceptable accuracy of the trained model.

Since the PS and edge devices must exchange their trained models iteratively over the wireless channels, FL performance can be significantly affected by imperfect and dynamic wireless transmission in both uplink and downlink. Compared to the PS broadcasting FL models to edge devices, edge devices uploading local models to the PS is more challenging due to their limited transmit power [8], [9], [10], [11], [12]. To tackle this challenge, over-the-air computation (also known as AirComp) techniques have recently been integrated into the implementation of FL [13], [14], [15], [16], [17]. Instead of decoding the individual local models of each device and then aggregating, AirComp allows edge devices to transmit their model parameters simultaneously over the same radio resources and decode the average model (global model) directly at the PS [18], [19], [20]. However, most of these existing works, such as [21] and [22], focused on the use of AirComp for analog modulation due to its simplicity for FL convergence analysis, which may not be desirable for practical wireless communication systems that almost exclusively use digital modulations. In consequence, it is necessary to study the implementation of AirComp-based FL over digital modulation-based wireless systems.

A. Related Works

Recent works such as [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], and [33] have studied several important

problems related to the implementation of AirComp-based FL over wireless networks. The authors in [23] minimized the mean-squared error (MSE) of the FL model during AirComp transmission under transmit power constraints in a multiuser multiple-input multiple-output (MIMO) system. In [24], the authors maximized the number of devices that can participate in FL training under certain MSE requirements in an AirComp-based MIMO framework. A joint machine learning rate and receiver beamforming matrix optimization method was proposed in [25] to reduce the aggregate distortion and satisfy an FL performance requirement. The authors in [26] investigated the deployment of FL over an AirComp-based wireless network to minimize the energy consumption of edge devices. In [27], the authors optimized the set of participating devices in an AirComp-assisted FL framework to speed up FL convergence. A receive beamforming scheme was designed in [28] to optimize FL performance without knowing channel state information (CSI). The authors in [29] minimized the FL model aggregation error under a channel alignment constraint in a MIMO system. The authors in [30] derived the optimal threshold-based regularized channel inversion power control solution to minimize the mean squared error (MSE) for an analog AirComp system with imperfect CSI. A channel inversion-based power control policy was developed in [31] to resist channel fading and meet the SNR threshold in an AirComp FL framework. In [32], the authors minimized the computation error by jointly optimizing the transmit power at devices and a signal scaling factor at the PS. The authors in [33] minimized the transmit power of each device while ensuring a minimum MSE performance. One key challenge faced by these works is their limited practical applicability to high-order digital modulation-based wireless systems. By harnessing high-order digital modulation, individual devices can encode FL model parameters into discrete symbols and transmit multiple symbols simultaneously within the same frequency band. This approach optimizes bandwidth utilization and enhances data transmission efficiency. Moreover, digital modulation leverages error correction coding and modulation schemes such as quadrature amplitude modulation (QAM) to enhance resistance against noise and interference. Hence, it is important to note that these works do not explicitly consider the incorporation of coding and digital modulation techniques, which are crucial components in real-world wireless communication systems.

Recently, several works [34], [35], [36], [37], [38], [39], [40], [41], [42] have studied the implementation of AirComp FL over digital modulation-based wireless systems. The authors in [34] designed one-bit quantization and modulation schemes for edge devices. One-bit gradient quantization scheme is proposed in [35] to achieve fast FL model aggregation. In [36], the authors designed a joint channel decoding and aggregation decoding scheme based on binary phase shift keying (BPSK) modulation for AirComp FL. The authors in [37] evaluated the performance of FL gradient quantization in digital AirComp. In [38], the convergence of FL implemented over an AirComp-based MIMO system is derived. The authors in [39] proposed a digital transmission protocol tailored to FL over wireless device-to-device networks. In [40], the authors proposed an AirComp method that utilizes non-coherent detection and digital modulation to achieve high test accuracy. The authors in [41] adopted BPSK for over-the-air computations to minimize the normalized MSE. A joint transmission and local computing strategy was designed in [42] that utilizes multiple amplitude shift keying (MASK) symbols to minimize the energy consumption used for FL training. However, these prior works [34], [35], [36], [37], [38], [39], [40], [41], [42] mainly used low order digital modulation (i.e., BPSK) and hence their designed AirComp FL cannot be easily extended to modern wireless systems that use high-order digital modulation schemes such as QAM. This is because the transmitted symbols that are processed by low-order digital modulation (such as the symbols -1 and $+1$ in BPSK) are linearly superimposed. This linear superimposition does not exist in high-order digital modulation schemes with complex mapping relationships between bits and symbols (such as Gray code).

B. Contributions

The main contribution of this paper is to develop a novel AirComp FL framework over high-order digital modulation-based wireless systems. Our key contributions include:

- We propose a novel AirComp-based MIMO system in which distributed wireless devices utilize high-order modulations to encode their trained local FL parameters into symbols and simultaneously transmit these modulated symbols over unreliable wireless channels to a PS that directly generates the global FL model via its received symbols. However, the introduction of high-order digital modulations leads to the loss of linearity and superpositivity among the symbols transmitted by each device, which in turn affects the convergence of the trained FL model and poses challenges to FL performance. To tackle this issue, the PS and devices must cooperatively adjust the transmit and receive beamforming matrices to capture the nonlinearity relationship among the modulated symbols and improve FL performance. To this end, we formulate this joint transmit and receive beamforming matrix design problem as an optimization problem whose goal is to minimize the FL training loss.
- To solve this problem, we first analytically characterize how the errors introduced by the proposed AirComp system affect FL training loss. Our analysis shows that the introduced errors caused by wireless transmission (i.e., fading and additive white Gaussian noise) and digital post-processing (i.e., digital demodulation) determine the gap between the optimal FL model that the FL targets to converge and the trained FL model. In particular, the errors caused by wireless transmission depend on the channel conditions and the trained FL model parameters. However, the errors caused by digital post-processing depend on the adopted modulation scheme and the number of devices participated in FL training. Hence, to minimize the errors caused by both wireless transmission and digital post-processing, the PS and the devices must dynamically adjust the transmit and receive beamforming matrices based on the adopted modulation

scheme, the trained FL model parameters, and channel conditions.

- To find the optimal transmit and receive beamforming matrices, we first introduce an artificial neural network (ANN)-based algorithm to predict the FL model parameters of all devices since optimizing beamforming matrices requires the information of each trained local model parameter which cannot be obtained by the PS. Then, given the predicted parameters, we derive a closed-form solution of the optimal transmit and receive beamforming matrices based on the adopted modulation scheme and channel conditions that minimize the distance between the received signals of all devices and the predicted parameters in the decision region, which ensures the accuracy for model aggregation and FL performance. We show that the added latency introduced by the ANN inference is marginal compared to the improvement in convergence speed provided by our beamforming optimization.
- Numerical evaluation results on real-world machine learning task datasets show that our proposed AirComp-based system can improve the test accuracy by 10%-30% compared to the AirComp-based system with analog modulation and BPSK, respectively.

The rest of this paper is organized as follows. The system model and problem formulation for the AirComp-based system in FL framework are described in Section II. Section III analyzes the convergence of the designed FL framework and derives a closed-form optimal design of the transmit and receive beamforming matrices based on the analysis. In Section IV, our numerical evaluation is presented and discussed. Finally, conclusions are drawn in Section V.

II. System Model and Problem Formulation

We consider an FL system implemented over a cellular network, where K wireless edge devices train their individual machine learning models and send the machine learning parameters to a central PS through a noisy wireless channel by using a media access control (MAC) protocol as shown in Fig. 1. In the considered model, the PS is equipped with N_r antennas while each device k is equipped with N_t antennas.

Each device has N_k training data samples and each training data sample n in device k consists of an input feature vector $\boldsymbol{x}_{k,n} \in \mathbf{R}^{N_I \times 1}$ and a corresponding label vector $\boldsymbol{y}_{k,n} \in \mathbf{R}^{N_O \times 1}$ where N_I and N_O are the dimension of input and output vectors, respectively. Table I provides a summary of the notations used throughout this paper. The objective of the training is to minimize the global loss function over all data samples, which is given by

$$F(\boldsymbol{g}) = \min_{\boldsymbol{g}} \frac{1}{N} \sum_{k=1}^{K} \sum_{n=1}^{N_k} f\left(\boldsymbol{g}, \boldsymbol{x}_{k,n}, \boldsymbol{y}_{k,n}\right), \tag{1}$$

where $\boldsymbol{g} \in \mathbf{R}^{V \times 1}$ is a vector that represents the global FL model of dimension V trained across the devices with $N = \sum_{k=1}^{K} N_k$ being the total number of training data samples of all devices. $f\left(\boldsymbol{g}, \boldsymbol{x}_{k,n}, \boldsymbol{y}_{k,n}\right)$ is the local loss function of each device k with FL model $\boldsymbol{g}$ and data sample $\left(\boldsymbol{x}_{k,n}, \boldsymbol{y}_{k,n}\right)$.

TABLE I
List of Notations

Notation	Description
K	Number of devices
M	Adopted modulation order
N_r	Number of antennas on the PS
N_t	Number of antennas on devices
N_k	Number of training data samples on device k
$(\boldsymbol{x}_{k,n}, \boldsymbol{y}_{k,n})$	Training data sample n on device k
N	Number of training data samples of all devices
$\boldsymbol{g}_t$	Global FL model
$\boldsymbol{w}_{k,t}$	Local FL model
$\Delta\boldsymbol{w}_{k,t}$	Updates of $\boldsymbol{w}_{k,t}$
$\hat{\boldsymbol{w}}_{k,t}$	Modulated symbol vector of $\boldsymbol{w}_{k,t}$
$\Delta\widetilde{\boldsymbol{w}}_{k,t}$	Prediction of $\Delta\boldsymbol{w}_{k,t}$
$\Delta\hat{\boldsymbol{w}}_{k,t}$	Modulated symbol vector of $\Delta\widetilde{\boldsymbol{w}}_{k,t}$
$\boldsymbol{n}_t$	Additive white Gaussian noise
$\boldsymbol{A}_{k,t}$	Transmit beamforming matrix
$\boldsymbol{B}_t$	Receive beamforming matrix
$\boldsymbol{H}_k$	Channel vector between device k and the PS
P_0	Maximal transmit power on device
ξ	Minimum Euclidean distance in decision region
$a_i^{\mathrm{I}}, a_i^{\mathrm{Q}}$	Constellation point of symbol i
$\boldsymbol{a}^*$	Vector of predicted constellation point
$\mathcal{M}$	Set of all constellation points

To minimize the global loss function in (1) in a distributed manner, each device can update its FL model using its local dataset with a backward propagation (BP) algorithm based on stochastic gradient descent (SGD), which can be expressed as

$$\boldsymbol{w}_{k,t} = \boldsymbol{g}_t - \frac{\lambda}{|\mathcal{N}_{k,t}|} \sum_{n \in \mathcal{N}_{k,t}} \frac{\partial f\left(\boldsymbol{g}, \boldsymbol{x}_{k,n}, \boldsymbol{y}_{k,n}\right)}{\partial \boldsymbol{g}}, \tag{2}$$

where λ is the learning rate, $\mathcal{N}_{k,t}$ is the subset of training data samples (i.e., minibatch) selected from device k's training dataset $\mathcal{N}_k$ at iteration t with $|\mathcal{N}_{k,t}|$ being the number of data samples in $\mathcal{N}_{k,t}$, and $\boldsymbol{w}_{k,t}$ is the updated local FL model of device k at iteration t. Here, each device must perform $\frac{|\mathcal{N}_k|}{|\mathcal{N}_{k,t}|}$ local updates to traverse the entire local training dataset.

Given $\boldsymbol{w}_{k,t}$, distributed devices must simultaneously exchange their model parameters with the PS via bandwidth-limited wireless fading channels for model aggregation. The equation of model aggregation is given by

$$\boldsymbol{g}_t = \sum_{k=1}^{K} \frac{|\mathcal{N}_k|}{N} \boldsymbol{w}_{k,t}, \tag{3}$$

where $|\mathcal{N}_k|$ represents the number of data samples in $\mathcal{N}_k$.

To ensure all devices can participate in FL model exchanging via wireless fading channels, each device adopts digital modulation to mitigate wireless fading and the PS adopts beamforming to maximize the number of devices scheduled for FL parameter transmission. Next, we will mathematically introduce the FL training and transmission process integrated with digital modulation in the considered MIMO communication system. In particular, we first introduce our designed digital modulation process that consists of two steps: (i) digital pre-processing at the devices and (ii) digital post-processing at the PS.

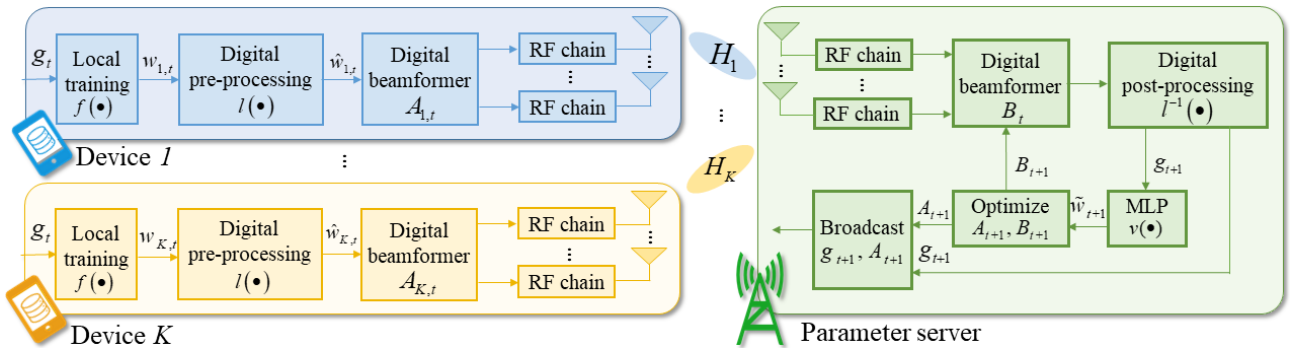


Fig. 1. Illustration of our methodology and system model. An FL algorithm is deployed over multiple devices and one PS in a MIMO communication system. We design the transmit and receive beamforming matrices to optimize the FL training process.

A. Digital Pre-Processing at the Devices

To transmit $\boldsymbol{w}_{k,t}$ over wireless fading channels, each device k leverages digital pre-processing to represent each numerical FL parameter in $\boldsymbol{w}_{k,t}$ using a symbol vector, which is

$$\hat{\boldsymbol{w}}_{k,t} = l\left(\boldsymbol{w}_{k,t}\right), \tag{4}$$

where $\hat{\boldsymbol{w}}_{k,t} \in \mathbf{R}^W$ is a modulated symbol vector with W being the number of symbols, and $l(\cdot)$ denotes the digital pre-processing function that combines decimal-to-binary conversion and digital modulation, where the decimal-to-binary conversion is used to represent each numerical FL parameter with a binary coded bit-interleaved vector, and the digital modulation is used to modulate several binary bits as a symbol [43]. For convenience, the modulated signal $\hat{\boldsymbol{w}}_{k,t}$ is normalized (i.e., $|\hat{\boldsymbol{w}}_{k,t}| = 1$). We use rectangular M-QAM for digital modulation and it can be extended to other types of digital modulation schemes.

Given the transmit beamforming matrix $\boldsymbol{A}_{k,t} \in \mathbf{C}^{N_t \times W}$ and the maximal transmit power P_0 at device k, the power constraint can be expressed as [44], [45], and [46]

$$\mathbb{E}\left(\left|\boldsymbol{A}_{k,t}\hat{\boldsymbol{w}}_{k,t}\right|^2\right) = \left|\boldsymbol{A}_{k,t}\right|^2 \leq P_0. \tag{5}$$

where $\mathbb{E}(x)$ represents the expectation of x and the equality in (5) is achieved since the modulated signal is normalized.

B. Post-Processing at the PS

Considering the multiple access channel property of wireless communication, the received signal at the PS is given by

$$\boldsymbol{s}_t\left(\boldsymbol{A}_t\right) = \sum_{k=1}^{K} \boldsymbol{H}_k \boldsymbol{A}_{k,t} \hat{\boldsymbol{w}}_{k,t} + \boldsymbol{n}_t \tag{6}$$

where $\boldsymbol{A}_t = [\boldsymbol{A}_{1,t}, \cdots, \boldsymbol{A}_{K,t}]$ denotes the transmit beamforming matrices of all devices, $\boldsymbol{H}_k \in \mathbf{C}^{N_r \times N_t}$ denotes the MIMO channel vector for the link from device k to the PS, and $\boldsymbol{n}_t \in \mathbf{C}^{N_r}$ is the complex additive Gaussian noise with zero mean and identity covariance matrix scaled by the noise power σ^2, i.e., $\boldsymbol{n}_t \sim \mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{I})$.

Since $\boldsymbol{s}_t\left(\boldsymbol{A}_t\right)$ is the weighted sum of all users' local FL models, we consider directly generating the global FL model $\boldsymbol{g}_{t+1}$ from $\boldsymbol{s}_t(\boldsymbol{A}_t)$. This is a major difference between the existing works and this work. The digital beamformer output signal can be expressed as

$$\hat{\boldsymbol{s}}_t\left(\boldsymbol{B}_t, \boldsymbol{A}_t\right) = \boldsymbol{B}_t{}^{\mathrm{H}} \boldsymbol{s}_t(\boldsymbol{A}_t), \tag{7}$$

where $\boldsymbol{B}_t \in \mathbf{C}^{N_r \times W}$ is the digital receive beamforming matrix.

Given the received symbol vector $\hat{\boldsymbol{s}}_t\left(\boldsymbol{B}_t, \boldsymbol{A}_t\right)$, the PS can reconstruct the numerical parameters in global FL model $\boldsymbol{g}_{t+1}$, which can be expressed as

$$\boldsymbol{g}_{t+1}\left(\boldsymbol{B}_t, \boldsymbol{A}_t\right) = l^{-1}\left(\hat{\boldsymbol{s}}_t\left(\boldsymbol{B}_t, \boldsymbol{A}_t\right)\right), \tag{8}$$

where $l^{-1}(\cdot)$ is the inverse function with respect to $l(\cdot)$ that combines the binary-to-decimal function and the digital demodulation function. From (7) and (8), we see that the designed transmit and receive beamforming matrices enable the PS and devices to collaboratively adjust the weights of the transmitted and received signals, thus achieving FL model aggregation.

C. Problem Formulation

Next, we introduce our optimization problem. Our goal is to minimize the FL training loss by designing the transmit and receive beamforming matrices under the total transmit power constraint of each device, which is formulated as follows:

$$\min_{\boldsymbol{B},\boldsymbol{A}} \quad F\left(\boldsymbol{g}\left(\boldsymbol{B}_T, \boldsymbol{A}_T\right)\right), \tag{9}$$

$$\text{s.t.} \quad \left|\boldsymbol{A}_{k,t}\right|^2 \leq P_0, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}. \tag{9a}$$

where $\boldsymbol{A} = [\boldsymbol{A}_1, \ldots, \boldsymbol{A}_T]$ and $\boldsymbol{B} = [\boldsymbol{B}_1, \ldots, \boldsymbol{B}_T]$ are the transmit and receive beamforming matrices for all iterations, respectively. T is a constant which is large enough to guarantee the convergence of FL.

From (9), we can see that the FL training loss $F\left(\boldsymbol{g}\left(\boldsymbol{B}_T, \boldsymbol{A}_T\right)\right)$ depends on the global FL model $g\left(\boldsymbol{B}_T, \boldsymbol{A}_T\right)$ that is trained iteratively. Meanwhile, as shown in (6) and (7), edge devices and the PS must dynamically adjust $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ based on current FL model parameters to minimize the gradient deviation caused by AirComp in the considered MIMO system with digital modulation. However, the PS does not know the gradient vector of each edge device and hence the PS cannot proactively adjust the receive beamforming matrix using traditional optimization algorithms. To tackle this challenge, we propose an ANN-based algorithm that enables

the PS to predict the local FL gradient parameters of each device. Based on the predicted local FL model parameters, the PS and edge devices can cooperatively optimize the beamforming matrices to improve the performance of FL. Next, we first mathematically analyze the FL update process in the considered AirComp-based system to capture the relationship between the beamforming matrix design and the FL training loss per iteration. Based on this relationship, we then derive the closed-form solution of optimal $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ that depends on the predicted FL models achieved by an ANN-based algorithm.

III. Solution for Problem (9)

To solve (9), we first analyze the convergence of the considered FL so as to find the relationship between digital beamforming matrices $\boldsymbol{A}_t$, $\boldsymbol{B}_t$, and FL training loss in (9). The analytical result shows that the optimization of beamforming matrices $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ depends on the FL parameters transmitted by each device. However, the PS does not know these FL parameters since it must determine the beamforming matrices $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ *before* the FL parameter transmission. Therefore, we propose to use neural networks to predict the local FL models of each device and proactively determine the beamforming matrices using these predicted FL parameters.

A. Convergence Analysis of Designed FL

We first analyze the convergence of the considered FL system. Since the update of the global FL model depends on the instantaneous signal-to-interference-plus-noise ratio (SINR) affected by the digital beamforming matrices $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$, we can only analyze the expected convergence rate of FL. To analyze the expected convergence rate of FL, we first assume that a) the loss function $F(\boldsymbol{g})$ is $L-$smooth with the Lipschitz constant $L > 0$, b) $F(\boldsymbol{g})$ is strongly convex with positive parameter μ, c) $F(\boldsymbol{g})$ is twice-continuously differentiable, and d) $\|\nabla f(\boldsymbol{g}_t, \boldsymbol{x}_{kn}, \boldsymbol{y}_{kn})\|^2 \leq \zeta_1 + \zeta_2 \|\nabla F(\boldsymbol{g}_t)\|^2$, as done in [47]. These assumptions can be satisfied by several widely used loss functions such as mean squared error, logistic regression, and cross entropy. Based on these assumptions, next, we first derive the upper bound of the FL training loss at one FL training step. The expected convergence rate of the designed FL algorithm can now be obtained by the following theorem.

Theorem 1: Given the optimal global FL model $\boldsymbol{g}^*$, the current global FL model $\boldsymbol{g}_t$, the transmit beamforming matrix $\boldsymbol{A}_t$, and the receive beamforming matrix $\boldsymbol{B}_t$, $\mathbb{E}\left(F\left(\boldsymbol{g}_{t+1}(\boldsymbol{A}_t, \boldsymbol{B}_t)\right) - F\left(\boldsymbol{g}^*\right)\right)$ can be upper bounded as

$$\begin{aligned}&\mathbb{E}\left(F\left(\boldsymbol{g}_{t+1}(\boldsymbol{A}_t, \boldsymbol{B}_t)\right) - F\left(\boldsymbol{g}^*\right)\right)\\ &\leqslant \mathbb{E}\left(F\left(\boldsymbol{g}_t\right) - F\left(\boldsymbol{g}^*\right)\right) - \frac{1}{2L}\left\|\nabla F\left(\boldsymbol{g}_t\right)\right\|^2\\ &\quad + \frac{1}{2L}\mathbb{E}\left(\|\boldsymbol{e}_t\| + \|\hat{\boldsymbol{e}}_t(\boldsymbol{A}_t, \boldsymbol{B}_t)\|\right)^2,\end{aligned} \tag{10}$$

where

$$\boldsymbol{e}_t = \left\| \frac{\sum_{k=1}^{K} \sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)}{\sum_{k=1}^{K} |\mathcal{N}_{k,t}|} - l^{-1}\left(\frac{\sum_{k=1}^{K} l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)}{\sum_{k=1}^{K} |\mathcal{N}_{k,t}|} \right) \right\| + \Upsilon \tag{11}$$

with the first term being the gradient trained by SGD, the second term being the gradient demodulated from a sum of all selected devices' symbols, and Υ being the variance bound introduced by random sampling of mini-batches of all devices. In particular, Υ depends on the variance of the local updated gradient in minibatch SGD, which is [48], [49], [50]

$$\mathbb{E}||\widehat{\boldsymbol{w}}_{k,t} - \boldsymbol{w}_{k,t}|| = \frac{\sigma_k^2}{|\mathcal{N}_k|}, \tag{12}$$

where $\widehat{\boldsymbol{w}}_{k,t}$ is the optimal updated local model without variance and σ_k^2 is the variance of the local gradient. Given $\frac{\sigma_k^2}{|\mathcal{N}_k|}$ on each device k, the relationship between Υ and $\frac{\sigma_k^2}{|\mathcal{N}_k|}$ is

$$\Upsilon = \sum_{k=1}^{K} \frac{\sigma_k^2}{|\mathcal{N}_k|}, \tag{13}$$

and

$$\begin{aligned}&\hat{\boldsymbol{e}}_t(\boldsymbol{A}_t, \boldsymbol{B}_t)\\ &= l^{-1}\left(\frac{\sum_{k=1}^{K} l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)}{\sum_{k=1}^{K} |\mathcal{N}_{k,t}|} \right)\\ &\quad - l^{-1}\left(\frac{\boldsymbol{B}_t\left(\sum_{k=1}^{K} \boldsymbol{H}_k \boldsymbol{A}_{k,t} l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right) + \boldsymbol{n}_t\right)}{\sum_{k=1}^{K} |\mathcal{N}_{k,t}|} \right).\end{aligned} \tag{14}$$

Proof: See Appendix A. ∎

From Theorem 1, we see that, since $\boldsymbol{e}_t$ does not depend on $\boldsymbol{A}_t$ or $\boldsymbol{B}_t$, the optimization of the digital beamforming matrices cannot minimize $\boldsymbol{e}_t$. In consequence, we can only minimize $\|\hat{\boldsymbol{e}}_t\|$ to decrease the gap between the FL training loss at iteration $t+1$ and the optimal FL training loss (i.e., $\mathbb{E}\left(F\left(\boldsymbol{g}_{t+1}\right) - F\left(\boldsymbol{g}^*\right)\right)$). Thus, problem (9) can be rewritten as

$$\min_{\boldsymbol{B}_t, \boldsymbol{A}_t} \quad \|\hat{\boldsymbol{e}}_t\|^2 \tag{15}$$

$$\text{s.t.} \quad |\boldsymbol{A}_{k,t}|^2 \leq P_0, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}. \tag{15a}$$

To minimize $\|\hat{\boldsymbol{e}}_t\|$ in (15), the PS and edge devices must obtain the information of MIMO channel vector $\boldsymbol{H}_k$ as well as the trained gradients $l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)$ so as to adjust $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$. However, the trained FL gradients $\Delta\boldsymbol{w}_{k,t} = \sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)$ cannot be obtained by the PS before edge devices sending FL model parameters. Hence, the PS must predict $\Delta\boldsymbol{w}_{k,t}$ for optimizing $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ and minimizing $\|\hat{\boldsymbol{e}}_t\|$.

B. Prediction of Local FL Models

Next, we explain the use of neural networks to predict the local FL model updates of all devices at the PS. Formally, the task is to predict parameters of device k's FL model update at time t, i.e., $\Delta \boldsymbol{w}_{k,t} \in \mathbf{R}^{V\times 1}$, from the global aggregation available at the PS at time $t-1$, i.e., $\boldsymbol{g}_{t-1} \in \mathbf{R}^{V\times 1}$. This is considered as a regression task since the values of the FL model are continuous. Therefore, we propose to use multilayer perceptrons (MLPs), a standard type of artificial neural network (ANN) for handling regression tasks [51].

Our proposed MLP-based prediction algorithm consists of three layers: (a) input layer, (b) a single hidden layer, and (c) output layer. These components are defined as follows:

- *Input layer:* The input to the MLP is a vector $\boldsymbol{g}'_{t-1} \in \mathbf{R}^{V'\times 1}$ that represents the previous aggregated results of the FL parameters being predicted. This is a subset of $\boldsymbol{g}_{t-1} \in \mathbf{R}^{V\times 1}$, where the number of parameters $V' \leq V$ is adjustable. Since the total parameter dimension V may be large (e.g., for the CIFAR-10 experiments in Sec. IV, $V = 1.17\times 10^7$), selecting a smaller number V' to predict can lead to complexity reduction advantages. As we mentioned in (6), all devices are able to connect with the PS so as to provide the input information for the MLP to predict the local FL models for next iteration.
- *Output layer:* The output of the MLP is a vector $\Delta\widetilde{\boldsymbol{w}}'_{k,t} \in \mathbf{R}^{V'\times 1}$ that represents prediction of the V' parameters in device k's local FL model update for the current iteration t. The resulting estimate $\Delta\widetilde{\boldsymbol{w}}_{k,t} \in \mathbf{R}^{V\times 1}$ of $\Delta \boldsymbol{w}_{k,t}$ used in the beamforming design in Section III-C is then a combination of (a) the V' parameter predictions $\Delta\widetilde{\boldsymbol{w}}'_{k,t}$ that the ANN has made, and (b) the other $V-V'$ parameters in $\boldsymbol{g}_{t-1}$ that are not part of the ANN.
- *Single hidden layer:* We employ a single hidden layer of dimension D to learn the nonlinear relationships between the input $\boldsymbol{g}'_{t-1}$ and the output $\Delta\widetilde{\boldsymbol{w}}'_{k,t}$. The weight matrix between the input vector and the neurons in the hidden layer for device k is denoted $\boldsymbol{v}_k^{\text{in}} \in \mathbf{R}^{V'\times D}$. Meanwhile, the weight matrix between the neurons in the hidden layer and the output vector is denoted $\boldsymbol{v}_k^{\text{out}} \in \mathbf{R}^{D\times V'}$.

Based on this, the states of the neurons in the hidden layer are given by

$$\boldsymbol{v}_{k,t} = \sigma\left(\boldsymbol{v}_k^{\text{in}}\boldsymbol{g}'_{t-1} + \boldsymbol{b}_k^{v}\right), \tag{16}$$

where $\sigma(x) = \frac{2}{1+\exp(-2x)} - 1$ is a sigmoidal activation and $\boldsymbol{b}_k^{v} \in \mathbf{R}^{D'\times 1}$ is the bias vector. Then, the output of the MLP is given by

$$\Delta\widetilde{\boldsymbol{w}}'_{k,t} = \boldsymbol{v}_k^{\text{out}}\boldsymbol{v}_{k,t} + \boldsymbol{b}_k^{o}, \tag{17}$$

where $\boldsymbol{b}_k^{o} \in \mathbf{R}^{V'\times 1}$ is another bias vector.

The MLP is trained through an online gradient descent method in parallel with the FL model updating. However, in the considered model, the PS only has access to the value of $\boldsymbol{g}_t$ that is directly demodulated from the received signal from all devices. Hence, the PS and the devices must exchange information to train the MLP for each device cooperatively. In particular, at each iteration, device k first calculates its local update $\Delta \boldsymbol{w}_{k,t}$ and $\boldsymbol{g}_{t-1}$ received from the PS, and extracts $\Delta \boldsymbol{w}'_{k,t} \in \mathbf{R}^{V'\times 1}$ from $\Delta \boldsymbol{w}_{k,t}$. Then, it calculates the gradient of the MLP parameters with respect to the MLP's prediction of $\Delta \boldsymbol{w}'_{k,t}$ from $\boldsymbol{g}'_{t-1}$. Finally, device k transmits these gradients to the PS, which updates the MLP parameters accordingly. We will see in Sec. IV that the per-iteration latency introduced by this MLP procedure is overcome by the reduction in the number of training rounds needed for convergence, so long as the size of the MLP is limited.

C. Optimization of the Beamforming Matrices

Having the predicted local FL model updates $\Delta\widetilde{\boldsymbol{w}}_{k,t}$, the PS can optimize the beamforming matrices $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ to solve Problem (15). Substituting $\Delta\widetilde{\boldsymbol{w}}_{k,t}$, (6), and (7) into (15), we have

$$\min_{\boldsymbol{B}_t,\boldsymbol{A}_t} \left\| l^{-1}\left(\frac{\sum_{k=1}^{K} l\left(\Delta\widetilde{\boldsymbol{w}}_{k,t}\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right) - l^{-1}\left(\frac{\boldsymbol{B}_t\left(\sum_{k=1}^{K}\boldsymbol{H}_k\boldsymbol{A}_{k,t}l\left(\Delta\widetilde{\boldsymbol{w}}_{k,t}\right)+\boldsymbol{n}_t\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right)\right\|^2 \tag{18}$$

$$\text{s.t. } |\boldsymbol{A}_{k,t}|^2 \leq P_0, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}. \tag{18a}$$

In (18), $l^{-1}\left(\frac{\sum_{k=1}^{K} l(\Delta\widetilde{\boldsymbol{w}}_{k,t})}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right)$ is independent of $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ and can be regarded as a constant. However, the existence of the inverse function $l^{-1}(\cdot)$ defined in (8) significantly increases the complexity for solving (18). Considering $l^{-1}(\cdot)$ that is used to demodulate the symbols into numerical FL parameters, the minimization of the gap between $l^{-1}\left(\frac{\sum_{k=1}^{K} l(\Delta\widetilde{\boldsymbol{w}}_{k,t})}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right)$ and $l^{-1}\left(\frac{\boldsymbol{B}_t\left(\sum_{k=1}^{K}\boldsymbol{H}_k\boldsymbol{A}_{k,t}l(\Delta\widetilde{\boldsymbol{w}}_{k,t})+\boldsymbol{n}_t\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right)$ is equivalent to minimize the distance between $\frac{\sum_{k=1}^{K} l(\Delta\widetilde{\boldsymbol{w}}_{k,t})}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}$ and $\frac{\boldsymbol{B}_t\left(\sum_{k=1}^{K}\boldsymbol{H}_k\boldsymbol{A}_{k,t}l(\Delta\widetilde{\boldsymbol{w}}_{k,t})+\boldsymbol{n}_t\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}$ in the decision region of digital demodulation, as shown in Fig. 2. To this end, in this section, we first derive the position of $\frac{\sum_{k=1}^{K} l(\Delta\widetilde{\boldsymbol{w}}_{k,t})}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}$ in the decision region and remove $l^{-1}(\cdot)$ from (18) for simplification. Then, we present a closed-form optimal design of the transmit and receive beamforming matrices.

Given $\Delta\widetilde{\boldsymbol{w}}_{k,t}$ and the digital pre-processing function $l(\cdot)$ defined in (4), the modulated symbol vector $\Delta\hat{\boldsymbol{w}}_{k,t} = l\left(\Delta\widetilde{\boldsymbol{w}}_{k,t}\right) = [\Delta\hat{w}^{\text{I}}_{k,t,1}\Delta\hat{w}^{\text{Q}}_{k,t,1},\ldots,\Delta\hat{w}^{\text{I}}_{k,t,L}\Delta\hat{w}^{\text{Q}}_{k,t,L}]$ can be obtained where $\Delta\hat{w}^{\text{I}}_{k,t,i}$ and $\Delta\hat{w}^{\text{Q}}_{k,t,i}$ are the i-th in-phase and quadrature symbols modulated by $\Delta\widetilde{\boldsymbol{w}}_{k,t}$, respectively. Since in-phase and quadrature-phase symbols that have vertical and horizontal decision regions are mutually independent, the

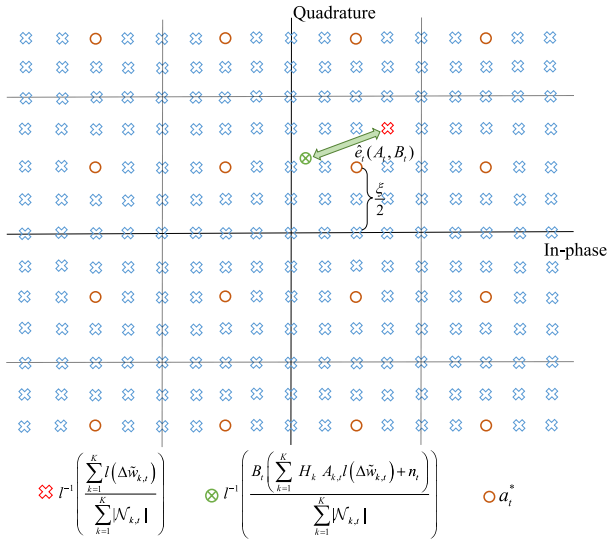


Fig. 2. An example of 16-QAM constellation at the PS with 4 devices.

value of $l^{-1}\left(\frac{\sum_{k=1}^{K}\Delta\hat{\boldsymbol{w}}_{k,t}}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right)$ can be obtained via individually analyzing the decision region of each in-phase and quadrature-phase symbols which are

$$\left|\frac{1}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\sum_{k=1}^{K}\Delta\hat{w}_{k,t,i}^{\mathrm{I}}-a_i^{\mathrm{I}}\right|\leqslant\frac{\xi}{2},$$
$$\left|\frac{1}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\sum_{k=1}^{K}\Delta\hat{w}_{k,t,i}^{\mathrm{Q}}-a_i^{\mathrm{Q}}\right|\leqslant\frac{\xi}{2},\tag{19}$$

where $a_{t,i}^{\mathrm{I}},a_{t,i}^{\mathrm{Q}}\in\mathcal{M}=\left\{\frac{1-\sqrt{M}}{2}\xi,\frac{3-\sqrt{M}}{2}\xi,\ldots,\frac{\sqrt{M}-1}{2}\xi\right\}$ are the constellation points in the decision region with $\mathcal{M}$ being the set of all constellation points. $\xi=\sqrt{\frac{4P_0}{\left(\sqrt{M}-1\right)^2}}$ is the minimum Euclidean distance between two constellation points. Using (19), $a_{t,i}^{\mathrm{I}}$ and $a_{t,i}^{\mathrm{Q}}$ are given by

$$a_{t,i}^{\mathrm{I}}=\left\{x\middle|-\frac{\xi}{2}+\frac{\sum_{k=1}^{K}\Delta\hat{w}_{k,t,i}^{\mathrm{I}}}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\leqslant x\leqslant\frac{\xi}{2}+\frac{\sum_{k=1}^{K}\Delta\hat{w}_{k,t,i}^{\mathrm{I}}}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\cap\mathcal{M}\right\}\tag{20}$$

and

$$a_{t,i}^{\mathrm{Q}}=\left\{x\middle|-\frac{\xi}{2}+\frac{\sum_{k=1}^{K}\Delta\hat{w}_{k,t,i}^{\mathrm{Q}}}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\leqslant x\leqslant\frac{\xi}{2}+\frac{\sum_{k=1}^{K}\Delta\hat{w}_{k,t,i}^{\mathrm{Q}}}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\cap\mathcal{M}\right\}.\tag{21}$$

Given $\boldsymbol{a}_t^*=[a_{t,1}^{\mathrm{I}}a_{t,1}^{\mathrm{Q}},\ldots,a_{t,W}^{\mathrm{I}}a_{t,W}^{\mathrm{Q}}]$, problem (18) can be rewritten as

$$\min_{\boldsymbol{B}_t,\boldsymbol{A}_t}\left\|\boldsymbol{a}_t^*-\boldsymbol{B}_t{}^{\mathrm{H}}\sum_{k=1}^{K}\boldsymbol{H}_k\boldsymbol{A}_{k,t}\Delta\hat{\boldsymbol{w}}_{k,t}-\boldsymbol{B}_t{}^{\mathrm{H}}\boldsymbol{n}_t\right\|^2\tag{22}$$
$$\text{s.t. }\|\boldsymbol{A}_{k,t}\|^2\leq P_0,\forall k\in\mathcal{K},\forall t\in\mathcal{T}.\tag{22a}$$

where $\Delta\hat{\boldsymbol{w}}_{k,t}=l\left(\Delta\widetilde{\boldsymbol{w}}_{k,t}\right)$ is a modulated symbol vector of $\Delta\widetilde{\boldsymbol{w}}_{k,t}$. Problem (22) can be solved by an iterative optimization algorithm. In particular, to solve problem (22), we first fix $\boldsymbol{B}_t$, then the objective functions and constraints with respect to $\boldsymbol{A}_t$ are convex and can be optimally solved by using a dual method [52]. Similarly, given $\boldsymbol{A}_t$, problem (22) is minimized as $\boldsymbol{B}_t^*=\left(\frac{\boldsymbol{a}_t^*}{\sum_{k=1}^{K}\boldsymbol{H}_k\boldsymbol{A}_{k,t}^*\Delta\hat{\boldsymbol{w}}_{k,t}}\right)^{\mathrm{H}}$.

D. Implementation and Complexity

Next, we discuss the implementation and complexity of the designed FL algorithm. With regards to the implementation of the proposed algorithm, the PS must a) use MLPs to predict the devices' local FL model update, and b) design the transmit and receive beamforming matrices based on the predicted updates. To train the MLPs that are used for the predictions of devices' local FL model update, the PS will use the global FL model $\boldsymbol{g}_{t-1}$ that is directly reconstructed from the received symbol vector $\hat{\boldsymbol{s}}_{t-1}$ at iteration $t-1$. Additionally, the PS needs the gradients of the MLP parameters $\boldsymbol{v}_k^{\mathrm{in}},\boldsymbol{v}_k^{\mathrm{out}}$ calculated on the device. To design the optimal transmit and receive beamforming matrices, the PS requires the maximal transmit power P_0 and the MIMO channel vector $\boldsymbol{H}_k$ of each device k. We ignore the overhead of each device transmitting P_0 to the PS since it is a scalar. With regards to $\boldsymbol{H}_k$, the PS can use channel estimation methods to learn $\boldsymbol{H}_k$ over each uplink channel so as to design optimal transmit and receive beamforming matrices.

To satisfy the synchronization requirement in the proposed AirComp framework, it is necessary to incorporate clock synchronization for precise control of information transmission timing and channel state information (CSI) feedback, for accurate estimation of transmission delay. For clock synchronization, the PS can broadcast a shared block to all devices to achieve time calibration, as in previous studies [53], [54], [55]. For accurate estimation of transmission delay, the PS can obtain the CSI directly from the uplink pilot signals transmitted from devices, as done in previous studies [56], [57], [58].

We identify two components of our algorithm that could potentially impact complexity and latency: (1) the MLP and (2) optimizing $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$. The training complexity of the MLP can be made small compared to the complexity of FL training. Specifically, the computational complexity of the MLP lies in the size of input $\boldsymbol{g}_{t-1}'$ and output $\Delta\widetilde{\boldsymbol{w}}_{k,t}'$, as well as the number of the neurons in the hidden layer. As discussed in Sec. III-B, the sizes of $\boldsymbol{g}_{t-1}'$ and $\Delta\widetilde{\boldsymbol{w}}_{k,t}'$ are both V', while the number of neurons in the hidden layer is D. Hence, the computational complexity is $\mathcal{O}(2V'D)$, which implies that the

Algorithm 1 Proposed FL Over AirComp-Based System

Init: Global FL model $\boldsymbol{g}_0$, beamforming matrices $\boldsymbol{A}_0$ and $\boldsymbol{B}_0$, MIMO channel matrix $\boldsymbol{H}$.
for iterations $t = 0, 1, \cdots, T$ **do**
  for $k \in \{1, 2, \cdots, K\}$ in parallel over K devices **do**
    Each device calculates and returns $\boldsymbol{w}_{k,t}$ based on local dataset and $\boldsymbol{g}_t$ in (2).
    Each device leverages digital pre-processing to modulate each model parameter into a symbol.
    Each device sends the symbol vector $\hat{\boldsymbol{w}}_{k,k}$ to the PS using the optimized transmit beamforming matrix $\boldsymbol{A}_{k,t}$.
  end for
  The PS directly demodulates the global FL model $\boldsymbol{g}_{t+1}$ from the received superpositioned signal using (8).
  The PS predicts the local FL model $\hat{\boldsymbol{w}}_{k,t+1}$ of each device based on demodulated $\boldsymbol{g}_{t+1}$ using trained ANNs.
  The PS proactively adjusts the transmit and receive beamforming matrices using the augmented Lagrangian method and broadcast the transmit beamforming matrix $\boldsymbol{A}_{k,t+1}$ to each device k.
end for

training complexity of the MLP can be controlled directly based on limiting the size of V' and D. In Sec. IV, for the ML task, we will set MLP to predict parameters for the last layer of the neural network, so that $V' \ll V$. We can further select D such that $2V'D \ll V$, so the MLP is substantially smaller than the FL model.

Regarding the optimization of $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$, the complexity scales in the number of iterations required for the solver to converge. For finding optimal $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$, problem (22) can be solved by a traditional augmented Lagrangian method that approaches the optimal solution via alternating updating $\boldsymbol{A}_t$, $\boldsymbol{B}_t$, and the Lagrangian multiplier vector. The introduced Lagrangian multiplier vector consists of K constraints, where K is the number of devices in the considered FL framework. Hence, the PS is required to sequentially update K Lagrangian multipliers, $\boldsymbol{A}_t = [\boldsymbol{A}_{1,t}, \cdots, \boldsymbol{A}_{K,t}]$, and $\boldsymbol{B}_t$ at each iteration. Letting L_O be the number of iterations until the augmented Lagrangian method converges, the complexity is $\mathcal{O}(L_O K^2)$.

IV. Experimental Results and Discussion

A. Simulation Setup

We consider a circular network area with a radius $r = 1500$ m with one PS at its center serving $K = 20$ uniformly distributed devices. In particular, the PS allocates 64 subcarriers to all devices and the bandwidth of each subcarrier is 15 kHz. By default, the channels between the PS and devices are modeled as independent and identically distributed with Rayleigh fading and a pathloss exponent of $\beta = 2$. The other default settings of parameters used in simulations are listed in Table II.

For comparison purposes, we consider several baseline AirComp FL frameworks: analog modulation, BPSK modulation, 64 QAM without the assistance of MLP, 64 QAM replacing the MLP with a tabular approach, and an ideal upper bound of 64 QAM over noiseless channels. These considered baselines are detailed as follows:

TABLE II
Default Simulation Parameters

Parameters	Values	Parameters	Values	Parameters	Values
K	20	M	64	σ^2	-90 dBW
N_r	2	N_t	2	N_k	2000
P_0	1 mW	T	50	W	7840
N_I	28	N_O	10	λ	0.01
r	1500 m	D	160	L_0	100

- The analog AirComp FL framework, from [27], enables each device to use digital beamforming and analog modulation for transmitting FL parameters over wireless fading channels (labeled "Analog FL" in plots).
- The BPSK AirComp FL framework, from [36], enables each device to quantize its trained FL model parameters into one bit and use digital beamforming and BPSK for transmitting quantized FL parameters over wireless fading channels (labeled "BPSK FL" in plots).
- An AirComp FL framework without the assistance of MLP enables the devices to quantize their trained FL model parameters into 6 bits and directly use digital beamforming and 64 QAM for transmitting quantized FL parameters over wireless fading channels (labeled "Proposed method without MLP" in plots).
- An AirComp FL framework considering noiseless channels enables the devices to quantize their trained FL model parameters into 6 bits and use digital beamforming and 64 QAM for quantized FL parameter transmission over noiseless channels. Then, the PS uses the receive beamforming matrix optimized by MLP to adjust the decision region of the received superimposed signals to minimize the transmission error (labeled "Proposed FL over noiseless channels" in plots).
- An AirComp FL framework where, as in our method, the devices quantize their trained FL model parameters into 6 bits and directly use digital beamforming and 64 QAM for transmitting quantized FL parameters. However, ChannelComp from [41] is employed to optimize the transmission error instead of the ANN (labeled "Proposed method with ChannelComp" in plots).

The Fashion-MNIST dataset [59] and CIFAR-10 dataset [60] are used as ML tasks in our performance evaluation. For Fashion-MNIST, each local FL model consists of five convolutional layers and one fully-connected layer, with $V = 9.2 \times 10^5$ total model parameters. For CIFAR-10, each local FL model is a standard ResNet-18 that consists of 17 convolutional layers and one fully-connected layer, with $V = 1.17 \times 10^7$ total model parameters.

For the MLP described in Sec. III-B, we focus on predicting the parameters of the last layer in each ML model (i.e., the fully connected layers). This leads to $V' = 640$ for both the CNN in Fashion-MNIST and the Resnet-18 in CIFAR-10. To restrict the ANN's total size to a small fraction of the ML model in each case, we set $D = 160$ for Fashion-MNIST and $D = 320$ for CIFAR-10, in line with CIFAR-10's higher

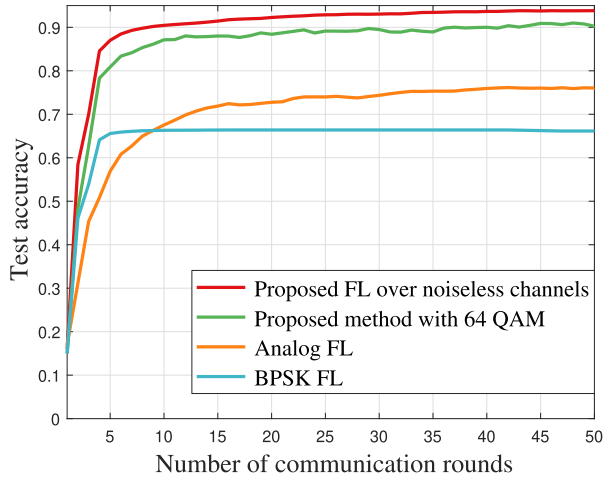


Fig. 3. Test accuracy of the proposed AirComp-based FL system over communication rounds, for an IID data allocation on the Fashion-MNIST task. We see that our proposed method provides substantial improvements over the baselines, approaching the noiseless channel case.

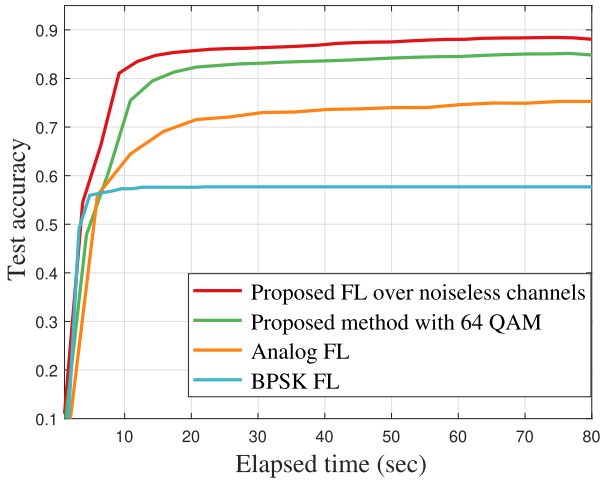


Fig. 4. Test accuracy of the proposed AirComp-based FL system over time on the Fashion-MNIST task, for an non-IID data allocation. The overall trends are similar to those observed in Fig. 3.

complexity (leading to the MLP being 22% and 3% of the model sizes, respectively). As we will see in Sec. IV-C, we find that this retains a strong MSE performance for each dataset. Similar considerations can be made for other ML tasks.

Throughout the simulations, we will consider data samples to be allocated across devices in either an independent and identically distributed (IID) or non-IID manner. In the IID data setting, each device is allocated data samples from all 10 class labels, while in the non-IID case, they each only receive data samples from a fraction of the labels, assumed to be 6 by default [61], [62].

It is worth noting that the convolution kernels exhibit sensitivity to errors, which significantly degrade the performance of the FL model, as will be demonstrated in the simulations. However, considering the relatively small number of parameters in convolution kernels, it is viable to utilize traditional methods such as orthogonal frequency division multiple access for transmitting the kernel parameters.

Finally, in some of our experiments (Fig. 4, Table III, Table IV, Fig. 9), we will conduct accuracy comparisons over time/latency incurred rather than communication rounds. These calculations employ standard models for communication and computation delays [63], [64], which we present in Appendix B. For experiments that refer to convergence (in particular, Tables III & IV), algorithms are considered to have converged when the value of the FL loss variance calculated over five consecutive iterations is less than 0.001.

B. Training Speed and Accuracy Comparisons

We first conduct several experiments comparing our methodology to the baselines in terms of convergence speed and accuracy. In Fig. 3, we show how the test accuracy of all considered algorithms changes over communication rounds, on the Fashion-MNIST task, for the IID data setting. In this figure, we can see that, the proposed algorithm improves the test accuracy by up to 15.5% and 24.5% compared to analog FL and BPSK FL, respectively. From Fig. 3, we can also

TABLE III

LATENCY VS. ACCURACY TRADEOFF FOR TRAINING ON THE FASHION-MNIST DATASET

	Avg. Lat. for Each Round	Avg. Lat. for Entire Training	Avg. Rounds to Converge	Converged Test Acc.
Proposed	1.26 s	18.90 s	15	90%
Analog FL	1.03 s	23.69 s	23	76%
BPSK FL	0.21 s	1.05 s	5	67%

see that the performance of analog FL experiences noticeable fluctuations. This is due to the noise over wireless channels introducing dynamic errors into FL parameter transmission process, thus affecting FL test accuracy. We also see that the proposed method without using MLP for predicting FL gradients cannot converge, which verifies that the optimal beamforming matrices design depends on the prediction of FL gradients.

Fig. 4 shows how the test accuracy changes over time elapsed, measured in seconds, as opposed to iterations. Here, the elapsed time consists of the FL model training time, inference time that is used for predicting FL parameters, and the model transmission time. The overall trends are similar to those observed in Fig. 3: compared to BPSK FL, while our proposed methodology converges more slowly, it ends up improving the test accuracy by up to 27%. This is because the proposed FL uses more bits instead of one bit in BPSK FL to represent each FL parameter, thus increasing the dynamics of global FL model generation. In addition, this figure also shows that the proposed algorithm can reduce the convergence time by 20.3% compared to analog FL. This finding underscores the fact that although the utilization of MLP introduces additional inference latency, it ultimately results in a decrease in the overall convergence time of FL. We will further investigate the latency vs. accuracy tradeoff in Tables III and IV.

Fig. 5 shows how the test accuracy changes as the number of communication rounds varies on the CIFAR-10 dataset, for the IID data distribution setting. Overall, we make similar observations from this figure as for Fashion-MNIST in Fig. 3. We see that our proposed method can improve the test accuracy by up to 20% compared to analog FL, due to the

TABLE IV
LATENCY VS. ACCURACY TRADEOFF FOR TRAINING ON THE CIFAR-10 DATASET

	Avg. Lat. for Each Round	Avg. Lat. for Entire Training	Avg. Rounds to Converge	Converged Test Acc.
Proposed	21.46 s	966 s	45	73%
Analog FL	20.92 s	1255 s	60	52%
BPSK FL	2.95 s	118 s	40	31%

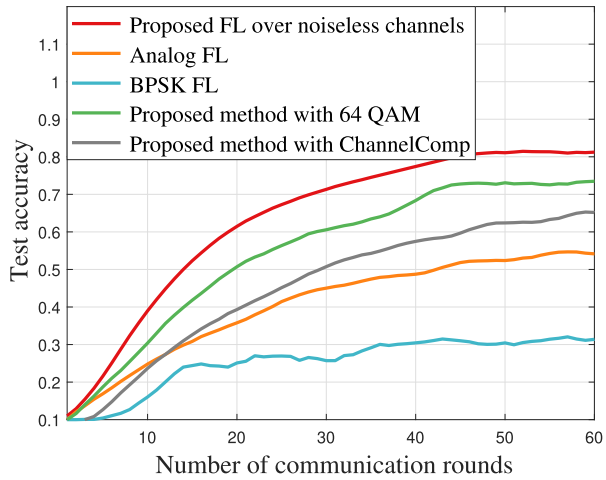


Fig. 5. Test accuracy of the proposed AirComp-based FL system across communication rounds, for an IID data allocation on the CIFAR-10 dataset. Compared to Fig. 3, we observe significant improvements over the baselines. The larger gap from the noiseless channel case shows the impact of noise on more complex learning tasks (CIFAR-10 vs. Fashion-MNIST).

proposed FL approach utilizing digital modulation (i.e., 64 QAM) to mitigate channel impairments and misalignments. We can also see that BPSK FL can only achieve 30% test accuracy. The lower overall accuracies of each algorithm are consistent with CIFAR-10 being a more difficult learning task than Fashion-MNIST. Along these lines, the impact of fading and additive white Gaussian noise is stronger for this case compared with the Fashion-MNIST experiments, which can be explained by CIFAR-10 having an ML model that is more complex, causing the results to be more sensitive to impairments. Fig. 5 also shows that our methodology with the ANN model prediction strategy leads to an improvement of up to 10% compared to employing ChannelComp [41]. This confirms that our prediction-based strategy obtains smaller errors compared to the transmission error minimization employed by this competing approach.

In Fig. 6, we repeat the experiment from Fig. 5, but this time under a non-IID setting, where the data samples of each device are drawn from 6 (instead of 10) classes. From this figure, we see that the proposed FL with 64-QAM can improve the test accuracy by up to 18% and 13% compared with analog FL and ChannelComp, similar to in the IID case. Fig. 6 also shows that under noiseless channels, the accuracy of the AirComp-based system is 6% better, consistent with the IID case.

Finally, in Tables III and IV, we compare the performance of the proposed method with that of analog FL and BPSK FL in terms of (i) latency per iteration, (ii) overall training time, (iii) average rounds to convergence, and (iv) converged

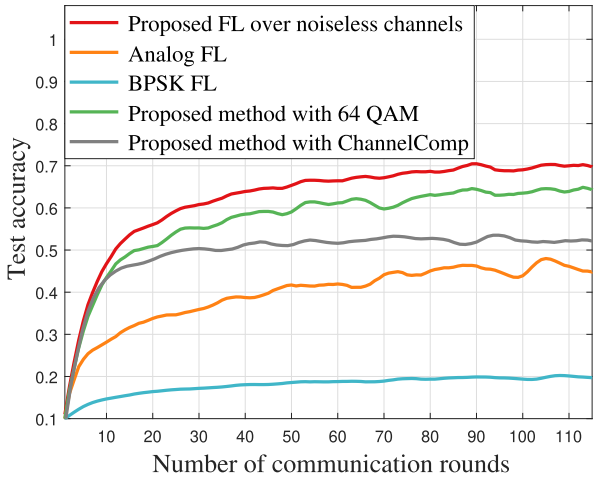


Fig. 6. Test accuracy of the proposed AirComp-based FL system across communication rounds, for a non-IID data allocation on the CIFAR-10 task. The overall trends are consistent with the IID case in Fig. 5, showing that our methodology obtains improvements under different data partitions.

test accuracy, on the Fashion-MNIST and CIFAR-10 datasets, respectively, for the non-IID cases. Compared to analog FL we see that while our method incurs a higher latency in each round (e.g., 1.26 vs. 1.03 sec in Fashion-MNIST), it has a smaller latency across the entire training process (e.g., 18.90 vs. 23.69 sec in Fashion-MNIST), since it requires less total rounds to converge. Employing the MLP within our method introduces additional latency and computational overhead, but aids in designing the beamforming matrices to reduce transmission errors, thereby reducing the total number of rounds required. BPSK FL has the lowest latencies of all, but suffers from poor testing accuracy at convergence, due to the fact that it quantizes the FL parameters to a single bit. We observe too that the per-round and total latencies are higher in CIFAR-10, which is the most complex of the tasks, with the largest ML model being trained and transmitted.

C. Robustness of MLP Predictor

Given the importance of the ANN-based predictor at the PS to our approach, we conduct further analysis to assess how it is impacted by the learning environment. First, in Fig. 7, we show the performance of the MLP predicting FL parameters on the CIFAR-10 dataset as the data partitioning varies across devices. In the non-IID cases, the data samples of each device are from a fraction of the labels: 3 of 10, 6 of 10, and 9 of 10, respectively. The MSE is used to evaluate the performance of the MLP on the test dataset at each iteration of training. In this figure, we see that the proposed ANN algorithm achieves a predominantly linear convergence speed prior to tapering off. This is because the proposed ANN has only three fully-connected layers thus having a relatively low computational complexity. Fig. 7 also shows that as the proposed ANN method converges, the MSE approaches 0, which implies the proposed ANN approach can predict FL parameters accurately. We also see that the values of the MLP MSE in both the IID and non-IID data distributions are similar. This indicates that MLP is capable of capturing the nonlinear

TABLE V

MSE PERFORMANCE OF THE PROPOSED ANN-BASED MODEL PARAMETER PREDICTOR WHEN THE DEVICE MODELS ARE TRANSMITTED OVER DIFFERENT CHANNEL CONDITIONS, FOR CIFAR-10 WITH A NON-IID DATA PARTITION

	MLP with perfect communication	Rayleigh, $\sigma^2 = -90$ dBm	Rayleigh, $\sigma^2 = -95$ dBm	Rician, $\sigma^2 = -90$ dBm, $k = 7$	Rician, $\sigma^2 = -90$ dBm, $k = 10$
Converged MSE ($\times 10^{-7}$)	7.73	13.21	11.14	8.51	7.80

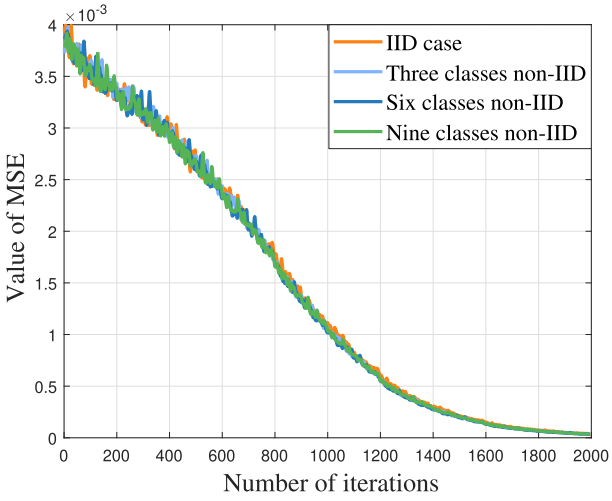


Fig. 7. MSE performance of the proposed ANN-based model parameter predictor across training iterations on the CIFAR-10 dataset. We see that the ANN prediction quality is robust to the data distribution across devices.

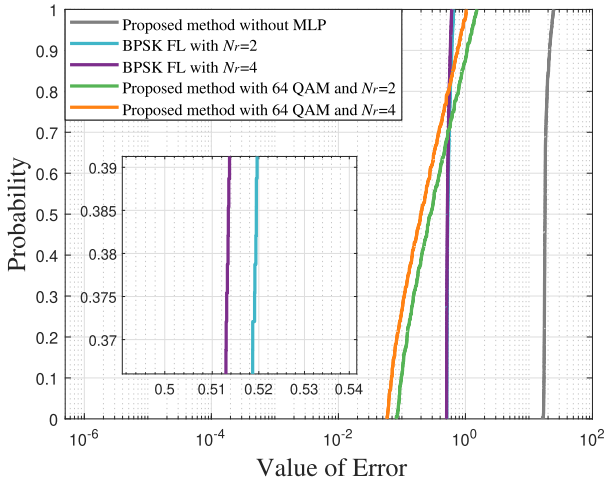


Fig. 8. Cumulative distribution function (CDF) of the errors in the aggregated models, for training on Fashion-MNIST dataset under the IID data partition.

relationships of the FL parameters for different distributions of data samples across the devices.

Next, in Table V, we show the MSE of the MLP when the FL models of devices are transmitted over different channel conditions, on the CIFAR-10 dataset. Here, we consider a non-IID data distribution where the data samples of each device are from 6 of 10 labels. We consider the implementation of the proposed framework over four channel conditions: (a) a Rayleigh model with noise variance $\sigma^2 = -90$ dBm, (b) a Rayleigh model with noise variance $\sigma^2 = -95$ dBm, (c) a Rician model with noise variance $\sigma^2 = -90$ dBm and Rician factor $k = 7$, and (d) a Rician model with noise variance $\sigma^2 = -90$ dBm and Rician factor $k = 10$. The pathloss exponent is set to $\beta = 2$ for Rayleigh and $\beta = 4$ for Rician. The MSE performance of the MLP without transmission errors is considered as an optimal baseline. Overall, we can see that the MLP demonstrates satisfactory performance in terms of MSE regardless of the channel model, which indicates the robustness of the proposed ANN algorithm. We also notice that the MSE of the MLP transmitted through the Rician fading channel is lower than the MSE of the MLP transmitted through the Rayleigh fading channel due to the influence of the line-of-sight signal. Additionally, as the Rician factor increases and the noise variance decreases, the MSE of the MLP transmitted over different channels approaches the optimal baseline.

Finally, Fig. 8 assesses the error in the obtained aggregated model $\boldsymbol{g}_t$ at the PS for different methods. The error is defined as the sum of the distances between all weights in the aggregated model at the PS and that in the true average of the models across the devices. The results are shown as the cumulative distribution function (CDF) of the value of the errors obtained over training rounds. We can see that the proposed method achieves a lower error rate compared to the proposed FL without using MLP for FL gradient prediction. This is because without predicting FL gradient vectors, the PS cannot proactively adjust the transmit and receive beamforming matrices to minimize transmission errors and can only use fixed beamforming design that directly aggregates all local models via linear superimposition. This linear superimposition is not available for digital modulation schemes since digital modulation may introduce complex mapping relationships between bits and symbols thus resulting in additional demodulation errors.

D. Impacts of Channel Conditions and SNR

We next investigate the impact of channel conditions on the performance of our methodology. In Fig. 9, we show the performance of the proposed method on the Fashion-MNIST dataset under various channel models. In particular, we consider the Rayleigh fading channel and the Rician fading channel. For each model, we consider three different settings, varying the noise variance $\sigma^2 = -90$ dBm, pathloss exponent β, and Rician factor k. Overall, from this figure, we can see that the proposed algorithm is reasonably robust to the specific channel conditions, with the testing accuracy staying within a window of ± 0.04. The best performance occurs when the channel is Rician and $k = 10$. This is consistent

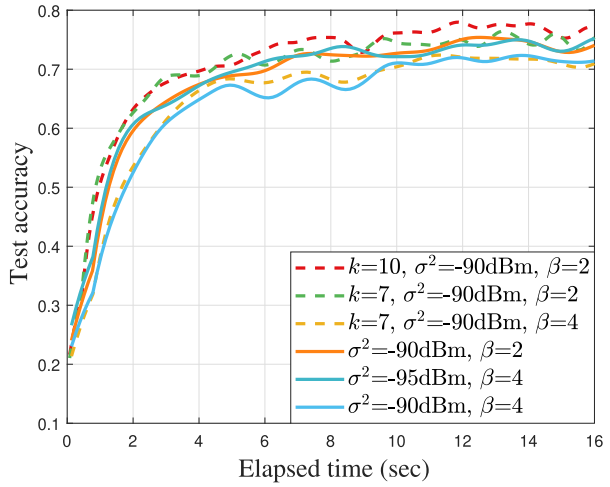


Fig. 9. Impact of Rician (dashed lines) and Rayleigh (solid lines) channel models on training performance, for the Fashion-MNIST dataset under an IID data partition. Overall, we see that our methodology is reasonably robust to variations in the channel model.

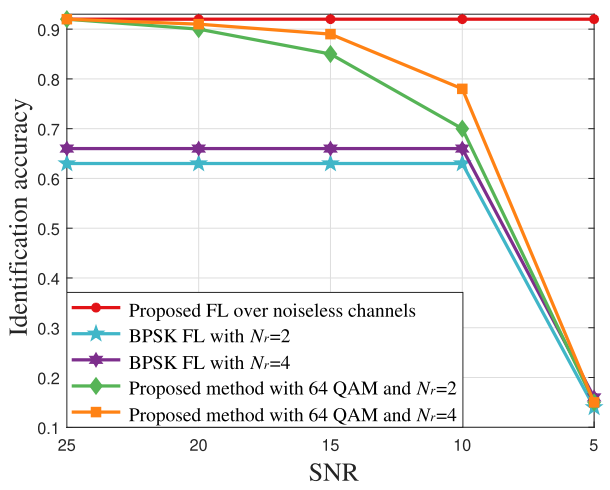


Fig. 10. Impact of SNR on test accuracy for the Fashion-MNIST dataset, for an IID data partition. We see that the performance of all methods begins to decay once the SNR drops below 10 dB.

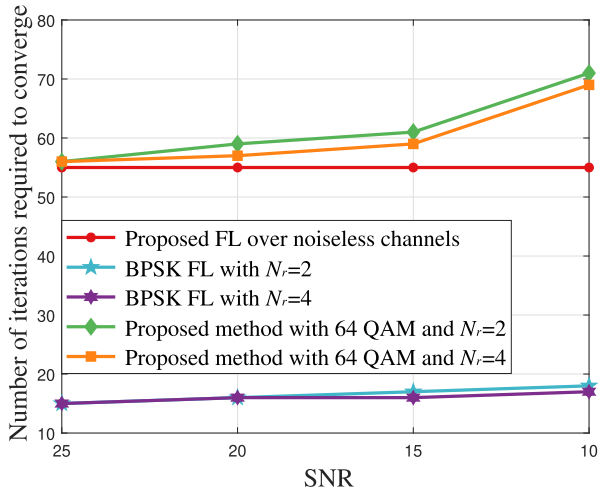


Fig. 11. Impact of SNR on convergence performance for the Fashion-MNIST dataset, for the same settings as in Fig. 10. We see that our method converges significantly faster at higher SNRs, with less transmission errors occurring.

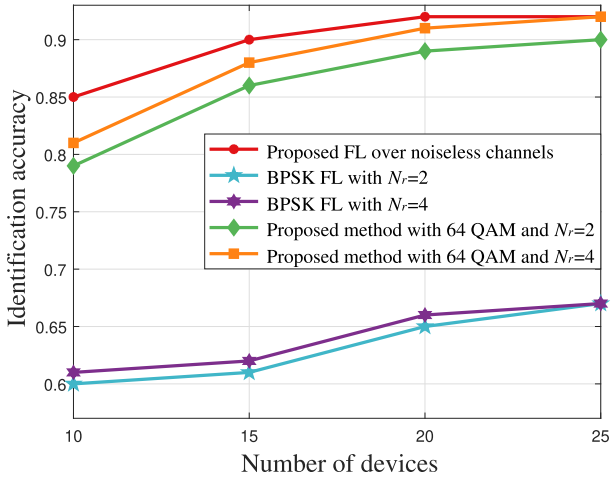


Fig. 12. Impact of network size on obtained accuracy for the Fashion-MNIST dataset under an IID data partition. A larger network has a positive impact on accuracy performance.

with the Rician factor being the ratio of the power of the dominant path component to the power of the whole signal. Thus, $k = 10$ is the case of the largest dominant path, which reduces FL transmission errors and improves the FL performance.

We next consider the impact of the received SNR. Fig. 10 shows how the test accuracy of considered FL algorithms changes as SNR decreases, for the Fashion-MNIST dataset. In Fig. 10, we can see that the test accuracy of the proposed method decreases as SNR decreases, while the test accuracy of BPSK FL remains unchanged, when the SNR is larger than 10 dB. Additionally, the test accuracy of BPSK FL is lower than the proposed method at any SNR values. This is because the quantization error in BPSK significantly affects the model training process and results in a degeneration of test accuracy. Fig. 10 also shows that the proposed method with $N_r = 4$ receiver antennas can achieve 8% gains in terms of test accuracy compared to that with $N_r = 2$ receiver antennas when SNR is 10 dB. Thus, an increase of the number of receiver antennas can improve the test accuracy in the proposed FL framework. This is because an increase of the number of receiver antennas enables the PS to exploit transmit diversity and reduce transmission error in the AirComp-based system.

Finally, in Fig. 11, we show how the number of communication rounds that the considered FL algorithms require to converge changes as SNR decreases, for the same settings as in Fig. 10. We can see that, compared with BPSK FL, the number of communication rounds required to converge for our methodology decreases noticeably as the SNR increases. This is due to the fact that, as SNR decreases, the probability of introducing additional transmission errors increases thus reducing the FL convergence speed.

E. Impact of Network Size

Finally, we consider the impact of the number of devices on the performance. Fig. 12 shows how the test accuracy changes as the number of devices varies. In Fig. 12, we can see that

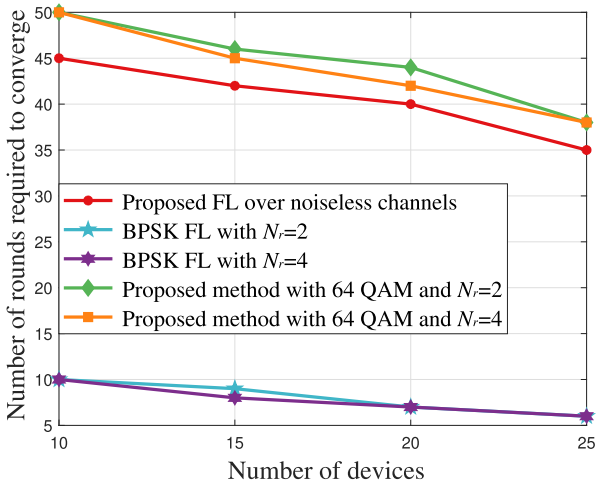


Fig. 13. Impact of network size on convergence performance, for the same settings as Fig. 12. A larger network reduces the iterations required.

the test accuracy increases as the number of devices increases. This is because as the number of devices increases, the number of data samples available in the network for training increases. This helps improve the performance for both of our higher-order modulation method as well as for BPSK FL. Fig. 12 also shows that as the number of receiver antennas increases from $N_r = 2$ to $N_r = 4$, the test accuracy of the proposed methodology experiences a slight improvement, because of the PS's ability to exploit transmit diversity.

In Fig. 13, we show how the number of rounds required to converge varies as the number of devices changes, for the same dataset and settings as Fig. 12. Overall, we see that as the number of devices increases, the number of rounds needed to converge decreases, in addition to improving the overall testing accuracy. Fig. 13 also shows that our method can exploit a larger number of antennas to reduce the iterations required for convergence, with appropriate design of the beamforming matrices.

V. Conclusion

In this article, we have developed a novel framework that enables the implementation of FL algorithms over a digital MIMO and AirComp based system. We have formulated an optimization problem that jointly considers transmit and receive beamforming matrices for the minimization of FL training loss. To solve this problem, we analyzed the expected improvement of FL training loss between two adjacent iterations that depends on the digital modulation mode, the number of devices, and the design of beamforming matrices. To obtain this bound in practice, we introduced an ANN based algorithm to estimate the local FL models of all devices; then, the optimal solution of beamforming matrices can be determined based on the predicted FL model and the derived expected improvement of FL training loss. Numerical evaluation on real-world machine learning tasks demonstrated that the proposed methodology yields significant gains in classification accuracy and convergence speed compared to conventional approaches.

Appendix A
Proof of Theorem 1

To prove Theorem 1, we first rewrite $F(\boldsymbol{g}_{t+1})$ using the second-order Taylor expansion and the property of the L-smooth in Assumption a), which can be expressed as

$$F\left(\boldsymbol{g}_{t+1}\right) \leq F\left(\boldsymbol{g}_t\right) + \left(\boldsymbol{g}_{t+1} - \boldsymbol{g}_t\right) \nabla F\left(\boldsymbol{g}_t\right) + \frac{L}{2}\left\|\boldsymbol{g}_{t+1} - \boldsymbol{g}_t\right\|^2.$$

Let $\boldsymbol{g}_{t+1} - \boldsymbol{g}_t = \nabla F\left(\boldsymbol{g}_t\right) - \boldsymbol{o}_t$ and the learning rate $\lambda = \frac{1}{L}$, we have

$$\begin{aligned}
&\mathbb{E}\left(F\left(\boldsymbol{g}_{t+1}\right)\right) - \mathbb{E}\left(F\left(\boldsymbol{g}_t\right)\right) \\
&\leq -\lambda \mathbb{E}\left(\nabla F\left(\boldsymbol{g}_t\right) - \boldsymbol{o}_t\right) \nabla F\left(\boldsymbol{g}_t\right) + \frac{L\lambda^2}{2} \mathbb{E}\left\|\nabla F\left(\boldsymbol{g}_t\right) - \boldsymbol{o}_t\right\|^2 \\
&\overset{(a)}{=} -\frac{1}{2L} \mathbb{E}\left\|\nabla F\left(\boldsymbol{g}_t\right)\right\|^2 + \frac{1}{2L} \mathbb{E}\left(\left\|\boldsymbol{o}_t\right\|^2\right),
\end{aligned}$$

where (a) stems from the fact that $\frac{L\lambda^2}{2}\left\|\nabla F\left(\boldsymbol{g}_t\right) - \boldsymbol{o}_t\right\|^2 = \frac{1}{2L}\left\|\nabla F\left(\boldsymbol{g}_t\right)\right\|^2 - \frac{1}{L}\boldsymbol{o}^T \nabla F\left(\boldsymbol{g}_t\right) + \frac{1}{2L}\mathbb{E}\left(\left\|\boldsymbol{o}_t\right\|^2\right)$ with $\boldsymbol{o}_t$ being a gradient deviation caused by the errors in local FL model transmission, which can be given as follows

$$\begin{aligned}
&\mathbb{E}\left(\left\|\boldsymbol{o}_t\right\|^2\right) \\
&= \mathbb{E}\left(\left\|\nabla F\left(\boldsymbol{g}_t\right) - \left(\boldsymbol{g}_{t+1} - \boldsymbol{g}_t\right)\right\|^2\right) \\
&= \mathbb{E}\left(\left\| \frac{\sum_{k=1}^{K}\sum_{n=1}^{N_k} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)}{N} \right.\right. \\
&\quad \left.\left. -l^{-1}\left(\hat{\boldsymbol{s}}_t\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)\right)\right\|^2\right) \\
&\leqslant \mathbb{E}\left(\left\| \frac{\sum_{k=1}^{K}\sum_{n=1}^{N_k} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)}{N} \right.\right. \\
&\quad \left. - \frac{\sum_{k=1}^{K}\sum_{n=1}^{N_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)}{\sum_{k=1}^{K}\left|\mathcal{N}_{k,t}\right|} \right\| \\
&\quad + \left\| \frac{\sum_{k=1}^{K}\sum_{n=1}^{N_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)}{N} \right. \\
&\quad \left.\left. -l^{-1}\left(\frac{\sum_{k=1}^{K} l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)}{\sum_{k=1}^{K}\left|\mathcal{N}_{k,t}\right|}\right)\right\|\right)
\end{aligned}$$

$$+\left\| l^{-1}\left(\frac{\sum_{k=1}^{K} l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right)\right.$$
$$\left. -l^{-1}\left(\hat{\boldsymbol{s}}_t\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)\right)\right\|^2 \quad (23)$$

where $\nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)$ is the gradient trained by $\left(\boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)$. $\hat{\boldsymbol{s}}_t\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right) = \frac{\boldsymbol{B}\sum_{k=1}^{K}\boldsymbol{H}_k\boldsymbol{A}_k l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)+\boldsymbol{B}\boldsymbol{n}_t}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}$ is the received signal. $\frac{\sum_{k=1}^{K} l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}$ is the theoretical signal that is obtained via modulation at devices and demodulation at the PS without channel impairments and misalignments. Given $\nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)$, the first term in (23) represents the discrepancy of gradients introduced by random sampling of mini-batches, which can be rewritten as

$$\begin{aligned}
&\mathbb{E}\left\| \frac{\sum_{k=1}^{K}\sum_{n=1}^{N_k}\nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)}{N} \right.\\
&\qquad \left. - \frac{\sum_{k=1}^{K}\sum_{n=1}^{N_{k,t}}\nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|} \right\|\\
&= \mathbb{E}\left\| \sum_{k=1}^{K}\widehat{\boldsymbol{w}}_{k,t} - \sum_{k=1}^{K}\boldsymbol{w}_{k,t} \right\|\\
&= \sum_{k=1}^{K}\frac{\sigma_k^2}{|\mathcal{N}_k|},
\end{aligned}$$

where $\widehat{\boldsymbol{w}}_{k,t}$ is the optimal updated local model without variance. Given $\frac{\sigma_k^2}{|\mathcal{N}_k|}$ on each device k, the variance bound of all devices is $\Upsilon = \sum_{k=1}^{K}\frac{\sigma_k^2}{|\mathcal{N}_k|}$.

And then, we have $\mathbb{E}\left(\|\boldsymbol{o}_t\|^2\right) = \mathbb{E}\left(\|\boldsymbol{e}_t + \hat{\boldsymbol{e}}_t(\boldsymbol{A}_t, \boldsymbol{B}_t)\|^2\right)$ where

$$\boldsymbol{e}_t = \left\| \frac{\sum_{k=1}^{K}\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|} \right.$$
$$\left. - l^{-1}\left(\frac{\sum_{k=1}^{K} l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right)\right\| + \Upsilon \quad (24)$$

and

$$\begin{aligned}
&\hat{\boldsymbol{e}}_t(\boldsymbol{A}_t, \boldsymbol{B}_t)\\
&= l^{-1}\left(\frac{\sum_{k=1}^{K} l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right)\\
&\quad - l^{-1}\left(\frac{\boldsymbol{B}_t\left(\sum_{k=1}^{K}\boldsymbol{H}_k\boldsymbol{A}_{k,t} l\left(\sum_{n\in\mathcal{N}_{k,t}} \nabla f\left(\boldsymbol{g}_t, \boldsymbol{x}_{n,k}, \boldsymbol{y}_{n,k}\right)\right)+\boldsymbol{n}_t\right)}{\sum_{k=1}^{K}|\mathcal{N}_{k,t}|}\right).
\end{aligned} \quad (25)$$

This completes the proof.

APPENDIX B
LATENCY MODELS

We present here the models employed for calculating latency/delay in the experiments. For our algorithm, the latency encompasses the time of updating and transmitting both the MLP and the local FL models. For the analog FL and BPSK FL baselines, the latency consists of the time of updating and uploading the FL parameters only. Our models of the four components are given below.

A. MLP Updating Time

The updating time of the MLP depends on the computational complexity of the MLP and the computational power of each device. The computational complexity of the MLP depends on the size of input $\boldsymbol{g}'_{t-1}$ and output $\Delta\widetilde{\boldsymbol{w}}'_{k,t}$, as well as the number of the neurons in the hidden layer. As discussed in Section III-B, the sizes of $\boldsymbol{g}'_{t-1}$ and $\Delta\widetilde{\boldsymbol{w}}'_{k,t}$ are both V', and the number of neurons in the hidden layer is D. Then, following the latency models in [63], the updating time of the MLP can be modeled as

$$L_{\mathrm{MLP,U}} = \frac{2\alpha^2 V' D}{\varepsilon f_{\mathrm{D}}}\rho, \quad (26)$$

where $2V'D$ is the total number of parameters, f_{D} is the CPU clock frequency of a device (in cycles/sec), ε is the number of parameter update operation executed by one CPU cycle, ρ is the time-consumption coefficient depending on the specific chip of each device, and α is the quantization precision.

B. FL Updating Time

The updating time of the FL model similarly depends on the computational complexity of the adopted model and the computational power of each device. For Fashion-MNIST and CIFAR-10, we employ a series of convolutional layers and one linear layer. The training complexity depends on the specific structure of each layer. In each convolutional layer i, let M_i denote the size of the convolutional kernel, K_i denotes the side length of the output feature map generated by each convolutional kernel, and C_i^{in} and C_i^{out} denote the number of input channels and output channels, respectively. Then, the

number of operations required for training one of the models is $O_{\rm FL} = \sum_{i=1}^{I} M_i^2 \times K_i^2 \times C_i^{\rm in} \times C_i^{\rm out} + C_{I+1}^{\rm out} \times L_{\rm out}$, where $L_{\rm out}$ is the output dimension of the entire network and I is the number of convolutional layers [65]. Then, the updating time of the FL model is given by

$$L_{\rm FL,U} = \frac{O_{\rm FL}\alpha^2}{\varepsilon f_{\rm D}}\rho. \tag{27}$$

C. MLP Transmission Time

The transmission time of the MLP depends on the size of the MLP and the channel states. Given the number of parameters is $2V'D$, the transmission time of the MLP can be modeled as

$$L_{\rm MLP,T} = \frac{l\left(2\alpha V'D\right)}{C}, \tag{28}$$

where $l(\cdot)$ denotes the digital pre-processing function described in Sec. II-A. C is the channel capacity (in bits/sec), which is given for MIMO channels as [64]

$$C = \mathbb{E}\left[\log\det\left(\mathbf{I}_{N_r} + \frac{\rm SNR}{N_t}\right)\mathbf{H}\mathbf{H}^*\right], \tag{29}$$

where ${\rm SNR} = P_0/\sigma^2$ is the common signal-to-noise ratio (SNR) at each receive antenna with σ^2 being the variance of the additive white Gaussian noise, N_t and N_r are the numbers of transmitting and receiving antennas, respectively, and $\mathbf{H}$ is the MIMO channel vector.

D. FL Transmission Time

The transmission time of the FL model depends on the size of the adopted FL model and the channel state. Given the neural network structures, the transmission delay can be modeled as

$$L_{\rm FL,T} = \frac{l\left(V\alpha\right)}{C}, \tag{30}$$

where C is as defined above and $V = \sum_{i=1}^{I} M_i^2 \times C_i^{\rm in} \times C_i^{\rm out} + C_{I+1}^{\rm out} \times L_{\rm out}$ is the number of parameters in the adopted FL model.

For the CIFAR-10 task, $I = 17$, $V = 1.17\times10^7$, and $O_{\rm FL} = 1.7 \times 10^{12}$, and for the Fashion MNIST task, $I = 5$, $V = 9.2\times10^5$, and $O_{\rm FL} = 1.2\times10^9$. The value of α is variable for different baselines: for BPSK, $\alpha = 1$ (i.e., single bit precision), for analog FL, $\alpha = 32$ (i.e., full precision computation), and for our proposed method using 64 QAM, $\alpha = 6$. We set $\varepsilon = 10^5$, $\rho = 1$, and $f_{\rm D} = 1$ GHz in our experiments.

REFERENCES

[1] D. Wen, K.-J. Jeon, and K. Huang, "Federated dropout—A simple approach for enabling federated learning on resource constrained devices," *IEEE Wireless Commun. Lett.*, vol. 11, no. 5, pp. 923–927, May 2022.

[2] M. Chen et al., "Distributed learning in wireless networks: Recent progress and future challenges," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3579–3605, Dec. 2021.

[3] S. Hosseinalipour, C. G. Brinton, V. Aggarwal, H. Dai, and M. Chiang, "From federated to fog learning: Distributed machine learning over heterogeneous wireless networks," *IEEE Commun. Mag.*, vol. 58, no. 12, pp. 41–47, Dec. 2020.

[4] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10700–10714, Dec. 2019.

[5] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1622–1658, 3rd Quart., 2021.

[6] Y. Shen, Y. Qu, C. Dong, F. Zhou, and Q. Wu, "Joint training and resource allocation optimization for federated learning in UAV swarm," *IEEE Internet Things J.*, vol. 10, no. 3, pp. 2272–2284, Feb. 2023.

[7] M. Chen, N. Shlezinger, H. V. Poor, Y. C. Eldar, and S. Cui, "Communication efficient federated learning," *Proc. Nat. Acad. Sci. USA*, vol. 118, no. 17, Apr. 2021, Art. no. e2024789118.

[8] X. Wei, C. Shen, J. Yang, and H. V. Poor, "Random orthogonalization for federated learning in massive MIMO systems," 2022, *arXiv:2201.12490*.

[9] W. Guo, R. Li, C. Huang, X. Qin, K. Shen, and W. Zhang, "Joint device selection and power control for wireless federated learning," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 8, pp. 2395–2410, Aug. 2022.

[10] H. Sun, X. Ma, and R. Q. Hu, "Adaptive federated learning with gradient compression in uplink NOMA," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 16325–16329, Dec. 2020.

[11] S. Wang, M. Chen, C. G. Brinton, C. Yin, W. Saad, and S. Cui, "Performance optimization for variable bitwidth federated learning in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 23, no. 3, pp. 2340–2356, Mar. 2024.

[12] T. Liu, B. Di, S. Wang, and L. Song, "A privacy-preserving incentive mechanism for federated cloud-edge learning," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Madrid, Spain, Dec. 2021, pp. 1–6.

[13] 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.

[14] N. Zhang and M. Tao, "Gradient statistics aware power control for over-the-air federated learning," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5115–5128, Aug. 2021.

[15] S. Jing and C. Xiao, "Federated learning via over-the-air computation with statistical channel state information," *IEEE Trans. Wireless Commun.*, vol. 21, no. 11, pp. 9351–9365, Nov. 2022.

[16] W. Ni, Y. Liu, Y. C. Eldar, Z. Yang, and H. Tian, "STAR-RIS integrated nonorthogonal multiple access and over-the-air federated learning: Framework, analysis, and optimization," *IEEE Internet Things J.*, vol. 9, no. 18, pp. 17136–17156, Sep. 2022.

[17] C. Zhong, H. Yang, and X. Yuan, "Over-the-air federated multi-task learning over MIMO multiple access channels," *IEEE Trans. Wireless Commun.*, vol. 22, no. 6, pp. 3853–3868, Jun. 2023.

[18] Y. Shao, D. Gunduz, and S. C. Liew, "Federated edge learning with misaligned over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 3951–3964, Jun. 2022.

[19] Z. Wang, Y. Zhou, Y. Shi, and W. Zhuang, "Interference management for over-the-air federated learning in multi-cell wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 8, pp. 2361–2377, Aug. 2022.

[20] W. Fang, Z. Yu, Y. Jiang, Y. Shi, C. N. Jones, and Y. Zhou, "Communication-efficient stochastic zeroth-order optimization for federated learning," *IEEE Trans. Signal Process.*, vol. 70, pp. 5058–5073, 2022.

[21] X. Ma, H. Sun, Q. Wang, and R. Q. Hu, "User scheduling for federated learning through over-the-air computation," in *Proc. IEEE 94th Veh. Technol. Conf. (VTC-Fall)*, Norman, OK, USA, Sep. 2021, pp. 1–5.

[22] H. Guo, A. Liu, and V. K. Lau, "Analog gradient aggregation for federated learning over wireless networks: Customized design and convergence analysis," *IEEE Internet Things J.*, vol. 8, no. 1, pp. 197–210, Jan. 2021.

[23] M. Huh, D. Yu, and S.-H. Park, "Signal processing optimization for federated learning over multi-user MIMO uplink channel," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2021, pp. 495–498.

[24] Z. Wang et al., "Federated learning via intelligent reflecting surface," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 808–822, Feb. 2022.

[25] C. Xu, S. Liu, Z. Yang, Y. Huang, and K.-K. Wong, "Learning rate optimization for federated learning exploiting over-the-air computation," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3742–3756, Dec. 2021.

[26] Y. Hu et al., "Energy minimization for federated learning with IRS-assisted over-the-air computation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Oronto, ON, Canada, Jun. 2021, pp. 3105–3109.

[27] S. Xia, J. Zhu, Y. Yang, Y. Zhou, Y. Shi, and W. Chen, "Fast convergence algorithm for analog federated learning," in *Proc. IEEE Int. Conf. Commun.*, Montreal, QC, Canada, Jun. 2021, pp. 1–6.

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

[29] H. Liu, X. Yuan, and Y.-J.-A. Zhang, "CSIT-free model aggregation for federated edge learning via reconfigurable intelligent surface," *IEEE Wireless Commun. Lett.*, vol. 10, no. 11, pp. 2440–2444, Nov. 2021.

[30] Y. Chen, G. Zhu, and J. Xu, "Over-the-air computation with imperfect channel state information," in *Proc. IEEE 23rd Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Oulu, Finland, Jul. 2022, pp. 1–5.

[31] B. Jiang, J. Du, C. Jiang, Y. Shi, and Z. Han, "Communication-efficient device scheduling via over-the-air computation for federated learning," in *Proc. IEEE Global Commun. Conf.*, Rio de Janeiro, Brazil, Dec. 2022, pp. 173–178.

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

[33] F. Wang, J. Xu, V. K. N. Lau, and S. Cui, "Amplify-and-forward relaying for hierarchical over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10529–10543, Dec. 2022.

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

[35] R. Jiang and S. Zhou, "Cluster-based cooperative digital over-the-air aggregation for wireless federated edge learning," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Chongqing, China, Aug. 2020, pp. 887–892.

[36] X. Zhao, L. You, R. Cao, Y. Shao, and L. Fu, "Broadband digital over-the-air computation for asynchronous federated edge learning," 2021, *arXiv:2111.10508*.

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

[38] Z. Lin, X. Li, V. K. N. Lau, Y. Gong, and K. Huang, "Deploying federated learning in large-scale cellular networks: Spatial convergence analysis," *IEEE Trans. Wireless Commun.*, vol. 21, no. 3, pp. 1542–1556, Mar. 2022.

[39] H. Xing, O. Simeone, and S. Bi, "Federated learning over wireless device-to-device networks: Algorithms and convergence analysis," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3723–3741, Dec. 2021.

[40] A. Sahin, B. Everette, and S. S. M. Hoque, "Over-the-air computation with DFT-spread OFDM for federated edge learning," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Austin, TX, USA, Apr. 2022, pp. 1886–1891.

[41] S. Razavikia, J. M. B. D. Silva Jr., and C. Fischione, "Computing functions over-the-air using digital modulations," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Rome, Italy, Jun. 2023, pp. 5780–5786.

[42] L. Li et al., "Energy and spectrum efficient federated learning via high-precision over-the-air computation," 2022, *arXiv:2208.07237*.

[43] H. Jedda, A. Mezghani, A. L. Swindlehurst, and J. A. Nossek, "Quantized constant envelope precoding with PSK and QAM signaling," *IEEE Trans. Wireless Commun.*, vol. 17, no. 12, pp. 8022–8034, Dec. 2018.

[44] X. Zang, W. Liu, Y. Li, and B. Vucetic, "Over-the-air computation systems: Optimal design with sum-power constraint," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1524–1528, Sep. 2020.

[45] G. Zhu and K. Huang, "MIMO over-the-air computation for high-mobility multimodal sensing," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6089–6103, Aug. 2019.

[46] X. Zhai, G. Han, Y. Cai, Y. Liu, and L. Hanzo, "Simultaneously transmitting and reflecting (STAR) RIS assisted over-the-air computation systems," *IEEE Trans. Commun.*, vol. 71, no. 3, pp. 1309–1322, Mar. 2023.

[47] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, Jan. 2021.

[48] N. Zhang, M. Tao, and J. Wang, "Sum-rate-distortion function for indirect multiterminal source coding in federated learning," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Melbourne, VIC, Australia, Jul. 2021, pp. 2161–2166.

[49] Z. Gao, A. Koppel, and A. Ribeiro, "Balancing rates and variance via adaptive batch-size for stochastic optimization problems," *IEEE Trans. Signal Process.*, vol. 70, pp. 3693–3708, 2022.

[50] L. Liu, J. Liu, and D. Tao, "Variance reduced methods for non-convex composition optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5813–5825, Sep. 2022.

[51] S. Haykin, *Neural Networks and Learning Machines*. Cranbury, NJ, USA: Pearson Education, 2009.

[52] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[53] O. Abari, H. Rahul, D. Katabi, and M. Pant, "AirShare: Distributed coherent transmission made seamless," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Hong Kong, Apr. 2015, pp. 1742–1750.

[54] X. Zhai, G. Han, Y. Cai, and L. Hanzo, "Joint beamforming aided over-the-air computation systems relying on both BS-side and user-side reconfigurable intelligent surfaces," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10766–10779, Dec. 2022.

[55] F. Ren, C. Lin, and F. Liu, "Joint beamforming aided over-the-air computation systems relying on both BS-side and user-side reconfigurable intelligent surfaces," *IEEE Wireless Commun.*, vol. 15, no. 4, pp. 79–85, Aug. 2008.

[56] A. Liu, F. Zhu, and V. K. N. Lau, "Closed-loop autonomous pilot and compressive CSIT feedback resource adaptation in multi-user FDD massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 65, no. 1, pp. 173–183, Jan. 2017.

[57] M. del Rosario and Z. Ding, "Learning-based MIMO channel estimation under practical pilot sparsity and feedback compression," *IEEE Trans. Wireless Commun.*, vol. 22, no. 2, pp. 1161–1174, Feb. 2023.

[58] X. Luo, P. Cai, X. Zhang, D. Hu, and C. Shen, "A scalable framework for CSI feedback in FDD massive MIMO via DL path aligning," *IEEE Trans. Signal Process.*, vol. 65, no. 18, pp. 4702–4716, Sep. 2017.

[59] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.

[60] A. Krizhevsky. (Apr. 2009). *Learning Multiple Layers of Features From Tiny Images*. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[61] Y. Zhang, C. Xu, H. H. Yang, X. Wang, and T. Q. S. Quek, "DPP-based client selection for federated learning with NON-IID DATA," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Rhodes Island, Greece, Jun. 2023, pp. 1–5.

[62] S. Hosseinalipour et al., "Parallel successive learning for dynamic distributed model training over heterogeneous wireless networks," *IEEE/ACM Trans. Netw.*, vol. 32, no. 1, pp. 222–237, Feb. 2024.

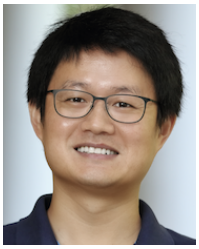
[63] Z. Wang et al., "Asynchronous federated learning over wireless communication networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 9, pp. 6961–6978, Sep. 2022.

[64] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2005.

[65] C. Xia, H. Guo, H. Ma, D. H. K. Tsang, and V. K. N. Lau, "Multi-resolution model compression for deep neural networks: A variational Bayesian approach," *IEEE Trans. Signal Process.*, vol. 72, pp. 1944–1959, 2024.

Sihua Wang (Student Member, IEEE) received the Ph.D. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2021. He is currently a Hong Kong Scholar Fellow with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong. He is also a Post-Doctoral Researcher with the School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications. His research interests include mobile edge computing, resource allocation, and machine learning in wireless networks.

Mingzhe Chen (Senior Member, IEEE) is currently an Assistant Professor with the Department of Electrical and Computer Engineering and the Frost Institute of Data Science and Computing, University of Miami. His research interests include federated learning, reinforcement learning, virtual reality, unmanned aerial vehicles, and the Internet of Things. He has received four IEEE Communication Society journal paper awards, including the IEEE Marconi Prize Paper Award in Wireless Communications in 2023, the Young Author Best Paper Award in 2021 and 2023, and the Fred W. Ellersick Prize Awards in 2022, and four Conference Best Paper Awards at ICCCN in 2023, IEEE WCNC in 2021, IEEE ICC in 2020, and IEEE GLOBECOM in 2020. He serves as an Associate Editor for IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE WIRELESS COMMUNICATIONS LETTERS, IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, and IEEE TRANSACTIONS ON MACHINE LEARNING IN COMMUNICATIONS AND NETWORKING.



Cong Shen (Senior Member, IEEE) received the B.E. and M.E. degrees from the Department of Electronic Engineering, Tsinghua University, China, and the Ph.D. degree in electrical engineering from the University of California at Los Angeles. He is currently an Assistant Professor with the Charles L. Brown Department of Electrical and Computer Engineering, University of Virginia. His general research interests include communications, wireless networks, and machine learning. He received the National Science Foundation CAREER Award in 2022 and the Best Paper Award in 2021 IEEE International Conference on Communications (ICC). He serves as an Associate Editor for IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, and IEEE TRANSACTIONS ON MACHINE LEARNING IN COMMUNICATIONS AND NETWORKING.

Changchuan Yin (Senior Member, IEEE) received the Ph.D. degree in signal and information processing from Beijing University of Posts and Telecommunications, Beijing, China, in 1998. In 2004, he was a Visiting Scholar with the Faculty of Science, The University of Sydney, Sydney, NSW, Australia. From 2007 to 2008, he held a visiting position with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA. He is currently a Professor with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications. His research interests include wireless networks and statistical signal processing. He was a co-recipient of the IEEE Guglielmo Marconi Prize Paper Award in 2023 and the IEEE International Conference on Wireless Communications and Signal Processing Best Paper Award in 2009. He has served as the symposium co-chair and a TPC member for many IEEE conferences.

Christopher G. Brinton (Senior Member, IEEE) received the M.S. and Ph.D. (Hons.) degrees in electrical engineering from Princeton University, in 2013 and 2016, respectively. He is currently the Elmore Rising Star Associate Professor of electrical and computer engineering (ECE) with Purdue University. Prior to joining Purdue University, he was the Associate Director of the EDGE Laboratory and a Lecturer in electrical engineering with Princeton University. He also co-founded Zoomi Inc., a big data startup company that has provided learning optimization to more than one million users worldwide and holds U.S. patents in machine learning for education. His book *The Power of Networks: Six Principles That Connect our Lives* and associated *Massive Open Online Courses* (MOOCs) have reached over 4,00,000 students. His research interests include the intersection of networking, communications, and machine learning, specifically in fog/edge network intelligence, distributed machine learning, and AI/ML-inspired wireless network optimization. He was a recipient of four of U.S. Top Early Career Awards from the National Science Foundation (CAREER), the Office of Naval Research (YIP), the Defense Advanced Research Projects Agency (YFA), and the Air Force Office of Scientific Research (YIP), the IEEE Communication Society William Bennett Prize Best Paper Award, the Intel Rising Star Faculty Award, the Qualcomm Faculty Award, and roughly $17M in sponsored research projects as the PI or the Co-PI. He has also been awarded Purdue College of Engineering Faculty Excellence Awards in Early Career Research, Early Career Teaching, and Online Learning. He serves as an Associate Editor for IEEE/ACM TRANSACTIONS ON NETWORKING and previously was an Associate Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.