Federated Learning over Hierarchical Wireless Networks: Training Latency Minimization via Submodel Partitioning

Wenzhi Fang, Dong-Jun Han, and Christopher G. Brinton

***Abstract*—Hierarchical federated learning (HFL) has demonstrated promising scalability advantages over the traditional "star-topology" architecture-based federated learning (FL). However, HFL still imposes significant computation, communication, and storage burdens on the edge, especially when training a large-scale model over resource-constrained wireless devices. In this paper, we propose *hierarchical independent submodel training* (HIST), a new FL methodology that aims to address these issues in hierarchical cloud-edge-client networks. The key idea behind HIST is to divide the global model into disjoint partitions (or submodels) per round so that each group of clients (i.e., cells) is responsible for training only one partition of the model. We characterize the convergence behavior of HIST under mild assumptions, showing the impacts of several key attributes (e.g., submodel sizes, number of cells, edge and global aggregation frequencies) on the rate and stationarity gap. Building upon the theoretical results, we propose a submodel partitioning strategy to minimize the training latency depending on network resource availability and a target learning performance guarantee. We then demonstrate how HIST can be augmented with over-the-air computation (AirComp) to further enhance the efficiency of the model aggregation over the edge cells. Through numerical evaluations, we verify that HIST is able to save training time and communication costs by wide margins while achieving comparable accuracy as conventional HFL. Moreover, our experiments demonstrate that AirComp-assisted HIST provides further improvements in training latency.**

***Index Terms*—Hierarchical federated learning, submodel training, wireless networks, over-the-air computation.**

I. Introduction

Massive amounts of training data collected by geographically distributed users have contributed to the huge success of modern machine learning (ML) applications. However, due to privacy and communication resource constraints, users may be unwilling to share their data with the service provider for centralized model training. To avoid this issue, federated learning (FL) [2], as a promising distributed learning approach, has been widely investigated recently. Different from centralized training, in the conventional version of FL, clients are responsible for training the model while the central server is only in charge of aggregating the client models and synchronizing them with the resulting global model. This enables clients to collaboratively learn a global model without any sharing of raw data. Owing to this benefit, FL has attracted significant attention in recent years [3]–[6].

In the traditional cloud-based FL [7], all clients in the system directly communicate with a central cloud server to exchange model information, resulting in communication scalability issues as the size of the network grows. To address this, hierarchical federated learning (HFL) has been proposed as an alternative [8]–[12], taking advantage of the fact that in many network systems, clusters of clients are served by intermediate edge servers (e.g., mobile devices partitioned into cells, with the base station containing an edge server). The introduction of edge servers in HFL reduces communication and scheduling complexity, as the cloud server now only needs to communicate with the edge servers. However, as the size of the model to be trained increases, the HFL training process still suffers from scalability issues. These issues manifest in several dimensions: (i) computation/storage costs at individual clients, (ii) communication burden between clients and edge servers, and (iii) communication load between edge servers and the cloud server. These are fundamental bottlenecks for the practical deployment of HFL, especially in the growing set of cases where resource-constrained devices (e.g., mobile phones) are aiming to collaboratively train a large-scale neural network (e.g., state-of-the-art image classifiers may have millions of parameters [13]).

Motivated by these challenges, in this paper, we investigate a training methodology for HFL, termed *hierarchical independent submodel training* (HIST), that is communication-, computation-, and storage efficient. One of our key innovations is to integrate independent submodel training (IST) into HFL. The core idea is to partition the global model into disjoint submodels in each training round and distribute them across different cells, so that devices in distinct cells are responsible for training different partitions of the full model. Such a submodel partitioning has the potential to reduce computation and storage loads at clients, and also alleviate communication burden on both the links between clients and edge servers and between edge servers and the cloud server. In particular, it makes the per-iteration communication complexity at the cloud server remain consistent regardless of the number of edge servers. Doing so, however, requires a careful analysis of how submodel partitioning impacts the learning performance

Wenzhi Fang and Christopher G. Brinton are with the Elmore Family School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, 47906 USA email:{fang375, cgb}@purdue.edu

Dong-Jun Han is with the Department of Computer Science and Engineering, Yonsei University, Republic of Korea. email: djh@yonsei.ac.kr.

This work was supported in part by the National Science Foundation (NSF) under grant CNS-2146171, the Office of Naval Research (ONR) under grant N00014-21-1-2472, and the Defense Advanced Research Projects Agency (DARPA) under grant D22AP00168.

An abridged version of this paper appeared in the 2024 IEEE International Conference on Communications (ICC) [1].

and training efficiency in HFL, which we address in this paper.

In addition to the training methodology, the client-edge wireless communication mechanism also has significant implications on the resource efficiency of HFL. Limited radio resources within wireless cells create communication bottlenecks when serving large transmissions from several users. For this reason, some researchers, e.g., [14], [15] have investigated optimizing radio resource allocation within cells, blended with techniques like partial client selection, to enhance the efficiency of HFL training. However, these works focus on orthogonal multiple access (OMA) transmissions in each cell, in which the communication latency incurred by each edge server in each round of training will increase with the number of clients in its coverage [16].

This motivates our second key idea, which is to consider over-the-air computation (AirComp) in HFL together with IST. With AirComp [17]–[19], clients in the same cell can transmit their models simultaneously to the edge server, resulting in significant training time savings during the model aggregation in each cell. When designed properly, the communication complexity at each edge server will not scale with the number of clients. This complements submodel partitioning, which allows the communication complexity at the cloud server to become independent of the number of cells. Achieving this, however, requires closely studying the distortion error introduced by AirComp, how that impacts the `HIST` training process, and how it can be controlled through physical-layer design. We will develop such an understanding in this paper and employ it within our optimization methodology.

A. Main Contributions

- We propose hierarchical independent submodel training (`HIST`) to enhance the scalability of FL over cloud-edge-client network topologies (Section II). `HIST` aims to reduce computation, communication, and storage costs incurred during HFL training via submodel partitioning across the edge. We demonstrate `HIST`'s applicability on fully connected and convolutional neural networks.
- We analytically characterize the convergence characteristics of `HIST` (Section III) for non-convex loss functions, under milder assumptions than those adopted in the existing IST literature. Based on the convergence result, we analyze the performance-efficiency trade-off induced by submodel partitioning, and provide guidelines on setting the key system parameters (e.g., aggregation frequencies, step size, partitioning strategy) of the proposed `HIST`.
- We analyze the impact of submodel size on the convergence bound and the training latency of `HIST`. Using these relationships, we propose a training latency minimization strategy (Section IV) which optimizes the submodel partitioning sizes without significantly compromising the learning performance (i.e., quantified through the convergence bound), while considering the network resource availability (e.g., computation powers, data rate).
- To further enhance training scalability, we propose an AirComp-assisted `HIST` for the client-edge wireless network within each cell (Section V). With AirComp in place, we show that each edge server is able to obtain an unbiased estimator of client local model averages within its coverage. We characterize the variance of this estimate on the convergence behavior of the proposed AirComp-assisted `HIST`. We leverage these relationships to augment our submodel partitioning optimization, as well as to minimize model distortion in receive beamforming.

We conduct experiments (Section VI) using both fully connected neural networks and convolutional neural networks to validate the effectiveness of the proposed algorithm in hierarchical network settings. Results show that `HIST` achieves significant resource savings for the same target accuracy compared with standard hierarchical FL in a variety of network configurations. Moreover, numerical experiments confirm that the optimized submodel partitioning strategy further reduces the training latency without model performance degradation. We also show that, under a wide signal-to-noise ratio (SNR) region, the AirComp-assisted `HIST` algorithm attains almost the same testing accuracy while significantly reducing the training latency compared with OMA-based aggregation. To the best of our knowledge, this work is one of the earliest attempts to successfully integrate HFL, IST, and AirComp and to analyze its performance over wireless networks.

This work is an extension of our conference paper [1]. Compared with [1], this paper offers the following contributions: (i) We derive a new convergence bound of our `HIST` algorithm that is explicitly a function of the mask sizes, and analyze its impact. (ii) Based on the newly derived bound, we develop our submodel partitioning optimization algorithm to minimize the training latency in each global round subject to a learning performance constraint. (iii) We develop the AirComp-assisted version of `HIST` to further enhance the efficiency of the model aggregation, and develop the associated beamforming and subnet partitioning optimizations. The combination of AirComp with IST provides communication scalability at the cloud server and edge server levels.

B. Related Works

Hierarchical federated learning: FL was first studied in a star-topology architecture where all clients are connected to a central cloud server [2]. The authors of [8], [10] extended FL to a hierarchical network that consists of a cloud server, edge servers, and clients, to reduce the communication complexity of the cloud server as well as the aggregation latency. Built upon this foundation, researchers have developed variants of hierarchical FedAvg to further improve training efficiency. Specifically, the authors of [20], [21], and [22] incorporated model quantization, sparsification, and compression into the standard hierarchical FedAvg to reduce the per-round transmission load. Furthermore, researchers in [14] focused on improving the communication efficiency of hierarchical FedAvg by optimizing bandwidth allocation during the model aggregation stage within each cell. Meanwhile, the authors of [15] formulated a joint client selection and resource allocation strategy to further enhance the efficiency when communication resources are limited. These works are orthogonal to ours, as we focus on a fundamentally different dimension of improving scalability in HFL, via submodel partitioning. These other techniques could be applied complementary to our approach.

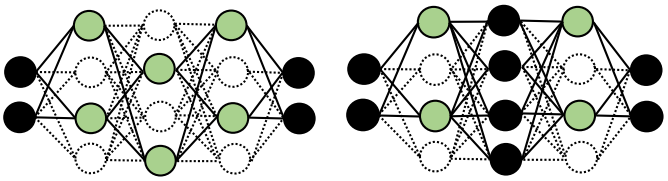
Independent submodel training: The exploration of submodel training commenced with the pioneering work [23], where the authors introduced the concept of IST for fully connected neural networks and provided theoretical analysis under centralized settings. Subsequently, submodel training was extended to graph neural networks [24] and ResNets [25]. Due to its effectiveness in addressing communication, computation, and storage issues, the concept of IST was extended to distributed scenarios in [26], where the authors empirically show the effectiveness of submodel training in FL. Several more recent studies have characterized the convergence behavior of distributed submodel training [27]–[29]. However, the aforementioned works have employed some restrictive assumptions in their analysis. Specifically, [27] assumes bounded gradients of the model, while [28] imposes a constraint on model partitioning which may be difficult to satisfy in practice (Appendix D of [29] gives an example of a strongly convex, quadratic model which violates the constraint in [28]). Furthermore, the convergence result in [28] imposes a step size adjustment which involves computing the gradient on the full model, which submodel partitioning and training aims to avoid. Additionally, some works, such as [29], focus specifically on quadratic models as opposed to more general (non-convex) ML models.

More importantly, existing works focus on cloud-based FL with a single server, and thus do not provide insights into the hierarchical case. To the best of our knowledge, `HIST` is the earliest work to integrate IST with HFL and provide theoretical analysis with experimental verification. As we will see, the multi-layer, multi-timescale nature of `HIST` introduces unique analytical challenges compared to star topology FL settings.

AirComp-assisted FL: AirComp has been investigated as a promising solution for improving the aggregation efficiency of FL over wireless networks. The authors of [30]–[32] proposed to utilize AirComp to accelerate the convergence of FedAvg in single-cell networks. Subsequently, the authors in [16] extended it to multi-cell FL and focused on the problem of inter-cell interference mitigation. [33] notably explored the incorporation of AirComp into the HFL architecture for edge model aggregations, demonstrating its advantages in terms of reducing the communication complexity of edge servers. More recently, the authors in [34] employed AirComp to support hierarchical personalized FL, and characterized the impact of AirComp on convergence, which they find introduces a non-diminishing term from aggregation distortion. Different from these works, we explore the impact of AirComp on an HFL training process that employs submodel partitioning. One of our key contributions in this paper is to optimize submodel partitioning sizes with and without AirComp, and to demonstrate the substantial reductions in latency that can be obtained by jointly designing AirComp and IST over hierarchical wireless networks.

II. Hierarchical Independent Submodel Training

In this section, we detail HFL's problem formulation, followed by the proposed `HIST` algorithm tailored to HFL.



(a) Proposed by [23], partition ratio of $\frac{1}{2^2}$.

(b) Our partition scheme, partition ratio of $\frac{1}{2}$.

Fig. 1: Example comparison of partition strategies for a fully connected neural network with multiple hidden layers.

A. System Model and Formulation

We consider an HFL system that consists of a single cloud server, N edge servers indexed by $j = 1, ..., N$, and $\sum_{j=1}^{N} n_j$ clients, where n_j is the number of clients located in the j-th cell. We let $\mathcal{C}_j$ denote the set of clients in cell j and index them $i \in \mathcal{C}_j$. Edge server j is responsible for coordinating the training process of the n_j clients in cell j. The cloud server is in charge of global model aggregations over N geographically distributed edge servers.

The system aims to train an ML model parameterized by a d-dimensional vector $\boldsymbol{x} \in \mathbb{R}^d$. Given the loss function $l(\boldsymbol{x}, \xi)$ which measures the loss on sample ξ for model $\boldsymbol{x}$, the training objective of HFL can be formulated as follows:

$$\begin{aligned} \min_{\boldsymbol{x}} f(\boldsymbol{x}) &\coloneqq \frac{1}{N}\sum_{j=1}^{N} f_j(\boldsymbol{x}), && \text{(Global loss)} \\ f_j(\boldsymbol{x}) &\coloneqq \frac{1}{n_j}\sum_{i\in\mathcal{C}_j} F_i(\boldsymbol{x}), && \text{(Cell loss)} \\ F_i(\boldsymbol{x}) &\coloneqq \mathbb{E}_{\xi_i\sim\mathcal{D}_i}\left[l(\boldsymbol{x},\xi_i)\right], && \text{(Client loss)} \end{aligned} \tag{1}$$

where $f : \mathbb{R}^d \to \mathbb{R}$, $f_j : \mathbb{R}^d \to \mathbb{R}$, and $F_i : \mathbb{R}^d \to \mathbb{R}$ represent the global loss, the loss across the j-th cell, and the loss of client i, respectively. $\mathcal{D}_i$ denotes the local data distribution of client i. In this work, we mainly consider the non-i.i.d. (non-independent and identically distributed) scenario where data distributions are heterogeneous across different clients. Our algorithm and analysis can be naturally extended to a weighted average form of (1) by introducing positive coefficients for each $f_j(\boldsymbol{x})$ and $F_i(\boldsymbol{x})$. For simplicity, following prior works [12], we assume these coefficients are incorporated into $F_i(\boldsymbol{x})$.

In conventional HFL, all clients in the system train local versions of the full model. To support such training, each client needs to be equipped with enough computation, storage, and communication resources. However, as the size of models continues to grow – the trend of deep learning – it is prohibitive for resource-constrained clients to handle full model training. This motivates us to develop a submodel partitioning strategy for HFL, which we will introduce in the rest of this section.

B. Preliminaries of Model Partitioning

Model partitioning aims to improve training efficiency by dividing a full neural network model into smaller submodels that are trained in parallel. Existing works studying such strategies include [23] and [27], which partition the hidden neurons and distribute them across clients. Fig. 1a summarizes the method proposed in [23] for fully connected layers, where

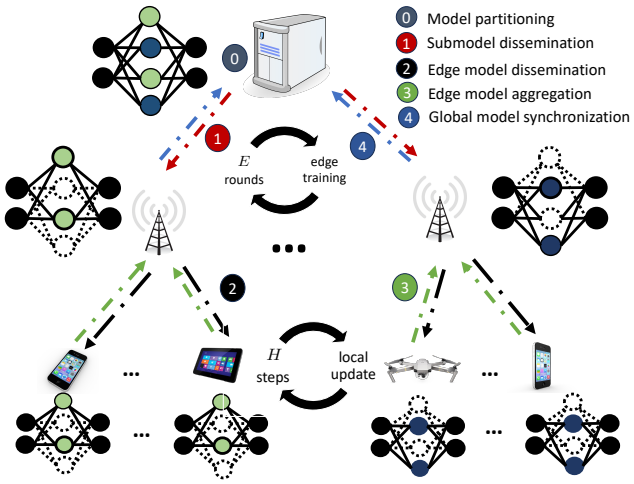


Fig. 2: Overview of the proposed HIST algorithm. Each cell is responsible for training only a specific partition of the full model in each global round, with the specific submodel partitioning changing over each round.

the neurons at every hidden layer are divided into N parts for N different submodels, one per client (shown here for $N = 2$, green and white), while leaving the input and output neurons independent of the partitioning. This results in a partition ratio of $1/N^2$, indicating the ratio of the submodel's size to the full model's size. With this strategy, the total number of parameters across N submodels is expected to be lower than that of the original full model, i.e., some parameters are expected to be missing. For example, in Fig. 1a, all link weights between green neurons of one layer and white neurons of an adjacent layer are severed, since the neurons are assigned to different clients during partitioning.

We instead consider a strategy where we in effect partition by *link* instead of neurons. To accomplish this, we can partition the neurons in one hidden layer (e.g., the l-th hidden layer) into N parts, and leave the subsequent layer (e.g., the $(l+1)$-th hidden layer) intact, with all its neurons shared across the N parts. This alternating approach is depicted in Fig. 1b (again for $N = 2$). Each client then receives one of the N parts, thereby preserving all the link weights (parameters) between two consecutive layers with none repeated (e.g., green-black links for client 1, white-black links for client 2). This results in a partition ratio of $1/N$.

C. Algorithm Description

An overview of our hierarchical federated submodel training (HIST) algorithm is presented in Fig. 2 and Algorithm 1. Similar to hierarchical FedAvg, the cloud server periodically aggregates edge models from edge servers, while each edge server periodically aggregates local models from the active clients within the corresponding cell. Active clients refer to those selected to participate in the current training round. The key difference in HIST is that clients will only store, update, and exchange a portion of the model in each training iteration.

Specifically, at the start of the t-th global round, with the current global model denoted as $\bar{\boldsymbol{x}}^t$, the cloud server initiates the training process by partitioning $\bar{\boldsymbol{x}}^t$ as follows:

$$\{\boldsymbol{p}_j^t \odot \bar{\boldsymbol{x}}^t \mid j = 1, 2, \ldots, N\}, \tag{2}$$

where $\odot$ denotes the Hadamard Product operation and $\boldsymbol{p}_j^t$ is the mask for edge server j. This mask will in general change over global rounds, and thus is indexed by t.[1] These masks are binary vectors and satisfy

$$\boldsymbol{p}_j^t \odot \boldsymbol{p}_{j'}^t = \boldsymbol{0}, \forall j' \neq j, \text{ and } \sum_{j=1}^{N} \boldsymbol{p}_j^t = \boldsymbol{1}, \tag{3}$$

which can be satisfied through our strategy in Sec. II-B.

These submodels are then distributed to the corresponding edge servers. Specifically, edge server j receives $\boldsymbol{p}_j^t \odot \bar{\boldsymbol{x}}^t$ from the cloud server and initializes its model for global round t as

$$\bar{\boldsymbol{x}}_j^{t,0} = \boldsymbol{p}_j^t \odot \bar{\boldsymbol{x}}^t, \forall j \in \{1, 2, \ldots, N\}. \tag{4}$$

Subsequently, edge server j disseminates $\bar{\boldsymbol{x}}_j^{t,0}$ to the clients in its cell for local model initialization, i.e.,

$$\boldsymbol{x}_{i,0}^{t,0} = \bar{\boldsymbol{x}}_j^{t,0}, \forall i \in \mathcal{C}_j.$$

In our notation, the 0's in the subscript and superscript refer to client-level and cell-level model initializations, respectively. Once the clients receive this model from the edge server, they commence local training on their own data.

In HIST, each global round consists of E steps of edge aggregation at each edge server and one global aggregation at the cloud server. We use (t, e) to denote the e-th edge round at global round t. Each edge round in turn includes H steps of local updates at each client and one edge aggregation. The essential steps conducted by clients, edge servers, and the cloud server in our algorithm are outlined as follows.

Clients: Let $\mathcal{C}_j^{t,e}$ denote the set of clients participating in the (t, e)-th round of training within the j-th cell, with cardinality $|\mathcal{C}_j^{t,e}| = n_j'$. This set is assumed to be a uniform sampling from the full set of clients $\mathcal{C}_j$. Each client $i \in \mathcal{C}_j^{t,e}$ computes stochastic gradients with respect to its corresponding submodel, and updates the local model for H steps via the following iteration:

$$\boldsymbol{x}_{i,h+1}^{t,e} = \boldsymbol{x}_{i,h}^{t,e} - \gamma \boldsymbol{p}_j^t \odot \nabla l(\boldsymbol{x}_{i,h}^{t,e}, \xi_{i,h}^{t,e}), h = 0, 1, \ldots, H-1, \tag{5}$$

where γ represents the learning rate and t, e, and h denote the number of global rounds, edge rounds, and local iterations, respectively. Notably, $\boldsymbol{p}_j^t \odot \nabla l(\boldsymbol{x}_{i,h}^{t,e}, \xi_{i,h}^{t,e})$ denotes the gradient of the sample loss to the submodel $\boldsymbol{x}_{i,h}^{t,e}$. Subsequently, clients upload the updated submodel to the edge servers.

Edge Servers: After every H steps of local submodel updates at the clients, each edge server j aggregates the local models of active clients within its coverage as

$$\bar{\boldsymbol{x}}_j^{t,e+1} = \frac{1}{n_j'} \sum_{i \in \mathcal{C}_j^{t,e}} \boldsymbol{x}_{i,H}^{t,e}, \forall j \in \{1, 2, \ldots, N\}. \tag{6}$$

If less than E rounds of edge training have passed, the servers disseminate (6) to their clients to initialize $\boldsymbol{x}_{i,0}^{t,e+1}$ for the next

[1] It is worth noting that for any submodel, denoted $\boldsymbol{x}_j$, there exists a mask $\boldsymbol{p}$ satisfying $\boldsymbol{x}_j = \boldsymbol{p} \odot \boldsymbol{x}$, where $\boldsymbol{x}$ denotes the full model.

iteration of local updates. Otherwise, each edge server uploads the aggregated model to the cloud to update the global model.

Cloud Server: Once the cloud server receives the latest models from edge servers, it updates the global model as:

$$\bar{\boldsymbol{x}}^{t+1} = \sum_{j=1}^{N} \bar{\boldsymbol{x}}_j^{t,E}. \tag{7}$$

Subsequently, the cloud server repartitions the global model $\bar{\boldsymbol{x}}^{t+1}$ based on a newly generated set of masks as $\bar{\boldsymbol{x}}_j^{t+1,0} = \boldsymbol{p}_j^{t+1} \odot \bar{\boldsymbol{x}}^{t+1}, \forall j \in \{1, 2, \ldots, N\}$. Finally, $\bar{\boldsymbol{x}}_j^{t+1,0}$ is sent to cell j to initiate the next round of training.

With the proposed algorithm, clients and edge servers are not required to store or manipulate the full size of the global model. This enables HIST to alleviate communication, computation, and storage burdens for clients and edge servers compared to conventional HFL. In particular, assuming each cell receives an equal-sized submodel partition, the resource requirements decrease by a factor of N. In Section III, our convergence analysis will reveal the impact of this mask selection and other HIST parameters on model training performance.

Remark 1. *While our partitioning strategy in Section II-B is presented for fully connected layers, it can be extended to convolutional layers as well. Concretely, the parameters of a convolutional layer can be represented as a 4D tensor of dimension $d^s \times d^s \times d^{in} \times d^{ker}$, where d^s denotes the spatial size, d^{in} is the number of channels of the input to this layer, and d^{ker} corresponds to the number of kernels, which also determines the number of channels in the output feature map of this layer. The output feature map serves as the input to the next convolutional layer. In other words, d^{in} for a given layer depends on the number of kernels in the last layer. Therefore, the l-th convolutional layer's tensor dimension can be expressed as $d_l^s \times d_l^s \times d_{l-1}^{ker} \times d_l^{ker}$. This can then be treated as a special matrix of size $d_{l-1}^{ker} \times d_l^{ker}$, where each element is a $d_l^s \times d_l^s$ matrix. These matrices are analogous to connections between neurons in a fully connected layer. Thus, for layer l, we can randomly assign a subset of convolutional kernels to each submodel. This partitioning in the l-th layer will specify a partitioning of the subsequent $(l+1)$-th layer as well, because the output channels of the l-th layer correspond to the input channels of the $(l+1)$-th layer. Hence, this becomes an alternating partitioning strategy, analogous the fully connected layer approach described in Sec. II-B. Specifically, we can partition the convolutional kernels in the l-th layer by selecting a subset of kernels for each group, assigning all kernels in the $(l+1)$-th layer to each group, and then partitioning again in the $(l+2)$-th layer.*

III. Convergence Analysis of HIST

This section provides convergence analysis for the proposed HIST algorithm. We note that the convergence proof of conventional hierarchical FedAvg cannot be directly extended to our case, due to the effect of the masks in submodel partitioning. Specifically, the mask $\boldsymbol{p}_j^t$ compresses not only the gradient but also the model, with an effect of the form $\boldsymbol{p}_j^t \odot \nabla F_i(\boldsymbol{p}_j^t \odot \boldsymbol{x})$, while many existing works only investigate compressing the gradient $\nabla F_i(\boldsymbol{x})$. Theoretical analysis on model compression [35] in FL is quite limited. Even in the single-cell scenario, existing analyses of IST [27], [28] rely on some stronger assumptions like bounded gradients (e.g., see Assumption 3 in [27]) and particular forms of mask partitions [28] which we aim to overcome, as discussed in Sec. I-B.

Algorithm 1: Hierarchical Independent Submodel Training (HIST) Algorithm

Initialization: masks $\{\boldsymbol{p}_1^0, \boldsymbol{p}_2^0, \ldots, \boldsymbol{p}_N^0\}$, initial models $\bar{\boldsymbol{x}}^0$, $\boldsymbol{x}_{i,0}^{0,0} = \bar{\boldsymbol{x}}_j^{0,0} = \boldsymbol{p}_j^0 \odot \bar{\boldsymbol{x}}^0, \forall i \in \mathcal{C}_j, \forall j$, learning rate γ
for $t = 0, 1, \ldots, T-1$ **do**
  for $e = 0, 1, \ldots, E-1$ **do**
    Uniformly sample a subset $\mathcal{C}_j^{t,e}$ of $\mathcal{C}_j$ for each j
    for $h = 0, 1, \ldots, H-1$ **do**
      Clients in set $\mathcal{C}_j^{t,e}, \forall j$ update local models by (5) in parallel
    end for
    Edge servers update edge models $\bar{\boldsymbol{x}}_j^{t,e+1}$ by (6) in parallel
    Edge servers broadcast the updated edge models to clients: $\boldsymbol{x}_{i,0}^{t,e+1} \leftarrow \bar{\boldsymbol{x}}_j^{t,e+1},\ \forall i \in \mathcal{C}_j$
  end for
  Cloud server updates the global model $\bar{\boldsymbol{x}}^{t+1}$ by (7)
  Generate new masks $\{\boldsymbol{p}_1^{t+1}, \boldsymbol{p}_2^{t+1}, \ldots, \boldsymbol{p}_N^{t+1}\}$
  Partition the global model according to (4) and send the obtained submodels $\bar{\boldsymbol{x}}_j^{t+1}$ to clients in cell j to initiate the next round of training, $\boldsymbol{x}_{i,0}^{t+1,0} \leftarrow \bar{\boldsymbol{x}}_j^{t+1}$, $\forall i \in \mathcal{C}_j, \forall j$
end for

The hierarchical architecture of HIST further complicates the analysis, due to the multiple layers and multi-timescale communications. Specifically, the analysis of IST over FL's star topology cannot be easily extended to HIST where the cloud server does not directly communicate with the clients; instead, communication occurs at two different timescales—between the cloud server and edge servers, and between edge servers and clients, with nested aggregation periods. This leads to what is in effect a hierarchical drift of submodels in the system, where (i) client submodels drift away from their group/cell aggregations, and (ii) cell submodels drift away from their global versions (i.e., across different partitionings), with respect to diversity in client data distributions.

A. Assumptions

Our theoretical analysis focuses on general smooth functions under a non-i.i.d data setting. The detailed assumptions are listed as follows.

Assumption 1. *The global loss function $f(\boldsymbol{x})$ has a lower bound f_*, i.e., $f(\boldsymbol{x}) \geq f_*, \forall \boldsymbol{x}$.*

Assumption 2. *The client loss F_i is differentiable and L-smooth, i.e., for any $\boldsymbol{x}$ and $\boldsymbol{y}$,*

$$\begin{aligned} &\|\nabla F_i(\boldsymbol{x}) - \nabla F_i(\boldsymbol{y})\|^2 \leq L\|\boldsymbol{y} - \boldsymbol{x}\|, \forall i, \\ &F_i(\boldsymbol{y}) \leq F_i(\boldsymbol{x}) + \langle \nabla F_i(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle + \frac{L}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2, \forall i. \end{aligned} \tag{8}$$

With Assumption 2, one can also claim that the cell losses f_j, $\forall j$ and global loss f are L-smooth.

Assumption 3. *The stochastic gradient $\nabla l(\boldsymbol{x}, \xi_i)$ is an unbiased estimator of the true gradient, i.e., $\mathbb{E}_{\xi_i \sim \mathcal{D}_i}[\nabla l(\boldsymbol{x}, \xi_i)] = \nabla F_i(\boldsymbol{x}), \forall \boldsymbol{x}, \forall i$.*

Assumption 4. *The variance of the stochastic gradient $\nabla l(\boldsymbol{x}, \xi)$ is bounded as*

$$\mathbb{E}_{\xi \sim \mathcal{D}_i} \|\nabla l(\boldsymbol{x}, \xi) - \nabla F_i(\boldsymbol{x})\|^2 \leq \sigma^2, \forall \boldsymbol{x}, \forall i. \tag{9}$$

Assumption 5. *There exists a constant δ_1^2 that bounds the gradient dissimilarity between the global loss function and edge loss functions, i,e.,*

$$\frac{1}{N} \sum_{j=1}^{N} \|\nabla f_j(\boldsymbol{x}) - \nabla f(\boldsymbol{x})\|^2 \leq \delta_1^2, \forall \boldsymbol{x}. \tag{10}$$

Assumption 6. *There exists a constant δ_2^2 that bounds gradient dissimilarities between edge loss functions and client loss functions, i,e.,*

$$\frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \|\nabla F_i(\boldsymbol{x}) - \nabla f_j(\boldsymbol{x})\|^2 \leq \delta_2^2, \forall \boldsymbol{x}, \forall j. \tag{11}$$

Assumptions 1-4 have been widely adopted in the context of stochastic non-convex and smooth optimization [36], [37]. Assumptions 5 and 6 serve to characterize the degree of data heterogeneity across cells and across clients, which are commonly used within the HFL literature [11], [38], [39].

B. Full Client Participation

The global synchronization (7) occurs at intervals of every E steps of edge model updates. If we were to directly consider $\{\nabla f(\bar{\boldsymbol{x}}^t)\}$, it would be difficult to capture the effect of local aggregation on the global model iteration as there are E local aggregations within each global iteration. Moreover, it is infeasible to establish a close connection between $\nabla f(\bar{\boldsymbol{x}}^t)$ and $\boldsymbol{x}_{i,h}^{t,e}, \forall i, h$, due to a large lag in updates, as $\boldsymbol{x}_{i,h}^{t,e}$ is updated incrementally with h, while $\bar{\boldsymbol{x}}^t$ is updated only after h completes E cycles from 0 to $H-1$. To tackle this, we introduce iterates

$$\{\hat{\boldsymbol{x}}^{t,e} = \sum_{j=1}^{N} \bar{\boldsymbol{x}}_j^{t,e} \,|\, t = 0, 1, \ldots, T-1; e = 0, 1, \ldots, E-1\}$$

serving as a virtual sequence of global models realized at each edge round e. We will capture the convergence of HIST by characterizing the bound of the gradient of the global loss function evaluated on the virtual global model iterate, i.e., $\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\|^2$.

1) Analytical Results: We first investigate the convergence properties of Algorithm 1 under full client participation (i.e., when $\mathcal{C}_j^{t,e} = \mathcal{C}_j$) by characterizing the evolution of $\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\|^2$, to capture how fast the global model converges to a stationary point. The following gives one of our main results in this paper:

Theorem 1. *Suppose that Assumptions 1-6 hold, $N \geq 2$, and the step size satisfies*

$$\gamma \leq \min\left\{\frac{1}{45\sqrt{N}EHL}, \frac{\tilde{N}}{NHL}, \frac{1}{NH^2L}, \frac{1}{N(N+1)E^2H^2L}\right\}. \tag{12}$$

Then, for an arbitrary mask partitioning satisfying (3) in each iteration, the HIST algorithm under full client participation satisfies

$$\begin{aligned}\frac{1}{TE}\sum_{t=0}^{T}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \leq\ & 4\frac{f(\bar{\boldsymbol{x}}^0) - f_*}{\gamma TEH} + 100\gamma\tilde{N}L\sigma^2 \\ & + 1356\gamma L\delta_1^2 + 60\gamma L\delta_2^2 + 6\frac{N}{d}\frac{1}{T}\sum_{t=0}^{T-1} d_{max}^t\delta_1^2 \\ & + 48(N-1)L^2\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2,\end{aligned} \tag{13}$$

where $\tilde{N} = \sum_{j=1}^{N} \frac{1}{n_j}$ and

$$d_{max}^t = \max\{\|\boldsymbol{p}_1^t\|_1, \|\boldsymbol{p}_2^t\|_1, \ldots, \|\boldsymbol{p}_N^t\|_1\}, \forall t. \tag{14}$$

Proof. Please refer to Appendix B1. □

Theorem 1 presents an upper bound for the optimality gap of HIST, characterized by the time-averaged squared gradient norm of the global function at the global virtual model sequence. The first term in this upper bound gives the effect of the initial optimality gap $f(\bar{\boldsymbol{x}}^0) - f_*$ on the convergence behavior. The second term shows how the optimality gap is related to the variance of stochastic gradients σ^2. This variance can be mitigated by enlarging the mini-batch size during the computation of stochastic gradients. The third, fourth, and fifth terms reflect the influence of non-i.i.d. characteristics within the cell (δ_2^2) and across cells (δ_1^2) on convergence. The fifth term shows that the impact of cross-cell data dissimilarity becomes more pronounced as $d_{\max}^t$ increases: certain cells must receive smaller model partitions $\|\boldsymbol{p}_i^t\|_1$ to accommodate an increasing $d_{\max}^t$, resulting in their datasets becoming reflected in relatively fewer parameters in round t. The last term demonstrates how the norm of the synchronized global model $\|\bar{\boldsymbol{x}}^t\|^2$ also impacts the optimality gap.

In addition, the step size γ is a configurable parameter that affects the first four terms of the derived upper bound. Focusing on one particular step size and employing a uniform random partitioning gives rise to the following corollary.

Corollary 1. *Suppose that Assumptions 1-6 hold, and the masks $\{\boldsymbol{p}_1^t, \boldsymbol{p}_2^t, \ldots, \boldsymbol{p}_N^t\}$ are uniformly and randomly generated based on (3). Let the step size $\gamma = (TEH)^{-\frac{1}{2}}$ in which T is large enough to satisfy (12). Then for $N \geq 2$, the HIST algorithm satisfies*

$$\begin{aligned}&\frac{1}{TE}\sum_{t=0}^{T}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \leq \mathcal{O}\left(\tilde{N}(TEH)^{-\frac{1}{2}}\right) \\ &+ \mathcal{O}\left((TEH)^{-\frac{1}{2}}\right) + \mathcal{O}\left(\delta_1^2 + (N-1)\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2\right),\end{aligned} \tag{15}$$

where $\tilde{N}$ is described in Theorem 1.

Proof. Please refer to Appendix B. □

Remark 2. *In Corollary 1, the first term is a function of $\tilde{N} = \sum_{j=1}^{N} \frac{1}{n_j}$, which is affected by the relationship between the number of clients in each cell, denoted as n_j, and the total*

number of cells, denoted as N. When the number of clients in each cell tends to be larger than the total number of cells, the convergence rate of the diminishing terms in the derived upper bound is primarily determined by $\mathcal{O}\left((TEH)^{-\frac{1}{2}}\right)$. On the other hand, if the number of clients in each cell is significantly smaller than the total number of cells, $\tilde{N}$ becomes influential, and the convergence rate is dominated by $\mathcal{O}\left(\tilde{N}^{\frac{1}{2}}(TEH)^{-\frac{1}{2}}\right)$.

2) Implications of Theorem 1 and Corollary 1: The results from Section III-A enable us to explore the performance-efficiency tradeoff induced by different parameters in HIST. We discuss several aspects here.

Non-diminishing terms: With the step size chosen in Corollary 1, the first four terms in (13) will diminish to zero as the number of total iterations T grows large. The remaining two terms are non-diminishing parts that arise due to submodel training. Therefore, HIST converges to the neighborhood of a stationary point of the loss function under the aforementioned conditions. A similar phenomenon has also been reported in the single-cell case [23], [29].

The impact of N: As N increases, i.e., as the clients in the system are divided into smaller cells, the size of the submodels gets smaller, which naturally provides computation, communication, and storage reductions for clients. However, as observed in Corollary 1, with all else constant, a larger N causes the sequence to deviate further from the stationary point, which is to be expected since each cell's data is only being used to update one model partition in each global round. Overall, this highlights the trade-off between convergence performance and resource utilization. Note also that in the extreme case when $N = 1$, $\delta_1^2 = 0$, since there is only one cell. The third, fifth, and sixth terms in (13) vanish in this case, and the result reduces to FedAvg with convergence to a stationary point, as expected. We will investigate the tradeoff associated with increasing N numerically in Sec. VI-B.

The effect of n_j: Under the bounded data heterogeneity assumptions among cells and clients, i.e., Assumptions 5 and 6, we see that increasing the number of clients n_j in each cell j has a positive effect on the convergence. This result aligns with the linear speedup characteristics observed in conventional FedAvg and hierarchical FedAvg algorithms [6], [11], [36]. To be more specific, $\tilde{N}$ monotonically decreases as n_j increases for all j, and in Corollary 1, we see that the smaller the value of $\tilde{N}$, the faster the convergence. This confirms our expectation that, all else constant, a larger cell size should provide a more well-crafted edge model in each training round.

The choices of H and E: The number of local updates, H, and the number of edge aggregations, E, are controllable parameters that impact the communication frequency. As H increases, the aggregation frequency at edge servers will become smaller, reducing the communication load between clients and the edge server. On the other hand, a large E induces fewer global synchronizations, which alleviates the communication burden between edge servers and the cloud server. Intuitively, however, these values must be upper bounded to guarantee a certain frequency of global aggregations. The maximum values of E and H can be derived from the condition on the step size γ in Theorem 1. Specifically, to ensure that the step size $\gamma = (TEH)^{-\frac{1}{2}}$ adheres to the conditions specified in (12) from Corollary 1 , H and E can be set as on the order of $\mathcal{O}\left(T^{\frac{1}{3}}N^{-\frac{4}{3}}H^{-\frac{1}{3}}\right)$ and $\min\left\{\mathcal{O}\left(\tilde{N}^2TEN^{-2}\right), \mathcal{O}\left(T^{\frac{1}{3}}N^{-\frac{4}{3}}E^{-\frac{1}{3}}\right)\right\}$ at most, respectively.

The effect of submodel partitioning: According to equations (13) and (14) in Theorem 1, a uniform partition with $\|\boldsymbol{p}_1^t\|_1 = \ldots = \|\boldsymbol{p}_N^t\|_1 = d/N$ leads the fifth term of the upper bound i.e., $6\frac{N}{d}\frac{1}{T}\sum_{t=0}^{T-1} d_{\max}^t \delta_1^2$, to attain its minimum. All else constant, varying the mask sizes across cells may cause the larger submodel partitions to be less refined from training in each global round. Nevertheless, this strategy may be problematic from a resource utilization perspective, when the communication and computation capabilities are heterogeneous across the clients. In other words, there will be a trade-off between learning performance and training efficiency/latency depending on the choice of mask sizes. This motivates us to optimize the mask sizes to balance between these competing objectives in Section III-D.

C. Partial Client Participation

We also analyze the convergence of the HIST algorithm under partial client participation. We have the following:

Theorem 2. *Suppose that Assumptions 1-6 hold, $N \geq 2$, and the step size satisfies*

$$\gamma \leq \min\left\{\frac{1}{64\sqrt{N}EHL}, \frac{\tilde{N}'}{2NHL}, \frac{1}{2NH^2L}, \frac{1}{2N(N+1)E^2H^2L}\right\}. \tag{16}$$

Then, for an arbitrary mask partitioning satisfying (3) in each iteration, the HIST algorithm under partial client participation satisfies

$$\begin{aligned}\frac{1}{TE}\sum_{t=0}^{T}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 &\leq \mathcal{O}\left(\tilde{N}'(TEH)^{-\frac{1}{2}}\right)\\ &+ \mathcal{O}\left((TEH)^{-\frac{1}{2}}\right) + \mathcal{O}\left(\tilde{N}'H^{\frac{1}{2}}(TE)^{-\frac{1}{2}}\right)\\ &+ \mathcal{O}\left(\frac{N}{d}\frac{1}{T}\sum_{t=0}^{T-1} d_{max}^t\delta_1^2 + (N-1)L^2\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2\right),\end{aligned} \tag{17}$$

where $\tilde{N}' = \sum_{j=1}^{N}\frac{1}{n_j'}$, n_j' denotes the cardinality of the participating client set at cell j, i.e., $|\mathcal{C}_j^{t,e}| = n_j'$, and d_{max}^t is defined in Theorem 1.

Proof. Please refer to Appendix B1. □

The impact of the number of participating clients in Theorem 2 is captured by $\tilde{N}'$. As the number of participating clients in any cell increases, $\tilde{N}'$ decreases, and a faster convergence speed can be achieved, as we would expect. Comparing (17) and (15), the key differences between the full and partial client participation bounds are as follows: (1) the noise variance term includes $\tilde{N}$ for full participation versus $\tilde{N}'$ for partial participation, reflecting reduced client involvement; (2) an additional divergence term $\mathcal{O}\left(\tilde{N}'H^{\frac{1}{2}}(TE)^{-\frac{1}{2}}\right)$ emerges in the partial participation case, accounting for the increased randomness introduced by limited client engagement, analogous

to observations in conventional FL scenarios [6], [36]; and (3) the bound shown in (17) is derived under a smaller learning rate (comparing (12) and (16)), which is necessary to ensure convergence to a stationary point under the higher level of randomness associated with partial participation.

D. Comparison with Convergence Analysis of Existing Works

A key distinction between our convergence analysis and previous works [23], [27]–[29] on IST is that we consider the hierarchical network architecture in this paper. Apart from this, our focus and assumptions are also different from these works. To be specific, the analysis in [23], [28] concentrates on the centralized setting. Consequently, the non-i.i.d. influence is not encompassed in their works, which is an indispensable factor in FL. Additionally, [23], [27], [28] make use of stronger assumptions in their analysis. In particular, [23], [27] assume Lipschitz continuity for the loss functions. Moreover, Assumption 5 adopted in [27] is difficult to ensure as it requires a constant to bound the client drift. On the other hand, [28] imposes some extra conditions on the masks, which may not hold in practical settings as argued in [29]. The authors in [29] then provided a tighter convergence bound for distributed IST based on milder assumptions. While their focus is on quadratic loss functions, ours is on general non-convex and smooth functions, which are more common in FL settings.

IV. Optimization for OMA-based HIST

In this section, we develop a methodology for optimizing submodel partitioning sizes to balance training efficiency and learning performance. For simplicity, we consider full client participation in this section. We construct a model for training latency with a standard orthogonal multiple access (OMA) transmission protocol in Section IV-A. The convergence performance for our mask size optimization in Section IV-B comes from Section III.

A. Training Latency Analysis

Since the edge and cloud servers possess significantly larger communication and computation resources than clients, we focus on the delays incurred from local model updates and uplink communication at the clients, as done in [40]–[42]. Additionally, we assume that the computational and communication capabilities of clients remain stable throughout a specific global training round [16], [43]. Formally, we let $\mathcal{R}_i^t$ represent the data rate (in bits/sec) for uplink communication of client i in the t-th global training round, and we let $\mathcal{F}_i^t$ represent the CPU frequency (in Hz). Furthermore, the communication load (in bits) for uploading a complete model is denoted by L_0, while the required number of CPU cycles for a single mini-batch update of the full model is represented by V_0. Without loss of generality, we assume that the latency required for local computation is linearly proportional to the model size[2] following prior works [44], [45].

[2]The analysis can be easily extended to any other functions that describe the relationship between the model size and the computation time.

Assuming a traditional frequency division multiple access (FDMA) scheme, which is one of the standard OMA protocols, the communication latency for each client in cell j is $\frac{\|\boldsymbol{p}_j^t\|_1 L_0}{\mathcal{R}_i^t d}$, where $\mathcal{R}_i^t = \frac{B}{n_j}\log(1+\text{SNR}_i\|\boldsymbol{h}_i^t\|^2)$ represent the data rate (in bits/sec) for uplink communication. $\frac{B}{n_j}$ denotes the bandwidth allocated to each client in cell j, SNR_i denotes the signal noise ratio for client i, and $\boldsymbol{h}_i^t$ represents the channel. As a result, the overall latency of one round training of HIST, including computation and communication latency, for the (t,e)-th round within cell j can be expressed as

$$\max_{i\in\mathcal{C}_j}\left\{H\frac{\|\boldsymbol{p}_j^t\|_1 V_0}{\mathcal{F}_i^t d}+\frac{\|\boldsymbol{p}_j^t\|_1 L_0}{\mathcal{R}_i^t d}\right\}. \tag{18}$$

Given that the duration required for a single update of the global model depends on the speed of the slowest edge server, we can express the overall latency for each global model update as follows:

$$\max_{j\in\{1,2,\ldots,N\}}\left\{E\max_{i\in\mathcal{C}_j}\left\{H\frac{\|\boldsymbol{p}_j^t\|_1 V_0}{\mathcal{F}_i^t d}+n_j\frac{\|\boldsymbol{p}_j^t\|_1 L_0}{\mathcal{R}_i^t d}\right\}\right\}. \tag{19}$$

B. Mask Size Optimization

Considering (13) and (19), we see that the mask sizes $\{\|\boldsymbol{p}_1^t\|_1, \|\boldsymbol{p}_2^t\|_1, \ldots, \|\boldsymbol{p}_N^t\|_1\}$ affect both the latency and the learning convergence bound. In particular, while an imbalanced mask size distribution may help speed up each global update in (19) – by assigning smaller partitions to cells with smaller communication/computation resources – it results in a larger deviation from the stationary point for the HIST algorithm in Theorem 1. Moreover, the uniform mask partition leads to the minimum convergence bound, but will only lead to the minimum latency if the resources are homogeneous. Choosing the mask partition thus induces a compromise between training efficiency and convergence performance.

In (13), recall that the impact of the mask size is contained within the fifth term. To suppress the impact of an imbalanced mask size distribution, we enforce this term to remain below to a predefined threshold ϵ_{th}. The latency minimization problem of HIST is thus formulated as

$$\min_{\{\|\boldsymbol{p}_j^t\|_1\}_{j=1}^N}\ \max_{j\in\{1,\ldots,N\}}\left\{E\max_{i\in\mathcal{C}_j}\left\{H\frac{\|\boldsymbol{p}_j^t\|_1 V_0}{\mathcal{F}_i^t d}+n_j\frac{\|\boldsymbol{p}_j^t\|_1 L_0}{\mathcal{R}_i^t d}\right\}\right\}$$

$$\text{s.t.}\quad \sum_{j=1}^N\|\boldsymbol{p}_j^t\|_1 = d \tag{20a}$$

$$6\frac{N\|\boldsymbol{p}_j^t\|_1}{d}\delta_1^2 \le \epsilon_{\text{th}},\ j=1,2,\ldots,N. \tag{20b}$$

To solve this problem, note that, (20) can be rewritten as

$$\begin{aligned}\min_{\{\|\boldsymbol{p}_j^t\|_1\}_{j=1}^N}\ & t\\ \text{s.t.}\quad & \max_{i\in\mathcal{C}_j}\left\{H\frac{\|\boldsymbol{p}_j^t\|_1 V_0}{\mathcal{F}_i^t d}+n_j\frac{\|\boldsymbol{p}_j^t\|_1 L_0}{\mathcal{R}_i^t d}\right\}\le t,\ \forall j\\ & \sum_{j=1}^N\|\boldsymbol{p}_j^t\|_1 = d\\ & 6\frac{N\|\boldsymbol{p}_j^t\|_1}{d}\delta_1^2\le\epsilon_{\text{th}},\ j=1,2,\ldots,N,\end{aligned} \tag{21}$$

which is in the form of a mixed integer linear programming (MILP) problem, given the objective and constraints are each linear terms in the variables, i.e., the mask sizes of each cell. This is solvable using standard MILP solvers, e.g., Gurobi. At the commencement of each global training round, the cloud server will solve this problem, then randomly generate masks based on these optimized mask sizes, partition the current global model $\bar{\boldsymbol{x}}^t$ accordingly, and send the obtained submodel partitions to the corresponding cells. We will see in Section VI-C how this optimization leads to improvements in testing accuracy and training latency compared with baselines.

Remark 3. *Our focus here is on HFL, where the system topology is specified based on geographical factors. The mask optimization in (20) is thus conducted based on a given topology. In clustered FL [46], by contrast, client groups are optimized based on some designated criteria (e.g., data similarity, computational capability). Our approach can be applied downstream from clustered FL, by utilizing the partitioning optimization method once the clustering is completed.*

V. AirComp-Assisted HIST Algorithm

In this section, we propose an AirComp-assisted version of HIST under full client participation, which takes advantage of the superposition property of multiple access channels at the edge layer, i.e., between clients and edge servers. We develop an optimization for AirComp as well as the partitioning.

Note that through our submodel partitioning strategy in HIST, the per-iteration communication complexity at the cloud server remains constant regardless of the number of edge servers. Conversely, the communication complexity experienced by edge servers increases in proportion to their client counts. This is one of the motivations for incorporating AirComp into the HIST methodology, i.e., to facilitate scalable model aggregations at each edge server. With AirComp-assisted HIST, the communication delay at each edge server is expected to be independent of the number of clients within its coverage, thereby accelerating the model aggregation in each cell. We assume that the downlink model dissemination in each cell is error-free, as edge servers possess sufficiently large transmit power compared to resource-constrained clients [40]–[42]. Our primary focus will be on the uplink model uploading within each cell. Different from the existing works on AirComp-assisted FL, in HIST, each edge server is in charge of aggregating a different part of the model compared to other edge servers. Thus, there is no superposition effect in terms of model parameter errors across different edge servers induced by AirComp. Additionally, the communication load of model aggregations in each edge server is configurable.

A. Signal Transmission Model

In Step 7 of Algorithm 1, each edge server aims to acquire an average of local models of clients within its coverage area, as represented by (6). We adopt AirComp to support this aggregation, which allows each edge server to directly estimate an average of signals transmitted from their clients, bypassing the decoding of individual signals. To mitigate the effect of wireless noise, instead of transmitting models, we let each client i in cell j upload its accumulated gradient $\boldsymbol{\delta}_i^{t,e}$ from H steps of local updates in the (t,e)-th round of edge training. Specifically, $\boldsymbol{\delta}_i^{t,e}$ can be written as

$$\boldsymbol{\delta}_i^{t,e} = \frac{1}{\gamma}\left(\boldsymbol{x}_{i,H}^{t,e} - \boldsymbol{x}_{i,0}^{t,e}\right) = \sum_{h=0}^{H-1} \boldsymbol{p}_j^t \odot \nabla l(\boldsymbol{x}_{i,h}^{t,e}, \xi_{i,h}^{t,e}). \quad (22)$$

Under this scheme, the impact of wireless noise on the aggregated model will be mitigated by the ratio γ compared to directly uploading the models [43], [47], as we will see later.

Consider a single-input-multiple-output (SIMO) AirComp system, where clients within each cell are deployed with a single antenna, and edge servers have M antennas. In each timeslot of AirComp, where each client i in cell j concurrently uploads a particular element of $\boldsymbol{\delta}_i^{t,e}$, i.e., $(\delta_i^{t,e})_k$, for the k-th element. The goal of the edge server in this timeslot is to estimate the average $(\bar{\delta}_j^{t,e})_k := \frac{1}{n_j}\sum_{i\in\mathcal{C}_j}(\delta_i^{t,e})_k$. The received signal becomes

$$(\hat{\delta}_j^{t,e})_k = (\boldsymbol{m}_j^{t,e})^{\mathrm{H}}\left(\sum_{i\in\mathcal{C}_j} \boldsymbol{h}_i^{t,e}\alpha_i^{t,e}(\delta_i^{t,e})_k + \boldsymbol{z}_{j,k}^{t,e}\right), \quad (23)$$

where $\alpha_i^{t,e}$ denotes the precoding factor at client i, and $\boldsymbol{h}_i^{t,e} \in \mathbb{C}^M$ represents the SIMO channel between client i and edge server j, which is assumed to be invariant during (t,e)-th communication round. We assume that $\boldsymbol{h}_i^{t,e}$ is known to clients and edge servers as in [40], [47]. $\boldsymbol{m}_j^{t,e} \in \mathbb{C}^M$ denotes the receive beamforming vector at edge server j, and $\boldsymbol{z}_{j,k}^{t,e} \sim \mathcal{CN}(0, \sigma_0^2\boldsymbol{I}_M)$ represents additive white Gaussian noise (AWGN). The precoding factor of clients in cell j is subject to a maximum power constraint, i.e., $\|\alpha_i^{t,e}\| \le P_j$.

B. AirComp Aggregations

It can be observed from (23) that the distortion between the received signal $(\hat{\delta}_j^{t,e})_k$ and the target model average $\sum_{i\in\mathcal{C}_j}(\delta_i^{t,e})_k$ comes from the misalignment between channel conditions and noise across devices. To mitigate the effect of non-uniform channels, we set the precoding factor $\alpha_i^{t,e}$ for client i as

$$(\alpha_i^{t,e})^\star = \frac{1}{n_j}\frac{\left((\boldsymbol{m}_j^{t,e})^{\mathrm{H}}\boldsymbol{h}_i^{t,e}\right)^\dagger}{\|(\boldsymbol{m}_j^{t,e})^{\mathrm{H}}\boldsymbol{h}_i^{t,e}\|^2}, \ \forall i \in \mathcal{C}_j, \quad (24)$$

where $()^\dagger$ represents a conjugate operation. To meet the energy constraint $\|\alpha_i^{t,e}\|^2 \le P_j$, let $\boldsymbol{m}_j^{t,e} = \frac{1}{\nu_j^{t,e}}\boldsymbol{a}_j^{t,e}$ where $\nu_j^{t,e} = \sqrt{P_j}\min_{i\in\mathcal{C}_j}\|(\boldsymbol{a}_j^{t,e})^{\mathrm{H}}\boldsymbol{h}_i^{t,e}\|$ and $\|\boldsymbol{a}_j^{t,e}\| = 1$, in which $\nu_j^{t,e}$ denotes the power normalization factor and $\boldsymbol{a}_j^{t,e}$ represents the normalized receive beamformer.

Under the precoding factor (24), the signal shown in (23) is simplified to $(\hat{\delta}_j^{t,e})_k = \frac{1}{n_j}\sum_{i\in\mathcal{C}_j}(\delta_i^{t,e})_k + \frac{1}{\nu_j^{t,e}}(\boldsymbol{a}_j^{t,e})^{\mathrm{H}}\boldsymbol{z}_{j,k}^{t,e}$, which is an unbiased estimator of $(\bar{\delta}_j^{t,e})_k$. The distortion of AirComp measured by mean-squared error (MSE), also known as the variance of $(\hat{\delta}_j^{t,e})_k$, is thus given by

$$\mathrm{MSE}_{j,k}^{t,e} = \frac{\sigma_0^2}{P_j\min_{i\in\mathcal{C}_j}\|(\boldsymbol{a}_j^{t,e})^{\mathrm{H}}\boldsymbol{h}_i^{t,e}\|^2}. \quad (25)$$

This transmission repeats for all elements k in the mask for cell j, i.e., $k \in \mathcal{S}_j^t = \{k|(p_j^t)_k \neq 0\}$. Consequently, the aggregated gradient at (t,e)-th round in cell j admits the following expression:

$$\hat{\boldsymbol{\delta}}_j^{t,e} = \frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\boldsymbol{\delta}_i^{t,e} + \boldsymbol{Z}_j^{t,e}, \tag{26}$$

where $\boldsymbol{Z}_j^{t,e}$ denotes a noise vector with its k-th element being $1/\nu_j^{t,e}(\boldsymbol{a}_j^{t,e})^{\mathrm{H}}\boldsymbol{z}_{j,k}^{t,e}$ if $(p_j^t)_k \neq 0$, and 0, otherwise. Each edge server j completes this process in parallel, and updates its edge model as

$$\bar{\boldsymbol{x}}_j^{t,e+1} = \bar{\boldsymbol{x}}_j^{t,e} - \gamma\hat{\boldsymbol{\delta}}_j^{t,e}, \forall j, \tag{27}$$

which is an unbiased estimator of the average of local models in $\mathcal{C}_j$, exhibiting a variance of

$$\gamma^2\mathbb{E}\|\boldsymbol{Z}_j^{t,e}\|^2 = \gamma^2\sum_{k\in\mathcal{S}_j^t}\mathrm{MSE}_{j,k}^{t,e}. \tag{28}$$

With the combination of HIST and AirComp, the communication delay at each cell is independent of the number of clients.

Remark 4. *By aggregating the accumulated gradient and then updating the edge model via an SGD step as shown in (27), the impact of channel noise on the convergence of the AirComp-Assisted HIST algorithm can be analogized to the noise induced by the inherent randomness of stochastic gradients. Such a strategy facilitates the convergence analysis in Section V-C and allows us to select an appropriate step size γ to mitigate the effect of channel noise.*

C. Convergence Analysis of AirComp-assisted HIST

In AirComp-assisted HIST, step 7 of Algorithm 1 is replaced by (27). Next, we analyze the convergence behavior of the AirComp-assisted HIST algorithm. The result is provided in Theorem 3 below.

Theorem 3. *Suppose that Assumptions 1-6 hold, $N \geq 2$, and the step size satisfies condition (12). Then the AirComp-assisted HIST algorithm satisfies*

$$\begin{aligned}\frac{1}{TE}\sum_{t=0}^{T}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 &\leq 4\frac{f(\bar{\boldsymbol{x}}^0)-f_*}{\gamma TEH} + 100\gamma\tilde{N}L\sigma^2\\ &+ 1356\gamma L\delta_1^2 + 60\gamma L\delta_2^2 + 48(N-1)L^2\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2\\ &+ \gamma\frac{1}{TE}\sum_{t=0}^{T}\sum_{e=0}^{E-1}\sum_{j=1}^{N}\mathrm{MSE}_j^{t,e} + 6\frac{N}{d}\frac{1}{T}\sum_{t=0}^{T-1}d_{max}^t\delta_1^2,\end{aligned} \tag{29}$$

where $\mathrm{MSE}_j^{t,e} = \sum_{k\in\mathcal{S}_j^t}\mathrm{MSE}_{j,k}^{t,e}$, $\mathcal{S}_j^t = \{k|p_j^t \neq 0\}$, and $\tilde{N}$ and d_{max}^t are defined in Theorem 1.

Proof. Please refer to Appendix B3. □

Compared with Theorem 1, the difference introduced by AirComp is quantified by the sixth term, i.e., the MSE from the aggregation in each cell. Plugging an appropriate step size into Theorem 3 gives rise to the following corollary.

Corollary 2. *Suppose that Assumptions 1-6 hold, and let the step size be $\gamma = (TEH)^{-\frac{1}{2}}$ for T is large enough to satisfy (12). Then the HIST algorithm satisfies*

$$\begin{aligned}\frac{1}{TE}\sum_{t=0}^{T}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 &\leq \mathcal{O}\left(\tilde{N}^{\frac{1}{2}}(TEH)^{-\frac{1}{2}}\right)\\ &+ \mathcal{O}\left((TEH)^{-\frac{1}{2}}\right) + \mathcal{O}\left((N-1)L^2\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2\right)\\ &+ \underbrace{\gamma\frac{1}{TE}\sum_{t=0}^{T}\sum_{e=0}^{E-1}\sum_{j=1}^{N}\mathrm{MSE}_j^{t,e} + 6\frac{N}{d}\frac{1}{T}\sum_{t=0}^{T-1}d_{max}^t\delta_1^2}_{\text{controllable terms}}.\end{aligned} \tag{30}$$

As shown in Theorem 3, the error term induced by AirComp, i.e., the sixth term, is of the same order as the variance of the stochastic gradient, i.e., the second term (growing proportionally to N or $\tilde{N}$). This similarity is attributed to the fact that, under the precoding design as specified in (24), the accumulated gradient estimated by AirComp remains unbiased. Consequently, the variance affects the algorithm in a manner akin to the inherent randomness of the stochastic gradient, and just adds more uncertainty.

In Theorem 3 and Corollary 2, we see there are two error terms that are controllable based on AirComp-assisted HIST variables. We address the minimization of these terms through beamforming design (Section V-D) and submodel partitioning (Section V-E) next.

D. Receive Beamforming Design

We aim to design the normalized receive beamformers to mitigate the impact of AirComp distortion, by minimizing the MSE term in Theorem 3. Note that the expression of MSE in (25) is invariant to k, i.e., $\mathrm{MSE}_{j,k}^{t,e} = \mathrm{MSE}_{j,k'}^{t,e}$, $\forall k,k' \in \mathcal{S}_j^t$. Therefore, the beamformer design problem at the (t,e)-th round can be formulated as

$$\begin{aligned}\min_{\boldsymbol{a}_1^{t,e},\boldsymbol{a}_2^{t,e},\ldots,\boldsymbol{a}_N^{t,e}} &\sum_{i=1}^{N}\frac{\|\boldsymbol{p}_j^t\|_1\sigma_0^2}{P_j\min_{i\in\mathcal{C}_j}\|(\boldsymbol{a}_j^{t,e})^{\mathrm{H}}\boldsymbol{h}_i^{t,e}\|^2}\\ \text{s.t.}\quad &\|\boldsymbol{a}_j^{t,e}\|^2 = 1,\ j = 1,2,\ldots,N.\end{aligned} \tag{31}$$

Due to the independence between $\boldsymbol{a}_j^{t,e}$ and $\boldsymbol{a}_{j'}^{t,e}$, $\forall j \neq j'$, problem (31) can be transformed into N independent subproblems, one for each cell j, as

$$\begin{aligned}\max_{\boldsymbol{a}_j^{t,e}}\min_{i\in\mathcal{C}_j}&\|(\boldsymbol{a}_j^{t,e})^{\mathrm{H}}\boldsymbol{h}_i^{t,e}\|^2\\ \text{s.t. }&\|\boldsymbol{a}_j^{t,e}\|^2 = 1,\ j = 1,2,\ldots,N.\end{aligned} \tag{32}$$

Since the objective function is increasing in $\|\boldsymbol{a}_j^{t,e}\|^2$, Problem (32) can be directly relaxed to

$$\begin{aligned}\max_{\boldsymbol{a}_j^{t,e}}\min_{i\in\mathcal{C}_j}&\|(\boldsymbol{a}_j^{t,e})^{\mathrm{H}}\boldsymbol{h}_i^{t,e}\|^2\\ \text{s.t. }&\|\boldsymbol{a}_j^{t,e}\|^2 \leq 1,\ j = 1,2,\ldots,N,\end{aligned} \tag{33}$$

without compromising optimality. Problem (33) is a non-convex quadratically constrained quadratic programming (QCQP) problem. First-order methods have been developed to solve problems in this class efficiently, e.g., the Mirror-Prox-based Successive Convex Approximation (SCA) algorithm detailed in [18].

E. Submodel Partitioning Optimization

The submodel partitioning term in Theorem 3 is the same as in Theorem 1. Different from Sec. IV that studies the OMA-based HIST, we now need a latency model that incorporates AirComp. When using AirComp, multiple devices transmit their submodel parameters concurrently in the same time slot and frequency band, as in the signal transmission model (23). Within this transmission mechanism, each parameter is amplitude-modulated to a single analog symbol and each sub-channel is dedicated to a single parameter transmission. Thus, uploading a model update of dimension $\|\boldsymbol{p}_j^t\|_1$, the total number of analog symbols to be transmitted is $\|\boldsymbol{p}_j^t\|_1$. Additionally, since AirComp employs analog aggregations, while the channel conditions and SNR will impact the aggregation error, they will in theory not impact the communication latency. Concretely, according to [41], [48], the uplink communication latency at cell j can be written as

$$\frac{\|\boldsymbol{p}_j^t\|_1}{B/\Delta f} t_s, \tag{37}$$

where B denotes the system bandwidth, Δf denotes the bandwidth of a sub-channel, and t_s denotes a symbol duration. For example, in LTE systems, each resource block with the duration of $t_s = \frac{1}{14}$ ms and sub-channel bandwidth $\Delta f = 15$ kHz [41]. Thus the total latency for each edge training round within cell j can be represented as

$$\max_{i \in \mathcal{C}_j} \left\{ H \frac{\|\boldsymbol{p}_j^t\|_1 V_0}{\mathcal{F}_i^t d} + \frac{\|\boldsymbol{p}_j^t\|_1}{B/\Delta f} t_s \right\}, \tag{38}$$

using the same computational latency model as in (18). As a result, the latency minimization problem of this AirComp-assisted HFL system can be formulated as

$$\min_{\{\|\boldsymbol{p}_j^t\|_1\}_{j=1}^N} \max_{j \in \{1,\ldots,N\}} \left\{ E \max_{i \in \mathcal{C}_j} \left\{ H \frac{\|\boldsymbol{p}_j^t\|_1 V_0}{\mathcal{F}_i^t d} + \frac{\|\boldsymbol{p}_j^t\|_1}{B/\Delta f} t_s \right\} \right\}$$
$$\text{s.t.} \quad (20a), (20b), \tag{39}$$

which can be resolved using the same method for (20).

The full implementation procedure of the AirComp-assisted HIST algorithm is illustrated in Fig. 3.

VI. Simulation Results

A. Simulation Settings

We consider a setup in which clients are evenly distributed across N cells, i.e., $n_j = n_{j'}, \forall j, j' \in \{1, 2, \ldots, N\}$, for various choices of N. Unless otherwise specified, the total number of clients in the system, i.e., Nn_j, is set to 60. We focus on two practical data distribution settings: (i) fully non-i.i.d. and (ii) i.i.d. data across cells but non-i.i.d data across the clients within the same cell. For case i), the client's dataset construction follows the approach outlined in [36]. For case (ii), we first uniformly and randomly divide the entire training set into N partitions, corresponding to N cells. We then distribute each partition to the clients within the respective cell in a non-i.i.d. manner following case (i). Case (ii) allows us to consider settings where clients that are closer in geographical proximity (i.e., within the same cell) have similar local datasets.

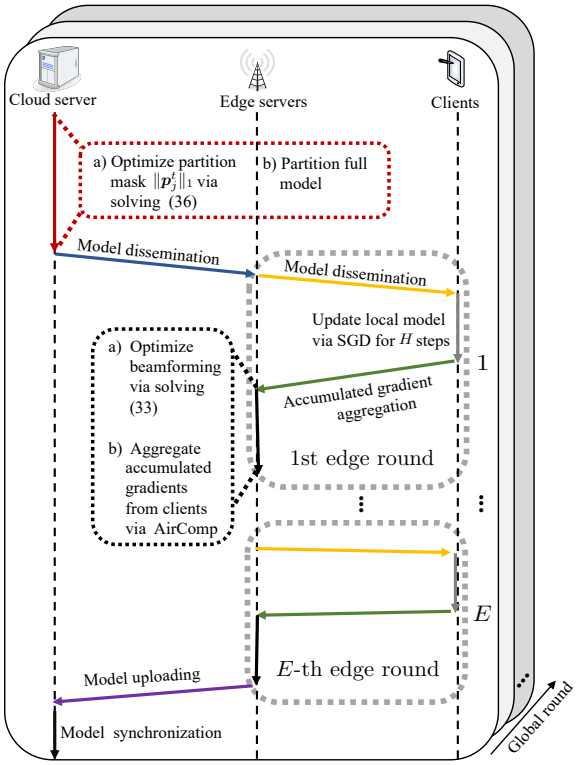


Fig. 3: Visualization of the AirComp-assisted HIST algorithm. The mask partitioning is solved once per global round, while the beamforming optimization is solved once per edge round.

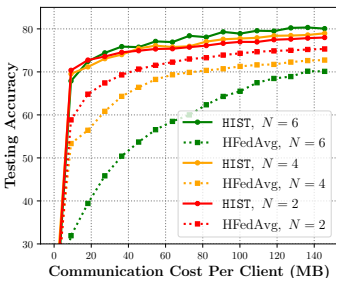
We consider training fully connected neural networks and convolutional neural networks using Fashion-MNIST and CIFAR-10 datasets, with the details given in the following subsections. In Appendix A, we show how our methodology can be applied to fine-tune transformer models with low rank adaptation (LoRA) as well. Unless stated otherwise, the experiments are conducted under full client participation.

B. Performance Evaluation of HIST

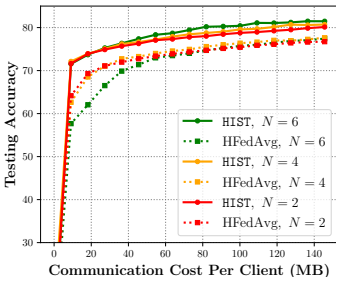
1) Fully connected neural networks: We first consider an image classification task on Fashion-MNIST using a two-layer fully connected neural network. In this model, we configure the input layer to have 784 neurons, corresponding to the size of the input image, and the output layer to have 10 neurons, which matches the number of classes. Additionally, we employ a hidden layer with 300 neurons. The model size is 0.91 MB.

The cloud server disjointly partitions the hidden neurons to construct different submodels. In this subsection, we first consider the case of the uniform partition, i.e., all submodels have the same sizes, before evaluating our partitioning optimization strategy in Section VI-C. This can be achieved by uniformly and randomly partitioning the hidden neurons. More details on our partitioning strategy and its difference from [23], [27] are discussed in Appendix II-B.

Comparison with Baseline: In Fig. 4, we compare our proposed HIST algorithm with the traditional hierarchical FedAvg algorithm (denoted as HFedAvg in our figures), where the full model is communicated over the hierarchical network.

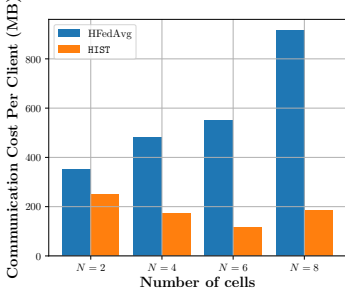


(a) Fully non-i.i.d.

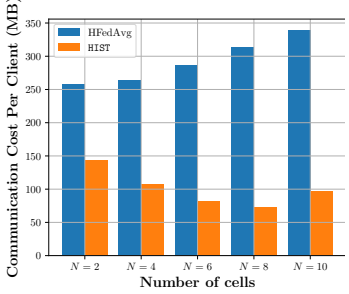


(b) Cell i.i.d., client non-i.i.d.

Fig. 4: The impact of the number of cells N on the convergence performance of HIST.

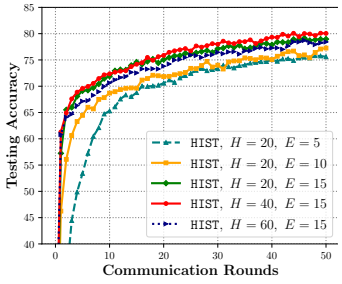


(a) Fully non-i.i.d.

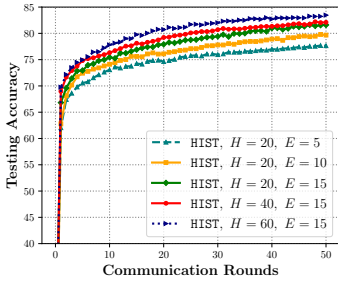


(b) Cell i.i.d., client non-i.i.d.

Fig. 5: Communication cost for achieving the testing accuracy of 80% in each scheme.

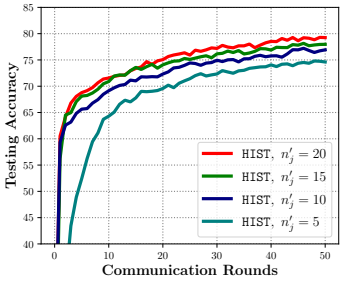


(a) Fully non-i.i.d.

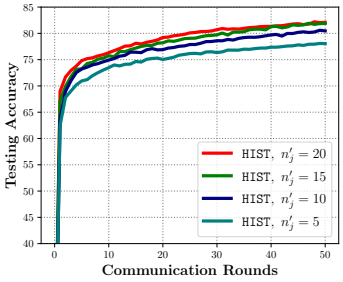


(b) Cell i.i.d., client non-i.i.d.

Fig. 6: Impacts of aggregation periods H and E on the performance of HIST.



(a) Fully non-i.i.d.



(b) Cell i.i.d., client non-i.i.d.

Fig. 7: Impact of the number of participating clients n'_j on the performance of HIST.

We evaluate the performance by comparing testing accuracy across different numbers of cells, $N \in \{2, 4, 6\}$. The X-axis here represents the communication load, quantified as the volume of parameters transmitted per client. We set H and E to 20 and 5, respectively.

As shown in Figs. 4a and 4b, the proposed HIST outperforms HFedAvg in terms of testing accuracy achieved for the same level of communication cost for both data distribution settings. In particular, under a fully non-i.i.d. data setup, as N increases, the extent of data dissimilarity across cells becomes more pronounced (since each cell has fewer class labels as N increases), leading to performance degradation for HFedAvg. In contrast, HIST achieves a higher testing accuracy for a given communication load when N increases from 2 to 6. This is due to the fact that, for HIST, the per-round communication cost per client decreases as the number of cells increases, partly compensating for the degradation of convergence induced by data heterogeneity.

Fig. 5 compares the communication cost of HIST and HFedAvg for achieving a target accuracy of 80%. The Y-axis measures the volume of parameters transmitted by each client during the training process, i.e., the X-axis of Fig. 4. It is observed that HIST takes less communication to achieve the preset accuracy, which demonstrates the efficiency of the proposed algorithm over HFedAvg. In addition, as the number of cells increases from 2 to 6, the communication cost shows a decreasing trend, which is a stark contrast to HFedAvg in the fully non-i.i.d. case. This further demonstrates the advantage of submodel partitioning in HIST, even without optimization applied. However, once the number of cells is increased to 8 (or 10 in the cell i.i.d., client non-i.i.d. case), HIST experiences performance degradation, demonstrating the efficiency-accuracy trade-off associated with the value of N. This performance drop is primarily due to the submodels becoming too small to leverage the local datasets on each device, necessitating more communication rounds to reach the same accuracy.[3] This impact of N is consistent with our theoretical result in (15): the last term increases proportionally to a larger number of cells, which eventually outpaces the reduction in per-round communication cost from smaller submodels.

Effects of System Parameters: The impacts of period of edge aggregation H and period of global synchronization E on the convergence behavior are demonstrated in Fig. 6. The X-axis represents the number of global synchronizations at the cloud server. We consider a scenario with $N = 3$. As E increases within $E \in \{5, 10, 15\}$, HIST attains a better convergence performance for both Figs. 6a and 6b. This is because a large E gives rise to more rounds of edge training within each global round. When H increases within $H \in \{20, 40, 60\}$, Fig. 6a shows that the convergence speed of HIST first increases, and then experiences a degradation. This trend occurs due to data heterogeneity, since edge devices become more prone to overfitting on their local datasets as the aggregations become less frequent. This is also reflected in our theoretical analysis: we showed in Section III-B2 that H has an upper bound to ensure the step size condition for the convergence result. On the other hand, Fig. 6b shows that HIST continues to improve in performance as H increases

[3]If the number of cells gets too large in specific scenarios, the HIST algorithm can be adapted by modifying the partition strategy (e.g., by including partial overlaps of model parameters among the partitions) to prevent such performance degradation.

from 40 to 60: this setting exhibits lower data heterogeneity, which allows for a local aggregation period before overfitting.

Effects of Client Participation Ratio: In Fig. 7, we investigate the performance of the HIST algorithm under partial client participation. Specifically, in this experiment, the number of cells is set to $N = 3$, and the aggregation periods are configured as $H = 20$ and $E = 15$. The number of clients in each cell is fixed at $n_j = 20$, but only a subset participates in each round. The results in Figs. 7a and 7b demonstrate the impact of varying the number of participating clients in each cell, denoted as n'_j, taking values $\{5, 10, 15, 20\}$. We see that, as expected, increasing the number of participating clients results in improved training performance for HIST. A higher participation ratio leads to less error in the aggregated model updates during each training round, especially when clients have diverse data distributions, which also explains why the impact of n'_j is larger in Fig. 7a. The observed speedups in performance align well with the theoretical observations discussed in Section III-C.

2) Convolutional neural networks: We numerically investigate the performance of HIST on CNNs by training LeNet-5 and ResNet-18 on Fashion MNIST and CIFAR-10, respectively. Note that the total number of parameters in the convolutional layers of LeNet-5 is 2572, while the fully connected layers consist of 59134 parameters, which is 95.8% of the entire model. Consequently, we only partition the fully connected layers in LeNet-5. In contrast, for ResNet-18, where the majority of parameters reside in the convolutional layers, we focus solely on partitioning those layers.

Similar to our previous experiments, we consider the communication cost incurred by HIST and HFedAvg for attaining some preset accuracies. For the Fashion-MNIST task using LeNet-5, we set the target accuracies to 70% and 80%, while for the CIFAR-10 with ResNet-18, the target accuracies are set to 75% and 80%. The results are shown in Tables I and II. It can be observed from both Tables I and II that HIST incurs significantly less communication cost than HFedAvg. Moreover, as we observed in Figs. 5a&5b, the advantage of the HIST algorithm gets more significant when the number of cells increases from $N = 2$ to $N = 4$. For example, referring to Table II, for the 80% accuracy level in the fully non-i.i.d. case, the communication cost of the HIST algorithm is $\frac{1}{1.65}$ of what is required by HFedAvg when the number of cells is $N = 2$. Notably, this ratio improves to $\frac{1}{2.83}$ as the number of cells increases to $N = 4$, indicating that the HIST algorithm becomes more advantageous (with a smaller ratio signifying better performance) in scenarios with more cells. This shows that the proposed hierarchical submodel training methodology significantly enhances efficiency in the training of convolutional layers over hierarchical networks as well.

C. Effect of Partitioning Optimization in OMA-based HIST

We next evaluate the effectiveness of our proposed partition optimization strategy from Section IV in reducing the training latency. We consider training the fully connected neural network on Fashion-MNIST under the non-i.i.d. data setup, using the same model architecture from Section VI-B1. In the following experiments, we determine the CPU frequency for each client based on their cell location. Specifically, for client $i \in \mathcal{C}_j,\ j \in [1, \frac{N}{2}]$, the CPU frequency is chosen randomly within the range $(1, 2)$ GHz. For clients in cells numbered in the latter half, the frequency range from which we draw is $(2, 4)$ GHz. Meanwhile, we set SNR to 30 dB for clients $i \in \mathcal{C}_j,\ j \in [1, \frac{N}{2}]$ and 40 dB for for clients $i \in \mathcal{C}_j,\ j \in (\frac{N}{2}, N]$. The channel is modeled as $\boldsymbol{h}_i^t \sim \mathcal{CN}(0, \mathbf{I}_M),\ \forall i, t$. In addition, V_0 is set to 10^6 cycles per update and ϵ_{th} is set to $9\delta_1^2$. The number of antennas at each edge server is set to $M = 10$.

In Fig. 8, we compare the performance of the HIST algorithm with and without integration of mask optimization. In this context, 'with mask optimization' is denoted as 'w', while 'uniform partition' is denoted as 'w/o'. This experiment was conducted with $N = 4$ and $E = 5$. As shown in Fig. 8a, when using the optimized mask tailored to training latency minimization, HIST exhibits a slightly slower convergence speed in terms of communication rounds, compared to the case with a uniform partition. Note that uniform partition is considered optimal in terms of the convergence bound but it does not consider the network resource availability of the system. On the other hand, as depicted in Fig. 8b, the optimized mask effectively reduces the training latency of the HIST algorithm across various settings. This is because an optimized partition will not compromise the convergence speed largely based on condition (20b) (as shown in Fig. 8a) but notably reduce per-round training latency.

Fig. 8c compares the training latency of HIST with mask optimization, the HIST with a uniform partitioning, and HFedAvg. The latency of all these algorithms decreases as the number of cells increases from 2 to 6, which can be attributed to the fact that the communication complexity at each edge server linear decreases as the number of clients in the cell decreases. Additionally, when the number of cells increases from 6 to 8, HFedAvg experiences a significant increase in training latency while HIST shows comparatively less degradation. This can be attributed to the fact that the per-round reduction in training latency partially compensates for the degradation in convergence speed. Additionally, it is clear that HIST with partitioning optimization can always beat these baselines for all $N \in \{2, 4, 6, 8\}$. These observations confirm that the HIST combined with an optimized mask could improve the training efficacy by a wide margin.

D. Performance Evaluation of the AirComp-assisted HIST

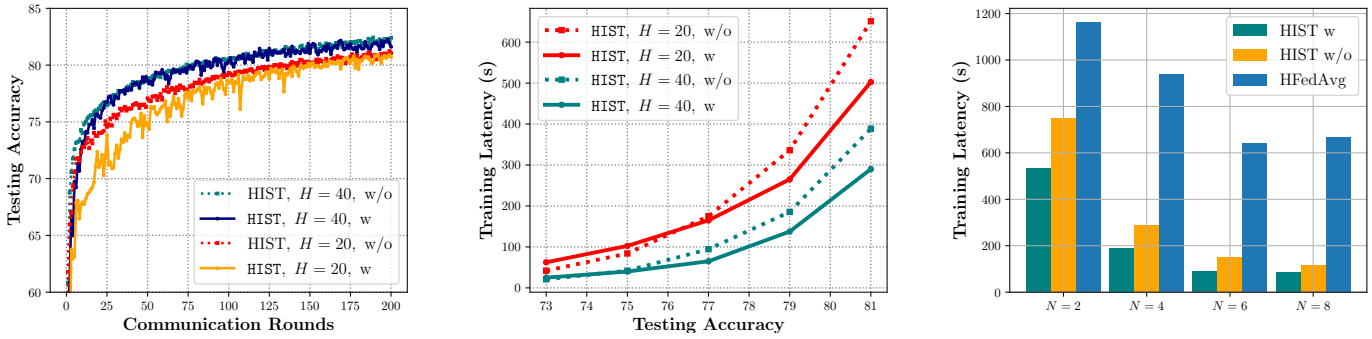
Finally, we evaluate the performance of AirComp-assisted HIST from Section V. Fig. 9a plots the convergence behavior of the AirComp-assisted HIST algorithm in training the fully connected neural network on Fashion-MNIST under different SNR values (defined as P_j/σ_0^2, as in Section VI-C). This experiment was carried out with $N = 4$, $H = 40$, and $E = 5$. We take OMA-based HIST, studied in the last subsection, as a benchmark whose convergence behavior (with respect to communication rounds) is independent of the SNR value. The results show that AirComp-assisted HIST performs well under a wide SNR region (as long as it is larger than -10 dB), which reveals that it is robust to channel conditions. However, when the SNR becomes extremely low (e.g., -20

LeNet-5 on Fashion-MNIST								
	Fully non-i.i.d.				Cell i.i.d., client non-i.i.d.			
	2 Cells		4 Cells		2 Cells		4 Cells	
Accuracy	HIST	HFedAvg	HIST	HFedAvg	HIST	HFedAvg	HIST	HFedAvg
70%	13.20	23.66	8.08	20.28	4.40	6.76	3.32	11.84
80%	115.30	206.18	76.06	246.74	38.72	86.20	24.72	89.58

TABLE I: Communication cost (MB) per client for achieving testing accuracy of 70% and 80% on Fashion-MNIST. The results show that HIST can provide significant communication savings compared with the HFedAvg baseline during training.

ResNet-18 on CIFAR-10.								
	Fully non-i.i.d.				Cell i.i.d., client non-i.i.d.			
	2 Cells		4 Cells		2 Cells		4 Cells	
Accuracy	HIST	HFedAvg	HIST	HFedAvg	HIST	HFedAvg	HIST	HFedAvg
75%	7.41	11.34	5.02	13.08	5.67	9.59	3.93	10.46
80%	37.06	61.04	23.11	65.40	24.42	40.98	14.60	43.60

TABLE II: Communication cost (GB) per client for achieving testing accuracy of 75% and 80% on CIFAR-10. The results are consistent with the Fashion-MNIST results, further underscoring the effectiveness of the proposed HIST algorithm.



(a) The impact of mask optimization on the convergence speed.

(b) The impact of mask optimization on the training latency.

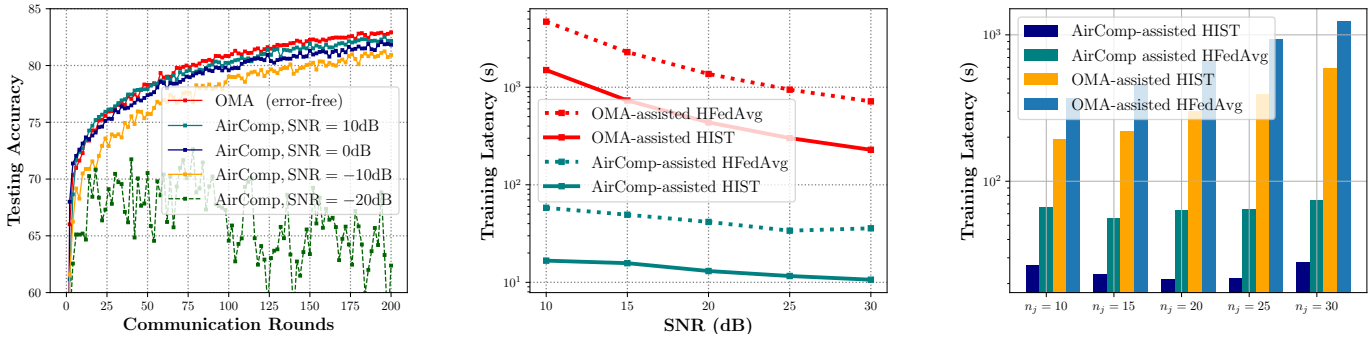
(c) Training latency comparison for attaining accuracy of 80% under different N.

Fig. 8: The impact of mask optimization in OMA-based HIST. We compare the cases with mask optimization (w) and without mask optimization (w/o) under different settings. The results show that the proposed HIST with mask optimization can achieve the target accuracy much faster with reduced training time.

dB), the convergence behavior degrades and leads to large oscillations. A low SNR leads to a large variance in the model aggregated by AirComp, preventing HIST from converging. Such a phenomenon fits well with our theoretical analysis provided in Section V, since the SNR will directly impact the MSE, which makes the convergence performance poorer.

Fig. 9b shows the required training latency for attaining a target testing accuracy of 80%. We compare the training latency of the OMA-based and AirComp-assisted HFedAvg and HIST under different SNR values. The CPU frequencies of clients are drawn from $(2, 4)$ GHz. Other settings are the same as that of Fig. 9a. As shown in Fig. 9b, the training latency decreases as the SNR increases for all algorithms. For the OMA scheme, an error-free aggregation, the SNR affects the upload rate, which in turn impacts the latency experienced in each round of communication. In the AirComp scheme, a larger SNR results in a smaller error during aggregation, and thus reduces the training latency for achieving the preset testing accuracy. Importantly, we see that AirComp-assisted HIST achieves substantially improved training latency for all choices of SNR, showing the joint advantage of AirComp and submodel partitioning design in improving efficiency.

Fig. 9c shows the impact of the number of clients in each cell on the training latency. We consider $N = 4$, $H = 40$, and $E = 5$. The number of clients in each cell is varied across $n_j \in \{10, 15, 20, 25, 30\}$, with the total number of clients Nn_j varying accordingly. Other parameters including CPU frequencies and SNRs are the same as in Section VI-C. As depicted in Fig. 9c, the AirComp-assisted HIST significantly reduces latency compared to both the OMA-based HIST and the AirComp-assisted HFedAvg. Additionally, the latency of the OMA-based HIST has a noticeable upward trend as the number of clients increases, while the fluctuation of the AirComp-assisted algorithms is slighter. This can be attributed to the fact that the communication latency of the OMA schemes linearly increases with the number of clients, whereas the per-round latency of the AirComp-assisted algorithms is independent of the number of clients. For AirComp-assisted HIST, the downward trend in training latency from $n_j = 10$ to 20 can be attributed to a speedup of convergence coming from



(a) The impact of SNR on the convergence speed over training rounds.

(b) The impact of SNR on the training latency to achieve 80% testing accuracy.

(c) Training latency comparison for attaining accuracy of 80% under different n_j.

Fig. 9: The performance of AirComp-assisted HIST. The proposed mask optimization strategy combined with AirComp-assisted HIST provides significant performance advantages compared with baselines under different settings.

having more clients in each cell, consistent with the effect of a decreasing $\tilde{N}$ in Theorem 3. On the other hand, as the number of clients grows large, this will induce a higher diversity across the data distributions in each cell. The impact of this is an increasing δ_2 (within-cell gradient dissimilarity), in Theorem 3 makes the convergence bound worse. Accordingly, in this experiment, we see that as the number of clients exceeds a specific value, i.e., 25, the impact of the induced data heterogeneity begins to dominate any speedup from having more devices, thus slowing down the overall convergence.

VII. Conclusion and Future Work

In this paper, we proposed hierarchical federated submodel training (HIST), which integrates independent submodel partitioning into hierarchical FL to obtain improvements in communication, computation, and storage efficiencies for training neural networks. We characterized the convergence behavior of HIST with arbitrary submodel partitioning under non-convex loss functions and non-i.i.d. data settings. This revealed the impacts of submodel partitioning sizes, the degree of non-i.i.d. data, and other factors on the convergence performance. Based on the derived convergence bound, we proposed an algorithm for optimizing the model partition strategy across cells, minimizing the training latency subject to maintaining a desired loss. Subsequently, we adopted AirComp-assisted local aggregations within each cell to further enhance the efficiency of HIST over hierarchical wireless networks. Numerical evaluations showed that HIST is able to achieve target accuracies significantly faster and with less resource costs compared to a baseline. Improvements in training latency from our submodel partitioning optimization and AirComp were also demonstrated empirically.

In this work, we verified that HIST is applicable to both fully connected layers and CNN layers without strict constraints. Additionally, when considering transformer-based models, we showed how the LoRA concept can be employed to apply our method for fine-tuning. Future work can investigate the HIST strategy in the context of other deep learning architectures.

References

[1] W. Fang, D.-J. Han, and C. G. Brinton, "Submodel partitioning in hierarchical federated learning: Algorithm design and convergence analysis," *arXiv preprint arXiv:2310.17890*, 2023.

[2] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*, pp. 1273–1282, PMLR, 2017.

[3] C. T. Dinh, N. H. Tran, M. N. Nguyen, C. S. Hong, W. Bao, A. Y. Zomaya, and V. Gramoli, "Federated learning over wireless networks: Convergence analysis and resource allocation," *IEEE/ACM Transactions on Networking*, vol. 29, no. 1, pp. 398–409, 2020.

[4] S. Wang, Y. Ruan, Y. Tu, S. Wagle, C. G. Brinton, and C. Joe-Wong, "Network-aware optimization of distributed learning for fog computing," *IEEE/ACM Transactions on Networking*, vol. 29, no. 5, pp. 2019–2032, 2021.

[5] L. Yuan, L. Sun, P. S. Yu, and Z. Wang, "Decentralized federated learning: A survey and perspective," *arXiv preprint arXiv:2306.01603*, 2023.

[6] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "A novel framework for the analysis and design of heterogeneous federated learning," *IEEE Transactions on Signal Processing*, vol. 69, pp. 5234–5249, 2021.

[7] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, 2019.

[8] L. Liu, J. Zhang, S. Song, and K. B. Letaief, "Client-edge-cloud hierarchical federated learning," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2020.

[9] F. P.-C. Lin, S. Hosseinalipour, S. S. Azam, C. G. Brinton, and N. Michelusi, "Semi-decentralized federated learning with cooperative d2d local model aggregations," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3851–3869, 2021.

[10] S. Hosseinalipour, S. S. Azam, C. G. Brinton, N. Michelusi, V. Aggarwal, D. J. Love, and H. Dai, "Multi-stage hybrid federated learning over large-scale d2d-enabled fog networks," *IEEE/ACM Transactions on Networking*, vol. 30, no. 4, pp. 1569–1584, 2022.

[11] J. Wang, S. Wang, R.-R. Chen, and M. Ji, "Demystifying why local aggregation helps: Convergence analysis of hierarchical SGD," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 8548–8556, 2022.

[12] W. Fang, D.-J. Han, E. Chen, S. Wang, and C. Brinton, "Hierarchical federated learning with multi-timescale gradient correction," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[14] W. Y. B. Lim, J. S. Ng, Z. Xiong, J. Jin, Y. Zhang, D. Niyato, C. Leung, and C. Miao, "Decentralized edge intelligence: A dynamic resource allocation framework for hierarchical federated learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 3, pp. 536–550, 2021.

[15] W. Wen, Z. Chen, H. H. Yang, W. Xia, and T. Q. Quek, "Joint scheduling and resource allocation for hierarchical federated edge learning," *IEEE Transactions on Wireless Communications*, vol. 21, no. 8, pp. 5857–5872, 2022.

[16] Z. Wang, Y. Zhou, Y. Shi, and W. Zhuang, "Interference management for over-the-air federated learning in multi-cell wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 8, pp. 2361–2377, 2022.

[17] G. Zhu, J. Xu, K. Huang, and S. Cui, "Over-the-air computing for wireless data aggregation in massive iot," *IEEE Wireless Communications*, vol. 28, no. 4, pp. 57–65, 2021.

[18] W. Fang, Y. Jiang, Y. Shi, Y. Zhou, W. Chen, and K. B. Letaief, "Over-the-air computation via reconfigurable intelligent surface," *IEEE Transactions on Communications*, vol. 69, no. 12, pp. 8612–8626, 2021.

[19] Z. Wang, Y. Zhao, Y. Zhou, Y. Shi, C. Jiang, and K. B. Letaief, "Over-the-air computation: Foundations, technologies, and applications," *arXiv preprint arXiv:2210.10524*, 2022.

[20] L. Liu, J. Zhang, S. Song, and K. B. Letaief, "Hierarchical federated learning with quantization: Convergence analysis and system design," *IEEE Transactions on Wireless Communications*, vol. 22, no. 1, pp. 2–18, 2022.

[21] M. S. H. Abad, E. Ozfatura, D. Gunduz, and O. Ercetin, "Hierarchical federated learning across heterogeneous cellular networks," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8866–8870, IEEE, 2020.

[22] G. Malinovsky, K. Yi, and P. Richtárik, "Variance reduced proxskip: Algorithm, theory and application to federated learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 15176–15189, 2022.

[23] B. Yuan, C. R. Wolfe, C. Dun, Y. Tang, A. Kyrillidis, and C. Jermaine, "Distributed learning of fully connected neural networks using independent subnet training," *Proceedings of the VLDB Endowment*, vol. 15, no. 8, pp. 1581–1590, 2022.

[24] C. R. Wolfe, J. Yang, F. Liao, A. Chowdhury, C. Dun, A. Bayer, S. Segarra, and A. Kyrillidis, "Gist: Distributed training for large-scale graph convolutional networks," *Journal of Applied and Computational Topology*, pp. 1–53, 2023.

[25] C. Dun, C. R. Wolfe, C. M. Jermaine, and A. Kyrillidis, "Resist: Layer-wise decomposition of resnets for distributed training," in *Uncertainty in Artificial Intelligence*, pp. 610–620, PMLR, 2022.

[26] E. Diao, J. Ding, and V. Tarokh, "Heterofl: Computation and communication efficient federated learning for heterogeneous clients," in *International Conference on Learning Representations*, 2021.

[27] H. Zhou, T. Lan, G. Venkataramani, and W. Ding, "Every parameter matters: Ensuring the convergence of federated learning with dynamic heterogeneous models reduction," *arXiv preprint arXiv:2310.08670*, 2023.

[28] A. Mohtashami, M. Jaggi, and S. Stich, "Masked training of neural networks with partial gradients," in *International Conference on Artificial Intelligence and Statistics*, pp. 5876–5890, PMLR, 2022.

[29] E. Shulgin and P. Richtárik, "Towards a better theoretical understanding of independent subnetwork training," *arXiv preprint arXiv:2306.16484*, 2023.

[30] 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, 2020.

[31] G. Zhu, Y. Du, D. Gündüz, and K. Huang, "One-bit over-the-air aggregation for communication-efficient federated edge learning: Design and convergence analysis," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 2120–2135, 2020.

[32] X. Cao, G. Zhu, J. Xu, Z. Wang, and S. Cui, "Optimized power control design for over-the-air federated edge learning," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 342–358, 2021.

[33] O. Aygün, M. Kazemi, D. Gündüz, and T. M. Duman, "Over-the-air federated edge learning with hierarchical clustering," *arXiv preprint arXiv:2207.09232*, 2022.

[34] F. Zhou, Z. Wang, X. Luo, and Y. Zhou, "Over-the-air computation assisted hierarchical personalized federated learning," in *ICC 2023-IEEE International Conference on Communications*, pp. 5940–5945, IEEE, 2023.

[35] A. Khaled and P. Richtárik, "Gradient descent with compressed iterates," *arXiv preprint arXiv:1909.04716*, 2019.

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

[37] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.

[38] C. Feng, H. H. Yang, D. Hu, Z. Zhao, T. Q. Quek, and G. Min, "Mobility-aware cluster federated learning in hierarchical wireless networks," *IEEE Transactions on Wireless Communications*, vol. 21, no. 10, pp. 8441–8458, 2022.

[39] M. F. Pervej, R. Jin, and H. Dai, "Hierarchical federated learning in wireless networks: Pruning tackles bandwidth scarcity and system heterogeneity," *IEEE Transactions on Wireless Communications*, 2024.

[40] 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, 2021.

[41] X. Cao, G. Zhu, J. Xu, and S. Cui, "Transmission power control for over-the-air federated averaging at network edge," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 5, pp. 1571–1586, 2022.

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

[43] J. Zhu, Y. Shi, Y. Zhou, C. Jiang, W. Chen, and K. B. Letaief, "Over-the-air federated learning and optimization," *arXiv preprint arXiv:2310.10089*, 2023.

[44] D.-J. Han, D.-Y. Kim, M. Choi, C. G. Brinton, and J. Moon, "Splitgp: Achieving both generalization and personalization in federated learning," in *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications*, pp. 1–10, 2023.

[45] D.-J. Han, D.-Y. Kim, M. Choi, D. Nickel, J. Moon, M. Chiang, and C. G. Brinton, "Federated split learning with joint personalization-generalization for inference-stage optimization in wireless edge networks," *IEEE Transactions on Mobile Computing*, doi:10.1109/TMC.2023.3331690.

[46] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 19586–19597, 2020.

[47] Y. Zou, Z. Wang, X. Chen, H. Zhou, and Y. Zhou, "Knowledge-guided learning for transceiver design in over-the-air federated learning," *IEEE Transactions on Wireless Communications*, vol. 22, no. 1, pp. 270–285, 2022.

[48] 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, 2019.

[49] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[50] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[51] K. Kuo, A. Raje, K. Rajesh, and V. Smith, "Federated lora with sparse communication," *arXiv preprint arXiv:2406.05233*, 2024.

Wenzhi Fang received the B.S. degree from Shanghai University in 2020 and completed his master's degree at ShanghaiTech University in 2023. Currently, he is pursuing a PhD in Electrical and Computer Engineering at Purdue University. His research interests focus on optimization and machine learning.

Dong-Jun Han is an Assistant Professor at the Department of Computer Science and Engineering at Yonsei University, South Korea. He received the B.S. degrees in mathematics and electrical engineering, and the M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), South Korea, in 2016, 2018, and 2022, respectively. His research interest is at the intersection of communications, networking, and machine learning, specifically in distributed/federated machine learning and network optimization.

Christopher G. Brinton is the Elmore Associate Professor of Electrical and Computer Engineering at Purdue University. His research interest is at the intersection of networking, communications, and machine learning, specifically in fog/edge network intelligence, distributed machine learning, and AI/ML-inspired wireless network optimization. Dr. Brinton received the Ph.D. (with honors) and M.S. degrees from Princeton in 2016 and 2013, respectively, both in Electrical Engineering.

APPENDIX

A. Extension to LoRA Fine-tuning of Transformers

Transformer-based models have received a lot of recent attention, particularly for large language model (LLM) tasks. Training them over edge networks is generally considered to be impractical due to their large sizes. Here, we examine the extension of HIST to fine-tuning such models for downstream tasks.

Consider applying the popular Low-Rank Adaptation (LoRA) [49] technique to fine-tune a pre-trained transformer model on data spread across a set of edge devices, e.g., for LLM personalization. In the forward pass of a LoRA module, the computation follows the transformation $\boldsymbol{y} = \boldsymbol{B}\boldsymbol{A}\boldsymbol{z}$, where $\boldsymbol{A}$ and $\boldsymbol{B}$ are the down-projection and up-projection matrices, respectively, and $\pm z$ is the input. This can be viewed as a fully connected network with a single hidden layer, where $\boldsymbol{A}$ and $\boldsymbol{B}$ represent the weight matrices of the first and second layers, respectively. To extend model partitioning to LoRA, we can introduce a sparse diagonal matrix $\boldsymbol{S}$ with diagonal elements restricted to 0 or 1, between $\boldsymbol{A}$ and $\boldsymbol{B}$, which functions similarly to the mask in partitioning hidden neurons, i.e., $\boldsymbol{X}_l = \boldsymbol{B}\boldsymbol{S}_l\boldsymbol{A}$ assigns the submatrix of parameters for partition l. For example, if the 2nd and 4th elements of the diagonal matrix $\boldsymbol{S}_l$ are non-zero, this is equivalent to $\boldsymbol{X}_l$ possessing the 2nd and 4th rows of $\boldsymbol{A}$ and the 2nd and 4th columns of $\boldsymbol{B}$, i.e., the 2nd and 4th neurons in the hidden layer.

Experiments. We consider fine-tuning GPT-2 small, which is a version of the GPT-2 architecture [50], a popular decoder-only transformer model available in 5 sizes. GPT-2 is designed for tasks involving text generation and understanding, and the "small" version has around 124 million parameters, making it more computationally manageable while still powerful for various LLM tasks. In our experiments, we applied this model to the 20 Newsgroups dataset (http://qwone.com/ jason/20Newsgroups), a widely used benchmark for text classification. The dataset consists of documents from 20 different categories, making it an appropriate test case for evaluating how well the model can classify complex text data. By partitioning the LoRA module across the transformer layers of GPT-2, we aim to enhance communication efficiency during fine-tuning, optimizing the model's performance for the Newsgroups classification task.

Following the setting in [51], we set the rank of the LoRA modules to 16. We then consider two partitioning scenarios: $N = 2$ and $N = 4$. In the first case, where $N = 2$, the LoRA module is split into two smaller modules, each with a rank of 8. Each client is responsible for training one of these smaller modules. In the second case, where $N = 4$, the LoRA module is divided into four smaller modules, each with a rank of 4, and each client trains one of these modules. In both cases, we set the number of local training rounds $H = 40$ and the number of communication rounds $E = 5$. We compared the communication costs required by the proposed HIST and HFedAvg algorithms to achieve target accuracies of 70% and 76%. The communication cost per client for reaching these accuracies is presented in Table III. Across all the data and cell configurations considered, we draw the same conclusions as with the fully connected and convolutional neural network models from Sec. VI-B: HIST consistently requires less communication, and the improvement is more pronounced as N is increased.

Fine-tuning GPT-2 Model on 20 Newsgroups Dataset.								
	Fully non-i.i.d.				Cell i.i.d., client non-i.i.d.			
	2 Cells		4 Cells		2 Cells		4 Cells	
Accuracy	HIST	HFedAvg	HIST	HFedAvg	HIST	HFedAvg	HIST	HFedAvg
70%	50.63	78.75	33.75	83.25	32.63	56.25	21.94	60.75
76%	168.75	258.75	100.69	267.75	111.38	166.50	72.56	177.45

TABLE III: Communication cost (MB) per client for achieving testing accuracy of 70% and 76% for fine-tuning GPT-2 small model on 20 Newsgroups dataset. The results show that the HIST algorithm obtains improvements for tasks beyond training fully connected and convolutional neural networks, such as LoRA fine-tuning.

B. Proof of Theorems 1 - 3

For analysis, we introduce virtual iterate $\hat{\boldsymbol{x}}^{t,e} = \sum_{j=1}^{N} \bar{\boldsymbol{x}}_j^{t,e}$, which denotes the virtual synchronized global model of the proposed HIST algorithm. We denote $\hat{\boldsymbol{x}}_{i,h}^{t,e}, \forall i \in \mathcal{C}_j$ as virtual local model for client i, defined as

$$\hat{\boldsymbol{x}}_{i,h+1}^{t,e} = \hat{\boldsymbol{x}}_{i,h}^{t,e} - \gamma \boldsymbol{p}_j^t \odot \nabla l(\hat{\boldsymbol{x}}_{i,h}^{t,e}, \xi_{i,h}^{t,e}), h = 0, 1, \ldots, H-1,$$

and $\hat{\boldsymbol{x}}_{i,0}^{t,e} = \boldsymbol{x}_{i,0}^{t,e}$. Due to the uniform sampling, we have

$$\mathbb{E}[\frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \hat{\boldsymbol{x}}_{i,h}^{t,e}] = \frac{1}{|\mathcal{C}_j|} \sum_{i \in \mathcal{C}_j} \boldsymbol{x}_{i,h}^{t,e}.$$

For notational ease, we define

$$\begin{aligned} D_t &= \sum_{j=1}^{N} \sum_{e=0}^{E-1} \mathbb{E} \left\| \hat{\boldsymbol{x}}^{t,e} - \bar{\boldsymbol{x}}_j^{t,e} \right\|^2 \text{ and} \\ Q_t &= \sum_{j=1}^{N} \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \frac{1}{H} \sum_{e=0}^{E-1} \sum_{h=0}^{H-1} \mathbb{E} \left\| \bar{\boldsymbol{x}}_j^{t,e} - \hat{\boldsymbol{x}}_{i,h}^{t,e} \right\|^2. \end{aligned} \tag{40}$$

The proofs of Theorems 1, 2, and 3 rely on the following four lemmas, which are in turn proven in the supplementary material.

Lemma 1. *Suppose that Assumptions 2-5 hold, $N \geq 2$, and $\gamma \leq \frac{1}{2HL}$. Then,*

*1) **With full participation**, the iterates generated by the* `HIST` *algorithm satisfy*

$$\begin{aligned} \mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})] \leq & \mathbb{E}[f(\bar{\boldsymbol{x}}^{t})] - \frac{\gamma H}{2} \sum_{e=0}^{E-1} \mathbb{E} \left\| \nabla f(\hat{\boldsymbol{x}}^{t,e}) \right\|^2 + \gamma^2 EH\tilde{N} L \sigma^2 \\ & + \frac{3\gamma}{2} \frac{EHNd_{max}}{d} \delta_1^2 + \frac{3H\gamma L^2}{2} \left(D_t + Q_t \right). \end{aligned} \tag{41}$$

*2) **With partial participation**, the iterates generated by the* `HIST` *algorithm satisfy*

$$\begin{aligned} \mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})] \leq & \mathbb{E}[f(\bar{\boldsymbol{x}}^{t})] - \frac{\gamma H}{2} \sum_{e=0}^{E-1} \mathbb{E} \left\| \nabla f(\hat{\boldsymbol{x}}^{t,e}) \right\|^2 + \gamma^2 EH\tilde{N}' L \sigma^2 \\ & + \frac{3\gamma}{2} \frac{EHNd_{max}}{d} \delta_1^2 + 3H\gamma L^2 \left(D_t + Q_t \right) \\ & + 3\gamma^2 EH^2 \tilde{N}' L \delta_2^2. \end{aligned} \tag{42}$$

Lemma 2. *Suppose that Assumptions 2-5 hold, $N \geq 2$, and $\gamma \leq \frac{1}{2HL}$. Then with full participation, the iterates generated by the AirComp-assisted* `HIST` *algorithm satisfy*

$$\begin{aligned} \mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})] \leq & \mathbb{E}[f(\bar{\boldsymbol{x}}^{t})] - \frac{\gamma H}{2} \sum_{e=0}^{E-1} \mathbb{E} \left\| \nabla f(\hat{\boldsymbol{x}}^{t,e}) \right\|^2 + \gamma^2 EH\tilde{N} L \sigma^2 \\ & + \frac{3\gamma}{2} \frac{EHNd_{max}}{d} \delta_1^2 + \frac{1}{2} \sum_{e=0}^{E-1} \sum_{j=1}^{N} \mathrm{MSE}_j^{t,e} \\ & + \frac{3H\gamma L^2}{2} \left(D_t + Q_t \right). \end{aligned} \tag{43}$$

Lemma 3. *Suppose that Assumptions 2-5 hold and $\gamma \leq \frac{1}{EL\sqrt{54(N+1)}}$. Then the difference between the edge models and the global model can be bounded as*

$$\begin{aligned} D_t \leq & 162\gamma^2 (N-1) E^2 H^2 \sum_{e=0}^{E-1} \mathbb{E} \| \nabla f(\hat{\boldsymbol{x}}^{t,e}) \|^2 + 4E(N-1) \mathbb{E} \left\| \bar{\boldsymbol{x}}^{t} \right\|^2 \\ & + 18\gamma^2 (N+1) E^2 H^2 L^2 Q_t + 6\gamma^2 (N+1) E^2 H \tilde{N}' \sigma^2 \\ & + 108\gamma^2 (N+1) E^3 H^2 \frac{Nd_{max}}{d} \delta_1^2. \end{aligned} \tag{44}$$

Lemma 4. *Suppose that Assumptions 2-4 and 6 hold and $\gamma \leq \frac{1}{\sqrt{15}HL}$. Then the difference between the local models and the edge models can be bounded as*

$$\begin{aligned} Q_t \leq & 5\gamma^2 H^2 L^2 D_t + 5\gamma^2 NH^2 \sum_{e=0}^{E-1} \mathbb{E} \left\| \nabla f(\hat{\boldsymbol{x}}^{t,e}) \right\|^2 \\ & + 2\gamma^2 NHE\sigma^2 + 5\gamma^2 NH^2 E\delta_2^2 + 5\gamma^2 NH^2 E\delta_1^2. \end{aligned} \tag{45}$$

Lemmas 1 and 2 characterize the dynamics of the global loss function. Lemmas 3 and 4 characterize the upper bound of the diversity between the virtual global model and edge models and between the edge model and local models, respectively. It is worth noting that Lemmas 3 and 4 are consistent for both full participation and partial participation.

1) Proof of Theorem 1: With the first part of Lemma 1, Lemmas 3 and 4, we can prove Theorem 1 as follows. Based on Lemmas 3 and 4, we have

$$D_t+Q_t \leq \tilde{\alpha}\Bigg\{162\gamma^2(N-1)E^2H^2\sum_{e=0}^{E-1}\mathbb{E}\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\|^2+4E(N-1)\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2+6\gamma^2(N+1)E^2H\tilde{N}\sigma^2 \\ +108\gamma^2(N+1)E^3H^2\frac{Nd_{\max}}{d}\delta_1^2\Bigg\}+\tilde{\beta}\Big\{2\gamma^2NHE\sigma^2+5\gamma^2NH^2E\delta_2^2+5\gamma^2NH^2E\delta_1^2+5\gamma^2NH^2\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2\Big\}, \quad (46)$$

where $\tilde{\alpha}=\frac{1+5\gamma^2H^2L^2}{1-18\gamma^2(N+1)E^2H^2L^2*5\gamma^2H^2L^2}$ and $\tilde{\beta}=\frac{1+18\gamma^2(N+1)E^2H^2L^2}{1-18\gamma^2(N+1)E^2H^2L^2*5\gamma^2H^2L^2}$. According to the setting of γ, one can claim $\tilde{\alpha}\leq 2,\ \tilde{\beta}\leq 2$. We thus have

$$D_t+Q_t \leq \gamma^2\left(324(N-1)E^2H^2+10NH^2\right)\sum_{e=0}^{E-1}\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\|^2+8E(N-1)\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2+\gamma^2\left((12(N+1)E^2H\tilde{N}+4NHE\right)\sigma^2 \\ +\gamma^2\left(216(N+1)\frac{Nd_{\max}}{d}E^3H^2+10NH^2E\right)\delta_1^2+10\gamma^2NH^2E\delta_2^2. \quad (47)$$

Additionally, according to (41) of Lemma 1, we obtain

$$\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \leq \frac{2}{\gamma H}\mathbb{E}[f(\bar{\boldsymbol{x}}^t)]-\mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})]+2\gamma E\tilde{N}L\sigma^2+3\frac{ENd_{\max}}{d}\delta_1^2+3L^2(D_t+Q_t). \quad (48)$$

Plugging (47) into (48), we can write

$$\left(1-\gamma^2L^2\left(972(N-1)E^2H^2+30NH^2\right)\right)\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \\ \leq 2\frac{\mathbb{E}[f(\bar{\boldsymbol{x}}^t)]-\mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})]}{\gamma H}+2\gamma E\tilde{N}L\sigma^2+3\frac{ENd_{\max}}{d}\delta_1^2+\gamma^2L^2\left((36(N+1)E^2H\tilde{N}+12NHE\right)\sigma^2 \\ +\gamma^2L^2\left(648(N+1)\frac{Nd_{\max}}{d}E^3H^2+30NH^2E\right)\delta_1^2+30\gamma^2L^2NH^2E\delta_2^2+24E(N-1)L^2\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2. \quad (49)$$

Based on the condition of γ (i.e., (12)), we have $1-\gamma^2L^2\left(972(N-1)E^2H^2+30NH^2\right)\leq\frac{1}{2}$. As a result, we obtain

$$\frac{1}{E}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \leq 4\frac{\mathbb{E}[f(\bar{\boldsymbol{x}}^t)]-\mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})]}{\gamma EH}+\left(4\gamma\tilde{N}L+\gamma^2L^2\left((72(N+1)EH\tilde{N}+24NH\right)\right)\sigma^2 \\ +\gamma^2L^2\left(1296(N+1)\frac{Nd_{\max}}{d}E^2H^2+60NH^2\right)\delta_1^2 \\ +60\gamma^2L^2NH^2\delta_2^2+6\frac{Nd_{\max}}{d}\delta_1^2+48(N-1)L^2\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2. \quad (50)$$

Utilizing the condition of γ again, we obtain Theorem 1.

2) Proof of Theorem 2: With the second part of Lemma 1, Lemmas 3 and 4, we can prove Theorem 1 as follows. First, according to (42) of Lemma 1, we obtain

$$\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \leq \frac{2}{\gamma H}\mathbb{E}[f(\bar{\boldsymbol{x}}^t)]-\mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})]+2\gamma E\tilde{N}'L\sigma^2+6\gamma EH\tilde{N}'L\delta_2^2+3\frac{ENd_{\max}}{d}\delta_1^2+6L^2(D_t+Q_t). \quad (51)$$

The bound for D_t+Q_t derived in (47) is based on Lemmas 3 and 4, which are consistent for both full and partial client participation. Therefore, we can directly apply this bound here by substituting $\tilde{N}$ with $\tilde{N}'$.

Plugging (47) into (48) gives rise to

$$\left(1-2\gamma^2L^2\left(972(N-1)E^2H^2+30NH^2\right)\right)\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \\ \leq 2\frac{\mathbb{E}[f(\bar{\boldsymbol{x}}^t)]-\mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})]}{\gamma H}+2\gamma E\tilde{N}'L\sigma^2+6\gamma EH\tilde{N}'L\delta_2^2+3\frac{ENd_{\max}}{d}\delta_1^2+2\gamma^2L^2\left((36(N+1)E^2H\tilde{N}+12NHE\right)\sigma^2 \\ +2\gamma^2L^2\left(648(N+1)\frac{Nd_{\max}}{d}E^3H^2+30NH^2E\right)\delta_1^2+60\gamma^2L^2NH^2E\delta_2^2+48E(N-1)L^2\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2. \quad (52)$$

Based on the condition of γ, we have $1-2\gamma^2L^2\left(972(N-1)E^2H^2+30NH^2\right)\leq\frac{1}{2}$. As a result, we obtain

$$\frac{1}{E}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \leq 4\frac{\mathbb{E}[f(\bar{\boldsymbol{x}}^t)]-\mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})]}{\gamma EH}+\left(4\gamma\tilde{N}'L+2\gamma^2L^2\left((72(N+1)EH\tilde{N}+24NH\right)\right)\sigma^2 \\ +2\gamma^2L^2\left(1296(N+1)\frac{Nd_{\max}}{d}E^2H^2+60NH^2\right)\delta_1^2+120\gamma^2L^2NH^2\delta_2^2+12\gamma H\tilde{N}'L\delta_2^2$$

$$+6\frac{Nd_{\max}}{d}\delta_1^2+96(N-1)L^2\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2. \tag{53}$$

Based on the setting of γ in (16), it follows that

$$\begin{aligned}\frac{1}{TE}\sum_{t=0}^{T}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \leq& 4\frac{f(\bar{\boldsymbol{x}}^0)-f_*}{\gamma TEH}+100\gamma\tilde{N}'L\sigma^2+1356\gamma L\delta_1^2+60\gamma L\delta_2^2+12\gamma H\tilde{N}'L\delta_2^2\\&+6\frac{N}{d}\frac{1}{T}\sum_{t=0}^{T-1}d_{\max}^t\delta_1^2+96(N-1)L^2\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2.\end{aligned}$$

Setting $\gamma=(TEH)^{-\frac{1}{2}}$ and ignoring constant multiplicative factors (including L), we have

$$\begin{aligned}\frac{1}{TE}\sum_{t=0}^{T}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \leq& 4\frac{f(\bar{\boldsymbol{x}}^0)-f_*}{\gamma TEH}+100\gamma\tilde{N}'L\sigma^2+1356\gamma L\delta_1^2+60\gamma L\delta_2^2+12\gamma H\tilde{N}'L\delta_2^2\\&+6\frac{N}{d}\frac{1}{T}\sum_{t=0}^{T-1}d_{\max}^t\delta_1^2+96(N-1)L^2\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2\\\sim&\mathcal{O}\left(\tilde{N}'(TEH)^{-\frac{1}{2}}\right)+\mathcal{O}\left((TEH)^{-\frac{1}{2}}\right)+\mathcal{O}\left(\tilde{N}'H^{\frac{1}{2}}(TE)^{-\frac{1}{2}}\right)\\&+\mathcal{O}\left(\frac{N}{d}\frac{1}{T}\sum_{t=0}^{T-1}d_{\max}^t\delta_1^2+(N-1)\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2\right).\end{aligned}$$

This completes the proof of Theorem 2.

3) Proof of Theorem 3: With Lemmas 2, 3, and 4, we can prove Theorem 3 following the same steps as in the proof of Theorem 1. Thus, the detailed proof is omitted here for brevity.

C. Proof of Lemmas

Before providing the proofs of lemmas, we introduce some notations. Denote $\mathbb{E}_t^e$ as an expectation conditioned on the historical information up to the start of the (t,e)-th round. Let $\boldsymbol{g}_{i,h}^{t,e}$ denote the stochastic gradient $\nabla l(\hat{\boldsymbol{x}}_{i,h}^{t,e},\xi_{i,h}^{t,e})$, $\xi_{i,h}^{t,e}\sim\mathcal{D}_i$. As $\hat{\boldsymbol{x}}_{i,h}^{t,e}=\boldsymbol{x}_{i,h}^{t,e}$, $i\in\mathcal{C}_j$, $\boldsymbol{g}_{i,h}^{t,e}$ is equivalent to the stochastic gradient $\nabla l(\boldsymbol{x}_{i,h}^{t,e},\xi_{i,h}^{t,e})$ of active client $i\in\mathcal{C}_j^{t,e}$. In addition, we present three propositions that will be used in this subsection.

Proposition 1. *Suppose that mask $\boldsymbol{p}_j$ is randomly generated via rule (3) and $\|\boldsymbol{p}_j\|_1=d_j$, then*

$$\mathbb{E}[\boldsymbol{p}_j\odot\boldsymbol{z}]=\frac{d_j}{d}\|\boldsymbol{z}\|^2,\ \forall j \text{ and } \sum_{j=1}^{N}\left\|\boldsymbol{p}_j^t\odot(\nabla f(\boldsymbol{x})-\nabla f_j(\boldsymbol{x}))\right\|^2\leq\frac{Nd_{max}}{d}\delta_1^2,\forall\boldsymbol{x},$$

where $d_{max}=\max\{d_1,d_2,\ldots,d_N\}$.

Proof. The former can be derived by the following series of transformations

$$\mathbb{E}[\|\boldsymbol{p}_j\odot\boldsymbol{z}\|^2]=\mathbb{E}[\sum_{k=1}^{d}((p_j)_kz_k)^2]=\mathbb{E}[\sum_{k=1}^{d}((p_j)_kz_k)^2]=\sum_{k=1}^{d}\mathbb{E}[((p_j)_kz_k)^2]=\sum_{k=1}^{d}\frac{d_j}{d}z_k^2=\frac{d_j}{d}\|\boldsymbol{z}\|^2,$$

where $(p_j)_k$ and z_k represent the k-th elements of $\boldsymbol{p}_j$ and $\boldsymbol{z}$, respectively. Furthermore, combining it with Assumption 5 gives rise to the latter. □

Proposition 2. *Any masks $\{\boldsymbol{p}_j\}_{j=1}^N$ generated by rule (3) satisfy*

$$\sum_{j=1}^{N}\|\boldsymbol{p}_j\odot\boldsymbol{z}-\boldsymbol{z}\|^2=(N-1)\|\boldsymbol{z}\|^2.$$

Proof.

$$\sum_{j=1}^{N}\|\boldsymbol{p}_j\odot\boldsymbol{z}-\boldsymbol{z}\|^2=\sum_{j=1}^{N}\{\|\boldsymbol{p}_j\odot\boldsymbol{z}\|-2\langle\boldsymbol{p}_j\odot\boldsymbol{z},\boldsymbol{z}\rangle+\|\boldsymbol{z}\|^2\}=(N-1)\|\boldsymbol{z}\|^2.$$

□

Proposition 3. *For the following inequalities,*

$$\begin{aligned}x&\leq\alpha y+a\\y&\leq\beta x+b,\end{aligned}\tag{54}$$

where $\alpha\beta<1$, we have $x+y\leq\frac{1+\beta}{1-\alpha\beta}a+\frac{1+\alpha}{1-\alpha\beta}b$.

Based on the above propositions, we proof the lemmas.

1) Proof of Lemma 1: Based on the virtual iteration $\hat{\boldsymbol{x}}^{t,e+1} = \hat{\boldsymbol{x}}^{t,e} - \gamma \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,e}$ and Assumption 2, we have

$$\mathbb{E}_t^e[f(\hat{\boldsymbol{x}}^{t,e+1})] \leq f(\hat{\boldsymbol{x}}^{t,e}) \underbrace{-\gamma \mathbb{E}_t^e \left\langle \nabla f(\hat{\boldsymbol{x}}^{t,e}), \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,e} \right\rangle}_{T_1} + \gamma^2 L \underbrace{\frac{1}{2} \mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,e} \right\|^2}_{T_2}. \tag{55}$$

Utilizing $\mathbb{E}_t^e \left[\frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,e} - \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) \right] = 0$ and $\mathbb{E}_t^e \left[\frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) - \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right] = 0$, we rewrite T_1 as follows

$$\begin{aligned} T_1 =& -\gamma H \mathbb{E}_t^e \left\langle \nabla f(\hat{\boldsymbol{x}}^{t,e}), \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \frac{1}{H} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\rangle \\ =& \frac{\gamma H}{2} \left\{ \underbrace{\mathbb{E}_t^e \left\| \nabla f(\hat{\boldsymbol{x}}^{t,e}) - \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \frac{1}{H} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2}_{T_3} - \left\| \nabla f(\hat{\boldsymbol{x}}^{t,e}) \right\|^2 - \mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \frac{1}{H} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2 \right\}. \end{aligned} \tag{56}$$

Based on the facts that $\nabla f(\hat{\boldsymbol{x}}^{t,e}) = \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \nabla f(\hat{\boldsymbol{x}}^{t,e})$ and $\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \boldsymbol{z}_j \|^2 = \sum_{j=1}^{N} \| \boldsymbol{p}_j^t \odot \boldsymbol{z}_j \|^2$, we can bound T_3 as

$$\begin{aligned} T_3 =& \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \boldsymbol{p}_j^t \odot \left(\nabla f(\hat{\boldsymbol{x}}^{t,e}) - \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \frac{1}{H} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right) \right\|^2 \\ \leq& \frac{1}{H} \sum_{h=0}^{H-1} \underbrace{\sum_{j=1}^{N} \mathbb{E}_t^e \left\| \boldsymbol{p}_j^t \odot \left(\nabla f(\hat{\boldsymbol{x}}^{t,e}) - \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right) \right\|^2}_{T_3'}, \end{aligned} \tag{57}$$

where the inequality follows Jensen's inequality. Inserting $\mp \nabla f_j(\hat{\boldsymbol{x}}^{t,e}) \mp \nabla f_j(\bar{\boldsymbol{x}}_j^{t,e})$ into T_3' and calling Cauchy-Schwartz inequality, we have

$$\begin{aligned} T_3' \leq& 3 \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \boldsymbol{p}_j^t \odot \left(\nabla f(\hat{\boldsymbol{x}}^{t,e}) - \nabla f_j(\hat{\boldsymbol{x}}^{t,e}) \right) \right\|^2 + 3 \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \boldsymbol{p}_j^t \odot \left(\nabla f_j(\hat{\boldsymbol{x}}^{t,e}) - \nabla f_j(\bar{\boldsymbol{x}}_j^{t,e}) \right) \right\|^2 \\ &+ 3 \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \boldsymbol{p}_j^t \odot \left(\nabla f_j(\bar{\boldsymbol{x}}_j^{t,e}) - \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right) \right\|^2. \end{aligned} \tag{58}$$

Given Proposition 1, it follows that

$$3 \sum_{j=1}^{N} \left\| \boldsymbol{p}_j^t \odot \left(\nabla f(\hat{\boldsymbol{x}}^{t,e}) - \nabla f_j(\hat{\boldsymbol{x}}^{t,e}) \right) \right\|^2 \leq 3 \frac{N d_{\max}}{d} \delta_1^2. \tag{59}$$

In addition, recalling Jensen's inequality, we have

$$\begin{aligned} \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \boldsymbol{p}_j^t \odot \left(\nabla f_j(\bar{\boldsymbol{x}}_j^{t,e}) - \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right) \right\|^2 =& \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \boldsymbol{p}_j^t \odot \left(\frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \nabla F_i(\bar{\boldsymbol{x}}_j^{t,e}) - \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right) \right\|^2 \\ \leq& \sum_{j=1}^{N} \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \mathbb{E}_t^e \left\| \nabla F_i(\bar{\boldsymbol{x}}_j^{t,e}) - \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2. \end{aligned} \tag{60}$$

Combining (57), (58), (59), and (60), and then utilizing the smoothness of F_i and f_j defined in Assumption 2, we have

$$T_3 = 3 \frac{N d_{\max}}{d} \delta_1^2 + 3L^2 \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \hat{\boldsymbol{x}}^{t,e} - \bar{\boldsymbol{x}}_j^{t,e} \right\|^2 + 3L^2 \sum_{j=1}^{N} \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \frac{1}{H} \sum_{h=0}^{H-1} \mathbb{E}_t^e \left\| \bar{\boldsymbol{x}}_j^{t,e} - \hat{\boldsymbol{x}}_{i,h}^{t,e} \right\|^2. \tag{61}$$

Resorting to Cauchy-Schwartz inequality, we bound T_2 as

$$T_2 \leq \underbrace{\mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,e} - \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) \right\|^2}_{T_2^l} + \mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) \right\|^2 . \tag{62}$$

Denoting $\boldsymbol{a}_i = \sum_{h=0}^{H-1} \left(\boldsymbol{g}_{i,h}^{t,e} - \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) \right)$, we have

$$\begin{aligned} T_2^l &= \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \boldsymbol{a}_i \right\|^2 = \sum_{j=1}^{N} \mathbb{E}_t^e \left[\frac{1}{|\mathcal{C}_j^{t,e}|^2} \sum_{i \in \mathcal{C}_j^{t,e}} \left\| \boldsymbol{p}_j^t \odot \boldsymbol{a}_i \right\|^2 \right] \leq \sum_{j=1}^{N} \frac{1}{n_j |\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j} \mathbb{E}_t^e \|\boldsymbol{a}_i\|^2 \\ &= \sum_{j=1}^{N} \frac{1}{n_j |\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j} \sum_{h=0}^{H-1} \mathbb{E}_t^e \|\boldsymbol{g}_{i,h}^{t,e} - \nabla F_i(\boldsymbol{x}_{i,h}^{t,e})\|^2 \leq \sum_{j=1}^{N} \frac{1}{|\mathcal{C}_j^{t,e}|} H \sigma^2 = H \tilde{N}' \sigma^2, \end{aligned} \tag{63}$$

where the first equality holds because there is no overlapping between any two different masks in the same round, the second equality follows the fact that $\mathbb{E}_t^e[\boldsymbol{a}_i] = 0$ and $\boldsymbol{a}_i$ is independent of $\boldsymbol{a}_j$ for any $j \neq i$, the third equality follows [6, Lemma 2], the first inequality is due to the uniform client participation and $\|\boldsymbol{p} \odot \boldsymbol{z}\|^2 \leq \|\boldsymbol{z}\|^2$, and the second inequality comes from Assumption 4.

We thus obtain

$$T_2 \leq H \tilde{N}' \sigma^2 + \mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) \right\|^2 . \tag{64}$$

1) For **full participation case**, $\mathcal{C}_j^{t,e} = \mathcal{C}_j$, $\boldsymbol{x}_{i,h}^{t,e} = \hat{\boldsymbol{x}}_{i,h}^{t,e}$, and $\tilde{N}' = \tilde{N}$, by combining (55), (56), (61), and (64), we obtain

$$\begin{aligned} \mathbb{E}_t^e[f(\hat{\boldsymbol{x}}^{t,e+1})] \leq & f(\hat{\boldsymbol{x}}^{t,e}) - \frac{\gamma H}{2} \mathbb{E}_t^e \left\| \nabla f(\hat{\boldsymbol{x}}^{t,e}) \right\|^2 + \gamma^2 H \tilde{N} L \sigma^2 + \frac{3\gamma H}{2} \frac{N d_{\max}}{d} \delta_1^2 \\ & + \frac{3\gamma H L^2}{2} \left\{ \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \hat{\boldsymbol{x}}^{t,e} - \bar{\boldsymbol{x}}_j^{t,e} \right\|^2 + \sum_{j=1}^{N} \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \frac{1}{H} \sum_{h=0}^{H-1} \mathbb{E}_t^e \left\| \bar{\boldsymbol{x}}_j^{t,e} - \boldsymbol{x}_{i,h}^{t,e} \right\|^2 \right\} . \end{aligned} \tag{65}$$

Taking an expectation over all the randomness for the above inequality and telescoping it from $e = 0$ to $e = H - 1$, we will obtain (41) in Lemma 1.

2) For **partial participation case**, we need to further bound T_2:

$$T_2 \leq H \tilde{N}' \sigma^2 + \underbrace{\mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) \right\|^2}_{T_2^r} . \tag{66}$$

For T_2^r, we have

$$\begin{aligned} T_2^r &= \mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) \mp \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2 \\ &= \mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) - \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2 + \mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2 \\ &= \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) - \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2 + \mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2 \\ &\leq \sum_{j=1}^{N} \mathbb{E}_t^e \left\| \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i \in \mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,e}) - \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2 + \mathbb{E}_t^e \left\| \sum_{j=1}^{N} \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i \in \mathcal{C}_j} \sum_{h=0}^{H-1} \nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e}) \right\|^2 . \end{aligned} \tag{67}$$

Furthermore, by substituting $\mp\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\sum_{h=0}^{H-1}\nabla F_i(\bar{\boldsymbol{x}}_j^{t,e})$ and $\mp\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\sum_{h=0}^{H-1}\nabla F_i(\bar{\boldsymbol{x}}_j^{t,e})$ into the first term of the above equality and utilizing Jensen's inequality and Assumption 2, we have

$$\begin{aligned}
&\sum_{j=1}^{N}\mathbb{E}_t^e\left\|\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\sum_{h=0}^{H-1}\nabla F_i(\boldsymbol{x}_{i,h}^{t,e})-\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\sum_{h=0}^{H-1}\nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e})\right\|^2\\
\leq&3\sum_{j=1}^{N}\mathbb{E}_t^e\left\|\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\sum_{h=0}^{H-1}\nabla F_i(\bar{\boldsymbol{x}}_j^{t,e})-\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\sum_{h=0}^{H-1}\nabla F_i(\bar{\boldsymbol{x}}_j^{t,e})\right\|^2+6H^2L^2\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\frac{1}{H}\sum_{h=0}^{H-1}\mathbb{E}_t^e\left\|\bar{\boldsymbol{x}}_j^{t,e}-\hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2\\
\leq&3H\sum_{j=1}^{N}\sum_{h=0}^{H-1}\mathbb{E}_t^e\left\|\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\nabla F_i(\bar{\boldsymbol{x}}_j^{t,e})-\nabla f_j(\bar{\boldsymbol{x}}_j^{t,e})\right\|^2+6H^2L^2\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\frac{1}{H}\sum_{h=0}^{H-1}\mathbb{E}_t^e\left\|\bar{\boldsymbol{x}}_j^{t,e}-\hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2\\
\leq&3H^2\sum_{j=1}^{N}\frac{1}{|\mathcal{C}_j^{t,e}|^2}\sum_{i\in\mathcal{C}_j^{t,e}}\mathbb{E}_t^e\left\|\nabla F_i(\bar{\boldsymbol{x}}_j^{t,e})-\nabla f_j(\bar{\boldsymbol{x}}_j^{t,e})\right\|^2+6H^2L^2\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\frac{1}{H}\sum_{h=0}^{H-1}\mathbb{E}_t^e\left\|\bar{\boldsymbol{x}}_j^{t,e}-\hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2\\
=&3H^2\tilde{N}'\delta_2^2+6H^2L^2\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\frac{1}{H}\sum_{h=0}^{H-1}\mathbb{E}_t^e\left\|\bar{\boldsymbol{x}}_j^{t,e}-\hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2,
\end{aligned}\tag{68}$$

where we use $\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\mathbb{E}_t^e\left\|\bar{\boldsymbol{x}}_j^{t,e}-\hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2=\mathbb{E}_t^e\left[\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\left\|\bar{\boldsymbol{x}}_j^{t,e}-\boldsymbol{x}_{i,h}^{t,e}\right\|^2\right]$ in the first inequality, and the third inequality follows by the fact that clients in $\mathcal{C}_j^{t,e}$ are independently sampled from $\mathcal{C}_j$ and $\mathbb{E}[\nabla F_i(\bar{\boldsymbol{x}}_j^{t,e})]=\nabla f_j(\bar{\boldsymbol{x}}_j^{t,e})$.

Based on (66), (67), and (68), we can derive an upper bound for T_2 as

$$T_2\leq H\tilde{N}'\sigma^2+\mathbb{E}_t^e\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\sum_{h=0}^{H-1}\nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,e})\right\|^2+3H^2\tilde{N}'\delta_2^2+6H^2L^2\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\frac{1}{H}\sum_{h=0}^{H-1}\mathbb{E}_t^e\left\|\bar{\boldsymbol{x}}_j^{t,e}-\hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2.\tag{69}$$

Combining (55), (56), (61), and (69), and utilizing $\gamma\leq\frac{1}{4HL}$, we obtain

$$\begin{aligned}
\mathbb{E}_t^e[f(\hat{\boldsymbol{x}}^{t,e+1})]\leq&f(\hat{\boldsymbol{x}}^{t,e})-\frac{\gamma H}{2}\mathbb{E}_t^e\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2+\gamma^2H\tilde{N}'L\sigma^2+3\gamma^2H^2L\tilde{N}'\delta_2^2+\frac{3\gamma H}{2}\frac{Nd_{\max}}{d}\delta_1^2\\
&+\frac{3\gamma HL^2}{2}\left\{\sum_{j=1}^{N}\mathbb{E}_t^e\left\|\hat{\boldsymbol{x}}^{t,e}-\bar{\boldsymbol{x}}_j^{t,e}\right\|^2+\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\frac{1}{H}\sum_{h=0}^{H-1}\mathbb{E}_t^e\left\|\bar{\boldsymbol{x}}_j^{t,e}-\hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2\right\}\\
&+6\gamma^2H^2L^3\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\frac{1}{H}\sum_{h=0}^{H-1}\mathbb{E}_t^e\left\|\bar{\boldsymbol{x}}_j^{t,e}-\hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2\\
\leq&f(\hat{\boldsymbol{x}}^{t,e})-\frac{\gamma H}{2}\mathbb{E}_t^e\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2+\gamma^2H\tilde{N}'L\sigma^2+3\gamma^2H^2L\tilde{N}'\delta_2^2+\frac{3\gamma H}{2}\frac{Nd_{\max}}{d}\delta_1^2\\
&+3\gamma HL^2\left\{\sum_{j=1}^{N}\mathbb{E}_t^e\left\|\hat{\boldsymbol{x}}^{t,e}-\bar{\boldsymbol{x}}_j^{t,e}\right\|^2+\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\frac{1}{H}\sum_{h=0}^{H-1}\mathbb{E}_t^e\left\|\bar{\boldsymbol{x}}_j^{t,e}-\hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2\right\}.
\end{aligned}\tag{70}$$

Taking an expectation over all the randomness for the above inequality and telescoping it from $e=0$ to $e=H-1$, we obtain (42) in Lemma 1:

$$\begin{aligned}
\mathbb{E}[f(\bar{\boldsymbol{x}}^{t+1})]\leq&\mathbb{E}[f(\bar{\boldsymbol{x}}^{t})]-\frac{\gamma H}{2}\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2+\gamma^2EH\tilde{N}'L\sigma^2\\
&+3\gamma^2EH^2L\tilde{N}'\delta_2^2+\frac{3\gamma}{2}\frac{EHNd_{\max}}{d}\delta_1^2+3H\gamma L^2\left(D_t+Q_t\right).
\end{aligned}\tag{71}$$

2) Proof of Lemma 2: With AirComp, the iteration becomes

$$\hat{\boldsymbol{x}}^{t,e+1}=\hat{\boldsymbol{x}}^{t,e}-\gamma\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\sum_{h=0}^{H-1}\boldsymbol{g}_{i,h}^{t,e}+\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\boldsymbol{n}_j^{t,e}.\tag{72}$$

Note that $\boldsymbol{p}_j^t \odot \boldsymbol{n}_j^{t,e} = \boldsymbol{n}_j^{t,e}$ holds. According to Assumption 2, we have

$$\begin{aligned}\mathbb{E}_t^e[f(\hat{\boldsymbol{x}}^{t,e+1})] \leq & f(\hat{\boldsymbol{x}}^{t,e}) \underbrace{-\gamma \mathbb{E}_t^e \left\langle \nabla f(\hat{\boldsymbol{x}}^{t,e}), \sum_{j=1}^N \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i\in\mathcal{C}_j} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,e} - \sum_{j=1}^N \boldsymbol{p}_j^t \odot \boldsymbol{n}_j^{t,e} \right\rangle}_{G_1} \\ & \underbrace{+ L\frac{1}{2}\mathbb{E}_t^e \left\| \gamma \sum_{j=1}^N \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i\in\mathcal{C}_j} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,e} - \sum_{j=1}^N \boldsymbol{p}_j^t \odot \boldsymbol{n}_j^{t,e} \right\|^2}_{G_2}.\end{aligned} \tag{73}$$

Since $\boldsymbol{n}_j^{t,e}$ is independent of $\boldsymbol{g}_{i,h}^{t,e}$ and $\mathbb{E}[\boldsymbol{n}_j^{t,e}] = 0$, the upper bound of G_1 is the same as T_1 in (55). Additionally, we can rewrite G_2 as

$$G_2 = \frac{1}{2}\gamma^2 \mathbb{E}_t^e \left\| \sum_{j=1}^N \boldsymbol{p}_j^t \odot \frac{1}{n_j} \sum_{i\in\mathcal{C}_j} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,e} \right\|^2 + \frac{1}{2} \sum_{j=1}^N \mathbb{E}\|\boldsymbol{p}_j^t \odot \boldsymbol{n}_j^{t,e}\|^2. \tag{74}$$

Next, following the same step in the proof of Lemma 3, we can obtain

$$\begin{aligned}\mathbb{E}_t^e[f(\hat{\boldsymbol{x}}^{t,e+1})] \leq & f(\hat{\boldsymbol{x}}^{t,e}) - \frac{\gamma H}{2} \mathbb{E}_t^e \left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 + \gamma^2 H \tilde{N} L \sigma^2 + \frac{3\gamma H}{2} \frac{N d_{\max}}{d} \delta_1^2 + \frac{1}{2} \sum_{j=1}^N \mathrm{MSE}_j^{t,e} \\ & + \frac{3\gamma H L^2}{2} \left\{ \sum_{j=1}^N \mathbb{E}_t^e \left\|\hat{\boldsymbol{x}}^{t,e} - \bar{\boldsymbol{x}}_j^{t,e}\right\|^2 + \sum_{j=1}^N \frac{1}{n_j} \sum_{i\in\mathcal{C}_j} \frac{1}{H} \sum_{h=0}^{H-1} \mathbb{E}_t^e \left\|\bar{\boldsymbol{x}}_j^{t,e} - \boldsymbol{x}_{i,h}^{t,e}\right\|^2 \right\},\end{aligned} \tag{75}$$

where $\mathrm{MSE}_j^{t,e} = \mathbb{E}\|\boldsymbol{p}_j^t \odot \boldsymbol{n}_j^{t,e}\|^2$. Taking an expectation over all the randomness for the above inequality and telescoping it from $e = 0$ to $e = H-1$, we will obtain Lemma 2.

3) Proof of Lemma 3: According to the iteration in Algorithm 1, we have

$$\begin{aligned}\mathbb{E}\left\|\hat{\boldsymbol{x}}^{t,e} - \bar{\boldsymbol{x}}_j^{t,e}\right\|^2 = & \mathbb{E}\left\| \bar{\boldsymbol{x}}^t - \gamma \sum_{\tau_1=0}^{e-1} \sum_{j=1}^N \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,\tau_1} - \bar{\boldsymbol{x}}_j^{t,0} + \gamma \sum_{\tau_1=0}^{e-1} \boldsymbol{p}_j^t \odot \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,\tau_1} \right\|^2 \\ \leq & 2\mathbb{E}\left\| \bar{\boldsymbol{x}}^t - \bar{\boldsymbol{x}}_j^{t,0} \right\|^2 + 2\gamma^2 \underbrace{\mathbb{E}\left\| \sum_{j=1}^N \boldsymbol{p}_j^t \odot \sum_{\tau_1=0}^{e-1} \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,\tau_1} - \boldsymbol{p}_j^t \odot \sum_{\tau_1=0}^{e-1} \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \boldsymbol{g}_{i,h}^{t,\tau_1} \right\|^2}_{T_{4,j}^{t,e}}\end{aligned} \tag{76}$$

For $T_{4,j}^{t,e}$, we can obtain

$$\begin{aligned}T_{4,j}^{t,e} = & \mathbb{E}\left\| \sum_{j=1}^N \boldsymbol{p}_j^t \odot \sum_{\tau_1=0}^{e-1} \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \left(\boldsymbol{g}_{i,h}^{t,\tau_1} \mp \nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\right) - \boldsymbol{p}_j^t \odot \sum_{\tau_1=0}^{e-1} \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \left(\boldsymbol{g}_{i,h}^{t,\tau_1} \mp \nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\right) \right\|^2 \\ \leq & \underbrace{3\mathbb{E}\left\| \sum_{j=1}^N \boldsymbol{p}_j^t \odot \sum_{\tau_1=0}^{e-1} \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \left(\boldsymbol{g}_{i,h}^{t,\tau_1} - \nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\right) \right\|^2 + 3\mathbb{E}\left\| \boldsymbol{p}_j^t \odot \sum_{\tau_1=0}^{e-1} \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \left(\boldsymbol{g}_{i,h}^{t,\tau_1} - \nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\right) \right\|^2}_{T_{5,j}^{t,e}} \\ & \underbrace{+ 3\mathbb{E}\left\| \sum_{j=1}^N \boldsymbol{p}_j^t \odot \sum_{\tau_1=0}^{e-1} \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1}) - \boldsymbol{p}_j^t \odot \sum_{\tau_1=0}^{e-1} \frac{1}{|\mathcal{C}_j^{t,e}|} \sum_{i\in\mathcal{C}_j^{t,e}} \sum_{h=0}^{H-1} \nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1}) \right\|^2}_{T_{6,j}^{t,e}}.\end{aligned} \tag{77}$$

For $T_{5,j}^{t,e}$, we have the following result:

$$\begin{aligned}
T_{5,j}^{t,e} =& 3\sum_{j=1}^{N}\mathbb{E}\left\|\boldsymbol{p}_j^t\odot\sum_{\tau_1=0}^{e-1}\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\sum_{h=0}^{H-1}\left(\boldsymbol{g}_{i,h}^{t,\tau_1}-\nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\right)\right\|^2+3\mathbb{E}\left\|\boldsymbol{p}_j^t\odot\sum_{\tau_1=0}^{e-1}\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\sum_{h=0}^{H-1}\left(\boldsymbol{g}_{i,h}^{t,\tau_1}-\nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\right)\right\|^2\\
\leq& 3\sum_{j=1}^{N}\sum_{\tau_1=0}^{e-1}\frac{1}{|\mathcal{C}_j^{t,e}|^2}\sum_{i\in\mathcal{C}_j^{t,e}}\sum_{h=0}^{H-1}\mathbb{E}\left\|\boldsymbol{g}_{i,h}^{t,\tau_1}-\nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\right\|^2+3\sum_{\tau_1=0}^{e-1}\frac{1}{|\mathcal{C}_j^{t,e}|^2}\sum_{i\in\mathcal{C}_j^{t,e}}\sum_{h=0}^{H-1}\mathbb{E}\left\|\boldsymbol{g}_{i,h}^{t,\tau_1}-\nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\right\|^2\\
\leq& 3eH\sum_{j=1}^{N}\frac{1}{|\mathcal{C}_j^{t,e}|}\sigma^2+3eH\frac{1}{|\mathcal{C}_j^{t,e}|}\sigma^2=\tilde{N}'\sigma^2+3eH\frac{1}{n_j'}\sigma^2,
\end{aligned}\tag{78}$$

where $\tilde{N}'=\sum_{j=1}^{N}\frac{1}{n_j'}$, the first inequality comes from [6, Lemma 2] and the inequality $\|\boldsymbol{p}\odot\boldsymbol{z}\|\leq\|\boldsymbol{z}\|^2$ and the second inequality follows Assumption 4.

For $T_{6,j}^{t,e}$, we have

$$\begin{aligned}
T_{6,j}^{t,e} =& 9eH\sum_{\tau_1=0}^{e-1}\sum_{h=0}^{H-1}\mathbb{E}\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\left(\nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\mp\nabla F_i(\bar{\boldsymbol{x}}_j^{t,\tau_1})\right)-\boldsymbol{p}_j^t\odot\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\left(\nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})\mp\nabla F_i(\bar{\boldsymbol{x}}_j^{t,\tau_1})\right)\right\|^2\\
\leq& 9eH\sum_{\tau_1=0}^{e-1}\sum_{h=0}^{H-1}\mathbb{E}\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\left(\nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})-\nabla F_i(\bar{\boldsymbol{x}}_j^{t,\tau_1})\right)\right\|^2+9eH\sum_{\tau_1=0}^{e-1}\sum_{h=0}^{H-1}\mathbb{E}\left\|\boldsymbol{p}_j^t\odot\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\left(\nabla F_i(\boldsymbol{x}_{i,h}^{t,\tau_1})-\nabla F_i(\bar{\boldsymbol{x}}_j^{t,\tau_1})\right)\right\|^2\\
&+9eH^2\sum_{\tau_1=0}^{e-1}\mathbb{E}\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\nabla F_i(\bar{\boldsymbol{x}}_j^{t,\tau_1})-\boldsymbol{p}_j^t\odot\frac{1}{|\mathcal{C}_j^{t,e}|}\sum_{i\in\mathcal{C}_j^{t,e}}\nabla F_i(\bar{\boldsymbol{x}}_j^{t,\tau_1})\right\|^2\\
\leq& 9eH\sum_{\tau_1=0}^{e-1}\sum_{h=0}^{H-1}\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\mathbb{E}\left\|\nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,\tau_1})-\nabla F_i(\bar{\boldsymbol{x}}_j^{t,\tau_1})\right\|^2+9eH\sum_{\tau_1=0}^{e-1}\sum_{h=0}^{H-1}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\mathbb{E}\left\|\nabla F_i(\hat{\boldsymbol{x}}_{i,h}^{t,\tau_1})-\nabla F_i(\bar{\boldsymbol{x}}_j^{t,\tau_1})\right\|^2\\
&+9eH^2\sum_{\tau_1=0}^{e-1}\mathbb{E}\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\nabla f_j(\bar{\boldsymbol{x}}_j^{t,\tau_1})-\boldsymbol{p}_j^t\odot f_j(\bar{\boldsymbol{x}}_j^{t,\tau_1})\right\|^2\\
\leq& 9eHL^2\sum_{\tau_1=0}^{e-1}\sum_{h=0}^{H-1}\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\mathbb{E}\left\|\hat{\boldsymbol{x}}_{i,h}^{t,\tau_1}-\bar{\boldsymbol{x}}_j^{t,\tau_1}\right\|^2+9eHL^2\sum_{\tau_1=0}^{e-1}\sum_{h=0}^{H-1}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\mathbb{E}\left\|\hat{\boldsymbol{x}}_{i,h}^{t,\tau_1}-\bar{\boldsymbol{x}}_j^{t,\tau_1}\right\|^2\\
&+9eH^2\underbrace{\sum_{\tau_1=0}^{e-1}\mathbb{E}\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\nabla f_j(\bar{\boldsymbol{x}}_j^{t,\tau_1})-\boldsymbol{p}_j^t\odot f_j(\bar{\boldsymbol{x}}_j^{t,\tau_1})\right\|^2}_{T_{7,j}^{t,e}}.
\end{aligned}\tag{79}$$

$$\begin{aligned}
T_{7,j}^{t,e} =& \mathbb{E}\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\left(\nabla f_j(\bar{\boldsymbol{x}}_j^{t,\tau_1})\mp\nabla f_j(\hat{\boldsymbol{x}}^{t,\tau_1})\right)-\boldsymbol{p}_j^t\odot\left(\nabla f_j(\bar{\boldsymbol{x}}_j^{t,\tau_1})\mp f_j(\hat{\boldsymbol{x}}^{t,\tau_1})\right)\right\|^2\\
\leq& 3\mathbb{E}\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\left(\nabla f_j(\bar{\boldsymbol{x}}_j^{t,\tau_1})-\nabla f_j(\hat{\boldsymbol{x}}^{t,\tau_1})\right)\right\|^2+3\left\|\boldsymbol{p}_j^t\odot\left(\nabla f_j(\bar{\boldsymbol{x}}_j^{t,\tau_1})-\nabla f_j(\hat{\boldsymbol{x}}^{t,\tau_1})\right)\right\|^2\\
&+3\mathbb{E}\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\nabla f_j(\hat{\boldsymbol{x}}^{t,\tau_1})-\boldsymbol{p}_j^t\odot\nabla f_j(\hat{\boldsymbol{x}}^{t,\tau_1})\right\|^2\\
\leq& 3L^2\sum_{j=1}^{N}\mathbb{E}\left\|\bar{\boldsymbol{x}}_j^{t,\tau_1}-\hat{\boldsymbol{x}}^{t,\tau_1}\right\|^2+3L^2\left\|\bar{\boldsymbol{x}}_j^{t,\tau_1}-\hat{\boldsymbol{x}}^{t,\tau_1}\right\|^2\\
&+\underbrace{3\mathbb{E}\left\|\sum_{j=1}^{N}\boldsymbol{p}_j^t\odot\left(\nabla f_j(\hat{\boldsymbol{x}}^{t,\tau_1})\mp\nabla f(\hat{\boldsymbol{x}}^{t,\tau_1})\right)-\boldsymbol{p}_j^t\odot\left(\nabla f_j(\hat{\boldsymbol{x}}^{t,\tau_1})\mp\nabla f(\hat{\boldsymbol{x}}^{t,\tau_1})\right)\right\|^2}_{T_{8,j}^{t,e}}.
\end{aligned}\tag{80}$$

Similarly, we further bound $\sum_{j=1}^{N} T_{8,j}^{t,e}$ as follows

$$\begin{aligned}\sum_{j=1}^{N} T_{8,j}^{t,e} \leq & 9(N+1)\sum_{j=1}^{N}\mathbb{E}\left\|\boldsymbol{p}_j^t \odot \left(\nabla f_j(\hat{\boldsymbol{x}}^{t,\tau_1}) - \nabla f(\hat{\boldsymbol{x}}^{t,\tau_1})\right)\right\|^2 + 9\sum_{j=1}^{N}\mathbb{E}\left\|\boldsymbol{p}_j^t \odot \nabla f(\hat{\boldsymbol{x}}^{t,\tau_1}) - \nabla f(\hat{\boldsymbol{x}}^{t,\tau_1})\right\|^2 \\ \leq & 9(N+1)\frac{N d_{\max}}{d}\delta_1^2 + 9(N-1)\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,\tau_1})\right\|^2,\end{aligned} \tag{81}$$

where the second inequality comes from Propositions 1, 2 and Assumption 5.

$$\begin{aligned}\sum_{e=0}^{E-1}\sum_{j=1}^{N} T_{4,j}^{t,e} \leq & \frac{3}{2}(N+1)\tilde{N}' E^2 H \sigma^2 + \frac{9}{2}(N+1)E^2 H L^2 \sum_{e=0}^{E-1}\sum_{h=0}^{H-1}\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\mathbb{E}\left\|\hat{\boldsymbol{x}}_{i,h}^{t,e} - \bar{\boldsymbol{x}}_j^{t,e}\right\|^2 \\ & \frac{27}{2}(N+1)E^2H^2L^2\sum_{e=0}^{E-1}\sum_{j=1}^{N}\mathbb{E}\left\|\bar{\boldsymbol{x}}_j^{t,e} - \hat{\boldsymbol{x}}^{t,e}\right\|^2 \\ & 27(N+1)E^3H^2\frac{N d_{\max}}{d}\delta_1^2 + \frac{81}{2}(N-1)E^2H^2\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2.\end{aligned} \tag{82}$$

Plugging (82) into (76) and utilizing Proposition 2, we obtain

$$\begin{aligned}D_t \leq & 2E(N-1)\mathbb{E}\left\|\bar{\boldsymbol{x}}^t\right\|^2 + 3\gamma^2(N+1)E^2H\tilde{N}'\sigma^2 + 9\gamma^2(N+1)E^2H^2L^2Q_t \\ & + 27\gamma^2(N+1)E^2H^2L^2D_t + 54\gamma^2(N+1)E^3H^2\frac{N d_{\max}}{d}\delta_1^2 \\ & + 81\gamma^2(N-1)E^2H^2\sum_{e=0}^{E-1}\mathbb{E}\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\|^2.\end{aligned} \tag{83}$$

As $1 - 27\gamma^2(N+1)E^2H^2L^2 \geq \frac{1}{2}$ when $\gamma \leq \frac{1}{\sqrt{54(N+1)}EHL}$, we thus obtain Lemma 3.

4) Proof of Lemma 4: Based on the iteration $\bar{\boldsymbol{x}}_j^{t,e} - \hat{\boldsymbol{x}}_{i,h}^{t,e} = \gamma \boldsymbol{p}_j^t \odot \sum_{\tau_2=0}^{h-1} \boldsymbol{g}_{i,\tau_2}^{t,e}$, we can write

$$\begin{aligned}\mathbb{E}\left\|\bar{\boldsymbol{x}}_j^{t,e} - \hat{\boldsymbol{x}}_{i,h}^{t,e}\right\|^2 \leq & 2\gamma^2\mathbb{E}\left\|\sum_{\tau_2=0}^{h-1}\boldsymbol{g}_{i,\tau_2}^{t,e} - \sum_{\tau_2=0}^{h-1}\nabla F_i(\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e})\right\|^2 + 2\gamma^2\mathbb{E}\left\|\sum_{\tau_2=0}^{h-1}\nabla F_i(\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e})\right\|^2 \\ = & 2\gamma^2\sum_{\tau_2=0}^{h-1}\mathbb{E}\left\|\boldsymbol{g}_{i,\tau_2}^{t,e} - \nabla F_i(\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e})\right\|^2 + 2\gamma^2\mathbb{E}\left\|\sum_{\tau_2=0}^{h-1}\nabla F_i(\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e})\right\|^2 \\ \leq & 2\gamma^2 h\sigma^2 + 2\gamma^2 h\sum_{\tau_2=0}^{h-1}\mathbb{E}\left\|\nabla F_i(\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e})\right\|^2,\end{aligned} \tag{84}$$

where the first inequality follows Cauchy-Schwartz inequality, the equality comes from [6, Lemma 2], and the second inequality follows Assumption 4. Additionally, we can bound $\mathbb{E}\left\|\nabla F_i(\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e})\right\|^2$ as

$$\begin{aligned}\mathbb{E}\left\|\nabla F_i(\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e})\right\|^2 = & \mathbb{E}\left\|\nabla F_i(\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e}) \mp \nabla F_i(\bar{\boldsymbol{x}}_j^{t,e}) \mp \nabla f_j(\bar{\boldsymbol{x}}_j^{t,e}) \mp \nabla f(\bar{\boldsymbol{x}}_j^{t,e}) \mp \nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \\ \leq & 5\mathbb{E}\left\|\nabla F_i(\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e}) - \nabla F_i(\bar{\boldsymbol{x}}_j^{t,e})\right\|^2 + 5\mathbb{E}\left\|\nabla F_i(\bar{\boldsymbol{x}}_j^{t,e}) - \nabla f_j(\bar{\boldsymbol{x}}_j^{t,e})\right\| + 5\mathbb{E}\left\|\nabla f_j(\bar{\boldsymbol{x}}_j^{t,e}) - \nabla f(\bar{\boldsymbol{x}}_j^{t,e})\right\|^2 \\ & + 5\mathbb{E}\left\|\nabla f(\bar{\boldsymbol{x}}_j^{t,e}) - \nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 + 5\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 \\ \leq & 5L^2\mathbb{E}\left\|\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e} - \bar{\boldsymbol{x}}_j^{t,e}\right\|^2 + 5\mathbb{E}\left\|\nabla F_i(\bar{\boldsymbol{x}}_j^{t,e}) - \nabla f_j(\bar{\boldsymbol{x}}_j^{t,e})\right\| + 5\mathbb{E}\left\|\nabla f_j(\bar{\boldsymbol{x}}_j^{t,e}) - \nabla f(\bar{\boldsymbol{x}}_j^{t,e})\right\|^2 \\ & + 5L^2\mathbb{E}\left\|\bar{\boldsymbol{x}}_j^{t,e} - \hat{\boldsymbol{x}}^{t,e}\right\|^2 + 5\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2,\end{aligned} \tag{85}$$

where the first inequality comes from Cauchy-Schwartz inequality and the second one follows Assumption 2. Hence, we have

$$\begin{aligned}\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\mathbb{E}\left\|\bar{\boldsymbol{x}}_j^{t,e} - \hat{\boldsymbol{x}}_{i,\tau_2}^{t,e}\right\|^2 \leq & 10\gamma^2 hL^2\sum_{\tau_2=0}^{h-1}\sum_{j=1}^{N}\frac{1}{n_j}\sum_{i\in\mathcal{C}_j}\mathbb{E}\left\|\hat{\boldsymbol{x}}_{i,\tau_2}^{t,e} - \bar{\boldsymbol{x}}_j^{t,e}\right\|^2 + 10\gamma^2h^2L^2\sum_{j=1}^{N}\mathbb{E}\left\|\bar{\boldsymbol{x}}_j^{t,e} - \hat{\boldsymbol{x}}^{t,e}\right\|^2 \\ & + 10\gamma^2Nh^2\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 + 2\gamma^2Nh\sigma^2 + 10\gamma^2Nh^2\delta_2^2 + 10\gamma^2Nh^2\delta_1^2.\end{aligned} \tag{86}$$

Recalling the definitions of Q_t and Q_t in (40), we have

$$Q_t \leq 5\gamma^2H^2L^2Q_t + \frac{10}{3}\gamma^2H^2L^2D_t + \frac{10}{3}\gamma^2NH^2\sum_{e=0}^{E-1}\mathbb{E}\left\|\nabla f(\hat{\boldsymbol{x}}^{t,e})\right\|^2 + \gamma^2NHE\sigma^2 + \frac{10}{3}\gamma^2NH^2E\delta_2^2 + \frac{10}{3}\gamma^2NH^2E\delta_1^2. \tag{87}$$

As $3\left(1 - 5\gamma^2H^2L^2\right) \geq 2$ when $\gamma \leq \frac{1}{\sqrt{15}HL}$, we thus obtain Lemma 4.