ON OVER-THE-AIR FEDERATED LEARNING OVER WIRELESS CHANNELS WITH ENERGY HARVESTING AND HETEROGENEOUS DATA DISTRIBUTION

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF

MASTER OF SCIENCE

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

By

Furkan Bağcı

August 2025

On Over-the-Air Federated Learning over Wireless Channels with Energy Harvesting and Heterogeneous Data Distribution
By Furkan Bağcı
August 2025

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Tolga Mete Duman (Advisor)

Orhan Arıkan

Javad Haghighat

Approved for the Graduate School of Engineering and Science:

Orhan Arıkan
Director of the Graduate School

ABSTRACT

ON OVER-THE-AIR FEDERATED LEARNING OVER WIRELESS CHANNELS WITH ENERGY HARVESTING AND HETEROGENEOUS DATA DISTRIBUTION

Furkan Bağcı
M.S. in Electrical and Electronics Engineering
Advisor: Tolga Mete Duman
August 2025

With the growing interest in machine learning (ML), federated learning (FL) has emerged as a prominent paradigm for collaboratively training high-quality models across decentralized edge devices, while minimizing server-side data access and preserving user privacy. In FL, multiple mobile devices (MDs) are used to collaboratively train a global model with coordination from a parameter server (PS), with devices using their local data to perform stochastic gradient descent (SGD) for the purpose of global training. While extensive research has been conducted on FL over wireless channels and its applications in practical scenarios, the need to investigate and develop solutions tailored to highly heterogeneous setups is an open research direction. This thesis aims to explore strategies and propose solutions for over-the-air (OTA) FL over wireless channels in such heterogeneous environments.

In the first part of this thesis, we focus on the scheduling strategies for the OTA FL with energy harvesting MDs and highly heterogeneous data distribution. We follow a paradigm to schedule users based on their data distribution characteristics and subsequently their transmitted updates, with the primary goal of improving the learning performance and making more efficient use of the limited harvested energy. We develop two scheduling approaches depending on whether the users' data distributions are known or unknown at the server. We provide a theoretical convergence analysis of the proposed OTA FL setup, which is also used to design scheduling strategies, and validate our findings through experimental analysis, demonstrating that scheduling strategies based on user characteristics can significantly enhance global learning performance and reduce redundancy in the system.

In the second part of the thesis, we study clustered federated learning (CFL) approaches in OTA FL setup, focusing on personalized global models depending on users' data characteristics for energy harvesting MDs. In this part, we assume that users are naturally partitioned into distinct clusters; for instance, each cluster may consist of users interested in different categories of sports. The primary goal is to train a dedicated model for each cluster to better capture their specific preferences, while simultaneously serving all the clusters using a single parameter server through over-the-air transmission to ensure communication efficiency. To enable the simultaneous serving of each cluster, we propose different combining methods at the server side, each designed for a specific level of available channel state information (CSI). Numerical results show that with a sufficient number of receiver antennas, it is possible to support simultaneous update transmissions from multiple clusters and achieve more specialized global models with improved performance.

Keywords: Federated learning, wireless communication, over-the-air transmission, energy harvesting devices, fading channels, clustered federated learning, heterogeneous data distribution.

ÖZET

ENERJİ HASADI VE HETEROJEN VERİ DAĞILIMI KOŞULLARINDA KABLOSUZ KANALLARDA HAVADAN BESLEMELİ FEDERE ÖĞRENME

Furkan Bağcı
Elektrik ve Elektronik Mühendisliği, Yüksek Lisans
Tez Danışmanı: Tolga Mete Duman
Ağustos 2025

Makine öğrenmesine (ML) olan ilginin giderek artmasıyla birlikte, federe öğrenme (FL), merkezi olmayan uç cihazlar arasında hem yüksek kaliteli modellerin iş birliği içinde eğitilmesini sağlayan, hem de sunucu tarafındaki kullanıcı verisinin erişimini en aza indiren ve kullanıcı gizliliğini koruyan bir paradigma olarak ortaya çıkmıştır. FL'de, birden fazla mobile cihaz (MD), bir parametre sunucusunun (PS) koordinasyonunda global bir modeli iş birliği içinde eğitmek için kullanılır; cihazlar, küresel eğitimi gerçekleştirmek amacıyla yerel verilerini kullanarak stokastik gradyan inişi (SGD) güncellemeleri yapar. Kablosuz kanallar üzerinden FL ve bunun pratik uygulamaları üzerine kapsamlı araştırmalar yapılmış olsa da, yüksek derecede heterojen sistemler için özel olarak geliştirilmiş çözümler geliştirme ve bu sistemleri inceleme ihtiyacı hâlâ açık bir araştırma alanı olarak durmaktadır. Bu tez, bu tür heterojen ortamlarda kablosuz kanallar üzerinden gerçekleştirilen havadan beslemeli (OTA) FL için stratejiler geliştirmeyi ve çözüm önerileri sunmayı amaçlamaktadır.

Tezin ilk bölümünde, enerji hasatı yapan mobil cihazlarla gerçekleştirilen OTA FL için kullanıcı seçimi stratejilerine odaklanılmıştır. Veri dağılımı özelliklerine ve buna bağlı olarak kullanıcıların ilettikleri güncellemelere dayalı bir kullanıcı seçimi paradigması izlenmiştir. Buradaki temel amaç, öğrenme performansını artırmak ve sınırlı şekilde hasat edilen enerjinin daha verimli kullanılmasını sağlamaktır. Kullanıcıların veri dağımlarının sunucu tarafından bilinip bilinmemesine bağlı olarak iki farklı kullanıcı seçimi yaklaşımı geliştirilmiştir. Önerilen OTA FL yapısının yakınsama analizi teorik olarak sunulmuş ve deneysel sonuçlarla desteklenmiştir. Elde edilen sonuçlar, kullanıcı özelliklerine dayalı seçim stratejilerinin global öğrenme performansını anlamlı ölçüde artırabileceğini

ve sistemdeki bilgi tekrarını azaltabileceğini göstermektedir.

Tezin ikinci bölümünde, enerji hasadı yapan mobil cihazlar için özelleştirilmiş global modelleri hedefleyen, OTA FL yapısında kümelenmiş federe öğrenme (CFL) yaklaşımları incelenmiştir. Bu bölümde, kullanıcıların doğal bir şekilde çeşitli kümelere ayrıldığını varsayıyoruz; örneğin, her bir küme belirli spor alanlarına ilgi duyan bireylerden oluşabilir. Kullanıcılar arasındaki yüksek heterojenlik ve enerji hasadı nedeniyle ortaya çıkan stokastik uygunluktan kaynaklanabilecek suboptimal model performansını önlemek amacıyla, farklı kümelere özel daha spesifik global modelleri aynı anda havadan iletim ile sunmayı hedefliyoruz. Sunucu tarafında, mevcut kanal durumu bilgisi (CSI) seviyesine bağlı olarak özel olarak tasarlanmış farklı birleştirme yöntemleri öneriyoruz. Sayısal sonuçlar, yeterli sayıda alıcı anteniyle birden fazla kümeden eşzamanlı güncelleme iletiminin desteklenebileceğini ve daha özelleştirilmiş, yüksek performanslı global modellerin elde edilebileceğini göstermektedir.

Anahtar sözcükler: Federe öğrenme, kablosuz iletişim, havadan iletim, enerji hasatı yapan cihazlar, sönme kanalları, kümelenmiş federe öğrenme, heterojen veri dağılımı.

Acknowledgement

I want to express my sincere gratitude to my advisor, Prof. Tolga M. Duman, for his insightful guidance, unwavering support, and unfailing patience over the past few years. His genuine care and interest have gone far beyond the duties of his position.

I would like to thank Prof. Orhan Arıkan and Prof. Javad Haghighat for their valuable time and thoughtful feedback.

I would like to express my heartfelt gratitude to my family, who not only helped shape me into who I am but also remind me every day of a simple truth: happiness is only real when shared.

Thank you to the love of my life, my newly wedded wife Rabia, for supporting me every day, both academically and socially.

I was fortunate to work with Dr. Büşra Tegin and Dr. Mohammad Kazemi, and I thank them both for their invaluable guidance and our brainstorming sessions. I extend my sincere appreciation to all the Bilkent Communication Theory and Application Research (CTAR) Lab members: Uras Kargı, Muhammad Atif Ali, Mert Özateş, Mohammad Javad Ahmadi, Ruslan Morozov, Reza Asvadi, Yajuan Liu, Ömer Faruk Keskin, Mustafa İlyas Çalışkan, Samet Senai Işık.

I feel lucky to be supported by Türk Telekomünikasyon A.S. within the framework of the 5G and Beyond Joint Graduate Support Programme coordinated by the Information and Communication Technologies Authority. This thesis is also partially funded by TUBITAK through CHIST-ERA project SONATA (CHIST-ERA-20-SICT-004, funded by TUBITAK, Turkey Grant 221N366).

Contents

List of Figures

List of Acronyms

AI	Artificial intelligence
AWGN	Additive white gaussian noise
CNN	Convolutional neural network
CSI	Channel state information
CSIT	CSI at the transmitter
CFL	Clustered federated learning
EH	Energy harvesting
FedAvg	Federated Averaging
FL	Federated learning
FMTL	Federated multitask learning
i.i.d.	Independent and identically distributed
LSE	Least-squares estimation
MAC	Multiple access channel
MSE	Mean squared error
ML	Machine learning
MMSE	Minimum mean square error
MD	Mobile device
MU	Mobile user
non-i.i.d.	Non-independent and identically distributed
OTA	Over-the-air
PS	Parameter server
SGD	Stochastic gradient descent
SNR	Signal-to-noise ratio

Chapter 1

Introduction

1.1 Overview

Machine learning (ML) is becoming increasingly integrated into our daily lives. As interest grows in areas like image classification, object detection, language models, and generative artificial intelligence (AI), ML continues to make everyday tasks more convenient and deeply embedded in our routines. With the growing number of devices capable of high-quality observations, vast amounts of data are available for ML applications. Traditional ML approaches require data to be collected and stored at a centralized location, which demands significant communication and computational resources. Moreover, sharing such data raises privacy concerns, as the information gathered by mobile devices (MDs) is often sensitive and personal. To address these limitations, a decentralized learning paradigm known as *Federated Learning* (FL) has been proposed [1], allowing multiple devices to collaboratively train a global model without exposing their local data. This approach is primarily motivated by the goal of achieving high model performance while preserving data locality and minimizing server-side data access.

FL allows multiple mobile devices to collaboratively train a global model with coordination from a parameter server (PS), offering benefits such as data privacy,

low latency, and improved model quality [1, 2]. During each round, the PS sends the current model to a subset of clients. These clients then use their local data to perform stochastic gradient descent (SGD) and return the local model updates to the server. The PS aggregates these updates to form a new global model. This iterative process continues until a specified criterion, e.g., convergence or a predetermined number of iterations, is satisfied. A key advantage of this approach lies in decoupling model training from direct access to raw data. This enables local data to stay on mobile devices without being transmitted to a central server, thereby ensuring a more secure learning setup. Nevertheless, although computational costs are reduced, model updates must still be transmitted from the mobile devices to the server.

As FL continues to scale across massive numbers of mobile devices, the need for communication-efficient and privacy-preserving aggregation strategies becomes increasingly pressing. In the case of wireless devices, FL relies on communication over potentially unreliable and resource-limited networks. This is particularly challenging in the uplink direction, as numerous devices with limited power and bandwidth must transmit their updates over a shared wireless channel to the parameter server. Hence, designing communication-efficient protocols is essential for practical deployment of federated learning solutions in wireless setups. Numerous studies have investigated key challenges associated with implementing FL over wireless networks [3, 4, 5]. Some works focus on reducing uplink communication costs through different techniques, while others address joint power and resource allocation to improve overall system efficiency.

Over-the-air (OTA) aggregation has gained popularity due to its efficient strategy of assigning all mobile users (MUs) to the same frequency band, allowing them to transmit their local updates simultaneously over a multiple access channel (MAC). Thanks to the superposition property of wireless MAC, these updates are naturally aggregated in the air, enabling simultaneous transmission and aggregation of gradient updates directly over the wireless medium. This method leverages the fact that, in federated learning, the server ultimately requires only the aggregated model update rather than individual updates. While interference in the shared wireless channel complicates the decoding of individual messages

compared to orthogonal transmission, OTA FL [6, 7] turns this challenge into an advantage by harnessing the superposition property of the channel. As a result, the aggregation is performed implicitly during transmission, allowing the server to directly recover the averaged global model. There are methods that build on this framework by aligning signals across multiple receiver antennas, enabling accurate recovery and combination of updates, thus allowing the global learning process to continue with minimal distortion.

Despite the promising advances in OTA FL, several key challenges remain overlooked in the existing literature. Most prior works assume either ideal transmission conditions or static user availability, neglecting the practical constraints posed by energy harvesting devices and highly dynamic participation. Additionally, while some studies consider user selection based on channel conditions or update magnitudes, they often ignore the statistical heterogeneity among users, which may severely degrade the learning performance. There are no scheduling strategies that simultaneously account for users' data distribution characteristics and their stochastic energy availability. Furthermore, few studies investigate personalized model delivery in the OTA setting, especially under the joint constraints of non-independent and identically distributed (non-i. i.d.) data and energy limitations. These challenges highlight the need for a more holistic approach, incorporating user scheduling under energy constraints, as developed in Chapter 3, and clustered model aggregation, as presented in Chapter 4, both tailored to the heterogeneous and resource-constrained nature of real-world federated learning systems.

1.2 Contributions of the Thesis

In this thesis, we aim to explore strategies and solutions to improve the performance of over-the-air federated learning over wireless channels in highly heterogeneous setups. By leveraging scheduling strategies and signal combining techniques, we analyze and demonstrate their effectiveness through both convergence analysis and numerical experiments to improve global model performance in OTA

FL systems. Regarding heterogeneity, we consider both non-uniform data distributions that negatively impact learning performance and convergence, and highly dynamic user availability due to the energy harvesting characteristics of mobile devices.

In the first part of this thesis, we focus on scheduling strategies for OTA FL with energy harvesting mobile devices and with highly heterogeneous data distribution. In this setup, energy-harvesting mobile devices are capable of collecting the energy needed for local computation and communication tasks, and they are equipped with a unit-sized battery to store harvested energy for future iterations. In this part of the work, we propose scheduling strategies based on users' data characteristics to make more efficient use of the limited harvested energy by reducing redundant transmissions. For scheduling purposes, we leverage the relationships among users due to their data distribution. Initially, we assume that the user data distributions are known at the server side and demonstrate that incorporating this information along with the battery levels of the users into the scheduling strategy can enhance global learning performance and reduce redundancy in the system. We then demonstrate that, even without explicit knowledge of the user data distributions, the PS can infer data characteristics from the global updates and obtain their estimates. These inferred patterns can subsequently be used to guide effective user scheduling and enhance overall learning efficiency.

Our contribution in the first part of this thesis is reported in the following paper:

- F. Bagci, B. Tegin, M. Kazemi, and T. M. Duman, "Update estimation and scheduling for over-the-air federated learning with energy harvesting devices," in *2025 IEEE International Conference on Communications Workshops (ICC Workshops)*, Montreal, Canada, June 2025. [8]

We then shift our focus to more personalized global models, particularly clustered federated learning (CFL) structures, aiming to enable federated multitask learning that provides improved models for users with similar data characteristics. It has been observed that highly different local data distributions often lead

to suboptimal global model performance. To address this challenge in a highly heterogeneous network setting, we aim to deliver more tailored models to groups of users with similar data characteristics, thereby maintaining high performance across the network. Building on the user estimates obtained in the first part, we group similar users into clusters and provide each cluster with its own group-level global model instead of obtaining a single global model to serve all the clusters. In this way, personalized cluster models can achieve higher performance when the data observed by users aligns with the characteristics of their respective clusters, compared to training a single global model under a uniform data distribution assumption. At the same time, the users can still benefit from joint inter-cluster training. We demonstrate that, with a sufficient number of receiver antennas, it is possible to support simultaneous update transmissions from all the clusters and users by employing the proposed combining techniques.

1.3 Thesis Outline

The thesis is organized as follows. Chapter 2 presents a comprehensive literature review on federated learning, covering its motivations and general system design. It also discusses the extension of FL to wireless communication systems, as well as the clustered federated learning approach, including its motivations and practical implementations.

Chapter 3 presents the proposed scheduling algorithms for OTA FL with energy harvesting (EH) mobile devices under highly heterogeneous data distributions. Two user scheduling approaches are proposed: one leveraging known user data distributions at the PS and the other relying on estimated user characteristics to achieve more balanced participation, improved performance, and reduced redundancy, all without revealing data distributions to the PS and thus preserving user privacy. The chapter also demonstrates that user updates carry information about data characteristics, which can be leveraged to cluster users based on data similarity to be utilized in the scheduling process. We provide a convergence analysis, which is also utilized to optimize the user scheduling strategy, and present

numerical results with different datasets and data distributions.

In Chapter 4, we propose combining strategies that allow users from different clusters to transmit their updates simultaneously over a shared communication medium using over-the-air aggregation. This approach facilitates group-level personalized models, allowing the system to serve different clusters more effectively for improved learning performance. We demonstrate that, by leveraging multiple antennas at the server and applying the proposed combining strategies, clustered federated learning can be effectively extended to wireless fading environments, enabling efficient and accurate model aggregation across diverse user groups with OTA aggregation.

Finally, Chapter 5 concludes the thesis and discusses future research directions for over-the-air FL in heterogeneous settings, aiming to enable shared models that are privacy-preserving, efficient in both communication and computation, unbiased, stable, and supported by convergence guarantees.

Chapter 2

Preliminaries and Literature Review

This chapter provides general preliminaries and a literature review of both foundational and recent works relevant to the development and presentation of this thesis. First, federated learning (FL) is introduced as a general framework, with key preliminaries outlined, followed by a review of the current literature. Second, the wireless aspect of the problem is introduced, and the setup for wireless federated learning is described. This is followed by a discussion of the clustered FL framework, which provides background for Chapter 4. Lastly, energy-harvesting devices and related work within the FL context are presented to support the discussions in the subsequent chapters.

The chapter is organized as follows. Section 2.1 discusses traditional FL schemes, including the general setup and a review of both foundational and recent works. Section 2.2 introduces wireless FL and over-the-air (OTA) FL. Section 2.3 presents preliminaries related to energy-harvesting devices. Finally, the chapter is concluded in Section 2.4.

2.1 Federated Learning

Federated learning is a machine learning paradigm in which multiple clients collaboratively train a shared model under the coordination of a central server, while keeping their training data decentralized and local to their devices. The term *federated learning* was originally introduced by the authors in [1] to describe a learning framework carried out by a loose federation of participating devices. Throughout this thesis, we refer to these participating devices as mobile devices or users. Since its initial introduction, the concept has gained significant interest from both the research community and applied sciences, owing to its importance and the new direction it offers for researchers and engineers. Over the years, many researchers have addressed various challenges introduced by federated learning. While significant progress has been made, there are still open problems to solve and systems that require further improvement.

FL offers several key advantages, including data privacy, low latency, and improved model quality [2, 9]. Since raw data remains on local devices and is not transmitted to a server during training, FL inherently provides a degree of data privacy. Hence, FL enables the training of globally optimized models that benefit from diverse data distributed across users, something that is often difficult to achieve in traditional centralized systems due to data ownership, bandwidth, and privacy constraints. Compared to centralized learning frameworks, FL also helps reduce communication latency by avoiding the need to offload large datasets to a central server.

FL exhibits several key characteristics that distinguish it from traditional distributed optimization problems [1]. The first key property is that the training data on each client is typically non-independent and identically distributed (non-i.i.d.), as it reflects the individual usage patterns of a specific mobile device. As a result, a user's local dataset is unlikely to be representative of the overall population distribution. Another key challenge is data imbalance, as users generate varying amounts of data depending on their usage patterns. Furthermore, FL is typically conducted in a massively distributed environment, where

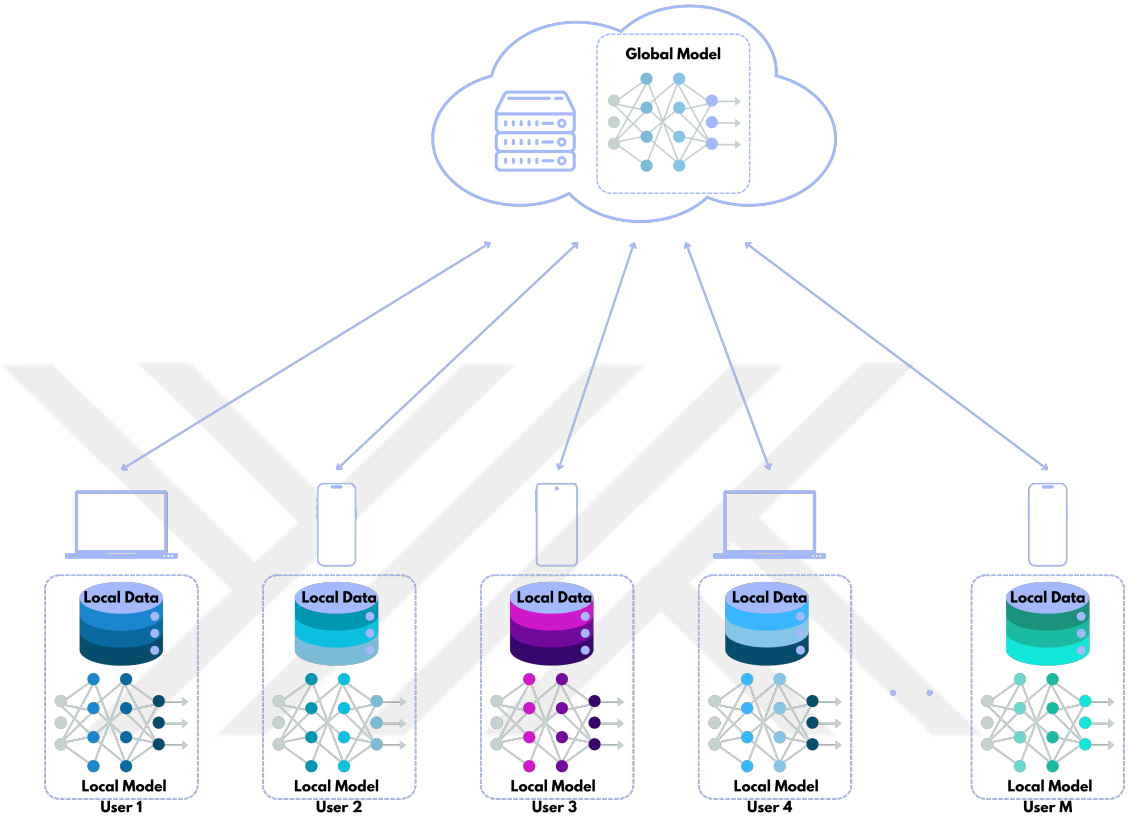


Figure 2.1: FL system model.

the number of participating clients is relatively large. Limited communication is another challenge, as mobile devices are often offline or connected via slow or costly networks, making reliable and efficient data exchange difficult. Federated Averaging (FedAvg) [1] was first introduced to address the challenges of training on decentralized data from mobile devices, highlighting the core issues of FL and proposing practical solutions for efficient model training in such settings.

In FL [1], the primary objective is to minimize a global loss function, denoted as $F(\boldsymbol{\theta})$, where $\boldsymbol{\theta} \in \mathbb{R}^{2N}$ represents the parameters of the global model to be optimized, and $2N$ denotes the total number of real-valued model parameters. The system consists of M mobile users and a parameter server as given in Figure 2.1. The global loss function is defined as

$$F(\boldsymbol{\theta}) = \sum_{m=1}^{M} \frac{|\mathcal{B}_m|}{B} F_m(\boldsymbol{\theta}), \tag{2.1}$$

where $F_m(\boldsymbol{\theta})$ represents the average empirical local loss for the m-th user with model parameters $\boldsymbol{\theta}$. For the m-th user with the local dataset $\mathcal{B}_m$, for $m \in [M]$, defining $[M] = \{1, \ldots, M\}$, and $B \triangleq \sum_{m=1}^{M} |\mathcal{B}_m|$, we have

$$F_m(\boldsymbol{\theta}) = \frac{1}{|\mathcal{B}_m|} \sum_{\boldsymbol{u} \in \mathcal{B}_m} f(\boldsymbol{\theta}, \boldsymbol{u}), \tag{2.2}$$

with $f(\boldsymbol{\theta}, u)$ denoting the empirical loss function corresponding to the u-th data sample in the local dataset $\mathcal{B}_m$.

At each global iteration t, the parameter server (PS) broadcasts the current global model, denoted by $\boldsymbol{\theta}(t)$, to a selected subset of participating mobile devices (MDs). Upon receiving the model, each selected MD performs τ local steps of stochastic gradient descent (SGD) to minimize its individual local loss function, $F_m(\boldsymbol{\theta})$. This approach reduces the frequency of communication with the server by allowing multiple local updates before synchronization.

To compute the local model updates, device m at the i-th local and t-th global iteration performs the local iterations of SGD with the following update rule:

$$\boldsymbol{\theta}_m^{i+1}(t) = \boldsymbol{\theta}_m^i(t) - \eta_m^i(t) \nabla F_m(\boldsymbol{\theta}_m^i(t), \xi_m^i(t)), \tag{2.3}$$

where $\boldsymbol{\theta}_m^1(t) = \boldsymbol{\theta}_m(t)$ is initialized with the global model, $i \in [\tau]$, $\eta_m^i(t)$ is the learning rate and $\nabla F_m\left(\boldsymbol{\theta}_m^i(t), \xi_m^i(t)\right)$ represents the stochastic gradient estimate for the $\boldsymbol{\theta}_m^i(t)$ and the local mini-batch sample ξ_m^i randomly chosen from the local dataset $\mathcal{B}_m$. The stochastic gradient estimate provides an unbiased estimate of the actual gradient $\nabla F_m(\boldsymbol{\theta}_m^i(t)$, and is given as

$$\mathbb{E}_\xi \left[\nabla F_m\left(\boldsymbol{\theta}_m^i(t), \xi_m^i(t)\ \right)\right] = \nabla F_m\left(\boldsymbol{\theta}_m^i(t)\right). \tag{2.4}$$

After completing the local SGD steps, device m computes the model update, which aimed to be shared with the PS as

$$\Delta\boldsymbol{\theta}_m(t) = \boldsymbol{\theta}_m^\tau(t) - \boldsymbol{\theta}_m^1(t). \tag{2.5}$$

In an optimal setting, where the PS successfully receives model updates from all devices without transmission errors, it accurately aggregates these updates

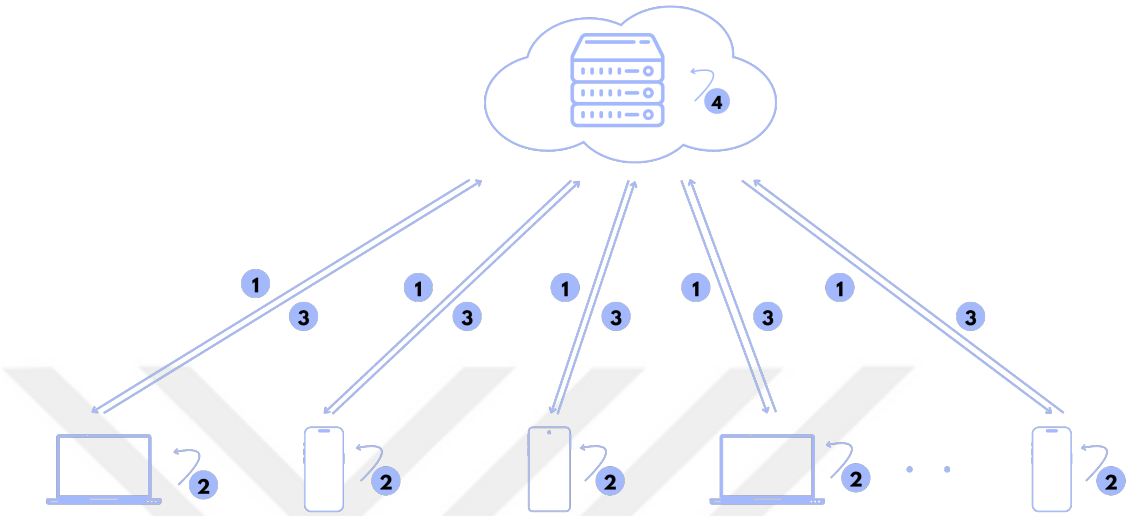


Figure 2.2: A typical FL training process.

to capture the collective learning of the user devices. The PS then updates the global model by applying the aggregate of these updates as

$$\boldsymbol{\theta}_{PS}(t+1) = \boldsymbol{\theta}_{PS}(t) + \frac{|\mathcal{B}_m|}{B} \sum_{m=1}^{M} \Delta\boldsymbol{\theta}_m(t), \tag{2.6}$$

where $\boldsymbol{\theta}_{PS}(t)$ represents the global model vector at the server at the global iteration t.

In a typical FL [1] training process (see Figure 2.2), the PS coordinates training by iteratively performing the following steps until a convergence criterion is met. First, a subset of available clients is selected, and the current global model is broadcast to them. Next, each selected mobile device performs several steps of local SGD steps using its local data to compute the local model updates. These local updates are then sent back to the PS. Finally, the server aggregates the received updates—typically by averaging—to produce an improved global model for the next round.

2.1.1 Literature Review on Federated Learning

FedAvg [1] is the first and the most popular FL algorithm that is used to aggregate local model updates to form a global model at the end of each communication cycle. In FedAvg, users run their individual local SGDs during the iteration, and at the end of each iteration, the server updates the shared global model as follows:

$$\begin{aligned} \boldsymbol{\theta}_{PS}(t+1) &= \boldsymbol{\theta}_{PS}(t) + \sum_{m=1}^{M} \frac{|\mathcal{B}_m|}{B} \Delta\boldsymbol{\theta}_m(t) \\ &= \boldsymbol{\theta}_{PS}(t) - \sum_{m=1}^{M} \frac{|\mathcal{B}_m|}{B} \eta(t) \sum_{i=1}^{\tau} \nabla F_m\big(\boldsymbol{\theta}_m^i(t),\, \xi_m^i(t)\big)\,. \end{aligned} \tag{2.7}$$

On the left-hand side, the expression $\boldsymbol{\theta}_{PS}(t+1)$ denotes the global model after the update applied to the global model between two consecutive rounds. This update is computed by aggregating the local updates $\Delta\boldsymbol{\theta}_m(t)$ received from each device m, weighted by the relative size of each device's local dataset, i.e., $|\mathcal{B}_m|/B$, where $B = \sum_{m=1}^{M} |\mathcal{B}_m|$ is the total number of training samples across all devices. Each local update $\Delta\boldsymbol{\theta}_m(t)$ results from performing τ steps of SGD on device m. This is reflected in the rightmost part of the equation, where $\Delta\boldsymbol{\theta}_m(t)$ is expressed as a scaled sum of local stochastic gradients. Specifically, the update becomes $-\eta(t)\sum_{i=1}^{\tau} \nabla F_m(\boldsymbol{\theta}_m^i(t), \xi_m^i(t))$, where $\eta(t)$ is the learning rate at round t, $\boldsymbol{\theta}_m^i(t)$ is the model at local iteration i, and $\xi_m^i(t)$ denotes the mini-batch sampled from the local dataset $\mathcal{B}_m$ at that iteration. This formulation highlights how the global model integrates the learning progress from all clients through weighted averaging of local updates, each computed over τ local steps using local data. As given in (2.1), the goal is to minimize the global loss function $F(\boldsymbol{\theta}) = \sum_{m=1}^{M} \frac{|\mathcal{B}_m|}{B} F_m(\boldsymbol{\theta})$, where each user aims to minimize their own local loss function $F_m(\boldsymbol{\theta})$.

Despite the strong empirical performance of FedAvg in independent and identically distributed (i.i.d.) settings, its effectiveness significantly degrades in non-i.i.d. scenarios [10, 11]. Hence, several modifications of FedAvg have been proposed to address the challenges posed by non-i.i.d. data distributions.

To address inconsistencies caused by non-i.i.d. data and heterogeneous local

updates, [12] adds a proximal term $\frac{\mu}{2}\|\boldsymbol{\theta}_m - \boldsymbol{\theta}_{PS}(t)\|^2$ to each local objective function, promoting local models to remain close to the global model during training. To account for statistical heterogeneity across users, each user m minimizes a modified local objective of the form $F_m(\boldsymbol{\theta}) + \frac{\mu}{2}\|\boldsymbol{\theta} - \boldsymbol{\theta}_{PS}(t)\|^2$ instead of the original $F_m(\boldsymbol{\theta})$, where the additional proximal term regularizes the local model to stay close to the global model. The proximal term helps address the issue of statistical heterogeneity by constraining local updates to remain closer to the initial global model. This is achieved through the regularizer, which penalizes updates that deviate significantly from the global model. Empirical results show that FedProx outperforms FedAvg, especially in scenarios with higher system heterogeneity.

The local objectives across users are generally not identical and may not share a common minimizer. As a result, local updates computed from the same initial global model can drift toward the minima of each user's individual objective. This phenomenon, known as *client drift*, leads to misaligned updates during aggregation, ultimately causing slow and unstable convergence of the global model [13]. In [13], the stochastic controlled averaging for FL (SCAFFOLD) algorithm is introduced that uses control variables to reduce the variance across clients and speed up the convergence. This algorithm uses control variates to correct client updates based on local gradient estimates. Specifically, each client i maintains an estimate of its local gradient, denoted by $\boldsymbol{c}_m$, while the server keeps the average of all client control variates as $\boldsymbol{c}$. During training, each client performs local SGD steps using a corrected gradient of the form $\nabla F_m(\boldsymbol{\theta}) + \boldsymbol{c} - \boldsymbol{c}_m$, which approximates the true global gradient $\nabla F(\boldsymbol{\theta_{PS}})$. This correction mechanism helps mitigate client drift and improves convergence stability in non-i.i.d. settings.

The FedAvg algorithm can be extended into a more flexible framework that enables algorithm designers to modify both client-side and server-side update strategies. Adaptive federated optimization (FedOpt) [14] represents a family of optimization-based methods that enhance FedAvg by introducing adaptive optimization techniques. It is defined by two gradient-based optimizers: *ClientOpt* and *ServerOpt*, which operate with learning rates η and η_s for the client and server, respectively. *ClientOpt* focuses on minimizing local objectives, as defined in (2.1), using data available on each client. In contrast, *ServerOpt* updates the

global model using a broader optimization strategy from the server's perspective. FedOpt allows the use of advanced optimizers and supports enhancements like server-side momentum [14, 15], which help improve convergence speed and stability, especially under non-i.i.d. conditions.

In the FL setup, the choice of weighting scheme for aggregating local updates plays a critical role. If not selected properly, the global model may converge to a stationary point of a surrogate objective that is inconsistent with the original global objective. As shown in [11], the aggregation weights should align with the client sampling strategy. Specifically, if clients are sampled with replacement based on their local dataset sizes, then the local updates should be uniformly averaged. On the other hand, if clients are sampled uniformly without replacement, their updates should be re-weighted according to the proportion of their local sample sizes to ensure correct convergence behavior. The key idea is to ensure that the expected aggregated local updates weigh each client in a manner consistent with the global objective [15]. If this condition is not met, a non-vanishing term appears in the upper bound of the optimization error [16]. However, in real-world applications, the server typically has limited knowledge of the overall population distribution and restricted control over client sampling. In [17], it is stated that naively applying weighted aggregation when clients perform different numbers of local updates can lead to objective inconsistency. Specifically, the global model may converge to a stationary point of a mismatched objective function, which can differ significantly from the true global objective. Heterogeneity in clients' local datasets and computational capacities leads to variations in the number of local updates performed across clients. Hence, standard averaging of client models after heterogeneous local updates can lead to convergence toward a stationary point of a biased or inconsistent objective. The authors in [17] introduce federated normalized averaging (FedNova), a simple approach designed to resolve the inconsistency between the surrogate objective and the true global objective. The method works by normalizing each client's local update, dividing it by the number of local steps taken, so that the aggregated update better reflects the global objective. This normalization acts as an alternative weighting mechanism, effectively reducing the influence of clients that perform more local updates and

promoting fairer aggregation in heterogeneous settings. The convergence and consistency of FL algorithms in practical, heterogeneous environments remain active areas of research, with ongoing efforts focused on improving stability, fairness, and alignment with the true global objective.

Studies on FL with non-i.i.d. data show that heterogeneous local datasets significantly impact model accuracy and convergence [10, 11]. In FL, data across clients often deviates from being identically distributed, which introduces significant challenges to model training and convergence. One common source of heterogeneity is feature distribution skew (covariate shift), where clients have different input feature distributions even if their label distributions are similar. Another is label distribution skew (prior probability shift), in which the prevalence of certain labels varies across clients. Concept drift refers to scenarios where the same label is associated with different features across clients, while concept shift describes the opposite, where the same features may correspond to different labels depending on the client. Additionally, quantity skew or data unbalancedness arises when clients possess varying amounts of training data, causing unequal influence during model aggregation. Real-world FL datasets often exhibit a combination of these non-i.i.d. characteristics [18]. However, most empirical studies on synthetic non-i.i.d. datasets have primarily focused on label distribution skew, overlooking other important forms of heterogeneity.

In [19], it is shown that among the various forms of non-i.i.d. data, label distribution skew, particularly the extreme case where each client holds data from only a single class, poses the greatest challenge for FL. In contrast, feature distribution skew and quantity imbalance tend to have a relatively minor impact on the performance of algorithms like FedAvg. While many advanced algorithms have been proposed to address these challenges, none consistently outperforms the others across all scenarios. In fact, most state-of-the-art methods offer substantial improvements over FedAvg only in select cases, highlighting the need for further exploration in diverse non-i.i.d. settings [19].

Another persistent challenge in FL is the lack of control over users' devices, which may become unreachable or unresponsive during training. Such devices

are referred to as stragglers, clients that respond significantly slower than others or fail to respond at all. Since the central server cannot control client availability, relying on responses from all devices in every round is impractical. Therefore, assuming full participation at all times is unrealistic in real-world FL deployments. The authors in [11] show that data heterogeneity and partial device participation significantly slow down the convergence of federated learning algorithms. They also empirically validate their theoretical findings through numerical experiments. In practice, only a small subset of user devices are available to participate in each training round, or the server may be limited to serving only a portion of users. Both situations can exacerbate the effects of data heterogeneity, further challenging model convergence. To address the issues arising from partial user participation, the problem of client scheduling, also referred to in the literature as selection or sampling, has been widely studied in the context of FL. Most existing works assume uniform client participation, which ensures an unbiased estimate of the global model update, even under partial participation. This assumption allows convergence results derived for the full participation setting to be extended to partial participation scenarios [11, 17]. In [16], it is also shown that biasing client selection toward those with higher local loss can lead to faster error convergence.

Despite offering clear advantages in security and privacy compared to traditional centralized training, FL still faces significant challenges related to privacy and system security. The decentralized nature of FL makes it particularly challenging to detect and defend against adversarial attacks. As shown in [20], certain techniques can be employed to mitigate these threats, with one notable example being the use of differential privacy [21, 22], which helps protect sensitive information in local updates by adding controlled noise during training. Backdoor attacks in FL aim to cause the model to behave abnormally when presented with inputs containing a specific trigger, while maintaining normal performance on benign data. These attacks can be carried out through data poisoning [23], where the attacker injects malicious samples into the training data, or through model poisoning [24], where the attacker manipulates model updates to embed the backdoor without altering the training data directly. There are a few general

defense strategies commonly used in federated learning: anomaly update detection, which targets the identification of malicious client updates; robust federated training, which minimizes the influence of adversarial updates during aggregation; and backdoored model restoration, which focuses on cleaning the global model after it has been compromised [25].

Another important aspect of privacy concerns in FL is that user updates inherently reflect user-specific characteristics. As shown in [26], these updates can unintentionally leak sensitive information about participants' training data. An adversarial client can exploit this to infer properties that apply only to a subset of the data, or even identify the presence of specific data points within user datasets. In [27], the authors leverage the implicit relationship between a device's local data distribution and the corresponding model updates, enabling them to profile the data distribution on individual devices based on the uploaded model weights. They propose an intelligent client selection strategy that carefully chooses which devices participate in each training round. This approach helps counterbalance the bias introduced by non-i.i.d. data and accelerates convergence, as randomly selected local datasets may not accurately represent the global data distribution. Specifically, they demonstrate that it is possible to profile and cluster each device by analyzing its local model weights and propose a deep reinforcement learning-based client selection framework. This framework leverages the implicit connection between local data distributions and model updates to mitigate the bias introduced by non-i.i.d. data and reduce overall communication costs. It does so by capturing not only the relationship between updates and individual users but also the inter-user relationships within the system.

Several studies have explored leveraging the relationships between users to improve model performance and accelerate convergence in FL. In [28], the authors introduced a client sampling strategy called *clustered sampling*, which aims to improve MD selection by increasing the likelihood of selecting clients with distinct data distributions, thereby promoting more representative updates and enabling smoother and faster FL convergence. The method uses cosine similarity to group clients based on the similarity of their updates and then samples a diverse set of users from these clusters to participate in each training round. In [29], the

authors introduced federated averaging with diverse client selection (DivFL), a method that selects a small but diverse subset of clients for participation in each training round. Rather than collecting updates from all users, only the selected clients transmit their updates to the server, thereby reducing communication overhead while preserving model quality. By emphasizing diversity, DivFL seeks to approximate the effect of full client participation and minimize redundancy arising from similar data distributions and, consequently, similar local updates. To achieve this, a greedy selection algorithm is used to choose clients whose combined updates closely approximate the true global gradient. The aforementioned studies are based on the premise that users with similar data distributions tend to produce similar model updates, enabling the use of these updates to infer characteristics of the underlying local data. While this similarity can potentially lead to privacy leakage, it can also be leveraged to improve model performance and accelerate convergence in FL.

Several studies [30, 31] utilize direct information about client data distributions to improve the effectiveness of FL. For example, [30] selects clients based on their data distribution to ensure that all major distribution types are represented in each training round. Similarly, [31] proposes clustering clients using similarity metrics derived from their data, and then selecting one representative client from each cluster to participate in each round. This approaches promote diversity among selected clients and leads to improved training efficiency and convergence.

2.2 Wireless Federated Learning

With the growing scale of FL across vast numbers of mobile devices, the demand for aggregation strategies that are both communication-efficient and privacy-preserving is becoming more urgent. In wireless environments, FL relies on data transmission over networks that are often unreliable and limited in resources. This is especially challenging during uplink communication, where numerous devices with bandwidth and power constraints must send updates to the parameter

server over a shared wireless channel. As a result, developing efficient communication protocols is crucial for enabling the practical deployment of FL in wireless settings.

In the FL framework, devices collaboratively train a shared model under the coordination of a PS. In real-world implementations, one of the primary bottlenecks is the limited bandwidth of the communication channel from devices to the PS. As a result, reducing the communication overhead in collaborative machine learning is essential for the efficient and scalable deployment of FL. To address communication bottlenecks in FL, there are few approaches have been widely explored in the literature: quantization, sparsification, and local updates, as well as their various combinations. Quantization methods reduce communication costs by applying lossy compression to gradient vectors, quantizing each entry to a low-precision, finite-bit representation. In [32], the authors propose Quantized SGD (QSGD), a family of compression schemes that offer both convergence guarantees and strong empirical performance. QSGD enables users to flexibly trade off between communication bandwidth and convergence speed, making it a practical choice in resource-constrained settings.

Sparsification is another effective approach to reduce communication overhead in FL. It works by transmitting only a subset of the gradient vector, typically the most significant elements, while ignoring the rest. This method can be viewed as a form of lossy compression; however, it is generally assumed that the selected gradient entries are transmitted reliably, often with high precision. By communicating only the most important components of the gradient, Gradient Sparsification (GS) significantly reduces the communication burden and improves the overall efficiency of FL, without severely impacting model convergence [33, 34].

2.2.1 Literature Review on Wireless Federated Learning

Various device scheduling techniques have been introduced for wireless federated learning to select a subset of devices that share limited wireless resources in each communication round. In [35], the authors propose a scheduling policy based on

a metric called Age of Update (AoU), which jointly considers the staleness of the received model parameters and the instantaneous channel quality of each device. This approach aims to enhance the overall efficiency and convergence speed of FL by prioritizing timely and reliable updates. In [3], the authors formulate a joint bandwidth allocation and device scheduling problem aimed at optimizing the long-term convergence performance of federated learning. This problem is decomposed into two sub-problems. For bandwidth allocation, the optimal solution suggests assigning more bandwidth to devices with poor channel conditions or limited computational capacity, thereby balancing update contributions. For device scheduling, the study reveals a trade-off between the number of training rounds needed to reach a desired accuracy and the latency incurred per round. This leads to a greedy scheduling policy, which iteratively selects the device that minimizes update time, striving to balance learning efficiency and communication latency. In [4], the authors develop a formal analytical framework to evaluate the performance of FL over wireless systems. Their analysis yields a tractable expression for the convergence rate, explicitly accounting for critical characteristics of wireless communication environments. These include the transmission scheduling policy, small-scale fading, large-scale path loss, and inter-cell interference, providing valuable insights into the impact of wireless factors on FL training efficiency. In [36], the authors utilize both channel conditions and the significance of local model updates to guide device scheduling decisions. The significance of a local update is quantified using its l_2-norm, reflecting its contribution to the global model. Based on this, channel resources are allocated to devices that not only have high-impact updates but also possess sufficient link quality to reliably transmit their information, ensuring efficient use of limited communication resources.

Another line of work focuses on the resource allocation problem to optimize the performance of FL systems. This involves jointly addressing learning, wireless resource allocation, and user selection by formulating an optimization problem aimed at minimizing an FL loss function that reflects the algorithm's performance. To tackle this, the authors [5] first derive a closed-form expression for the expected convergence rate of the FL algorithm, establishing a direct relationship between packet error rates and FL performance. First, the optimal transmit

power is determined for a given user selection and resource block (RB) allocation; then, the problem is transformed into a bipartite matching problem, which is solved to obtain the optimal FL-aware user selection and RB allocation strategy. In [37], it is shown that a device's contribution to reducing the global loss function is theoretically bounded by the inner product between its local gradient and the global gradient. Leveraging this insight, the authors introduce a near-optimal device selection strategy that samples devices based on the magnitude of these inner products. Building on this foundation, they propose a fast-converging federated learning (FOLB) algorithm that performs intelligent device sampling in each training round to optimize the expected convergence speed. In [38], the authors address the challenge posed by the limited number of resource blocks in wireless networks, which restricts the number of users that can transmit their local model updates in each round. To overcome this, they propose a probabilistic user selection scheme, where users whose local models have a greater impact on the global model are assigned higher probabilities of being selected to connect to the server. The motivation behind this approach is that the global model may lack sufficient information from certain users' local datasets. By increasing the selection probability for these underrepresented users, the global model can better capture their data characteristics, thereby accelerating the convergence of the system.

2.2.2 Federated Learning with Over-the-Air Aggregation

Storing the data locally to perform training at the edge device requires sending the local updates to the server, as conventional FL has become popular. With the increasing interest, the communication bandwidth becomes one of the main bottlenecks for aggregating the updates [6]. OTA computation has been proposed and utilized for model aggregation by exploiting the superposition property of a wireless multiple access channel (MAC), enabling updates to be aggregated during transmission [6, 7].

In [7], the authors place greater emphasis on the physical and network layer

aspects of the problem, which are crucial for implementing FL at the network edge, where training occurs over a shared wireless medium. The authors first propose a Digital Distributed Stochastic Gradient Descent (D-DSGD) scheme, in which devices compress their gradient estimates into a finite number of bits using quantization. These compressed gradients are then transmitted to the parameter server using an access scheme, such as time division, frequency division, or code division multiple access, combined with error correction coding to ensure reliable communication. They also propose an alternative analog communication approach, where devices transmit their local gradient estimates directly over the wireless channel without digital encoding. This scheme is motivated by the observation that the parameter server is not concerned with individual gradients, but rather with their average. Leveraging the superposition property of the wireless MAC, this method enables the PS to naturally receive a noisy sum of the transmitted gradients, effectively performing OTA aggregation without the need for explicit coordination or decoding of individual updates.

OTA aggregation in wireless FL setups is also discussed in [6], demonstrating how the signal superposition property of a wireless multiple-access channel can be harnessed to enable fast global model aggregation in on-device distributed training. To further enhance communication efficiency and statistical performance, the authors propose a joint device selection and receiver beamforming design. This approach aims to maximize the number of participating devices while satisfying a predefined mean-squared-error (MSE) requirement, thereby accelerating the global model aggregation process under wireless constraints.

In [39], the authors demonstrate that interference and noise effects in over-the-air aggregation can be mitigated by increasing the number of receive antennas at the parameter server. Remarkably, they show that even in the absence of channel state information (CSI) at the transmitters and with imperfect CSI at the receiver, the system can still achieve strong performance, highlighting the robustness of multi-antenna receiver designs in wireless federated learning.

In this thesis, we follow a similar setup. We consider an FL system for M devices. We assume that the mobile users do not have CSI and transmit their

updates to the PS through a fading MAC using over-the-air transmission. The PS employs K receive antennas to align the received signals in the absence of CSI at the transmitters (CSIT) using aggregated channel information. The model updates of the users, are transmitted as complex signals at iteration t, represented as

$$\Delta\boldsymbol{\theta}_m^{re}(t) \triangleq \left[\Delta\theta_m^1(t), \Delta\theta_m^2(t), \ldots, \Delta\theta_m^N(t)\right]^T, \tag{2.8a}$$

$$\Delta\boldsymbol{\theta}_m^{im}(t) \triangleq \left[\Delta\theta_m^{N+1}(t), \Delta\theta_m^{N+2}(t), \ldots, \Delta\theta_m^{2N}(t)\right]^T, \tag{2.8b}$$

$$\Delta\boldsymbol{\theta}_m^{cx}(t) \triangleq \Delta\boldsymbol{\theta}_m^{re}(t) + j\Delta\boldsymbol{\theta}_m^{im}(t). \tag{2.8c}$$

where $\Delta\boldsymbol{\theta}_m^{cx}(t) \in \mathbb{C}^N$, $m \in \mathcal{S}(t)$, and $\mathcal{S}(t)$ denotes the set of users participating in round t.

The received signal at the k-th antenna of the PS at iteration t is given as

$$\boldsymbol{y}_{PS,k}(t) = \sum_{m\in M} \boldsymbol{h}_{m,k}(t) \circ \Delta\boldsymbol{\theta}_m^{cx}(t) + \boldsymbol{z}_{PS,k}(t), \tag{2.9}$$

where $\Delta\boldsymbol{\theta}_m^{cx}(t)$ is the complex signal transmitted by the m-th user, and $\boldsymbol{h}_{m,k}(t)$ is the i.i.d. channel gains from the m-th user to the k-th antenna with complex Gaussian entries $h_{m,k}^n(t) \sim \mathcal{CN}(0, \sigma_h^2)$. The symbol $\circ$ denotes element-wise product. Similarly, $z_{PS,k}^n(t)$ denotes the n-th entry of the channel noise, $\boldsymbol{z}_{PS,k}(t)$, which is i.i.d. circularly symmetric white Gaussian noise (AWGN), i.e., it is distributed according to $\mathcal{CN}(0, \sigma_z^2)$.

The PS uses the signals received from the K antennas along with appropriate combining techniques, to update the global model as

$$\boldsymbol{\theta}_{PS}(t+1) = \boldsymbol{\theta}_{PS}(t) + \Delta\hat{\boldsymbol{\theta}}_{PS}(t), \tag{2.10}$$

where $\boldsymbol{\theta}_{\mathrm{PS}}(t)$ represents the global model vector at the server at global iteration t and $\Delta\hat{\boldsymbol{\theta}}_{PS}(t)$ is the noisy estimate of the average of the local updates.

2.2.3 Literature Review on Federated Learning with Over-the-Air Aggregation

Wireless FL has gained significant attention due to the efficiency of OTA aggregation, where gradients from multiple devices can be transmitted simultaneously over the same wireless medium. Several studies have leveraged OTA aggregation and proposed solutions to address practical challenges encountered in real-world FL applications.

Few studies have addressed user selection in OTA FL settings. One such work [40] proposes an energy-aware dynamic scheduling algorithm for an OTA FL system with analog gradient aggregation. It optimizes training under device energy constraints by accounting for both communication and computation energy. Another study [41] formulates the device scheduling problem as a sparse support selection task, where the goal is to select a subset of devices under a combined cost constraint. To solve this, the authors propose a class of greedy algorithms inspired by the matching pursuit method, effectively balancing cost and update contribution in each communication round. In [42], the authors propose a dynamic device scheduling mechanism for federated learning via OTA aggregation, where selected edge devices transmit their local models using a suitable power control strategy. The scheduling decision jointly considers the data importance, measured by the local gradient magnitude, along with the channel conditions and the energy consumption of each device. Also, in [43], the authors address beamforming vector design and device selection for over-the-air computation in FL. Recognizing that model performance benefits from broader device participation, they formulate an optimization problem that seeks to maximize the number of selected devices while ensuring the aggregation meets a target MSE threshold.

Some studies on OTA FL are designed for systems with varying levels of CSI. In particular, MUs are often assumed to be blind, meaning they have no access to CSI. However, the PS is assumed to have either full or partial CSI, which it can exploit to facilitate effective aggregation despite the lack of CSI at the transmitters. Studies such as [44, 39, 45] demonstrate that even in the absence of

CSI at the transmitters, reliable model aggregation at the PS is still feasible. With a sufficient number of receive antennas and an estimate of the superposed channel (i.e., the aggregate of channel gains from all MUs), the PS can obtain a noisy yet usable estimate of the global update. This allows the global model to be updated with provable convergence guarantees, despite lacking both individual CSI at the devices and full per-user CSI at the PS. In [46, 47], a similar scenario is studied where there is no CSI at the transmitters and only partial CSI is available at the PS. The authors consider time-varying wireless channels and show that such variations do not degrade the performance of OTA FL. Moreover, the inter-carrier interference caused by channel fluctuations is shown to diminish as the number of receive antennas increases, further reinforcing the robustness of OTA FL in dynamic wireless environments.

In wireless fading environments, FL performance degrades when MUs are located far from the PS. To mitigate this, [48] introduces Hierarchical OTA FL (HOTAFL), which leverages intermediary servers (ISs) to form local clusters near MUs. These ISs first aggregate local updates within their clusters and then transmit the aggregated results to the PS. Moreover, [49], a wireless-based hierarchical FL scheme is proposed, where ISs are deployed to form clusters in regions with dense MU presence. The scheme leverages OTA aggregation at two levels: first, for local model aggregation within each cluster between MUs and their corresponding IS, and second, for global aggregation between ISs and the PS.

Recent studies have extended the OTA FL framework by integrating it with emerging wireless technologies such as integrated sensing and communication (ISAC) and intelligent reflecting surfaces (IRS), aiming to further enhance communication efficiency, reliability, and adaptability in practical deployments. [50] proposes optimizing node selection, transmit equalization coefficients, IRS phase shifts, and receiver configurations to minimize aggregation error in OTA FL. This integrated design enhances model aggregation accuracy under wireless impairments and demonstrates the potential of IRS-assisted OTA FL systems. In [51], the authors propose an integrated framework that combines sensing and federated learning by leveraging shared wireless resources, where the server simultaneously

performs sensing and FL within the same time-frequency resources. They introduce techniques such as receive beamforming and over-the-air computation to manage interference and enable efficient simultaneous execution of both tasks. Also in [52], authors develop a joint scheduling and beamforming framework to optimize the performance of OTA FL when the PS is also performing sensing. They address a joint beamforming and device scheduling problem, aiming to maximize the number of active devices while ensuring that both sensing and model aggregation errors stay within acceptable bounds.

2.3 Federated Learning with Energy Harvesting Devices

Energy harvesting (EH) has emerged as a promising solution to support sustainable and autonomous operation in wireless networks. In recent years, a wide range of natural and artificial energy sources have been explored to power low-energy devices. These include solar radiation, indoor lighting, mechanical vibrations, thermal gradients, and even biochemical processes [53]. Each energy source and harvesting method differs in capacity, efficiency, and deployment feasibility, which directly affects how they are integrated into wireless communication systems. Moreover, energy harvesting introduces new challenges, such as the intermittency and randomness of available energy, adding complexity to system design, scheduling, and resource management in energy-constrained environments.

Energy harvesting devices are increasingly employed in wireless communication systems due to their ability to continuously obtain energy from the environment. Additionally, the fundamental limits of such systems, including the channel capacity with unit-sized batteries, have been analyzed in works such as [53, 54]. FL for energy harvesting devices is another area of interest, and recently, various studies have approached it from multiple perspectives. The studies' scope includes maximizing data utility [55], achieving balanced client participation [56], and utilizing a device scheduling strategy based on available energy and channel

coefficients [57]. OTA FL with energy harvesting devices has been studied in different scenarios. The studies include client selection and receive beamforming design [58], weighted averaging with respect to their latest energy arrivals to prevent bias [59]. Recent research focuses on user scheduling with online transceiver design optimization [60] and modeling Markov decision processes for joint device scheduling and power management [61]. However, current approaches overlook the impact of user data distributions in energy harvesting OTA FL for wireless fading channel setups.

In FL systems involving EH devices, energy consumption on the client side becomes a key bottleneck due to the unpredictable and limited nature of harvested energy. Although each FL round typically begins with the PS broadcasting the global model to selected users, this downlink reception phase is often not where the primary energy consumption occurs for clients. The parameter server is generally assumed to have ample energy and bandwidth resources, enabling high-quality and efficient broadcasting. Moreover, in OTA FL setups, this phase does not require active decoding at the client side, as the global model can be transmitted using analog signaling or shared reference schemes. As a result, the reception cost is minimal compared to the energy demands of local computation and uplink communication. Nevertheless, clients must still allocate a small amount of energy for synchronization and basic signal processing to receive the transmitted model.

The dominant sources of energy consumption arise during the local update and uplink transmission phases. Local training involves executing multiple steps of SGD, each requiring a large number of floating-point operations during the forward and backward passes of the model. The energy cost scales with both the number of local iterations and the complexity of the model, i.e., its depth and number of parameters. For low-power EH devices, this computational load can be significant, especially when accounting for memory access and on-chip caching operations. Once training is completed, the uplink transmission of the local model update back to the PS often becomes the most energy-intensive task. In OTA FL settings, the users transmit their updates simultaneously over a shared wireless channel, which necessitates sufficient transmit power to ensure reliable

superposition and aggregation at the PS. This requirement places further strain on the limited energy budgets of participating devices, particularly in the presence of channel fading or path loss. Consequently, efficient scheduling, adaptive power control, and energy-aware participation policies are critical for maintaining long-term operability and ensuring fairness across EH-enabled FL systems [8, 56, 57, 58, 59, 60, 61].

In large-scale networks with many participating devices and increasingly complex models, the cost of local training in FL becomes a significant concern. As the number of model parameters N increases, the energy and time required for each local update grow rapidly, often scaling quadratically as $\mathcal{O}(N^2)$ due to the underlying matrix operations involved in forward and backward passes [62]. This becomes especially problematic for EH devices, which operate under strict energy budgets and cannot sustain prolonged or intensive computations. When combined with multiple local iterations and the coordination overhead in massive networks, the cumulative training cost can render FL impractical. Dealing with EH devices adds another layer of complexity, requiring careful consideration of energy availability, computational efficiency, and transmission costs when deciding between FL and centralized learning. In such scenarios, particularly when data privacy is not a critical constraint, traditional centralized machine learning, where raw data is transmitted to a cloud server, may be a more efficient alternative. By shifting the computation to energy-rich cloud infrastructure, centralized learning can more easily accommodate large models and large user populations, making it preferable in some large-scale EH network deployments. However, it should also be noted that if the total training data volume is extremely large, transmitting all of it to a central server may itself become prohibitively expensive in terms of bandwidth and energy, potentially offsetting the benefits of centralized learning.

In our setup, we adopt a simplified abstraction for modeling client-side energy consumption, where all major operations within a communication round, including initialization, synchronization with the PS, performing the local update, and transmitting the resulting model update, are collectively assumed to consume one unit of energy. Note that there is also energy consumption associated with listening to the PS and performing initialization operations at the user side. The

required energy for these operations is very small, i.e., negligible compared to the amount needed for active participation. Hence, in our abstraction, we assume that the users have sufficient energy for them in every round. This abstraction does not rely on hardware-specific metrics such as floating-point operation counts or exact transmission power levels; instead, it focuses on the binary decision of whether a device possesses the required one unit of harvested energy to participate in the current iteration. At the start of each round, only devices meeting this energy requirement are eligible to join the training process. This model enables a direct coupling between the stochastic energy arrival process and user participation, allowing for a clear and tractable evaluation of the proposed scheduling and aggregation strategies under energy harvesting constraints.

2.4 Chapter Summary

In this chapter, we have covered foundational topics that are essential for understanding the material presented in the following chapters. We began by introducing traditional FL, outlining its general framework and key algorithms. We then discussed challenges related to non-i.i.d. data distributions and heterogeneous system setups, followed by an overview of privacy concerns in FL. Additionally, we reviewed various user selection strategies designed to enhance learning performance and efficiency. We then turned our attention to wireless FL, with a particular focus on OTA aggregation and its transmission model. Finally, we introduced the motivation behind incorporating energy-harvesting MUs into the FL framework and summarized relevant literature in that context. The rest of this thesis presents our original contributions and findings on FL over wireless communication channels.

Chapter 3

Update Estimation and Scheduling for Over-the-Air Federated Learning with Energy Harvesting Devices

In this chapter, we explore user scheduling strategies for federated learning (FL) with energy harvesting (EH) devices and heterogeneous data distribution by scheduling active users based on their data distribution using over-the-air (OTA) transmission. Firstly, we select a subset of users that achieves a uniform data distribution using exact data distribution information to ensure the aggregate of selected users' updates approximates that of all users. Next, since the sharing of data distribution raises privacy concerns, we estimate the user updates from aggregated signals at the parameter server (PS) via least-squares estimation (LSE), enabling user selection without revealing data distribution information. Unlike existing works, we use OTA aggregated signals instead of individual user updates for transmission efficiency, along with participation information to estimate local model update representations. These representative updates are utilized to group users into clusters, improving learning performance and identifying redundant information in the system.

The chapter is organized as follows. Section 3.1 introduces the FL setup and EH devices. Section 3.2 analyzes OTA FL for EH devices with a unit-sized battery and provides a convergence analysis. In Section 3.3, we propose update estimation and user scheduling policies, and present extensive numerical results in Section 3.4. We conclude the chapter in Section 3.5.

3.1 System Model

We consider an FL system with heterogeneous data distribution for M energy harvesting devices. We assume that the mobile users do not have channel state information (CSI) and transmit their updates to the PS through a fading multiple access channel (MAC) using over-the-air transmission. The PS employs K receive antennas to align the received signals in the absence of CSI at the transmitters (CSIT) using aggregated channel information.

In FL, the primary objective is to minimize a global loss function, denoted as $F(\boldsymbol{\theta})$, collaboratively across M devices, where $\boldsymbol{\theta} \in \mathbb{R}^{2N}$ represents the parameters of the global model to be optimized. The global loss function is defined as

$$F(\boldsymbol{\theta}) = \sum_{m=1}^{M} \frac{|\mathcal{B}_m|}{B} F_m(\boldsymbol{\theta}), \tag{3.1}$$

where $F_m(\boldsymbol{\theta})$ represents the average empirical local loss for the m-th user with model parameters $\boldsymbol{\theta}$. For the m-th user with the local dataset $\mathcal{B}_m$, for $m \in [M]$, defining $[M] = \{1, \ldots, M\}$, and $B \triangleq \sum_{m=1}^{M} |\mathcal{B}_m|$, we have

$$F_m(\boldsymbol{\theta}) = \frac{1}{|\mathcal{B}_m|} \sum_{\boldsymbol{u} \in \mathcal{B}_m} f(\boldsymbol{\theta}, \boldsymbol{u}), \tag{3.2}$$

with $f(\boldsymbol{\theta}, u)$ denoting the empirical loss function corresponding to the u-th data sample in the local dataset B_m.

In FL with EH devices, unlike traditional FL, the limited energy availability can cause some users to lack sufficient energy to perform local computations or transmissions. Consequently, contributions will only come from the mobile

users (MUs) that have enough energy and are selected based on the adopted scheduling policy. At each global iteration t, the PS broadcasts the latest global model, $\boldsymbol{\theta}(t)$. In response, the selected MUs perform τ local stochastic gradient descent (SGD) iterations to minimize their individual local loss functions, $F_m(\boldsymbol{\theta})$, for $m \in \mathcal{S}(t)$, where $\mathcal{S}(t)$ is the set of scheduled users. Subsequently, the model updates obtained by the mobile users are transmitted back to the PS to contribute to the global learning process.

To compute the local model updates, the m-th user (for the i-th local and t-th global iteration) employs the following update rule:

$$\boldsymbol{\theta}_m^{i+1}(t) = \boldsymbol{\theta}_m^i(t) - \eta_m^i(t) \nabla F_m(\boldsymbol{\theta}_m^i(t), \xi_m^i(t)), \tag{3.3}$$

where $i \in [\tau]$, with τ representing the number of local iterations, $\eta_m^i(t)$ is the learning rate and $\nabla F_m\left(\boldsymbol{\theta}_m^i(t), \xi_m^i(t)\right)$ represents the stochastic gradient estimate for the $\boldsymbol{\theta}_m^i(t)$ and the local mini-batch sample ξ_m^i randomly chosen from the local dataset $\mathcal{B}_m$.

After the local SGD steps, the m-th user computes the model update, which is aimed to be shared with the PS as

$$\Delta\boldsymbol{\theta}_m(t) = \boldsymbol{\theta}_m^\tau(t) - \boldsymbol{\theta}_m^1(t). \tag{3.4}$$

Using OTA transmission over a fading MAC, the received signal at the k-th antenna of the PS at iteration t is given as

$$\boldsymbol{y}_{PS,k}(t) = \sum_{m \in \mathcal{S}(t)} \boldsymbol{h}_{m,k}(t) \circ \boldsymbol{x}_m(t) + \boldsymbol{z}_{PS,k}(t), \tag{3.5}$$

where $\boldsymbol{x}_m(t)$ is the signal transmitted by the m-th user, which corresponds to the complex-valued model update defined in (2.8c), and $\boldsymbol{h}_{m,k}(t)$ is the independent and identically distributed (i.i.d.) channel gains from the m-th user to the k-th antenna with complex Gaussian entries $h_{m,k}^n(t) \sim \mathcal{CN}(0, \sigma_h^2)$. Similarly, $z_{PS,k}^n(t)$ denotes the n-th entry of the channel noise, $\boldsymbol{z}_{PS,k}(t)$, which is i.i.d. circularly symmetric white Gaussian noise (AWGN), i.e., it is distributed according to $\mathcal{CN}(0, \sigma_z^2)$.

The PS uses the received signals from the K antennas to update the global model as

$$\boldsymbol{\theta}_{PS}(t+1) = \boldsymbol{\theta}_{PS}(t) + \Delta\hat{\boldsymbol{\theta}}_{PS}(t), \tag{3.6}$$

where $\boldsymbol{\theta}_{\mathrm{PS}}(t)$ represents the global model vector at the server at global iteration t and $\Delta\hat{\boldsymbol{\theta}}_{PS}(t)$ is the noisy estimate of the average of the local updates. Note that if there were no noise or fading, the average of the local updates would be

$$\Delta\boldsymbol{\theta}_{PS}(t) = \frac{1}{|\mathcal{S}(t)|}\sum_{m\in\mathcal{S}(t)} \Delta\theta_m(t). \tag{3.7}$$

3.1.1 EH Devices

We study FL with OTA for EH devices, each with a unit-sized battery. At each global iteration, devices harvest energy with varying success, storing it for future use following the harvest-store-use approach [63]. Surplus energy is lost if the battery is full during energy arrivals. Local SGD and update transmissions consume one energy unit global per iteration, highlighting stochastic energy availability as a key constraint.

We consider a Bernoulli energy arrival process where, at each global iteration t, the m-th user receives unit energy with probability $p_e^m(t)$. Active users consume available energy in the battery, while inactive users store it for future use. Energy arrivals are shared with the PS after each iteration, as also adopted in [60, 58].

3.2 OTA FL with EH Devices with Unit Battery

In this section, we consider FL with OTA aggregation, where only active users contribute to the iterations due to limited energy arrivals and provide a brief convergence analysis.

The model updates of the scheduled users, $\Delta\boldsymbol{\theta}_m^{cx}(t) \in \mathbb{C}^N$, $m \in \mathcal{S}(t)$, are

transmitted as complex signals at iteration t, represented as

$$\Delta\boldsymbol{\theta}_m^{re}(t) \triangleq \left[\Delta\theta_m^1(t), \Delta\theta_m^2(t), \ldots, \Delta\theta_m^N(t)\right]^T, \tag{3.8a}$$

$$\Delta\boldsymbol{\theta}_m^{im}(t) \triangleq \left[\Delta\theta_m^{N+1}(t), \Delta\theta_m^{N+2}(t), \ldots, \Delta\theta_m^{2N}(t)\right]^T, \tag{3.8b}$$

$$\Delta\boldsymbol{\theta}_m^{cx}(t) \triangleq \Delta\boldsymbol{\theta}_m^{re}(t) + j\Delta\boldsymbol{\theta}_m^{im}(t). \tag{3.8c}$$

Using the channel output at each antenna as given in (3.5), the PS combines the signals from the K antennas using the sum of the channel gains from the scheduled users as follows:

$$\boldsymbol{y}_{PS}(t) = \frac{1}{K}\sum_{k=1}^{K}\left(\sum_{m\in\mathcal{S}(t)}\boldsymbol{h}_{m,k}(t)\right)^* \circ \boldsymbol{y}_{PS,k}(t). \tag{3.9}$$

where the received signals $\boldsymbol{y}_{PS,k}(t)$'s, corresponding to the transmitted signals $\Delta\boldsymbol{\theta}_m^{cx}(t)$'s, are given in (3.5), and $\circ$ denotes the element-wise product.

The n-th symbol of (3.9) can be partition into three signals

$$\begin{aligned} y_{PS}^n(t) = &\underbrace{\sum_{m\in\mathcal{S}(t)}\left(\frac{1}{K}\sum_{k=1}^{K}|h_{m,k}^n(t)|^2\right)\Delta\theta_m^{n,cx}(t)}_{y_{PS}^{n,sig}(t)\text{(signal term)}} \\ &+\underbrace{\frac{1}{K}\sum_{m\in\mathcal{S}(t)}\sum_{\substack{m'\in\mathcal{S}(t)\\ m'\neq m}}\sum_{k=1}^{K}(h_{m,k}^n(t))^* h_{m',k}^n(t)\Delta\theta_{m'}^{n,cx}(t)}_{y_{PS}^{n,int}(t)\text{(interference term)}} \\ &+\underbrace{\frac{1}{K}\sum_{m\in\mathcal{S}(t)}\sum_{k=1}^{K}(h_{m,k}^n(t))^* z_{PS,k}^n(t)}_{y_{PS}^{n,noise}(t)\text{(noise term)}}. \end{aligned} \tag{3.10}$$

As shown in [39], the variance of the interference coefficient $\Delta\theta_{m'}^{n,cx}(t)$ decreases with the number of antennas K. Hence, a sufficient number of antennas allows for accurate estimation and recovery of noisy aggregated updates as

$$\Delta\hat{\boldsymbol{\theta}}_{PS}^n(t) = \frac{1}{|\mathcal{S}(t)|\,\sigma_h^2}\mathrm{Re}\{y_{PS}^n(t)\}, \tag{3.11a}$$

$$\Delta\hat{\boldsymbol{\theta}}_{PS}^{n+N}(t) = \frac{1}{|\mathcal{S}(t)|\,\sigma_h^2}\mathrm{Im}\{y_{PS}^n(t)\}, \tag{3.11b}$$

to update the global model, as given in (3.6).

3.2.1 Convergence Analysis

In this section, we provide convergence analysis for the OTA FL with EH devices and no CSIT by upper-bounding the distance between our model estimate and the optimal model.

The optimal solution minimizing (3.1) is $\boldsymbol{\theta}^* \triangleq \arg\min_{\boldsymbol{\theta}} F(\boldsymbol{\theta})$, with optimal loss $F^* = F(\boldsymbol{\theta}^*)$. For user $m \in [M]$, the optimal local model is $\boldsymbol{\theta}_m^* \triangleq \arg\min_{\boldsymbol{\theta}_m} F_m(\boldsymbol{\theta}_m)$, with corresponding loss $F_m^* = F(\boldsymbol{\theta}_m^*)$.

3.2.1.1 Preliminaries

The amount of bias and heterogeneity across devices is represented by the following non-negative parameter

$$\Gamma = F^* - \sum_{m=1}^{M} \frac{B_m}{B} F_m^*. \tag{3.12}$$

A higher Γ value indicates significant non-independent and identically distributed (non-i.i.d.) data distribution, while $\Gamma \to 0$ reflects near i.i.d. data.

We consider the same learning rate across users and local iterations, $\eta_m^i(t) = \eta(t)$, but allow it to vary between global iterations. The local model update at the m-th user for global iteration t and local iteration $i \in [\tau]$ is given as

$$\boldsymbol{\theta}_m^{i+1}(t) \ - \boldsymbol{\theta}_m^1(t) \ = -\eta(t) \ \sum_{l=1}^{i} \nabla F_m \left(\boldsymbol{\theta}_m^l(t), \xi_m^l(t) \ \right). \tag{3.13}$$

To perform the convergence analysis, similar to the existing studies [39, 47, 49], we assume the loss functions $F_1, \ldots, F_M$ are all L-smooth and μ-strongly convex; that is, for all $v, w \in \mathbb{R}^{2N}$, and $m \in [M]$,

$$F_m(\boldsymbol{v}) - F_m(\boldsymbol{w}) \leq \langle \boldsymbol{v} - \boldsymbol{w}, \nabla F_m(\boldsymbol{w}) \rangle + \frac{L}{2} \|\boldsymbol{v} - \boldsymbol{w}\|_2^2, \tag{3.14}$$

$$F_m(\boldsymbol{v}) - F_m(\boldsymbol{w}) \geq \langle \boldsymbol{v} - \boldsymbol{w}, \nabla F_m(\boldsymbol{w}) \rangle + \frac{\mu}{2} \|\boldsymbol{v} - \boldsymbol{w}\|_2^2. \tag{3.15}$$

Also, it is assumed that the expected squared ℓ_2-norm of the stochastic gradients are bounded; that is, for all $i \in [\tau]$, $m \in [M]$, and t,

$$\mathbb{E}_\xi \left[\left\| \nabla F_m \left(\boldsymbol{\theta}_m^i(t), \xi_m^i(t) \right) \right\|_2^2 \right] \leq G^2. \tag{3.16}$$

3.2.1.2 Convergence Rate

For the convergence analysis of OTA FL with EH devices and data heterogeneity, we compare the optimal model with our system model, where only a subset of users participate in each iteration. Using these findings, we will make user scheduling decisions to minimize the discrepancy in the resulting bound. Our main result is as follows.

Theorem 1. *For* $0 < \eta(t) \leq \min\left\{1, \frac{1}{\mu\tau}\right\}, \forall t$. *We have*

$$\mathbb{E}\left[\|\boldsymbol{\theta}(t) - \boldsymbol{\theta}^*\|_2^2\right] \leq \left(\prod_{i=0}^{t-1} A(i)\right) \|\boldsymbol{\theta}(0) - \boldsymbol{\theta}^*\|_2^2 + \sum_{j=0}^{t-1} B(j) \prod_{i=j+1}^{t-1} A(i), \tag{3.17}$$

with

$$A(i) \triangleq 1 - \mu\eta(i) \ \left(\tau - \eta(i)(\tau - 1)\right), \tag{3.18}$$

$$\begin{aligned} B(i) \triangleq & \frac{\eta^2(i)\tau^2 G^2}{K} + \frac{\sigma_z^2 N}{K|\mathcal{S}(i)|\sigma_h^2} \\ & + \left(1 + \mu(1 - \eta(i))\right) \eta^2(i) G^2 \frac{\tau(\tau-1)(2\tau-1)}{6} \\ & + \eta^2(i)(\tau^2 + \tau - 1)G^2 + 2\eta(i)(\tau - 1)\Gamma \\ & + \left(\eta^2(i)\tau(\tau-1)LG + \eta(i)\tau\epsilon\right)^2 \\ & + \left(\eta^2(i)\tau(\tau-1)LG + \eta(i)\tau\epsilon\right) c, \end{aligned} \tag{3.19}$$

for the gradient approximation error ϵ *and some constant* $c \geq 0$.

Proof. We define auxiliary variables:

$$\boldsymbol{w}(t+1) \triangleq \boldsymbol{\theta}_{PS}(t) + \frac{1}{|\mathcal{S}(t)|} \sum_{m \in \mathcal{S}(t)} \Delta\boldsymbol{\theta}_m(t), \tag{3.20}$$

$$\boldsymbol{v}(t+1) \triangleq \boldsymbol{\theta}_{PS}(t) + \frac{1}{M} \sum_{m=1}^{M} \Delta\boldsymbol{\theta}_m(t). \tag{3.21}$$

From (3.6), we have $\boldsymbol{\theta}_{PS}(t+1) = \boldsymbol{\theta}_{PS}(t) + \Delta\hat{\boldsymbol{\theta}}_{PS}(t)$. Using this, we can derive

$$\begin{aligned}
&\|\boldsymbol{\theta}_{PS}(t+1) - \boldsymbol{\theta}^*\|_2^2 \\
&= \|\boldsymbol{\theta}_{PS}(t+1) - \boldsymbol{w}(t+1) + \boldsymbol{w}(t+1) - \boldsymbol{\theta}^*\|_2^2 \\
&= \|\boldsymbol{\theta}_{PS}(t+1) - \boldsymbol{w}(t+1)\|_2^2 + \|\boldsymbol{w}(t+1) - \boldsymbol{\theta}^*\|_2^2 \\
&\quad + 2\langle \boldsymbol{\theta}_{PS}(t+1) - \boldsymbol{w}(t+1), \boldsymbol{w}(t+1) - \boldsymbol{\theta}^* \rangle.
\end{aligned} \tag{3.22}$$

To bound these terms, we employ the following lemmas.

Lemma 1. *For the first and third terms in (3.22), we have*

$$\mathbb{E}\left[\|\boldsymbol{\theta}_{PS}(t+1) - \boldsymbol{w}(t+1)\|_2^2\right] \leq \frac{\eta^2(t)\tau^2 G^2}{K} + \frac{\sigma_z^2 N}{K|\mathcal{S}(t)|\sigma_h^2},$$

and

$$\mathbb{E}\big[\langle \boldsymbol{\theta}_{PS}(t+1) - \boldsymbol{w}(t+1), \boldsymbol{w}(t+1) - \boldsymbol{\theta}^* \rangle\big] = 0.$$

Proof. The proofs are similar to Lemmas 1 and 3 in [39]. □

For the second term in (3.22), we proceed as follows:

$$\begin{aligned}
&\|\boldsymbol{w}(t+1) - \boldsymbol{\theta}^*\|_2^2 \\
&= \|\boldsymbol{w}(t+1) - \boldsymbol{v}(t+1) + \boldsymbol{v}(t+1) - \boldsymbol{\theta}^*\|_2^2 \\
&= \|\boldsymbol{w}(t+1) - \boldsymbol{v}(t+1)\|_2^2 + \|\boldsymbol{v}(t+1) - \boldsymbol{\theta}^*\|_2^2 \\
&\quad + 2\langle \boldsymbol{w}(t+1) - \boldsymbol{v}(t+1), \boldsymbol{v}(t+1) - \boldsymbol{\theta}^* \rangle.
\end{aligned} \tag{3.23}$$

Lemma 2. *For the second term in (3.23), we have*

$$\begin{aligned}
&\mathbb{E}\left[\|\boldsymbol{v}(t+1) - \boldsymbol{\theta}^*\|_2^2\right] \\
&\leq (1 - \mu\eta(t)\ \ (\tau - \eta(t)(\tau-1)))\, \mathbb{E}\left[\|\boldsymbol{\theta}_{PS}(t)\ \ - \boldsymbol{\theta}^*\|_2^2\right] \\
&\quad + (1 + \mu(1-\eta(t)))\, \eta^2(t) G^2 \frac{\tau(\tau-1)(2\tau-1)}{6} \\
&\quad + \eta^2(t)(\tau^2 + \tau - 1)G^2 + 2\eta(t)(\tau-1)\Gamma.
\end{aligned} \tag{3.24}$$

Proof. The proof follows the same argument in [39, Lemma 2]. □

Lemma 3. *For the first term in (3.23), we have*

$$\mathbb{E}\left[\|\boldsymbol{w}(t+1)-\boldsymbol{v}(t+1)\|_2^2\right] = \left(\eta^2(t)\tau(\tau-1)LG+\eta(t)\tau\epsilon\right)^2.$$

Proof. The proof is similar to [29, Lemma 1]. □

Lemma 4. *The third term in (3.23) is bounded by*

$$\mathbb{E}\left[2\langle \boldsymbol{w}(t+1)-\boldsymbol{v}(t+1), \boldsymbol{v}(t+1)-\boldsymbol{\theta}^*\rangle\right] \leq \left(\eta^2(t)\tau(\tau-1)LG+\eta(t)\tau\epsilon\right)c, \tag{3.25}$$

for some constant $c \geq 0$, which is related to Γ, G and μ.

Proof. The proof follows a similar line as Lemma 1 in [29]. □

By recursively iterating through the results of these Lemmas, we reach the conclusion of the theorem. □

We note that A(i) represents the decay rate of the distance from the initial starting point to the optimal solution. In $B(i)$, the first two terms represent the transmission error due to the wireless fading MAC with blind transmitters, and the third and fourth terms are related to federated averaging. Additionally, we emphasize that the last two terms represent the error caused by partial user participation similar to [29], with ϵ in (3.19) being the gradient approximation error defined as follows

$$\epsilon \triangleq \left\| \frac{1}{M}\sum_{m=1}^{M}\nabla F_m(\theta_m(t)) - \frac{1}{|S(t)|}\sum_{m\in\mathcal{S}(t)}\nabla F_m(\theta_m(t)) \right\|_2. \tag{3.26}$$

Via user scheduling, our aim is to select the subset of users with the average local updates closest to the full participation case, thereby minimizing the error ϵ in (3.26). Hence, in the next section, we propose user selection and scheduling

approaches that minimize the distance between the trained FL model and the optimal model, even in scenarios involving EH devices and highly non-i.i.d. data distributions.

3.3 Update Estimation and User Scheduling

In this section, we demonstrate that the data distribution of users is critical in the scheduling procedure, especially for highly non-i.i.d data distributions, and can be used to minimize the error bound related to the partial participation characteristics of EH devices. First, we propose an entropy-based user scheduling policy with known data distribution. Then, we extend our discussion to the unknown data distribution case and show that the data distribution characteristics can be estimated via least-squares estimation on user update representations to be used in user scheduling.

3.3.1 Entropy-based User Scheduling with Known Data Distributions

Assuming that all the MUs reveal their data distribution to the PS beforehand, we can select a subset of users that effectively represents all data labels in the network. Based on this, the PS characterizes the label distribution of each user $m \in [M]$ as $L_m = [l_{m,0}, l_{m,1}, \ldots, l_{m,N_c-1}]$, where N_c is the total number of classes, and l_{m,n_c} represents the portion of the m-th user's data corresponding to label n_c. At each iteration, the PS computes the label distribution for all available user subsets as a probability mass function and selects the one with the highest Shannon entropy, indicating the most balanced label distribution available. While this strategy is similar to that in [30], we extend our approach to a more practical setup that incorporates OTA transmission, wireless fading MAC, and blind transmitters, showing the effectiveness of entropy-based user selection for EH devices under practical constraints.

3.3.2 User Clustering and Scheduling with Unknown Data Distribution

We consider a more realistic case, where the PS lacks data distribution information, ensuring a level of user privacy. In this case, we rely on the relationship between the model updates from users and their underlying data distribution, similar to [27, 28, 29]. Unlike these studies, our approach is constrained to using a noisy estimate of the sum of updates provided by all selected users at a given iteration. We demonstrate that a representation of the user updates can be estimated at the PS, allowing them to be grouped into clusters based on the similarities between their representations to minimize the error from partial participation in (3.26). This approach selects suitable users while preventing redundant information transfer and conserving energy under the constraints of EH devices.

To achieve this, we use LSE to create a representation of the updates. Over T estimation iterations, the PS stores normalized global updates from (3.11) while all the active users participate without scheduling, termed the *estimation phase*. Note that we normalize the user updates to mitigate possible scale discrepancies. At the end of this estimation window, PS estimates the representative updates based on stored global updates and participation information. We emphasize that the goal is to estimate a representation of user updates rather than recovering the individual updates themselves.

We define a matrix $\hat{\boldsymbol{\Theta}}_{PS}$, whose rows represent global model updates $\Delta\hat{\boldsymbol{\theta}}_{PS}(t)$ from (3.11).

$$\hat{\boldsymbol{\Theta}}_{PS} = \begin{bmatrix} \hat{\boldsymbol{\Theta}}_{PS,t-T+1} \\ \hat{\boldsymbol{\Theta}}_{PS,t-T+2} \\ \vdots \\ \hat{\boldsymbol{\Theta}}_{PS,j} \\ \vdots \\ \hat{\boldsymbol{\Theta}}_{PS,t} \end{bmatrix}_{T\times 2N} \tag{3.27}$$

For the j-th iteration with $j \leq T$, the j-th row of this matrix can be expressed as $\hat{\boldsymbol{\Theta}}_{PS,j} = \boldsymbol{A}_j \boldsymbol{\Theta}_j + \boldsymbol{N}_j^{'}$, where $\boldsymbol{A}_j$ is a binary participation vector with $\boldsymbol{A}_j \in \{0,1\}^{1\times M}$, and $\boldsymbol{\Theta}_j \in \mathbb{R}^{M\times 2N}$, with each row representing the local model update for a specific user $m \in [M]$, denoted as $\Delta\boldsymbol{\theta}_{j,m}$. Additionally, $\boldsymbol{N}_j^{'} \in \mathbb{R}^{1\times 2N}$, whose d-th element is denoted by $N_{j,d}^{'}$ for $d \in [2N]$, represents the effective noise arising from MAC fading, AWGN, and PS combining errors. $\hat{\boldsymbol{\Theta}}_{PS,j}$ is expressed as:

$$\hat{\boldsymbol{\Theta}}_{PS,j} = \boldsymbol{A}_j \boldsymbol{\Theta}_j + \boldsymbol{N}_j^{'}, \tag{3.28}$$

which is equivalent to and can be written as

$$\hat{\boldsymbol{\Theta}}_{PS,j} = \boldsymbol{A}_j \begin{bmatrix} \Delta\boldsymbol{\theta}_{j,1} \\ \Delta\boldsymbol{\theta}_{j,2} \\ \vdots \\ \Delta\boldsymbol{\theta}_{j,M} \end{bmatrix}_{M\times 2N} + \begin{bmatrix} N_{j,1}^{'} N_{j,2}^{'} & \dots & N_{j,2N}^{'} \end{bmatrix}_{1\times 2N}. \tag{3.29}$$

We also define $\boldsymbol{\Theta}_{rep} \in \mathbb{R}^{M\times 2N}$ as a representation of local updates. Using this, $\hat{\boldsymbol{\Theta}}_{PS,j}$ can be written as:

$$\hat{\boldsymbol{\Theta}}_{PS,j} = \boldsymbol{A}_j (\boldsymbol{\Theta}_{rep} + \boldsymbol{\Theta}_{diff,j}) + \boldsymbol{N}_j^{'}, \tag{3.30}$$

$$\hat{\boldsymbol{\Theta}}_{PS,j} = \boldsymbol{A}_j \left(\begin{bmatrix} \Delta\boldsymbol{\theta}_{rep,1} \\ \Delta\boldsymbol{\theta}_{rep,2} \\ \vdots \\ \Delta\boldsymbol{\theta}_{rep,M} \end{bmatrix}_{M\times 2N} + \begin{bmatrix} \Delta\boldsymbol{\theta}_{diff,j,1} \\ \Delta\boldsymbol{\theta}_{diff,j,2} \\ \vdots \\ \Delta\boldsymbol{\theta}_{diff,j,M} \end{bmatrix}_{M\times 2N} \right) + \begin{bmatrix} N_{j,1}^{'} N_{j,2}^{'} & \dots & N_{j,2N}^{'} \end{bmatrix}_{1\times 2N}. \tag{3.31}$$

where $\boldsymbol{\Theta}_{diff,j}$ is defined as the difference between $\boldsymbol{\Theta}_j - \boldsymbol{\Theta}_{rep}$. Combining (3.30) for $j \in \{1, \ldots, T\}$ and defining a total noise term $\boldsymbol{N}_j^* \triangleq \boldsymbol{A}_j \boldsymbol{\Theta}_{diff,j} + \boldsymbol{N}_j^{'}$, which represents the noise due to the channel, interference from the blind transmitters, and the difference between representative updates and the real updates, we obtain

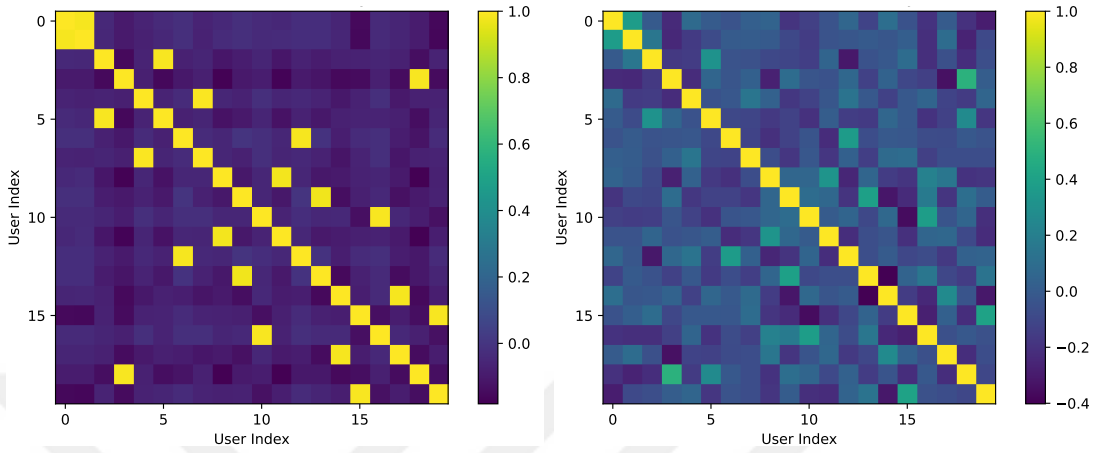
$$\hat{\boldsymbol{\Theta}}_{PS} = \boldsymbol{A}\boldsymbol{\Theta}_{rep} + \boldsymbol{N}^*, \tag{3.32}$$

$$\hat{\boldsymbol{\Theta}}_{PS} = \begin{bmatrix} \hat{\boldsymbol{\Theta}}_{PS,t-T+1} \\ \hat{\boldsymbol{\Theta}}_{PS,t-T+2} \\ \vdots \\ \hat{\boldsymbol{\Theta}}_{PS,t} \end{bmatrix}_{T\times 2N} = \begin{bmatrix} 1 & \cdots & 1 \\ 0 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 0 \end{bmatrix}_{T\times M} \begin{bmatrix} \Delta\boldsymbol{\theta}_{rep,1} \\ \Delta\boldsymbol{\theta}_{rep,2} \\ \vdots \\ \Delta\boldsymbol{\theta}_{rep,M} \end{bmatrix}_{M\times 2N} + \begin{bmatrix} \boldsymbol{N}_1^* \\ \boldsymbol{N}_2^* \\ \vdots \\ \boldsymbol{N}_T^* \end{bmatrix}_{T\times 2N} \tag{3.33}$$

where $\hat{\boldsymbol{\Theta}}_{PS} = [\Delta\hat{\boldsymbol{\theta}}_{PS}(t-T+1); \cdots ; \Delta\hat{\boldsymbol{\theta}}_{PS}(t)] \in \mathbb{R}^{T\times 2N}$ and ';' represents row-wise concatenation. Additionally, we have $\boldsymbol{A} \in \{0,1\}^{T\times M}$, $\boldsymbol{\Theta}_{rep} \in \mathbb{R}^{M\times 2N}$ and $\boldsymbol{N}^* \in \mathbb{R}^{T\times 2N}$.

By solving LSE of $\boldsymbol{\Theta}_{rep}$ in (3.32), we can get an estimate for the representative updates as $\hat{\boldsymbol{\Theta}}_{rep}$. Using this representation, the PS can infer the characteristics of the users' data distribution as a similarity between user representations, which can then be utilized in the user selection procedure. Notably, the PS infers similarity among user representations without accessing their data distribution, preserving user privacy.

Due to the limited and stochastic energy arrivals, some users might dominate the training and create a bias towards certain labels and certain users. By employing *cosine similarity*, users are clustered into groups to promote diverse user contributions by selecting the expected number of users based on energy distribution from each cluster to ensure unbiased training. This approach helps reduce the bias due to the non-i.i.d data and provides a fair performance among clients, as noted similarly in [27, 29].



(a) Cosine similarity on real user updates. (b) Cosine similarity on estimated user representations.

Figure 3.1: The visualization of cosine similarity on the user clusters.

3.3.3 Visualization of Cosine Similarity Based Clustering

To demonstrate the effectiveness and relevance of cosine similarity in our setup, we present a simple example using the MNIST dataset. A single-layer neural network with $2N = 7850$ parameters is used, and the training involves 20 users, each assigned data from only one class. First, we compute the cosine similarity between user updates at the initial iteration to visualize the inherent relationships. Then, we apply our LSE-based user update estimation method and recompute the cosine similarity, illustrating how well the estimated updates capture the true relationships between users.

In Fig. 3.1, we present the cosine similarity between users based on both their actual updates and their estimated representations. As observed in Fig. 3.1a, the similarity patterns among users are also reflected in the estimated similarities shown in Fig. 3.1b. This demonstrates that user characteristics can be successfully inferred using our proposed strategy, enabling the recovery of the natural clustering structure among users.

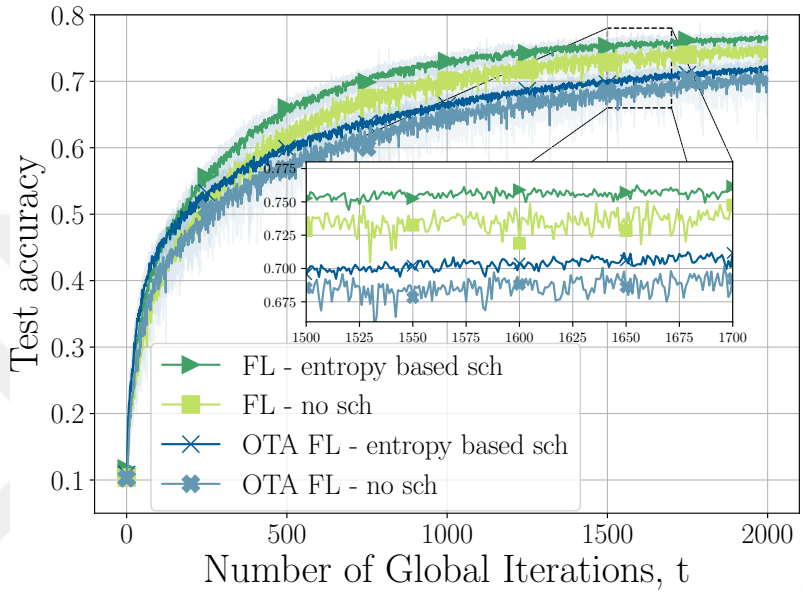
3.4 Numerical Examples

In this section, we evaluate the performance of our proposed user scheduling methods across multiple scenarios. We consider image classification tasks on the MNIST [64], FMNIST [65], and CIFAR-10 [66] datasets under non-i.i.d. data distributions. For MNIST and FMNIST, we use a single-layer neural network with 784 input and 10 output neurons ($2N = 7850$). For CIFAR-10, we use a convolutional neural network (CNN) ($2N = 797962$) as in [67]. Training is performed by SGD with a learning rate of 0.05 and a scheduler, $\tau = 5$ and mini-batch size $|\xi_m(t)| = 100$ for MNIST and FMNIST, and $\tau = 3$ and $|\xi_m(t)| = 128$ for CIFAR-10.

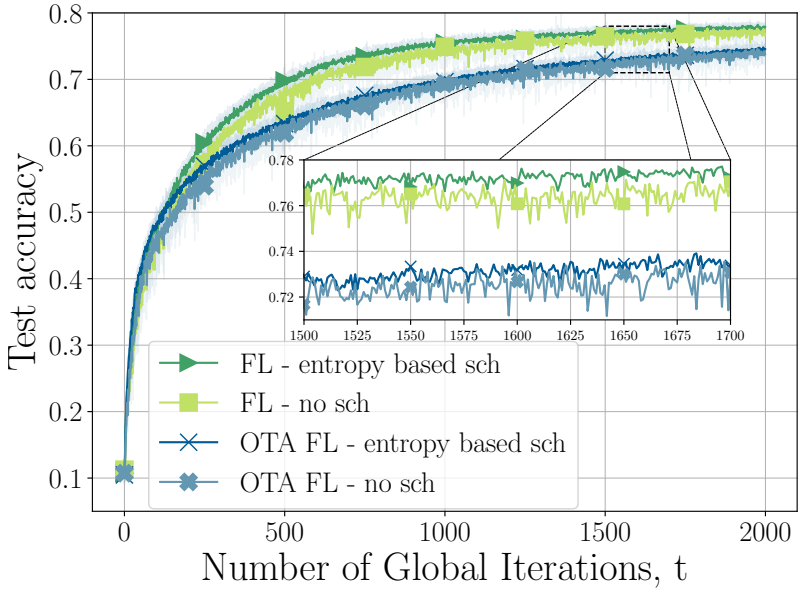
To simulate highly non-i.i.d. data, we consider two different distribution scenarios. In the first scenario, users' data are limited to a fixed number of labels, either 1 or 2 classes assigned per user. In the second one, we use the Dirichlet distribution to sample $\boldsymbol{p_m} \sim \text{Dir}_{N_c}(\beta)$ with $\boldsymbol{p_m} = [p_{m,0}, \cdots, p_{m,N_c-1}]$, and user m receives p_{m,n_c} portion of its data from class $n_c \in [N_c]$. β is a Dirichlet distribution parameter, where smaller values of β lead to more unbalanced partitions.

We evaluate the performance of our wireless OTA FL with highly non-i.i.d data against a baseline scenario without user scheduling, for which the users participate in the global learning process whenever they have energy. Throughout the simulations, users are connected to a PS through a wireless fading MAC where channel gains from each user to each PS antenna are i.i.d.. The selected parameters are $K = 200$, $\sigma_h^2 = 1$, and $\sigma_z^2 = 0.1$.

In Fig. 3.2, we demonstrate the performance of entropy-based scheduling for CIFAR-10 dataset with $\beta \in \{0.1, 0.2\}$ for $M = 100$ users with $|\mathcal{B}_m| = 500$ and $p_e^m(t) = 0.1$ for $m \in [M]$. We observe that the gains from our scheme are more significant in scenarios with greater heterogeneity for both the error-free and OTA FL cases. As shown in the figure, as the distribution becomes more heterogeneous as in Fig 3.2a, (i.e., $\beta = 0.1$), the impact of the proposed entropy-based scheduling increases. These plots demonstrate that diverse user selection, which leads to a



(a) $\beta = 0.1$.



(b) $\beta = 0.2$.

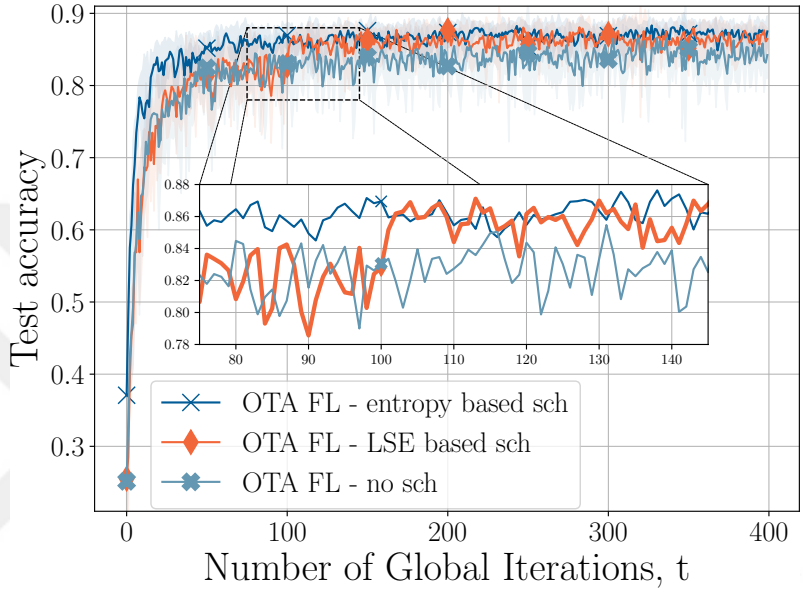
Figure 3.2: The mean test accuracy of entropy-based scheduling for CIFAR-10 with $M = 100$, $|\mathcal{B}_m| = 500$ and $p_e^m(t) = 0.1$, $\forall m, t$.

more balanced aggregated data distribution, results in higher accuracy. The improvements are evident in both error-free FL and OTA FL setups, highlighting the effectiveness of diversity-aware scheduling in mitigating data heterogeneity. This shows that when data distributions are known in the OTA FL setup, our proposed method can be effectively leveraged to achieve notable performance improvements.

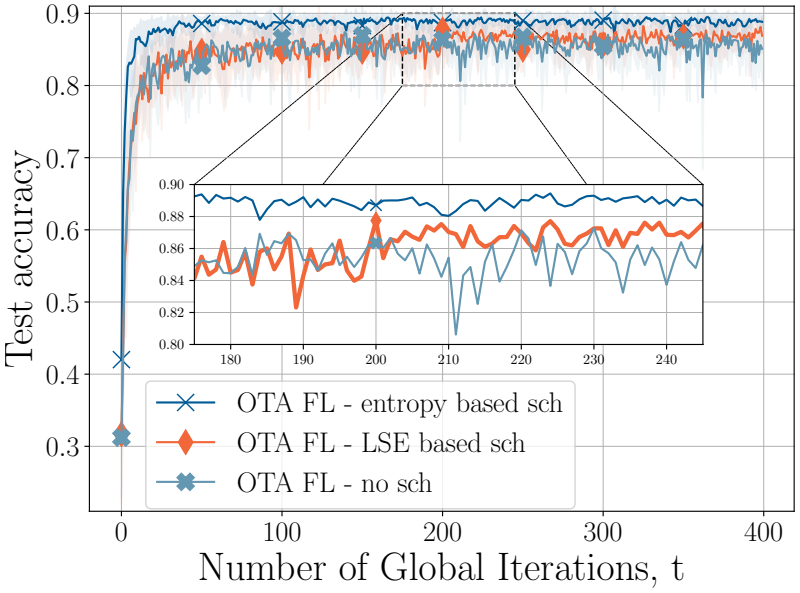
Fig. 3.3 shows the mean test accuracies for data distributions with 1 class and 2 classes per user on the MNIST dataset with $M = 40$, $|\mathcal{B}_m| = 1250$, $p_e^m(t) = 0.25$ for $m \in [M]$ and estimation phases of $T = 100$ and $T = 200$ iterations. In both cases, entropy-based scheduling results in higher and more stable accuracy levels. For the unknown data distribution cases, the PS estimates local user representations after T iterations and groups users into 10 clusters, scheduling one user per cluster for each iteration, similar to [27]. Similar to the previous case, our scheme achieves higher gains in more heterogeneous scenarios. Notably, while estimation and clustering become more difficult in less heterogeneous cases, the proposed scheduling approach still consistently outperforms the baseline without scheduling.

To demonstrate the effectiveness of our proposed method on a different dataset, Fig. 3.4 presents the performance of our scheduling policies applied to the FMNIST dataset. In this setup, we consider $M = 40$ usersusers, each with a local dataset size of $|\mathcal{B}_m| = 1250$, and a fixed energy arrival probability of $p_e^m(t) = 0.25$ for $m \in [M]$. The users are divided into 10 clusters for scheduling, and one user from each cluster is selected per iteration, following the same approach as in the previous case. Similarly to the MNIST case, for known data distributions, the entropy-based scheduling method outperforms the no-scheduling baseline. In the case of unknown data distributions, the global model's performance improves after the estimation phase, approaching the performance achieved in the entropy-based case.

For known data distributions, the entropy-based scheduling method outperforms the no-scheduling baseline approach. For unknown data distributions, the global model's performance improves after the estimation phase, approaching the

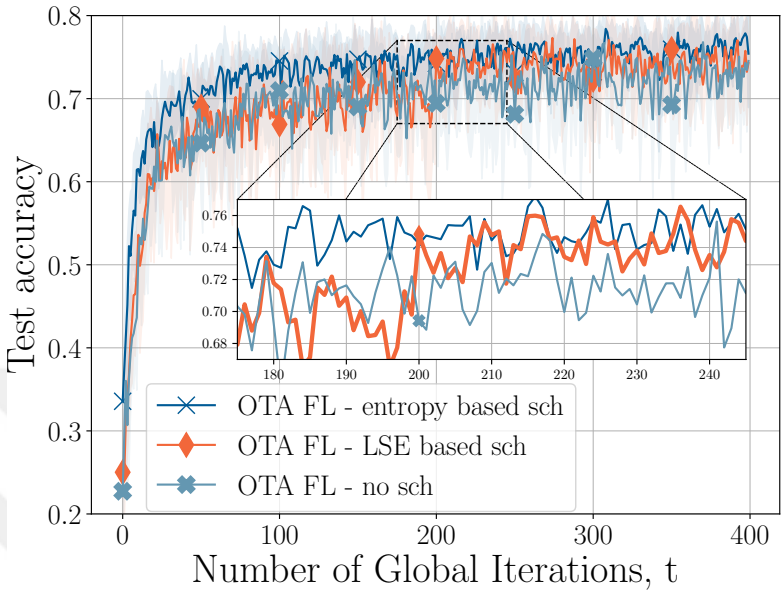


(a) 1 class per user and $T = 100$.



(b) 2 class per user and $T = 200$.

Figure 3.3: The mean test accuracy for MNIST with $M = 40$, $|\mathcal{B}_m| = 1250$ and $p_e^m(t) = 0.25$, $\forall m, t$.



(a) 1 class per user and $T = 200$.

Figure 3.4: The mean test accuracy for FMNIST with $M = 40$, $|\mathcal{B}_m| = 1250$ and $p_e^m(t) = 0.25$, $\forall m, t$.

entropy-based case.

The results show that the entropy-based scheduling, leveraging users' data distributions, improves performance in OTA FL with EH devices over wireless MAC with blind transmitters, achieving higher and more stable accuracy. Moreover, in cases of unknown data distributions, the PS can estimate user representations to schedule diverse users, preserve privacy, reduce redundant updates, and enhance learning. Both entropy- and LSE-based methods promote diversity to approximate full participation and minimize the error in (3.26). This highlights the importance of intelligent user selection in resource-constrained environments, where balancing communication efficiency and statistical performance is crucial. Ultimately, diversity-aware scheduling enables more effective use of the available energy and communication resources, leading to faster convergence and better generalization.

3.5 Chapter Summary

In this chapter, we study user scheduling and representation estimation for OTA FL with EH devices over a wireless fading MAC under highly heterogeneous data. We analyze the convergence of OTA FL and show that scheduling improves learning by minimizing gradient approximation error in the convergence rate. Our framework schedules users based on inter-user relationships and is applicable to both known and unknown data distributions. The entropy-based method leverages label distribution to reduce this error, while least-squares estimation enables PS to infer user relationships for scheduling without compromising privacy. Results show that user scheduling and update estimation significantly improve performance, particularly under high data heterogeneity. A potential future direction is implementing clustered FL based on the user groups derived from our estimation.

Chapter 4

Clustered Federated Learning with Over-the-Air Aggregation for Energy Harvesting Devices

As discussed in Chapters 2 and 3, federated learning (FL) can yield suboptimal results when the data distributions across local clients are highly diverse. To address this limitation, federated multitask learning (FMTL) has been proposed. FMTL aims to provide each client with a personalized model that best fits its unique local data distribution, rather than relying on a single global model for all users.

Clustered federated learning (CFL) is a federated multitask learning framework that leverages the geometric properties of the FL loss surface to partition the client population into clusters with similar and jointly trainable data distributions. Unlike the conventional FL framework, which treats all clients equally and trains a single global model, CFL addresses client heterogeneity by learning specialized models for each cluster, thereby improving personalization and overall performance [68, 69]. In many practical applications, users can be naturally partitioned into clusters based on shared characteristics or interests. For

instance, in recommendation systems, users may belong to different clusters reflecting preferences for specific content categories, such as news topics or product types. Similarly, in healthcare settings, hospitals may form clusters based on patient demographics or treatment protocols. The goal in such CFL setups is to train a separate model for each cluster, tailored to the data distribution of the corresponding user group. Recognizing and exploiting this inherent cluster structure can significantly enhance model performance and personalization. CFL in wireless communication setups is also an emerging research direction, gaining increasing attention due to the challenges posed by data heterogeneity and communication constraints [70, 71, 72].

Training multiple personalized models aimed at each cluster requires separate transmission (e.g., sequential transmission) of local updates from each cluster or the use of multiple parameter servers (PS), which increases either latency or setup cost. Alternatively, in our subsequent work, we consider an over-the-air clustered wireless FL approach that can recover the local updates of each group separately, even with synchronized transmission, using only a single parameter server.

4.1 System Model

We consider a CFL system with heterogeneous data distribution for M energy harvesting devices. In this setup, the goal is to minimize a global loss function collaboratively across H clusters, with each cluster $h \in [H]$ maintaining a personalized distinct model. Each cluster h is composed of a subset of the M devices, where the clusters are mutually exclusive, that is, no device belongs to more than one cluster. For each cluster, the global model parameters $\boldsymbol{\theta}_h \in \mathbb{R}^{2N}$ are optimized. Each model $\boldsymbol{\theta}_h$ is designed to minimize the local loss function corresponding to its cluster h. The global loss function is defined as:

$$F(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_H) = \sum_{h=1}^{H} \sum_{m \in \mathcal{C}_h} \frac{|\mathcal{B}_m|}{B} F_m(\boldsymbol{\theta}_h), \tag{4.1}$$

where $F_m(\boldsymbol{\theta}_h)$ represents the average empirical local loss for the m-th user in cluster h with model parameters $\boldsymbol{\theta}_h$, and $\mathcal{C}_h$ is the set of users in cluster $h \in [H]$. For the m-th user with local dataset $\mathcal{B}_m$, the local loss function is defined as:

$$F_m(\boldsymbol{\theta}_h) = \frac{1}{|\mathcal{B}_m|} \sum_{\boldsymbol{u} \in \mathcal{B}_m} f(\boldsymbol{\theta}_h, \boldsymbol{u}), \tag{4.2}$$

where $f(\boldsymbol{\theta}_h, \boldsymbol{u})$ is the empirical loss corresponding to the u-th data sample in the local dataset $\mathcal{B}_m$. The total number of data samples across all users is denoted by $B = \sum_{m=1}^{M} |\mathcal{B}_m|$.

The loss for cluster h is defined as the expected local loss for all users within that cluster:

$$F_h(\boldsymbol{\theta}_h) := \mathbb{E}_{\xi \sim P_h} \left[f(\boldsymbol{\theta}_h, \xi) \right], \tag{4.3}$$

where P_h is the data distribution for cluster h, and ξ represents a data sample from this distribution.

For the energy harvesting (EH) devices, we consider a Bernoulli energy arrival process, where at each global iteration t, the m-th user harvests one unit of energy with probability $p_e^m(t)$, as discussed in Chapter 3.1. Both local stochastic gradient descent (SGD) computation and update transmission consume one unit of energy per iteration, making stochastic energy availability a critical constraint. However, unlike the setup in the previous chapter, we assume that the harvested energy is immediately consumed in the next iteration. That is, no energy storage or scheduling is performed in this setting. In the context of energy harvesting (EH) devices, the system accounts for the varying energy availability across devices, influencing the local updates and the set of active devices within each cluster. We define $\mathcal{S}_h(t)$ as the set of active users in cluster h at global iteration t. Note that $\mathcal{S}_h(t)$ is a subset of the overall user set in the cluster, i.e., $\mathcal{S}_h(t) \subseteq \mathcal{C}_h$, where $\mathcal{C}_h$ denotes all users assigned to cluster h.

Our approach in the clustering process is to group similar users based on their data or update characteristics, enabling the training of a separate personalized model for each cluster tailored to their observed data pattern. When the users are EH devices, limited energy availability may result in some users not having

sufficient energy to perform local computations and transmissions. In those cases, similar users in the group can make meaningful contributions and help one another to create a more personalized model for that cluster, rather than relying on the standard FL approach. In the standard FL framework, all the clients are treated uniformly and a single global model is learned, which often fails to serve all users optimally due to underlying data heterogeneity.

At each global iteration t, the PS broadcasts the latest global models, $\boldsymbol{\theta}_1(t), \boldsymbol{\theta}_2(t), \ldots, \boldsymbol{\theta}_H(t)$, corresponding to each cluster $h \in [H]$. In response, the selected mobile users (MUs) in each cluster h perform τ iterations of local SGD to minimize their individual local loss functions, $F_m(\boldsymbol{\theta})$, for $m \in \mathcal{S}_h(t)$, Subsequently, the model updates obtained by the mobile users are transmitted back to the PS to contribute to the global learning process, with each cluster providing its local update to improve the group-level global models.

To compute the local model updates in the clustered FL approach, the m-th user in cluster h at the i-th local and t-th global iteration performs the local iterations of SGD with the following update rule:

$$\boldsymbol{\theta}_m^{i+1}(t) = \boldsymbol{\theta}_m^i(t) - \eta_m^i(t) \nabla F_m(\boldsymbol{\theta}_m^i(t), \xi_m^i(t)), \tag{4.4}$$

where $i \in [\tau]$, $\eta_m^i(t)$ is the learning rate, and $\nabla F_m\left(\boldsymbol{\theta}_m^i(t), \xi_m^i(t)\right)$ represents the stochastic gradient estimate for the $\boldsymbol{\theta}_m^i(t)$ and the local mini-batch sample ξ_m^i randomly chosen from the local dataset $\mathcal{B}_m$. After completing the local SGD steps, the m-th user in cluster h computes the model update, which is aimed to be shared with the PS as:

$$\Delta\boldsymbol{\theta}_m(t) = \boldsymbol{\theta}_m^\tau(t) - \boldsymbol{\theta}_m^1(t). \tag{4.5}$$

Once the model updates are computed locally by each user within a cluster, the users simultaneously transmit their updates to the PS with over-the-air (OTA) aggregation. Similar to the previous chapter, the model updates of the users, $\Delta\boldsymbol{\theta}_m^{cx}(t) \in \mathbb{C}^N$, $m \in \mathcal{S}(t)$, are transmitted as complex signals at iteration t,

represented as

$$\Delta\boldsymbol{\theta}_m^{re}(t) \triangleq \left[\Delta\theta_m^1(t), \Delta\theta_m^2(t), \ldots, \Delta\theta_m^N(t)\right]^T, \tag{4.6a}$$

$$\Delta\boldsymbol{\theta}_m^{im}(t) \triangleq \left[\Delta\theta_m^{N+1}(t), \Delta\theta_m^{N+2}(t), \ldots, \Delta\theta_m^{2N}(t)\right]^T, \tag{4.6b}$$

$$\Delta\boldsymbol{\theta}_m^{cx}(t) \triangleq \Delta\boldsymbol{\theta}_m^{re}(t) + j\Delta\boldsymbol{\theta}_m^{im}(t). \tag{4.6c}$$

The PS then combines the received signals from all users in each cluster, forming cluster-specific global updates that are subsequently used to refine the global models for each cluster.

In an ideal setup, where the server can perfectly identify the individual user updates, each model can be updated easily by averaging the updates within the corresponding cluster as:

$$\Delta\boldsymbol{\theta}_h(t) = \frac{1}{|\mathcal{S}_h(t)|} \sum_{m \in \mathcal{S}_h(t)} \Delta\boldsymbol{\theta}_m(t), \tag{4.7}$$

where $h \in [H]$ represents each cluster, and $\mathcal{S}_h(t)$ is the set of users scheduled in cluster h at global iteration t, as mentioned earlier.

After averaging the updates within each cluster, the global model for cluster h can be updated as:

$$\boldsymbol{\theta}_h(t+1) = \boldsymbol{\theta}_h(t) + \Delta\boldsymbol{\theta}_h(t), \tag{4.8}$$

where $\Delta\boldsymbol{\theta}_h(t)$ represents the aggregated update for the model of cluster h, based on the updates from all users in the cluster.

We assume that the mobile users do not have channel state information (CSI), and they transmit their updates to the PS through a fading multiple access channel (MAC) using OTA aggregation. The PS is equipped with K receive antennas to align the received signals, even in the absence of CSI at the transmitters (CSIT), by leveraging aggregated channel state information. The key challenge arises because the server needs to recover the sum of the local updates for each cluster. However, when users send their updates concurrently over the wireless channel, the PS only observes the sum of the local updates from all users. Since the PS only receives the over-the-air aggregated global update, it cannot distinguish the contributions of each cluster, making the recovery of individual

cluster updates challenging. This problem is exacerbated in the absence of CSI at the transmitters, as the users become blind to the channel conditions, making it significantly more difficult for the PS to separate and correctly attribute the aggregated updates to their respective clusters.

A potential solution lies in leveraging the channel knowledge at the server side. By utilizing the CSI that the PS can gather, one can employ minimum mean square error (MMSE) estimation at the receiver side to recover the individual updates from each user. In this case, the PS can use the aggregated received signal and apply the MMSE technique to estimate the individual updates for each cluster, despite the concurrent transmissions. The MMSE approach optimally separates the mixed signals, allowing the server to recover each cluster's local updates accurately by exploiting the signals observed at the different receive antennas. By using MMSE, the server can effectively perform CFL, even without CSI at the users' side. This enables the PS to aggregate the updates from each cluster, recover the individual contributions, and perform the necessary model updates while maintaining the benefits of concurrent transmission over a shared wireless channel.

4.2 Clustered Federated Learning with Over-the-Air Aggregation

Using OTA transmission over a fading MAC, the received signal at the k-th antenna of the PS at iteration t is given as

$$\boldsymbol{y}_{PS,k}(t) = \sum_{m \in \mathcal{S}(t)} \boldsymbol{h}_{m,k}(t) \circ \boldsymbol{x}_m(t) + \boldsymbol{z}_{PS,k}(t), \tag{4.9}$$

where $\mathcal{S}(t)$ is the set of all active users, $\circ$ denotes the element-wise product, $\boldsymbol{x}_m(t)$ is the signal transmitted by the m-th user, and $\boldsymbol{h}_{m,k}(t)$ is the independent and identically distributed (i.i.d.) channel gains from the m-th user to the k-th antenna with complex Gaussian entries $h^n_{m,k}(t) \sim \mathcal{CN}(0, \sigma_h^2)$. Similarly, $z^n_{PS,k}(t)$ denotes the n-th entry of the channel noise, $\boldsymbol{z}_{PS,k}(t)$, which is i.i.d. circularly

symmetric additive white Gaussian noise (AWGN), i.e., it is distributed according to $\mathcal{CN}(0, \sigma_z^2)$.

Equivalently, in the clustered FL setup, the received signal for the k-th antenna at the PS at iteration t can be expressed as a sum over clusters, which is also equivalent to (4.9), with each cluster h transmitting its updates. The received signal becomes:

$$\boldsymbol{y}_{PS,k}(t) = \sum_{h\in[H]} \sum_{m\in\mathcal{S}_h(t)} \boldsymbol{h}_{m,k}(t) \circ \boldsymbol{x}_m(t) + \boldsymbol{z}_{PS,k}(t), \tag{4.10}$$

where the first sum runs over the clusters, and the second sum runs over the users within each cluster $\mathcal{S}_h(t)$ at global iteration t.

Our goal is to design a combining technique at the receivers to combine $\boldsymbol{y}_{PS,k}(t)$ for each cluster $h \in [H]$ to ensure convergence guarantees for the global model of each cluster, and update the global models as in (4.8).

We consider an OTA federated learning setup in which the PS receives aggregated signals over a fading MAC and aims to estimate the model updates for each cluster of users. The core challenge lies in the accuracy of signal recovery under limited CSI. In this context, we investigate different estimation strategies based on the granularity of CSI available at the PS. We first present the case with full per-user CSI, and then extend our analysis to more practical scenarios where only partial CSI, such as per-cluster, is available.

There exists a fundamental trade-off between estimation accuracy and signaling overhead in acquiring CSI for wireless federated learning systems. Full per-user CSI at the PS can be obtained via pilot-based training, where each user transmits known pilot symbols, enabling the PS to estimate individual channel responses. In multi-carrier systems such as OFDM, these pilots can be embedded on designated subcarriers to avoid occupying the entire bandwidth [7, 47]. Assigning distinct pilot subcarriers to different users minimizes interference during the estimation phase, allowing accurate reconstruction of each user's channel, which corresponds to the full user-level CSI case considered in this work. On the other hand, this incurs a high signaling overhead, which may not be feasible

when the channel varies rapidly.

Since acquiring full CSI may be impractical due to large user populations, the PS may instead rely on partial CSI in the form of per-cluster channel information. That is, the sum of the channel coefficients for users in each cluster can be obtained and utilized. This can be accomplished by assigning common pilots in the same set of subcarriers to all the users within a cluster, allowing the PS to recover the effective aggregated channel for the cluster. Such a setting corresponds to what we call the partial cluster-level CSI throughout this work, where only the combined channel response of each cluster is available for model aggregation.

4.2.1 Full User-Level CSI Available at the PS

In this setting, the PS has access to full CSI for each user to each antenna at the PS. That is, it knows the complete channel coefficient array $\boldsymbol{H_f} \in \mathbb{C}^{N\times K\times M_s}$, where N denotes the number of transmitted symbols, K is the number of receive antennas at the PS, and M_s is the total number of active users and defined as $M_s \triangleq |\mathcal{S}(t)|$. With this detailed channel knowledge, the PS can apply estimation and combining methods to recover either per-user updates or aggregate cluster-level updates with high precision. This setup represents an ideal case, enabling the most accurate signal separation, but comes at the cost of requiring extensive CSI estimation, which may not be feasible due to the estimation cost in practical wireless FL systems.

4.2.1.1 MMSE Combining with Full CSI

For the MMSE combining with the full CSI we consider the equivalent channel model for (4.10). Specifically, for the n-th symbol, the channel model is expressed as:

$$\boldsymbol{Y} = \boldsymbol{H}\boldsymbol{X} + \boldsymbol{N}, \tag{4.11}$$

where $\boldsymbol{H} \in \mathbb{C}^{K \times M_s}$ is the channel matrix known at the receiver side, where each element of the matrix corresponds to the $h_{m,k}$ defined in (4.9). Also, $\boldsymbol{Y} \in \mathbb{C}^K$ is the received signal, $\boldsymbol{X} \in \mathbb{C}^{M_s}$ is the transmitted signal, and $\boldsymbol{N} \in \mathbb{C}^K$ is the additive noise, where K is the number of antennas at the server, and M_s is the number of active users at that iteration. Note that for ease of illustration, we omit the iteration index t and symbol index n from the parameters

Given the received signal $\boldsymbol{Y} \in \mathbb{C}^K$ and the full channel matrix $\boldsymbol{H} \in \mathbb{C}^{K \times M_s}$, P) aims to recover the transmitted signal $\boldsymbol{X} \in \mathbb{C}^{M_s}$ using a linear estimator of the form $\hat{\boldsymbol{X}} = \boldsymbol{W}\boldsymbol{Y}$. The optimal estimator minimizes the mean squared error (MSE) between the true and estimated signals:

$$\mathbb{E}\left[\|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{Y}\|_F^2\right]. \tag{4.12}$$

Here, $\boldsymbol{X}$ and $\boldsymbol{Y}$ corresponds to a transmitted and received symbols, while $\boldsymbol{H} \in \mathbb{C}^{K \times M_s}$ denotes the channel matrix associated with the n-th symbol, governing the transformation from user updates to received signals at that index.

Expanding this loss:

$$\begin{aligned}\mathcal{L}(\boldsymbol{W}) &= \mathbb{E}\left[\|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{Y}\|^2\right] = \mathbb{E}\left[(\boldsymbol{X} - \boldsymbol{W}\boldsymbol{Y})^H(\boldsymbol{X} - \boldsymbol{W}\boldsymbol{Y})\right] \\ &= \underbrace{\mathbb{E}[\boldsymbol{X}^H\boldsymbol{X}]}_{\text{A}} - \underbrace{\mathbb{E}[\boldsymbol{X}^H\boldsymbol{W}\boldsymbol{Y}]}_{\text{B}} - \underbrace{\mathbb{E}[\boldsymbol{Y}^H\boldsymbol{W}^H\boldsymbol{X}]}_{\text{B}^*} + \underbrace{\mathbb{E}[\boldsymbol{Y}^H\boldsymbol{W}^H\boldsymbol{W}\boldsymbol{Y}]}_{\text{C}}.\end{aligned} \tag{4.13}$$

The first term, A, is the expected signal energy, which simplifies to $\mathbb{E}[\boldsymbol{X}^H\boldsymbol{X}] = \mathrm{Tr}(\boldsymbol{C_{xx}})$,, where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix, and $\boldsymbol{C_{xx}}$ is the covariance matrix of the transmitted signal. The second term, B, is the cross term, which becomes $\mathbb{E}[\boldsymbol{X}^H\boldsymbol{W}\boldsymbol{Y}] = \mathrm{Tr}\left(\boldsymbol{W}\,\mathbb{E}[\boldsymbol{Y}\boldsymbol{X}^H]\right)$. Using the signal model $\boldsymbol{Y} = \boldsymbol{H}\boldsymbol{X} + \boldsymbol{N}$, the cross-covariance is $\mathbb{E}[\boldsymbol{Y}\boldsymbol{X}^H] = \boldsymbol{H}\,\boldsymbol{C_{xx}}$, which leads to $\mathrm{Tr}(\boldsymbol{W}\boldsymbol{H}\boldsymbol{C_{xx}})$. The final term, C, is the expected energy of the estimation, expressed as $\mathbb{E}[\boldsymbol{Y}^H\boldsymbol{W}^H\boldsymbol{W}\boldsymbol{Y}] = \mathrm{Tr}(\boldsymbol{W}(\boldsymbol{H}\boldsymbol{C_{xx}}\boldsymbol{H}^H + \boldsymbol{C_{nn}})\boldsymbol{W}^H)$, where $\boldsymbol{C_{nn}}$ is the noise covariance matrix.

Putting these components together, the MMSE cost becomes:

$$\mathcal{L}(\boldsymbol{W}) = \mathrm{Tr}\boldsymbol{C_{xx}} - 2\mathrm{Re}\left\{\mathrm{Tr}(\boldsymbol{W}\boldsymbol{H}\boldsymbol{C_{xx}})\right\} + \mathrm{Tr}\left(\boldsymbol{W}(\boldsymbol{H}\boldsymbol{C_{xx}}\boldsymbol{H}^H + \boldsymbol{C_{nn}})\boldsymbol{W}^H\right). \tag{4.14}$$

To minimize this expression with respect to $\boldsymbol{W}$:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{W}} = -2\,\boldsymbol{C_{xx}}\boldsymbol{H}^H + 2\,\boldsymbol{W}(\boldsymbol{H}\boldsymbol{C_{xx}}\boldsymbol{H}^H + \boldsymbol{C_{nn}}). \tag{4.15}$$

Setting the derivative to zero, to minimize the MSE cost function, we have:

$$\boldsymbol{W} = \boldsymbol{C_{xx}}\boldsymbol{H}^H \left(\boldsymbol{H}\boldsymbol{C_{xx}}\boldsymbol{H}^H + \boldsymbol{C_{nn}}\right)^{-1}. \tag{4.16}$$

Thus, the MMSE estimate of $\boldsymbol{X}$ is given by:

$$\hat{\boldsymbol{X}} = \boldsymbol{C_{xx}}\boldsymbol{H}^H \left(\boldsymbol{H}\boldsymbol{C_{xx}}\boldsymbol{H}^H + \boldsymbol{C_{nn}}\right)^{-1} \boldsymbol{Y}. \tag{4.17}$$

This is the MMSE estimator for $\boldsymbol{X}$, given the observation $\boldsymbol{Y}$, full CSI $\boldsymbol{H}$, and known signal and noise statistics.

Once the transmitted updates are estimated as $\hat{\boldsymbol{X}}$, they correspond to the recovered versions of individual user updates $\Delta\hat{\boldsymbol{\theta}}_m(t)$ for each scheduled user $m \in \mathcal{S}(t)$. The PS can then compute the aggregated update for each cluster by averaging the estimated updates of users assigned to cluster h as:

$$\Delta\hat{\boldsymbol{\theta}}_h(t) = \frac{1}{|\mathcal{S}_h(t)|} \sum_{m \in \mathcal{S}_h(t)} \Delta\hat{\boldsymbol{\theta}}_m(t), \tag{4.18}$$

where $h \in [H]$ denotes the cluster index and $\mathcal{S}_h(t)$ is the corresponding set of scheduled users. These aggregated updates are then used to update the cluster-specific global models as:

$$\boldsymbol{\theta}_h(t+1) = \boldsymbol{\theta}_h(t) + \Delta\hat{\boldsymbol{\theta}}_h(t), \tag{4.19}$$

allowing each cluster model to evolve independently based on the updates contributed by its own user group.

4.2.2 Partial Cluster-Level CSI Available at the PS

In this setting, the PS does not have access to individual user-level CSI. Instead, it obtains partial channel knowledge in the form of aggregated CSI at the cluster

level. Specifically, for each cluster $h \in [H]$, the PS is assumed to know the aggregated channel vector obtained by summing the channel vectors of users in that cluster. For the k-th antenna, this can be written as $\sum_{m \in \mathcal{S}_h(t)} \boldsymbol{h}_{m,k}(t)$, where $\mathcal{S}_h(t)$ denotes the set of participating users in cluster h at iteration t. These aggregated vectors are used to construct the cluster-level channel matrix $\boldsymbol{H}_c \in \mathbb{C}^{K \times H}$, where each column corresponds to a cluster. The estimation of such aggregated CSI can be performed by assigning a common pilot signal to all users within a cluster, allowing the PS to capture the superimposed channel response for that cluster in a single measurement. This approach significantly reduces CSI acquisition overhead compared to the full per-user CSI case, but it also limits the resolution of the received signal and the ability to distinguish among users within the same cluster.

4.2.2.1 MMSE Combining with Partial CSI

The received signal still follows the standard linear model introduced in (4.11): $\boldsymbol{Y} = \boldsymbol{H_c X_c} + \boldsymbol{N}$, where $\boldsymbol{Y} \in \mathbb{C}^{K \times 1}$ denotes the received signal across K antennas for symbol n, $\boldsymbol{H_c} \in \mathbb{C}^{K \times H}$ is the aggregated cluster-level channel matrix, $\boldsymbol{X_c} \in \mathbb{C}^{H}$ is the matrix of transmitted cluster updates (each column corresponding to one cluster), $\boldsymbol{N} \in \mathbb{C}^{K \times 1}$ represents additive white Gaussian noise.

Each transmitted cluster signal in $\boldsymbol{X_c}$ is assumed to be the sum of updates from all the users in the corresponding cluster. Since individual user-level recovery is not possible in this case, the PS applies a linear MMSE estimator to recover cluster-level updates.

We also note that the estimated $\hat{\boldsymbol{X}}_c$ corresponds to the cluster-level aggregated updates with size $\hat{\boldsymbol{X}}_c \in \mathbb{C}^{H}$. Each row $\hat{\boldsymbol{x}}_c^{(h)}$ represents the MMSE estimate of the aggregated update from cluster h. This can be seen by replacing $\boldsymbol{H}$ with $\boldsymbol{H}_c$ in (4.17), where $\boldsymbol{A} \in \mathbb{R}^{M \times H}$ is the cluster assignment matrix (each column indicates users belonging to a given cluster), mapping each of the M users to one of the H clusters, and $\boldsymbol{H}_c = \boldsymbol{H}\boldsymbol{A}$ is the resulting cluster-level channel matrix. Note that each column of $\boldsymbol{A}$ has unit norm and they are orthogonal to each other,

with pseudo-inverse of $\boldsymbol{A}$ is equal to $\boldsymbol{A}^{\dagger} = (\boldsymbol{A}^H \boldsymbol{A})^{-1} \boldsymbol{A}^H$. Note that due to the structure of $\boldsymbol{A}$, the h-th row of $\boldsymbol{A}^{\dagger}$ consists of ones corresponding to the users in the h-th cluster, and zeros elsewhere, resulting in $\boldsymbol{X}_c = \boldsymbol{A}^{\dagger} \boldsymbol{X}$ with cluster-summed local updates. Hence, the corresponding input-output relationship can be rewritten as:

$$\boldsymbol{Y} = \boldsymbol{H}_c \boldsymbol{X}_c + \boldsymbol{N} \tag{4.20}$$

$$= \boldsymbol{H} \boldsymbol{A} \boldsymbol{A}^{\dagger} \boldsymbol{X} + \boldsymbol{N} \tag{4.21}$$

$$= \boldsymbol{H} \boldsymbol{X} + \boldsymbol{N}, \tag{4.22}$$

and each element of X_c represents the average of the transmitted updates from the corresponding cluster. Hence, estimating $\boldsymbol{X_c}$ using only cluster-level partial CSI yields the average of the local updates from each cluster. Then, the cluster-level MMSE estimate becomes

$$\hat{\boldsymbol{X}}_c = \boldsymbol{C}_{x_c x_c} \boldsymbol{H}_c^H \left(\boldsymbol{H}_c \boldsymbol{C}_{x_c x_c} \boldsymbol{H}_c^H + \boldsymbol{C}_{nn} \right)^{-1} \boldsymbol{Y}, \tag{4.23}$$

with $\boldsymbol{C}_{x_c x_c} = \boldsymbol{A}^{\dagger} \boldsymbol{C}_{xx} \boldsymbol{A}$ and $\boldsymbol{H}_c = \boldsymbol{H} \boldsymbol{A}$. Solving (4.23) in a similar manner to (4.17), by substituting $\boldsymbol{C}_{xx} = \boldsymbol{C}_{x_c x_c}$ and $\boldsymbol{H} = \boldsymbol{H}_c$, and using the received signal from all antennas $\boldsymbol{Y}$ along with the cluster-level CSI $\boldsymbol{H}_c$, one can directly obtain the desired signal, i.e., the average of the transmitted updates from each cluster, without needing to recover individual user updates. This shows that the cluster-level estimate is simply the cluster-wise sum of the user-level MMSE estimates.

Using the estimate $\hat{\boldsymbol{X}}_{\boldsymbol{c}}$, the PS can construct cluster-specific model updates in the same manner as (4.18) and (4.19). This approach enables the PS to recover the aggregated updates from each cluster even under partial CSI availability, allowing effective model updates for each cluster without requiring individual user-level CSI.

4.2.2.2 Cluster-wise Weighted Combining (CWC) with Partial CSI

Using the received signal defined in (4.10), the PS can isolate the contribution of cluster $h \in [H]$ by applying a channel-weighted combining operation. Specifically, to get the combined signal for cluster h, the PS performs:

$$\boldsymbol{y}_{PS}^{(h)}(t) = \frac{1}{K}\sum_{k=1}^{K}\left(\sum_{m\in\mathcal{S}_h(t)}\boldsymbol{h}_{m,k}(t)\right)^{*}\circ\boldsymbol{y}_{PS,k}(t). \tag{4.24}$$

This operation acts as a beamformer targeting cluster h.

We focus on the n-th symbol of the PS's beamformed signal for cluster h, denoted by $y_{\mathrm{PS}}^{n,(h)}(t)$:

$$y_{PS}^{n,(h)}(t) = \frac{1}{K}\sum_{k=1}^{K}\left(\sum_{m\in\mathcal{S}_h(t)}h_{m,k}^{n}(t)\right)^{*}\cdot y_{PS,k}^{n}(t). \tag{4.25}$$

In this expression, $h_{m,k}^{n}(t)$ is the channel coefficient from user $m \in \mathcal{S}_h(t)$ to antenna k for the n-th symbol, and $y_{PS,k}^{n}(t)$ denotes the received symbol at antenna k. The sum $\sum_{m\in\mathcal{S}_h(t)} h_{m,k}^{n}(t)$ forms a cluster-specific channel signature at antenna k. Averaging across antennas aligns signals from cluster h, while incoherent signals from other clusters are suppressed, resulting in a coherent estimate of the desired cluster update.

The n-th symbol becomes:

$$\begin{aligned}
y_{PS}^{n,(h)}(t) = {} & \underbrace{\sum_{m\in\mathcal{S}_h(t)}\left(\frac{1}{K}\sum_{k=1}^{K}|h_{m,k}^{n}(t)|^2\right)\Delta\theta_{m}^{n,cx}(t)}_{\text{(1) Signal Term}} \\
& + \underbrace{\sum_{m\in\mathcal{S}_h(t)}\sum_{\substack{m'\in\mathcal{S}_h(t)\\ m'\neq m}}\left(\frac{1}{K}\sum_{k=1}^{K}(h_{m,k}^{n}(t))^{*}h_{m',k}^{n}(t)\right)\Delta\theta_{m'}^{n,cx}(t)}_{\text{(2) Intra-Cluster Interference}} \\
& + \underbrace{\sum_{\substack{g\in[H]\\ g\neq h}}\sum_{m\in\mathcal{S}_h(t)}\sum_{m'\in\mathcal{S}_g(t)}\left(\frac{1}{K}\sum_{k=1}^{K}(h_{m,k}^{n}(t))^{*}h_{m',k}^{n}(t)\right)\Delta\theta_{m'}^{n,cx}(t)}_{\text{(3) Inter-Cluster Interference}} \\
& + \underbrace{\sum_{m\in\mathcal{S}_h(t)}\left(\frac{1}{K}\sum_{k=1}^{K}(h_{m,k}^{n}(t))^{*}z_{PS,k}^{n}(t)\right)}_{\text{(4) Noise Term}}
\end{aligned} \tag{4.26}$$

The combined signal $y_{PS}^{n,(h)}(t)$ consists of four main components: (1) the **signal term**, which represents the coherent sum of the intended updates from users in cluster h; (2) **intra-cluster interference**, arising from imperfect alignment among users within the same cluster, which diminishes as the number of antennas $K \to \infty$; (3) **inter-cluster interference**, caused by signal leakage from other clusters due to overlapping channels, which also vanishes under i.i.d. channel assumptions and large K; and (4) the **noise term**, which is the additive Gaussian noise at the receiver, scaled by the conjugated cluster-specific channel signature. Together, these terms determine the quality of the beamformed estimate for cluster h.

Ignoring the interference and noise terms in the combined signal, one can use the signal term to estimate the desired cluster average as

$$\sum_{m \in \mathcal{S}_h(t)} \left(\frac{1}{K} \sum_{k=1}^{K} |h_{m,k}^n(t)|^2 \right) \Delta\theta_m^{n,\mathrm{cx}}(t), \tag{4.27}$$

which aggregates the updates of the users in cluster h with corresponding effective channel gains.

The PS then recovers the n-th component of the update for cluster h as:

$$\Delta\hat{\boldsymbol{\theta}}_h^n(t) = \frac{1}{|\mathcal{S}_h(t)|\sigma_h^2} \operatorname{Re}\left\{ y_{PS}^{n,(h)}(t) \right\}, \tag{4.28}$$

$$\Delta\hat{\boldsymbol{\theta}}_h^{n+N}(t) = \frac{1}{|\mathcal{S}_h(t)|\sigma_h^2} \operatorname{Im}\left\{ y_{PS}^{n,(h)}(t) \right\}, \tag{4.29}$$

where σ_h^2 is the average per-user channel power for cluster h. Finally, the PS updates the model of cluster h by:

$$\boldsymbol{\theta}_h(t+1) = \boldsymbol{\theta}_h(t) + \Delta\hat{\boldsymbol{\theta}}_h(t). \tag{4.30}$$

This combining strategy closely relates to the one introduced in Section 3, particularly in (3.9), where the global model is recovered using the sum of all user channel gains across the entire network. In contrast, the current method focuses on cluster-level combining, utilizing only the sum of the channel gains within a specific cluster. While the core idea remains similar, leveraging conjugated

channel responses for coherent combining, the interference characteristics differ. Specifically, the current approach introduces both intra-cluster interference due to misalignment among users within the same cluster and inter-cluster interference caused by overlapping channels from users in other clusters.

4.3 Numerical Examples

In this section, we assess the effectiveness of our proposed user combining strategies for clustered federated learning across various scenarios. We focus on image classification tasks using the MNIST dataset [64], considering non-independent and identically distributed (non-i.i.d.) data distributions. For MNIST, we employ a single-layer neural network with 784 input neurons and 10 output neurons (total of $2N = 7850$ parameters). Training is conducted using stochastic gradient descent with a learning rate of 0.05, a learning rate scheduler, and a local update period of $\tau = 5$. Each device uses a mini-batch size of $|\xi_m(t)| = 100$ during local training.

We evaluate the three combining methods introduced earlier. In the first setting, MMSE combining with full CSI, the PS has access to complete CSI for each user-to-antenna link, enabling precise recovery of individual user updates. In the second setting, MMSE combining with partial CSI, the PS only has access to aggregated CSI at the cluster level, allowing it to recover combined updates for each cluster. Finally, in the third setting, cluster-wise weighted combining (CWC) with partial CSI, the PS employs a beamforming-like approach using the same aggregated cluster-level CSI to extract cluster-specific updates.

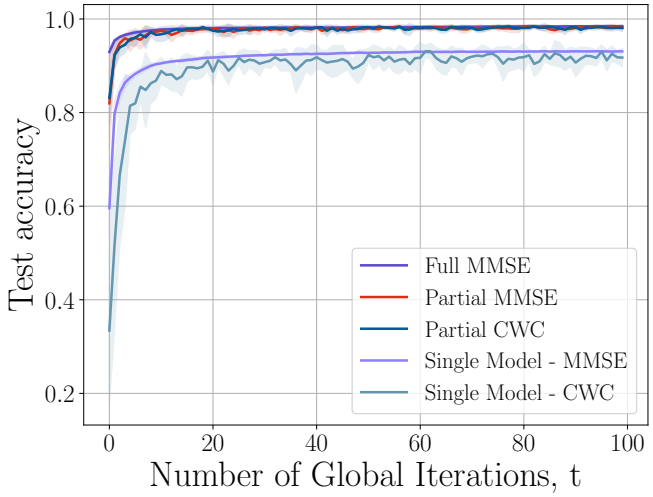
To simulate a highly non-i.i.d. setting, we adopt a label-partitioning strategy where each user is assigned data from only a single class. For the initial experiments, we use a straightforward clustering technique based on class labels: users are grouped according to the class they hold. Specifically, we define three clusters ($H = 3$) to evaluate the effectiveness of the proposed methods in a controlled environment. The first cluster contains users with data from classes 0–2, the second

cluster from classes 3–6, and the third cluster includes users with the remaining classes.

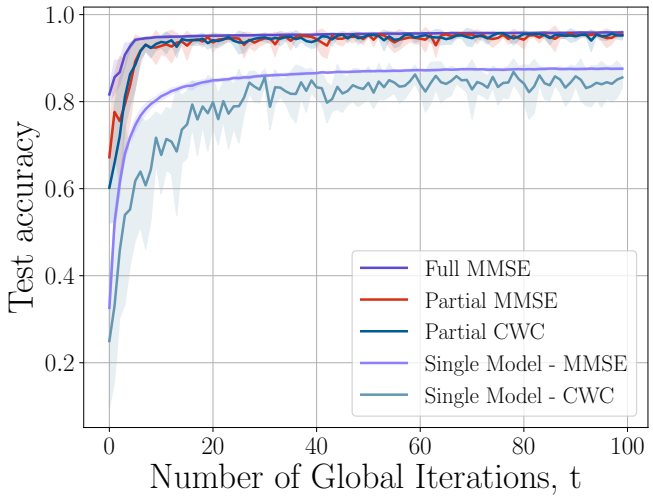
We compare our cluster-specific methods with two standard FL baselines, where a single global model is trained for all users. In the first baseline, a global model is updated using individual user updates recovered under the assumption of full CSI, similar to the approach described in the MMSE Combining with full CSI method. However, instead of maintaining separate models per cluster, all recovered user updates are averaged to update a single global model shared across all users. In the second baseline, we follow the approach outlined in Chapter 3, where the PS combines the received signals from K antennas using the sum of the overall channel gains, again producing one global model for all users.

First, to isolate the combining gain from the adverse impact of energy harvesting constraints, we consider the ideal case in which all users participate in every iteration (i.e. each user has sufficient energy). The results under this full-participation assumption are shown in Fig. 4.1. we set the number of users to $M = 20$, the number of antennas to $K = 20$, the noise variance to $\sigma_z^2 = 10^{-6}$, the fading coefficient variance to $\sigma_h^2 = 1$. We observe that all three cluster-based models, which provide more specialized and personalized global models for users, consistently outperform the single global model approaches. Notably, even without access to full CSI, both the partial MMSE and cluster-wise weighted combining methods still benefit significantly from clustering. This highlights that leveraging clustered structure can yield performance gains that surpass even a full CSI-based single model. As expected, the full CSI MMSE method outperforms both partial CSI-based approaches—MMSE and CWC—highlighting the performance advantage of having complete channel knowledge. This aligns with our intuition, as access to fine-grained CSI enables more accurate update recovery and model aggregation.

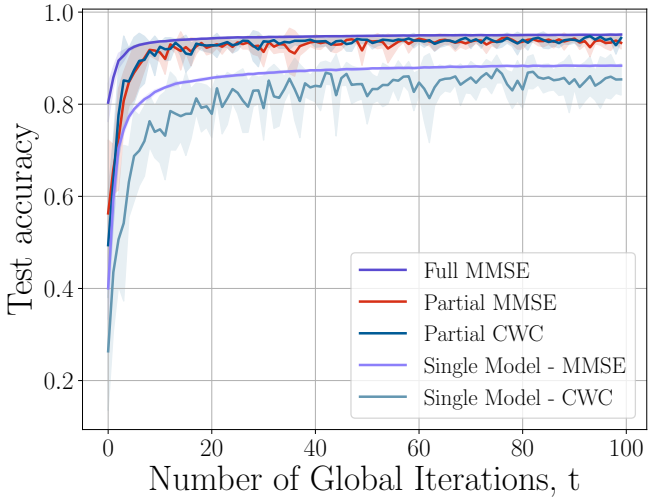
In Fig. 4.2, we compare the performance of our proposed combining methods under different numbers of receiver antennas. Specifically, we evaluate the setups with $K = 5$ and $K = 20$ antennas, keeping the channel model the same as the previous setup, to observe the effect of antenna count on the performance of each



(a) Cluster 1

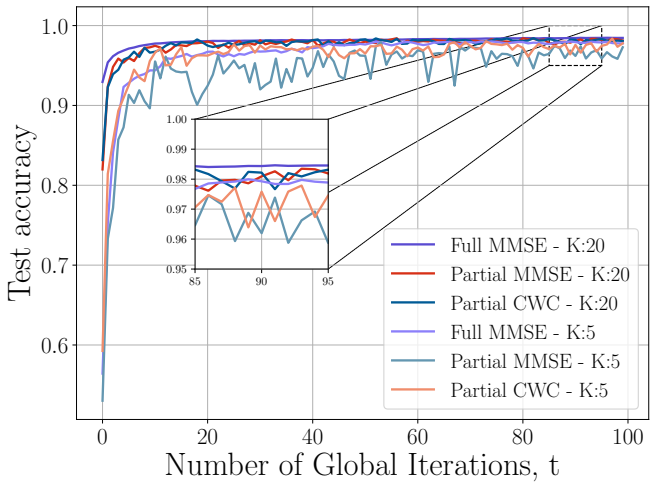


(b) Cluster 2

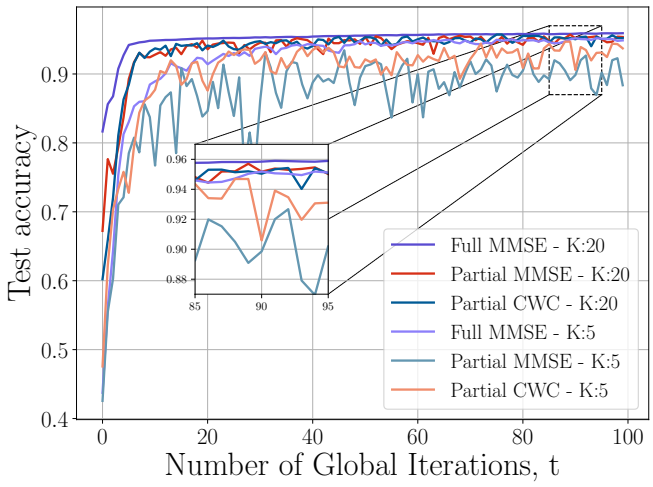


(c) Cluster 3

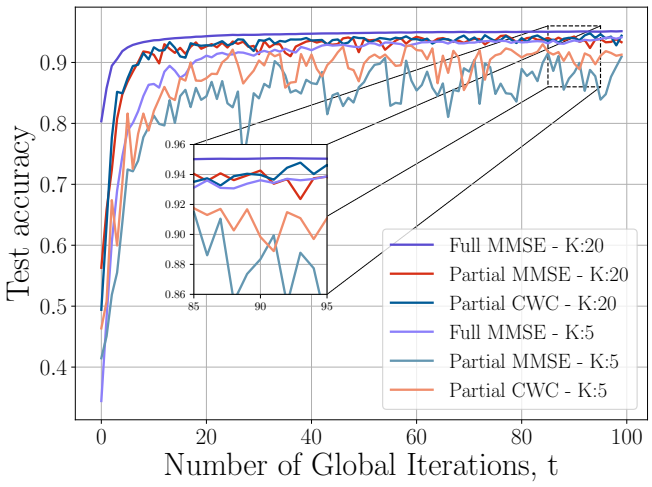
Figure 4.1: The mean test accuracy for CFL with $M = 20$, $H = 3$ and $K = 20$.



(a) Cluster 1



(b) Cluster 2



(c) Cluster 3

Figure 4.2: The mean test accuracy for CFL with $M = 20$, $H = 3$ and $K \in \{5, 20\}$.

combining method. We observe that the full CSI MMSE method exhibits the highest resilience to the reduced number of antennas, with only a slight drop in performance compared to the case with $K = 20$. In contrast, the partial CSI methods (both MMSE and CWC) show noticeably lower and less stable learning performance under limited antenna conditions. Nevertheless, they are still able to reach a relatively acceptable accuracy level, typically around 5–10 percentage points lower than the full CSI MMSE setup with $K = 20$.

Next, we present the performance of our proposed combining methods on the CIFAR-10 dataset [66]. For CIFAR-10, we use a convolutional neural network (CNN) with a total of $2N = 61{,}006$ parameters. The architecture consists of two convolutional layers, each followed by a max pooling operation, and three fully connected layers. The convolutional layers extract spatial features from the RGB input images, while the fully connected layers perform classification over the 10 CIFAR-10 classes. We also incorporate energy harvesting characteristics for each user, assuming a constant energy harvesting probability of $p_e^m(t) = 0.25$, $\forall m, t$. We compare our proposed combining methods with the standard FL setup that uses a single global model. For the highly non-i.i.d. setup, we divide users into 5 categories, where each category is associated with a subset of 4 classes. Each user receives a data distribution drawn from the label subset of its assigned category using a Dirichlet distribution, i.e., $\boldsymbol{p}_m \sim \text{Dir}_4(\beta)$, where $\boldsymbol{p}_m = [p_{m,0}, \cdots, p_{m,3}]$. Accordingly, user m receives p_{m,n_c} portion of its local data from class $n_c \in [4]$, with the Dirichlet concentration parameter set to $\beta = 1$. Note that, for ease of comparison, we only report the average of the cluster-wise accuracies as a single curve.

In Fig. 4.3, we observe that the proposed combining methods successfully demonstrate the feasibility of delivering distinct personalized models to different clusters simultaneously over-the-air. As expected, the full CSI MMSE method outperforms both the partial MMSE and partial CWC approaches, consistent with earlier results. We also observe that the standard FL approach, which uses a single global model to serve all users, performs significantly worse compared to the results obtained with the MNIST dataset in Fig. 4.1. This performance gap is likely due to the increased complexity of the CIFAR-10 dataset, highlighting

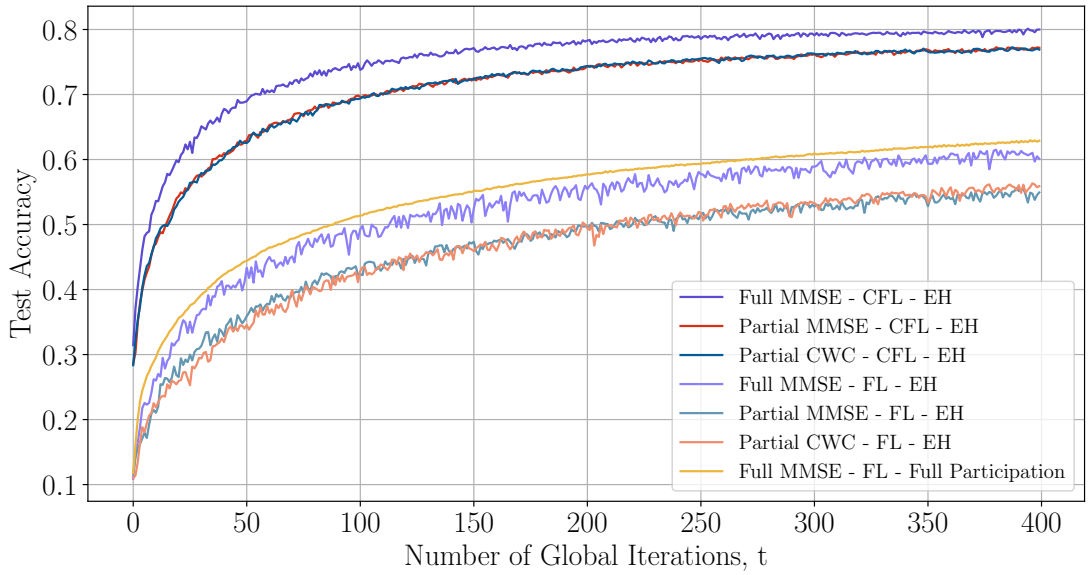


Figure 4.3: The mean test accuracy for CIFAR-10, CFL with $M = 40$, $H = 5$ and $K = 80$.

the importance of serving more personalized models to users in such settings.

The results show that when full CSI is available, the full MMSE method enables precise recovery of individual updates and yields the highest performance and robustness, especially with fewer antennas. In scenarios with only partial CSI, both partial MMSE and partial CWC offer practical alternatives, successfully extracting useful update signals to support clustered model training. While these partial methods exhibit some performance degradation compared to the full MMSE case, they still achieve meaningful improvements over standard FL baselines and enable cluster-level personalization even under limited CSI. Overall, the findings highlight the trade-offs between performance and CSI availability and validate the proposed strategies for OTA FL in heterogeneous wireless environments.

4.4 Chapter Summary

In this chapter, we studied personalized federated learning under OTA aggregation by focusing on clustered model training. We propose multiple combining

strategies at the parameter server tailored to different levels of CSI availability, including full MMSE, partial MMSE, and partial CWC. These methods enable the estimation of either individual or cluster-level model updates over noisy wireless channels. Numerical results show that the proposed approaches can significantly improve learning performance over standard single-model FL baselines, especially in heterogeneous user settings. In particular, the full MMSE method demonstrates strong robustness to limited antenna resources, while the partial CSI methods remain effective in more practical, constrained setups. As a future direction, the clustering mechanism introduced in Chapter 3, based on the cosine similarity of user updates, can be integrated into the CFL framework of this chapter to allow for dynamic and data-driven cluster formation. Moreover, it would be valuable to further investigate the partial CSI scenario where only the aggregate of all channel gains is available at the server.

Chapter 5

Conclusions and Future Directions

We conclude this thesis with a summary of the main contributions and a discussion of potential future research directions.

In this thesis, we have focused on exploring federated learning over wireless communication channels, with particular emphasis on highly heterogeneous setups. The primary motivation behind this work is to enhance the learning performance of federated systems operating under energy harvesting constraints and highly heterogeneous data distributions.

The first part of the thesis proposes a federated learning (FL) framework with a diverse user scheduling strategy, in which the parameter server (PS) selects mobile devices (MDs) based on its knowledge about the users' data distribution characteristics. This information can be either direct, when the PS knows the data distributions of all users, or indirect, where the PS estimates user updates based on previous iterations. Through numerical and experimental analysis, we demonstrate that the proposed methods, entropy-based scheduling for known data distributions and least-squares estimation (LSE)-based scheduling for unknown data distributions, both aiming to select a diverse set of users for

participation at each iteration, achieve higher and more stable performance.

Moreover, in Chapter 4, we propose clustered federated learning (CFL) structures aimed at enabling federated multitask learning, which provides a more robust system against the adverse effects of highly heterogeneous data distributions and stochastic user participation. We propose different combining methods tailored to varying levels of channel state information (CSI) available at the receiver, with the primary goal of training a dedicated model for each cluster. We show that the proposed methods enable the PS to serve different models for different user clusters, allowing for more personalized models while preserving the core benefits of standard FL, such as collaboratively training models without sharing raw data.

There are several promising directions for future research that build upon the ideas developed in this thesis. For instance, the update estimation techniques discussed in Chapters 3 and 4 could be further enhanced by incorporating alternative estimation algorithms. Additionally, a more detailed analysis of inter-cluster and intra-cluster similarity based on the estimated updates and the chosen metrics for user grouping, could lead to improved clustering and scheduling strategies and model personalization. It is also important to note that the underlying relationship between data distribution and the corresponding model updates, leveraged in this work, still needs deeper investigation. While this connection has been acknowledged in the literature, it remains largely underexplored and presents an open area for further research.

The practical implementation and real-world impact of the methods and structures discussed in this thesis also need further investigation. For example, the energy harvesting devices and models used in this work adopt a relatively simple abstraction, focusing primarily on stochastic energy arrivals and resulting user participation. However, more realistic and complex models of energy harvesting, reflecting actual hardware capabilities and environmental conditions, should be considered. The proposed methods should also be evaluated under these more practical settings to assess their robustness and effectiveness.

Another promising area of future research is the use of the proposed methods, or their variants, for enhancing privacy and defending against backdoor attacks in federated learning. For instance, the LSE-based method introduced in Chapter 3.3 could be leveraged to estimate and identify malicious users in the system, allowing the parameter server to exclude them from participation entirely. Similarly, the CFL framework described in Chapter 4 can be used to group together only trusted users, contributing to a more secure and robust global learning process.

Bibliography

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. of the 20th Int. Conf. on Artificial Intelligence and Statistics*, vol. 54, pp. 1273–1282, PMLR, Apr 2017.

[2] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2017.

[3] W. Shi, S. Zhou, and Z. Niu, "Device scheduling with fast convergence for wireless federated learning," in *IEEE International Conference on Communications (ICC)*, (Dublin, Ireland), pp. 1–6, Jul 2020.

[4] H. H. Yang, Z. Liu, T. Q. S. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Transactions on Communications*, vol. 68, pp. 317–333, Sep 2020.

[5] 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 Transactions on Wireless Communications*, vol. 20, pp. 269–283, Oct. 2021.

[6] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Transactions on Wireless Communications*, vol. 19, pp. 2022–2035, Mar. 2020.

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

[8] F. Bagci, B. Tegin, M. Kazemi, and T. M. Duman, "Update estimation and scheduling for over-the-air federated learning with energy harvesting devices," in *IEEE International Conference on Communications Workshops (ICC Workshops)*, (Montreal, Canada), pp. 1435–1440, Jun. 2025.

[9] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. Vincent Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, pp. 1622–1658, Third Quarter 2021.

[10] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data," *arXiv preprint arXiv:1806.00582*, 2018.

[11] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on non-IID data," in *International Conference on Learning Representations (ICLR)*, pp. 1–26, Apr. 2020.

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

[13] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning," in *International conference on machine learning*, vol. 119, pp. 5132–5143, PMLR, Jul. 2020.

[14] S. J. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečný, S. Kumar, and H. B. McMahan, "Adaptive federated optimization," in *International Conference on Learning Representations (ICLR)*, pp. 1–38, May 2021.

[15] Jianyu Wang et al., "A field guide to federated optimization," *arXiv preprint arXiv:2107.06917*, 2021.

[16] Y. Jee Cho, J. Wang, and G. Joshi, "Towards understanding biased client selection in federated learning," in *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, vol. 151 of *Proceedings of Machine Learning Research*, pp. 10351–10375, PMLR, 28–30 Mar 2022.

[17] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *arXiv preprint arXiv:2007.07481*, 2020.

[18] Peter Kairouz et al., "Advances and open problems in federated learning," *arXiv preprint arXiv:1912.04977*, 2021.

[19] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-IID data silos: An experimental study," in *IEEE 38th International Conference on Data Engineering (ICDE)*, (Kuala Lumpur, Malaysia), pp. 965–978, Aug 2022.

[20] Z. Sun, P. Kairouz, A. T. Suresh, and H. B. McMahan, "Can you really backdoor federated learning?," *arXiv preprint arXiv:1911.07963*, 2019.

[21] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, (New York, NY, USA), p. 308–318, Association for Computing Machinery, Aug 2016.

[22] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang, "Learning differentially private recurrent language models," *arXiv preprint arXiv:1710.06963*, 2018.

[23] H. Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J.-y. Sohn, K. Lee, and D. Papailiopoulos, "Attack of the tails: Yes, you really can backdoor federated learning," *Advances in neural information processing systems*, vol. 33, pp. 16070–16084, Dec. 2020.

[24] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *International conference on artificial intelligence and statistics*, pp. 2938–2948, PMLR, Aug. 2020.

[25] X. Gong, Y. Chen, Q. Wang, and W. Kong, "Backdoor attacks and defenses in federated learning: State-of-the-art, taxonomy, and future directions," *IEEE Wireless Communications*, vol. 30, pp. 114–121, Jun. 2022.

[26] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE Symposium on Security and Privacy (SP)*, (San Francisco, CA, USA), pp. 691–706, IEEE, Sep. 2019.

[27] H. Wang, Z. Kaplan, D. Niu, and B. Li, "Optimizing federated learning on non-IID data with reinforcement learning," in *Proc. IEEE Int. Conf. Comput. Commun.*, (Toronto, ON, Canada), pp. 1698–1707, July 2020.

[28] Y. Fraboni, R. Vidal, L. Kameni, and M. Lorenzi, "Clustered sampling: Low-variance and improved representativity for clients selection in federated learning," in *International Conference on Machine Learning*, pp. 3407–3416, PMLR, July 2021.

[29] R. Balakrishnan, T. Li, T. Zhou, N. Himayat, V. Smith, and J. Bilmes, "Diverse client selection for federated learning via submodular maximization," in *International Conference on Learning Representations (ICLR)*, pp. 1–18, Apr. 2022.

[30] A. Lutz, G. Steidl, K. Müller, and W. Samek, "Optimizing federated learning by entropy-based client selection," *arXiv preprint 2411.01240*, 2024.

[31] F. Famá, C. Kalalas, S. Lagen, and P. Dini, "Measuring data similarity for efficient federated learning: A feasibility study," in *2024 International Conference on Computing, Networking and Communications (ICNC)*, (Big Island, HI, USA), pp. 394–400, Jun. 2024.

[32] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-efficient sgd via gradient quantization and encoding," *Advances in neural information processing systems*, vol. 30, 2017.

[33] Y. Lin, S. Han, H. Mao, Y. Wang, and W. J. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," *arXiv preprint arXiv:1712.01887*, 2020.

[34] P. Han, S. Wang, and K. K. Leung, "Adaptive gradient sparsification for efficient federated learning: An online learning approach," in *IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*, (Singapore, Singapore), pp. 300–310, Feb. 2020.

[35] H. H. Yang, A. Arafa, T. Q. Quek, and H. V. Poor, "Age-based scheduling policy for federated learning in mobile edge networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (Barcelona, Spain), pp. 8743–8747, Apr. 2020.

[36] M. M. Amiri, D. Gündüz, S. R. Kulkarni, and H. V. Poor, "Convergence of update aware device scheduling for federated learning at the wireless edge," *IEEE Transactions on Wireless Communications*, vol. 20, pp. 3643–3658, Jan 2021.

[37] H. T. Nguyen, V. Sehwag, S. Hosseinalipour, C. G. Brinton, M. Chiang, and H. Vincent Poor, "Fast-convergent federated learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, pp. 201–218, Nov. 2021.

[38] M. Chen, H. V. Poor, W. Saad, and S. Cui, "Convergence time optimization for federated learning over wireless networks," *IEEE Transactions on Wireless Communications*, vol. 20, pp. 2457–2471, Dec. 2021.

[39] M. M. Amiri, T. M. Duman, D. Gündüz, S. R. Kulkarni, and H. V. Poor, "Blind federated edge learning," *IEEE Transactions on Wireless Communications*, vol. 20, pp. 5129–5143, Aug. 2021.

[40] Y. Sun, S. Zhou, Z. Niu, and D. Gündüz, "Dynamic scheduling for over-the-air federated edge learning with energy constraints," *IEEE Journal on Selected Areas in Communications*, vol. 40, pp. 227–242, Nov. 2022.

[41] A. Bereyhi, A. Vagollari, S. Asaad, R. R. Müller, W. Gerstacker, and H. V. Poor, "Device scheduling in over-the-air federated learning via matching pursuit," *IEEE Transactions on Signal Processing*, vol. 71, pp. 2188–2203, Jun. 2023.

[42] J. Du, B. Jiang, C. Jiang, Y. Shi, and Z. Han, "Gradient and channel aware dynamic scheduling for over-the-air computation in federated edge learning systems," *IEEE Journal on Selected Areas in Communications*, vol. 41, pp. 1035–1050, Feb. 2023.

[43] M. Kim, A. L. Swindlehurst, and D. Park, "Beamforming vector design and device selection in over-the-air federated learning," *IEEE Transactions on Wireless Communications*, vol. 22, pp. 7464–7477, Mar. 2023.

[44] M. M. Amiri, T. M. Duman, and D. Gündüz, "Collaborative machine learning at the wireless edge with blind transmitters," in *2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, (Ottawa, ON, Canada), pp. 1–5, Jan. 2019.

[45] B. Tegin and T. M. Duman, "Blind federated learning at the wireless edge with low-resolution ADC and DAC," *IEEE Transactions on Wireless Communications*, vol. 20, pp. 7786–7798, Jun. 2021.

[46] B. Tegin and T. M. Duman, "Federated learning over time-varying channels," in *IEEE Global Communications Conference (GLOBECOM)*, (Madrid, Spain), pp. 01–06, Dec 2021.

[47] B. Tegin and T. M. Duman, "Federated learning with over-the-air aggregation over time-varying channels," *IEEE Transactions on Wireless Communications*, vol. 22, pp. 5671–5684, Aug. 2023.

[48] O. Aygün, M. Kazemi, D. Gündüz, and T. M. Duman, "Hierarchical over-the-air federated edge learning," in *IEEE International Conference on Communications (ICC)*, (Seoul, Korea), pp. 3376–3381, Jun. 2022.

[49] O. Aygün, M. Kazemi, D. Gündüz, and T. M. Duman, "Over-the-air federated edge learning with hierarchical clustering," *IEEE Transactions on Wireless Communications*, vol. 23, pp. 17856–17871, Dec. 2024.

[50] D. Zhang, M. Xiao, Z. Pang, L. Wang, and H. V. Poor, "IRS assisted federated learning: A broadband over-the-air aggregation approach," *IEEE Transactions on Wireless Communications*, vol. 23, pp. 4069–4082, Jun. 2024.

[51] M. Du, H. Zheng, M. Gao, X. Feng, J. Hu, and Y. Chen, "Integrated sensing, communication, and computation for over-the-air federated learning in 6G wireless networks," *IEEE Internet of Things Journal*, vol. 11, pp. 35551–35567, Aug. 2024.

[52] S. Asaad, P. Wang, and H. Tabassum, "Over-the-air feel with integrated sensing: Joint scheduling and beamforming design," *IEEE Transactions on Wireless Communications*, vol. 24, pp. 3273–3288, Jan. 2025.

[53] S. Ulukus, A. Yener, E. Erkip, O. Simeone, M. Zorzi, P. Grover, and K. Huang, "Energy harvesting wireless communications: A review of recent advances," *IEEE Journal on Selected Areas in Communications*, vol. 33, pp. 360–381, Mar. 2015.

[54] K. Tutuncuoglu, O. Ozel, A. Yener, and S. Ulukus, "The binary energy harvesting channel with a unit-sized battery," *IEEE Transactions Information Theory*, vol. 63, pp. 4240–4256, July 2017.

[55] L. Zeng, D. Wen, G. Zhu, C. You, Q. Chen, and Y. Shi, "Federated learning with energy harvesting devices," *IEEE Transactions on Green Communications and Networking*, vol. 8, pp. 190–204, Aug. 2024.

[56] C. Shen, J. Yang, and J. Xu, "On federated learning with energy harvesting clients," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (Singapore, Singapore), pp. 8657–8661, Apr. 2022.

[57] R. Hamdi, M. Chen, A. B. Said, M. Qaraqe, and H. V. Poor, "Federated learning over energy harvesting wireless networks," *IEEE Internet of Things Journal*, vol. 9, pp. 92–103, Jun. 2022.

[58] C. Chen, Y.-H. Chiang, H. Lin, J. C. Lui, and Y. Ji, "Joint client selection and receive beamforming for over-the-air federated learning with energy harvesting," *IEEE Open Journal of the Communications Society*, vol. 4, pp. 1127–1140, May. 2023.

[59] O. Aygün, M. Kazemi, D. Gündüz, and T. M. Duman, "Over-the-air federated learning with energy harvesting devices," in *IEEE Global Communications Conference (GLOBECOM)*, (Rio de Janeiro, Brazil), pp. 1942–1947, Dec 2022.

[60] Q. An, Y. Zhou, Z. Wang, H. Shan, Y. Shi, and M. Bennis, "Online optimization for over-the-air federated learning with energy harvesting," *IEEE Transactions on Wireless Communications*, vol. 23, pp. 7291–7306, July 2024.

[61] K. Zhang and X. Cao, "Federated learning with energy harvesting devices: An MDP framework," *arXiv preprint arXiv:2405.10513*, 2024.

[62] X. Zhou, W. ZHANG, Z. Chen, S. DIAO, and T. Zhang, "Efficient neural network training via forward and backward propagation sparsification," in *Advances in Neural Information Processing Systems* (A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, eds.), 2021.

[63] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: Survey and implications," *IEEE Commun. Surveys Tuts.*, vol. 13, pp. 443–461, Third Quarter 2011.

[64] Y. LeCun, "The MNIST database of handwritten digits," *http://yann.lecun.com/exdb/mnist/*, 1998.

[65] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[66] A. Krizhevsky, "Learning multiple layers of features from tiny images," M.S. thesis, University of Toronto, Department of Computer Science, Toronto, ON, Canada, 2009.

[67] D. A. E. Acar, Y. Zhao, R. Matas, M. Mattina, P. Whatmough, and V. Saligrama, "Federated learning based on dynamic regularization," in *International Conference on Learning Representations (ICLR)*, p. 1–36, May 2021.

[68] F. Sattler, K.-R. Müller, and W. Samek, "Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, pp. 3710–3722, Aug. 2021.

[69] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," *IEEE Transactions on Information Theory*, vol. 68, pp. 8076–8091, Dec. 2022.

[70] H. U. Sami and B. Güler, "Over-the-air clustered federated learning," *IEEE Transactions on Wireless Communications*, vol. 23, pp. 7877–7893, July 2024.

[71] Y. Lin, K. Wang, and Z. Ding, "Rethinking clustered federated learning in noma enhanced wireless networks," *IEEE Transactions on Wireless Communications*, vol. 23, pp. 16875–16890, Nov. 2024.

[72] Z. Li, Z. Chen, T. Q. S. Quek, and H. H. Yang, "Personalized federated learning over the air," *IEEE Transactions on Wireless Communications*, pp. 1–1, Jun. 2025.