Over-the-Air Computation Assisted Hierarchical Personalized Federated Learning

Fangtong Zhou*†‡, Zhibin Wang*, Xiliang Luo*, Yong Zhou*
*School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China
†Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, China
‡University of Chinese Academy of Sciences, Beijing 100049, China
E-mail: {zhouft, wangzhb, luoxl, zhouyong}@shanghaitech.edu.cn

***Abstract*—Communication bottleneck and statistical heterogeneity are two critical challenges of federated learning (FL) over wireless networks. To tackle both challenges, in this paper we propose an over-the-air computation (AirComp) assisted hierarchical personalized FL (HPFL) framework, where a device-edge-cloud based three-tier network architecture is adopted to simultaneously learn a global model and multiple personalized local models. We analyze the convergence of the AirComp-assisted HPFL framework and formulate an optimization problem to minimize the transmission distortion, which is an essential component of the convergence upper bound. An efficient algorithm is subsequently developed to optimize the transceiver design by leveraging successive convex approximation and Lagrangian duality. We conduct extensive simulations to demonstrate that our developed algorithm achieves a near-optimal performance and a much greater test accuracy than the baseline algorithms.**

I. Introduction

Federated learning (FL) is a distributed learning framework for training a statistical model with geographically distributed data and computing power [1]. By exchanging model parameters instead of raw data, FL is capable of guaranteeing the data privacy and reducing the communication cost. With these advantages, FL can be applied in many applications, including smart health, autonomous driving, and Internet of Things [2].

Despite the aforementioned benefits, implementing FL over wireless networks face many challenges introduced by adverse channel fading and limited spectrum resources. To overcome these challenges, hierarchical FL (HFL) [3] is gaining incresing attention given its potential to mitigate the detrimental impact of channel fading by allowing the devices to transmit their model parameters to the nearby edge servers, which then forward the model parameters over the central server. Such a hierarchical architecture reduces the communication distance and further mitigates the communication delay [4]–[6]. To tackle the issue of limited radio resources, over-the-air computation (AirComp) [7], [8] has been employed to achieve efficient model aggregation by allowing the concurrent but non-orthogonal transmission from multiple devices. Specifically, by exploiting the natural feature of analog waveform superposition, a specific function of concurrently transmitted signals can be directly received at the edge server via AirComp without decoding each signal. Such a feature can be leveraged to achieve low-latency model aggregation in FL, as the server only requires the average of local models to obtain an updated global model [9]–[14]. Meanwhile, FL also confronts the issue introduced by statistical heterogeneity of local datasets, which leads to severe training performance degradation. Personalized FL (PFL) has been recognized as an effective way to address the statistical heterogeneity challenge [15]–[17].

Most existing works on wireless FL either focus on using hierarchical architecture and AirComp-based aggregation to reduce communication cost, or applying the personalized strategy to reduce the impact of data heterogeneity. However, none of them exploit both AirComp and personalization in the hierarchical architecture to simultaneously tackle the communication bottleneck and data heterogeneity problems.

In this paper, we propose an AirComp-assisted hierarchical personalized FL (HPFL) over wireless networks, where a global model and multiple personalized local models are jointly learned by leveraging regularized local loss functions. We aim at characterizing the convergence of the AirComp-assisted HPFL and optimizing the transceiver design to promote the learning performance. The main contributions of this paper are three-fold. First, the convergence of our proposed HPFL is theoretically derived and analyzed. Second, an optimization problem is formulated to minimize the transmission distortion, which is an essential component of the convergence upper bound. To tackle the formulated non-convex optimization problem, an efficient algorithm on the basis of successive convex approximation (SCA) and Lagrangian duality is developed. Third, simulations demonstrate that our proposed transceiver design in AirComp-assisted HPFL system achieves a near-optimal performance and outperforms the baseline algorithms.

II. System Model

A. HPFL Model

Consider a device-edge-cloud based three-tier hierarchical FL framework, as shown in Fig. 1, where one cloud server connects to N edge servers and each edge server associates with its proximal M devices that own non-independent and identically distributed (non-i.i.d.) local datasets. We denote the j-th device in the i-th cluster as $c_{i,j}$ and its local dataset as $\mathcal{B}_{i,j}$. To tackle data heterogeneity, we jointly optimize global model $\boldsymbol{w} \in \mathbb{R}^D$ and local personalized model $\boldsymbol{\theta}_{i,j} \in \mathbb{R}^D$ for each device $c_{i,j}$. To this end, we solve the following problem

$$\min_{\boldsymbol{w}} F(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N} F_i(\boldsymbol{w}) \tag{1}$$

with $F_i(\boldsymbol{w}) = \frac{1}{M}\sum_{j=1}^{M} F_{i,j}(\boldsymbol{w})$. Herein, $F_{i,j}(\boldsymbol{w})$ can be expressed as

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

This work was supported by the National Natural Science Foundation of China (NSFC) under grants 61971286 and 62001294.

ICC 2023 - IEEE International Conference on Communications | 978-1-5386-7462-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICC45041.2023.10278799

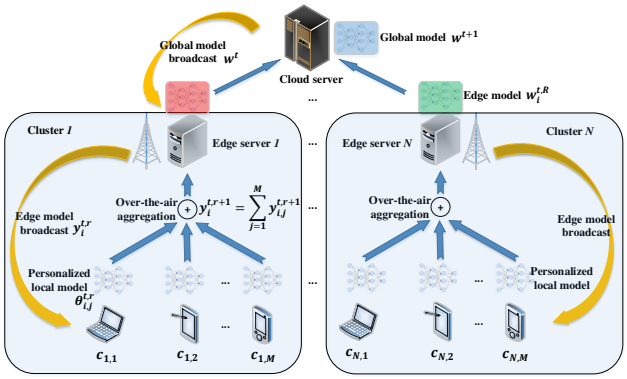


Fig. 1. Illustration of AirComp-assisted HPFL.

where $\ell_{i,j}(\cdot)$ denotes the sample-wise loss function over a data sample in dataset $\mathcal{B}_{i,j}$ and $\boldsymbol{y}_{i,j}$ denotes the intermediate model at device $c_{i,j}$. We denote $\boldsymbol{y}_i \in \mathbb{R}^D$ as the aggregation of $\boldsymbol{y}_{i,j} \in \mathbb{R}^D$ in the i-th cluster. Parameters λ_1 and λ_2 in (2) control the degree of the connection between global model $\boldsymbol{w}$ and personalized model $\boldsymbol{\theta}_{i,j}$. In the hierarchical FL system, the loss function is optimized as follows:

- At the t-th communication round, the cloud server broadcasts global model $\boldsymbol{w}^t$ to each edge server.
- After receiving the global model, each edge server i initializes intermediate model $\boldsymbol{y}_i^{t,0}$ and edge model $\boldsymbol{w}_i^{t,0}$ as $\boldsymbol{w}^t$, and then broadcasts $\boldsymbol{y}_i^{t,0}$ to its associated devices in the first edge communication round.
- At the device side, after receiving $\boldsymbol{y}_i^{t,r}$ in the r-th edge communication round, the personalized local model $\boldsymbol{\theta}_{i,j}^{t,r}$ and intermediate model $\boldsymbol{y}_{i,j}^{t,r+1}$ are, respectively, updated by minimizing (2), i.e.,

$$\boldsymbol{\theta}_{i,j}^{t,r} = \arg\min_{\boldsymbol{\theta}_{i,j}} \ell_{i,j}(\boldsymbol{\theta}_{i,j}) + \frac{\lambda_1}{2}\|\boldsymbol{\theta}_{i,j} - \boldsymbol{y}_{i,j}^{t,r}\|_2^2 \tag{3}$$

$$\boldsymbol{y}_{i,j}^{t,r+1} = \arg\min_{\boldsymbol{y}_{i,j}} \frac{\lambda_1}{2}\|\boldsymbol{\theta}_{i,j}^{t,r} - \boldsymbol{y}_{i,j}\|_2^2 + \frac{\lambda_2}{2}\|\boldsymbol{y}_{i,j} - \boldsymbol{w}_i^{t,r}\|_2^2. \tag{4}$$

- After the update of all local models, edge servers perform the edge aggregation to obtain $\boldsymbol{y}_i^{t,r+1}$ and update edge model $\boldsymbol{w}_i^{t,r+1}$ as

$$\begin{aligned}\boldsymbol{y}_i^{t,r+1} &= \frac{1}{M}\sum_{j=1}^{M}\boldsymbol{y}_{i,j}^{t,r+1} \\ &= \frac{1}{M}\sum_{j=1}^{M}\frac{\lambda_1}{\lambda_1+\lambda_2}\boldsymbol{\theta}_{i,j}^{t,r} + \frac{\lambda_2}{\lambda_1+\lambda_2}\boldsymbol{w}_i^{t,r}\end{aligned} \tag{5}$$

$$\boldsymbol{w}_i^{t,r+1} = \boldsymbol{w}_i^{t,r} - \eta_1\lambda_2(\boldsymbol{w}_i^{t,r} - \boldsymbol{y}_i^{t,r+1}) \tag{6}$$

where η_1 is the step size.

- After performing the edge updates for R times, each edge server transmits its edge model $\boldsymbol{w}_i^{t,R}$ to the cloud server. Then, at the cloud server, the global model update is performed as

$$\boldsymbol{w}^{t+1} = (1-\beta)\boldsymbol{w}^t + \frac{\beta}{N}\sum_{i=1}^{N}\boldsymbol{w}_i^{t,R} \tag{7}$$

where $\beta \in [0,1]$ is the global update coefficient [16]. Subsequently, the cloud server broadcasts new global model $\boldsymbol{w}^{t+1}$ to each edge server to initiate the $(t+1)$-th communication round.

To solve problem (3), we uniformly sample a mini-batch of dataset $\mathcal{D}_{i,j}$ to compute a stochastic gradient as

$$\nabla\tilde{\ell}_{i,j}(\boldsymbol{\theta}_{i,j}, \mathcal{D}_{i,j}) = \frac{1}{|\mathcal{D}_{i,j}|}\sum_{\xi_{i,j}\in\mathcal{D}_{i,j}}\nabla\ell_{i,j}(\boldsymbol{\theta}_{i,j}, \xi_{i,j}) \tag{8}$$

where $\mathbb{E}[\nabla\tilde{\ell}_{i,j}(\boldsymbol{\theta}_{i,j}, \mathcal{D}_{i,j})] = \nabla\ell_{i,j}(\boldsymbol{\theta}_{i,j})$, $|\mathcal{D}_{i,j}|$ denotes the cardinality of $\mathcal{D}_{i,j}$, and $\xi_{i,j}$ is the training data randomly sampled from $\mathcal{D}_{i,j}$. Since closed-form $\boldsymbol{\theta}_{i,j}$ cannot be straightforward obtained, we obtain an approximate $\tilde{\boldsymbol{\theta}}_{i,j}$ by minimizing

$$h(\boldsymbol{\theta}_{i,j}; \boldsymbol{y}_{i,j}^{t,r}, \mathcal{D}_{i,j}) = \tilde{\ell}_{i,j}(\boldsymbol{\theta}_{i,j}, \mathcal{D}_{i,j}) + \frac{\lambda_1}{2}\|\boldsymbol{\theta}_{i,j} - \boldsymbol{y}_{i,j}^{t,r}\|_2^2 \tag{9}$$

which can be solved by Nesterov's accelerated gradient descent. The iteration goes until satisfying

$$\|\nabla h(\tilde{\boldsymbol{\theta}}_{i,j}; \boldsymbol{y}_{i,j}^{t,r}, \mathcal{D}_{i,j})\|_2^2 \le \nu \tag{10}$$

where $\nu > 0$ is the pre-determined accuracy level.

B. Model Aggregation via AirComp

To realize spectral-efficient model aggregation between the edge server and its associated devices, we adopt AirComp, where all devices in the same cluster concurrently transmit their locally-trained models to their associated edge server. We assume that each cluster occupies an orthogonal channel of the same bandwidth and the communication between each edge server and the cloud server is error-free as in [18]. We normalize D-dimensional local model $\boldsymbol{\theta}_{i,j}^{t,r}$ before the uplink transmission to facilitate the power control. In particular, device $c_{i,j}$ calculates mean $\bar{\theta}_{i,j}^{t,r}$ and variance $(\pi_{i,j}^{t,r})^2$ of $\boldsymbol{\theta}_{i,j}^{t,r}$, $\forall\, i,j$, as follows

$$\bar{\theta}_{i,j}^{t,r} = \frac{1}{D}\sum_{d=1}^{D}\theta_{i,j,d}^{t,r},\ (\pi_{i,j}^{t,r})^2 = \frac{1}{D}\sum_{d=1}^{D}(\theta_{i,j,d}^{t,r} - \bar{\theta}_{i,j}^{t,r})^2 \tag{11}$$

where $\theta_{i,j,d}^{t,r}$ denotes the d-th element of $\boldsymbol{\theta}_{i,j}^{t,r}$. By setting $\bar{\theta}_i^{t,r} = \frac{1}{M}\sum_{j=1}^{M}\bar{\theta}_{i,j}^{t,r}$ and $(\pi_i^{t,r})^2 = \frac{1}{M}\sum_{j=1}^{M}(\pi_{i,j}^{t,r})^2$, $\boldsymbol{\theta}_{i,j}^{t,r}$ can be normalized as

$$\boldsymbol{s}_{i,j}^{t,r} = \frac{\boldsymbol{\theta}_{i,j}^{t,r} - \bar{\theta}_i^{t,r}}{\pi_i^{t,r}},\ \forall\, i,j. \tag{12}$$

We assume that $\{\boldsymbol{s}_{i,j}^{t,r}\}_{j=1}^M$ are independent and have zero mean and unit variance, i.e., $\mathbb{E}[\boldsymbol{s}_{i,j}^{t,r}(\boldsymbol{s}_{i,j}^{t,r})^H] = \boldsymbol{I}_D$ and $\mathbb{E}[\boldsymbol{s}_{i,j}^{t,r}(\boldsymbol{s}_{i,k}^{t,r})^H] = \boldsymbol{0}$, $\forall\, j \ne k$. All devices in the i-th cluster transmit their normalized local models $\{\boldsymbol{s}_{i,j}^{t,r}\}_{j=1}^M$ to the i-th edge server simultaneously. We consider that each device owns a single antenna and each edge server is equipped with K antennas. In the r-th edge iteration from the t-th global communication round (denoted as (t,r)), the channel coefficient between device $c_{i,j}$ and the i-th edge server is denoted as $\boldsymbol{h}_{i,j}^{t,r} \in \mathbb{C}^K$, and assume that the channel follows block fading. By denoting $w_{i,j}^{t,r} \in \mathbb{C}$ as the transmit scalar of device $c_{i,j}$, the

aggregation of local models received at the i-th edge server is

$$\tilde{\boldsymbol{q}}_{i,d}^{t,r+1} = \sum_{j=1}^{M} \boldsymbol{h}_{i,j}^{t,r} w_{i,j}^{t,r} s_{i,j,d}^{t,r} + \boldsymbol{n}_i^{t,r} \tag{13}$$

where $\boldsymbol{n}_i \sim \mathcal{CN}(0, \sigma_0^2 \boldsymbol{I}_K)$ is the additive white Gaussian noise (AWGN). In practice, each device has a maximum transmit power constant, i.e., $|w_{i,j}^{t,r}|^2 \leq P, \forall\, i, j$. The estimated function of the received signal can be computed as

$$\begin{aligned} \check{q}_{i,d}^{t,r+1} &= \frac{1}{\sqrt{\eta}} \boldsymbol{m}_i^H \tilde{\boldsymbol{q}}_{i,d}^{t,r+1} \\ &= \frac{1}{\sqrt{\eta}} \boldsymbol{m}_i^H \sum_{j=1}^{M} \boldsymbol{h}_{i,j}^{t,r} w_{i,j}^{t,r} s_{i,j,d}^{t,r} + \frac{1}{\sqrt{\eta}} \boldsymbol{m}_i^H \boldsymbol{n}_i^{t,r} \end{aligned} \tag{14}$$

where $\boldsymbol{m} \in \mathbb{C}^K$ and η are the receive beamforming vector and denoising factor at the edge server, respectively. After denormalization, we obtain

$$\begin{aligned} \check{\boldsymbol{\theta}}_i^{t,r+1} &= \frac{1}{M}(\pi_i^{t,r} \check{\boldsymbol{q}}_i^{t,r+1} + M \bar{\theta}_i^{t,r}) \\ &= \frac{1}{M} \pi_i^{t,r} (\check{\boldsymbol{q}}_i^{t,r+1} - \boldsymbol{q}_i^{t,r+1}) + \boldsymbol{\theta}_i^{t,r+1} \end{aligned} \tag{15}$$

where $\boldsymbol{\theta}_i^{t,r+1} = \frac{1}{M} \sum_{j=1}^{M} \frac{\lambda_1}{\lambda_1+\lambda_2} \boldsymbol{\theta}_{i,j}^{t,r}$ and $\boldsymbol{q}_i^{t,r+1} = \sum_{j=1}^{M} s_{i,j}^{t,r}$. Thus, the updated edge model can be expressed as

$$\check{\boldsymbol{y}}_i^{t,r+1} = \frac{\lambda_1}{\lambda_1+\lambda_2} \check{\boldsymbol{\theta}}_i^{t,r+1} + \frac{\lambda_2}{\lambda_1+\lambda_2} \boldsymbol{w}_i^{t,r}. \tag{16}$$

Afterwards, the edge model update (6) can be rewritten as

$$\begin{aligned} \boldsymbol{w}_i^{t,r+1} &= \boldsymbol{w}_i^{t,r} - \eta_1 \lambda_2 (\boldsymbol{w}_i^{t,r} - \check{\boldsymbol{y}}_i^{t,r+1}) \\ &= \boldsymbol{w}_i^{t,r} - \eta_1 \underbrace{\frac{\lambda_1}{\lambda_1+\lambda_2} \lambda_2 (\boldsymbol{w}_i^{t,r} - \check{\boldsymbol{\theta}}_i^{t,r+1})}_{\check{\boldsymbol{g}}_i^{t,r}}. \end{aligned} \tag{17}$$

By accumulating over R iterates, we have

$$\eta_1 \sum_{r=0}^{R-1} \check{\boldsymbol{g}}_i^{t,r} = \sum_{r=0}^{R-1} (\boldsymbol{w}_i^{t,r} - \boldsymbol{w}_i^{t,r+1}) = \boldsymbol{w}^t - \boldsymbol{w}_i^{t,R} \tag{18}$$

where $\check{\boldsymbol{g}}_i^{t,r}$ is a biased estimate of $\nabla F_i(\boldsymbol{w}_i^{t,r})$. Then, we rewrite the global model update (7) as

$$\begin{aligned} \boldsymbol{w}^{t+1} &= \boldsymbol{w}^t - \frac{\beta}{N} \sum_{i=1}^{N} (\boldsymbol{w}^t - \boldsymbol{w}_i^{t,R}) \\ &= \boldsymbol{w}^t - \underbrace{\eta_1 \beta R}_{\tilde{\eta}} \underbrace{\frac{1}{NR} \sum_{i=1}^{N} \sum_{r=0}^{R-1} \check{\boldsymbol{g}}_i^{t,r}}_{\check{\boldsymbol{g}}^t} \end{aligned} \tag{19}$$

where $\tilde{\eta}$ and $\check{\boldsymbol{g}}^t$ can be regarded as the customized step size and approximate stochastic gradient of the global update, respectively. Note that $\check{\boldsymbol{g}}^t$ contains the aggregation error introduced by wireless communication over the fading channel, which hinders the convergence of HPFL. For the ideal case without communication error, the biased estimate of $\nabla F_i(w_i^{t,r})$ becomes

$$\begin{aligned} \boldsymbol{g}^t &= \frac{1}{NR} \sum_{i=1}^{N} \sum_{r=0}^{R-1} \boldsymbol{g}_i^{t,r} \\ &= \frac{1}{NR} \sum_{i=1}^{N} \sum_{r=0}^{R-1} \frac{\lambda_1 \lambda_2}{\lambda_1+\lambda_2} \left(\boldsymbol{w}_i^{t,r} - \boldsymbol{\theta}_i^{t,r+1} \right). \end{aligned} \tag{20}$$

Hence, the gradient aggregation error can be calculated as $\boldsymbol{e}^t = \check{\boldsymbol{g}}^t - \boldsymbol{g}^t$.

III. Convergence Analysis and Problem Formulation

We provide the convergence analysis of the AirComp-assisted HPFL framework and formulate an optimization problem to minimize the transmission distortion in the convergence upper bound.

A. Assumptions

Assumption 1 (Smoothness)**.** *We assume that $\ell(\boldsymbol{w})$ is L-smooth on $\mathbb{R}^D$, i.e.,*

$$\|\nabla \ell(\boldsymbol{w}) - \nabla \ell(\boldsymbol{w}')\|_2 \leq L \|\boldsymbol{w} - \boldsymbol{w}'\|_2. \tag{21}$$

Assumption 2 (Bounded Variance)**.** *The stochastic gradient in each device has a bounded variance, i.e.,*

$$\mathbb{E}_{\xi_{i,j}} \left[\left\| \nabla \tilde{\ell}_{i,j}(\boldsymbol{w}; \xi_{i,j}) - \nabla \ell_{i,j}(\boldsymbol{w}) \right\| \right]^2 \leq \gamma_l^2. \tag{22}$$

Assumption 3 (Bounded Diversity)**.** *The dissimilarity between the local loss function and global loss funtion can be constrained as*

$$\frac{1}{NM} \sum_{i=1}^{N} \sum_{j=1}^{M} \|\nabla \ell_{i,j}(\boldsymbol{w}) - \nabla \ell(\boldsymbol{w})\|^2 \leq \sigma_l^2. \tag{23}$$

Assumption 4 (Local Model Variance Bound)**.** *The variance of D elements of $\boldsymbol{\theta}_{i,j}$ has the constant upper bound $\Phi \geq 0$, i.e., $\pi_{i,j}^2 \leq \Phi$.*

Remark 1. *Assumptions 1, 2, and 3 are commonly made in the FL convergence analysis [19]. In Assumption 4, we assume that $\pi_{i,j}^2$ has a non-negative upper bound as in [12] since the value of elements in local model $\boldsymbol{\theta}_{i,j}$ are finite.*

B. Convergence Analysis

Theorem 1. *Under Assumptions 1, 2, and 3, if $\lambda \leq \sqrt{16(L^2+1)}$, $\tilde{\eta} \leq \min\left\{ \frac{\beta}{2L_F}, \frac{1}{2(4L_F+192\lambda^2-1)} \right\}$, and $\lambda_2 > 4L$, the time-averaged norm of global gradients after T communication rounds has a upper bound, i.e.,*

$$\begin{aligned} &\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left[\|\nabla F(\boldsymbol{w}^t)\|_2^2 \right] \\ &\leq 4 \left(\frac{\Delta_F}{\tilde{\eta} T} + A_1 + \tilde{\eta} A_2 \right) + \frac{1}{T} \sum_{t=0}^{T-1} \frac{5}{2} \mathbb{E}\left[\|\boldsymbol{e}^t\|^2 \right] \end{aligned} \tag{24}$$

where $L_F = \lambda_2$, $\Delta_F = \mathbb{E}[F(\boldsymbol{w}^0) - F(\boldsymbol{w}^*)]$, $A_1 = 4\beta\bar{\lambda}^2\delta^2$, $A_2 = \frac{32 L_F (3R\sigma_F^2 + 2\bar{\lambda}^2\delta^2)}{R}$, $\sigma_F^2 \triangleq \frac{\lambda^2}{\lambda^2 - 16L^2} \sigma_l^2$, $\lambda = \frac{\lambda_1\lambda_2}{\sqrt{\lambda_1^2+\lambda_2^2}}$, and $\bar{\lambda} = \frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2}$. The last term is denoted as MSE. With Assumption 4, we have

$$\mathbb{E}\left[|e_d^t|^2 \right]$$

$$= \frac{1}{NRM^2} \sum_{i=1}^{N} \sum_{r=0}^{R-1} \times$$
$$\mathbb{E}\left[(\pi_i^{t,r})^2 \left| \sum_{j=1}^{M} \left(1 - \frac{\boldsymbol{m}_i^H \boldsymbol{h}_{i,j}^{t,r} w_{i,j}^{t,r}}{\sqrt{\eta}} \right) s_{i,j,d}^{t,r} - \frac{\boldsymbol{m}_i^H \boldsymbol{n}_i^{t,r}}{\sqrt{\eta}} \right|^2 \right]$$
$$\leq \frac{\Phi}{NRM^2} \sum_{i=1}^{N} \sum_{r=0}^{R-1} \times$$
$$\underbrace{\left(\sum_{j=1}^{M} \left| \frac{\boldsymbol{m}_i^H \boldsymbol{h}_{i,j}^{t,r} w_{i,j}^{t,r}}{\sqrt{\eta}} - 1 \right|^2 + \frac{\sigma_0^2 \|\boldsymbol{m}_i\|^2}{\eta} \right)^2}_{\mathrm{MSE}_{i,d}^{t,r}}. \quad (25)$$

The proof follows a similar idea to that of Theorem 3 in [19] and is omitted due to space limitation.

Remark 2. *The three terms in the parenthesis of (24) are caused by the initial optimality gap, the random mini-batch sampling, and the model diversity with respect to the mini-batch sampling and the edge server drift error, respectively, while the last term of (24) is the time-averaged MSE due to channel fading and receiver noise. As T goes to infinity, the initial optimality gap is decreased to 0. With given R, T, β, η_1, λ_1, and λ_2, the first three terms of (24) are constant according to the assumptions. Therefore, the convergence of HPFL is mainly hampered by the time-average MSE given in (25), which needs to be minimized to improve the learning performance.*

C. Problem Formulation

We omit the constant terms in (25) and rewrite the MSE in communication round (t, r) as

$$\mathrm{MSE}_i^{t,r} = \sum_{j=1}^{M} \left| \frac{\boldsymbol{m}_i^H \boldsymbol{h}_{i,j} w_{i,j}}{\sqrt{\eta}} - 1 \right|^2 + \frac{\sigma_0^2 \|\boldsymbol{m}_i\|^2}{\eta}. \quad (26)$$

Given receive beamforming vector $\boldsymbol{m}_i$, we can express the optimal transmit scalars as

$$w_{i,j} = \sqrt{\eta} \frac{(\boldsymbol{m}_i^H \boldsymbol{h}_{i,j})^H}{\|\boldsymbol{m}_i^H \boldsymbol{h}_{i,j}\|_2^2}, \forall j \quad (27)$$

where η is given by

$$\eta = P \min_j \|\boldsymbol{m}_i^H \boldsymbol{h}_{i,j}\|_2^2 \quad (28)$$

due to the transmit power constraint. For simplicity, we omit notation (t, r) in the following. With (27) and (28), the MSE can be further rewritten as

$$\mathrm{MSE}_i = \frac{\sigma_0^2 \|\boldsymbol{m}_i\|_2^2}{\eta} = \frac{\sigma_0^2 \|\boldsymbol{m}_i\|_2^2}{P \min_j \|\boldsymbol{m}_i^H \boldsymbol{h}_{i,j}\|_2^2}. \quad (29)$$

In order to minimize MSE, we optimize the receive beamforming vector $\boldsymbol{m}_i$ as

$$\min_{\boldsymbol{m}_i} \frac{\sigma_0^2 \|\boldsymbol{m}_i\|_2^2}{P \min_j \|\boldsymbol{m}_i^H \boldsymbol{h}_{i,j}\|_2^2} \quad (30)$$

which can be equivalently transformed into (31) [20]

$$\mathcal{P}_1 : \min_{\boldsymbol{m}_i} \|\boldsymbol{m}_i\|_2^2 \quad (31a)$$
$$\text{s.t. } \|\boldsymbol{m}_i^H \boldsymbol{h}_{i,j}\|_2^2 \geq 1, \forall j. \quad (31b)$$

To solve non-convex problem $\mathcal{P}_1$, the authors in [21] proposed a semidefinite relaxation (SDR) based algorithm, which achieves poor performance in the scenario with a large-scale antenna array at the BS. Besides, the optimal solution proposed in [22] requires a high computation complexity and makes it unsuitable for practical scenarios. As problem $\mathcal{P}_1$ is independent in each cluster, we denote $\boldsymbol{m}_i$ as $\boldsymbol{m}$ for convenience hereafter.

IV. Efficient SCA-Based Algorithm

An efficient algorithm is developed to optimize the transceiver design by leveraging SCA and Lagrangian duality in this section.

A. Problem Transformation

We denote $\boldsymbol{z} \in \mathbb{C}^K$ as an auxiliary vector and $\mathcal{M}$ as the device set. For matrix $\mathbf{A} \succeq \mathbf{0}$, we have $(\boldsymbol{m}-\boldsymbol{z})^H \mathbf{A} (\boldsymbol{m}-\boldsymbol{z}) \geq 0$, $\forall \boldsymbol{z}$. It follows that $\boldsymbol{m}^H \mathbf{A} \boldsymbol{m} \geq 2\mathfrak{Re}\{\boldsymbol{m}^H \mathbf{A} \boldsymbol{z}\} - \boldsymbol{z}^H \mathbf{A} \boldsymbol{z}$. With a given $\boldsymbol{z}$ by applying the inequality above to (31), we obtain the following optimization problem

$$\mathcal{P}_{\mathrm{SCA}}(\boldsymbol{z}) : \min_{\boldsymbol{m}} \|\boldsymbol{m}\|_2^2 \quad (32a)$$
$$\text{s.t. } -2\mathfrak{Re}\{\boldsymbol{m}^H \boldsymbol{h}_j \boldsymbol{h}_j^H \boldsymbol{z}\} + \|\boldsymbol{z}^H \boldsymbol{h}_j\|_2^2 \leq -1, \forall j \quad (32b)$$

which is a convex approximation of (31). Therefore, its optimal solution can be obtained from its Lagrangian dual domain. Particularly, the Lagrangian function of $\mathcal{P}_{\mathrm{SCA}}(\boldsymbol{z})$ is

$$\mathcal{L}(\boldsymbol{z}, \boldsymbol{m}, \boldsymbol{\lambda}) = \|\boldsymbol{m}\|_2^2 + \sum_{j=1}^{M} \lambda_j \left(-2\mathfrak{Re}\left\{ \boldsymbol{m}^H \boldsymbol{h}_j \boldsymbol{h}_j^H \boldsymbol{z} \right\} + \|\boldsymbol{z}^H \boldsymbol{h}_j\|_2^2 + 1 \right) \quad (33)$$

where $\boldsymbol{\lambda} = [\lambda_1, \lambda_2 \ldots, \lambda_M]^T$ is the Lagrange multiplier. The Lagrange dual problem for $\mathcal{P}_{\mathrm{SCA}}(\boldsymbol{z})$ is given by

$$\mathcal{D}_{\mathrm{SCA}}(\boldsymbol{z}) : \max_{\boldsymbol{\lambda}} \ g(\boldsymbol{\lambda}, \boldsymbol{z}) \quad \text{s.t. } \boldsymbol{\lambda} \succeq \mathbf{0} \quad (34)$$

where $g(\boldsymbol{\lambda}, \boldsymbol{z}) = \min_{\boldsymbol{m}} \mathcal{L}(\boldsymbol{z}, \boldsymbol{m}, \boldsymbol{\lambda})$. We rewrite (33) as

$$\mathcal{L}(\boldsymbol{z}, \boldsymbol{m}, \boldsymbol{\lambda}) = \sum_{j=1}^{M} \lambda_j (\|\boldsymbol{z}^H \boldsymbol{h}_j\|_2^2 + 1) + \|\boldsymbol{m}\|_2^2 - 2\mathfrak{Re}\left\{ \boldsymbol{z}^H \left(\sum_{j=1}^{M} \lambda_j \boldsymbol{h}_j \boldsymbol{h}_j^H \right) \boldsymbol{m} \right\}. \quad (35)$$

By defining $\boldsymbol{\nu} = \left(\sum_{j=1}^{M} \lambda_j \boldsymbol{h}_j \boldsymbol{h}_j^H \right) \boldsymbol{z}$, $g(\boldsymbol{\lambda}, \boldsymbol{z})$ can be written as

$$\min_{\boldsymbol{m}} \|\boldsymbol{m}\|_2^2 - 2\mathfrak{Re}\{\boldsymbol{\nu}^H \boldsymbol{m}\} \quad (36)$$

whose optimal solution can be derived by using KKT conditions. Specifically, by defining $\boldsymbol{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_M]^T$ as the channel matrix and $\boldsymbol{R}(\boldsymbol{\lambda}) = \boldsymbol{I} + \sum_{j=1}^{M} \lambda_j \boldsymbol{h}_j \boldsymbol{h}_j^H$, we have

$$\boldsymbol{m}^{\mathrm{opt}} = \boldsymbol{R}^{-1}(\boldsymbol{\lambda}^{\mathrm{opt}}) \sum_{j=1}^{M} a_j^{\mathrm{opt}} \boldsymbol{h}_j = \boldsymbol{R}^{-1}(\boldsymbol{\lambda}^{\mathrm{opt}}) \boldsymbol{H} \boldsymbol{a}^{\mathrm{opt}} \quad (37)$$

Algorithm 1: Computing $\boldsymbol{\lambda}$

Input: Channel gain vector $\boldsymbol{h}_j$, $j \in \mathcal{M}$.

Initialize $\boldsymbol{\lambda}^0$ and $\ell = 1$.
repeat
  for each $j \in \mathcal{M}$ **do**
    $\lambda_j^\ell \leftarrow \frac{1}{2\boldsymbol{h}_j^H \boldsymbol{R}^{-1}(\boldsymbol{\lambda}^{\ell-1})\boldsymbol{h}_j}$;
  end for
  $\ell \leftarrow \ell + 1$;
until $\ell = 100$.

where $\boldsymbol{\lambda}^{\text{opt}}$ is the optimal dual solution for $\mathcal{D}_{\text{SCA}}(\boldsymbol{m}^{\text{opt}})$, $a_j^{\text{opt}} = 2\lambda_j^{\text{opt}}\delta_j$ with $\delta_j = \boldsymbol{h}_j^H \boldsymbol{m}^{\text{opt}}$, $\forall j$, and $\boldsymbol{a}^{\text{opt}} = [a_1^{\text{opt}}, a_2^{\text{opt}}, \ldots, a_M^{\text{opt}}]^T$. The proof is similar to Theorem 1 in [23]. To get $\boldsymbol{m}^{\text{opt}}$, we need to compute $\boldsymbol{\lambda}^{\text{opt}}$ and $\boldsymbol{a}^{\text{opt}}$ numerically. It's worth noting that computing $\boldsymbol{\lambda}^{\text{opt}}$ and $\boldsymbol{a}^{\text{opt}}$ is challenging because the original problem (31) is NP-hard. We present the numerical algorithm in the following subsections.

B. Algorithm for Computing Lagrange Multiplier $\boldsymbol{\lambda}$

By defining $\boldsymbol{D}_\lambda = \text{diag}(\boldsymbol{\lambda})$ and $\boldsymbol{\delta} = [\delta_1, \delta_2, \ldots, \delta_M]^T$, we have $\boldsymbol{a} = 2\boldsymbol{D}_\lambda \boldsymbol{\delta}$. According to (37), we have

$$\delta_j = \boldsymbol{h}_j^H \boldsymbol{m}^{\text{opt}} = 2\boldsymbol{h}_j^H \boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{H} \boldsymbol{D}_\lambda \boldsymbol{\delta}. \tag{38}$$

For any $j \in \mathcal{M}$, it follows that $\boldsymbol{\delta} = 2\boldsymbol{H}^H \boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{H} \boldsymbol{D}_\lambda \boldsymbol{\delta}$, which can be transformed to

$$\left(2\boldsymbol{H}^H \boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{H} \boldsymbol{D}_\lambda - \boldsymbol{I}\right) \boldsymbol{\delta} = \boldsymbol{0}. \tag{39}$$

Since $\boldsymbol{\delta}$ is unknown, it's difficult to directly solve (39). Hence, we develop a computation-efficient algorithm to calculate a suboptimal $\boldsymbol{\lambda}$ below.

A sufficient condition that satisfies (39) is

$$2\boldsymbol{H}^H \boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{H} \boldsymbol{D}_\lambda = \boldsymbol{I} \tag{40}$$

which is equal to

$$\begin{cases} 2\lambda_j \boldsymbol{h}_j^H \boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{h}_j = 1, & j \in \mathcal{M} \\ 2\lambda_j \boldsymbol{h}_j^H \boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{h}_{j'} = 1, & j' \neq j, \forall\, j' \in \mathcal{M}. \end{cases} \tag{41}$$

Since the above conditions may not satisfy all λ_j, we only solve the first equation in (41) to obtain $\boldsymbol{\lambda}$. The solution can be obtained by the iterative method given in Algorithm 1.

C. Algorithm for Computing Weight Vector $\boldsymbol{a}$

According to (37), (31) can be transformed as follows

$$\mathcal{P}_2 : \min_{\boldsymbol{a}} \ \|\boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{H} \boldsymbol{a}\|^2 \tag{42a}$$

$$\text{s.t. } |\boldsymbol{a}^H \boldsymbol{H}^H \boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{h}_j|^2 \geq 1, \forall\, j \in \mathcal{M}. \tag{42b}$$

With such a transformation, the problem dimension is reduced from K to M. Then, SCA is adopted to calculate $\boldsymbol{a}$ for $\mathcal{P}_2$.

By denoting $\boldsymbol{G} = \boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{H}$ and $\boldsymbol{f}_j = \boldsymbol{G}^H \boldsymbol{h}_j$, $\forall\, j \in \mathcal{M}$, similar to solving problem (32), we use an auxiliary variable $\boldsymbol{v}$ and apply the convex approximation to constraint (42) in $\mathcal{P}_2$ as follows

$$\mathcal{P}_{\text{SCA1}}(\boldsymbol{v}) : \min_{\boldsymbol{a}} \ \|\boldsymbol{G}\boldsymbol{a}\|^2 \tag{43a}$$

$$\text{s.t. } -4\Re\mathfrak{e}\left\{\boldsymbol{a}^H \boldsymbol{f}_j \boldsymbol{f}_j^H \boldsymbol{v}\right\} + 2 \cdot |\boldsymbol{v}^H \boldsymbol{f}_j|^2 \leq -1, \forall\, j. \tag{43b}$$

Algorithm 2: SCA-based algorithm for problem (31)

Input: Channel gain vector $\boldsymbol{h}_j$, $j \in \mathcal{M}$.
Output: The optimal receive beamforming vector $\boldsymbol{m}^{\text{opt}}$.

Initialize $\boldsymbol{v}^0$, $\ell = 0$, and convergence tolerance ϵ_2.
repeat
  Compute $\boldsymbol{\lambda}$ according to Algorithm 1;
  Solve problem (43) and obtain the optimal solution $\boldsymbol{a}^{\text{opt}}(\boldsymbol{v}^\ell)$;
  $\boldsymbol{v}^{\ell+1} \leftarrow \boldsymbol{a}^{\text{opt}}(\boldsymbol{v}^\ell)$;
  $\ell \leftarrow \ell + 1$;
until $\|\boldsymbol{v}^\ell - \boldsymbol{v}^{\ell-1}\|^2 \leq \epsilon_2$;
$\boldsymbol{m}^{\text{opt}} \leftarrow \boldsymbol{R}^{-1}(\boldsymbol{\lambda}) \boldsymbol{H} \boldsymbol{a}^\ell$.

Subsequently, $\mathcal{P}_2$ can be solved by iteratively solving $\mathcal{P}_{\text{SCA1}}(\boldsymbol{v})$ and updating $\boldsymbol{v}$ with solution $\boldsymbol{a}^{\text{opt}}(\boldsymbol{v})$ to $\mathcal{P}_{\text{SCA1}}(\boldsymbol{v})$ until convergence.

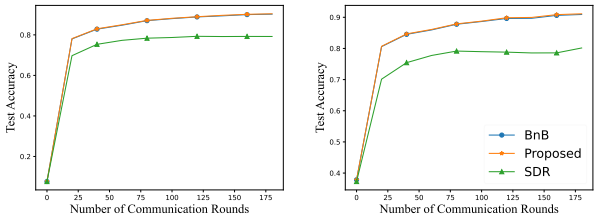
Note that solving (43) has a complexity of $\mathcal{O}(M^3)$ in each iteration, which is lower than complexity $\mathcal{O}(K^3 M^{3.5})$ [22]. The developed SCA-based algorithm is summarized in Algorithm 2.

V. Simulation Results

A. Experiment Setting

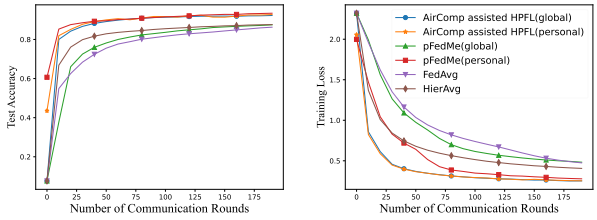
In this experiment, we consider a three-dimensional actual scene consisting of a central server and two edge servers (i.e. $N = 2$), which are all located at $(0, 0, 20)$ meters. Each edge server manages 10 devices (i.e. $M = 10$), and they are uniformly distributed within the circles with centers of $(120, 20, 0)$ and $(-120, 20, 0)$ meters, respectively. Both circles have a radius of 20 meters. We evaluate our method based on MNIST dataset and split it so that each device only contains 2 kinds of labels out of 10. We suppose the large-scale fading to be modeled as $T_0(\frac{u}{u_0})^{-\alpha}$, where u represents the distance from the receiver to the transmitter, α denotes the path loss exponent, and T_0 denotes the path loss when $u_0 = 1$ meter. Besides, Rician fading with racian factor γ is used to model the small-scale fading. Setting $\alpha = 3$, $T_0 = -30$ dB, $P = 30$ dBm, $\gamma = 3$, and $\sigma_0^2 = -100$ dBm. As for the algorithm, we set the iteration number to $T_1 = 100$ when computing $\boldsymbol{\lambda}$ and the SCA convergence accuracy factor to $\epsilon_2 = 10^{-4}$. Deep neural network (DNN) with two hidden layers is used and we set $3/4$ of each user's dataset as the training set and the others are for testing. The hyperparameters are set as follows: $\eta_1 = 0.05$, $\beta = 1$, $\lambda_1 = \lambda_2 = 20$, $R = 20$, $T = 200$, and the batch size is 20.

When testing the performance of different optimization algorithms and different number of antennas, the SDR algorithm and the BnB algorithm proposed by [22] are used as the baseline algorithms, and we set the number of antennas as $K = 40$. Then, we also compare our proposed framework with the following three baseline frameworks: FedAvg, HierFedAvg [3], pFedMe [16]. To balance the setting between one-layer FL system and hierarchical FL system, we set the device number to 20 in FedAvg and pFedMe, and other hyperparameters remain the same.



(a) Global model (b) Personalized model

Fig. 2. Learning performance under different optimization algorithms.



(a) Test Accuracy (b) Training Loss

Fig. 3. Learning performance under different FL frameworks.

B. Performance Comparison

In Fig. 2, the performance of both global model and personalized model is compared when using different optimization algorithms in our AirComp-assisted HPFL system. It can be seen that our developed SCA-based algorithm has a considerable performance improvement compared with the SDR algorithm and a nearly invisible gap compared with optimal BnB algorithm. It demonstrates that our proposed SCA-based algorithm can maintain near-optimal performance while greatly reduce the computation cost compared with the optimal BnB algorithm.

Fig. 3 compares the proposed AirComp-assisted HPFL with other three baselines, i.e., FedAvg, HierFedAvg, and pFedMe. For pFedMe and our proposed AirComp-assisted HPFL system, we consider the performance of both the global model and personalized model at the same time. For all schemes, we use the proposed algorithm to complete the receive beamforming design, and set the number of antennas to $K = 10$. It is observed that both the global and personalized models of AirComp-assisted HPFL converge to a higher test accuracy than FedAvg and HierFedAvg due to our L_2-norm regularized loss function design. We can also see that, although the personalized model in pFedMe converges to a rarely high test accuracy and is close to the proposed AirComp-assisted HPFL, the test accuracy of its global model still has a big gap compared to our proposed system, due to the fact that the L_2-norm regularization term encourages edge servers and devices to seek their personalized models, in the meanwhile not deviate too much from the global model.

VI. Conclusion

In this paper, we proposed an AirComp-assisted HPFL framework. We analyzed the convergence and formulate an optimization problem to minimize transmission distortion. To this end, an efficient algorithm that leverages SCA and Lagrangian duality is developed to optimize the transceiver design, which achieves a near-optimal performance and a much greater test accuracy than the baseline algorithms.

References

[1] K. Yang, Y. Shi, Y. Zhou, Z. Yang, L. Fu, and W. Chen, "Federated machine learning for intelligent iot via reconfigurable intelligent surface," *IEEE Network*, vol. 34, no. 5, pp. 16–22, 2020.

[2] Y. Yang, M. Ma, H. Wu, Q. Yu, P. Zhang, X. You, J. Wu, C. Peng, T.-S. P. Yum, S. Shen *et al.*, "6g network ai architecture for everyone-centric customized services," *IEEE Network*, to appear.

[3] L. Liu, J. Zhang, S. Song, and K. B. Letaief, "Client-edge-cloud hierarchical federated learning," in *Proc. IEEE Int. Conf. on Comm. (ICC)*, Jun. 2020.

[4] 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. (ICASSP)*, May 2020.

[5] 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.

[6] S. Liu, G. Yu, X. Chen, and M. Bennis, "Joint user association and resource allocation for wireless hierarchical federated learning with IID and non-IID data," *IEEE Trans. Wireless Commun.*, no. 10, pp. 7852–7866, Oct. 2022.

[7] B. Nazer and M. Gastpar, "Computation over multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3498–3516, Oct. 2007.

[8] Z. Wang, Y. Zhao, Y. Zhou, Y. Shi, C. Jiang, and K. B. Letaief, "Over-the-air computation: Foundations, technologies, and applications," 2022. [Online]. Available: https://arxiv.org/abs/2210.10524

[9] 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.

[10] 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. 2019.

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

[12] Z. Wang, J. Qiu, Y. Zhou, Y. Shi, L. Fu, W. Chen, and K. B. Letaief, "Federated learning via intelligent reflecting surface," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 808–822, Feb. 2021.

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

[14] 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, 2022.

[15] H. U. Sami and B. Güler, "Over-the-air personalized federated learning," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2022.

[16] C. T Dinh, N. Tran, and J. Nguyen, "Personalized federated learning with moreau envelopes," *Proc. Neural Inf. Process. Syst. (NeurIPS)*, Dec. 2020.

[17] C. You, D. Feng, K. Guo, H. H. Yang, C. Feng, and T. Q. Quek, "Semi-synchronous personalized federated learning over mobile edge networks," *IEEE Trans. Wireless Commun.*, 2022, early access.

[18] O. Aygün, M. Kazemi, D. Gündüz, and T. M. Duman, "Hierarchical over-the-air federated edge learning," in *Proc. IEEE Int. Conf. on Comm. (ICC)*, May 2022.

[19] X. Liu, Y. Li, Y. Shao, and Q. Wang, "Sparse federated learning with hierarchical personalization models," 2022. [Online]. Available: https://arxiv.org/abs/2203.13517

[20] 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.

[21] 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.

[22] W. Fang, Y. Zou, H. Zhu, Y. Shi, and Y. Zhou, "Optimal receive beamforming for over-the-air computation," in *Proc. IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Sep. 2021.

[23] M. Dong and Q. Wang, "Multi-group multicast beamforming: Optimal structure and efficient algorithms," *IEEE Trans. Signal Process.*, vol. 68, pp. 3738–3753, May 2020.