# Over-the-Air Clustered Federated Learning

Hasin Us Sami, *Graduate Student Member, IEEE*, and Başak Güler, *Member, IEEE*

***Abstract*— Over-the-air federated learning (FL) is a recent paradigm to address the communication bottleneck of FL, where a machine learning model is trained by aggregating the local gradients directly in the wireless medium. On the other hand, due to the inherent data heterogeneity across wireless users, training a single model to serve all users can severely degrade individual user performance. Towards addressing this challenge, in this work we propose *over-the-air clustered FL*, where multiple models are trained concurrently over-the-air, and each model is adapted gradually to a group of users with similar data distributions. We introduce AirCluster, an over-the-air clustered FL framework with coordinated zero-forcing MIMO beamforming, along with a sketching-based dimensionality reduction mechanism to enable over-the-air training with limited number of antennas. Our theoretical analysis provides formal convergence guarantees for the trained models, while identifying the key performance trade-offs in terms of the convergence rate, compression ratio, channel quality, and the number of antennas. Through extensive experiments on multiple datasets, we observe significant increase in the test accuracy for individual users over state-of-the-art FL benchmarks. Our results demonstrate over-the-air FL to be a promising approach in addressing the communication bottleneck of FL, even under severe data heterogeneity.**

***Index Terms*— Over-the-air machine learning, clustered federated learning, distributed training.**

## I. Introduction

FEDERATED learning (FL) is a distributed framework to train machine learning models over the data collected locally by a large number of wireless edge devices [2]. Training is often coordinated by a server who maintains a *global model*, which is updated iteratively by the wireless devices (users) through local training. The user updates are then aggregated by the server to update the global model. Due to the need for allocating limited spectrum resources across a large number of users (which can reach millions/billions), a major bottleneck in real-world settings is the communication overhead of sending the user updates to the server [3].

Over-the-air FL (OTA-FL) has recently been introduced to address this challenge by utilizing the superposition property of the multi-access channel [4], [5], [6]. OTA-FL aggregates the local updates *over-the-air*, reducing the communication overhead by a factor of the number of users. In contrast to conventional FL, where the server reconstructs each local update to aggregate them, OTA-FL enables aggregation concurrently in the channel. Despite the recent advances in spectrum efficiency, training a single model to serve all users can lead to severe performance drop when the local dataset distributions of the users are heterogeneous [7], [8], [9]. Data heterogeneity leads to slower convergence, and the model tends to favor some users while heavily degrading the performance of others, particularly the underrepresented users [9], [10], [11].

Personalized FL is a recent paradigm to address data heterogeneity in FL, by incorporating the local data characteristics of individual users during model training. Broadly, personalization approaches can be categorized into two subclasses. The first one takes a *user-level* approach, where an individual model is trained for each user, through various techniques such as fine tuning or meta learning [12], [13], [14], [15], [16]. The second one takes a *group-level* approach, where different models are trained for different groups of users with similar data characteristics [7], [17], [18], [19], [20], [21], [22], [23]. Also known as *clustered FL*, this approach iteratively clusters users with respect to their data distributions, while training a separate model for each cluster. Clustered FL allows the server to maintain personalized models for serving users with similar characteristics, while avoiding excessive memory and storage costs for handling a large number of personalized models.

In this work, we introduce AirCluster, an *over-the-air clustered FL* framework, to enable group-level personalization in OTA-FL. OTA-FL allows the server to observe the *sum* of *all user updates*, but the server loses access to the individual updates, hence is not able to separate the updates belonging to different clusters of users. In contrast, allocating dedicated spectrum resources for each cluster eliminates the benefits of spectrum co-existence, the primary prospect of OTA-FL. AirCluster enables multiple models to be trained simultaneously in a shared spectrum, while ensuring model convergence for all clusters. To do so, AirCluster leverages a MIMO system to *align* the transmitted waveforms for the local gradients belonging to the users in the same cluster, while ensuring that the *aggregate* of the local gradients for each cluster can be decoded by the server. To ensure reliable training with limited number of antennas, we propose a compressed clustered FL framework by leveraging gradient sketching [24], [25], [26], where we adapt the dimensionality of the local updates to the resource limitations, while providing formal convergence guarantees for the models of all clusters simultaneously.

Manuscript received 26 December 2022; revised 20 June 2023 and 17 October 2023; accepted 17 December 2023. Date of publication 29 December 2023; date of current version 12 July 2024. This work was supported in part by OUSD (R&E)/RT&L under Grant W911NF-20-2-0267, in part by the NSF CAREER Award CCF-2144927, and in part by the UCR Opportunity to Advance Sustainability Innovation and Social Inclusion (OASIS) Funding Award. The views and conclusions contained in this document are those of the authors. An earlier version of this paper was presented at the 2022 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'22) [DOI: 10.1109/ICASSP43922.2022.9746750]. The associate editor coordinating the review of this article and approving it for publication was P. Hu. *(Corresponding author: Başak Güler.)*

The authors are with the Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA 92521 USA (e-mail: hsami003@ucr.edu; bguler@ece.ucr.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2023.3345730.

Digital Object Identifier 10.1109/TWC.2023.3345730

In our theoretical analysis, we present a novel model convergence analysis for clustered OTA-FL, by incorporating the joint impact of clustering, channel noise, and compression, and identify the key trade-offs between the convergence rate, number of antennas, channel quality and data heterogeneity. We perform extensive experiments to evaluate the performance of AirCluster, for various image classification tasks on the MNIST and CIFAR-10 datasets [27], [28] under highly heterogeneous data distributions across the users. We further demonstrate the impact of channel conditions, as well as the number of antennas and compression ratio on the test accuracy of the trained models. Our experiments demonstrate that AirCluster can significantly improve the performance of OTA-FL in heterogeneous settings. Our contributions are summarized as follows:

1) We propose a personalized OTA-FL framework, AirCluster, to jointly address data heterogeneity and bandwidth limitations in FL. AirCluster trains multiple models concurrently, where models are adapted to the data heterogeneity across the users, while allowing all users to share the same spectrum resources.
2) We develop a coordinated MIMO beamforming and gradient compression mechanism to enable spectrum coexistence for clustered FL under resource limitations.
3) We provide the theoretical convergence guarantees of AirCluster, and identify the key trade-offs between the convergence rate, compression ratio, channel noise, and the number of antennas.
4) Through extensive experiments, we demonstrate significant increase in the *test accuracy* against state-of-the-art FL benchmarks, even when the latter is evaluated under *ideal channel conditions and uncompressed gradients*.

## II. Related Work

*Personalized FL:* For *user-level* personalization, a multi-task learning approach is proposed in [12], whereas [13], [14], and [15] propose a meta-learning approach. To enhance model accuracy under user heterogeneity [8] uses a proximal term to minimize divergence among local updates. For *group-level* personalization, [23] introduces hierarchical clustering, whereas [18] leverages cosine similarity between the local updates. References [7] and [19] propose clustered FL with formal convergence guarantees, where multiple adaptive models are trained simultaneously, and models are adapted to groups of users with similar data distributions. Reference [20] studies mixture of source distributions, whereas [21] addresses fairness. Dynamic clustering is studied in [22] and [29], whereas [30] studies the number of clusters in heterogeneous settings.

*Over-the-air FL (OTA-FL):* OTA computing performs transmission and computation simultaneously by leveraging the waveform superposition property of the multi access channel [31], [32], [33], [34], [35]. Recently, OTA-FL has been used to aggregate the local gradients in FL [4], [5], [6], [36], [37]. Reference [38] studies digital OTA-FL, [39] explores time-varying precoding, whereas [40] studies power allocation, and [41] and [42] consider user scheduling. Reference [43] explores user privacy, whereas [44] and [45] leverage intermediate parties to mitigate adverse channel effects. In contrast, our focus is on OTA-FL with personalization, where the goal is to mitigate the adverse effects of data heterogeneity across the local datasets.

*Gradient Compression and Sketching:* Broadly, there are three complementary techniques to gradient compression in FL: 1) *Gradient sketching* is rooted in compressed sensing principles, with the intuition that gradient vectors are sparse, to map them to a lower dimensional subspace through a sketching matrix [24], [25], [26]. Sketching enables gradient aggregation without increasing dimensionality, hence is particularly suitable for antenna-limited settings. 2) *Gradient quantization* reduces the number of bits used to represent each gradient parameter [46]. 3) *Gradient sparsification* allows users to send a small fraction of local gradient parameters [47], [48]. Parameter coordinates often differ across the users, increasing the size of the aggregated gradient.

*Organization:* The remainder of the paper is organized as follows. In Section III, we present the system model. Section IV introduces the AirCluster framework. Section V presents the theoretical analysis, and Section VI provides the experimental results. Section VII concludes the paper.

*Notation:* In the following, $x$ is a scalar, $\mathbf{x}$ is a vector, and $\mathbf{X}$ is a matrix. $\mathcal{X}$ is a set with cardinality $|\mathcal{X}|$, and $[N] = \{1, \ldots, N\}$. $\mathbf{X}^H$ is the Hermitian transpose, $\text{tr}(\mathbf{X})$ is the trace, and $\|\mathbf{X}\|_F$ is the Frobenius norm of $\mathbf{X}$. We use $x \gtrsim y$ when $x \geq cy$ for some sufficiently large constant $c > 0$, and $x \lesssim y$ when $x \leq cy$ for some sufficiently small constant $c > 0$.

## III. Problem Formulation

We consider a clustered FL task in a network of $N$ users, where user $i \in [N]$ has a local dataset $\mathcal{D}_i$ with $|\mathcal{D}_i| := D_i$ data points. The local dataset of each user is realized from a class of $K$ distributions $\mathcal{P}_1, \ldots, \mathcal{P}_K$. The set of users $i \in [N]$ for which $\mathcal{D}_i \sim \mathcal{P}_k$ is denoted by the set $\mathcal{C}_k^* = \{i \in [N] : \mathcal{D}_i \sim \mathcal{P}_k\} \subseteq [N]$. The goal is to train $K$ models, where model $\mathbf{w}_k \in \mathbb{R}^d$ is designed for cluster $k \in [K]$ to minimize the loss,

$$F_k(\mathbf{w}_k) := \mathbb{E}_{\xi \sim \mathcal{P}_k}[f(\mathbf{w}_k, \xi)], \tag{1}$$

where $f(\mathbf{w}_k, \xi)$ is the stochastic loss function computed on data sample $\xi \in \mathcal{D}_i$, $d$ denotes the dimension of $\mathbf{w}_k$, and

$$\mathbf{w}_k^* = \arg\min_{\mathbf{w}_k \in \mathbb{R}^d} F_k(\mathbf{w}_k), \tag{2}$$

denotes the minimizer of (1), hence the optimal model for cluster $k \in [K]$. The dataset distributions and cluster identities are unknown to the users and server apriori, hence any solution to (1) should identify both the set of users assigned to each cluster, and the associated model parameters jointly. Training is done through an iterative process. At each iteration, users select the cluster that minimizes the loss on their local dataset, and train the model for the selected cluster. The state of model $\mathbf{w}_k$ at training round $t$ is denoted by $\mathbf{w}_k(t)$, which we refer to as a *global model* for cluster $k \in [K]$. At each training round, the server broadcasts the current state of the $K$ global models

$\{\mathbf{w}_k(t)\}_{k\in[K]}$ to the users. Then, user $i$ computes a local loss,

$$F_i(\mathbf{w}_k(t), \widehat{\mathcal{Z}}_i(t)) := \frac{1}{|\widehat{\mathcal{Z}}_i(t)|} \sum_{\xi\in\widehat{\mathcal{Z}}_i(t)} F_i(\mathbf{w}_k(t), \xi) \quad \forall k \in [K], \tag{3}$$

where $\widehat{\mathcal{Z}}_i(t) \subseteq \mathcal{D}_i$ denotes the set of data samples used for cluster estimation at round $t$, and $F_i(\mathbf{w}_k(t), \xi)$ is the local loss computed on the local data sample $\xi \in \widehat{\mathcal{Z}}_i(t)$. Then, user $i$ selects the cluster that minimizes the local loss,

$$c_{it} := \arg\min_{k\in[K]} F_i(\mathbf{w}_k(t), \widehat{\mathcal{Z}}_i(t)), \tag{4}$$

and updates the global model for the selected cluster, by creating a *local model* $\overline{\mathbf{w}}_i(t) \leftarrow \mathbf{w}_{c_{it}}(t)$, and updating it through $E$ local gradient descent steps,

$$\overline{\mathbf{w}}_i(t) \leftarrow \overline{\mathbf{w}}_i(t) - \eta \nabla F_i(\overline{\mathbf{w}}_i(t), \mathcal{Z}_i(t)) \tag{5}$$

where $\mathcal{Z}_i(t) \subseteq \mathcal{D}_i$ denotes a minibatch of data samples used for training at round $t$,

$$\nabla F_i(\overline{\mathbf{w}}_i(t), \mathcal{Z}_i(t)) := \frac{1}{|\mathcal{Z}_i(t)|} \sum_{\xi\in\mathcal{Z}_i(t)} \nabla F_i(\overline{\mathbf{w}}_i(t), \xi) \tag{6}$$

represents the average of the stochastic gradients evaluated on the data samples $\xi \in \mathcal{Z}_i(t)$, and $\eta$ is the learning rate. After $E$ local training rounds, user $i$ sends the model difference,

$$\overline{\mathbf{g}}_i(t) := \mathbf{w}_{c_{it}}(t) - \overline{\mathbf{w}}_i(t), \tag{7}$$

to the server. Note that $\overline{\mathbf{g}}_i(t)$ denotes the accumulated gradient over $E$ local training iterations (scaled by learning rate), hence will be referred to as the *local gradient* of user $i$ in the sequel. After receiving the local updates from (7), the server updates the global model for each cluster,

$$\mathbf{w}_k(t+1) = \mathbf{w}_k(t) - \frac{1}{|\mathcal{C}_k(t)|} \sum_{i\in\mathcal{C}_k(t)} \overline{\mathbf{g}}_i(t) \tag{8}$$

$$= \mathbf{w}_k(t) - \mathbf{g}_k(t) \quad \forall k \in [K], \tag{9}$$

where the set of users assigned to cluster $k$ according to (4) at training round $t$ (which may be different than the ground truth $\mathcal{C}_k^*$) is denoted by,

$$\mathcal{C}_k(t) := \{i \in [N] : c_{it} = k\} \tag{10}$$

and the empirical average of the gradients from users in cluster $k$ is given by,

$$\mathbf{g}_k(t) := \frac{1}{|\mathcal{C}_k(t)|} \sum_{i\in\mathcal{C}_k(t)} \overline{\mathbf{g}}_i(t). \tag{11}$$

The main intuition behind the clustering mechanism is that the optimal model for each distribution should minimize the local loss for the users sampled from that distribution [19]. The clustering mechanism first identifies the group of users for which a given model performs the best, and then updates the model using the local datasets of those users.

*Main Problem:* Our goal is to develop an over-the-air clustered FL framework to enable spectrum co-existence for group-level personalization in FL. We ask the question,

- How can we develop a communication-efficient over-the-air clustered FL framework, where all cluster models from (9) are trained concurrently in the wireless medium?

The key challenge is that (9) requires the server to recover the sum of the local updates *for each cluster*. On the other hand, when users send their updates concurrently over the wireless channel, the server only observes the sum of the local updates from all users, and can not distinguish the updates belonging to different clusters. In contrast, allocating dedicated spectrum resources for each cluster eliminates spectrum co-existence, the primary benefit of over-the-air FL.

To address this challenge, in this work we introduce AirCluster, an over-the-air clustered FL framework, that enables spectrum sharing across different groups (clusters) of users with heterogeneous data distributions, where the local updates from (11) are aggregated concurrently over-the-air, while ensuring that the server can recover the aggregate of the local models belonging to each cluster. To do so, we utilize spatial dimensions enabled by a MIMO beamforming architecture, and propose a coordinated precoder design that aligns the signals designated for each cluster over-the-air, while directing signals designated for different clusters in orthogonal subspaces. To enable robust training with a limited number of antennas, we leverage an unbiased gradient compression methodology with sketching, to transform the local gradients to a lower dimensional subspace prior to transmission. We next describe the details of AirCluster.

## IV. AirCluster: Over-the-Air Clustered Federated Learning

We first describe the details of the underlying MIMO transmission architecture.

*Network Model:* We consider a wireless access point (AP) integrated with the server, equipped with $N_R$ receive antennas. User $i \in [N]$ has $N_T$ transmit antennas. We consider a block Rayleigh fading channel model where the channel coefficients stay constant within a given training round, but may vary from one round to another. The channel coefficients from user $i$ to the AP are represented with an $N_R \times N_T$ matrix $\mathbf{H}_i(t)$ at round $t \in [T]$, where each element is distributed i.i.d. from a complex Gaussian distribution $\mathcal{CN}(0, \sigma^2)$.

*Over-the-Air Gradient Aggregation:* We consider a clustered FL task when the local gradients from (11) for all clusters are aggregated over-the-air. We utilize a linear MIMO precoding architecture, where user $i \in [N]$ is equipped with an $N_T \times d$ dimensional precoding matrix $\mathbf{V}_i(t)$ at round $t$, using which the user encodes its local gradient $\overline{\mathbf{g}}_i(t)$, and sends the encoded gradient $\mathbf{V}_i(t)\overline{\mathbf{g}}_i(t)$ to the AP. The maximum average transmit power constraint of user $i$ is given by $P_{T,i}$. The received signal at the AP is denoted by an $N_R \times 1$ vector,

$$\mathbf{y}(t) = \sum_{i\in[N]} \mathbf{H}_i(t)\mathbf{V}_i(t)\overline{\mathbf{g}}_i(t) + \mathbf{n}(t)$$

$$= \sum_{k\in[K]} \sum_{i\in\mathcal{C}_k(t)} \mathbf{H}_i(t)\mathbf{V}_i(t)\overline{\mathbf{g}}_i(t) + \mathbf{n}(t) \tag{12}$$

at round $t$, where $\mathbf{n}(t)$ represents the noise vector consisting of independent zero mean Gaussian random variables with

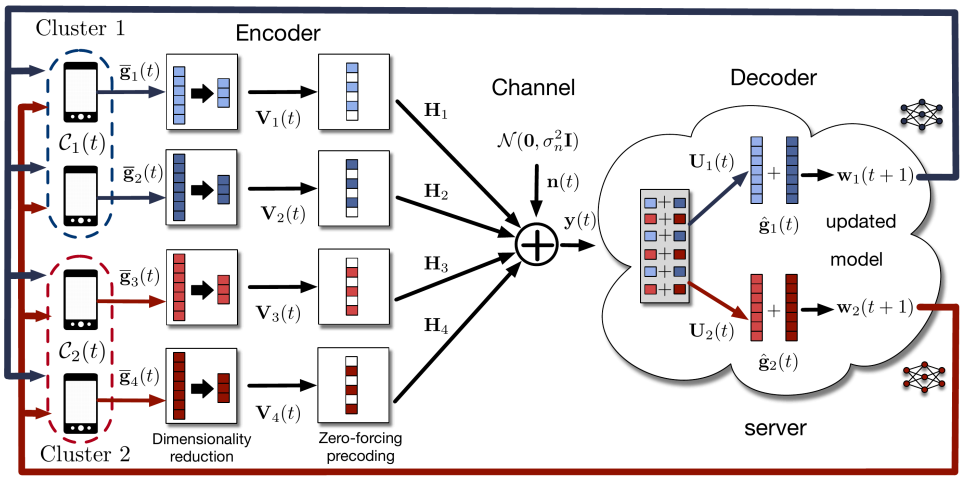



Fig. 1. **Over-the-air clustered FL (AirCluster).** A motivating example with $N = 4$ users and $K = 2$ clusters. Users receive the global models $\{\mathbf{w}_k(t)\}_{k\in[2]}$ for each cluster from the server, and compute the local loss on each model. Then, each user $i \in [N]$ selects the cluster with the minimum loss, computes a local gradient $\overline{\mathbf{g}}_i(t)$, and sends the encoded gradient to the server. Server decodes the aggregated gradients and updates the global model for each cluster.

$\mathbb{E}[\mathbf{n}(t)\mathbf{n}(t)^H] = \sigma_n^2\mathbf{I}$. Upon receiving (12), the AP decodes the sum of the local gradients from (11) for each cluster $k \in [K]$, using an $N_R \times d$ decoding matrix $\mathbf{U}_k(t)$ for each cluster $k \in [K]$. The decoding for cluster $k$ is given as,

$$\widehat{\mathbf{g}}_k(t) = \frac{1}{|\mathcal{C}_k(t)|}\mathbf{U}_k^H(t)\mathbf{y}(t) \tag{13}$$

where $\widehat{\mathbf{g}}_k(t)$ is the estimate of the aggregated local gradients for cluster $k$. Finally, the AP updates the global models as shown in (9) for the next round,

$$\mathbf{w}_k(t+1) = \mathbf{w}_k(t) - \widehat{\mathbf{g}}_k(t) \text{ for all } k \in [K]. \tag{14}$$

Our goal is to design the precoders and decoders $\{\mathbf{V}_i(t)\}_{i\in[N]}, \{\mathbf{U}_k(t)\}_{k\in[K]}$ to ensure formal convergence guarantees for the global model of each cluster $k \in [K]$, with a limited number of antennas. To this end, we propose AirCluster, a coordinated gradient compression and zero-forcing beamforming mechanism for clustered learning, with provable convergence guarantees for the trained model of each cluster, while ensuring the resource constraints in the number of antennas and transmit power of each user. AirCluster builds on a coordinated zero-forcing precoding approach, where the precoders are designed to *align* the local gradients designated for each cluster over-the-air, while zero-forcing the interference from other clusters. We next provide the individual components of AirCluster, which are also illustrated in Fig. 1.

*Precoder Design:* The precoder consists of two components: 1) Gradient compression, and 2) Zero-forcing beamforming. The former transforms the local gradients to a reduced dimensional space, to reduce the number of antennas required in the later stages. The latter aggregates the compressed gradients for each cluster over-the-air, while enabling zero-forcing for the interference received from other clusters. For gradient compression, we leverage sketching through Gaussian random projections. Each user compresses its local gradient $\overline{\mathbf{g}}_i(t) \in \mathbb{R}^d$ by projecting it to a reduced dimensional space $\mathbf{R}(t)\overline{\mathbf{g}}_i(t) \in \mathbb{R}^b$, using a random Gaussian sketching matrix $\mathbf{R}(t) \in \mathbb{R}^{b\times d}$ for some $b \ll d$ such that $N_R = Kb$, where each element is generated i.i.d from a Gaussian distribution $\mathcal{N}(0, \sigma_R^2)$ with $\sigma_R^2 = \frac{1}{b}$. $\mathbf{R}(t)$ can be determined by the AP offline and sent to the users prior to training. After compressing the local gradient, each user encodes the compressed gradient via zero-forcing. The key intuition is to *align* the local gradients received from each cluster over-the-air, in a way that enables the AP to recover the aggregate of the local gradients for each cluster, when the received signal consists of the signals received from all clusters. To do so, we define an $N_T \times b$ zero-forcing matrix $\overline{\mathbf{V}}_i(t)$ for user $i \in \mathcal{C}_k(t)$ as,

$$\overline{\mathbf{V}}_i(t) = \sqrt{P_k(t)}\mathbf{H}_i^\dagger(t)\mathbf{A}_k \tag{15}$$

where $\mathbf{H}_i^\dagger(t) = \mathbf{H}_i(t)^H(\mathbf{H}_i(t)\mathbf{H}_i(t)^H)^{-1}$, and,

$$\mathbf{A}_k = \left[\mathbf{0}_{b\times b} \cdots \mathbf{0}_{b\times b}\ \mathbf{I}_{b\times b}\ \mathbf{0}_{b\times b} \cdots \mathbf{0}_{b\times b}\right]^T \tag{16}$$

is a concatenation of $K$ submatrices of size $b\times b$, where the $k^{th}$ submatrix is an identity matrix $\mathbf{I}_{b\times b}$, and all other submatrices are equal to the zero matrix $\mathbf{0}_{b\times b}$. Matrix $\mathbf{A}_k$ has full column rank, and $\mathbf{A}_k^H\mathbf{A}_k = \mathbf{I}_{b\times b}$. Finally, $P_k(t)$ is the transmit power scaling factor of users $i \in \mathcal{C}_k(t)$ at training round $t$, given as,

$$P_k(t) = \min_{i\in\mathcal{C}_k(t)} \frac{P_{T,i}}{\|\mathbf{H}_i^\dagger(t)\mathbf{A}_k\|_F^2\|\overline{\mathbf{g}}_i(t)\|^2}. \tag{17}$$

Equation (15) has two key features. First, it *aligns* the local gradients received from all users assigned to cluster $k$, since (15) guarantees that,

$$\mathbf{A}_k := \frac{1}{\sqrt{P_k(t)}}\mathbf{H}_i(t)\overline{\mathbf{V}}_i(t) \quad \text{for all} \quad i \in \mathcal{C}_k(t). \tag{18}$$

Second, it cancels the inter-cluster interference received from other clusters, as all users send their local gradients concurrently. The final precoder $\mathbf{V}_i(t)$ of user $i \in [N]$ is defined as,

$$\mathbf{V}_i(t) := \overline{\mathbf{V}}_i(t)\mathbf{R}(t) = \sqrt{P_k(t)}\mathbf{H}_i^\dagger(t)\mathbf{A}_k\mathbf{R}(t) \tag{19}$$

hence the average transmit power of each user $i \in \mathcal{C}_k(t)$ satisfies $\mathbb{E}_{\mathbf{R}}[\|\mathbf{V}_i(t)\overline{\mathbf{g}}_i(t)\|^2] \leq P_{T,i}$. The received signal at the AP can be written from (12) as,

$$\mathbf{y}(t) = \sum_{k\in[K]}\sum_{i\in\mathcal{C}_k(t)} \mathbf{H}_i(t)\mathbf{V}_i(t)\overline{\mathbf{g}}_i(t) + \mathbf{n}(t) \tag{20}$$

$$= \sum_{k\in[K]} \sqrt{P_k(t)}\mathbf{A}_k\mathbf{R}(t)\Big(\sum_{i\in\mathcal{C}_k(t)} \overline{\mathbf{g}}_i(t)\Big) + \mathbf{n}(t) \quad (21)$$

*Decoder Design:* After receiving the aggregated signal from (21), the AP decodes the aggregate of the local gradients for each cluster $k \in [K]$. The decoding process consists of two components. The first component is interference cancellation, to remove the interference received from other clusters. The second component is decompression of the local gradients, where the compressed gradients are projected back to $\mathbb{R}^d$. The decoder for cluster $k \in [K]$ is then defined as,

$$\mathbf{U}_k(t) := \overline{\mathbf{U}}_k(t)\mathbf{R}(t) \quad (22)$$

where $\overline{\mathbf{U}}_k(t)$ is an interference suppression matrix given by,

$$\overline{\mathbf{U}}_k(t) = \frac{1}{\sqrt{P_k(t)}}\mathbf{U}_k^0\left(\mathbf{A}_k^H\mathbf{U}_k^0\right)^{-1} \quad (23)$$

where $\mathbf{U}_k^0$ is an $N_R \times b$ matrix whose columns correspond to a null-space basis of,

$$\mathbf{A}_{\bar{k}} \triangleq \left[\mathbf{A}_1 \cdots \mathbf{A}_{k-1}\ \mathbf{A}_{k+1} \cdots \mathbf{A}_K\right], \quad (24)$$

which is a cascaded matrix of $\mathbf{A}_j$ for $j \in [K]\backslash\{k\}$. Matrix $\mathbf{U}_k^0$ can be obtained from the Singular Value Decomposition (SVD) of $\mathbf{A}_{\bar{k}}$ given by $[\mathbf{U}_k^1\ \mathbf{U}_k^0]\Sigma_k\mathbf{B}_k^H$. Note that the column vectors of $\mathbf{A}_k$ from (16) defines a left null-space of $\mathbf{A}_{\bar{k}}$ from (24), as $\mathbf{A}_k^H\mathbf{A}_{\bar{k}} = \mathbf{0}$. Therefore, without loss of generality, we can let the null subspace of matrix $\mathbf{A}_{\bar{k}}$ to be $\mathbf{U}_k^0 = \mathbf{A}_k$, where $\mathbf{A}_k$ is as defined in (16), from which we have,

$$\overline{\mathbf{U}}_k(t) = \frac{1}{\sqrt{P_k(t)}}\mathbf{U}_k^0\left(\mathbf{A}_k^H\mathbf{U}_k^0\right)^{-1} = \frac{1}{\sqrt{P_k(t)}}\mathbf{A}_k\left(\mathbf{A}_k^H\mathbf{A}_k\right)^{-1} = \frac{1}{\sqrt{P_k(t)}}\mathbf{A}_k \quad (25)$$

since $\mathbf{A}_k^H\mathbf{A}_k = \mathbf{I}_{b\times b}$. The interference supression matrix $\overline{\mathbf{U}}_k(t)$ has two key features:

$$\sqrt{P_k(t)}\overline{\mathbf{U}}_k^H(t)\mathbf{A}_k = \mathbf{I}_{b\times b}, \quad (26)$$

which guarantees intra-cluster model aggregation, i.e., correct recovery of the aggregate of local gradients for cluster $k$, and,

$$\overline{\mathbf{U}}_k^H(t)\mathbf{A}_1 = \cdots = \overline{\mathbf{U}}_k^H(t)\mathbf{A}_{k-1} = \overline{\mathbf{U}}_k^H(t)\mathbf{A}_{k+1} = \cdots = \overline{\mathbf{U}}_k^H(t)\mathbf{A}_K = \mathbf{0} \quad (27)$$

which guarantees inter-cluster interference cancellation from clusters $[K]\backslash\{k\}$. The zero-forcing constraint (27) implies that $\mathbf{U}_k^H(t)$ is in the null space of $\mathbf{A}_{\bar{k}}$ [49]. Finally, multiplication with $\mathbf{R}^H(t)$ in (22) decompresses the gradients by projecting the decoded signal back to $\mathbb{R}^d$. Using (22), the AP decodes the aggregated local gradients from (13) for each cluster $k \in [K]$,

$$\widehat{\mathbf{g}}_k(t) = \frac{1}{|\mathcal{C}_k(t)|}\mathbf{U}_k^H(t)\mathbf{y}(t) \quad (28)$$

$$= \mathbf{R}^H(t)\mathbf{R}(t)\Big(\frac{1}{|\mathcal{C}_k(t)|}\sum_{i\in\mathcal{C}_k(t)} \overline{\mathbf{g}}_i(t)\Big) + \frac{1}{\sqrt{P_k(t)}|\mathcal{C}_k(t)|}\mathbf{R}^H(t)\mathbf{n}_k(t) \quad (29)$$

**Algorithm 1** AirCluster: Over-the-Air Clustered FL

**for** each cluster $k \in [K]$ **do**
  Initialize $\mathbf{w}_k(0)$ ▷ $\mathbf{w}_k(t)$ is the global model of cluster $k$ at round $t$
**for** each round $t = 0, 1, \ldots, T-1$ **do**
  **for** each client $i \in [N]$ in parallel **do**
    **for** $k \in [K]$ **do**
      Compute $F_i(\mathbf{w}_k(t), \widehat{\mathcal{Z}}_i(t))$ ▷ Equation (3)
      Find cluster estimate $c_{it} = \operatorname{argmin}_{k\in[K]} F_i\big(\mathbf{w}_k(t), \widehat{\mathcal{Z}}_i(t)\big)$ ▷ Equation (4)
    $\overline{\mathbf{g}}_i(t) \leftarrow$ CLIENTUPDATE$(i, \mathcal{Z}_i(t), \mathbf{w}_{c_{it}}(t))$
    Send the encoded gradient $\mathbf{V}_i(t)\overline{\mathbf{g}}_i(t)$ to server ▷ Equation (19)
  **for** $k \in K$ server **do**
    Decode cluster aggregate $\rightarrow \widehat{\mathbf{g}}_k(t) = \frac{1}{|\mathcal{C}_k(t)|}\mathbf{U}_k^H(t)\mathbf{y}(t)$ ▷ Equation (28)
    Update the cluster global model $\rightarrow \mathbf{w}_k(t+1) = \mathbf{w}_k(t) - \widehat{\mathbf{g}}_k(t)$ ▷ Equation (14)
    Sends the cluster model $\mathbf{w}_k(t+1)$ to the users
**function** CLIENTUPDATE$(u, \mathcal{Z}, \mathbf{w})$
  $\mathbf{g} = \mathbf{0}$ ▷ Initializing the gradient
  **for** $i = 1, \ldots, E$ **do** ▷ $E$ is the number of local iterations
    $\mathbf{g} \leftarrow \mathbf{g} + \eta\nabla F_u(\mathbf{w}; \mathcal{Z})$ ▷ Accumulating gradient
    $\mathbf{w} \leftarrow \mathbf{w} - \eta\nabla F_u(\mathbf{w}; \mathcal{Z})$ ▷ $\eta$ is the learning rate

$$= \mathbf{R}^H(t)\mathbf{R}(t)\mathbf{g}_k(t) + \frac{1}{\sqrt{P_k(t)}|\mathcal{C}_k(t)|}\mathbf{R}^H(t)\mathbf{n}_k(t) \quad (30)$$

where $\mathbf{n}_k$ for all $k \in [K]$ is a $b \times 1$ dimensional vector,

$$\mathbf{n}(t) = [\mathbf{n}_1^H(t) \cdots \mathbf{n}_{k-1}^H(t)\ \mathbf{n}_k^H(t)\ \mathbf{n}_{k+1}^H(t) \cdots \mathbf{n}_K^H(t)]^H$$

Finally, using (29), the AP updates the global model for each cluster $k \in [K]$ as shown in (14). The individual steps of AirCluster is provided in Algorithm 1. AirCluster can further be extended to transmission of the compressed gradient over multiple time slots when $b > \frac{N_R}{K}$, as detailed in Appendix A-A. In Appendix A-B, we also provide a generalized precoder/decoder design when $b < \frac{N_R}{K}$.

*Coordination:* To compute $P_k(t)$, users can locally compute $\frac{P_{T,i}}{\|\mathbf{H}_i^\dagger(t)\mathbf{A}_k\|_F^2\|\overline{\mathbf{g}}_i(t)\|^2}$ (a scalar), and send it to the AP. The AP then evaluates $P_k(t)$ from (17), and sends it back to the users $i \in \mathcal{C}_k(t)$. Then, each user can compute the precoder from (15) locally, without any additional communication overhead. Similarly, the AP can compute the decoders from (23) locally. In the following, we present the theoretical performance guarantees and key trade-offs for AirCluster.

## V. Performance Analysis

The performance of AirCluster is governed by the relation between the number of receive and transmit antennas, gradient size and the compression ratio (size of the compressed gradients), and the number of clusters. We first demonstrate the relation between the minimum number of antennas required with respect to the model size and the number of clusters.

*Theorem 1: In a network of $N$ users, where each user has $N_T$ transmit antennas, along with an AP with $N_R$ receive antennas, over-the-air clustered FL with AirCluster requires $N_T \geq N_R \geq Kb$, where $K$ is the number of clusters, and $b$ is the size of the compressed gradients.*

*Proof:* From (27), the decoder for each cluster $k \in [K]$ should zero-force $\mathbf{A}_j$ for all $j \neq k$, while diagonalizing the aggregate of the compressed gradients, $\mathbf{R}(t)\mathbf{g}_k(t)$, for the desired cluster $k$, where $\mathbf{g}_k(t)$ is the gradient aggregate for cluster $k$ as defined in (11). Therefore, to decode the global model of cluster $k$, the rank of the null space of $\mathbf{A}_{\bar{k}}$, which is $N_R - (K-1)b$, should be at least $b$, which is equal to the dimension of the compressed aggregated gradient $\mathbf{R}(t)\mathbf{g}_k(t)$ for cluster $k$. Accordingly, the number of receive antennas should satisfy $N_R \geq Kb$. Combined with (18), where the system of linear equations has a solution if and only if $N_T \geq N_R$, a necessary condition for the minimum number of transmit and receive antennas is $N_T \geq N_R \geq Kb$. □

*Convergence Analysis:* We next present the convergence guarantees of AirCluster. Let $T$ be the number of total training iterations, and $|\mathcal{D}_i| = D \; \forall i \in [N]$. For the theoretical analysis, we consider a random partitioning of the local dataset of each user into $2T$ disjoint segments, where each segment contains $D' = \frac{D}{2T}$ data points. For user $i \in [N]$, the segments are denoted by $\widehat{\mathcal{Z}}_i(0), \ldots, \widehat{\mathcal{Z}}_i(T-1)$ and $\mathcal{Z}_i(0), \ldots, \mathcal{Z}_i(T-1)$, where $\widehat{\mathcal{Z}}_i(t)$ is used for cluster estimation and $\mathcal{Z}_i(t)$ is used to compute the local gradient at iteration $t$. Doing so allows cluster estimation and gradient computation to be performed on independent sets of data points. For tractability of theoretical analysis, in this section we let $E = 1$, along the line of [19]. We next state a few technical assumptions [19], [20], [21], [47], [50].

*Assumption 1 Smoothness and Convexity: The global loss function $F_k$ for cluster $k \in [K]$ in (1) is $\lambda$-strongly convex, i.e., for all $\mathbf{w}, \mathbf{w}'$, $F_k(\mathbf{w}') \geq F_k(\mathbf{w}) + \langle \nabla F_k(\mathbf{w}), \mathbf{w}' - \mathbf{w} \rangle + \frac{\lambda}{2} \|\mathbf{w}' - \mathbf{w}\|^2$, and $L$-smooth, $F_k(\mathbf{w}') \leq F_k(\mathbf{w}) + \langle \nabla F_k(\mathbf{w}), \mathbf{w}' - \mathbf{w} \rangle + \frac{L}{2} \|\mathbf{w}' - \mathbf{w}\|^2$.*

Note that convexity and smoothness is not imposed on the local loss function of any user.

*Assumption 2 Bounded Loss Variance: For any $\mathbf{w}$ and $k \in [K]$, the variance of the stochastic loss $f(\mathbf{w}; \xi)$ is bounded, i.e., $\mathbb{E}_{\xi \sim \mathcal{P}_k} \left[ \left( f(\mathbf{w}; \xi) - F_k(\mathbf{w}) \right)^2 \right] \leq \mu^2$ for some $\mu > 0$.*

*Assumption 3 Bounded Gradient Variance: For any $\mathbf{w}$ and $k \in [K]$, variance of stochastic gradient $\nabla f(\mathbf{w}; \xi)$ is bounded, i.e., $\mathbb{E}_{\xi \sim \mathcal{P}_k} \left[ \|\nabla f(\mathbf{w}; \xi) - \nabla F_k(\mathbf{w})\|^2 \right] \leq v^2$ for some $v > 0$.*

The next assumption defines a good initialization $\mathbf{w}_k(0)$ of the global models $k \in [K]$, and that the iterates stay within a bounded region around $\mathbf{w}_k^*$. To this end, let,

$$\Delta := \min_{k \neq k'} \|\mathbf{w}_k^* - \mathbf{w}_{k'}^*\|, \tag{31}$$

where $\mathbf{w}_k^*$ is the optimal model for $k \in [K]$ from (2), and

$$p := \min_{k \in [K]} p_k \text{ where } p_k := \frac{|\mathcal{C}_k^*|}{N}. \tag{32}$$

*Assumption 4: Let $\mathbf{w}_k^*$ be the optimal model for cluster $k \in [K]$ from (2). Then, $\max_{k \in [K]} \|\mathbf{w}_k^*\| \lesssim 1$, $\|\mathbf{w}_k(t) - \mathbf{w}_k^*\| \leq (\frac{1}{2} - \alpha)\sqrt{\frac{\lambda}{L}}\Delta$, where $0 < \alpha < \frac{1}{2}$, $D' \gtrsim \frac{K\mu^2}{\alpha^2 \lambda^2 \Delta^4}$, $p \gtrsim \frac{\log(ND')}{N}$ and $\mathbb{E}\left[ \left\| \mathbf{H}_i^\dagger(t) \mathbf{A}_k \right\|_F^2 \right] \leq H_{max}$.*

*Theorem 2 Convergence: In a network with $N$ users and $K$ clusters, let $\mathbf{w}_k^*$ be the optimal model for cluster $k \in [K]$ as defined in (2). After $T$ training rounds, with a learning rate,*

$$\eta = \min \left\{ \frac{1}{6L\frac{d}{b}(K+2)}, \frac{1}{L\frac{H_{max}}{P_{min}} d\sigma_n^2} \right\} \tag{33}$$

*where $P_{min} := \min_{i \in [N]} P_{T,i}$, the global model $\mathbf{w}_k(T)$ of cluster $k \in [K]$ satisfies,*

$$\begin{aligned}
&\mathbb{E}\left[ F_k(\mathbf{w}_k(T)) - F_k(\mathbf{w}_k^*) \right] \\
&\leq \left(1 - \frac{\lambda\eta}{2}\right)^T \mathbb{E}\left[ F_k(\mathbf{w}_k(0)) - F_k(\mathbf{w}_k^*) \right] \\
&\quad + \frac{1}{\lambda}(K+1)\Big( \frac{4v^2}{pND'} + \frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4 p^2 N} \frac{v^2}{(D')^2} \\
&\quad + 320L^2 \frac{c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4 D'} \Big) \\
&\quad + \frac{1}{2\lambda}\Big( \frac{4}{pN}\frac{v^2}{D'} + \frac{144c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4 D'} L^2 \\
&\quad + \frac{16}{p^2} c_1 \frac{\mu^2 v^2}{\alpha^2\lambda^2\Delta^4 (D')^2 N} \Big) + \frac{1}{\lambda}\frac{16}{p^2 N^2}\Big( 9L^2 + \frac{v^2}{D'} \Big)
\end{aligned} \tag{34}$$

*with probability at least $1 - 2TKe^{-cpN}$ for some $c, c_1 > 0$.*

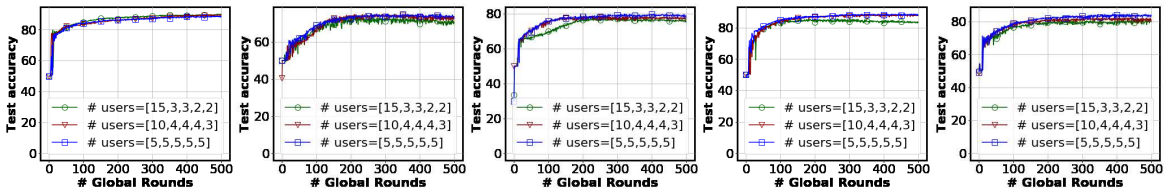
*Proof:* The proof is provided in Appendix B. □

*Remark 1: In (34), the first term vanishes as $T$ increases, as $1 - \frac{\lambda\eta}{2} < 1$ from (33). The remaining terms represent an optimality gap of $O\big(\frac{1}{pND'} + \frac{1}{p^2 D'} + \frac{1}{p^2 N^2}\big)$. Specifically, the second term in (34) is due to clustering (heterogeneity in data distributions), whereas the third term is due to the variance of the compression-decompression mechanism and the fourth term is due to the channel noise. By setting $D' = \Theta(N^2)$, the optimality gap of AirCluster is $O\big(\frac{1}{p^2 N^2}\big)$, which is of the same order as the optimality gap of conventional clustered FL [19]. Since $p \gtrsim \frac{\log(ND')}{N}$ (Assumption 4), choosing $D' = \Theta(N^2)$ ensures that the optimality gap $O\big(\frac{1}{\log^2(ND')}\big) \to 0$ as the number of users $N$ and data samples $D'$ increase.*

*Remark 2: As observed from (33), an increased degree of channel noise ($\sigma_n$), or lower transmit power ($P_{min}$), requires a smaller learning rate $\eta$ to achieve the same optimality gap, which slows down the convergence of the first term in (34). Moreover, since $p \gtrsim \frac{\log(ND')}{N}$, the failure probability $2TKe^{-cpN} \to 0$ for sufficiently large $N$ and $D'$.*

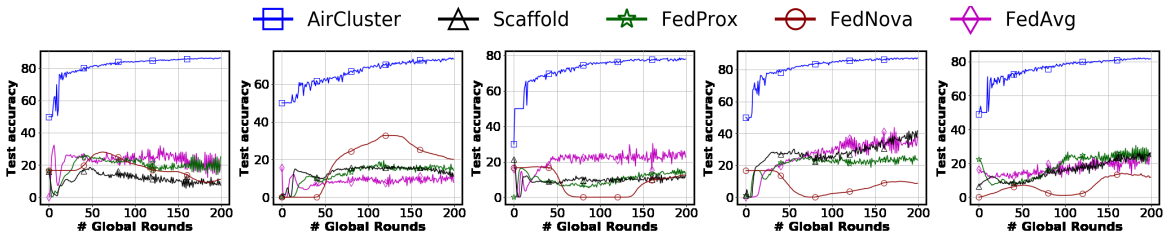
*Remark 3: The number of data samples used for cluster estimation satisfies $D' \gtrsim \frac{K\mu^2}{\alpha^2\lambda^2\Delta^4}$ from Assumption 4. As we demonstrate in Lemma 2 in Appendix B, this ensures that the probability of incorrect clustering is sufficiently small, which is a key technical step for the convergence guarantees of the global model for each cluster.*

*Remark 4: Compression with the random Gaussian sketching matrix $\mathbf{R}(t) \in \mathbb{R}^{b \times d}$ ensures two key properties, unbiasedness and bounded variance, i.e., $\mathbb{E}_{\mathbf{R}}[\mathbf{R}^T(t)\mathbf{R}(t)\overline{\mathbf{g}}(t)] = \overline{\mathbf{g}}(t)$ and $\mathbb{E}_{\mathbf{R}}[\left\| \mathbf{R}^T(t)\mathbf{R}(t)\overline{\mathbf{g}}(t) - \overline{\mathbf{g}}(t) \right\|_2^2] \leq \frac{3d}{b} \|\overline{\mathbf{g}}(t)\|_2^2$ (Lemma 4 in Appendix B), where $\overline{\mathbf{g}}(t) \in \mathbb{R}^d$ is the sketched gradient. This property is utilized in Lemma 6, which is a critical step for formal convergence guarantees in (34). The first term in the*



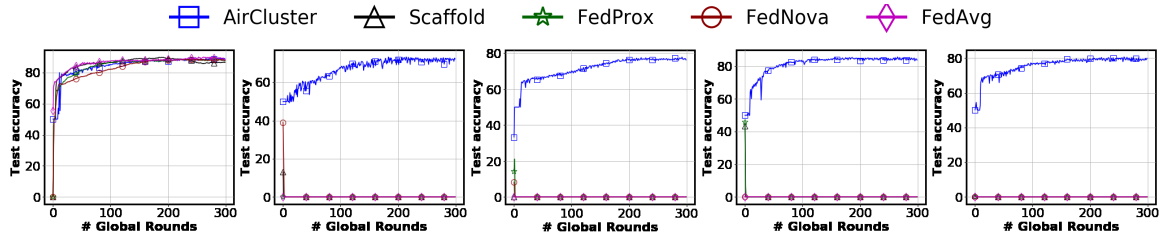

(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

Fig. 2. Impact of the cluster heterogeneity (number of users per cluster) on the model performance (CIFAR-10).



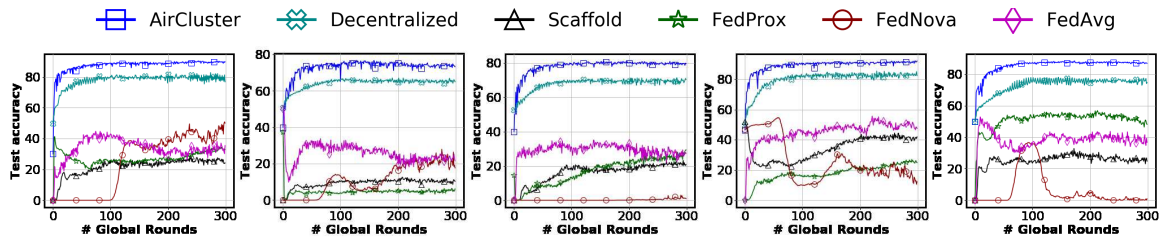

(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

Fig. 3. AirCluster vs FL baselines with $[|C_1^*|, |C_2^*|, |C_3^*|, |C_4^*|, |C_5^*|] = [5, 5, 5, 5, 5]$ users across the clusters (CIFAR-10).



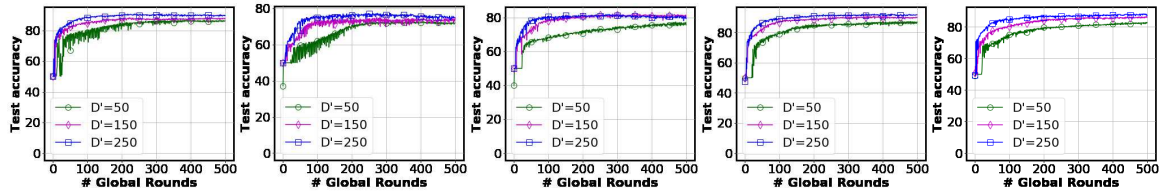

(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

Fig. 4. AirCluster vs FL baselines with $[|C_1^*|, |C_2^*|, |C_3^*|, |C_4^*|, |C_5^*|] = [15, 3, 3, 2, 2]$ users across the clusters (CIFAR-10).




(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

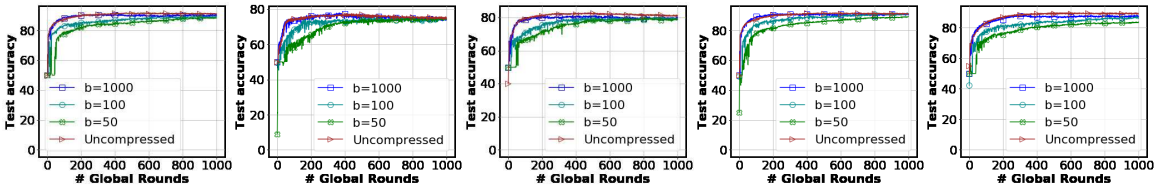
Fig. 5. Performance comparison of AirCluster with non-iid FL baselines (CIFAR-10) with Dirichlet distribution.




(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

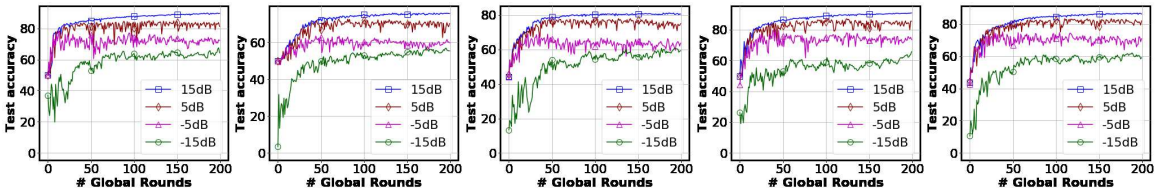
Fig. 6. Impact of varying $D'$, the number of data samples per training round on the model performance (CIFAR-10).

*right hand side of (34) vanishes with $T$. The remaining terms represent an optimality gap that diminishes as the number of users $N$ and data samples $D'$ increase. This also presents a trade-off between compression and convergence rate; a larger compression ratio $\frac{d}{b}$ requires a smaller learning rate $\eta$ as shown in (33) to reach the same target accuracy as conventional clustered FL [19], which can increase the total number of training rounds to reach the target accuracy.*



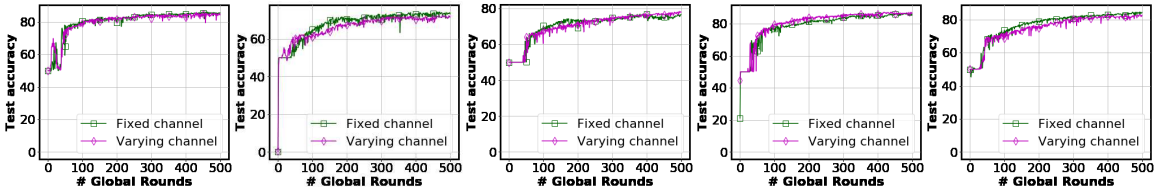

(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

Fig. 7. Impact of varying $b$, the size of the compressed gradient, on the model performance (CIFAR-10).



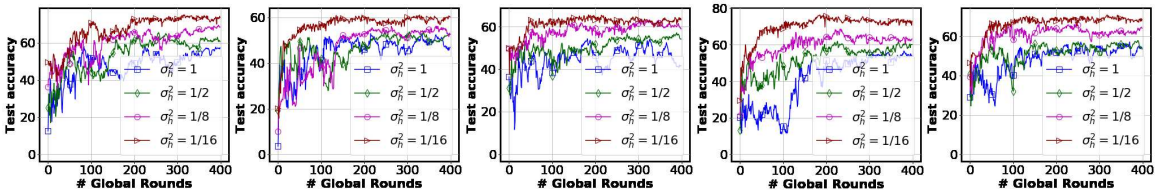

(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

Fig. 8. Impact of varying SNR on the model performance (CIFAR-10).




(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

Fig. 9. Performance analysis of varying channel vs fixed channel across training rounds (CIFAR-10).




(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

Fig. 10. Impact of imperfect CSI on the model accuracy (CIFAR-10).

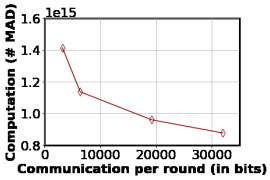



Fig. 11. Total computation cost to reach target accuracy vs per-round communication cost of AirCluster (CIFAR-10).

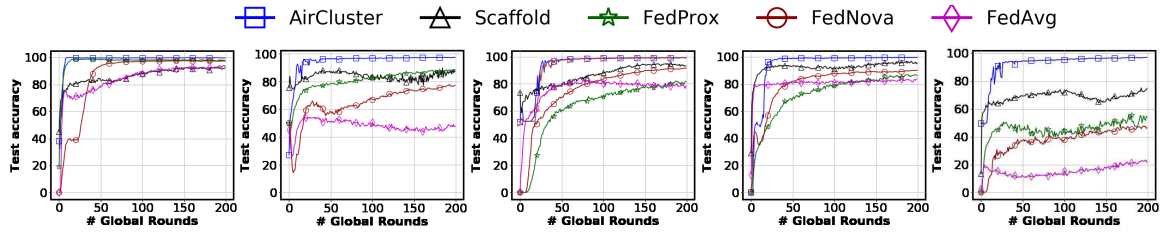
*Theorem 3 Computation Complexity: The computation complexity of the encoding and compression process is $O(N_R^2 N_T + N_T bd)$ per-user per-round, where $N_T$ ($N_R$) is the number of transmit (receive) antennas, and $b$ ($d$) is the size of the compressed (uncompressed) gradient.*

*Proof:* The per-user computation overhead consists of the following components: 1) $O(N_R^2 N_T)$ to compute $\mathbf{H}_i(t)\mathbf{H}_i^H(t)$, 2) $O(N_R^3)$ to compute $(\mathbf{H}_i(t)\mathbf{H}_i^H(t))^{-1}$ using Gauss-Jordan elimination, 3) $O(N_R^2 N_T)$ to compute $\mathbf{H}_i^\dagger(t) = \mathbf{H}_i^H(t)(\mathbf{H}_i(t)\mathbf{H}_i^H(t))^{-1}$, 4) $O(d)$ to compute $\|\overline{\mathbf{g}}_i(t)\|^2$, 5) $O(N_T b)$ to compute $\|\mathbf{H}_i^\dagger(t)\mathbf{A}_k\|_F^2$, 6) $O(N_T N_R b)$ to compute $\overline{\mathbf{V}}_i(t)$ from (15), 7) $O(N_T bd)$ to compute $\overline{\mathbf{V}}_i(t)\mathbf{R}(t)$ in (19), and 8) $O(N_T d)$ to compute $\mathbf{V}_i(t)\overline{\mathbf{g}}_i(t)$. □

## VI. Experiments

**Setup.** We study image classification on CIFAR-10 [28] and MNIST [27] datasets, with $N = 25$ users across $K = 5$ groups (clusters), where the local dataset of users within each group consists of samples from two distinct classes (both datasets have 10 classes). Training is done using the CNN architectures from [2], where the gradient size is $d = 62006$ (CIFAR-10) and $d = 21840$ (MNIST), respectively. The remaining hyper-parameters are $b = 1000$, $E = 5$, $\eta = 0.0001$ with a batch size of 50, $\sigma_n^2 = 1$, and $P_{T,i} = 1000$.

*User Heterogeneity:* We first evaluate the model accuracy of AirCluster with respect to the imbalance in the number of users across the clusters. In Fig. 2, we present the average test accuracy of each cluster for varying levels of




(a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4. (e) Cluster 5.

Fig. 12. Performance comparison of AirCluster with FL baselines in terms of test accuracy (MNIST).

heterogeneity, where $[|C_1^*|, |C_2^*|, |C_3^*|, |C_4^*|, |C_5^*|]$ denotes that the number of users originally belong to cluster $k \in [K]$ is $|C_k^*|$. To ensure a fair comparison, in all scenarios the local dataset size of each user is 600. In Figs. 3 and 4, we evaluate the model accuracy of AirCluster with respect to well-known FL benchmarks, including the conventional `FedAvg` algorithm [2] and also heterogeneity-aware baselines `Scaffold` [51], `FedProx` [8], and `FedNova` [52]. The baselines are evaluated under ideal conditions, without channel noise or compression, whereas AirCluster is subject to both. We observe from Fig. 4 that under severe imbalance, AirCluster significantly outperforms the baselines for the minor clusters (clusters 2-5), where the data imbalance between the clusters causes severe data mismatch between the training and test sets, leading to a catastrophic failure for the minor clusters.

We next consider a non-iid data distribution across the users within each cluster, where the number of samples and the proportion of samples belonging to each class are unbalanced [52]. Specifically, the $N = 25$ users are partitioned into $K = 5$ clusters, and each cluster $k \in [K]$ is assigned to samples from two distinct classes $c \in \{2(k-1), 2(k-1)+1\}$. Then, the data samples from each class is distributed across the users using a Dirichlet distribution $\mathbf{p}_c = Dir_{|C_k^*|}(0.5)$, where the $i^{th}$ element $p_{c,i}$ of $\mathbf{p}_c$ denotes the fraction of samples from class $c$ assigned to user $i \in C_k^*$ in cluster $k \in [K]$. In Fig. 5, we compare the test accuracy with the FL benchmarks, and also decentralized training, where each user performs training using its local dataset only. We observe that AirCluster consistently outperforms all baselines across all clusters.

*Impact of the Size of Data Samples:* In Fig. 6, we demonstrate the impact of the size of the data samples $D'$ used by each user for cluster estimation and gradient computation. We observe that the test accuracy for each cluster increases as $D'$ increases. This observation also aligns with Remark 1, that the optimality gap (in convergence) decreases as $D'$ increases.

*Varying Degree of Compression:* In Fig. 7, we demonstrate the impact of gradient compression on model performance, by varying the size of the compressed gradient as $b \in \{1000, 100, 50\}$ and comparing the test accuracy with respect to training with uncompressed gradients (the ideal case [19] without compression or channel noise, which represents our target accuracy). We observe that compressed gradients achieve comparable accuracy to uncompressed gradients.

*Channel Noise:* We next demonstrate the impact of channel noise on model accuracy. In Fig. 8, we report the average test accuracy of users within each cluster with varying average receive SNR, by varying the maximum average transmit power constraint. The maximum average transmit power constraints used are $1000, 300, 100, 40$, which result in 15dB, 5dB, $-5$dB and $-15$dB receive SNRs respectively. As expected, higher SNR provides increased robustness against the channel noise, hence increasing the model performance for each cluster.

*Fixed Channel vs Varying Channel:* In Fig. 9, we let $b = 50$ and compare the model performance when the channel coefficients $\mathbf{H}_i(t)$ are: 1) fixed across all training rounds $t \in [T]$, and 2) varying across different training rounds. As we observe, both settings achieve comparable test accuracy.

*Impact of Imperfect CSI:* We next study the impact of CSI accuracy on training, by investigating the model performance under noisy (imperfect) CSI. In Fig. 10, we present the impact of imperfect CSI, where the estimated channel is subject to additive complex Gaussian noise $\mathcal{CN}(0, \sigma_h^2)$, by varying the noise power $\sigma_h^2$. We observe that the model convergence is robust against imperfect CSI, and model performance (test accuracy) increases as noise level decreases.

*Computation and Communication Overhead:* We next analyze the impact of gradient compression on the total computation overhead of each user. The communication cost per training round increases as $b$ increases, however, a larger $b$ can also speed up convergence, leading to less computation overall to reach a desired accuracy. In Fig. 11, we demonstrate this trade-off between the per-round communication cost and the total computation cost (with respect to the total number of multiply-add (MAD) operations to reach 75% accuracy).

*MNIST Dataset:* Fig. 12 presents the average test accuracy for the (simpler) MNIST dataset, with respect to FL baselines. We observe that AirCluster again outperforms the baselines.

*Remark 5: When no prior knowledge is available on the user distributions (e.g., demographic information), the number of clusters $K$ can be treated as a hyperparameter, which can be tuned by increasing $K$ until empty clusters start to appear [19]. Then, the number of clusters $K$ can be set to the maximum value for which no empty clusters emerge.*

## VII. Conclusion and Future Directions

This work proposes over-the-air personalized FL for communication efficient distributed learning under heterogeneous settings. We introduce AirCluster, to train concurrent models over-the-air, each one adapted to a group of users with similar data characteristics. We provide the theoretical convergence guarantees of AirCluster under limited resources, and present

extensive numerical experiments to demonstrate its performance with respect to multiple benchmarks.

Future directions include digital over-the-air clustered learning, as well as training under dynamic and time-varying data distributions, and integrating our framework with resource heterogeneities, energy constraints [53], and learning-based reconstruction techniques for compressed sensing [54]. Other directions include extending our analysis to non-convex global objective functions and minimizing the impact of channel noise and imperfect channel estimation [55]. In scenarios where a large number of receive antennas is available, the additional antennas can be further allocated to increase the diversity gain for enhanced network reliability or for handling complementary tasks, such as benign communication or sensing, which are also interesting future directions.

## APPENDIX A
## GENERALIZED AIRCLUSTER FOR FLEXIBLE COMPRESSION

In this section, we present the generalization of AirCluster for flexible compression ratios.

### A. For Compression Parameter $b > \frac{N_R}{K}$

We first discuss the scenario with $b > \frac{N_R}{K}$, to accommodate very large models in practice, by utilizing multiple time slots for transmission. We first define the compressed gradient of user $i \in [N]$ at training round $t$ as,

$$\widetilde{\mathbf{g}}_i(t) \triangleq \mathbf{R}(t)\overline{\mathbf{g}}_i(t) \in \mathbb{R}^{b\times 1} \tag{35}$$

partitioned into $s \triangleq bK/N_R$ equal-sized shards,

$$\widetilde{\mathbf{g}}_i(t) = \left[\widetilde{\mathbf{g}}_i^{\mathrm{T}}(t,1) \cdots \widetilde{\mathbf{g}}_i^{\mathrm{T}}(t,s)\right]^{\mathrm{T}}, \tag{36}$$

where each shard of size $b' \triangleq b/s$ is sent over a single time slot. We next define $\mathbf{R}(t,t') \in \mathbb{R}^{b'\times d}$ to represent the submatrix of $\mathbf{R}(t)$ that contains $b'$ rows of $\mathbf{R}(t)$ such that,

$$\widetilde{\mathbf{g}}_i(t,t') = \mathbf{R}(t,t')\overline{\mathbf{g}}_i(t) \tag{37}$$

hence $\mathbb{E}_{\mathbf{R}}[\mathbf{R}^H(t,t')\mathbf{R}(t,t')] = \frac{b'}{b}\mathbf{I}_{d\times d}$, where $\mathbf{I}_{d\times d}$ is the $d\times d$ identity matrix. Then, the received signal at time slot $t'$ is,

$$\begin{aligned}\mathbf{y}(t,t') &= \sum_{k\in[K]}\sum_{i\in\mathcal{C}_k(t)} \mathbf{H}_i(t,t')\overline{\mathbf{V}}_i(t,t')\widetilde{\mathbf{g}}_i(t,t') + \mathbf{n}(t,t') \\ &= \sum_{k\in[K]} \sqrt{P_k(t,t')}\mathbf{A}_k\left(\sum_{i\in\mathcal{C}_k(t)} \widetilde{\mathbf{g}}_i(t,t')\right) + \mathbf{n}(t,t')\end{aligned}$$

where $\mathbf{A}_k$ is a $Kb'\times b'$ matrix defined as,

$$\mathbf{A}_k = \left[\mathbf{0}_{b'\times b'} \cdots \mathbf{0}_{b'\times b'}\ \mathbf{I}_{b'\times b'}\ \mathbf{0}_{b'\times b'} \cdots \mathbf{0}_{b'\times b'}\right]^T,$$

and $\mathbf{H}_i(t,t')$ is an $N_R \times N_T$ matrix holding the channel coefficients from user $i$ to the server at time slot $t'$ of training round $t \in [T]$, with $N_R = Kb'$, and

$$P_k(t,t') = \min_{i\in\mathcal{C}_k(t)} \frac{P_{T,i}}{\|\mathbf{H}_i^{\dagger}(t,t')\mathbf{A}_k\|_F^2 \frac{b'}{b}\|\overline{\mathbf{g}}_i(t)\|^2}, \tag{38}$$

is the transmit power scaling factor of users $i \in \mathcal{C}_k(t)$ at time slot $t'$ of training round $t$. $\overline{\mathbf{V}}_i(t,t') = \sqrt{P_k(t,t')}\mathbf{H}_i^{\dagger}(t,t')\mathbf{A}_k$ is an $N_T \times b'$ zero forcing matrix for user $i \in \mathcal{C}_k(t)$ such that the average transmit power of each user satisfies $\mathbb{E}_{\mathbf{R}}[\|\overline{\mathbf{V}}_i(t)\widetilde{\mathbf{g}}_i(t)\|^2] \le P_{T,i}$, where $P_{T,i}$ is the maximum average transmit power constraint of user $i$, and $\mathbf{n}(t,t') \in \mathbb{R}^{N_R\times 1}$ denotes the channel noise at time slot $t'$ of training round $t$. Then, the server decodes the compressed gradient aggregate for cluster $k$ at time slot $t'$ as,

$$\begin{aligned}\widehat{\mathbf{g}}_k(t,t') &= \frac{1}{|\mathcal{C}_k(t)|}\overline{\mathbf{U}}_k(t,t')\mathbf{y}(t,t') \\ &= \frac{1}{|\mathcal{C}_k(t)|}\sum_{i\in\mathcal{C}_k(t)} \widetilde{\mathbf{g}}_i(t,t') + \frac{1}{\sqrt{P_k(t,t')}|\mathcal{C}_k(t)|}\mathbf{n}_k(t,t')\end{aligned}$$

where $\mathbf{n}_k(t,t')$ is a $b'\times 1$ dimensional vector such that,

$$\begin{aligned}\mathbf{n}(t,t') = [&\mathbf{n}_1^H(t,t') \cdots \mathbf{n}_{k-1}^H(t,t') \quad \mathbf{n}_k^H(t,t') \\ &\mathbf{n}_{k+1}^H(t,t') \cdots \mathbf{n}_K^H(t,t')]^H\end{aligned} \tag{39}$$

and $\overline{\mathbf{U}}_k(t,t')$ denotes the interference suppression matrix,

$$\overline{\mathbf{U}}_k(t,t') = \frac{1}{\sqrt{P_k(t,t')}}\mathbf{A}_k \tag{40}$$

where

$$\sqrt{P_k(t,t')}\overline{\mathbf{U}}_k^H(t,t')\mathbf{A}_k = \mathbf{I}_{b'\times b'}, \tag{41}$$

which guarantees the correct recovery of the aggregate of the local gradients for cluster $k$, and,

$$\begin{aligned}\overline{\mathbf{U}}_k^H(t,t')\mathbf{A}_1 &= \cdots = \overline{\mathbf{U}}_k^H(t,t')\mathbf{A}_{k-1} \\ &= \overline{\mathbf{U}}_k^H(t,t')\mathbf{A}_{k+1} = \cdots = \overline{\mathbf{U}}_k^H(t,t')\mathbf{A}_K = \mathbf{0}\end{aligned}$$

which guarantees interference cancellation from clusters $[K]\backslash\{k\}$. At the end of $s$ time slots, the server concatenates the decoded signals across all $s$ time slots $t' \in [s]$ to recover the uncompressed gradient aggregate for training round $t$,

$$\begin{aligned}\widehat{\mathbf{g}}_k(t) &= \frac{1}{|\mathcal{C}_k(t)|}\mathbf{R}^H(t)\left[\widehat{\mathbf{g}}_k^{\mathrm{T}}(t,1) \cdots \widehat{\mathbf{g}}_k^{\mathrm{T}}(t,s)\right]^{\mathrm{T}} \\ &= \mathbf{R}^H(t)\mathbf{R}(t)\mathbf{g}_k(t) + \frac{1}{|\mathcal{C}_k(t)|}\mathbf{R}^H(t)\mathbf{P}_k(t)\mathbf{n}_k(t)\end{aligned} \tag{42}$$

where $\mathbf{P}_k(t) \triangleq \left[1/\sqrt{P_k(t,1)} \cdots 1/\sqrt{P_k(t,s)}\right]$, and $\mathbf{n}_k(t) \triangleq \left[\mathbf{n}_k^H(t,1) \cdots \mathbf{n}_k^H(t,s)\right]^H$. Hence, the generalized framework enables the use of larger compression parameters $b$ for large models and antenna-limited settings, by increasing the number of time slots $s \triangleq bK/N_R$ for transmission as $b$ increases (required for more complex training tasks), or as $N_R$ decreases (limited number of antennas).

### B. Compression Parameter $b < \frac{N_R}{K}$

We next discuss the generalized encoder/decoder structure when $b < \frac{N_R}{K}$, for which the compressed gradient can be transmitted over a single time slot at each training round $t \in [T]$. For the precoders, we first define an $N_T \times b$ zero-forcing matrix $\overline{\mathbf{V}}_i(t)$ for each user $i \in \mathcal{C}_k(t)$ in cluster $k \in [K]$,

$$\overline{\mathbf{V}}_i(t) = \sqrt{P_k(t)}\mathbf{H}_i^{\dagger}(t)\mathbf{A}_k \tag{43}$$

where $\mathbf{A}_k$ is a $N_R \times b$ random Gaussian matrix where each element is generated i.i.d. from a standard normal distribution $\mathcal{N}(0,1)$. Parameter $P_k(t)$ is the transmit power scaling factor

as defined in (17). Then, the final precoder $\mathbf{V}_i(t)$ of user $i \in [N]$ is given by (19).

For the decoders, we first define the interference suppression matrix $\overline{\mathbf{U}}_k(t)$ for each cluster $k \in [K]$ as given in (23), which satisfies the conditions (26) (intra-cluster model aggregation) and (27) (inter-cluster interference cancellation). Then, the final decoder is defined as (22). Finally, the aggregate of the gradients for each cluster $k \in [K]$ is decoded as in (30).

## APPENDIX B
## PROOF OF THEOREM 2

In the following, $\mathbb{E}_{\mathbf{R}}[\cdot]$ denotes the expectation over the random compression matrix $\mathbf{R}$, $\mathbb{E}_{\mathbf{n}}$ over the channel noise, $\mathbb{E}_{\mathbf{H}}$ over the channel coefficients, $\mathbb{E}_{\xi}$ over the random sampling during training, and $\mathbb{E}_{\mathcal{C}_k}$ over the randomness in clustering $\mathcal{C}_k(t)$. Our proof is inspired by [19], [20], and [21]. On the other hand, due to the channel noise and the compression-decompression mechanism, the standard convergence analysis for clustered FL does not apply to our problem (our aggregation rule is also different). As such, in the following we provide a novel convergence analysis for over-the-air clustered FL, which bounds the distance between the loss functions for the optimal vs. trained model of each cluster (as opposed to bounding the distance between the optimal vs. trained models directly). By doing so, we can guarantee a vanishing optimality gap under channel noise and compression in the asymptotic analysis. We next review a few useful lemmas.

*Lemma 1 Bounded Gradient Difference, [19]: Let $i \in \mathcal{C}_k^*$, i.e., user $i \in [N]$ belongs to cluster $k \in [K]$, and $\mathcal{Z}_i \in \mathcal{D}_i$ be a minibatch of $D'$ samples from the local dataset $\mathcal{D}_i$ of user $i$. Then,*

$$\mathbb{E}_{\xi}\left[\left\|\nabla F_k(\mathbf{w}) - \nabla F_i(\mathbf{w}, \mathcal{Z}_i)\right\|^2\right] \leq \frac{v^2}{D'} \tag{44}$$

*for any $\mathbf{w} \in \mathbb{R}^d$, where $v$ is as defined in Assumption 3.*

*Lemma 2 Misclustering Probability, [19], Lemma 3: Let $\mathcal{E}_i^{k,k'}(t)$ be the event that user $i$ belongs to cluster $k \in [K]$, and classified to cluster $k' \in [K]$ at round $t$. Then, for any $k' \neq k$, there exists a universal constant $c_1$ such that:*

$$\mathbb{P}\left(\mathcal{E}_i^{k,k'}(t)\right) \leq c_1 \frac{\mu^2}{\alpha^2 \lambda^2 \Delta^4 D'} \tag{45}$$

*Let $\mathcal{E}_i(t) := \mathcal{E}_i^{k,k}(t)$ be the event that user $i$ is assigned to the correct cluster. From union bound,*

$$\mathbb{P}\left(\overline{\mathcal{E}_i}(t)\right) \leq c_1 \frac{K\mu^2}{\alpha^2 \lambda^2 \Delta^4 D'} \tag{46}$$

*where $\overline{\mathcal{E}_i}(t) := \cup_{k' \in [K]\setminus\{k\}} \mathcal{E}_i^{k,k'}(t)$, $\Delta$ is as defined in (31), and $\mu$ is from Assumption 2.*

*Lemma 3 Cluster Cardinality, [19]: Let $p$ be as defined in (32). Then, for any $k \in [K]$ and for any $t \in [T]$, the following holds for some $c > 0$ with probability at least $1 - 2\exp(-cpN)$:*

$$|\mathcal{C}_k(t) \cap \mathcal{C}_k^*| \geq \frac{1}{4} pN \tag{47}$$

*Lemma 4: (Unbiasedness of sketching, [26], Lemma D.11, D.13) Let $\mathbf{R} \in \mathbb{R}^{b \times d}$ denote a random Gaussian matrix, where all entries are sampled i.i.d. from $\mathcal{N}(0, \frac{1}{b})$. Then, for any $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$, the following holds: 1) $\mathbb{E}_{\mathbf{R}}\left[\mathbf{u}^T \mathbf{R}^T \mathbf{R} \mathbf{v}\right] = \mathbf{u}^T \mathbf{v}$, 2) $\mathbb{E}_{\mathbf{R}}\left[\left(\mathbf{u}^T \mathbf{R}^T \mathbf{R} \mathbf{v} - \mathbf{u}^T \mathbf{v}\right)^2\right] \leq \frac{3}{b}\|\mathbf{u}\|_2^2 \cdot \|\mathbf{v}\|_2^2$, 3) $\mathbb{E}_{\mathbf{R}}\left[\mathbf{R}^T \mathbf{R} \mathbf{v}\right] = \mathbf{v}$, 4) $\mathbb{E}_{\mathbf{R}}\left[\left\|\mathbf{R}^T \mathbf{R} \mathbf{v} - \mathbf{v}\right\|_2^2\right] \leq \frac{3d}{b}\|\mathbf{v}\|_2^2$.*

The next relation is utilized in [19]. For completeness, we include an analysis for our setting.

*Lemma 5: For any $k, j \in [K]$ and $k \neq j$, the following holds at iteration $t \in [T]$:*

$$\mathbb{E}_{\mathcal{C}_k}[|\mathcal{C}_k(t) \cap \mathcal{C}_j^*|] \leq \frac{c_1 \mu^2 p_j N}{\alpha^2 \lambda^2 \Delta^4 D'} \tag{48}$$

$$\mathbb{E}_{\mathcal{C}_k}[|\mathcal{C}_k(t) \cap \mathcal{C}_j^*|^2] \leq \frac{c_1 \mu^2 p_j^2 N^2}{\alpha^2 \lambda^2 \Delta^4 D'} \tag{49}$$

*where $p_j$ is defined in (32) and $c_1$ is defined in Lemma 2.*

*Proof:* Let $U_1, \ldots, U_{p_j N}$ be the users that truly belong to cluster $j \in [K]\setminus\{k\}$. Then,

$$\begin{aligned} |\mathcal{C}_k(t) \cap \mathcal{C}_j^*| &= |\mathcal{C}_k(t) \cap (U_1 \cup \cdots \cup U_{p_j N})| \\ &= |\mathcal{C}_k(t) \cap U_1| + \cdots + |\mathcal{C}_k(t) \cap U_{p_j N}| \end{aligned} \tag{50}$$

For all $i \in [p_j N]$, we define a Bernoulli random variable,

$$|\mathcal{C}_k(t) \cap U_i| = \begin{cases} 1 \text{ with probability } \mathbb{P}\left(\mathcal{E}_{U_i}^{j,k}(t)\right) \\ 0 \quad \text{otherwise} \end{cases} \tag{51}$$

where $\mathbb{P}(\mathcal{E}_{U_i}^{j,k}(t))$ is the probability that user $U_i \in \mathcal{C}_j^*$ is misclustered to cluster $k$ (Lemma 2). Then,

$$\begin{aligned} \mathbb{E}_{\mathcal{C}_k}[|\mathcal{C}_k(t) \cap U_i|] &= \mathbb{E}_{\mathcal{C}_k}[|\mathcal{C}_k(t) \cap U_i|^2] \\ &= \mathbb{P}\left(\mathcal{E}_{U_i}^{j,k}(t)\right) \leq \frac{c_1 \mu^2}{\alpha^2 \lambda^2 \Delta^4 D'} \end{aligned} \tag{52}$$

where (52) follows from (45). Hence,

$$\begin{aligned} \mathbb{E}_{\mathcal{C}_k}[|\mathcal{C}_k(t) \cap \mathcal{C}_j^*|] &= \mathbb{E}_{\mathcal{C}_k}\left[\sum_{i=1}^{p_j N} |\mathcal{C}_k(t) \cap U_i|\right] \\ &= \sum_{i=1}^{p_j N} \mathbb{E}_{\mathcal{C}_k}[|\mathcal{C}_k(t) \cap U_i|] \leq p_j N \frac{c_1 \mu^2}{\alpha^2 \lambda^2 \Delta^4 D'} \end{aligned}$$

which follows from (52). Using (52), one can find that,

$$\mathbb{E}_{\mathcal{C}_k}[|\mathcal{C}_k(t) \cap \mathcal{C}_j^*|^2] = \mathbb{E}_{\mathcal{C}_k}\left[\left(\sum_{i=1}^{p_j N} |\mathcal{C}_k(t) \cap U_i|\right)^2\right] \tag{53}$$

$$\leq p_j N \sum_{i=1}^{p_j N} \mathbb{E}_{\mathcal{C}_k}[|\mathcal{C}_k(t) \cap U_i|^2] \tag{54}$$

$$\leq p_j^2 N^2 \frac{c_1 \mu^2}{\alpha^2 \lambda^2 \Delta^4 D'} \tag{55}$$

where (54) holds since $\left\|\sum_{i=1}^n \mathbf{a}_i\right\|^2 \leq n \sum_{i=1}^n \|\mathbf{a}_i\|^2$ for any $\mathbf{a}_1, \ldots, \mathbf{a}_n \in \mathbb{R}^d$ [47]. □

On the other hand, due to the channel noise and gradient compression, the standard convergence analysis for clustered FL ( [19], [20], [21]) does not apply to our over-the-air clustered FL problem. As such, we next introduce a few lemmas that will be instrumental in our further analysis. The following lemmas are proved under the condition that

$$|\mathcal{C}_k(t) \cap \mathcal{C}_k^*| \geq \frac{1}{4} pN \tag{56}$$

which holds with probability at least $1-2\exp(-cpN)$ according to Lemma 3.

*Lemma 6: For all clusters $k \in [K]$ and for any $t \in [T]$, the following holds for the desketched (decompressed) gradients within each cluster $k \in [K]$,*

$$\mathbb{E}_{\mathbf{R},\xi,\mathcal{C}_k}\Big[\Big\|\frac{1}{|\mathcal{C}_k(t)|}\sum_{i\in\mathcal{C}_k(t)}\eta\nabla F_i\left(\mathbf{w}_k(t),\mathcal{Z}_i(t)\right) -\mathbf{R}^H(t)\mathbf{R}(t)\Big(\frac{1}{|\mathcal{C}_k(t)|}\sum_{i\in\mathcal{C}_k(t)}\eta\nabla F_i\left(\mathbf{w}_k(t),\mathcal{Z}_i(t)\right)\Big)\Big\|^2\Big] \le 3\frac{d}{b}\eta^2(K+2)\Big(\big\|\nabla F_1(\mathbf{w}_1(t))\big\|^2+\frac{4v^2}{pND'} +\frac{144c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}L^2+\frac{16}{p^2}c_1\frac{\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big) \tag{57}$$

*Proof:* Without loss of generality, we prove (57) for $k=1$. Then,

$$\mathbb{E}_{\mathbf{R},\xi,\mathcal{C}_1}\Big[\Big\|\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right) -\mathbf{R}^H(t)\mathbf{R}(t)\Big(\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big)\Big\|^2\Big] \le 3\frac{d}{b}\mathbb{E}_{\xi,\mathcal{C}_1}\Big[\Big\|\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\|^2\Big] \tag{58}$$

$$= 3\frac{d}{b}\eta^2\mathbb{E}_{\xi,\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)|^2}\Big\|\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_1^*}\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right) +\sum_{j=2}^{K}\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_j^*}\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\|^2\Big] \le 3\frac{d}{b}\eta^2\mathbb{E}_{\xi,\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2}\Big\||\mathcal{C}_1(t)\cap\mathcal{C}_1^*|\nabla F_1(\mathbf{w}_1(t)) +\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_1^*}\left(\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)-\nabla F_1(\mathbf{w}_1(t))\right) +\sum_{j=2}^{K}|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\nabla F_j(\mathbf{w}_1(t)) +\sum_{j=2}^{K}\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_j^*}\left(\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)-\nabla F_j(\mathbf{w}_1(t))\right)\Big\|^2\Big] \tag{59}$$

$$\le 3\frac{d}{b}\eta^2(K+2)\Big(\big\|\nabla F_1(\mathbf{w}_1(t))\big\|^2+\mathbb{E}_{\xi,\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2} \times\Big\|\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_1^*}\left(\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)-\nabla F_1(\mathbf{w}_1(t))\right)\Big\|^2\Big] +\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2}\big\||\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\nabla F_j(\mathbf{w}_1(t))\big\|^2\Big] +\mathbb{E}_{\xi,\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2}\Big\|\sum_{j=2}^{K}\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_j^*}\left(\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)-\nabla F_j(\mathbf{w}_1(t))\right)\Big\|^2\Big]\Big) \tag{60}$$

$$= 3\frac{d}{b}\eta^2(K+2)\Big(\big\|\nabla F_1(\mathbf{w}_1(t))\big\|^2+\mathbb{E}_{\xi,\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2} \times\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_1^*}\big\|\left(\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)-\nabla F_1(\mathbf{w}_1(t))\right)\big\|^2\Big] +\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2}\big\||\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\nabla F_j(\mathbf{w}_1(t))\big\|^2\Big] +\sum_{j=2}^{K}\mathbb{E}_{\xi,\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2}\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_j^*}\big\|\left(\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)-\nabla F_j(\mathbf{w}_1(t))\right)\big\|^2\Big]\Big) \tag{61}$$

$$\le 3\frac{d}{b}\eta^2(K+2)\Big(\big\|\nabla F_1(\mathbf{w}_1(t))\big\|^2+\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2} \times|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\Big]\frac{v^2}{D'}+\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|}\Big]\frac{v^2}{D'} +\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2}\big\||\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\nabla F_j(\mathbf{w}_1(t))\big\|^2\Big]\Big) \tag{62}$$

$$\le 3\frac{d}{b}\eta^2(K+2)\Big(\big\|\nabla F_1(\mathbf{w}_1(t))\big\|^2+\frac{4}{pN}\frac{v^2}{D'} +\frac{16}{p^2N^2}\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|^2]\big\|\nabla F_j(\mathbf{w}_1(t))\big\|^2 +\frac{16}{p^2N^2}\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|]\frac{v^2}{D'}\Big) \tag{63}$$

$$\le 3\frac{d}{b}\eta^2(K+2)\Big(\big\|\nabla F_1(\mathbf{w}_1(t))\big\|^2+\frac{4}{pN}\frac{v^2}{D'} +\frac{16}{p^2N^2}\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|^2]L^2\big\|\mathbf{w}_1(t)-\mathbf{w}_j^*\big\|^2 +\frac{16}{p^2N^2}\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|]\frac{v^2}{D'}\Big) \tag{64}$$

$$\le 3\frac{d}{b}\eta^2(K+2)\Big(\big\|\nabla F_1(\mathbf{w}_1(t))\big\|^2+\frac{4}{pN}\frac{v^2}{D'} +\frac{16}{p^2N^2}\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|^2]9L^2 +\frac{16}{p^2N^2}\sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|]\frac{v^2}{D'}\Big) \tag{65}$$

$$\le 3\frac{d}{b}\eta^2(K+2)\Big(\big\|\nabla F_1(\mathbf{w}_1(t))\big\|^2+\frac{4}{pN}\frac{v^2}{D'} +\frac{16}{p^2N^2}c_1\frac{\mu^2N^2}{\alpha^2\lambda^2\Delta^4D'}9L^2+\frac{16}{p^2N^2}c_1\frac{\mu^2N}{\alpha^2\lambda^2\Delta^4D'}\frac{v^2}{D'}\Big) \tag{66}$$

$$= 3\frac{d}{b}\eta^2(K+2)\Big(\big\|\nabla F_1(\mathbf{w}_1(t))\big\|^2+\frac{4}{pN}\frac{v^2}{D'} +\frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2+\frac{16}{p^2}c_1\frac{\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big) \tag{67}$$

where (58) follows from Lemma 4; (59) holds since $|\mathcal{C}_1(t)| \ge |\mathcal{C}_1(t)\cap\mathcal{C}_1^*|$, (60) holds since $\|\sum_{i=1}^n \mathbf{a}_i\|^2 \le n\sum_{i=1}^n\|\mathbf{a}_i\|^2$ for any $\mathbf{a}_1,\ldots,\mathbf{a}_n\in\mathbb{R}^d$ [47]; (61) holds since $\mathbb{E}_\xi[\nabla F_i(\mathbf{w},\xi)] = \nabla F_j(\mathbf{w})$ for all $i\in\mathcal{C}_j^*$ and $\xi\in\mathcal{D}_i$; (62) follows from

Lemma 1; (63) follows from (56); (64) follows from Assumption 1. In (65), we leverage Assumption 4 as follows,

$$\left\|\mathbf{w}_1(t)-\mathbf{w}_j^*\right\| \le \|\mathbf{w}_j^*\|+\|\mathbf{w}_1^*\|+\|\mathbf{w}_1(t)-\mathbf{w}_1^*\| \quad (68)$$

$$\le 1+1+\left(\frac{1}{2}-\alpha\right)\sqrt{\frac{\lambda}{L}}\times 2 \le 3 \quad (69)$$

since $\lambda \le L$ according to Assumption 1 (convexity and smoothness) and

$$\Delta := \min_{k\neq k'} \|\mathbf{w}_k^*-\mathbf{w}_{k'}^*\| \le \min_{k\neq k'}(\|\mathbf{w}_k^*\|+\|\mathbf{w}_{k'}^*\|) \lesssim 2. \quad (70)$$

Finally (66) follows from Lemma 5. □

*Lemma 7: For all $k \in [K]$, the following holds for the received noise (after desketching),*

$$\mathbb{E}_{\mathbf{H},\mathbf{R},\mathbf{n},\xi}\left[\frac{1}{P_k(t)}\left\|\mathbf{R}^H(t)\mathbf{n}_k(t)\right\|^2\right] \le \frac{H_{max}}{P_{min}} d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right) \quad (71)$$

*where $P_{min} := \min_{i\in[N]} P_{T,i}$ and $H_{max}$ is the upper bound of $\mathbb{E}_{\mathbf{H}}\left\|\mathbf{H}_i^\dagger(t)\mathbf{A}_k\right\|_F^2$ as defined in Assumption 4, whereas $\mathbf{n}_k(t)$ is a $b\times 1$ dimensional vector from (31), $P_k(t)$ is defined in (17), and $\mathbf{A}_k$ is given in (18).*

*Proof:* First, note that $\mathbb{E}_{\mathbf{R},\mathbf{n}}\left[\left\|\mathbf{R}^H(t)\mathbf{n}_k(t)\right\|^2\right] = db\sigma_R^2\sigma_n^2 = d\sigma_n^2$. From (17), we observe that,

$$P_k(t) = \min_{i\in\mathcal{C}_k(t)} \frac{P_{T,i}}{\left\|\mathbf{H}_i^\dagger(t)\mathbf{A}_k\right\|_F^2 \|\overline{\mathbf{g}}_i(t)\|^2} \ge \frac{P_{min}}{\left\|\mathbf{H}_i^\dagger(t)\mathbf{A}_k\right\|_F^2 \max_{i\in\mathcal{C}_k(t)}\|\eta\nabla F_i(\mathbf{w}_k(t),\mathcal{Z}_i(t))\|^2}$$

Let $u \in \mathcal{C}_k(t)$ be the user such that

$$\|\eta\nabla F_u(\mathbf{w}_k(t),\mathcal{Z}_u(t))\|^2 = \max_{i\in\mathcal{C}_k(t)}\|\eta\nabla F_i(\mathbf{w}_k(t),\mathcal{Z}_i(t))\|^2$$

and let $u \in \mathcal{C}_j^*$ for some $j \in [K]$. Then,

$$\mathbb{E}_{\mathbf{H},\mathbf{R},\mathbf{n},\xi}\left[\frac{1}{P_k(t)}\left\|\mathbf{R}^H(t)\mathbf{n}_k(t)\right\|^2\right] \le \frac{1}{P_{min}} d\sigma_n^2 \mathbb{E}_{\mathbf{H},\xi}\left[\left\|\mathbf{H}_i^\dagger(t)\mathbf{A}_k\right\|_F^2 \times \max_{i\in\mathcal{C}_k(t)}\|\eta\nabla F_i(\mathbf{w}_k(t),\mathcal{Z}_i(t))\|^2\right] \quad (72)$$

$$\le \frac{H_{max}}{P_{min}} d\sigma_n^2\eta^2\mathbb{E}_\xi\left[\|\nabla F_j(\mathbf{w}_k(t)) + \nabla F_u(\mathbf{w}_k(t),\mathcal{Z}_u(t)) - \nabla F_j(\mathbf{w}_k(t))\|^2\right] \quad (73)$$

$$= \frac{H_{max}}{P_{min}} d\sigma_n^2\eta^2\Big(\|\nabla F_j(\mathbf{w}_k(t))\|^2 + \mathbb{E}_\xi\left[\|\nabla F_u(\mathbf{w}_k(t),\mathcal{Z}_u(t)) - \nabla F_j(\mathbf{w}_k(t))\|^2\right]\Big) \quad (74)$$

$$\le \frac{H_{max}}{P_{min}} d\sigma_n^2\eta^2\left(L^2\|\mathbf{w}_k(t)-\mathbf{w}_j^*\|^2+\frac{v^2}{D'}\right) \quad (75)$$

$$\le \frac{H_{max}}{P_{min}} d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right) \quad (76)$$

where (73) follows from Assumption 4, (74) is from the unbiasedness of local gradients, and $\mathbb{E}_\xi[\nabla F_u(\mathbf{w}_k(t),\mathcal{Z}_u(t))] = \nabla F_j(\mathbf{w}_k(t))$; (75) is from Lemma 1 and Assumption 1; (76) is from Assumption 4 and from (70). Finally,

$$\|\mathbf{w}_k(t)-\mathbf{w}_j^*\| = \|\mathbf{w}_k(t)-\mathbf{w}_k^*\| \le \left(\frac{1}{2}-\alpha\right)\sqrt{\frac{\lambda}{L}}\Delta \le 1 \quad \text{for } j=k \quad (77)$$

$$\|\mathbf{w}_k(t)-\mathbf{w}_j^*\| \le \|\mathbf{w}_k(t)-\mathbf{w}_k^*\|+\|\mathbf{w}_k^*\|+\|\mathbf{w}_j^*\| \le \left(\frac{1}{2}-\alpha\right)\sqrt{\frac{\lambda}{L}}\Delta+1+1 \le 3 \text{ for } j\neq k \quad (78)$$

*Lemma 8: For all clusters $k \in [K]$ and for any $t \in [T]$, the squared-difference of the average of the local gradients within cluster $k$ and the gradient of the cluster loss function $F_k(\cdot)$ can be bounded in expectation as follows*:

$$\mathbb{E}_{\xi,\mathcal{C}_k}\left[\left\|\nabla F_k(\mathbf{w}_k(t)) - \frac{1}{|\mathcal{C}_k(t)|}\sum_{i\in\mathcal{C}_k(t)}\nabla F_i(\mathbf{w}_k(t),\mathcal{Z}_i(t))\right\|^2\right] \le (K+1)\Big(\frac{4v^2}{pND'} + \frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4p^2N}\frac{v^2}{(D')^2} + 320L^2\frac{c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big) \quad (79)$$

*Proof:* Without loss of generality, assume $k = 1$. Then,

$$\mathbb{E}_{\xi,\mathcal{C}_1}\left[\left\|\nabla F_1(\mathbf{w}_1(t)) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\right\|^2\right]$$
$$= \mathbb{E}_{\xi,\mathcal{C}_1}\Big[\Big\|\nabla F_1(\mathbf{w}_1(t)) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_1^*}\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t)) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{j=2}^K\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_j^*}(\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t)) - \nabla F_j(\mathbf{w}_1(t))) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{j=2}^K\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_j^*}\nabla F_j(\mathbf{w}_1(t))\Big\|^2\Big]$$
$$= \mathbb{E}_{\xi,\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)|^2}\Big\|\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_1^*}\big(\nabla F_1(\mathbf{w}_1(t)) - \nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\big) - \sum_{j=2}^K\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_j^*}\big(\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t)) - \nabla F_j(\mathbf{w}_1(t))\big) + \sum_{j=2}^K|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\big(\nabla F_1(\mathbf{w}_1(t)) - \nabla F_j(\mathbf{w}_1(t))\big)\Big\|^2\Big] \quad (80)$$

$$\le (K+1)\mathbb{E}_{\xi,\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)|^2}\Big\|\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_1^*}\big(\nabla F_1(\mathbf{w}_1(t)) - \nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\big)\Big\|^2 + \frac{1}{|\mathcal{C}_1(t)|^2}\Big\|\sum_{j=2}^K\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_j^*}\big(\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t)) - \nabla F_j(\mathbf{w}_1(t))\big)\Big\|^2 + \frac{1}{|\mathcal{C}_1(t)|^2}\sum_{j=2}^K\||\mathcal{C}_1(t)\cap\mathcal{C}_j^*| \times \big(\nabla F_1(\mathbf{w}_1(t)) - \nabla F_j(\mathbf{w}_1(t))\big)\|^2\Big] \quad (81)$$

$$= (K+1)\mathbb{E}_{\xi,\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)|^2}\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_1^*}\|\nabla F_1(\mathbf{w}_1(t)) - \nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\|^2 + \frac{1}{|\mathcal{C}_1(t)|^2}\sum_{j=2}^K\sum_{i\in\mathcal{C}_1(t)\cap\mathcal{C}_j^*}$$

$$\|\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))-\nabla F_j(\mathbf{w}_1(t))\|^2 + \frac{1}{|\mathcal{C}_1(t)|^2}\sum_{j=2}^{K}\Big|\,|\mathcal{C}_1(t)\cap\mathcal{C}_j^*| \times\big(\nabla F_1(\mathbf{w}_1(t))-\nabla F_j(\mathbf{w}_1(t))\big)\Big\|^2\Big] \tag{82}$$

$$\le (K+1)\Big(\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)|^2}|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|\Big]\frac{v^2}{D'} + \sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)|^2}|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\Big]\frac{v^2}{D'} + \sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)|^2}\Big|\,|\mathcal{C}_1(t)\cap\mathcal{C}_j^*| \times\big(\nabla F_1(\mathbf{w}_1(t))-\nabla F_j(\mathbf{w}_1(t))\big)\Big\|^2\Big]\Big) \tag{83}$$

$$\le (K+1)\Big(\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|}\Big]\frac{v^2}{D'} + \sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2}|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\Big]\frac{v^2}{D'} + \sum_{j=2}^{K}\mathbb{E}_{\mathcal{C}_1}\Big[\frac{1}{|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|^2}\Big|\,|\mathcal{C}_1(t)\cap\mathcal{C}_j^*| \times\big(\nabla F_1(\mathbf{w}_1(t))-\nabla F_j(\mathbf{w}_1(t))\big)\Big\|^2\Big]\Big) \tag{84}$$

$$\le (K+1)\Big(\frac{4}{pN}\frac{v^2}{D'} + \sum_{j=2}^{K}\frac{16}{p^2N^2}\mathbb{E}_{\mathcal{C}_1}\big[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\big]\frac{v^2}{D'} + \sum_{j=2}^{K}\frac{16}{p^2N^2}\mathbb{E}_{\mathcal{C}_1}\Big[\big|\,|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\big(\nabla F_1(\mathbf{w}_1(t)) - \nabla F_j(\mathbf{w}_1(t))\big)\big\|^2\Big]\Big) \tag{85}$$

$$\le (K+1)\Big(\frac{4}{pN}\frac{v^2}{D'} + \sum_{j=2}^{K}\frac{16}{p^2N^2}\mathbb{E}_{\mathcal{C}_1}\big[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\big]\frac{v^2}{D'} + \sum_{j=2}^{K}\frac{16}{p^2N^2}\mathbb{E}_{\mathcal{C}_1}\big[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|^2\big]2\big(\|\nabla F_1(\mathbf{w}_1(t))\|^2 + \|\nabla F_j(\mathbf{w}_1(t))\|^2\big)\Big) \tag{86}$$

$$\le (K+1)\Big(\frac{4}{pN}\frac{v^2}{D'} + \sum_{j=2}^{K}\frac{16}{p^2N^2}\mathbb{E}_{\mathcal{C}_1}\big[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|\big]\frac{v^2}{D'} + \sum_{j=2}^{K}\frac{16}{p^2N^2}\mathbb{E}_{\mathcal{C}_1}\big[|\mathcal{C}_1(t)\cap\mathcal{C}_j^*|^2\big]2L^2\big(\|\mathbf{w}_1(t)-\mathbf{w}_1^*\|^2 + \|\mathbf{w}_1(t)-\mathbf{w}_j^*\|^2\big)\Big) \tag{87}$$

$$\le (K+1)\Big(\frac{4}{pN}\frac{v^2}{D'} + \frac{16}{p^2N^2}\frac{c_1\mu^2N}{\alpha^2\lambda^2\Delta^4}\frac{v^2}{(D')^2} + \frac{16}{p^2N^2}\frac{c_1\mu^2N^2}{\alpha^2\lambda^2\Delta^4D'}2L^2(1+9)\Big) \tag{88}$$

where (81), (86) hold since $\|\sum_{i=1}^n\mathbf{a}_i\|^2\le n\sum_{i=1}^n\|\mathbf{a}_i\|^2$ for any $\mathbf{a}_1,\dots,\mathbf{a}_n\in\mathbb{R}^d$ [47]; (82) is from the unbiasedness of the local loss function $\mathbb{E}_\xi[F_i(\mathbf{w},\xi)]=F_j(\mathbf{w})$ for $i\in\mathcal{C}_j^*$, $\xi\in\mathcal{D}_i$; (83) is from Lemma 1; (84) holds since $|\mathcal{C}_1(t)|\ge|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|$; (85) is from (56); (87) is from Assumption 1; (88) is from (77), (78), Lemma 5, which concludes the proof. □

Using the above lemmas, we next proceed with the convergence. Without loss of generality, we consider cluster 1 (same analysis holds for all clusters). From the smoothness of the loss functions in Assumption 1 and by taking the expectation over all randomness (denoted by $\mathbb{E}[\cdot]$),

$$\mathbb{E}[F_1(\mathbf{w}_1(t+1))] \le \mathbb{E}[F_1(\mathbf{w}_1(t))] + \mathbb{E}[\langle\nabla F_1(\mathbf{w}_1(t)),\mathbf{w}_1(t+1)-\mathbf{w}_1(t)\rangle] + \frac{L}{2}\mathbb{E}\Big[\|\mathbf{w}_1(t+1)-\mathbf{w}_1(t)\|^2\Big] \tag{89}$$

From (56) and Assumption 4, we then have,

$$\mathbb{E}\big[\,\|\mathbf{w}_1(t+1)-\mathbf{w}_1(t)\|^2\,\big] = \mathbb{E}\Big[\Big\|\mathbf{R}^H(t)\mathbf{R}(t)\Big(\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\Big) + \frac{1}{|\mathcal{C}_1(t)|\sqrt{P_1(t)}}\mathbf{R}^H(t)\mathbf{n}_1(t)\Big\|^2\Big] \tag{90}$$

$$= \mathbb{E}\Big[\Big\|\mathbf{R}^H(t)\mathbf{R}(t)\Big(\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\Big) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t)) + \frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t)) + \frac{1}{|\mathcal{C}_1(t)|\sqrt{P_1(t)}}\mathbf{R}^H(t)\mathbf{n}_1(t)\Big\|^2\Big] \tag{91}$$

$$= \mathbb{E}\Big[\Big\|\mathbf{R}^H(t)\mathbf{R}(t)\Big(\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\Big) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\Big\|^2\Big] + \mathbb{E}\Big[\Big\|\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\Big\|^2\Big] + \mathbb{E}\Big[\Big\|\frac{1}{|\mathcal{C}_1(t)|\sqrt{P_1(t)}}\mathbf{R}^H(t)\mathbf{n}_1(t)\Big\|^2\Big] \tag{92}$$

$$\le 3\frac{d}{b}\eta^2(K+2)\Big(\mathbb{E}\big[\,\|\nabla F_1(\mathbf{w}_1(t))\|^2\,\big] + \frac{4}{pN}\frac{v^2}{D'} + \frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2 + \frac{16c_1}{p^2}\frac{\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big) + \mathbb{E}\Big[\Big\|\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i(\mathbf{w}_1(t),\mathcal{Z}_i(t))\Big\|^2\Big] + \frac{16}{p^2N^2}\frac{H_{max}}{P_{min}}d\sigma_n^2\eta^2\Big(9L^2+\frac{v^2}{D'}\Big) \tag{93}$$

where (92) follows from the unbiasedness of sketching from Lemma 4 and that $\mathbb{E}_{\mathbf{n}}[\mathbf{n}_1(t)]=0$, and (93) follows from Lemmas 6-7 and (56). We further find that,

$$\mathbb{E}[\langle\nabla F_1(\mathbf{w}_1(t)),\mathbf{w}_1(t+1)-\mathbf{w}_1(t)\rangle] = -\mathbb{E}\Big[\Big\langle\nabla F_1(\mathbf{w}_1(t)),\mathbf{R}^H(t)\mathbf{R}(t) \tag{94}$$

$$\times \Big(\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big) + \frac{1}{|\mathcal{C}_1(t)|\sqrt{P_1(t)}}\mathbf{R}^H(t)\mathbf{n}_1(t)\Big\rangle\Big] \tag{95}$$

$$= -\mathbb{E}\Big[\Big\langle\nabla F_1\left(\mathbf{w}_1(t)\right),\mathbf{R}^H(t)\mathbf{R}(t) \times \Big(\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\rangle + \Big\langle\nabla F_1\left(\mathbf{w}_1(t)\right),\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\eta\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\rangle + \Big\langle\nabla F_1\left(\mathbf{w}_1(t)\right),\frac{1}{|\mathcal{C}_1(t)|\sqrt{P_1(t)}}\mathbf{R}^H(t)\mathbf{n}_1(t)\Big\rangle\Big] \tag{96}$$

$$= -\eta\mathbb{E}\Big[\Big\langle\nabla F_1\left(\mathbf{w}_1(t)\right),\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\rangle\Big] \tag{97}$$

$$= -\frac{1}{2}\eta\mathbb{E}\Big[\left\|\nabla F_1\left(\mathbf{w}_1(t)\right)\right\|^2 + \Big\|\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\|^2 - \Big\|\nabla F_1\left(\mathbf{w}_1(t)\right) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\|^2\Big] \tag{98}$$

where (97) follows from the unbiasedness of sketching from Lemma 4 and that $\mathbb{E}_{\mathbf{n}}[\mathbf{n}_1(t)] = \mathbf{0}$. Equation (98) holds since, for any vector $\mathbf{a} \in \mathbb{R}^d$, $\langle\mathbf{a}_1,\mathbf{a}_2\rangle = \frac{1}{2}\left(\|\mathbf{a}_1\|^2 + \|\mathbf{a}_2\|^2 - \|\mathbf{a}_1-\mathbf{a}_2\|^2\right)$ [56]. Using (93) and (98), we rewrite (89) as follows,

$$\mathbb{E}\left[F_1\left(\mathbf{w}_1(t+1)\right)\right] \le \mathbb{E}\left[F_1\left(\mathbf{w}_1(t)\right)\right] - \frac{\eta-\eta^2L}{2}\mathbb{E}\Big[\Big\|\frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\|^2\Big] - \frac{\eta}{2}\mathbb{E}\left[\left\|\nabla F_1\left(\mathbf{w}_1(t)\right)\right\|^2\right] + \frac{\eta}{2}\mathbb{E}\Big[\Big\|\nabla F_1\left(\mathbf{w}_1(t)\right) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\|^2\Big] + \frac{L}{2}\frac{3d}{b}\eta^2(K+2) \times \Big(\mathbb{E}\left[\left\|\nabla F_1\left(\mathbf{w}_1(t)\right)\right\|^2\right] + \frac{4}{pN}\frac{v^2}{D'} + \frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2 + \frac{16c_1\mu^2v^2}{p^2\alpha^2\lambda^2\Delta^4(D')^2N}\Big) + \frac{L}{2}\frac{16H_{max}}{p^2N^2P_{min}}d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right)$$

$$\le \mathbb{E}\left[F_1(\mathbf{w}_1(t))\right] - \Big(\frac{\eta}{2} - L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big)\mathbb{E}\Big[\left\|\nabla F_1\left(\mathbf{w}_1(t)\right)\right\|^2\Big] + \frac{\eta}{2}\mathbb{E}\Big[\Big\|\nabla F_1\left(\mathbf{w}_1(t)\right) - \frac{1}{|\mathcal{C}_1(t)|}\sum_{i\in\mathcal{C}_1(t)}\nabla F_i\left(\mathbf{w}_1(t),\mathcal{Z}_i(t)\right)\Big\|^2\Big] + L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big(\frac{4}{pN}\frac{v^2}{D'} + \frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2 + \frac{16c_1}{p^2}\frac{\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big) + \frac{8LH_{max}}{p^2N^2P_{min}}d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right) \tag{99}$$

$$\le \mathbb{E}\left[F_1(\mathbf{w}_1(t))\right] - \Big(\frac{\eta}{2} - L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big)\mathbb{E}\Big[\left\|\nabla F_1\left(\mathbf{w}_1(t)\right)\right\|^2\Big] + \frac{\eta}{2}(K+1)\Big(\frac{4v^2}{pND'} + \frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4p^2N}\frac{v^2}{(D')^2} + 320L^2\frac{c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big) + L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big(\frac{4}{pN}\frac{v^2}{D'} + \frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2 + \frac{16c_1}{p^2}\frac{\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big) + \frac{8L}{p^2N^2}\frac{H_{max}}{P_{min}}d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right) \tag{100}$$

$$\le \mathbb{E}\left[F_1\left(\mathbf{w}_1(t)\right)\right] - \frac{\eta}{4}\mathbb{E}\left[\left\|\nabla F_1\left(\mathbf{w}_1(t)\right)\right\|^2\right] + \frac{\eta}{2}(K+1) \times \Big(\frac{4v^2}{pND'} + \frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4p^2N}\frac{v^2}{(D')^2} + 320L^2\frac{c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big) + L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big(\frac{4}{pN}\frac{v^2}{D'} + \frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2 + \frac{16c_1}{p^2}\frac{\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big) + \frac{8LH_{max}}{p^2N^2P_{min}}d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right) \tag{101}$$

$$\le \mathbb{E}\left[F_1\left(\mathbf{w}_1(t)\right)\right] - \lambda\frac{\eta}{2}\mathbb{E}\left[F_1(\mathbf{w}_1(t)) - F_1(\mathbf{w}_1^*)\right] + \frac{\eta}{2}(K+1)\Big(\frac{4v^2}{pND'} + \frac{16c_1\mu^2v^2}{\alpha^2\lambda^2\Delta^4p^2N(D')^2} + \frac{320L^2c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big) + L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big(\frac{4}{pN}\frac{v^2}{D'} + \frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2 + \frac{16c_1\mu^2v^2}{p^2\alpha^2\lambda^2\Delta^4(D')^2N}\Big) + \frac{8LH_{max}}{p^2N^2P_{min}}d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right) \tag{102}$$

where (99) holds since $\eta \le \frac{1}{L}$ from (33); (100) holds from Lemma 8; (101) is from (33); and (102) is from Assumption 1. Next, after generating (102) for $t = T-1$ and subtracting $F_1(\mathbf{w}_1^*)$ from both sides, we observe,

$$\mathbb{E}\left[F_1(\mathbf{w}_1(T)) - F_1(\mathbf{w}_1^*)\right] \le (1-\frac{\lambda\eta}{2})\mathbb{E}\left[F_1(\mathbf{w}_1(T-1)) - F_1(\mathbf{w}_1^*)\right] + \frac{\eta}{2}(K+1)\Big(\frac{4v^2}{pND'} + \frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4p^2N}\frac{v^2}{(D')^2} + \frac{320L^2c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big) + L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big(\frac{4}{pN}\frac{v^2}{D'} + \frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2 + \frac{16c_1}{p^2}\frac{\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big) + \frac{8LH_{max}}{p^2N^2P_{min}}d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right)$$

$$\le (1-\frac{\lambda\eta}{2})^T\mathbb{E}\left[F_1(\mathbf{w}_1(0)) - F_1(\mathbf{w}_1^*)\right] + \sum_{t=0}^{T-1}(1-\frac{\lambda\eta}{2})^t\Big(\frac{\eta}{2}(K+1)\Big(\frac{4v^2}{pND'} + \frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4p^2N}\frac{v^2}{(D')^2} + 320L^2\frac{c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big) + L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big(\frac{4}{pN}\frac{v^2}{D'} \tag{103}$$

$$+\frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4 D'}L^2+\frac{16c_1}{p^2}\frac{\mu^2 v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big)+\frac{8L}{p^2N^2}\frac{H_{max}}{P_{min}}d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right)\Bigg) \tag{104}$$

$$=(1-\frac{\lambda\eta}{2})^T\mathbb{E}\left[F_1(\mathbf{w}_1(0))-F_1(\mathbf{w}_1^*)\right]+\frac{1-(1-\frac{\lambda\eta}{2})^{T-1}}{1-(1-\frac{\lambda\eta}{2})}\times\Bigg(\frac{\eta}{2}(K+1)\Big(\frac{4v^2}{pND'}+\frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4p^2N}\frac{v^2}{(D')^2}+320L^2\frac{c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big)+L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big(\frac{4}{pN}\frac{v^2}{D'}+\frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2+\frac{16c_1}{p^2}\frac{\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big)+\frac{8L}{p^2N^2}\frac{H_{max}}{P_{min}}d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right)\Bigg) \tag{105}$$

$$\leq(1-\frac{\lambda\eta}{2})^T\mathbb{E}\left[F_1(\mathbf{w}_1(0))-F_1(\mathbf{w}_1^*)\right]+\frac{2}{\lambda\eta}\Bigg(\frac{\eta}{2}(K+1)\times\Big(\frac{4v^2}{pND'}+\frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4p^2N}\frac{v^2}{(D')^2}+320L^2\frac{c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big)+L\frac{3}{2}\frac{d}{b}\eta^2(K+2)\Big(\frac{4}{pN}\frac{v^2}{D'}+\frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2+\frac{16c_1}{p^2}\frac{\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big)+\frac{8L}{p^2N^2}\frac{H_{max}}{P_{min}}d\sigma_n^2\eta^2\left(9L^2+\frac{v^2}{D'}\right)\Bigg) \tag{106}$$

$$=(1-\frac{\lambda\eta}{2})^T\mathbb{E}\left[F_1(\mathbf{w}_1(0))-F_1(\mathbf{w}_1^*)\right]+\frac{1}{\lambda}(K+1)\times\Big(\frac{4v^2}{pND'}+\frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4p^2N}\frac{v^2}{(D')^2}+320L^2\frac{c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big)+3\frac{L}{\lambda}\frac{d}{b}\eta(K+2)\Big(\frac{4}{pN}\frac{v^2}{D'}+\frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2+\frac{16}{p^2}\frac{c_1\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big)+\frac{L}{\lambda}\frac{16}{p^2N^2}\frac{H_{max}}{P_{min}}d\sigma_n^2\eta\Big(9L^2+\frac{v^2}{D'}\Big)$$

$$\leq(1-\frac{\lambda\eta}{2})^T\mathbb{E}\left[F_1(\mathbf{w}_1(0))-F_1(\mathbf{w}_1^*)\right]+\frac{1}{\lambda}(K+1)\times\Big(\frac{4v^2}{pND'}+\frac{16c_1\mu^2}{\alpha^2\lambda^2\Delta^4p^2N}\frac{v^2}{(D')^2}+320L^2\frac{c_1\mu^2}{p^2\alpha^2\lambda^2\Delta^4D'}\Big)+\frac{1}{2\lambda}\Big(\frac{4}{pN}\frac{v^2}{D'}+\frac{144c_1}{p^2}\frac{\mu^2}{\alpha^2\lambda^2\Delta^4D'}L^2+\frac{16}{p^2}\frac{c_1\mu^2v^2}{\alpha^2\lambda^2\Delta^4(D')^2N}\Big)+\frac{1}{\lambda}\frac{16}{p^2N^2}\Big(9L^2+\frac{v^2}{D'}\Big) \tag{107}$$

where (104) follows from recursively expressing $\mathbb{E}\left[F_1(\mathbf{w}_1(t+1))-F_1(\mathbf{w}_1^*)\right]$ using (102); (106) holds since $\frac{1-(1-\frac{\lambda\eta}{2})^{T-1}}{1-(1-\frac{\lambda\eta}{2})}\leq\frac{1}{1-(1-\frac{\lambda\eta}{2})}$; and (107) follows from (33). Note that (107) is conditioned on $|\mathcal{C}_1(t)\cap\mathcal{C}_1^*|\geq\frac{1}{4}pN$, which holds with probability at least $1-2\exp(-cpN)$ for each cluster at any round. Then, from the union bound, convergence is guaranteed within an optimality gap of no greater than $O\big(\frac{1}{pND'}+\frac{1}{p^2N^2}+\frac{1}{p^2D'}\big)$ with probability at least $1-2KT\exp(-cpN)\ \forall k\in[K]$.

## REFERENCES


[1] H. U. Sami and B. Güler, "Over-the-air personalized federated learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 8777–8781.

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

[3] P. Kairouz and H. B. McMahan, "Advances and open problems in federated learning," *Found. Trends¥ Mach. Learn.*, vol. 14, pp. 1–120, Jun. 2021.

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

[5] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, Mar. 2020.

[6] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 491–506, Jan. 2020.

[7] Y. Mansour, M. Mohri, J. Ro, and A. T. Suresh, "Three approaches for personalization with applications to federated learning," 2020, *arXiv:2002.10619*.

[8] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Mach. Learn. Syst.*, 2020, pp. 429–450.

[9] T. Li, M. Sanjabi, A. Beirami, and V. Smith, "Fair resource allocation in federated learning," in *Proc. Int. Conf. Learn. Represent.*, 2020, pp. 1–27.

[10] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on non-IID data," in *Proc. Int. Conf. Learn. Rep. (ICLR)*, 2019, pp. 1–26.

[11] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-i.I.d. data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3400–3413, Sep. 2020.

[12] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 4427–4437.

[13] F. Chen, M. Luo, Z. Dong, Z. Li, and X. He, "Federated meta-learning with fast convergence and efficient communication," 2018, *arXiv:1802.07876*.

[14] A. Fallah, A. Mokhtari, and A. E. Ozdaglar, "Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach," in *Proc. Adv. Neural Inf. Process. Sys. (NeurIPS)*, 2020, pp. 3557–3568.

[15] C. T. Dinh, N. H. Tran, and T. D. Nguyen, "Personalized federated learning with Moreau envelopes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 21394–21405.

[16] K. Singhal, H. Sidahmed, Z. Garrett, S. Wu, J. Rush, and S. Prakash, "Federated reconstruction: Partially local federated learning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2021, pp. 11220–11232.

[17] Y. Luo, X. Liu, and J. Xiu, "Energy-efficient clustering to address data heterogeneity in federated learning," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.

[18] F. Sattler, K.-R. Müller, and W. Samek, "Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 8, pp. 3710–3722, Aug. 2021.

[19] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," *IEEE Trans. Inf. Theory*, vol. 68, no. 12, pp. 8076–8091, Dec. 2022.

[20] Y. Ruan and C. Joe-Wong, "FedSoft: Soft clustered federated learning with proximal local updating," in *Proc. 36th AAAI Conf. Artif. Intell.*, 2022, pp. 8124–8131.

[21] M. Nafea, E. Shin, and A. Yener, "Proportional fair clustered federated learning," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2022, pp. 2022–2027.

[22] Y. Kim, E. A. Hakim, J. Haraldson, H. Eriksson, J. M. B. da Silva, and C. Fischione, "Dynamic clustering in federated learning," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.

[23] C. Briggs, Z. Fan, and P. Andras, "Federated learning with hierarchical clustering of local updates to improve training on non-IID data," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2020, pp. 1–9.
[24] N. Ivkin, D. Rothchild, E. Ullah, I. Stoica, and R. Arora, "Communication-efficient distributed SGD with sketching," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 13144–13154.
[25] D. Rothchild et al., "FetchSGD: Communication-efficient federated learning with sketching," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 8253–8265.
[26] Z. Song, Z. Yu, and L. Zhang, "Iterative sketching and its application to federated learning," OpenReview, Tech. Rep., 2022. [Online]. Available: https://openreview.net/pdf?id=U_Jog0t3fAu
[27] Y. LeCun, C. Cortes, and C. Burges, "MNIST database," Courant Inst., New York, NY, USA, Tech. Rep., 2010. [Online]. Available: http://yann.lecun.com/exdb/mnist/
[28] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf
[29] E. Jothimurugesan, K. Hsieh, J. Wang, G. Joshi, and P. B. Gibbons, "Federated learning under distributed concept drift," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2023, pp. 5834–5853.
[30] H. Vardhan, A. Ghosh, and A. Mazumdar, "An improved algorithm for clustered federated learning," arXiv:2210.11538, 2022.
[31] M. Goldenbaum, H. Boche, and S. Stańczak, "Analyzing the space of functions analog-computable via wireless multiple-access channels," in *Proc. Int. Symp. Wireless Commun. Syst.*, 2011, pp. 779–783.
[32] M. Goldenbaum and S. Stanczak, "Robust analog function computation via wireless multiple-access channels," *IEEE Trans. Commun.*, vol. 61, no. 9, pp. 3863–3877, Sep. 2013.
[33] O. Salehi-Abari, H. Rahul, and D. Katabi, "Over-the-air function computation in sensor networks," 2016, *arXiv:1612.02307*.
[34] G. Zhu and K. Huang, "MIMO over-the-air computation for high-mobility multimodal sensing," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6089–6103, Aug. 2019.
[35] D. Wen, G. Zhu, and K. Huang, "Reduced-dimension design of MIMO over-the-air computing for data aggregation in clustered IoT networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5255–5268, Nov. 2019.
[36] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, May 2020.
[37] M. M. Amiri, T. M. Duman, and D. Gündüz, "Collaborative machine learning at the wireless edge with blind transmitters," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Nov. 2019, pp. 1–5.
[38] 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 Trans. Wireless Commun.*, vol. 20, no. 3, pp. 2120–2135, Mar. 2021.
[39] T. Sery, N. Shlezinger, K. Cohen, and Y. C. Eldar, "Over-the-air federated learning from heterogeneous data," *IEEE Trans. Signal Process.*, vol. 69, pp. 3796–3811, 2021.
[40] X. Cao, G. Zhu, J. Xu, Z. Wang, and S. Cui, "Optimized power control design for over-the-air federated edge learning," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 342–358, Jan. 2022.
[41] X. Ma, H. Sun, Q. Wang, and R. Q. Hu, "User scheduling for federated learning through over-the-air computation," in *Proc. IEEE 94th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2021, pp. 1–5.
[42] Y. Sun, S. Zhou, Z. Niu, and D. Gündüz, "Dynamic scheduling for over-the-air federated edge learning with energy constraints," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 227–242, Jan. 2022.
[43] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, "Differentially private aircomp federated learning with power adaptation harnessing receiver noise," in *Proc. IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–6.
[44] O. Aygün, M. Kazemi, D. Gündüz, and T. M. Duman, "Over-the-air federated edge learning with hierarchical clustering," 2022, *arXiv:2207.09232*.
[45] A. Madhan-Sohini, D. Dominic, N. Shah, and R. Prasad, "Over-the-air clustered wireless federated learning," 2022, *arXiv:2211.03363*.
[46] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-efficient SGD via gradient quantization and encoding," in *Proc. Adv. neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1709–1720.
[47] S. U. Stich, J. Cordonnier, and M. Jaggi, "Sparsified SGD with memory," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2018, pp. 4452–4463.
[48] J. Wangni, J. Wang, J. Liu, and T. Zhang, "Gradient sparsification for communication-efficient distributed optimization," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2018, pp. 1306–1316.
[49] Q. H. Spencer, A. L. Swindlehurst, and M. Haardt, "Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels," *IEEE Trans. Signal Process.*, vol. 52, no. 2, pp. 461–471, Feb. 2004.
[50] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," *SIAM Rev.*, vol. 60, no. 2, pp. 223–311, Jan. 2018.
[51] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 5132–5143.
[52] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2020, pp. 7611–7623.
[53] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, "Energy efficient federated learning over wireless communication networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1935–1949, Mar. 2021.
[54] C. A. Metzler, A. Mousavi, and R. G. Baraniuk, "Learned D-AMP: Principled neural network based compressive image recovery," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1772–1783.
[55] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, Jan. 2021.
[56] H. Yu, S. Yang, and S. Zhu, "Parallel restarted SGD with faster convergence and less communication: Demystifying why model averaging works for deep learning," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 5693–5700.

**Hasin Us Sami** (Graduate Student Member, IEEE) received the B.Sc. degree in electrical and electronic engineering from the Bangladesh University of Engineering and Technology, Dhaka, Bangladesh, in 2019. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of California at Riverside. His research interests include federated and distributed machine learning, information theory, secure and private computing, and wireless networks.

**Başak Güler** (Member, IEEE) received the B.Sc. degree in electrical and electronics engineering from Middle East Technical University (METU), Ankara, Turkey, and the Ph.D. degree from the Wireless Communications and Networking Laboratory, The Pennsylvania State University, in 2017. From 2018 to 2020, she was a Post-Doctoral Scholar at the University of Southern California. She is currently an Assistant Professor at the Department of Electrical and Computer Engineering, University of California, Riverside. She is a recipient of the 2022 NSF CAREER Award. Her research interests include information theory, distributed computing, machine learning, and wireless networks.