

# UAV-Assisted Hierarchical Aggregation for Over-the-Air Federated Learning

Xiangyu Zhong, Xiaojun Yuan, *Senior Member, IEEE*, Huiyuan Yang, Chenxi Zhong

Yangtze Delta Region Institute, University of Electronic Science and Technology of China, Huzhou, China

National Key Lab of Sci. and Tech. on Communications, University of Electronic Sci. and Tech. of China, Chengdu, China

Emails: xyzh@std.uestc.edu.cn, xjyuan@uestc.edu.cn, {hyyang, cxzhong}@std.uestc.edu.cn

**Abstract**—With huge amounts of data explosively increasing on the mobile edge, over-the-air federated learning (OA-FL) emerges as a promising technique to reduce communication costs and privacy leak risks. However, when devices in a relatively large area cooperatively train a machine learning model, the attendant straggler issue will significantly reduce the learning performance. In this paper, we propose an unmanned aerial vehicle (UAV) assisted OA-FL system, where the UAV acts as a parameter server (PS) to aggregate the local gradients hierarchically for global model updating. Under this UAV-assisted hierarchical aggregation scheme, we carry out a gradient-correlation-aware FL performance analysis. We then formulate a mean squared error (MSE) minimization problem to tune the UAV trajectory and the global aggregation coefficients based on the analysis results. An algorithm based on alternating optimization (AO) and successive convex approximation (SCA) is developed to solve the formulated problem. Simulation results demonstrate the great potential of our UAV-assisted hierarchical aggregation scheme.

## I. INTRODUCTION

In the era of big data, the explosion of distributed edge data exposes the weaknesses of centralized machine learning (ML), which requests edge devices to upload local data to a central server for training, leading to huge communication costs and high data leakage risks. Federated learning (FL) has recently arisen as a promising distributed learning paradigm to tackle these challenges by replacing data transmission with model/gradient exchange [1]. Specifically, in each FL training iteration, the PS first broadcasts the global model parameters to the devices, and the latter then calculates their local gradients separately based on the local datasets and uploads the local gradients to the PS. Subsequently, the PS aggregates the received local gradients and updates the global model.

However, due to the iterative nature, the huge communication cost is still a bottleneck for FL. A promising approach to reduce the communication cost of FL in the wireless edge is to introduce the over-the-air computation (AirComp) technique [2] into FL uplink [3]. By utilizing the superposition property of wireless signals, AirComp allows the uplink cost to be unchanged with the growth of the device number (with fixed aggregation accuracy), thereby significantly reducing the communication cost of large FL systems. The corresponding FL paradigm is referred to as over-the-air FL (OA-FL).

Despite these advantages, the introduction of AirComp also brings some unique problems, such as the straggler issue [4]–[6]. Since AirComp requires local gradients to be aligned at

the PS (i.e., the PS receives a linear combination of local gradients with desired coefficients), the devices with good channel conditions have to lower their transmitting powers to match the devices with relatively poor channel conditions (i.e., the stragglers) to ensure correct aggregation. Existing works like [4] and [5] discard the stragglers to relieve the straggler issue. However, such a coarse-grained discarding strategy will inevitably cast away some exploitable information, resulting in a learning performance loss. Ref. [6] suggests a soft aggregation approach to tackle the straggler issue without discarding devices, which, however, works well only in small service area cases (e.g.,  $100 \times 100 \text{ m}^2$ ). When it comes to a relatively large service area (e.g.,  $2 \times 2 \text{ km}^2$ ), it fails due to the large difference in path loss between nearby devices and those far from the PS (which results in the de facto discarding of remote devices).

Allowing PS mobility is a natural idea to perform OA-FL in a large service area since the PS can thus move close to the stragglers for better services. Therefore, in this paper, an unmanned aerial vehicle (UAV) is introduced as the PS. Due to the ground-to-air nature of UAV uplink communication, the device-UAV links are generally not blocked, potentially leading to good channel conditions [7], [8]. Besides, UAVs have extremely high mobility and can thus cover a large area quickly. There are already some works on UAV-assisted FL. For example, Ref. [9] employs a UAV as a mobile relay in the sky to assist FL; in [10], a UAV acts as an orchestrator, coordinating the transmissions and the learning schedule within a preset deadline. However, these works consider orthogonal FL uplink and thus have various limitations in device number, service area, and communication condition, which means the state-of-art design is far from fully unleashing the potential of UAV-assisted FL.

In this paper, we make the first attempt to employ the UAV to assist the OA-FL system by designing a hierarchical over-the-air aggregation scheme. As shown in Fig. 1, we consider a UAV-assisted OA-FL system, where the UAV acts as the PS, denoted by UAV-PS, to serve devices distributed in a relatively large area. To avoid serious straggler issues, we restrict the UAV-PS to serve nearby devices, i.e., only aggregate the local gradients from nearby devices over the air. Thus the UAV-PS must fly across its large service area to serve more devices. After the UAV-PS completes one round, it further aggregates the received partially aggregated local gradients to obtain a

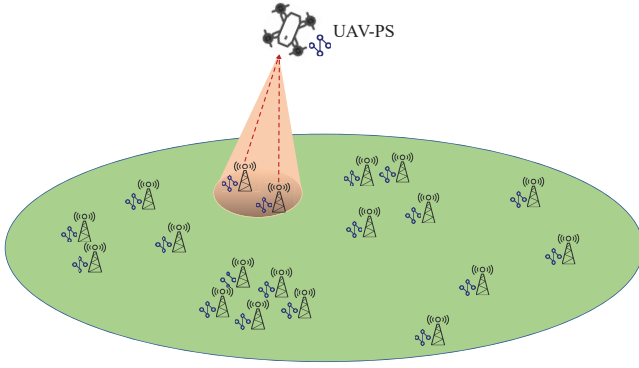


Fig. 1. UAV-assisted OA-FL in a large service area.

noisy version of the desired global gradient for global model updating. We call this two-step aggregation hierarchical aggregation. Under this hierarchical gradient aggregation scheme, we carry out a gradient-correlation-aware FL convergence analysis, which leads us to formulate a mean squared error (MSE) minimization problem to tune the UAV trajectory and the global aggregation coefficients. An algorithm based on alternating optimization (AO) and successive convex approximation (SCA) is proposed to solve the formulated problem. Simulation results demonstrate that our proposed scheme achieves robust performance improvement in complicated FL scenarios compared with existing solutions.

## II. SYSTEM MODEL

In this section, we detail the underlying models and the hierarchical aggregation scheme.

### A. Federated Learning Model

We consider an FL system where  $M$  devices collaboratively train a machine learning model assisted by a UAV-PS. Let  $\mathbf{w} \in \mathbb{R}^D$  be the parameter vector of the machine learning model and  $\mathcal{D}_m$  be the dataset on device  $m$ . Also let  $Q_m \triangleq |\mathcal{D}_m|$  and  $Q \triangleq \sum_{m=1}^M Q_m$ . Then the global loss function can be written as

$$F(\mathbf{w}) = \sum_{m=1}^M b_m F_m(\mathbf{w}) \quad (1)$$

with the local loss function

$$F_m(\mathbf{w}) = \frac{1}{Q_m} \sum_{q=1}^{Q_m} f(\mathbf{w}; \xi_{m,q}), \quad (2)$$

where  $f(\mathbf{w}; \xi_{m,q})$  denotes the sample-wise loss function based on the  $q$ -th data sample  $\xi_{m,q} \in \mathcal{D}_m$  and  $b_m \triangleq Q_m/Q$ .

FL requires, say,  $T$  training rounds for convergence. The  $t$ -th training round consists of four steps:

- *Global model broadcasting*: The UAV-PS broadcasts the current global model parameters,  $\mathbf{w}^{(t)}$ , to  $M$  devices in the service area.
- *Local gradient computation*: Each device computes its local gradient by

$$\mathbf{g}_m^{(t)} = \nabla F_m(\mathbf{w}^{(t)}), \quad (3)$$

where  $\nabla F_m(\mathbf{w}^{(t)})$  denotes the gradient of the local loss function.

- *Hierarchical gradient aggregation*: The UAV-PS flies across the service area to hierarchically aggregate the local gradients to obtain  $\hat{\mathbf{g}}^{(t)}$ , which is a noisy version of the desired global gradient

$$\mathbf{g}^{(t)} \triangleq \sum_{m=1}^M b_m \mathbf{g}_m^{(t)}. \quad (4)$$

- *Global model updating*: The UAV-PS updates the global model  $\mathbf{w}^{(t)}$  by

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \hat{\mathbf{g}}^{(t)}, \quad (5)$$

where  $\eta$  denotes the learning rate.

### B. UAV Channel Model

In this paper, we consider a three-dimensional Cartesian coordinate system. Assume that the  $m$ -th device is located on the ground with horizontal coordinate  $\mathbf{v}_m = [x_m, y_m]^\top$ ,  $m \in M$ . Also assume that the UAV-PS flies at a fixed height  $z$  above the ground with horizontal coordinate  $\mathbf{u}(\tau) = [x(\tau), y(\tau)]^\top$ ,  $0 \leq \tau \leq \Delta t$ , at instant  $\tau$ , where  $\Delta t$  is the duration required by the UAV-PS for flying one round. In each training round, the UAV trajectory satisfies the following constraints:

$$\mathbf{u}(0) = \mathbf{u}(\Delta t), \quad (6)$$

$$\|\dot{\mathbf{u}}(\tau)\| \leq V_{\max}, 0 \leq \tau \leq \Delta t, \quad (7)$$

where  $V_{\max}$  denotes the maximum speed of the UAV-PS. Note that (6) ensures that the UAV-PS starts and ends at the same point and that (7) imposes the practical maximal speed constraint. We discretize the duration  $\Delta t$  into  $N$  time slots indexed by  $n$  with equal interval  $\delta = \Delta t/N$ . We assume that  $\delta$  is sufficiently small such that the UAV-PS can be considered stationary in a time slot, even if it flies at the maximum speed  $V_{\max}$ . Therefore the trajectory constraints are rewritten as

$$\mathbf{u}[0] = \mathbf{u}[N], \quad (8)$$

$$\|\mathbf{u}[n+1] - \mathbf{u}[n]\|^2 \leq (V_{\max}\delta)^2, n = 0, 2, \dots, N-1. \quad (9)$$

We assume the Doppler effect caused by the mobility of the UAV-PS can be well compensated at the receiver and perfect channel state information (CSI) is accessible at devices [11]. Further, due to the ground-to-air nature of UAV uplink communication, we consider the line-of-sight (LoS) channels for the device-UAV links. Thus the channel from the  $m$ -th device to the UAV-PS during the  $n$ -th time slot follows the free-space path loss model, expressed as

$$h_m[n] = \sqrt{\varrho d_m^{-2}[n]} \vartheta_m[n], \quad (10)$$

where  $\varrho$  denotes the channel power gain at the reference distance  $d_{\text{ref}} = 1$  m,  $\vartheta_m[n] \triangleq e^{j\theta_m[n]}$  with  $\theta_m[n]$  denoting the phase of  $h_m[n]$ , and  $d_m[n] = \sqrt{z^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2}$  denotes the distance between the device  $m$  and the UAV-PS at the  $n$ -th time slot.

### C. Transmitting Signal Model

As mentioned before, the UAV-PS only serves nearby devices. We now define “nearby” precisely. At the  $n$ -th time slot, given a preset coverage distance threshold  $d_{\text{thr}}$ , we define set  $\mathcal{M}[n] \triangleq \{m \in [M] \mid \|\mathbf{u}[n] - \mathbf{v}_m\| \leq d_{\text{thr}}\}$ . The devices with indexes in  $\mathcal{M}[n]$  are referred to as the nearby devices.

At the  $n$ -th time slot, only the nearby devices need to transmit signals. Assume that device  $m$  is a nearby device. We now introduce how device  $m$  generates its transmitting signal. It first normalizes the local gradient  $\mathbf{g}_m^{(t)}$  for efficient transmission. Specifically, each entry of the normalized gradient  $\tilde{\mathbf{g}}_m^{(t)}$  is computed by

$$\tilde{g}_m^{(t)}(d) = \frac{g_m^{(t)}(d) - \bar{g}_m^{(t)}}{\sqrt{v_m^{(t)}}}, d \in [D], \quad (11)$$

where  $g_m^{(t)}(d)$  denotes the  $d$ -th entry of  $\mathbf{g}_m^{(t)}$ ,  $\bar{g}_m^{(t)}$  and  $v_m^{(t)}$  are computed by  $\bar{g}_m^{(t)} = \sum_{d=1}^D g_m^{(t)}(d)/D$  and  $v_m^{(t)} = \sum_{d=1}^D (g_m^{(t)}(d) - \bar{g}_m^{(t)})^2/D$  respectively. Since the cost of transmitting the scalars  $\{\tilde{g}_m^{(t)}, v_m^{(t)}\}$  is negligible compared to that of transmitting the gradients, we assume that these scalars are sent to the UAV-PS losslessly following [12] and [13].

The normalized gradient  $\tilde{\mathbf{g}}_m^{(t)}$  is then modulated as a complex vector  $\mathbf{r}_m^{(t)}$  by

$$\mathbf{r}_m^{(t)} \triangleq \tilde{\mathbf{g}}_m^{(t)}(1 : C) + j\tilde{\mathbf{g}}_m^{(t)}((C+1) : 2C) \in \mathbb{C}^C, \quad (12)$$

where  $C = D/2$  and  $\tilde{\mathbf{g}}_m^{(t)}(c_1 : c_2)$  denotes the sub-vector of  $\tilde{\mathbf{g}}_m^{(t)}$  containing the entries with indexes from  $c_1$  to  $c_2$ .<sup>1</sup> Then the transmitting signal is given by

$$\mathbf{x}_m^{(t)}[n] \triangleq \beta_m[n]\mathbf{r}_m^{(t)}, \quad (13)$$

where  $\beta_m[n] \in \mathbb{C}$  is the transmitting coefficient satisfying the power constraints  $|\beta_m[n]|^2 = P_0$ .<sup>2</sup>

### D. Hierarchical Aggregation

In our hierarchical aggregation scheme for the UAV-assisted OA-FL, the local gradients are aggregated in two steps, called *over-the-air partial aggregation* and *global aggregation*. In this subsection, we separately introduce the two steps.

1) *Over-the-air partial aggregation*: In this step, the UAV-PS flies around to receive the partially aggregated local gradients for  $N$  times. Specifically, the received signal at the  $n$ -th time slot is given by

$$\mathbf{y}^{(t)}[n] = \sum_{m \in \mathcal{M}[n]} h_m[n]\mathbf{x}_m^{(t)}[n] + \mathbf{n}^{(t)}[n], \quad (14)$$

where  $\mathbf{n}^{(t)}[n] \in \mathbb{C}^C$  is the additive white Gaussian noise (AWGN) with elements independently drawn from  $\mathcal{CN}(0, \sigma^2)$ .

For convenience, we introduce a binary indicator variable  $\alpha_m[n] \triangleq \mathbb{1}_{\|\mathbf{u}[n] - \mathbf{v}_m\| \leq d_{\text{thr}}}(\mathbf{u}[n])$ , where the indicator function

<sup>1</sup>We assume an even  $D$  for simplicity.

<sup>2</sup>We assume that all the devices transmit with full power, which is reasonable since the mobility of the UAV-PS allows the signals to be approximately aligned with desired coefficients (by designing an appropriate trajectory).

$\mathbb{1}_{\|\mathbf{u}[n] - \mathbf{v}_m\| \leq d_{\text{thr}}}(\cdot)$  takes 1 when  $\mathbf{u}[n]$  satisfies  $\|\mathbf{u}[n] - \mathbf{v}_m\| \leq d_{\text{thr}}$ . Then the received signal can be rewritten as

$$\mathbf{y}^{(t)}[n] = \sum_{m=1}^M \alpha_m[n]h_m[n]\mathbf{x}_m^{(t)}[n] + \mathbf{n}^{(t)}[n]. \quad (15)$$

By substituting (10) and (13) into (15), we have

$$\mathbf{y}^{(t)}[n] = \sum_{m=1}^M \frac{\sqrt{\rho}\alpha_m[n]\beta_m[n]\vartheta_m[n]}{\sqrt{z^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2}} \mathbf{r}_m^{(t)}[n] + \mathbf{n}^{(t)}[n]. \quad (16)$$

Further, we set  $\beta_m[n] = \overline{\vartheta_m[n]}\sqrt{P_0}$  to compensate for the phase shift of the channel. Thus the partially aggregated signal can finally be rewritten as

$$\mathbf{y}^{(t)}[n] = \sum_{m=1}^M \frac{\sqrt{\rho P_0}\alpha_m[n]}{\sqrt{z^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2}} \mathbf{r}_m^{(t)}[n] + \mathbf{n}^{(t)}[n]. \quad (17)$$

2) *Global aggregation*: After the UAV-PS completes one round, it aggregates  $\{\mathbf{y}^{(t)}[n]\}_{n=1}^N$  using coefficients  $\{\zeta[n] \in \mathbb{R}\}_{n=1}^N$  to obtain globally aggregated signal  $\mathbf{a}^{(t)}$ , i.e.,

$$\mathbf{a}^{(t)} = \sum_{n=1}^N \zeta[n]\mathbf{y}^{(t)}[n]. \quad (18)$$

Then  $\hat{\mathbf{g}}^{(t)}$  is computed by

$$\hat{\mathbf{g}}^{(t)} = \left[ \Re\{\mathbf{a}^{(t)}\}^\top, \Im\{\mathbf{a}^{(t)}\}^\top \right]^\top + \bar{g}^{(t)}\mathbf{1}_D, \quad (19)$$

where  $\bar{g}^{(t)} \triangleq \sum_{m=1}^M b_m \bar{g}_m^{(t)}$ .

## III. PERFORMANCE ANALYSIS

For the convenience of convergence analysis, we first introduce two standard assumptions [5], [14]:

**Assumption 1.** The global loss function  $F$  is strongly convex with positive parameter  $\mu$ :  $F(\mathbf{w}) \geq F(\mathbf{w}') + (\mathbf{w} - \mathbf{w}')^\top \nabla F(\mathbf{w}') + \frac{\mu}{2}\|\mathbf{w} - \mathbf{w}'\|^2, \forall \mathbf{w}, \mathbf{w}' \in \mathbb{R}^D$ .

**Assumption 2.** The gradient  $\nabla F(\cdot)$  is uniformly Lipschitz continuous with parameter  $\omega$ , i.e.,  $\|\nabla F(\mathbf{w}) - \nabla F(\mathbf{w}')\| \leq \omega\|\mathbf{w} - \mathbf{w}'\|, \forall \mathbf{w}, \mathbf{w}' \in \mathbb{R}^D$ .

Following [6], we further assume the following gradient correlation model:

**Assumption 3.** (Gradient correlation model) Define  $\tilde{\mathbf{G}}^{(t)} \triangleq [\tilde{\mathbf{g}}_1^{(t)}, \dots, \tilde{\mathbf{g}}_M^{(t)}] \in \mathbb{R}^{D \times M}$ . Let  $\mathbf{z}_d^{(t)\top} \in \mathbb{R}^M$  denote the  $d$ -th row of  $\tilde{\mathbf{G}}^{(t)}$ ,  $\forall d \in [D]$ . We assume  $\{\mathbf{z}_d^{(t)}\}_{d=1}^D$  to be independent and identically distributed (i.i.d.) with  $\mathbb{E}[\mathbf{z}_d^{(t)}] = \mathbf{0}$  and  $\mathbb{E}[\mathbf{z}_d^{(t)}(\mathbf{z}_d^{(t)})^\top] = \boldsymbol{\rho}^{(t)}, \forall d \in [D]$ .

An upper bound of the convergence performance is given in the following theorem.

**Theorem 1.** Set the learning rate  $\eta = 1/\omega$ . Under Assumptions 1 and 2, we have:

$$\begin{aligned} \mathbb{E}[\mathcal{F}(\mathbf{w}^{(T+1)}) - \mathcal{F}(\mathbf{w}^*)] &\leq \sum_{t=0}^T \left(1 - \frac{\mu}{\omega}\right)^{T-t} \mathbb{E}[\|\mathbf{e}^{(t)}\|^2] \\ &\quad + \left(\mathcal{F}(\mathbf{w}^{(0)}) - \mathcal{F}(\mathbf{w}^*)\right) \left(1 - \frac{\mu}{\omega}\right)^{T+1}, \end{aligned} \quad (20)$$

where  $\mathbf{e}^{(t)} \triangleq \mathbf{g}^{(t)} - \hat{\mathbf{g}}^{(t)}$  and the expectations are taken with respect to the gradients and the AWGN.

*Proof.* Take expectation of both sides of the recursion in [14, Lemma 2.1] and iterate it for  $t + 1$  times.  $\square$

Clearly, by Theorem 1, to improve the learning performance, we only have to minimize the mean squared error (MSE)  $\mathbb{E}[\|\mathbf{e}^{(t)}\|^2]$ , which can be written as

$$\begin{aligned} & \mathbb{E}[\|\mathbf{e}^{(t)}\|^2] \\ &= \mathbb{E}\left[\left\|\mathbf{g}^{(t)} - \hat{\mathbf{g}}^{(t)}\right\|^2\right] \\ &\stackrel{(a)}{=} \mathbb{E}\left[\left\|\sum_{m=1}^M b_m g_m^{(t)} - \sum_{m=1}^M b_m \bar{g}_m^{(t)} \mathbf{1}_D - \begin{bmatrix} \Re\{\mathbf{a}^{(t)}\} \\ \Im\{\mathbf{a}^{(t)}\} \end{bmatrix}\right\|^2\right] \\ &\stackrel{(b)}{=} \mathbb{E}\left[\left\|\sum_{m=1}^M \left(b_m \sqrt{v_m^{(t)}} - \sum_{n=1}^N \frac{\zeta[n] \sqrt{\varrho P_0} \alpha_m[n]}{\sqrt{z^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2}}\right) \tilde{\mathbf{g}}_m^{(t)} - \sum_{n=1}^N \zeta[n] \mathbf{n}_r^{(t)}[n]\right\|^2\right], \end{aligned} \quad (21)$$

where  $\mathbf{n}_r^{(t)}[n] \triangleq [\Re\{\mathbf{n}^{(t)}[n]\}^\top, \Im\{\mathbf{n}^{(t)}[n]\}^\top]^\top$ , step (a) follows from (4) and (19), and step (b) follows from (11), (12), (17) and (18).

To proceed, define  $\boldsymbol{\zeta} \triangleq [\zeta[1], \dots, \zeta[N]]^\top$ ,  $\mathbf{v}^{(t)} \triangleq [b_1 \sqrt{v_1^{(t)}}, \dots, b_M \sqrt{v_M^{(t)}}]^\top$ , and  $\mathbf{K} \in \mathbb{R}^{M \times N}$  with  $K_{m,n} = \frac{\alpha_m[n] \sqrt{\varrho P_0}}{\sqrt{z^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2}}$ . Recall  $\tilde{\mathbf{G}}^{(t)} = [\tilde{\mathbf{g}}_1^{(t)}, \dots, \tilde{\mathbf{g}}_M^{(t)}]$ . Then (21) can be rewritten as

$$\begin{aligned} & \mathbb{E}[\|\mathbf{e}^{(t)}\|^2] \\ &= \mathbb{E}\left[\left\|\tilde{\mathbf{G}}^{(t)} (\mathbf{v}^{(t)} - \mathbf{K}\boldsymbol{\zeta}) - \sum_{n=1}^N \zeta[n] \mathbf{n}_r^{(t)}[n]\right\|^2\right] \\ &\stackrel{(a)}{=} D (\mathbf{v}^{(t)} - \mathbf{K}\boldsymbol{\zeta})^\top \boldsymbol{\rho}^{(t)} (\mathbf{v}^{(t)} - \mathbf{K}\boldsymbol{\zeta}) + \frac{D\sigma^2}{2} \boldsymbol{\zeta}^\top \boldsymbol{\zeta}, \end{aligned} \quad (22)$$

where step (a) follows from Assumption 3 and the fact that the AWGN is independent of the gradients.

#### IV. SYSTEM OPTIMIZATION

##### A. Problem Formulation

Define  $\mathcal{U} = \{\mathbf{u}[n]\}_{n=1}^N$ . The MSE minimization problem can be formulated as

$$\min_{\boldsymbol{\zeta}, \mathcal{U}} \frac{\sigma^2}{2} \boldsymbol{\zeta}^\top \boldsymbol{\zeta} + (\mathbf{v}^{(t)} - \mathbf{K}\boldsymbol{\zeta})^\top \boldsymbol{\rho}^{(t)} (\mathbf{v}^{(t)} - \mathbf{K}\boldsymbol{\zeta}) \quad (23a)$$

$$\text{s. t.} \quad (8), (9). \quad (23b)$$

To properly solve problem (23), we first introduce  $\mathbf{K}$  as an optimization variable, and then slack the consequent

equality constraints. By introducing  $\mathbf{K}$ , problem (23) can be reformulated as

$$\min_{\boldsymbol{\zeta}, \mathbf{K}, \mathcal{U}} \frac{\sigma^2}{2} \boldsymbol{\zeta}^\top \boldsymbol{\zeta} + (\mathbf{v}^{(t)} - \mathbf{K}\boldsymbol{\zeta})^\top \boldsymbol{\rho}^{(t)} (\mathbf{v}^{(t)} - \mathbf{K}\boldsymbol{\zeta}) \quad (24a)$$

$$\text{s. t.} \quad K_{m,n} = \frac{\alpha_m[n] \sqrt{\varrho P_0}}{\sqrt{z^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2}}, \forall m \in [M], n \in [N], \quad (24b)$$

$$(8), (9). \quad (24c)$$

Note that the optimal  $\boldsymbol{\zeta}^*$  and  $\mathcal{U}^*$  for (24) are also optimal for (23). We then relax constraints (24b) by slacking  $\alpha_m[n]$ . Recall  $\alpha_m[n] = \mathbb{1}_{\|\mathbf{u}[n] - \mathbf{v}_m\| \leq d_{\text{thr}}}(\mathbf{u}[n])$ , which is a binary variable. Thus a natural way to slack  $\alpha_m[n]$  is to replace the constraint  $\alpha_m[n] = \mathbb{1}_{\|\mathbf{u}[n] - \mathbf{v}_m\| \leq d_{\text{thr}}}(\mathbf{u}[n])$  by

$$0 \leq \alpha_m[n] \leq \frac{d_{\text{thr}}^2}{d_{\text{thr}}^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2}. \quad (25)$$

Constraints (24b) thus can be relaxed as

$$0 \leq K_{m,n} \leq \frac{d_{\text{thr}}^2 \sqrt{\varrho P_0}}{(d_{\text{thr}}^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2)(z^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2)^{\frac{1}{2}}}, \quad (26)$$

resulting in the following optimization problem:

$$\min_{\boldsymbol{\zeta}, \mathbf{K}, \mathcal{U}} \frac{\sigma^2}{2} \boldsymbol{\zeta}^\top \boldsymbol{\zeta} + (\mathbf{v}^{(t)} - \mathbf{K}\boldsymbol{\zeta})^\top \boldsymbol{\rho}^{(t)} (\mathbf{v}^{(t)} - \mathbf{K}\boldsymbol{\zeta}) \quad (27a)$$

$$\text{s. t.} \quad (8), (9), (26). \quad (27b)$$

The objective of problem (27) is a multi-convex function of  $\boldsymbol{\zeta}$  and  $\mathbf{K}$ , which inspires us to apply the alternating optimization (AO) technique.

##### B. Optimization of $\{\mathbf{K}, \mathcal{U}\}$ with a fixed $\boldsymbol{\zeta}$

In this subsection, we optimize  $\mathbf{K}$  and  $\mathcal{U}$  with a fixed  $\boldsymbol{\zeta} = \tilde{\boldsymbol{\zeta}}$ . The corresponding optimization problem is

$$\min_{\mathbf{K}, \mathcal{U}} \tilde{\boldsymbol{\zeta}}^\top \mathbf{K}^\top \boldsymbol{\rho}^{(t)} \mathbf{K} \tilde{\boldsymbol{\zeta}} - 2\mathbf{v}^{(t)\top} \boldsymbol{\rho}^{(t)} \mathbf{K} \tilde{\boldsymbol{\zeta}} \quad (28a)$$

$$\text{s. t.} \quad (8), (9), (26). \quad (28b)$$

Note that constraint (26) is non-convex for  $\mathbf{u}[n]$ . However, the right side of (26) is convex with respect to the whole expression  $\|\mathbf{u}[n] - \mathbf{v}_m\|^2$ . Thus we can obtain a lower bound of the right side of (26) by expanding it as its first-order Taylor polynomial with respect to  $\|\mathbf{u}[n] - \mathbf{v}_m\|^2$  [15] based on the idea of successive convex approximation (SCA), i.e.,

$$\frac{d_{\text{thr}}^2 \sqrt{\varrho P_0}}{(d_{\text{thr}}^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2)(z^2 + \|\mathbf{u}[n] - \mathbf{v}_m\|^2)^{\frac{1}{2}}} \geq \Psi^{(t)} + \Psi'^{(t)} (\|\mathbf{u}[n] - \mathbf{v}_m\|^2 - \|\tilde{\mathbf{u}}[n] - \mathbf{v}_m\|^2), \quad (29)$$

where

$$\Psi^{(t)} = \frac{d_{\text{thr}}^2 \sqrt{\varrho P_0}}{(d_{\text{thr}}^2 + \|\tilde{\mathbf{u}}[n] - \mathbf{v}_m\|^2)(z^2 + \|\tilde{\mathbf{u}}[n] - \mathbf{v}_m\|^2)^{\frac{1}{2}}},$$

$$\Psi'^{(t)} = \frac{-d_{\text{thr}}^2 (1 + \frac{1}{2} d_{\text{thr}}^2 + \frac{3}{2} \|\tilde{\mathbf{u}}[n] - \mathbf{v}_m\|) \sqrt{\varrho P_0}}{(d_{\text{thr}}^2 + \|\tilde{\mathbf{u}}[n] - \mathbf{v}_m\|^2)^2 (z^2 + \|\tilde{\mathbf{u}}[n] - \mathbf{v}_m\|^2)^{\frac{3}{2}}},$$

**Algorithm 1** Algorithm for Solving Problem (23)

- 1: Input:  $\rho^{(t)}$  and  $v^{(t)}$ .
- 2: Initialize  $\{\mathbf{u}^0[n]\}_{n=1}^N$  and  $\zeta^0$  to feasible values, and set  $i = 0$ .
- 3: **repeat**
- 4:   Set  $\tilde{\zeta} = \zeta^i$  and  $\tilde{\mathbf{u}}[n] = \mathbf{u}^i[n]$ ,  $\forall n \in [N]$ , then solve problem (31) to obtain  $\{\mathbf{u}^{i+1}[n]\}_{n=1}^N$  and  $\mathbf{K}^{i+1}$ .
- 5:   Set  $\tilde{\mathbf{K}} = \mathbf{K}^{i+1}$ , then  $\zeta^{i+1} = (\frac{\sigma^2}{2}\mathbf{I}_{N \times N} + \tilde{\mathbf{K}}^\top \rho^{(t)} \tilde{\mathbf{K}})^{-1} \tilde{\mathbf{K}}^\top \rho^{(t)} v^{(t)}$ .
- 6:   Update  $i \leftarrow i + 1$ .
- 7: **until** The fractional decrease of the objective (27a) is below a threshold  $\epsilon > 0$ .
- 8: Set  $\zeta^* = \zeta^i$  and  $\mathbf{u}^*[n] = \mathbf{u}^i[n]$ ,  $\forall n \in [N]$ .

and  $\|\tilde{\mathbf{u}}[n] - \mathbf{v}_m\|^2$  denotes the expand point (i.e., expanding at point  $\|\tilde{\mathbf{u}}[n] - \mathbf{v}_m\|^2$ ). The equality in (29) holds when  $\mathbf{u}[n] = \tilde{\mathbf{u}}[n]$ . Then (26) can be transformed into

$$0 \leq K_{m,n} \leq \Psi^{(t)} + \Psi'^{(t)} (\|\mathbf{u}[n] - \mathbf{v}_m\|^2 - \|\tilde{\mathbf{u}}[n] - \mathbf{v}_m\|^2). \quad (30)$$

Since  $\|\mathbf{u}[n] - \mathbf{v}_m\|^2$  is a convex function of  $\mathbf{u}[n]$  and the right side is a linear function of  $\|\mathbf{u}[n] - \mathbf{v}_m\|^2$ , constraint (30) is convex. Finally, problem (28) can be reformulated into

$$\min_{\mathbf{K}, \mathcal{U}} \quad \tilde{\zeta}^\top \mathbf{K}^\top \rho^{(t)} \mathbf{K} \tilde{\zeta} - 2v^{(t)\top} \rho^{(t)} \mathbf{K} \tilde{\zeta} \quad (31a)$$

$$\text{s. t.} \quad (8), (9), (30), \quad (31b)$$

which is convex and can be efficiently solved by standard convex optimization solvers such as CVX.

### C. Optimization of $\zeta$ with fixed $\mathbf{K}$ and $\mathcal{U}$

Since (27a) is convex with respect to  $\zeta$  with fixed  $\mathbf{K} = \tilde{\mathbf{K}}$  and  $\mathcal{U} = \tilde{\mathcal{U}}$ , by letting  $\frac{\partial \mathbb{E}[\|\mathbf{e}^{(t)}\|^2]}{\partial \zeta} = 0$ , the optimal  $\zeta^*$  is given by

$$\zeta^* = \left( \frac{\sigma^2}{2} \mathbf{I}_{N \times N} + \tilde{\mathbf{K}}^\top \rho^{(t)} \tilde{\mathbf{K}} \right)^{-1} \tilde{\mathbf{K}}^\top \rho^{(t)} v^{(t)}. \quad (32)$$

### D. Overall Algorithm

By alternatively optimizing  $\{\mathbf{K}, \mathcal{U}\}$  and  $\zeta$ , problem (27) can be solved, and thus problem (23). We summarize the proposed algorithm in Algorithm 1. This algorithm converges since the objective (27a) is monotonically non-decreasing in the iterative process.

## V. SIMULATION RESULTS

In this section, we validate the effectiveness of the proposed UAV-PS assisted OA-FL scheme. We consider an FL system with  $M = 20$  devices on the ground in a large service area of  $2 \times 2 \text{ km}^2$ . The devices are randomly distributed in groups in order to simulate the users in clusters as in the village. The UAV-PS flies in an assumed fixed height  $z = 50 \text{ m}$  above the area and has a fixed starting and ending horizontal location  $\mathbf{u}[0] = \mathbf{u}[N] = [885, -10]^\top$ . The channel power gain, the noise power, and the maximum transmitting power

are set as  $\rho = -60 \text{ dB}$ ,  $\sigma^2 = -90 \text{ dBm}$ , and  $P_0 = 0.32 \text{ W}$ . The maximum speed and the flying time interval of the UAV-PS are assumed as  $V_{\max} = 50 \text{ m/s}$  and  $\delta = 1 \text{ s}$ . The coverage distance threshold is set as  $d_{\text{thr}} = 158 \text{ m}$  and after optimization the reconstruction to the binary  $\alpha_m[n]$  follows the rounded principle based on the optimized results. And we set the optimization threshold  $\epsilon = 10^{-4}$ .

The learning task is conducted over the MNIST dataset with  $Q = 60000$  training samples. We train a convolutional neural network (CNN) with two  $5 \times 5$  convolution layers in which the first has 16 channels, the second has 32 channels, and both have a  $2 \times 2$  max pooling, and with a fully connected layer of 50 neurons and ReLu activation, and with a softmax output layer (total parameters  $D = 39408$ ). The loss function is the cross-entropy loss. The learning rate is set as  $\eta = 0.05$  with momentum = 0.5 and the local updates include 5 mini-batches of stochastic gradient descent (SGD). We conduct the experiments over 10 Monte Carlo trials and 20 Monte Carlo trials for i.i.d. datasets and non-i.i.d. datasets respectively. For the i.i.d. data condition, data samples are assigned evenly to all devices; for the non-i.i.d. data condition, each device randomly selects 5 classes with  $Q/5M$  samples for each selected class. The correlation matrices of the gradients are approximated by  $\rho^{(t)} = \frac{1}{D} \sum_{d=1}^D \mathbf{z}_d^{(t)} \mathbf{z}_d^{(t)\top}$ .

We first present the PS location/UAV-PS trajectory optimization results in the following simulation conditions: (1) Static PS without UAV assisted, which is located on the coordinate barycenter of all devices; (2) UAV-PS without optimization in a circular trajectory with the flying period  $\Delta t = 120 \text{ s}$ ; (3) UAV-PS with optimization by Algorithm 1 and the flying period  $\Delta t = 120 \text{ s}$ ; and (4) UAV-PS with optimization by Algorithm 1 and the flying period  $\Delta t = 80 \text{ s}$ . All the above simulations are on the i.i.d. datasets.

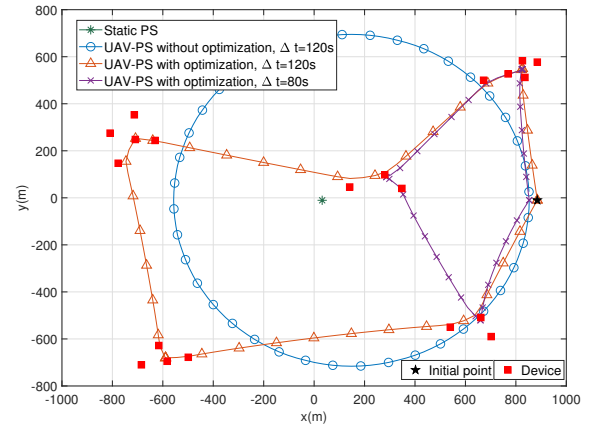


Fig. 2. PS location/UAV-PS trajectory with 20 devices.

In Fig. 2, by appropriate trajectory optimization, the UAV-PS with sufficient time, like  $\Delta t = 120 \text{ s}$ , has no need to fly right above each device but can serve for all on account of the AirComp technique to aggregate in clusters, which definitely verifies the effectiveness of the UAV assisted hierarchical over-the-air aggregation. However, when the flying time is limited,

for instance, the UAV-PS with  $\Delta t = 80$  s, it fails to fly across the whole service area and thus only be optimized to fly above nearby clusters.

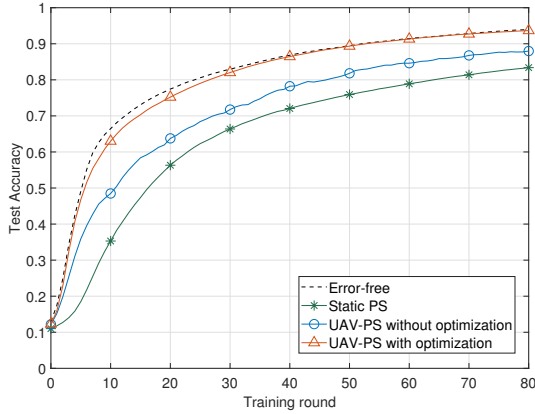


Fig. 3. OA-FL test accuracy versus training rounds, i.i.d. data.

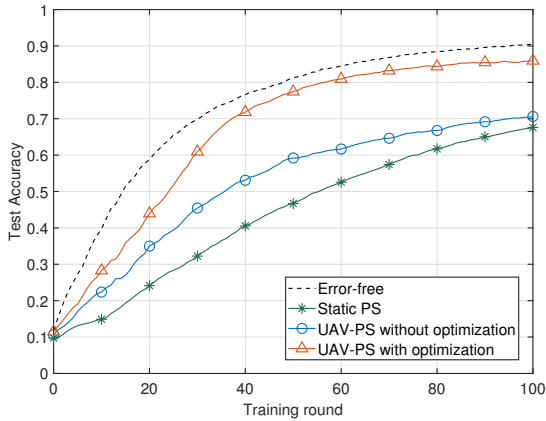


Fig. 4. OA-FL test accuracy versus training rounds, non-i.i.d. data.

Figs. 3 and 4 show the test accuracy of the UAV-assisted OA-FL system with i.i.d. data and non-i.i.d. data respectively in the following schemes: (1) Error-free bound with the PS aggregating free of error; (2) Static PS without UAV assisted, as the green location in Fig. 2, whose limitation of coverage area is waived for serving all devices, and which conducts the aggregation design as [6] to relieve the straggler issue; (3) UAV-PS in a circular trajectory, as the blue line in Fig. 2,  $\Delta t = 120$  s, which conducts the aggregation design optimization following IV-C, and the empirical optimization for the radius and the center of the circle; and (4) UAV-PS with joint optimization for the trajectory and aggregation design by Algorithm 1, as the orange line in Fig. 2,  $\Delta t = 120$  s.

For the static PS though located on the barycenter, large communication distances for remote devices as the stragglers severely limit the quality of gradient aggregation, thus leading to bad learning results. The UAV-PS with trajectory optimization by Algorithm 1 is closer to the error-free baseline when compared to the one with a circular trajectory. This indicates that the latter, without the trajectory optimization for the UAV-

PS, still constrains the training effect. And in Fig. 3 and 4, our scheme performs far better than any others on both i.i.d. and non-i.i.d. datasets, which demonstrates the effectiveness of the scheme. Further, we discover that on non-i.i.d. datasets, the learning accuracy is of higher dependence on communication and aggregation precision. Therefore our scheme is proven to have great potential on complex datasets and in complex learning scenarios.

## VI. CONCLUSION

In this paper, we proposed a UAV-PS assisted hierarchical aggregation for over-the-air FL in large service area scenarios and analyzed the convergence performance of the proposed system with gradient correlation in consideration. We formulated an MSE optimization problem to jointly optimize the UAV trajectory and the aggregation coefficients. An algorithm based on AO and SCA was proposed to solve the problem. Numerical results demonstrated the effectiveness of our scheme.

## REFERENCES

- [1] J. Konečný, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," 2016. [Online]. Available: <https://arxiv.org/pdf/1610.02527>
- [2] B. Nazer and M. Gastpar, "Computation over multiple-access channels," *IEEE Transactions on information theory*, vol. 53, no. 10, pp. 3498–3516, Oct. 2007.
- [3] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 2022–2035, Mar. 2020.
- [4] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 491–506, Jan. 2019.
- [5] H. Liu, X. Yuan, and Y.-J. A. Zhang, "Reconfigurable intelligent surface enabled federated learning: A unified communication-learning design approach," *IEEE Transactions on Wireless Communications*, vol. 20, no. 11, pp. 7595–7609, Nov. 2021.
- [6] C. Zhong, H. Yang, and X. Yuan, "Over-the-air multi-task federated learning over MIMO interference channel," 2021. [Online]. Available: <https://arxiv.org/pdf/2112.13603>
- [7] X. Lin, V. Yajnanarayana, S. D. Muruganathan, S. Gao, H. Asplund, H.-L. Maattanen, M. Bergstrom, S. Euler, and Y.-P. E. Wang, "The sky is not the limit: LTE for unmanned aerial vehicles," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 204–210, Apr. 2018.
- [8] Q. Wu, Y. Zeng, and R. Zhang, "Joint trajectory and communication design for multi-UAV enabled wireless networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 2109–2121, Jan. 2018.
- [9] W. Y. B. Lim, S. Garg, Z. Xiong, Y. Zhang, D. Niyato, C. Leung, and C. Miao, "UAV-assisted communication efficient federated learning in the era of the artificial intelligence of things," *IEEE Network*, vol. 35, no. 5, pp. 188–195, Oct. 2021.
- [10] I. Donevski, N. Babu, J. J. Nielsen, P. Popovski, and W. Saad, "Federated learning with a drone orchestrator: Path planning for minimized staleness," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 1000–1014, Apr. 2021.
- [11] X. Cao, G. Zhu, J. Xu, and K. Huang, "Cooperative interference management for over-the-air computation networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 4, pp. 2634–2651, Apr. 2020.
- [12] Z. Lin, X. Li, V. K. Lau, Y. Gong, and K. Huang, "Deploying federated learning in large-scale cellular networks: Spatial convergence analysis," Mar. 2021. [Online]. Available: <https://arxiv.org/pdf/2103.06056>
- [13] H. Yang, T. Ding, and X. Yuan, "Federated learning with lossy distributed source coding: Analysis and optimization," 2022. [Online]. Available: <https://arxiv.org/pdf/2204.10985>
- [14] M. P. Friedlander and M. Schmidt, "Hybrid deterministic-stochastic methods for data fitting," *SIAM J. Sci. Comput.*, vol. 34, no. 3, pp. A1380–A1405, Jan. 2012.
- [15] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.