







# Over-the-Air Hierarchical Personalized Federated Learning

Fangtong Zhou , Graduate Student Member, IEEE, Zhibin Wang , Member, IEEE, Hangguan Shan , Senior Member, IEEE, Liantao Wu, Member, IEEE, Xiaohua Tian , Senior Member, IEEE, Yuanming Shi , Senior Member, IEEE, and Yong Zhou , Senior Member, IEEE

**Abstract**—Data heterogeneity and communication bottleneck are two critical factors that limit the performance of federated learning (FL) over wireless networks. To address these challenges, this paper introduces a hierarchical personalized federated learning (HPFL) framework, which employs a three-tier network architecture to enable the simultaneous learning of a global model and multiple personalized local models. Meanwhile, over-the-air computation (AirComp) is leveraged to support communication-efficient device-to-edge and edge-to-cloud model aggregations. To provide useful guidance for enhancing learning performance, we derive the convergence bound of the proposed AirComp-assisted HPFL, taking into account the interference among different clusters as well as data heterogeneity across different devices. To minimize the impact of accumulated transmission distortion on learning performance, we formulate an optimization problem involving the beamforming design at both cloud and edge servers, followed by developing a successive convex approximation-based algorithm at the cloud server and an interference-aware algorithm at each edge server to perform the receive beamforming design. Simulation

results demonstrate that our proposed framework outperforms other FL frameworks and transceiver design algorithms in terms of test accuracy.

**Index Terms**—Federated learning (FL), over-the-air computation, interference management, hierarchical architecture.

## I. INTRODUCTION

FEDERATED learning (FL), as a promising distributed machine learning (ML) paradigm, emerges to enable collaborative model training with distributed devices while ensuring that the sensitive data of each device remains local and private. FL has promising applications in various emerging scenarios, including industrial Internet of Things, smart healthcare, and autonomous driving [2], [3], [4], [5]. However, deploying FL in wireless networks faces critical challenges of severe communication bottlenecks and inevitable statistical heterogeneity.

The transmission of large-scale local models between numerous devices and the cloud server introduces a significant amount of communication load and leads to severe network congestion. Meanwhile, the long distance between devices and the cloud server makes the transmission unreliable, leading to low training efficiency. To this end, hierarchical FL (HFL) [6], [7], [8] emerges as a promising framework to enhance communication efficiency and reduce transmission delay, given the fact that it is capable of alleviating the adverse effects of channel fading by enabling devices to send local models to edge servers in close proximity, which then transmit the edge models to a central server after multiple rounds of local model aggregation and dissemination. The edge servers and the cloud server are wirelessly connected in many practical scenarios. For instance, in autonomous vehicular networks, the edge servers (e.g., roadside units) aggregate the trained models from vehicles and send updates to the cloud server through wireless networks. Guided by the convergence analysis of HFL, advanced device clustering strategies [9] and efficient quantization schemes [10] can be integrated with the optimization of aggregation intervals in different tiers to enhance the convergence performance. However, the aforementioned studies did not take into account how the wireless network parameters affect the learning performance of HFL. To fill this gap, resource allocation among mobile devices was optimized in [11] to reduce the communication cost, which, however, did not take into account statistical heterogeneity. A joint communication and computation resource allocation problem under the HFL framework was formulated in [12], where

Received 28 June 2024; revised 20 October 2024; accepted 10 November 2024. Date of publication 15 November 2024; date of current version 5 March 2025. The work of Yong Zhou was supported in part by the National Natural Science Foundation of China under Grant U20A20159 and in part by the National Science Foundation of Shanghai under Grant 23ZR1442800. The work of Hangguan Shan was supported in part by the National Natural Science Foundation of China (NSFC) under Grant U21B2029 and Grant U21A20456, in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LR23F010006, in part by the Science and Technology Development Fund under Grant SKLIOTSC(UM)-2024-2026, and in part by the State Key Laboratory of Internet of Things for Smart City, University of Macau, under Grant SKL-IoTSC(UM)-2024-2026/ORP/GA01/2023. The work of Liantao Wu was supported by the National Natural Science Foundation of China (NSFC) under Grant 62202307. The work of Yuanming Shi was supported in part by the National Natural Science Foundation of China under Grant 62271318 and in part by Shanghai Rising-Star Program under Grant 22QA1406100. An earlier version of this paper was presented at the IEEE International Conference on Communications, Rome, Italy, May, 2023 [DOI: 10.1109/ICC45041.2023.10278799]. The review of this article was coordinated by Dr. Zehui Xiong. (Corresponding author: Yong Zhou.)

Fangtong Zhou, Zhibin Wang, Yuanming Shi, and Yong Zhou are with the School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China (e-mail: zhouft@shanghaitech.edu.cn; wangzhb@shanghaitech.edu.cn; shiym@shanghaitech.edu.cn; zhouyong@shanghaitech.edu.cn).

Hanguan Shan is with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: hshan@zju.edu.cn).

Liantao Wu is with the Software Engineering Institute, Shanghai Key Laboratory of Trustworthy Computing, East China Normal University, Shanghai 200050, China (e-mail: ltwu@sei.ecnu.edu.cn).

Xiaohua Tian is with the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: xtian@sjtu.edu.cn).

Digital Object Identifier 10.1109/TVT.2024.3499349

a resource scheduling algorithm was proposed. However, the aforementioned works on HFL did not exploit the unique feature of FL (i.e., the edge/cloud server only needs the aggregation of local/edge models for model updating) to facilitate efficient model aggregation, thereby leading to excessive convergence delay, especially when the spectral resource is limited.

Over-the-air computation (AirComp) [13], [14] is capable of achieving low-latency model aggregation by enabling devices to transmit their local models or gradients to the server simultaneously. Specifically, AirComp exploits the superposition characteristic of wireless channels to perform model aggregation directly over the air, without requiring the server to decode each local model. Meanwhile, for AirComp, the number of required radio resources remains unaffected by the number of participating devices, resulting in high spectral efficiency compared to orthogonal multiple access (OMA) counterparts. Various strategies were proposed to align the magnitudes and phase shifts of different signals at the receiver to facilitate AirComp, including transmit power control [15], receive beamforming design [16], and error control and channel estimation [17]. The effectiveness of AirComp-assisted FL was verified in [18], where the authors maximize the number of clients participating in model training under the model aggregation error constraint. In [19], the authors designed a low-latency FL framework, where the local models are uploaded to the server through broadband channels using AirComp. A dynamic device scheduling scheme was developed in [20], where the learning performance with energy constraints of local computation and gradient transmission was optimized. In addition, a transmit power control strategy was developed in [21] to decrease the aggregation error in AirComp-assisted FL. The authors utilized the intelligent reflecting surface (IRS) to achieve fast and reliable aggregation in AirComp-assisted FL in [22], where a two-step optimization algorithm was proposed to maximize the number of selected devices. Moreover, AirComp was leveraged to assist the zeroth-order FL to promote learning performance [23]. Nevertheless, all the aforementioned works cannot be directly extended to HFL because the transceiver design should account for extra noise introduced by edge model aggregation. To fill this gap, the authors utilized AirComp in HFL in [24], [25], which, however, assumed that the model aggregation between the edge and cloud servers was error-free. Furthermore, the authors in [26] utilized AirComp in both cloud and edge model aggregation, which, however, only considered a single edge communication round and ignored the accumulated model aggregation error. Moreover, neither of the aforementioned AirComp-assisted HFL accounted for co-channel interference among different clusters.

Statistical heterogeneity is another notable challenge in determining the performance of FL [27], [28]. Traditional FL performs poorly when the data of different devices are not independent and identically distributed (non-i.i.d.), because the generalization error of the global model increases with the data heterogeneity. Hence, personalized FL (PFL) was proposed to build models that are tailored to individual devices under the condition of statistical heterogeneity [29], [30], [31], [32]. Global model personalization and learning personalized models were two major strategies in PFL [29]. With the first strategy, the model personalization performance was improved by training

a globally shared FL model on heterogeneous data. FedProx in [33] and Scaffold in [34] were two well-known model-based approaches to develop a robust global model for each device's personalization and enhance the local model's adaptation performance through regularization, respectively. With the second strategy, the personalized models were trained individually for each device. FedMd in [35] was an architecture-based method to achieve personalization by designing a customized model specifically tailored to each client. MOCHA in [36] extended distributed multi-task learning (MTL) to the FL framework and achieved personalization by treating each device as a task in MTL. Leveraging personalized techniques in HFL combines the advantages of both, while introducing extra challenges, e.g., the edge aggregation leads to an increase in model diversity, which intensifies the complexity of personalized algorithm design. The authors in [37] developed a dynamic weighting scheme and deployed PFL in hierarchical architectures, which, however, did not take into account the impact of multiple edge communication rounds on the model convergence, and assumed the transmission within each cluster to be error-free.

#### A. Challenges and Contributions

In this paper, we develop an AirComp-assisted hierarchical personalized FL (HPFL) framework, which leverages a device-edge-cloud hierarchical architecture to jointly optimize loss functions on both the edge and device sides. Each edge server and the nearby devices form a cluster, and devices use their local data to train personalized models and transmit intermediate local models to corresponding edge servers, which further upload edge models to the cloud server to aggregate for a global model. Applying AirComp for both cloud and edge model aggregation faces the following challenges. First, deploying AirComp in a hierarchical architecture leads to the propagation of the error accumulated at the edge server to the cloud server, making the convergence analysis more complicated. Second, co-channel interference leads to the coupling of AirComp transceiver design among different clusters, requiring interference management to jointly optimize the overall learning performance. Third, since the number of devices may be large in hierarchical networks, it is challenging to efficiently design the transceiver with high accuracy and low complexity. Fourth, the model updates transmitted by different devices may not be directly compatible, as they might represent different levels of personalization. Aggregating such heterogeneous updates via AirComp is non-trivial. To cope with these challenges, we perform beamforming design to align the signal amplitude within each cluster and mitigate co-channel interference among different clusters, which in turn reduces the aggregation error on both edge and cloud servers. Our main contributions are summarized as follows:

- We propose an AirComp-assisted HPFL, where a cloud-edge-device structure is leveraged to concurrently train a global model and multiple local models. AirComp is adopted in both cloud model aggregation (i.e., the cloud server aggregates edge model updates from all edge servers) and edge model aggregation (i.e., each edge server aggregates local model updates from its associated devices) to realize low-latency model aggregation.



- Section II introduces the proposed AirComp-assisted HPFL framework. Section III shows the convergence analysis and the problem formulation. In Sections IV and V, we present the receive beamforming design algorithm at the cloud side and

Consider a hierarchical FL framework that consists of devices, edge servers, and the cloud server, as depicted in Fig. 1. In this setup, a single cloud server establishes wireless connections with  $N$  edge servers, where each of them then associates with  $M$  nearby devices. These devices possess non-i.i.d. local datasets. By representing device  $j$  from cluster  $i$  as  $c_{i,j}$ , we denote its local dataset as  $\mathcal{B}_{i,j}$ , the set of edge servers as  $\mathcal{N}$  with  $|\mathcal{N}| = N$ , and the set of devices within cluster  $i$  as  $\mathcal{M}_i$ , where  $i \in \mathcal{N}$ , and  $|\mathcal{M}_i| = M$ . To address data heterogeneity, we perform joint optimization of the global model  $\mathbf{w} \in \mathbb{R}^D$  and the local personalized model  $\boldsymbol{\theta}_{i,j} \in \mathbb{R}^D$  by solving the following problem

$$F_i(\mathbf{w}) = \frac{1}{M} \sum_{j=1}^M F_{i,j}(\mathbf{w}), \quad (2)$$

where  $F_{i,j}(\mathbf{w})$  represents the loss function of device  $c_{i,j}$  and can be expressed as

$$F_{i,j}(\mathbf{w}) = \min_{\mathbf{y}_{i,j}, \boldsymbol{\theta}_{i,j}} \ell_{i,j}(\boldsymbol{\theta}_{i,j}) + \frac{\lambda_1}{2} \|\boldsymbol{\theta}_{i,j} - \mathbf{y}_{i,j}\|_2^2 + \frac{\lambda_2}{2} \|\mathbf{y}_{i,j} - \mathbf{w}\|_2^2, \quad (3)$$

with  $\ell_{i,j}(\cdot)$  denoting the sample-wise loss function of dataset  $\mathcal{B}_{i,j}$  and  $\mathbf{y}_{i,j} \in \mathbb{R}^D$  denoting the intermediate local model at device  $c_{i,j}$ . At edge server  $i$ ,  $\mathbf{y}_i \in \mathbb{R}^D$  is denoted as the intermediate edge model, which is an aggregation of  $\mathbf{y}_{i,j}$ . The second and third terms in (3) enable each device to find its local model  $\boldsymbol{\theta}_{i,j}$  within an appropriate distance to the intermediate model  $\mathbf{y}_{i,j}$  as well as the global model  $\mathbf{w}$ . By adjusting coefficients  $\lambda_1$  and  $\lambda_2$ , the extent of local model personalization can be controlled. In the following, we denote the  $r$ -th edge communication round within the  $t$ -th global communication round as  $(t, r)$  for simplicity.

Problem (1) is solved by iteratively performing the following steps.

- The cloud server broadcasts global model  $\mathbf{w}^t$  to all edge servers at the  $t$ -th global round.
- Upon receiving  $\mathbf{w}^t$ , edge server  $i$  initializes intermediate edge model  $\mathbf{y}_i^{t,0}$  and edge model  $\mathbf{w}_i^{t,0}$  as  $\mathbf{w}^t$ . Subsequently, it disseminates  $\mathbf{y}_i^{t,0}$  to its associated devices to start the edge iteration.
- As for device  $c_{i,j}$ , upon receiving  $\mathbf{y}_i^{t,r}$  at communication round  $(t, r)$ , it updates the personalized local model  $\boldsymbol{\theta}_{i,j}^{t,r}$  by solving the following problem

$$\boldsymbol{\theta}_{i,j}^{t,r} = \arg \min_{\boldsymbol{\theta}_{i,j}} \ell_{i,j}(\boldsymbol{\theta}_{i,j}) + \frac{\lambda_1}{2} \|\boldsymbol{\theta}_{i,j} - \mathbf{y}_i^{t,r}\|_2^2. \quad (4)$$

- When each local model is done updating, edge server  $i$  performs edge aggregation to obtain  $\mathbf{y}_i^{t,r+1}$  and updates edge model  $\mathbf{w}_i^{t,r+1}$ . Intermediate edge model  $\mathbf{y}_i^{t,r+1}$  is the aggregation of  $\mathbf{y}_{i,j}^{t,r+1}$ , which is obtained by solving (5) at the edge server:

$$\mathbf{y}_i^{t,r+1} = \arg \min_{\mathbf{y}_{i,j}} \frac{\lambda_1}{2} \|\boldsymbol{\theta}_{i,j}^{t,r} - \mathbf{y}_{i,j}\|_2^2 + \frac{\lambda_2}{2} \|\mathbf{y}_{i,j} - \mathbf{w}_i^{t,r}\|_2^2. \quad (5)$$

- Upon completing edge update for  $R$  times, the edge model  $\mathbf{w}_i^{t,R}$  is transmitted to the cloud server by edge server  $i$ . Finally, the global model at the cloud server is updated as

$$\mathbf{w}^{t+1} = (1 - \beta)\mathbf{w}^t + \frac{\beta}{N} \sum_{i=1}^N \mathbf{w}_i^{t,R}, \quad (6)$$

where  $\beta \in (0, 1]$  is a constant parameter to control the convergence speed [31]. Afterward, the cloud server disseminates global model  $\mathbf{w}^{t+1}$  to all edge servers to launch a new round of training.

To address problem (4), we randomly select a mini-batch of dataset  $\mathcal{D}_{i,j}$  from  $\mathcal{B}_{i,j}$  to calculate a stochastic gradient as

follows,

$$\nabla \tilde{\ell}_{i,j}(\boldsymbol{\theta}_{i,j}, \mathcal{D}_{i,j}) = \frac{1}{|\mathcal{D}_{i,j}|} \sum_{\xi_{i,j} \in \mathcal{D}_{i,j}} \nabla \ell_{i,j}(\boldsymbol{\theta}_{i,j}, \xi_{i,j}). \quad (7)$$

Here  $\mathbb{E}[\nabla \tilde{\ell}_{i,j}(\boldsymbol{\theta}_{i,j}, \mathcal{D}_{i,j})] = \nabla \ell_{i,j}(\boldsymbol{\theta}_{i,j})$ ,  $\xi_{i,j}$  represents the training data randomly sampled from  $\mathcal{D}_{i,j}$ , and  $|\mathcal{D}_{i,j}|$  represents the cardinality of  $\mathcal{D}_{i,j}$ . Since closed-form  $\boldsymbol{\theta}_{i,j}$  is not obtainable, we derive its approximation, denoted as  $\tilde{\boldsymbol{\theta}}_{i,j}$ , through minimizing

$$h(\boldsymbol{\theta}_{i,j}; \mathbf{y}_i^{t,r}, \mathcal{D}_{i,j}) = \tilde{\ell}_{i,j}(\boldsymbol{\theta}_{i,j}, \mathcal{D}_{i,j}) + \frac{\lambda_1}{2} \|\boldsymbol{\theta}_{i,j} - \mathbf{y}_i^{t,r}\|_2^2. \quad (8)$$

Problem (8) is solved through Nesterov's accelerated gradient descent [38]. The iteration continues until

$$\|\nabla h(\tilde{\boldsymbol{\theta}}_{i,j}; \mathbf{y}_i^{t,r}, \mathcal{D}_{i,j})\|_2^2 \leq \nu, \quad (9)$$

where  $\nu$  is a preset positive constant used to control the accuracy of convergence. We set  $\nu$  to 0.0001 in the later experiment.

### B. Model Aggregation Via AirComp

In order to attain low communication latency, we adopt AirComp for both edge model aggregation (i.e., each edge server aggregates local model updates from its associated devices) and global model aggregation (i.e., the cloud server aggregates edge model updates from all edge servers).

1) *Edge Model Aggregation*: To facilitate the transmit power control of devices, we normalize the  $D$ -dimensional local model  $\boldsymbol{\theta}_{i,j}^{t,r}$  before the uplink transmission. In particular, device  $c_{i,j}$  calculates mean  $\bar{\boldsymbol{\theta}}_{i,j}^{t,r}$  and variance  $(\pi_{i,j}^{t,r})^2$  of  $\boldsymbol{\theta}_{i,j}^{t,r}$  as  $\bar{\boldsymbol{\theta}}_{i,j}^{t,r} = \frac{1}{D} \sum_{d=1}^D \boldsymbol{\theta}_{i,j,d}^{t,r}$ ,  $\forall j \in \mathcal{M}_i, \forall i \in \mathcal{N}$  and  $(\pi_{i,j}^{t,r})^2 = \frac{1}{D} \sum_{d=1}^D (\boldsymbol{\theta}_{i,j,d}^{t,r} - \bar{\boldsymbol{\theta}}_{i,j}^{t,r})^2$ ,  $\forall j \in \mathcal{M}_i, \forall i \in \mathcal{N}$ , where  $\boldsymbol{\theta}_{i,j,d}^{t,r}$  denotes the  $d$ -th element of  $\boldsymbol{\theta}_{i,j}^{t,r}$ . By denoting  $\bar{\boldsymbol{\theta}}_i^{t,r} = \frac{1}{M} \sum_{j=1}^M \bar{\boldsymbol{\theta}}_{i,j}^{t,r}$  and  $(\pi_i^{t,r})^2 = \frac{1}{M} \sum_{j=1}^M (\pi_{i,j}^{t,r})^2$ ,  $\boldsymbol{\theta}_{i,j}^{t,r}$  is normalized to have zero mean and unit variance as

$$\mathbf{s}_{i,j}^{t,r} = \frac{\boldsymbol{\theta}_{i,j}^{t,r} - \bar{\boldsymbol{\theta}}_{i,j}^{t,r}}{\pi_{i,j}^{t,r}}, \quad \forall j \in \mathcal{M}, \forall i \in \mathcal{N}. \quad (10)$$

Simultaneously, each device  $c_{i,j}$ , equipped with a single transmit antenna, transmits the normalized local model  $\{\mathbf{s}_{i,j}^{t,r}\}_{j=1}^M$  to its associated edge server  $i$ , which are equipped with  $K$  receive antennas. In communication round  $(t, r)$ , we denote  $\mathbf{h}_{i,j}^{t,r} \in \mathbb{C}^K$  and  $\mathbf{h}_{i,j'}^{t,r} \in \mathbb{C}^K$  as the channel coefficients between edge server  $i$  and device  $c_{i,j}$ , and between edge server  $i$  and devices from cluster  $\ell \in \mathcal{N} \setminus \{i\}$  (i.e.,  $c_{\ell,j'}$ ), respectively. We assume that each channel follows block fading. Note that although the local models of other clusters can also contribute to the global model training, long-distance propagation of the cross-cluster links leads to poor channel conditions that may become the performance-limiting factor of AirComp-based model aggregation, as the model aggregation error is determined by the links with the worst channel conditions. As a result, to facilitate scalable model aggregation, the local model updates from neighboring clusters are considered as co-channel interference. By denoting  $w_{i,j}^{t,r} \in \mathbb{C}$  as the transmit scalar of device  $c_{i,j}$ , the received local models'



aggregation at edge server  $i$  is

$$\begin{aligned} \tilde{q}_i^{t,r+1}(d) &= \sum_{j=1}^M \mathbf{h}_{i,j}^{t,r} w_{i,j}^{t,r} s_{i,j}^{t,r}(d) \\ &+ \underbrace{\sum_{l \in \mathcal{N} \setminus \{i\}} \sum_{j'=1}^M \mathbf{h}_{i,j'}^{t,r} w_{l,j'}^{t,r} s_{l,j'}^{t,r}(d)}_{\text{inter-cluster interference}} + \underbrace{\mathbf{n}_i^{t,r}}_{\text{noise}}, \end{aligned} \quad (11)$$

where  $\mathbf{n}_i^{t,r} \sim \mathcal{CN}(0, \sigma^2 \mathbf{I}_K)$  is the additive white Gaussian noise (AWGN) with zero mean and variance  $\sigma^2$ . In practical scenarios, all devices adhere to a maximum power constraint, i.e.,  $|w_{i,j}^{t,r}|^2 \leq P$ . We apply the receive beamforming at edge server  $i$  via multiplying both sides of (11) by  $\frac{\mathbf{m}_i^H}{\sqrt{\eta_i}}$ , the estimation of  $\tilde{q}_i^{t,r+1}(d)$  is calculated as

$$\begin{aligned} \check{q}_i^{t,r+1}(d) &= \frac{1}{\sqrt{\eta_i}} \mathbf{m}_i^H \tilde{q}_i^{t,r+1}(d) \\ &= \frac{1}{\sqrt{\eta_i}} \mathbf{m}_i^H \sum_{j=1}^M \mathbf{h}_{i,j}^{t,r} w_{i,j}^{t,r} s_{i,j}^{t,r}(d) \\ &+ \frac{1}{\sqrt{\eta_i}} \mathbf{m}_i^H \sum_{\ell \in \mathcal{N} \setminus \{i\}} \sum_{j'=1}^M \mathbf{h}_{i,j'}^{t,r} w_{\ell,j'}^{t,r} s_{\ell,j'}^{t,r}(d) \\ &+ \frac{1}{\sqrt{\eta_i}} \mathbf{m}_i^H \mathbf{n}_i^{t,r}, \end{aligned} \quad (12)$$

where  $\mathbf{m}_i \in \mathbb{C}^K$  and  $\eta_i$  respectively denote the receive beamforming vector and the de-noising factor at edge server  $i$ . The de-normalized signal at edge server  $i$  is obtained as follows,

$$\begin{aligned} \check{\theta}_i^{t,r+1} &= \frac{1}{M} \left( \pi_i^{t,r} \check{q}_i^{t,r+1} + M \bar{\theta}_i^{t,r} \right) \\ &= \frac{1}{M} \pi_i^{t,r} \left( \check{q}_i^{t,r+1} - \mathbf{q}_i^{t,r+1} \right) + \theta_i^{t,r+1}, \end{aligned} \quad (13)$$

where  $\theta_i^{t,r+1} = \frac{1}{M} \sum_{j=1}^M \frac{\lambda_1}{\lambda_1 + \lambda_2} \theta_{i,j}^{t,r}$  and  $\mathbf{q}_i^{t,r+1} = \sum_{j=1}^M \frac{\lambda_1}{\lambda_1 + \lambda_2} s_{i,j}^{t,r}$  are the target received signal and the target normalized received signal, respectively. To update  $\mathbf{y}_i^{t,r+1}$ , we solve (5) and express the new aggregation of the intermediate models with de-normalized estimated function as

$$\check{\mathbf{y}}_i^{t,r+1} = \frac{\lambda_1}{\lambda_1 + \lambda_2} \check{\theta}_i^{t,r+1} + \frac{\lambda_2}{\lambda_1 + \lambda_2} \mathbf{w}_i^{t,r}. \quad (14)$$

Subsequently, the edge model is updated as

$$\begin{aligned} \mathbf{w}_i^{t,r+1} &= \mathbf{w}_i^{t,r} - \eta_1 \nabla F_i(\mathbf{w}_i^{t,r}) \\ &= \mathbf{w}_i^{t,r} - \eta_1 \lambda_2 (\mathbf{w}_i^{t,r} - \check{\mathbf{y}}_i^{t,r+1}). \end{aligned} \quad (15)$$

**2) Global Model Aggregation:** Similar to edge model aggregation, after  $R$  rounds of edge updates, we normalize the  $D$ -dimensional edge model  $\mathbf{w}_i^{t,R}$  before transmitting to the cloud server as  $\bar{w}_i^{t,R} = \frac{1}{D} \sum_{d=1}^D w_i^{t,R}(d)$ ,  $\forall i \in \mathcal{N}$  and  $(\pi_i^{t,R})^2 = \frac{1}{D} \sum_{d=1}^D (w_i^{t,R}(d) - \bar{w}_i^{t,R})^2$ ,  $\forall i \in \mathcal{N}$ .

By denoting  $\bar{w}^{t,R} = \frac{1}{N} \sum_{i=1}^N \bar{w}_i^{t,R}$  and  $(\pi^{t,R})^2 = \frac{1}{N} \sum_{i=1}^N (\pi_i^{t,R})^2$ ,  $\mathbf{w}_i^{t,R}$  can be normalized as

$$\mathbf{s}_i^{t,R} = \frac{\mathbf{w}_i^{t,R} - \bar{w}^{t,R}}{\pi^{t,R}}, \quad \forall i \in \mathcal{N}. \quad (16)$$

All edge servers transmit the normalized edge models  $\{\mathbf{s}_i^{t,R}\}_{i=1}^N$  to the cloud server over wireless fading channels. Similarly, we assume that  $\{\mathbf{s}_i^{t,R}\}_{i=1}^N$  are independent, i.e.,  $\mathbb{E}[\mathbf{s}_i^{t,R}(\mathbf{s}_k^{t,R})^H] = \mathbf{0}$ ,  $\forall i \neq k$  [39]. The edge servers are equipped with a single transmit antenna and the cloud server is equipped with  $K$  receive antennas. Our work can also be extended to the scenario where each edge server is equipped with multiple transmit antennas, as in [40]. The channel coefficient between the  $i$ -th edge server and the cloud server in the  $t$ -th global round is denoted as  $\mathbf{h}_i^t \in \mathbb{C}^K$ . We denote  $w_i^t \in \mathbb{C}$  as the transmit scalar of the  $i$ -th edge server. The aggregation of the edge models at the cloud server is

$$\tilde{\mathbf{q}}^{t,R}(d) = \sum_{i=1}^N \mathbf{h}_i^t w_i^t s_i^{t,R}(d) + \mathbf{n}^t, \quad (17)$$

where  $\mathbf{n}^t \sim \mathcal{CN}(0, \sigma^2 \mathbf{I}_K)$  is the AWGN at the cloud server. The target received signal at the cloud server is  $\mathbf{q}^{t,R}(d) = \sum_{i=1}^N s_i^{t,R}(d)$ . With limited transmit power  $|w_i^{t,R}|^2 \leq P$ , the estimation of  $\tilde{\mathbf{q}}^{t,R}(d)$  is calculated as

$$\begin{aligned} \check{\mathbf{q}}^{t,R}(d) &= \frac{1}{\sqrt{\zeta}} \mathbf{v}^H \tilde{\mathbf{q}}^{t,R}(d) \\ &= \frac{1}{\sqrt{\zeta}} \mathbf{v}^H \sum_{i=1}^N \mathbf{h}_i^{t,R} w_i^{t,R} s_i^{t,R}(d) + \frac{1}{\sqrt{\zeta}} \mathbf{v}^H \mathbf{n}^{t,R}, \end{aligned} \quad (18)$$

where  $\mathbf{v} \in \mathbb{C}^K$  and  $\zeta$  denote the receive beamformer and the denoising factor at the cloud server, respectively. The received signal is then de-normalized as

$$\begin{aligned} \check{\mathbf{w}}^{t,R} &= \frac{1}{N} (\pi^{t,R} \check{\mathbf{q}}^{t,R} + N \bar{w}^{t,R}) \\ &= \frac{1}{N} \pi^{t,R} (\check{\mathbf{q}}^{t,R} - \mathbf{q}^{t,R}) + \mathbf{w}^{t,R}, \end{aligned} \quad (19)$$

where  $\mathbf{w}^{t,R} = \frac{1}{N} \sum_{i=1}^N \mathbf{w}_i^{t,R}$ . To update the global model  $\mathbf{w}^t$ , we rewrite (15) as

$$\begin{aligned} \mathbf{w}_i^{t,r+1} &= \mathbf{w}_i^{t,r} - \eta_1 \lambda_2 (\mathbf{w}_i^{t,r} - \check{\mathbf{y}}_i^{t,r+1}) \\ &= \mathbf{w}_i^{t,r} - \eta_1 \underbrace{\frac{\lambda_1}{\lambda_1 + \lambda_2} \lambda_2 (\mathbf{w}_i^{t,r} - \check{\theta}_i^{t,r+1})}_{\check{\mathbf{g}}_i^{t,r}}. \end{aligned} \quad (20)$$

Accumulating (20) over  $r$ , we have

$$\eta_1 \sum_{r=0}^{R-1} \check{\mathbf{g}}_i^{t,r} = \sum_{r=0}^{R-1} (\mathbf{w}_i^{t,r} - \mathbf{w}_i^{t,r+1}) = \mathbf{w}^t - \mathbf{w}_i^{t,R}, \quad (21)$$

where  $\check{\mathbf{g}}_i^{t,r}$  can be regarded as the biased estimate of  $\nabla F_i(\mathbf{w}_i^{t,r})$ . Then we can rewrite (6) as

$$\mathbf{w}^{t+1} = (1 - \beta) \mathbf{w}^t + \beta \check{\mathbf{w}}^{t,R}$$

$$\begin{aligned}
&= (1 - \beta) \mathbf{w}^t + \beta \left[ \frac{1}{N} \pi^{t,R} (\tilde{\mathbf{q}}^{t,R} - \mathbf{q}^{t,R}) + \mathbf{w}^{t,R} \right] \\
&= \mathbf{w}^t - \frac{\beta}{N} \sum_{i=1}^N (\mathbf{w}^t - \mathbf{w}_i^{t,R}) + \frac{\beta}{N} \pi^{t,R} (\tilde{\mathbf{q}}^{t,R} - \mathbf{q}^{t,R}) \\
&= \mathbf{w}^t - \underbrace{\eta_1 \beta R}_{\tilde{\eta}} \underbrace{\left[ \frac{1}{NR} \sum_{i=1}^N \sum_{r=0}^{R-1} \tilde{\mathbf{g}}_i^{t,r} - \frac{\beta}{\tilde{\eta} N} \pi^{t,R} (\tilde{\mathbf{q}}^{t,R} - \mathbf{q}^{t,R}) \right]}_{\tilde{\mathbf{g}}^t}, \quad (22)
\end{aligned}$$

where  $\tilde{\mathbf{g}}^t$  and  $\tilde{\eta}$  are the learning rate and approximate stochastic gradient of the global update, respectively. It is essential to note that  $\tilde{\mathbf{g}}^t$  encompasses the aggregation error arising from inter-cluster interference, channel fading, and receiver noise, which impedes learning convergence. In the absence of communication errors, the biased estimate of  $\nabla F_i(\mathbf{w}_i^t)$  is

$$\begin{aligned}
\mathbf{g}^t &= \frac{1}{NR} \sum_{i=1}^N \sum_{r=0}^{R-1} \mathbf{g}_i^{t,r} \\
&= \frac{1}{NR} \sum_{i=1}^N \sum_{r=0}^{R-1} \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2} (\mathbf{w}_i^{t,r} - \boldsymbol{\theta}_i^{t,r+1}). \quad (23)
\end{aligned}$$

Hence, the gradient aggregation error at the cloud server can be calculated as  $\mathbf{e}^t = \tilde{\mathbf{g}}^t - \mathbf{g}^t$ .

The following Algorithm 1 illustrates the process of the proposed AirComp-assisted HPFL.

### III. CONVERGENCE ANALYSIS AND PROBLEM FORMULATION

#### A. Assumptions

*Assumption 1:* (Smoothness) The local loss function  $\ell_{i,j}(\cdot)$  is nonconvex and L-smooth on  $\mathbb{R}^D$ , i.e.,

$$\|\nabla \ell_{i,j}(\mathbf{x}) - \nabla \ell_{i,j}(\mathbf{x}')\|_2 \leq L \|\mathbf{x} - \mathbf{x}'\|_2. \quad (24)$$

*Assumption 2:* (Variance bound) The stochastic gradient with sampling noise is characterized by a bounded variance, denoted as  $\gamma_i^2 \geq 0$ , i.e.,

$$\mathbb{E}_{\mathcal{D}_{i,j}} \left\| \nabla \tilde{\ell}_{i,j}(\mathbf{w}; \mathcal{D}_{i,j}) - \nabla \ell_{i,j}(\mathbf{w}) \right\|^2 \leq \gamma_i^2, \quad (25)$$

where  $\mathcal{D}_{i,j}$  is the mini-batch training dataset from device  $c_{i,j}$ .

*Assumption 3:* (Dissimilarity bound) The disparity between local and global loss functions has an upper bound of constant  $\sigma_i^2 \geq 0$  as

$$\frac{1}{NM} \sum_{i,j=1}^{N,M} \|\nabla \ell_{i,j}(\mathbf{w}) - \nabla \ell(\mathbf{w})\|^2 \leq \sigma_i^2. \quad (26)$$

*Assumption 4:* (Local and edge models variance bound) The variance of  $D$  elements of local model  $\boldsymbol{\theta}_{i,j}$  and edge model  $\mathbf{w}_i$  have an upper bound of constant  $\Gamma \geq 0$ , i.e.,

$$\max\{\pi_{i,j}^2, \pi_i^2\} \leq \Gamma. \quad (27)$$

*Remark 1:* Assumptions 1, 2, and 3 find widespread application in the convergence analysis of FL [41]. Since the elements

---

#### Algorithm 1: AirComp-Assisted Hierarchical Personalized Federated Learning.

---

**Input:** Hyperparameters  $\lambda_1, \lambda_2, \beta, \nu$ , learning rate  $\eta_1$ , global and edge communication round  $T, R$ , initial global model  $\mathbf{w}^0$ .

##### # Cloud server

**For**  $t = 0, 1, \dots, T - 1$ :

##### Global Model Update:

$$\mathbf{w}^{t+1} = (1 - \beta) \mathbf{w}^t + \frac{\beta}{N} \sum_{i=1}^N \mathbf{w}_i^{t,R}.$$

##### Global Model Dissemination:

Broadcast model  $\mathbf{w}^{t+1}$  to all edge servers.

##### # Edge Server

$$\mathbf{y}_i^{t,0} = \mathbf{w}_i^{t,0} = \mathbf{w}^t.$$

**For**  $r = 0, 1, \dots, R - 1$ :

##### Intermediate Edge Model Update:

$$\tilde{\mathbf{y}}_i^{t,r+1} = \frac{\lambda_1}{\lambda_1 + \lambda_2} \tilde{\boldsymbol{\theta}}_i^{t,r+1} + \frac{\lambda_2}{\lambda_1 + \lambda_2} \mathbf{w}_i^{t,r}.$$

##### Edge Model Update:

$$\mathbf{w}_i^{t,r+1} = \mathbf{w}_i^{t,r} - \eta_1 \lambda_2 (\mathbf{w}_i^{t,r} - \tilde{\mathbf{y}}_i^{t,r+1}).$$

##### Edge Model Dissemination:

Broadcast intermediate edge model  $\mathbf{y}_i^{t,r+1}$  to its associated devices.

##### Model Normalization:

$$\mathbf{s}_i^{t,R} = \frac{\mathbf{w}_i^{t,R} - \bar{\mathbf{w}}^{t,R}}{\pi_i^{t,R}}.$$

Transmit normalized edge model  $\mathbf{s}_i^{t,R}$  to the cloud server through AirComp in (17).

##### # Device

##### Local Model Update:

$$\boldsymbol{\theta}_{i,j}^{t,r} = \arg \min_{\boldsymbol{\theta}_{i,j}} \ell_{i,j}(\boldsymbol{\theta}_{i,j}) + \frac{\lambda_1}{2} \|\boldsymbol{\theta}_{i,j} - \mathbf{y}_i^{t,r}\|_2^2.$$

##### Model Normalization:

$$\mathbf{s}_{i,j}^{t,r} = \frac{\boldsymbol{\theta}_{i,j}^{t,r} - \bar{\boldsymbol{\theta}}_i^{t,r}}{\pi_{i,j}^{t,r}}.$$

Transmit normalized local model  $\mathbf{s}_{i,j}^{t,r}$  to the edge server through AirComp in (11).

---

of  $\boldsymbol{\theta}_{i,j}$  and  $\mathbf{w}_i$  are bounded, it is fair to assume that  $\pi_{i,j}^2$  and  $\pi_i^2$  have non-negative upper bounds in Assumption 4.

#### B. Convergence Analysis

*Theorem 1:* With Assumptions 1, 2 and 3, if  $\tilde{\eta} \leq \min\{\frac{\beta}{8L_F}, \frac{1}{384L_F\lambda^2}\}$  and  $\lambda \geq \sqrt{16(L^2 + 1)}$ , then the upper bound of the time-averaged norm of the gradients over  $T$  rounds is

$$\begin{aligned}
&\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla F(\mathbf{w}^t)\|_2^2] \\
&\leq 2 \left( \frac{\Delta_F}{\tilde{\eta} T} + A_1 + \tilde{\eta} A_2 \right) + \frac{1}{T} \sum_{t=0}^{T-1} 2(1 + 2L_F \tilde{\eta}) \mathbb{E} \|\mathbf{e}^t\|^2, \quad (28)
\end{aligned}$$

where  $\Delta_F = \mathbb{E}[F(\mathbf{w}^0) - F(\mathbf{w}^*)]$ ,  $A_1 = 8\beta\bar{\lambda}^2\delta^2 + 16\sigma_F^2$ ,  $A_2 = \frac{128L_F(3R\sigma_F^2 + \bar{\lambda}^2\delta^2)}{R}$ ,  $\delta^2 = \frac{2}{(\lambda_1 - L)^2} (\frac{\gamma_i^2}{|\mathcal{D}|} + \nu)$ , and  $\lambda = \frac{\lambda_1 \lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}}$ . According to  $\mathbf{e}^t$  calculated in Section II, we

have

$$\begin{aligned}
\mathbb{E}\|e^t\|_2^2 &= \mathbb{E}\left\|\frac{1}{NR}\frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2}\sum_{i=1}^N\sum_{r=0}^{R-1}\frac{1}{M}\pi_i^{t,r}\right. \\
&\quad \left[\sum_{j=1}^M\left(1-\frac{1}{\sqrt{\eta_i}}\mathbf{m}_i^H\mathbf{h}_{i,j}^{t,r}w_{i,j}^{t,r}\right)\mathbf{s}_{i,j}^{t,r}-\frac{1}{\sqrt{\eta_i}}\mathbf{m}_i^H\mathbf{n}_i^{t,r}\right. \\
&\quad \left.-\frac{1}{\sqrt{\eta_i}}\mathbf{m}_i^H\sum_{\ell\in\mathcal{N}\setminus\{i\}}\sum_{j'=1}^M\mathbf{h}_{i,j'}^{t,R}w_{\ell,j'}^{t,r}\mathbf{s}_{\ell,j'}^{t,r}\right] \\
&\quad \left.+\frac{\beta\pi^{t,R}}{\tilde{\eta}N}\left[\sum_{i=1}^N\left(1-\frac{1}{\sqrt{\zeta}}\mathbf{v}^H\mathbf{h}_i^{t,R}w_i^{t,R}\right)\mathbf{s}_i^{t,R}+\frac{1}{\sqrt{\zeta}}\mathbf{v}^H\mathbf{n}^{t,R}\right]\right\|_2^2 \\
&\leq \frac{2(MN+1)\Gamma D}{NRM^2}\left(\frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2}\right)^2\sum_{i=1}^N\sum_{r=0}^{R-1} \\
&\quad \left[\sum_{j=1}^M\left(1-\frac{1}{\sqrt{\eta_i}}\mathbf{m}_i^H\mathbf{h}_{i,j}^{t,r}w_{i,j}^{t,r}\right)^2+\left(\frac{\mathbf{m}_i^H\mathbf{n}_i^{t,r}}{\sqrt{\eta_i}}\right)^2\right. \\
&\quad \left.+\sum_{\ell\in\mathcal{N}\setminus\{i\}}\sum_{j'=1}^M\left(\frac{1}{\sqrt{\eta_i}}\mathbf{m}_i^H\mathbf{h}_{i,j'}^{t,R}w_{\ell,j'}^{t,r}\right)^2\right]+\frac{2\beta^2\Gamma D(N+1)}{\tilde{\eta}^2N^2} \\
&\quad \left[\sum_{i=1}^N\left(1-\frac{\mathbf{v}^H\mathbf{h}_i^{t,R}w_i^{t,R}}{\sqrt{\zeta}}\right)^2+\left(\frac{\mathbf{v}^H\mathbf{n}^{t,R}}{\sqrt{\zeta}}\right)^2\right], \quad (30)
\end{aligned}$$

where the inequality is derived from Jensen inequality and Assumption 4.

*Proof:* Please refer to Appendix D. ■

*Remark 2:* The time-averaged norm of the gradients is utilized as the convergence metric. The first three terms of (28) are introduced by the initial optimality gap, the variance of stochastic gradient, the model dissimilarity, as well as the edge server drift-error (i.e., a discrepancy that occurs between the model parameters on edge servers and cloud server during the training process).  $A_1$  and  $A_2$  depend on edge communication round  $R$ , penalty coefficients  $\lambda_1$  and  $\lambda_2$ , and global convergence coefficient  $\beta$ .  $\tilde{\eta}$  is positively correlated with  $\eta_1$ . The first term decreases as  $\tilde{\eta}$  increases, suggesting that larger values of  $\tilde{\eta}$  decrease the contribution of the initial error  $\Delta F$  over the total number of iterations  $T$ . Meanwhile, the third and the last terms increase linearly with  $\tilde{\eta}$ . Thus, if  $\tilde{\eta}$  is too large, these terms become dominant, slowing down the convergence rate. In summary,  $\tilde{\eta}$  affects the convergence bound by controlling the balance between how quickly the initial error decreases and how much noise or approximation error affects the process. The last term represents the time-averaged MSE. As  $T$  approaches infinity, the initial optimality gap decreases to 0. With fixed  $R$ ,  $T$ ,  $\beta$ ,  $\eta_1$ ,  $\lambda_1$  and  $\lambda_2$ , terms in parentheses in (28) are constants. Hence, the primary impediment to the convergence lies in the time-averaged MSE specified in (29), which is required to be minimized since it directly influences the convergence rate by controlling the gradient norm decay over time. As the time-averaged MSE scales

inversely with  $T$ , ensuring tighter bounds on  $\mathbb{E}\|e^t\|_2^2$  for each communication round helps maintain faster convergence rates, further reducing the gradient variance and error propagation across rounds, and eventually enhance learning performance.

### C. Problem Formulation

When receive beamforming vectors  $\mathbf{m}_i$  and  $\mathbf{v}$  are given, we express the optimal transmit scalars  $w_{i,j}$  and  $w_i$  of each device and each edge server, respectively, as follows

$$w_{i,j} = \sqrt{\eta_i} \frac{(\mathbf{m}_i^H \mathbf{h}_{i,j})^H}{\|\mathbf{m}_i^H \mathbf{h}_{i,j}\|_2^2}, \quad \forall j, \quad w_i = \sqrt{\zeta} \frac{(\mathbf{v}^H \mathbf{h}_i)^H}{\|\mathbf{v}^H \mathbf{h}_i\|_2^2}, \quad \forall i. \quad (31)$$

According to the transmit power constraint,  $\eta_i$  and  $\zeta$  are

$$\eta_i = P \min_{j \in \mathcal{M}} \|\mathbf{m}_i^H \mathbf{h}_{i,j}\|_2^2, \quad \zeta = P \min_{i \in \mathcal{N}} \|\mathbf{v}^H \mathbf{h}_i\|_2^2. \quad (32)$$

With (31) and (32), by omitting constant terms in (29), we express the MSE in  $(t, r)$  as

$$\begin{aligned}
\text{MSE}^{t,r} &= \sum_{i=1}^N \sum_{\ell \in \mathcal{N} \setminus \{i\}} \sum_{j'=1}^M \underbrace{\left[ \frac{\eta_i \|\mathbf{m}_i^H \mathbf{h}_{i,j'}\|^2}{\eta_i \|\mathbf{m}_i^H \mathbf{h}_{\ell,j'}\|^2} + \frac{\|\mathbf{m}_i\|^2 \sigma^2}{\eta_i} \right]}_{(a) \text{ MSE}_{\text{edge}}^{t,r}} \\
&\quad + \underbrace{\frac{\sigma_0^2 \|\mathbf{v}\|^2}{P \min_i \|\mathbf{v}^H \mathbf{h}_i\|^2}}_{(b) \text{ MSE}_{\text{cloud}}^t}. \quad (33)
\end{aligned}$$

We define term (a) in (33) as  $\text{MSE}_{\text{edge}}^{t,r}$ , which is the error introduced by edge model aggregation, and term (b) as  $\text{MSE}_{\text{cloud}}^t$ , which is the error introduced by global model aggregation.  $\text{MSE}_{\text{cloud}}^t$  only exists when the edge iterations are completed, i.e.,  $r = R$ . To minimize the  $\text{MSE}^{t,r}$ , we formulate problem  $\mathcal{P}_0$  to jointly optimize the edge server receive beamforming vector  $\mathbf{m}_i$  and the cloud server receive beamforming vector  $\mathbf{v}$  as follows:

$$\mathcal{P}_0 : \min_{\mathbf{v}, \mathbf{m}_i} \text{MSE}_{\text{edge}}^{t,r} + \text{MSE}_{\text{cloud}}^t. \quad (34)$$

From (33), we observe that  $\text{MSE}_{\text{cloud}}^t$  is affected by cloud receive beamforming vector  $\mathbf{v}$  and channel states  $\mathbf{h}_i$ ,  $\forall i$ , while  $\text{MSE}_{\text{edge}}^{t,r}$  is related with edge receive beamforming vectors  $\mathbf{m}_i$ ,  $\forall i$ , and channel states between edge servers and devices  $\mathbf{h}_{i,j}$ ,  $\forall i, j$ . Hence, we decompose problem  $\mathcal{P}_0$  into two sub-problems

$$\mathcal{P}_1 : \min_{\mathbf{v}} \frac{\sigma_0^2 \|\mathbf{v}\|^2}{P \min_i \|\mathbf{v}^H \mathbf{h}_i\|^2}, \quad (35)$$

$$\mathcal{P}_2 : \min_{\mathbf{m}_i} \sum_{i=1}^N \sum_{\ell \in \mathcal{N} \setminus \{i\}} \sum_{j'=1}^M \left[ \frac{\eta_i \|\mathbf{m}_i^H \mathbf{h}_{i,j'}\|^2}{\eta_i \|\mathbf{m}_i^H \mathbf{h}_{\ell,j'}\|^2} + \frac{\|\mathbf{m}_i\|^2 \sigma^2}{\eta_i} \right]. \quad (36)$$

The following two sections present two algorithms for solving  $\mathcal{P}_1$  and  $\mathcal{P}_2$ , respectively.

#### IV. GLOBAL MODEL AGGREGATION OPTIMIZATION

In this section, an efficient algorithm is developed to optimize the receive beamforming design at the cloud server by utilizing Lagrangian duality and successive convex approximation.

##### A. Problem Transformation

According to Proposition 1 in [42],  $\mathcal{P}_1$  can be equivalently transformed into  $\mathcal{P}_3$  as follows

$$\mathcal{P}_3 : \min_{\mathbf{v}} \|\mathbf{v}\|_2^2 \quad (37a)$$

$$\text{s.t. } \|\mathbf{v}^H \mathbf{h}_i\|_2^2 \geq 1, \forall i. \quad (37b)$$

By denoting  $\mathbf{z} \in \mathbb{C}^K$ , an auxiliary variable, and considering a positive semidefinite matrix  $\mathbf{A} \succeq \mathbf{0}$ , we have  $(\mathbf{v} - \mathbf{z})^H \mathbf{A} (\mathbf{v} - \mathbf{z}) \geq 0, \forall \mathbf{z}$ . This leads to  $\mathbf{v}^H \mathbf{A} \mathbf{v} \geq 2\Re\{\mathbf{v}^H \mathbf{A} \mathbf{z}\} - \mathbf{z}^H \mathbf{A} \mathbf{z}$ . Given a specific  $\mathbf{z}$ , applying the above inequality to constraint (37b) yields the optimization problem  $\mathcal{P}_{SCA}$ :

$$\begin{aligned} \mathcal{P}_{SCA}(\mathbf{z}) : \\ \min_{\mathbf{v}} \|\mathbf{v}\|_2^2 \\ \text{s.t. } -2\Re(\mathbf{v}^H \mathbf{h}_i \mathbf{h}_i^H \mathbf{z}) + \|\mathbf{z}^H \mathbf{h}_i\|_2^2 + 1 \leq 0, i \in \mathcal{N}, \end{aligned} \quad (38)$$

which is a convex approximation of (37). Since the constraint in (38) is convex, we derive its optimal solution from the Lagrangian dual domain:

$$\begin{aligned} \mathcal{L}(\mathbf{z}, \mathbf{v}, \mathbf{\Lambda}) = \|\mathbf{v}\|_2^2 \\ + \sum_{i=1}^N \Lambda_i [-2\Re\{\mathbf{v}^H \mathbf{h}_i \mathbf{h}_i^H \mathbf{z}\} + \|\mathbf{z}^H \mathbf{h}_i\|_2^2 + 1], \end{aligned} \quad (39)$$

where  $\mathbf{\Lambda} = [\Lambda_1, \Lambda_2, \dots, \Lambda_N]^T$  is the Lagrangian multiplier. Hence, we can express the Lagrangian dual problem of  $\mathcal{P}_{SCA}(\mathbf{z})$  as

$$\mathcal{D}_{SCA}(\mathbf{z}) : \max_{\mathbf{\Lambda}} g(\mathbf{z}, \mathbf{\Lambda}) \text{ s.t. } \mathbf{\Lambda} \succeq \mathbf{0}, \quad (40)$$

where

$$g(\mathbf{\Lambda}, \mathbf{z}) = \min_{\mathbf{v}} \mathcal{L}(\mathbf{z}, \mathbf{v}, \mathbf{\Lambda}). \quad (41)$$

By reorganizing the terms in (39), the Lagrangian function is written as

$$\begin{aligned} \mathcal{L}(\mathbf{z}, \mathbf{v}, \mathbf{\Lambda}) = \sum_{i=1}^N \Lambda_i (\|\mathbf{z}^H \mathbf{h}_i\|_2^2 + 1) + \|\mathbf{v}\|_2^2 \\ - 2\Re\left\{\mathbf{z}^H \left(\sum_{i=1}^N \Lambda_i \mathbf{h}_i \mathbf{h}_i^H\right) \mathbf{v}\right\}. \end{aligned} \quad (42)$$

By defining  $\boldsymbol{\nu} \triangleq (\sum_{i=1}^N \Lambda_i \mathbf{h}_i \mathbf{h}_i^H) \mathbf{z}$ , problem (41) can be written as

$$\min_{\mathbf{v}} \|\mathbf{v}\|_2^2 - 2\Re\{\boldsymbol{\nu}^H \mathbf{v}\}. \quad (43)$$

Since the above optimization problem is convex, we derive the optimal solution using KKT conditions. By defining  $\mathbf{H} =$

##### Algorithm 2: Calculating $\mathbf{\Lambda}$ .

**Require:** Channel gain vector  $\mathbf{h}_i, i \in \mathcal{N}$ .

- 1: **Initialization**  $\mathbf{\Lambda}^0$  and  $\iota = 1$ .
- 2: **repeat**
- 3:   **for each**  $i \in \mathcal{N}$  **do**
- 4:      $\Lambda_i^\iota \leftarrow \frac{1}{2\mathbf{h}_i^H \mathbf{G}^{-1}(\mathbf{\Lambda}^{\iota-1}) \mathbf{h}_i}$ ;
- 5:   **end for**
- 6:    $\iota \leftarrow \iota + 1$ ;
- 7: **until**  $\iota = 100$ .

$[\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_N]^T$  and

$$\mathbf{G}(\mathbf{\Lambda}) = \mathbf{I}_N + \sum_{i=1}^N \Lambda_i \mathbf{h}_i \mathbf{h}_i^H, \quad (44)$$

the optimal receive beamforming vector is calculated as

$$\mathbf{v}^{opt} = \mathbf{G}^{-1}(\mathbf{\Lambda}^{opt}) \sum_{i=1}^N a_i^{opt} \mathbf{h}_i = \mathbf{G}^{-1}(\mathbf{\Lambda}^{opt}) \mathbf{H} \mathbf{a}^{opt}. \quad (45)$$

*Proof:* Please refer to Theorem 1 in [43] with  $G = 1$  and  $\mathbf{R}_i^{-1}(\mathbf{\Lambda}) = \mathbf{0}$ . ■

*Remark 3:*  $\mathbf{\Lambda}^{opt}$  is denoted as the optimal dual solution for  $\mathcal{D}_{SCA}(\mathbf{v}^{opt})$ . By denoting  $\delta_i = \mathbf{h}_i^H \mathbf{v}^{opt}$ ,  $\forall i$ , we have  $a_i^{opt} = 2\Lambda_i^{opt} \delta_i$ , and  $\mathbf{a}^{opt} = [a_1^{opt}, a_2^{opt}, \dots, a_N^{opt}]^T$ . The optimal solution is expressed in a semi-closed form where  $\mathbf{\Lambda}^{opt}$  and  $\mathbf{a}^{opt}$  are computed numerically. Computing  $\mathbf{\Lambda}^{opt}$  and  $\mathbf{a}^{opt}$  poses a challenge due to the NP-hard nature of the original problem (37). The numerical algorithm is outlined in the subsequent parts.

##### B. Calculating $\mathbf{\Lambda}$

We define  $\mathbf{L}_{\mathbf{\Lambda}} = \text{diag}(\mathbf{\Lambda})$ ,  $\mathbf{\Delta} = [\delta_1, \delta_2, \dots, \delta_N]^T$ , and  $\mathbf{a} = 2\mathbf{L}_{\mathbf{\Lambda}} \mathbf{\Delta}$ . According to (45) we have

$$\delta_i = \mathbf{h}_i^H \mathbf{v}^{opt} = 2\mathbf{h}_i^H \mathbf{G}^{-1}(\mathbf{\Lambda}) \mathbf{H} \mathbf{L}_{\mathbf{\Lambda}} \mathbf{\Delta}. \quad (46)$$

For  $i \in \mathcal{N}$ , it follows that

$$\mathbf{\Delta} = 2\mathbf{H}^H \mathbf{G}^{-1}(\mathbf{\Lambda}) \mathbf{H} \mathbf{L}_{\mathbf{\Lambda}} \mathbf{\Delta}, \quad (47)$$

which can be rewritten as

$$2(\mathbf{H}^H \mathbf{G}^{-1}(\mathbf{\Lambda}) \mathbf{H} \mathbf{L}_{\mathbf{\Lambda}} - \mathbf{I}) \mathbf{\Delta} = \mathbf{0}. \quad (48)$$

Since  $\mathbf{\Delta}$  is unknown, it is challenging to solve (48). Hence we adopt a suboptimal algorithm to calculate  $\mathbf{\Lambda}$ . A condition sufficient to meet (48) is

$$2\mathbf{H}^H \mathbf{G}^{-1}(\mathbf{\Lambda}) \mathbf{H} \mathbf{L}_{\mathbf{\Lambda}} = \mathbf{I}, \quad (49)$$

which equals

$$\begin{cases} 2\Lambda_i \mathbf{h}_i^H \mathbf{G}^{-1}(\mathbf{\Lambda}) \mathbf{h}_i = 1, & i \in \mathcal{N}, \\ 2\Lambda_i \mathbf{h}_i^H \mathbf{G}^{-1}(\mathbf{\Lambda}) \mathbf{h}_{i'} = 0, & i' \neq i, i' \in \mathcal{N}. \end{cases} \quad (50)$$

The solution to (50) is obtained through Algorithm 2.

##### C. Calculating $\mathbf{a}$

According to (45) and  $\mathbf{\Lambda}$ , problem (37) is rewritten as:

$$\mathcal{P}_4 : \min_{\mathbf{a}} \|\mathbf{G}^{-1}(\mathbf{\Lambda}) \mathbf{H} \mathbf{a}\|_2^2 \quad (51a)$$



**Algorithm 3:** SCA-Based Algorithm for Problem (37).**Require:**  $\mathbf{h}_i, i \in \mathcal{N}$ .

- 1: **Initialization**  $\mathbf{u}^0, \iota = 0$ , and stopping condition  $\varepsilon_1$ .
- 2: **repeat**
- 3:   Calculate  $\Lambda$  through Algorithm 2;
- 4:   Derive the optimal solution  $\mathbf{a}^{\text{opt}}(\mathbf{u}^\iota)$  through addressing (52);
- 5:    $\mathbf{u}^{\iota+1} \leftarrow \mathbf{a}^{\text{opt}}(\mathbf{u}^\iota)$ ;
- 6:    $\iota \leftarrow \iota + 1$ ;
- 7: **until**  $\|\mathbf{u}^\iota - \mathbf{u}^{\iota-1}\| \leq \varepsilon_1$ ;
- 8:  $\mathbf{v}^{\text{opt}} \leftarrow \mathbf{G}^{-1}(\Lambda)\mathbf{H}\mathbf{a}^\iota$ .

**Output:** Receive beamforming vector  $\mathbf{v}^{\text{opt}}$ .

$$\text{s.t. } |\mathbf{a}^H \mathbf{H}^H \mathbf{G}^{-1}(\Lambda) \mathbf{h}_i|^2 \geq 1, \forall i \in \mathcal{N}. \quad (51b)$$

For problem  $\mathcal{P}_3$ , variable  $\mathbf{v}$  is of size  $K$ , which is the number of receiving antennas at the cloud server. In contrast, variable  $\mathbf{a}$  in problem  $\mathcal{P}_4$  is of size  $N$ , where  $N$  is the number of the edge servers, which is much smaller than  $K$ . This significantly reduces the computation complexity. In the following, we apply the SCA method to compute  $\mathbf{a}$  for  $\mathcal{P}_4$ . By defining  $\mathbf{R} \triangleq \mathbf{G}^{-1}(\Lambda)\mathbf{H}$  and  $\mathbf{f}_i \triangleq \mathbf{R}^H \mathbf{h}_i, i \in \mathcal{N}$ , similar to the problem (38), we use a  $N \times 1$  auxiliary variable  $\mathbf{u}$  and employ the convex approximation on constraint (51) in  $\mathcal{P}_4$  as outlined below:

$$\begin{aligned} \mathcal{P}_{SCA1}(\mathbf{u}) : \\ \min_{\mathbf{a}} \|\mathbf{R}\mathbf{a}\|^2 \\ \text{s.t. } -4\Re\{\mathbf{a}^H \mathbf{f}_i \mathbf{f}_i^H \mathbf{u}\} + 2|\mathbf{u}^H \mathbf{f}_i|^2 + 1 \leq 0, i \in \mathcal{N}. \end{aligned} \quad (52a)$$

$$(52b)$$

To acquire  $\mathbf{a}$  for  $\mathcal{P}_4$ , we iteratively address  $\mathcal{P}_{SCA1}(\mathbf{u})$ , updating  $\mathbf{u}$  with the optimal solution  $\mathbf{a}^*(\mathbf{u})$  for  $\mathcal{P}_{SCA1}$  till reaching convergence.

**D. Computation Complexity**

In the  $t$ -th global communication round, (45) and Algorithm 2 are firstly leveraged to transform problem  $\mathcal{P}_3$  into problem  $\mathcal{P}_4$ . Furthermore, a semidefinite relaxation algorithm is adopted to calculate an initializing beamforming vector  $\mathbf{u}^0$  with the complexity of ( $\mathcal{O}(N^6)$ ). Then, the computation complexity of solving problem  $\mathcal{P}_4$  through Algorithm 3 is  $\mathcal{O}(I(N^3))$ , where  $I(\cdot)$  is related to the convergence iteration rounds. However, if we directly solve problem  $\mathcal{P}_3$  iteratively through an SCA-based algorithm, the computation cost is  $\mathcal{O}(I(K^3))$ . As we declared before, the number of receiving antennas  $K$  is much larger than the number of edge servers  $N$  in most practical cases. Hence combing Algorithms 2 and 3 is an efficient way to solve problem  $\mathcal{P}_3$ .

**V. EDGE MODEL AGGREGATION OPTIMIZATION**

In this section, we develop an interference-aware algorithm to cooperatively optimize the transceiver design of each edge server.

**A. Problem Transformation**

To simplify problem  $\mathcal{P}_2$ , we introduce auxiliary variable  $\omega_i = \min_j \|\mathbf{m}_i^H \mathbf{h}_{i,j}\|^2$  for edge server  $i$ . By letting  $\mathbf{x}_i = \frac{\mathbf{m}_i}{\sqrt{\omega_i}}$ , problem  $\mathcal{P}_2$  can be transformed as follows

$$\mathcal{P}_5 : \min_{\mathbf{x}_i} \sum_{i=1}^N \sum_{l \in \mathcal{N} \setminus \{i\}} \sum_{j'=1}^M \frac{\|\mathbf{x}_i^H \mathbf{h}_{i,j'}\|^2}{\|\mathbf{x}_l^H \mathbf{h}_{l,j'}\|^2} + \sum_{i=1}^N \frac{\sigma^2 \|\mathbf{x}_i\|^2}{P} \quad (53a)$$

$$\text{s.t. } \|\mathbf{x}_i^H \mathbf{h}_{i,j}\|^2 \geq 1, \forall i, j. \quad (53b)$$

Compared with the single-cluster beamforming design in (37), problem  $\mathcal{P}_5$  is more complex because of the non-convexity of both (53a) and (53b) introduced by the interference among different clusters. Therefore, by decomposing  $\mathcal{P}_5$  into  $N$  single-cluster beamforming design subproblems, we leverage the SCA-based algorithm to solve each subproblem.

**B. Receive Beamforming Optimization**

The  $i$ -th receive beamforming optimization problem can be written as

$$\min_{\mathbf{x}_i} \sum_{l \in \mathcal{N} \setminus \{i\}} \sum_{j'=1}^M \frac{\|\mathbf{x}_i^H \mathbf{h}_{i,j'}\|^2}{\|\mathbf{x}_l^H \mathbf{h}_{l,j'}\|^2} + \frac{\sigma^2 \|\mathbf{x}_i\|^2}{P} \quad (54a)$$

$$\text{s.t. } \|\mathbf{x}_i^H \mathbf{h}_{i,j}\|^2 \geq 1, \forall i, j. \quad (54b)$$

We introduce an auxiliary variable  $e_i$  to further simplify the problem as follows

$$\min_{\mathbf{x}_i} \sum_{l \in \mathcal{N} \setminus \{i\}} \sum_{j'=1}^M e_i + \frac{\sigma^2 \|\mathbf{x}_i\|^2}{P} \quad (55a)$$

$$\text{s.t. } \|\mathbf{x}_i^H \mathbf{h}_{i,j}\|^2 \geq 1, \forall i, j, \quad (55b)$$

$$\frac{\|\mathbf{x}_i^H \mathbf{h}_{i,j'}\|^2}{\|\mathbf{x}_l^H \mathbf{h}_{l,j'}\|^2} \leq e_i, \forall i, j'. \quad (55c)$$

Due to the non-convexity of constraint (55b) and the fractional structure of constraint (55c), the optimal solution is challenging, if not impossible, to be obtained. However, constraints (55b) and (55c) are still non convex. To address the problem, the SCA method is applied to transform the quadratic terms to linear constraints by denoting  $\mathbf{b}_{i,j} = [\Re(\mathbf{x}_i^H \mathbf{h}_{i,j}), \Im(\mathbf{x}_i^H \mathbf{h}_{i,j})]$  and  $\mathbf{b}_{i,j'} = [\Re(\mathbf{x}_i^H \mathbf{h}_{i,j'}), \Im(\mathbf{x}_i^H \mathbf{h}_{i,j'})]$ , the transformed linear constraints can be expressed as:

$$-\|\mathbf{b}_{i,j}^{(\tau)}\|^2 + 2(\mathbf{b}_{i,j}^{(\tau)})^T (\mathbf{b}_{i,j}^{(\tau)} - \mathbf{b}_{i,j}) + 1 \leq 0,$$

$$\|\mathbf{b}_{i,j'}^{(\tau)}\|^2 + 2(\mathbf{b}_{i,j'}^{(\tau)})^T (\mathbf{b}_{i,j'} - \mathbf{b}_{i,j'}^{(\tau)}) - e_i \|\mathbf{x}_l^H \mathbf{h}_{l,j'}\|^2 \leq 0, \quad (56)$$

where  $\mathbf{b}_{i,j}^{(\tau)}$  and  $\mathbf{b}_{i,j'}^{(\tau)}$  are solutions in the  $\tau$ -th iteration of the problem. At the beginning of the iteration, the initial solutions  $\mathbf{b}_{i,j}^{(0)}$  and  $\mathbf{b}_{i,j'}^{(0)}$  can be obtained through the semidefinite relaxation. By substituting (56) into (55a), and denoting  $E_i = e_i \|\mathbf{x}_l^H \mathbf{h}_{l,j'}\|^2$ , we have the following optimization problem

$$\mathcal{P}_6 : \min_{\mathbf{x}_i} \sum_{l \in \mathcal{N} \setminus \{i\}} \sum_{j'=1}^M e_i + \frac{\sigma^2 \|\mathbf{x}_i\|^2}{P} \quad (57a)$$

---

**Algorithm 4:** Multi-Cluster Edge Server Receive Beamforming Optimization for Problem (53).

---

**Input:** The number of edge servers  $N$ , the average power of the noise  $\sigma^2$  and the transmit power  $P$ .

- 1: **Initialize** the auxiliary beamforming vector  $\mathbf{x}_i^{(0)}$  for each cluster; Set  $\tau = 0$  and convergence tolerance.
- 2: **for**  $i \leftarrow 1$  to  $N$  **do**
- 3:   Fixing the other clusters' beamforming vector  $\mathbf{x}_{i'}, i' \neq i, i' \in \mathcal{N}$ . Introduce the auxiliary vector  $\mathbf{e}_i, \mathbf{b}_{i,j}^{(0)}, \mathbf{b}_{i,j'}^{(0)}, j, j' \in \mathcal{M}$ .
- 4:   **repeat**
- 5:     Solve problem (57a) and obtain the solution  $\mathbf{x}_i^*(\mathbf{e}_i, \mathbf{b}_{i,j}^{(\tau)}, \mathbf{b}_{i,j'}^{(\tau)})$ ;
- 6:      $\mathbf{x}_i^{(\tau+1)} \leftarrow \mathbf{x}_i^*(\mathbf{e}_i, \mathbf{b}_{i,j}^{(\tau)}, \mathbf{b}_{i,j'}^{(\tau)})$ ;
- 7:      $\tau \leftarrow \tau + 1$ ;
- 8:   **until**  $\|\mathbf{x}_i^\tau - \mathbf{x}_i^{\tau-1}\|^2 \leq \varepsilon_2$ .
- 9: **end for**

**Output:** The optimal auxiliary receive beamforming vector  $\mathbf{x}_i, i \in \mathcal{N}$ .

---

$$\text{s.t. } -\|\mathbf{b}_{i,j}^{(\tau)}\|^2 + 2(\mathbf{b}_{i,j}^{(\tau)})^T(\mathbf{b}_{i,j}^{(\tau)} - \mathbf{b}_{i,j}) + 1 \leq 0, \quad (57b)$$

$$\|\mathbf{b}_{i,j'}^{(\tau)}\|^2 + 2(\mathbf{b}_{i,j'}^{(\tau)})^T(\mathbf{b}_{i,j} - \mathbf{b}_{i,j'}) - E_i \leq 0, \quad (57c)$$

$$\mathbf{b}_{i,j} = [\Re(\mathbf{x}_i^H \mathbf{h}_{i,j}), \Im(\mathbf{x}_i^H \mathbf{h}_{i,j})], \quad (57d)$$

$$\mathbf{b}_{i,j'} = [\Re(\mathbf{x}_i^H \mathbf{h}_{i,j'}), \Im(\mathbf{x}_i^H \mathbf{h}_{i,j'})]. \quad (57e)$$

As for the above optimization problem, the objective function and constraints are all convex for each variable. Hence, CVX tools [44] can be utilized to attain the optimal auxiliary variable  $\mathbf{x}_i$  to complete the transceiver design of the  $i$ -th edge server. Then we solve  $N$  sub-problems in turn and finish the edge model aggregation optimization. The steps of the interference-aware edge server receive beamforming optimization are shown in Algorithm 4.

### C. Computation Complexity

In communication round  $(t, r)$ , Algorithm 4 is leveraged to design the receive beamforming of each edge server. As discussed before, we initially decouple problem (54) into  $N$  sub-problems, hence we analyze the computation complexity in each cluster. As mentioned in [45], initializing the auxiliary beamforming vector  $\mathbf{x}_i^{(0)}$  through semidefinite programming will cost the complexity of  $\mathcal{O}((M + K^2)^{3.5})$  in the worst case. And then solving  $\mathcal{P}_6$  through SCA-based algorithm has a complexity of  $\mathcal{O}(I((N - 1)MK^3))$ , where  $I(\cdot)$  is related to the convergence iteration rounds.

## VI. SIMULATION RESULTS

### A. Experimental Settings

A three-dimensional scene is considered, in which a cloud server is spotted at  $(0, 0, 145)$  meters, and two edge servers are located at  $(20, 0, 20)$  meters and  $(0, 20, 20)$  meters, respectively.

Each edge server is associated with 10 devices (i.e.,  $M = 10$ ), scattered evenly within circles centered at  $(20, 0, 0)$  meters and  $(0, 20, 0)$  meters with a radius of 10 meters. The large-scale fading is represented as  $T_0(\frac{d_l}{d_0})^{-\alpha}$  [46], in which  $d_l$  is the link distance,  $\alpha$  is the path loss exponent, and  $T_0$  is the path loss when  $d_0 = 1$  meter. The small-scale fading is modeled as Rician fading with rician factor  $\kappa$ . Unless specified otherwise, we set  $\alpha = 3$ ,  $T_0 = -30$  dB,  $\kappa = 3$ ,  $P = 30$  dBm, and  $\sigma^2 = -100$  dBm. The SCA convergence accuracy factors  $\varepsilon_1$  and  $\varepsilon_2$  of the global and edge model aggregation algorithms are both set to  $10^{-4}$ .

1) *Learning Model, and Parameters:* A deep neural network with 2 fully connected layers is used. We consider both the MNIST and FMNIST datasets, where each device is designed to encompass only 2 types of labels to reflect data heterogeneity. Besides, we set  $\eta_1 = 0.05$ ,  $\beta = 1.0$ ,  $\lambda_1 = \lambda_2 = 15$ ,  $R = 10$ , and  $T = 200$ . The batch size is 20.

2) *Baseline Schemes:* In the case of evaluating the performance of distinct cloud server optimization algorithms, the SDR [45] algorithm is used as the baseline. When testing the performance of multi-cluster edge server optimization, the following schemes are considered for comparison.

- **Error-free:** This baseline achieves the best performance, where the uplink transmissions from devices to edge servers and from edge servers to the cloud server are error-free.
- **IgnInter:** In this scheme, each cluster independently optimizes its receive beamforming vector without considering the inter-cluster interference. Therefore, each edge server solves problem (35) through Algorithm 3.

In addition, the proposed AirComp-assisted HPFL is compared to the following baseline FL algorithms, i.e., FedAvg [47], FedProx [33], pFedMe [31], and HierFedAvg [6]. For a fair comparison of one-layer FL and HFL, the device number is fixed to 20 in all one-layer FL frameworks (i.e., FedAvg, pFedMe, and FedProx), and the rest hyperparameter settings remain unchanged.

### B. Performance Under Different Algorithms

Fig. 2 depicts the test accuracy of both global and personalized models within the AirComp-assisted HPFL framework under various cloud server optimization algorithms. Both cloud and edge servers are equipped with 20 receive antennas. Our developed SCA-based algorithm exhibits significant performance enhancements over the SDR algorithm and has an approximately negligible gap compared to the optimal BnB algorithm. This shows the capability of the developed SCA-based approach to maintain near-optimal performance while substantially reducing computational costs in comparison to the optimal BnB algorithm.

In Fig. 3, the test accuracy of both global and personalized models under different edge model aggregation schemes are compared. Our proposed algorithm performs similarly to the error-free case in both global and personalized model training. Therefore, our proposed scheme can effectively balance the intra-cluster noise and inter-cluster interference during uplink transmission, and achieve a near-optimal convergence

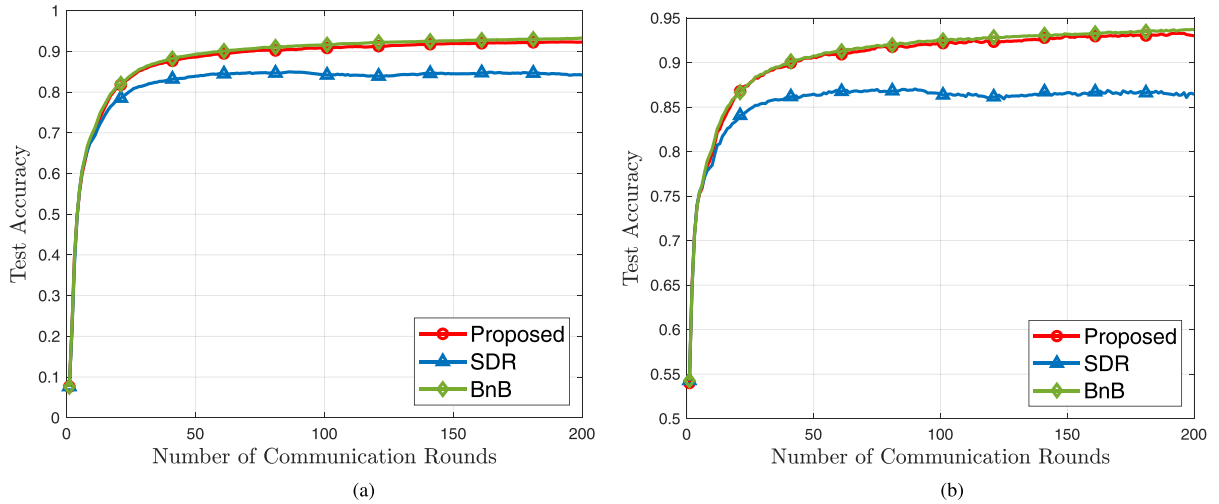


Fig. 2. Test accuracy versus number of communication rounds for different transceiver design algorithms at the cloud side. (a) Global model. (b) Personalized model.

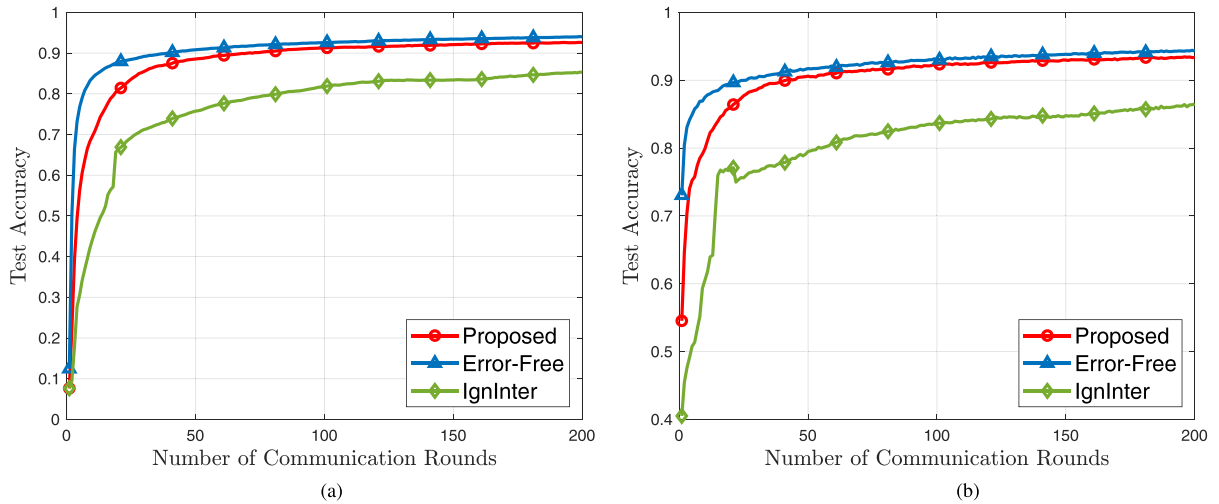


Fig. 3. Test accuracy versus number of communication rounds for different transceiver design schemes at the edge side. (a) Global model. (b) Personalized model.

performance. Since the IgnInter scheme does not account for the inter-cluster interference and channel conditions change independently over different clusters and different global and edge communication rounds, the edge servers will not be able to design the receive beamforming vector according to the signals from other clusters, which leads to severe distortion when aggregating the signals, leading to poor learning performance.

In Fig. 4, we compare the AirComp-assisted HPFL framework with four other baselines: FedAvg, HierFedAvg, pFedMe, and FedProx. Among all frameworks, we utilize the proposed algorithms for receive beamforming design at the cloud server and edge servers (in the hierarchical structure), with the number of receive antennas set to  $K = 20$ . The result shows that the global model of our proposed framework achieves a higher test accuracy compared to FedAvg, FedProx, and HierFedAvg. This improvement is attributed to the hierarchical structure and the setup of an  $L_2$ -norm regularized loss function. Furthermore, while personalized models in pFedMe demonstrate convergence

to a high level of accuracy, closely approaching that of the proposed AirComp-assisted HPFL, a substantial disparity persists in the accuracy rate of its global model when compared to our framework. This discrepancy arises from the impact of the  $L_2$ -norm regularization term, which promotes edge servers and devices to train their personalized models without deviating excessively from the global model. More specifically, the drop of the test accuracy achieved by pFedMe is due to its personalization versus generalization trade-off under a heterogeneous dataset environment.

In Fig. 5(a), (b), and (c), we compare the learning performance of the proposed AirComp-assisted HPFL under different hyperparameters. i.e., penalty coefficient  $\lambda$ , global update coefficient  $\beta$ , and the number of edge epochs  $R$ . Penalty coefficient  $\lambda$  controls the distance between the personalized models and the global model during the training process. We observe from Fig. 5(a) that, with a larger  $\lambda$ , the device can overcome the heterogeneity of its local data to train a more generalized local

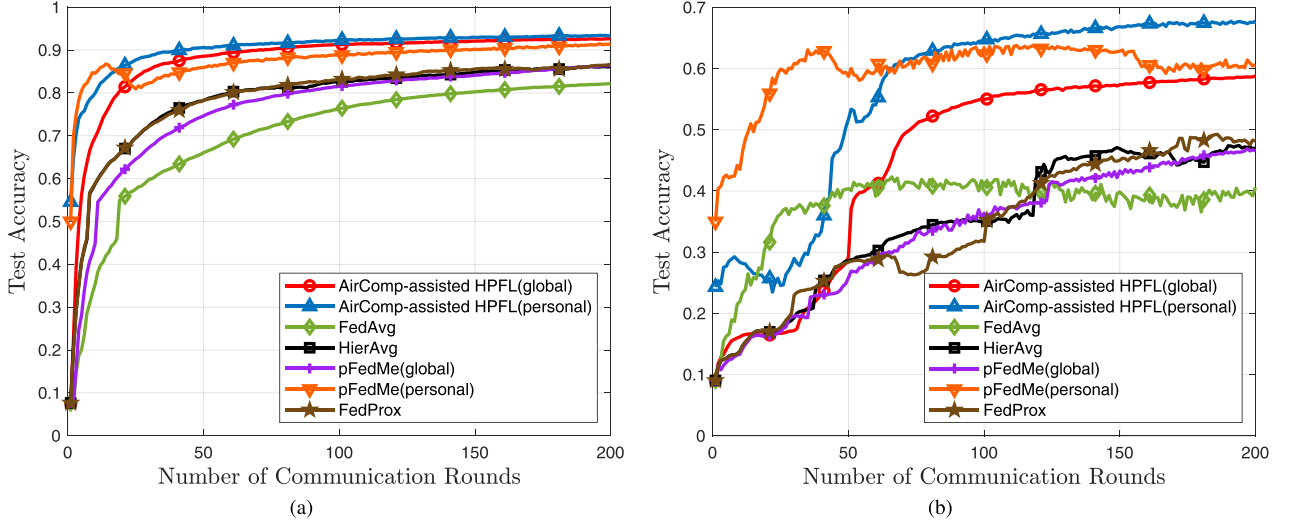


Fig. 4. Learning performance under different frameworks. (a) MNIST dataset. (b) FMNIST dataset.

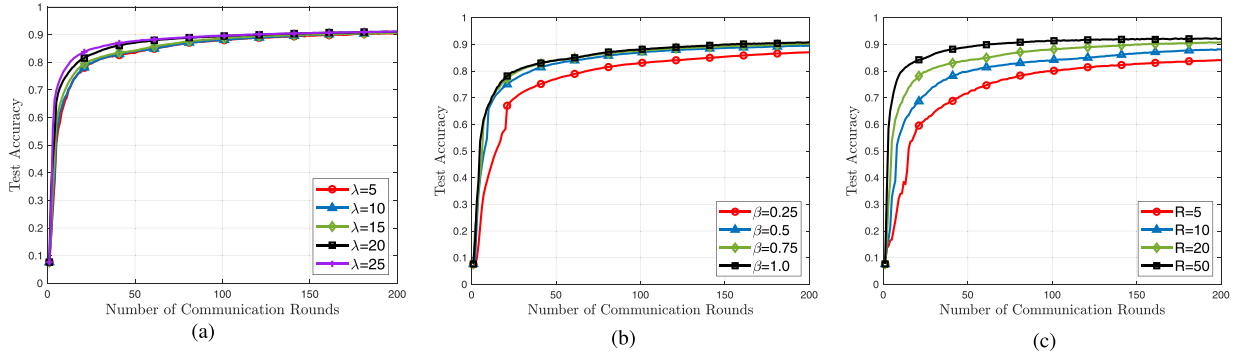


Fig. 5. Learning performance under different parameters. (a) Penalty coefficient  $\lambda$ . (b) Global update coefficient  $\beta$ . (c) Number of Edge Epochs  $R$ .

model that is close to the global model. On the other hand, with a smaller  $\lambda$ , the device can train a local model that is more adaptive to its local data. As for parameter  $\beta$ , which is used to control the smoothness and convergence speed of the global model update. In Fig. 5(b), we observe that with a larger  $\beta$ , the global model will converge faster. Fig. 5(c) shows that with a larger  $R$  (i.e., more interactions between edge servers and devices), learning performance can be improved.

## VII. CONCLUSION

In this paper, we introduced an innovative AirComp-assisted HPFL to tackle the problem of communication cost and data heterogeneity in FL. The convergence analysis of the framework was presented and efficient algorithms were proposed to minimize the signal distortion at both the cloud and the edge servers. Experimental results illustrate that the proposed framework with two developed algorithms can significantly improve the learning performance compared with other FL frameworks and other optimization algorithms.

## APPENDIX

To prove Theorem 1, we initially propose the following 4 Lemmas.

**Lemma 1:**  $\nabla F_{i,j}$  is  $L_F$ -smooth with  $L_F = \lambda_2$  (where  $\lambda_2 > 4L$ ), when  $\ell_{i,j}$  is nonconvex with  $L$ -Lipschitz  $\nabla \ell_{i,j}$ .

**Lemma 2:** For the solution  $\tilde{\theta}_{i,j}$  to (8), we have:

$$\mathbb{E} \left[ \left\| \tilde{\theta}_{i,j}^{t,r} - \hat{\theta}_{i,j}^{t,r} \right\|^2 \right] \leq \delta^2,$$

with

$$\delta^2 \triangleq \frac{2}{(\lambda_1 - L)^2} \left( \frac{\gamma_l^2}{|\mathcal{D}|} + \nu \right),$$

where  $\lambda_1 > L$  is required and  $|\mathcal{D}|$  is the mini-batch size.

**Lemma 3:** If  $\tilde{\eta} \leq \frac{\beta}{2L_F}$ , we have

$$\begin{aligned} & \frac{1}{NR} \sum_{i,r=1}^{N,R} \mathbb{E} \left[ \left\| \mathbf{g}_i^{t,r} - \nabla F_i(\mathbf{w}^t) \right\|^2 \right] \\ & \leq 2 \left[ \bar{\lambda}^2 \delta^2 \right. \\ & \quad \left. + \frac{8L_F \tilde{\eta}}{\beta} \left( \frac{3}{NM} \sum_{i,j=1}^{N,M} \mathbb{E} \left[ \left\| \nabla F_{i,j}(\mathbf{w}^t) \right\|^2 \right] + \frac{2\bar{\lambda}^2 \delta^2}{R} \right) \right], \end{aligned}$$

where  $\mathbf{g}_i^{t,r} = \frac{\lambda_1}{\lambda_1 + \lambda_2} \lambda_2 (\mathbf{w}_i^{t,r} - \boldsymbol{\theta}_i^{t,r+1})$  and  $\bar{\lambda} = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}$ .



*Lemma 4:* If  $\lambda \triangleq \frac{\lambda_1 \lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}} > 4L$ , we have

$$\begin{aligned} & \frac{1}{NM} \sum_{i,j=1}^{N,M} \|\nabla F_{i,j}(\mathbf{w}) - \nabla F(\mathbf{w})\|^2 \\ & \leq \frac{16L^2}{\lambda^2 - 16L^2} \|\nabla F(\mathbf{w})\|^2 + 2\sigma_F^2, \end{aligned}$$

where  $\sigma_F^2 \triangleq \frac{\lambda^2}{\lambda^2 - 16L^2} \sigma_l^2$ .

Lemma 1 can be proved due to Theorem 1 in [48] by setting  $\gamma_2 = 0$ , the proof of Lemmas 2, 3, 4 can be found in Appendixes A, B, and C, respectively.

#### A. Proof of Lemma 2

Note that the last term of (8) only with respect to  $\mathbf{y}_{i,j}$ , we have

$$\begin{aligned} \tilde{\boldsymbol{\theta}}_{i,j}(\mathbf{y}_{i,j}) &= \arg \min_{\boldsymbol{\theta}_{i,j} \in \mathbb{R}^D} \tilde{\ell}_{i,j}(\boldsymbol{\theta}_{i,j}) + \frac{\lambda_1}{2} \|\boldsymbol{\theta}_{i,j} - \mathbf{y}_{i,j}\|^2, \\ \hat{\boldsymbol{\theta}}_{i,j}(\mathbf{y}_{i,j}) &= \arg \min_{\boldsymbol{\theta}_{i,j} \in \mathbb{R}^D} \hat{\ell}_{i,j}(\boldsymbol{\theta}_{i,j}) + \frac{\lambda_1}{2} \|\boldsymbol{\theta}_{i,j} - \mathbf{y}_{i,j}\|^2. \end{aligned}$$

By denoting  $\check{h}(\boldsymbol{\theta}_{i,j}; \mathbf{y}_{i,j}^{t,r}, \mathcal{D}_{i,j}) = \ell_{i,j}(\boldsymbol{\theta}_{i,j}, \mathcal{D}_{i,j}) + \frac{\lambda_1}{2} \|\boldsymbol{\theta}_{i,j} - \mathbf{y}_{i,j}^{t,r}\|_2^2$ , we can derive that when Assumption 1 and  $\lambda_1 > L$  hold,  $\check{h}(\boldsymbol{\theta}_{i,j}; \mathbf{y}_{i,j}^{t,r}, \mathcal{D}_{i,j})$  is  $(\lambda_1 - L)$ -strong convex. Hence the result in Lemma 2 is obtained following the proof in [31].

#### B. Proof of Lemma 3

$$\begin{aligned} & \mathbb{E} [\|\mathbf{g}_i^{t,r} - \nabla F_i(\mathbf{w}^t)\|^2] \\ & \stackrel{(a)}{\leq} 2\mathbb{E} [\|\mathbf{g}_i^{t,r} - \nabla F_i(\mathbf{w}_i^{t,r})\|^2 + \|\nabla F_i(\mathbf{w}_i^{t,r}) - \nabla F_i(\mathbf{w}^t)\|^2] \\ & \stackrel{(b)}{\leq} 2\lambda_2^2 \mathbb{E} \left[ \left\| \frac{\lambda_1}{\lambda_1 + \lambda_2} (\mathbf{w}_i^{t,r} - \tilde{\boldsymbol{\theta}}_i^{t,r+1}) - (\mathbf{w}_i^{t,r} - \hat{\mathbf{y}}_i^{t,r}) \right\|^2 \right] \\ & \quad + L_F^2 \mathbb{E} [\|\mathbf{w}_i^{t,r} - \mathbf{w}^t\|^2] \\ & = 2\lambda_2^2 \mathbb{E} \left[ \left\| \frac{\lambda_1}{\lambda_1 + \lambda_2} (\tilde{\boldsymbol{\theta}}_i^{t,r+1} - \hat{\boldsymbol{\theta}}_i^{t,r}) \right\|^2 \right] \\ & \quad + L_F^2 \mathbb{E} [\|\mathbf{w}_i^{t,r} - \mathbf{w}^t\|^2] \\ & \stackrel{(c)}{\leq} \frac{2}{M} \sum_{j=1}^M \left( \bar{\lambda}^2 \mathbb{E} [\|\tilde{\boldsymbol{\theta}}_i^{t,r+1} - \hat{\boldsymbol{\theta}}_i^{t,r}\|^2] + L_F^2 \mathbb{E} [\|\mathbf{w}_i^{t,r} - \mathbf{w}^t\|^2] \right) \\ & \leq 2(\bar{\lambda}^2 \delta^2 + L_F^2 \mathbb{E} [\|\mathbf{w}_i^{t,r} - \mathbf{w}^t\|^2]). \end{aligned}$$

Where (a) is due to Jensen's inequality, (b) follows by the Lemma 1, and (c) follows by the Cauchy-Schwarz inequality. The last inequality is due to the Lemma 2 where  $\bar{\lambda} = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}$ . Moreover, the last term above can be bounded by

$$\begin{aligned} & \mathbb{E} [\|\mathbf{w}_i^{t,r} - \mathbf{w}^t\|^2] \\ & \stackrel{(a)}{\leq} \frac{8\tilde{\eta}}{\beta L_F} \left( 3\mathbb{E} [\|\nabla F_i(\mathbf{w}^t)\|^2] + \frac{2\bar{\lambda}^2 \delta^2}{R} \right) \end{aligned}$$

$$\leq \frac{8\tilde{\eta}}{\beta L_F} \left( 3\mathbb{E} \left[ \frac{1}{M} \sum_{j=1}^M [\|\nabla F_{i,j}(\mathbf{w}^t)\|^2] \right] + \frac{2\bar{\lambda}^2 \delta^2}{R} \right),$$

where (a) is due to Lemma 2 in [48].

#### C. Proof of Lemma 4

$$\begin{aligned} & \|\nabla F_{i,j}(\mathbf{w}) - \nabla F(\mathbf{w})\|^2 \\ & \stackrel{(1)}{=} \left\| \nabla \ell_{i,j}(\hat{\boldsymbol{\theta}}_{i,j}) - \frac{1}{NM} \sum_{p,q}^{N,M} \nabla \ell_{p,q}(\hat{\boldsymbol{\theta}}_{p,q}) \right\|^2 \\ & \stackrel{(2)}{\leq} 2 \left\| \nabla \ell_{i,j}(\hat{\boldsymbol{\theta}}_{i,j}) - \frac{1}{NM} \sum_{p,q}^{N,M} \nabla \ell_{p,q}(\hat{\boldsymbol{\theta}}_{i,j}) \right\|^2 \\ & \quad + 2 \left\| \frac{1}{NM} \sum_{p,q}^{N,M} \nabla \ell_{p,q}(\hat{\boldsymbol{\theta}}_{i,j}) - \frac{1}{NM} \sum_{p,q}^{N,M} \nabla \ell_{p,q}(\hat{\boldsymbol{\theta}}_{p,q}) \right\|^2, \end{aligned}$$

where (1) follows by the first order condition in (3), and (2) is due to the Jensen's inequality. Then we take the average over all devices in each cluster:

$$\begin{aligned} & \frac{1}{NM} \sum_{i,j=1}^{N,M} \|\nabla F_{i,j}(\mathbf{w}) - \nabla F(\mathbf{w})\|^2 \\ & \stackrel{(1)}{\leq} 2\sigma_l^2 + \frac{2}{(NM)^2} \sum_{i,j}^{N,M} \sum_{p,q=1}^{N,M} \left\| \nabla \ell_{p,q}(\hat{\boldsymbol{\theta}}_{i,j}) - \nabla \ell_{p,q}(\hat{\boldsymbol{\theta}}_{p,q}) \right\|^2 \\ & \stackrel{(2)}{\leq} 2\sigma_l^2 + \frac{2L^2}{(NM)^2} \sum_{i,j}^{N,M} \sum_{p,q=1}^{N,M} \|\hat{\boldsymbol{\theta}}_{i,j} - \hat{\boldsymbol{\theta}}_{p,q}\|^2, \end{aligned} \quad (58)$$

where (1) is due to Assumption 3 and the Jensen's inequality, (2) follows by  $L$ -smooth of  $\ell_{i,j}(\cdot)$ . Then the last term can be written as

$$\begin{aligned} & \sum_{i,j}^{N,M} \sum_{p,q=1}^{N,M} \|\hat{\boldsymbol{\theta}}_{i,j} - \hat{\boldsymbol{\theta}}_{p,q}\|^2 \\ & \stackrel{(a)}{\leq} 4 \sum_{i,j}^{N,M} \sum_{p,q=1}^{N,M} (\|\hat{\boldsymbol{\theta}}_{i,j} - \hat{\mathbf{y}}_{i,j}\|^2 + \|\hat{\boldsymbol{\theta}}_{p,q} - \hat{\mathbf{y}}_{p,q}\|^2 \\ & \quad + \|\hat{\mathbf{y}}_{i,j} - \mathbf{w}\|^2 + \|\hat{\mathbf{y}}_{p,q} - \mathbf{w}\|^2) \\ & \stackrel{(b)}{=} \sum_{i,j}^{N,M} \sum_{p,q=1}^{N,M} \left( \frac{1}{\lambda_1^2} + \frac{1}{\lambda_2^2} \right) (\|\nabla F_{i,j}(\mathbf{w})\|^2 + \|\nabla F_{p,q}(\mathbf{w})\|^2) \\ & = \frac{8NM(\lambda_1^2 + \lambda_2^2)}{\lambda_1^2 \lambda_2^2} \sum_{i,j=1}^{N,M} \|\nabla F_{i,j}(\mathbf{w})\|^2, \end{aligned} \quad (59)$$

where (a) is derived from the Jensen's inequality and (b) is derived from the first order condition in (3). By substituting

(59) into (58), we get the following inequality:

$$\begin{aligned}
 & \frac{1}{NM} \sum_{i,j=1}^{N,M} \|\nabla F_{i,j}(\mathbf{w}) - \nabla F(\mathbf{w})\|^2 \\
 & \leq 2\sigma_l^2 + \frac{16L^2(\lambda_1^2 + \lambda_2^2)}{NM\lambda_1^2\lambda_2^2} \sum_{i,j=1}^{N,M} \|\nabla F_{i,j}(\mathbf{w})\|^2 \\
 & \stackrel{(a)}{\leq} 2\sigma_l^2 + \frac{16L^2(\lambda_1^2 + \lambda_2^2)}{NM\lambda_1^2\lambda_2^2} \\
 & \quad \left( \|\nabla F(\mathbf{w})\|^2 + \frac{1}{NM} \sum_{i,j=1}^{N,M} \|\nabla F_{i,j}(\mathbf{w}) - \nabla F(\mathbf{w})\|^2 \right) \\
 & \stackrel{(b)}{\leq} \frac{16L^2}{\lambda^2 - 16L^2} \|\nabla F(\mathbf{w})\|^2 + \frac{2\lambda^2}{\lambda^2 - 16L^2} \sigma_l^2,
 \end{aligned}$$

where we denote  $\lambda \triangleq \frac{\lambda_1\lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2}}$ , (a) is due to  $\mathbb{E}[\|\mathbf{a} - \mathbb{E}[\mathbf{a}]\|^2] = \mathbb{E}[\|\mathbf{a}\|^2] - \mathbb{E}[\|\mathbb{E}[\mathbf{a}]\|^2]$ , (b) is derived by rearranging the terms.

#### D. Proof of Theorem 1

We denote  $\Theta^t = F(\mathbf{w}^{t+1}) - F(\mathbf{w}^t)$ , and due to the  $L$ -smooth of  $F(\cdot)$ :

$$\begin{aligned}
 \Theta^t & \leq \langle \nabla F(\mathbf{w}^t), \mathbf{w}^{t+1} - \mathbf{w}^t \rangle + \frac{L_F}{2} \|\mathbf{w}^{t+1} - \mathbf{w}^t\|_2^2 \\
 & = -\tilde{\eta} \langle \nabla F(\mathbf{w}^t), \mathbf{g}^t \rangle - \tilde{\eta} \langle \nabla F(\mathbf{w}^t), \mathbf{e}^t \rangle + \\
 & \quad \frac{L_F}{2} \tilde{\eta}^2 \|\mathbf{g}^t\|_2^2 + \frac{L_F}{2} \tilde{\eta}^2 \|\mathbf{e}^t\|_2^2 + L_F \tilde{\eta}^2 \langle \mathbf{g}^t, \mathbf{e}^t \rangle \\
 & \stackrel{(a)}{\leq} -\tilde{\eta} \langle \nabla F(\mathbf{w}^t), \mathbf{g}^t \rangle + \frac{L_F}{2} \tilde{\eta}^2 \|\mathbf{g}^t\|_2^2 + \frac{L_F}{2} \tilde{\eta}^2 \|\mathbf{e}^t\|_2^2 \\
 & \quad + \tilde{\eta} \left( \frac{\|\nabla F(\mathbf{w}^t)\|_2^2}{2} + \frac{\|\mathbf{e}^t\|_2^2}{2} \right) \\
 & \quad + L_F \tilde{\eta}^2 \left( \frac{\|\mathbf{g}^t\|_2^2}{2} + \frac{\|\mathbf{e}^t\|_2^2}{2} \right),
 \end{aligned}$$

where (a) follows by  $\mathbf{m}^T \mathbf{n} = \frac{1}{2} \|\mathbf{m}\|_2^2 + \frac{1}{2} \|\mathbf{n}\|_2^2 - \frac{1}{2} \|\mathbf{m} - \mathbf{n}\|_2^2$ . Then we take an expectation at both sides:

$$\begin{aligned}
 & \mathbb{E}[\Theta^t] \\
 & \leq -\tilde{\eta} \mathbb{E} \langle \nabla F(\mathbf{w}^t), \mathbf{g}^t \rangle + \frac{\tilde{\eta}}{2} \mathbb{E} \|\nabla F(\mathbf{w}^t)\|_2^2 \\
 & \quad + \left( \frac{L_F}{2} \tilde{\eta}^2 + \frac{L_F}{2} \tilde{\eta}^2 \right) \mathbb{E} \|\mathbf{g}^t\|_2^2 \\
 & \quad + \left( \frac{\tilde{\eta}}{2} + \frac{L_F}{2} \tilde{\eta}^2 + \frac{L_F}{2} \tilde{\eta}^2 \right) \mathbb{E} \|\mathbf{e}^t\|_2^2 \\
 & = \underbrace{-\tilde{\eta} \mathbb{E} \langle \nabla F(\mathbf{w}^t), \mathbf{g}^t \rangle}_{(1)} + \underbrace{\frac{\tilde{\eta}}{2} \mathbb{E} \|\nabla F(\mathbf{w}^t)\|_2^2}_{(2)} + \underbrace{L_F \tilde{\eta}^2 \mathbb{E} \|\mathbf{g}^t\|_2^2}_{(3)} \\
 & \quad + \underbrace{\left( \frac{\tilde{\eta}}{2} + L_F \tilde{\eta}^2 \right) \mathbb{E} \|\mathbf{e}^t\|_2^2}_{(4)}.
 \end{aligned}$$

As for part (1), (2) and (3), we have:

$$\begin{aligned}
 & (1) + (2) + (3) \\
 & = -\frac{\tilde{\eta}}{2} \mathbb{E} \|\nabla F(\mathbf{w}^t)\|^2 - \tilde{\eta} \mathbb{E} \langle \nabla F(\mathbf{w}^t), \mathbf{g}^t - \nabla F(\mathbf{w}^t) \rangle \\
 & \quad + L_F \tilde{\eta}^2 \mathbb{E} \|\mathbf{g}^t\|^2 \\
 & \stackrel{(b)}{=} -\frac{\tilde{\eta}}{2} \mathbb{E} \|\nabla F(\mathbf{w}^t)\|^2 - \frac{\tilde{\eta}}{2} \mathbb{E} \|\nabla F(\mathbf{w}^t)\|^2 - \frac{\tilde{\eta}}{2} \|\mathbf{g}^t - \nabla F(\mathbf{w}^t)\|^2 \\
 & \quad + \frac{\tilde{\eta}}{2} \|\mathbf{g}^t\|^2 + L_F \tilde{\eta}^2 \mathbb{E} \|\mathbf{g}^t\|^2 \\
 & \stackrel{(c)}{\leq} -\tilde{\eta} \mathbb{E} \|\nabla F(\mathbf{w}^t)\|^2 \\
 & \quad + \tilde{\eta} (1 + 3L_F \tilde{\eta}) \mathbb{E} \left\| \frac{1}{NR} \sum_{i,r=1}^{N,R} \mathbf{g}_i^{t,r} - \nabla F_i(\mathbf{w}^t) \right\|^2 \\
 & \quad + \tilde{\eta} \left( \frac{3}{2} + 3L_F \tilde{\eta} \right) \mathbb{E} \left\| \frac{1}{N} \sum_{i=1}^N \nabla F_i(\mathbf{w}^t) - \nabla F(\mathbf{w}^t) \right\|^2 \\
 & \quad + \tilde{\eta} \left( \frac{3}{2} + 3L_F \tilde{\eta} \right) \mathbb{E} \|\nabla F(\mathbf{w}^t)\|^2 \\
 & \stackrel{(d)}{\leq} -\tilde{\eta} \mathbb{E} \|\nabla F(\mathbf{w}^t)\|^2 \\
 & \quad + \tilde{\eta} (1 + 3L_F \tilde{\eta}) \mathbb{E} \left\| \frac{1}{NR} \sum_{i,r=1}^{N,R} \mathbf{g}_i^{t,r} - \nabla F_i(\mathbf{w}^t) \right\|^2 \\
 & \quad + \tilde{\eta} \left( \frac{3}{2} + 3L_F \tilde{\eta} \right) \mathbb{E} \left\| \frac{1}{NM} \sum_{i,j=1}^{N,M} \nabla F_{i,j}(\mathbf{w}^t) \right\|^2 \\
 & \stackrel{(e)}{\leq} -\tilde{\eta} A \mathbb{E} [\|\nabla F(\mathbf{w}^t)\|^2] + \tilde{\eta} A_1 + \tilde{\eta}^2 A_2,
 \end{aligned}$$

where (b) is due to  $\mathbf{m}^T \mathbf{n} = \frac{1}{2} \|\mathbf{m}\|_2^2 + \frac{1}{2} \|\mathbf{n}\|_2^2 - \frac{1}{2} \|\mathbf{m} - \mathbf{n}\|_2^2$ , (c) is due to Jensen inequality, (d) is due to the Jensen inequality and  $\mathbb{E}[\|\mathbf{s} - \mathbb{E}[\mathbf{s}]\|^2] = \mathbb{E}[\|\mathbf{s}\|^2] - \mathbb{E}[\|\mathbb{E}[\mathbf{s}]\|^2]$ , and (e) follows by  $\frac{3}{2} + 3L_F \tilde{\eta} \leq 4\beta$ , since we let  $\tilde{\eta} \leq \frac{\beta}{8L_F}$  and  $A_1 = 8\beta\lambda^2\delta^2 + 16\sigma_F^2$ ,  $A_2 = \frac{128L_F(3R\sigma_F^2 + \lambda^2\delta^2)}{R}$ , and by letting  $\lambda^2 - 16L^2 \geq 1$  and  $\tilde{\eta} \leq \frac{1}{384L_F\lambda^2}$  hold, we have

$$A = 1 - 192L_F\tilde{\eta} \frac{\lambda^2}{\lambda^2 - 16L^2} \geq \frac{1}{2}.$$

As a result, we have

$$\begin{aligned}
 & \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla F(\mathbf{w}^t)\|^2] \\
 & \leq 2 \left( \frac{\mathbb{E}[F(\mathbf{w}^0) - F(\mathbf{w}^T)]}{\tilde{\eta}T} + A_1 + \tilde{\eta} A_2 \right) \\
 & \quad + \frac{1}{T} \sum_{t=0}^{T-1} (1 + 2L_F \tilde{\eta}) \mathbb{E} \|\mathbf{e}^t\|^2
 \end{aligned}$$

$$\leq 2 \left( \frac{\Delta_F}{\tilde{\eta}T} + A_1 + \tilde{\eta}A_2 \right) + \frac{1}{T} \sum_{t=0}^{T-1} 2(1 + 2L_F\tilde{\eta})\mathbb{E}\|e^t\|^2,$$

where  $\Delta_F = \mathbb{E}[F(\mathbf{w}^0) - F(\mathbf{w}^*)]$ .

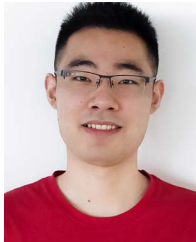
## REFERENCES

- [1] F. Zhou, Z. Wang, X. Luo, and Y. Zhou, "Over-the-air computation assisted hierarchical personalized federated learning," in *Proc. IEEE Int. Conf. Commun.*, May 2023, pp. 5940–5945.
- [2] Y. Zhou, Y. Shi, H. Zhou, J. Wang, L. Fu, and Y. Yang, "Towards scalable wireless federated learning: Challenges and solutions," *IEEE Internet Things Mag.*, vol. 6, no. 4, pp. 10–16, Dec. 2023.
- [3] Y. Shi, Y. Zhou, D. Wen, Y. Wu, C. Jiang, and K. B. Letaief, "Task-oriented communications for 6G: Vision, principles, and technologies," *IEEE Wireless Commun. Mag.*, vol. 30, no. 3, pp. 78–85, Jun. 2023.
- [4] Y. Shi et al., "Machine learning for large-scale optimization in 6G wireless networks," *IEEE Commun. Surv. Tuts.*, vol. 25, no. 4, pp. 2088–2132, fourthquarter 2023.
- [5] H. Yang, J. Zhao, Z. Xiong, K.-Y. Lam, S. Sun, and L. Xiao, "Privacy-preserving federated learning for UAV-enabled networks: Learning-based joint scheduling and resource management," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 10, pp. 3144–3159, Oct. 2021.
- [6] L. Liu, J. Zhang, S. Song, and K. B. Letaief, "Client-edge-cloud hierarchical federated learning," in *Proc. IEEE Int. Conf. Commun.*, May 2020, pp. 1–6.
- [7] W. Y. B. Lim et al., "Hierarchical incentive mechanism design for federated machine learning in mobile networks," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9575–9588, Oct. 2020.
- [8] W. Y. B. Lim, J. S. Ng, Z. Xiong, D. Niyato, C. Miao, and D. I. Kim, "Dynamic edge association and resource allocation in self-organizing hierarchical federated learning networks," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3640–3653, Dec. 2021.
- [9] J. Wang, S. Wang, R.-R. Chen, and M. Ji, "Demystifying why local aggregation helps: Convergence analysis of hierarchical SGD," in *Proc. Conf. Artif. Intell. (AAAI)*, Jan. 2022.
- [10] L. Liu, J. Zhang, S. Song, and K. B. Letaief, "Hierarchical federated learning with quantization: Convergence analysis and system design," *IEEE Trans. Wireless Commun.*, vol. 22, no. 1, pp. 2–18, Jan. 2023.
- [11] M. S. H. Abad, E. Ozfatura, D. Gunduz, and O. Ercetin, "Hierarchical federated learning across heterogeneous cellular networks," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, May 2020, pp. 8866–8870.
- [12] S. Luo, X. Chen, Q. Wu, Z. Zhou, and S. Yu, "HFEL: Joint edge association and resource allocation for cost-efficient hierarchical federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 10, pp. 6535–6548, Oct. 2020.
- [13] Z. Wang, Y. Zhou, Y. Zou, Q. An, Y. Shi, and M. Bennis, "A graph neural network learning approach to optimize RIS-assisted federated learning," *IEEE Trans. Wireless Commun.*, vol. 22, no. 9, pp. 6092–6106, Sep. 2023.
- [14] Z. Wang, Y. Zhao, Y. Zhou, Y. Shi, C. Jiang, and K. B. Letaief, "Over-the-air computation for 6G: Foundations, technologies, and applications," *IEEE Internet Things J.*, vol. 11, no. 14, pp. 24634–24658, Jul. 2024.
- [15] 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.
- [16] L. Chen, N. Zhao, Y. Chen, F. R. Yu, and G. Wei, "Over-the-air computation for IoT networks: Computing multiple functions with antenna arrays," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 5296–5306, Dec. 2018.
- [17] Y. Chen, H. Xing, J. Xu, L. Xu, and S. Cui, "Over-the-air computation in OFDM systems with imperfect channel state information," *IEEE Trans. Commun.*, vol. 72, no. 5, pp. 2929–2944, May 2024.
- [18] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, Mar. 2020.
- [19] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 491–506, Jan. 2020.
- [20] Y. Sun, S. Zhou, Z. Niu, and D. Gundüz, "Dynamic scheduling for over-the-air federated edge learning with energy constraints," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 227–242, Jan. 2022.
- [21] X. Cao, G. Zhu, J. Xu, Z. Wang, and S. Cui, "Optimized power control design for over-the-air federated edge learning," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 342–358, Jan. 2022.
- [22] Z. Wang et al., "Federated learning via intelligent reflecting surface," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 808–822, Feb. 2022.
- [23] 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.
- [24] O. Aygün, M. Kazemi, D. Gundüz, and T. M. Duman, "Hierarchical over-the-air federated edge learning," in *Proc. IEEE Int. Conf. Commun.*, May 2022, pp. 3376–3381.
- [25] F. Zhou, X. Chen, H. Shan, and Y. Zhou, "Adaptive transceiver design for wireless hierarchical federated learning," in *Proc. IEEE Veh. Technol. Conf.*, Oct. 2023, pp. 1–6.
- [26] W. Guo, C. Huang, X. Qin, L. Yang, and W. Zhang, "Dynamic clustering and power control for two-tier wireless federated learning," *IEEE Trans. Wireless Commun.*, vol. 23, no. 2, pp. 1356–1371, Feb. 2024.
- [27] Z. Chen, Z. Li, H. H. Yang, and T. Q. S. Quek, "Personalizing federated learning with over-the-air computations," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Jun. 2023, pp. 1–5.
- [28] H. U. Sami and B. Güler, "Over-the-air personalized federated learning," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, May 2022, pp. 8777–8781.
- [29] A. Z. Tan, H. Yu, L. Cui, and Q. Yang, "Towards personalized federated learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 12, pp. 9857–9603, Dec. 2023.
- [30] A. Li, J. Sun, X. Zeng, M. Zhang, H. Li, and Y. Chen, "Fedmask: Joint computation and communication-efficient personalized federated learning via heterogeneous masking," in *Proc. ACM Conf. Embedded Networked Sensor Syst.*, Nov. 2022, pp. 42–55.
- [31] C. T. Dinh, N. Tran, and J. Nguyen, "Personalized federated learning with moreau envelopes," in *Proc. Neural Inf. Process. Syst.*, Dec. 2020, pp. 21394–21405.
- [32] Z. Ma, Y. Xu, H. Xu, J. Liu, and Y. Xue, "Like attracts like: Personalized federated learning in decentralized edge computing," *IEEE Trans. Mobile Comput.*, vol. 23, no. 2, pp. 1080–1096, Feb. 2024.
- [33] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Mach. Learn. Syst.*, Mar. 2020, pp. 429–450.
- [34] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *Proc. Int. Conf. Mach. Learn.*, Jul. 2020, pp. 5132–5143.
- [35] D. Li and J. Wang, "Fedmd: Heterogeneous federated learning via model distillation," 2019, *arXiv:1910.03581*.
- [36] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Proc. Neural Inf. Process. Syst.*, Dec. 2017, pp. 4427–4437.
- [37] M. Mortaheb, C. Vahapoglu, and S. Ulukus, "Personalized federated multi-task learning over wireless fading channels," *Algorithms*, vol. 15, no. 11, 2022, Art. no. 421.
- [38] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2013, pp. 1139–1147.
- [39] Y. Zou, Z. Wang, X. Chen, H. Zhou, and Y. Zhou, "Knowledge-guided learning for transceiver design in over-the-air federated learning," *IEEE Trans. Wireless Commun.*, vol. 22, no. 1, pp. 270–285, Jan. 2023.
- [40] 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.
- [41] M. P. Friedlander and M. Schmidt, "Hybrid deterministic-stochastic methods for data fitting," *SIAM J. Sci. Comput.*, vol. 34, no. 3, pp. A1380–A1405, 2012.
- [42] L. Chen, X. Qin, and G. Wei, "A uniform-forcing transceiver design for over-the-air function computation," *IEEE Wireless Commun. Lett.*, vol. 7, no. 6, pp. 942–945, Dec. 2018.
- [43] M. Dong and Q. Wang, "Multi-group multicast beamforming: Optimal structure and efficient algorithms," *IEEE Trans. Signal Process.*, vol. 68, pp. 3738–3753, 2020.
- [44] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," 2014. [Online]. Available: <http://cvxr.com/cvx>
- [45] Z.-Q. Luo, W.-K. Ma, A. M. -C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.
- [46] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2005.

- [47] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2017, pp. 1273–1282.
- [48] X. Liu, Q. Wang, Y. Shao, and Y. Li, "Sparse federated learning with hierarchical personalization models," *IEEE Internet Things J.*, vol. 11, no. 5, pp. 8539–8551, Mar. 2024.



**Fangtong Zhou** (Graduate Student Member, IEEE) received the B.Eng. degree in information engineering from the South China University of Technology, Guangzhou, China, in 2021, and the M.S. degree in information and communication engineering from ShanghaiTech University, Shanghai, China, in 2024. He is currently working toward the Ph.D. degree with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA. His research interests include Internet of Things, and federated learning.



**Zhibin Wang** (Member, IEEE) received the B.S. degree in telecommunications engineering from Xi-dian University, Xi'an, China, in 2019, and the Ph.D. degree in computer science from ShanghaiTech University, Shanghai, China, in 2024. His research interests include Internet of Things, intelligent reflecting surface, and federated learning.



**Hanguan Shan** (Senior Member, IEEE) received the B.Sc. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2004, and the Ph.D. degree in electrical engineering from Fudan University, Shanghai, China, in 2009. From 2009 to 2010, he was a Postdoctoral Research Fellow with the University of Waterloo, Waterloo, ON, Canada. Since 2011, he has been with the College of Information Science and Electronic Engineering, Zhejiang University, where he is currently an Associate Professor. He is also with the Zhejiang Provincial Key

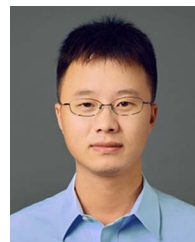
Laboratory of Information Processing and Communication Networks, Zhejiang University. His research interests include machine learning-enabled resource allocation and quality-of-service provisioning in wireless networks. Dr. Shan was the recipient of the Best Industry Paper Award from the IEEE WCNC'11 and Best Paper Award from the IEEE WCSP'23. He was a Technical Program Committee Member of various international conferences. He was the Editor of the IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING.



**Liantao Wu** (Member, IEEE) received the B.E. degree in automation from Shandong University, Jinan, China, in 2012, and the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2017. He is currently an Associate Professor with East China Normal University, Shanghai, China. His research interests include IoT, edge intelligence, and edge computing.



**Xiaohua Tian** (Senior Member, IEEE) received the B.E. and M.E. degrees in communication engineering from Northwestern Polytechnical University, Xi'an, China, in 2003 and 2006, respectively, and the Ph.D. degree from the Department Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, IL, USA, in 2010. Since 2011, he has been with the Department of Electronics Engineering, Shanghai Jiao Tong University, Shanghai, China. He is currently a Professor and the Associate Dean with the Department of Electronics Engineering, Shanghai Jiao Tong University. He received the Excellent Young Scientists Fund from the National Science Foundation of China in 2019. Since 2016, he has been the Scanning Literature Column Editor of *IEEE Network Magazine*. From 2018 to 2019, he was the Guest Editor of IEEE INTERNET OF THINGS JOURNAL. He was also the Vice Program Co-Chair for the ACM Turing Celebration Conference in 2019, a TPC Member for IEEE INFOCOM from 2014 to 2018, and 2020, IEEE GLOBECOM from 2011 to 2018, and IEEE ICC from 2011 to 2018, and the Symposium Co-Chair for IEEE/CIC ICC in 2015 and 2019. He was the recipient of the ACM China Rising Star Award in 2017, ACM MobiCom 2018 Best Mobile APP Award, IEEE ICNC 2015, and IEEE VTC-Fall 2017 Best Paper Award.



**Yuanming Shi** (Senior Member, IEEE) received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, in 2011, and the Ph.D. degree in electronic and computer engineering from The Hong Kong University of Science and Technology, Hong Kong, in 2015. Since 2015, he has been with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China, where he is currently a Full Professor. From 2016 to 2017, he visited University of California, Berkeley, CA, USA. His research interests include edge AI,

federated edge learning, task-oriented communications, and satellite networks. He is also the Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, AND JOURNAL OF COMMUNICATIONS AND INFORMATION NETWORKS. He was the recipient of the IEEE Marconi Prize Paper Award in Wireless Communications in 2016, Young Author Best Paper Award by the IEEE Signal Processing Society in 2016, IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2021, and the Chinese Institute of Electronics First Prize in Natural Science in 2022. He is an IET Fellow.



**Yong Zhou** (Senior Member, IEEE) received the B.Sc. and M.Eng. degrees from Shandong University, Jinan, China, in 2008 and 2011, respectively, and the Ph.D. degree from the University of Waterloo, Waterloo, ON, Canada, in 2015. From 2015 to 2018, he was a Postdoctoral Researcher Fellow with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada. Since 2018, he has been with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China, where he is currently a

Tenured Associate Professor. His research interests include 6G communications, edge intelligence, and Internet of Things. He was an Associate Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY. He was the track co-chair of IEEE VTC 2020 Fall and IEEE VTC 2023 Spring, and the Co-Chair of IEEE ICC 2022 workshop on edge artificial intelligence for 6G.