Participation Strategies in Federated Learning with Heterogeneous Client Communications

By
Shubham Agrawal

Under the Supervision of:
Dr. Bapi Chatterjee

Submitted in partial fulfillment of the requirements for the degree of
Master of Technology

to

Indraprastha Institute of Information Technology Delhi

June, 2024

Certificate

This is to certify that the thesis titled "**Participation Strategies in Federated Learning with Heterogeneous Client Communications**," being submitted by Shubham Agrawal to the Indraprastha Institute of Information Technology Delhi, for the award of the Master of Technology, is an original research work carried out by him under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

June, 2024

Dr. Bapi Chaterjee
Department of Computer Science and Engineering
Indraprastha Institute of Information Technology Delhi
New Delhi, 110020

Acknowledgements

I would like to extend my deepest gratitude to my advisor, Dr. Bapi Chaterjee, for their invaluable guidance, support, and expertise throughout this journey. Their dedication to teaching and commitment to excellence have been a profound source of inspiration. Dr. Chaterjee's mentorship has significantly shaped my research skills and intellectual development, and I am incredibly fortunate to have had the privilege of learning from them.

My sincere thanks also go to my lab mates for their camaraderie, collaboration, and engaging discussions. Their diverse viewpoints, unwavering support, and willingness to share their knowledge have greatly enriched my research experience.

Additionally, I want to express my heartfelt appreciation to my friends and family who have consistently provided encouragement and motivation.

I also wish to acknowledge the broader research community for their contributions to the field. The wealth of knowledge and resources shared through publications, conferences, and online forums have been crucial in shaping my understanding and advancing my research.

Last but not least, I am deeply grateful to Dr. Malcolm Egan for extending his guidance throughout my work. He was the go-to person for any technical assistance along with Dr. Bapi, and without his involvement, this work wouldn't have been a success.

Abstract

In cross-device federated learning (FL) [1], a machine learning model is developed by leveraging communication between a central server and numerous edge device clients. In practical scenarios, the ability of these edge devices to consistently relay updates can vary significantly. Consequently, some devices may communicate more frequently than others. This paper explores strategies for forming groups of active clients to enhance the accuracy of the server model. Specifically, we examine the case where these groups are organized and activated cyclically. We address the open question of whether leveraging the heterogeneity in communication capabilities by allowing certain clients to be part of multiple groups is beneficial. Our theoretical convergence analysis and experimental results show that enabling clients to join multiple groups significantly improves accuracy. We also show empirically that utilizing local data partitioning does not cause significant performance loss, while it can have huge benefits like reducing local computation and power utilisation as well as reduced training times. The proposed strategies have the potential to boost the effectiveness of cross-device FL in environments with heterogeneous communication capabilities.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Federated Learning (FL) has emerged as a revolutionary paradigm, enabling a myriad of devices to collaboratively learn a shared prediction model while keeping the training data decentralized, private, and secure. This learning framework, characterized by its distributed nature, hinges on the orchestration of a central server that aggregates model updates from clients that have trained models on local data. Such a structure ensures that sensitive information remains on the clients' devices, thus aligning with the principles of privacy by design and minimizing potential attack surfaces [2, 3]. Originally conceptualized for mobile devices (McMahan et al., 2017), FL's applicability has seen an expansive growth across domains, from healthcare to finance, due to its inherent capability to learn from diverse and distributed datasets without compromising user privacy.

The quintessential process in FL involves clients performing local computations on their private data and the server aggregating these updates to enhance the global model. However, the conventional approach assumes either complete or random participation of clients in each round, which may not align with practical scenarios where clients are available only under certain conditions, such as when devices are idle or connected to a non-metered network [4]. Moreover, the inherent non-IID nature of data across clients presents ad- ditional challenges, potentially leading to skewed model training and biased global models [5, 6].

Client devices typically connect to the server via wireless communication links, resulting in highly varied communication quality. This variability arises from the differences in the devices themselves and their locations. For instance, a client device situated near the server can transmit data more reliably and with lower power consumption.

The interaction between numerous client devices and the server exemplifies massive multiple access. Consequently, simultaneous transmissions from multiple active devices may interfere with each other, reducing the chances that updates are successfully received by the server. Therefore, it is often advantageous to schedule transmissions to maximize the number of successful updates. This scheduling involves grouping clients, with each group becoming active in a cyclic manner.

1.1 Motivation

Imagine a scenario where a fleet of autonomous drones is being trained to recognize various objects in real-time. These drones are scattered across a vast area with differing levels of connectivity and computational capabilities. Some drones, located closer to communication hubs, can frequently send and receive updates, while others, in more remote areas, struggle with intermittent connections. In such a dynamic and heterogeneous environment, ensuring that the machine learning model used by these drones is accurate and up-to-date becomes a significant challenge. This intuitive example highlights the complexities and the need for effective strategies to manage and optimize communication between a central server and numerous, varied edge devices.

Addressing the heterogeneity in communication capabilities is crucial for the broader adoption and success of federated learning (FL) in real-world applications. With the proliferation of IoT devices, smart sensors, and mobile devices, FL presents an opportunity to harness distributed data while preserving privacy. However, the variability in devices' communication abilities can lead to inefficiencies and degraded model performance if not properly managed. Developing strategies that allow for effective partial participation of clients, leveraging their varying communication capacities, can substantially enhance model accuracy and reliability. Tackling this problem not only pushes the boundaries of FL but also ensures that the technology can be robustly applied in diverse and challenging environments, from smart cities to remote monitoring systems.

1.2 Contribution

In this paper, we explore cyclic overlapping participation strategies where clients can be active in multiple groups. We address two primary questions: (Q1) Should clients compute updates based on their entire data set when they are active in multiple groups? (Q2) Given a data-splitting strategy, how many groups should clients be associated with?

To tackle (Q1), we conducted an experimental study, revealing that the performance between updates based on data splitting and the full data is comparable, with a maximum difference of 1.5% over the experiments. This suggests data splitting is a desirable strategy to reduce computational requirements at the clients

For (Q2), we analyzed the convergence of FedAvg with cyclic participation, where clients can join multiple groups. Guided by our findings for (Q1), we focused on client updates derived from subsets of their local data, varying by the active group. Our analysis shows a clear relationship with the partial participation strategy. Specifically, we found:

- Expanding group size by adding clients also present in other groups reduces performance.
- Increasing the number of groups and allowing clients to participate in multiple groups can enhance performance.

We validated our analysis through additional experiments, which clearly demonstrate the potential benefits of assigning clients to multiple groups. Notably, we observed gains up to 4% in test accuracy for the standard image classification tasks on the CIFAR10/100 dataset using Resnet18 model.

Chapter 2

Related Work

In the realm of federated learning (FL), the commonly deployed Federated Averaging (FedAvg)[1] and its derivatives [7, 8] have been the cornerstone of optimization strategies. They traditionally rely on either full or uniform partial client participation. Yet, real-world applications reflect a scenario where client availability adheres to a cyclic pattern, driven by local conditions such as device battery life, network access, and privacy considerations. One study [9] offers the first theoretical exploration of FedAvg's convergence with cyclic client participation, suggesting that under certain conditions, cyclic participation could enhance convergence rates compared to the traditional uniform approach. Another investigation delves into block-cyclic structures within convex SGD updates [5], common in federated settings with temporally varied device characteristics. The study reveals that while a block-cyclic pattern may impede SGD performance, employing a pluralistic averaging method could circumvent these issues, allowing for performance on par with non-cyclic, i.i.d. sampling. A further exploration [10] considers the inherent data challenges in FL, where unbalanced and non-i.i.d. data exhibit a block-cyclic distribution across clients, leading to biases in the global model. To mitigate this, the study proposes their versions of pluralistic models, demonstrating improved accuracy and robustness over conventional methods.

One body of work [11] has shifted the perspective from a block-cyclic model to a more fluid one, depicting client data availability as a mixture of distributions varying between different times of the day, such as daytime and nighttime. This approach, encapsulated in the Federated Expectation-Maximization algorithm with Temporal priors (FedTEM), acknowledges the dynamic nature of client activity and adapts the training process to these shifts. Parallel to this, the concept of multi-server federated learning [12] has been explored as a means to alleviate communication constraints inherent in single-server setups. In scenarios where regional servers' coverage areas overlap, clients in these zones benefit from an enriched model obtained from the amalgamation of regional models.

A lot of work has been proposed to tackle statistical heterogeneity [13, 14, 15] and system heterogeneity [16, 17, 18]. However, the explicit consideration of communication heterogeneity between clients makes this work stand apart—an aspect that previous studies have not incorporated into their analyses. By focusing on tackling varying client communication capabilities, our research aims to bridge the gap and provide deeper insights into the effects of client overlap on the convergence and efficacy of federated learning models.

Chapter 3

Proposed Methodology

3.1 Problem Formulation

Consider a federated learning system with M clients. Each client $m \in [M]$ has access to a local dataset $\mathcal{B}_m \subset Xi$ with $\mathcal{B}_m \cap \mathcal{B}_{m'} = \emptyset,\ m \neq m'$, and is associated with the local empirical risk

$$F_m(\mathbf{w}) = \frac{1}{|\mathcal{B}_m|} \sum_{\xi \in \mathcal{B}_m} \ell(\mathbf{w}, \xi),\ \mathbf{w} \in \mathbb{R}^d, \tag{3.1}$$

where $\ell(\mathbf{w}, \xi)$ is the local loss function for the model $\mathbf{w} \in \mathbb{R}^d$ and data sample ξ. We assume that each client has a common loss function. The optimization task for the system is then given by

$$\min_{\mathbf{w} \in \mathbb{R}^d} F(\mathbf{w}) := \frac{1}{M} \sum_{m=1}^{M} F_m(\mathbf{w}). \tag{3.2}$$

Optimization proceeds by distinct communication rounds, with each round corresponding to a different set of active clients. We consider the scenario where clients are associated with one or more groups $\sigma(1), \ldots, \sigma(K)$. The groups are activated in a cyclic fashion: in communication round $r = 1, 2, \ldots$, group $\sigma(r \mod K)$ is active.

Client $m \in [M]$ may be associated with one or more groups. In the case client m is associated with more than one group, only a portion of its data set is accessible in each communication round. In particular, when client m is active in group $\sigma(k),\ k \in \{j : m \in \sigma(j)\}$, it has access to the data set $\mathcal{B}_{m,k} \subset \mathcal{B}_m$ satisfying

$$\begin{aligned} &\bigcup_{j : m \in \sigma(j)} \mathcal{B}_{m,j} = \mathcal{B}_m, \\ &\mathcal{B}_{m,k} \cap \mathcal{B}_{m,k'} = \emptyset,\ k \neq k',\ k, k' \in \{j : m \in \sigma(j)\}. \end{aligned} \tag{3.3}$$

We denote the local risk of client m in group $k \in \{j : m \in \sigma(j)\}$ by

$$F_{m,k}(\mathbf{w}) = \frac{1}{|\mathcal{B}_{m,k}|} \sum_{\xi \in \mathcal{B}_{m,k}} \ell(\mathbf{w}, \xi),\ \mathbf{w} \in \mathbb{R}^d. \tag{3.4}$$

Algorithm 1 Federated Learning with Overlapping Client Groups

Input: Global Model $w^{(1,0)}$, total cycles I, total groups K
Output: Global Model $w^{(I+1,0)}$
Create K groups of clients by selecting $\lfloor \frac{M}{K} \rfloor$ distinct clients per group and sample T clients from group 1.
Cycle = GenerateCycle(M, K, T)
for $i \in [I]$ **do**
  for $k \in |Cycle|$ **do**
    Initialise clients of the group with a subset of data $\mathcal{B}_{m,k} \subseteq \mathcal{B}_m$
    Send global model $w^{(i,k-1)}$ to clients in $\sigma(k)$
    for each client $m \in \sigma(k)$ in parallel **do**
      $w_{m,k}^{(i+1,0)} \leftarrow$ LocalUpdate$(m, w^{(i,k-1)})$
    end for
    $w^{(i,k)} \leftarrow \frac{1}{|\sigma(k)|} \sum_{m \in \sigma(k)} w_{m,k}^{(i+1,0)}$

    if $k == |Cycle|$ **then**
      $w^{(i+1,0)} = w^{(i,k)}$
      $k \leftarrow 0$
      $i \leftarrow i + 1$
    end if
  end for
end for

Function GENERATECYCLE(M, K, T)
  if $K \leq 0$ **or** $K > M$ **then**
    return []
  $b \leftarrow \lfloor M/K \rfloor$
  $groups \leftarrow$ Create b non-overlapping groups of size K each
  $common\_clients \leftarrow$ randomly sample T clients from groups[0]
  if $CycleStructure = MoreClientsPerGroup$ **then**
    for $i \in [1, total\_groups)$ **do**
      Add common_clients to groups[i]
  return $groups$
  else if $CycleStructure = MoreGroups$ **then**
  $reserve \leftarrow []$
  for $i \in [1, total\_groups)$ **do**
    $shifted\_clients \leftarrow$ randomly sample T clients from groups[i]
    $groups[i] \leftarrow common\_clients \cup (groups[i] \setminus shifted\_clients)$
    $reserve = reserve \cup shifted\_clients$
  while $|reserve| > 0$ **do**
    Select a subset from $reserve$ of size $\min(|reserve|, b - T)$ to form new_group
    $reserve \leftarrow reserve \setminus new\_group$
    $new\_group \leftarrow new\_group \cup common\_clients$
    $groups \leftarrow groups \cup \{new\_group\}$
  return $groups$

Notation	Description
M	Number of clients in the federated learning system
$m \in [M]$	Index of a client
$\mathcal{B}_m$	Local dataset of client m
$F_m(\mathbf{w})$	Local empirical risk of client m
$\ell(\mathbf{w}, \xi)$	Local loss function for model $\mathbf{w}$ and data sample ξ
$\mathbf{w} \in \mathbb{R}^d$	Model parameter vector
$F(\mathbf{w})$	Global objective function to minimize
$\sigma(1), \ldots, \sigma(K)$	Groups of clients
r	Communication round
$\mathcal{B}_{m,k}$	Subset of the local dataset accessible by client m in group $\sigma(k)$
$F_{m,k}(\mathbf{w})$	Local risk of client m in group k
$n = r \mod K$	Current active group in communication round r
$\mathbf{w}^{(i,n-1)}$	Current global iterate at the beginning of communication round r
$\mathbf{w}_m^{(i,n)}$	Local iterate of client m after local update
$\mathbf{w}^{(i,n)}$	Updated global iterate after server update

Table 3.1: Notation Table

As a consequence, we have

$$F(\mathbf{w}) = \frac{1}{M} \sum_{k=1}^{K} \sum_{m \in \sigma(k)} \frac{|\mathcal{B}_{m,k}|}{|\mathcal{B}_m|} F_{m,k}(\mathbf{w}). \tag{3.5}$$

Let $n = r \mod K$ and $r = (i-1)K + n$, where i denotes the cycle. Communication round r then consists of the following steps (additional details are provided in Alg. 1):

(i) Client local update: client $m \in \sigma(n)$ receives the current global iterate from the server $\mathbf{w}^{(i,n-1)}$, where $\mathbf{w}^{(i,0)} = \mathbf{w}^{(i-1,K)}$. The local iterate is then updated via FedAvg based on the full local gradient on the active data set (Option 1) or via local training with shuffled data (Option 2).

(ii) Server update: the server receives local iterates from each client in group $\sigma(n)$ and computes the updated global iterate

$$\mathbf{w}^{(i,n)} = \frac{1}{|\sigma(n)|} \sum_{m \in \sigma(n)} \mathbf{w}_m^{(i,n)}. \tag{3.6}$$

A summary of the algorithm is provided in Algorithm 1. The notations used are summarised in Table 3.1.

3.2 Data Partitioning among Clients

Data partitioning is a critical step in FL, as it determines how data is distributed among clients. In our methodology, we use the Dirichlet distribution to partition the dataset among clients. This approach allows for flexible and controllable data heterogeneity, which is essential for simulating real-world scenarios.

Dirichlet Distribution Implementation: We apply the Dirichlet distribution to allocate data samples to clients, ensuring that each client receives a unique subset of the data. The parameter alpha controls the degree of heterogeneity, with lower values resulting in more heterogeneous data distributions. Data partitioning is implemented by processing both training and testing datasets, ensuring balanced and representative data splits across clients.

3.3 Data Splitting for Multi-group Participation

When clients participate in multiple groups, it is crucial to manage their data effectively to prevent performance degradation. We propose a data splitting strategy where each client splits its data into subsets corresponding to the groups they participate in.

Strategy for Data Splitting: Clients divide their local dataset into non-overlapping subsets, each associated with a specific group. This ensures that clients do not train on the same data multiple times within a cycle, maintaining data diversity and reducing overfitting. Our experiments show that this approach yields comparable or improved performance compared to training on the entire dataset.

3.4 Simulation of Group-Cyclic Participation:

The real-life setting of our work involves group-cyclic client participation, which was simulated for the purpose of our experiments. This setup ensures that clients participate in different groups cyclically over multiple communication rounds.

Clients are scheduled to participate in a cyclic fashion, joining different groups in each cycle. This approach replicates real-world scenarios where clients may not be consistently available, enhancing the robustness of the FL framework.

Chapter 4

Theoretical Analysis of Partial Participation Strategies

In this Chapter, we analyze the impact of partial participation strategies. To obtain a clear relationship between the performance of the federated learning system and the participation strategies, we focus on federated gradient descent (Option Local GD in Alg. 1). We verify our theoretical conclusions with numerical experiments in Sec. **??** based on local training with minibatches, widely used in the context of deep neural networks.

4.1 Convergence Theorem

In order to analyze the convergence of FedAvg, we impose the following standard assumptions.

Assumption 1. *The local objective functions* $F_{m,k},\ k \in \{j : m \in \sigma(j)\}$ *are* L*-smooth. Moreover, there exists a constant* $C > 0$ *such that*

$$\|\nabla \ell(\mathbf{w}, \xi)\| \leq C, \quad \mathbf{w} \in \mathbb{R}^d,\ \xi \in \mathcal{B}_m,\ m = 1, \ldots, M. \tag{4.1}$$

Assumption 2. *For some* $\mu > 0$*, the global objective* F *satisfies*

$$\frac{1}{2}\|\nabla F(\mathbf{w})\|^2 \geq \mu(F(\mathbf{w}) - \min_{\mathbf{w}' \in \mathbb{R}^d} F(\mathbf{w}')),\ \forall \mathbf{w} \in \mathbb{R}^d. \tag{4.2}$$

Assumption 3. *There exists a constant* $\alpha \geq 0$ *such that*

$$\left\| \frac{1}{|\sigma(k)|} \sum_{m \in \sigma(k)} \nabla F_m(\mathbf{w}) - \nabla F(\mathbf{w}) \right\| \leq \alpha,\ \forall \mathbf{w} \in \mathbb{R}^d,\ k = 1, \ldots, K. \tag{4.3}$$

We note that that, with the exception of bounded gradients in Assumption 1, the same assumptions are also considered in [**?**].

Theorem 1. *Let Assumptions 1, 2, and 3 hold. Then, for sufficiently small step-size* $\eta > 0$*,*

$$F(\mathbf{w}^{(K,0)}) - F^* \leq (F(\mathbf{w}^{(0,0)}) - F^*)(1 - \eta M \mu)^K + \frac{1}{M\mu}\left(U_1^2/2 + L\eta^3 V_1\right), \tag{4.4}$$

where

$$U_1 = \sum_{k=1}^{K}\sum_{j=1}^{k-1}\left(L\sum_{m\in\sigma(k)}\frac{|\mathcal{B}_{m,k}|}{|\mathcal{B}_m|}\right)\left(\prod_{j'=j+1}^{k-1}\left(1+\eta L\sum_{m\in\sigma(j')}\frac{|\mathcal{B}_{m,j'}|}{|\mathcal{B}_m|}\right)\right)$$
$$\cdot\left(C\sum_{m\in\sigma(j)}\left(1-\frac{|\mathcal{B}_{m,j}|}{|\mathcal{B}_m|}\right)+|\sigma(j)|\alpha\right), \tag{4.5}$$

and

$$V_1 = K\sum_{k=1}^{K}(k-1)\sum_{j=1}^{k-1}L^2\left(\sum_{m\in\sigma(k)}\frac{|\mathcal{B}_{m,k}|}{|\mathcal{B}_m|}\right)^2\left(\prod_{j'=j+1}^{k-1}\left(1+\eta L\sum_{m\in\sigma(j')}\frac{|\mathcal{B}_{m,j'}|}{|\mathcal{B}_m|}\right)^2\right)$$
$$\cdot\left(2|\sigma(j)|\sum_{m\in\sigma(j)}\left(1-\frac{|\mathcal{B}_{m,j}|}{|\mathcal{B}_m|}\right)^2+4|\sigma(j)|^2\alpha^2\right). \tag{4.6}$$

Before presenting the proof of Theorem 1, we establish preliminary lemmas bounding the change in the server iterate over one cycle. The proofs of the lemmas are available in the appendix of the supplementary material.

Lemma 1. Suppose Assumption 1 holds and iterates $\mathbf{w}^{(i,k)}$ are obtained via Algorithm 2 (Option LocalGD). Then,

$$\mathbf{w}^{(i,K)} - \mathbf{w}^{(i,0)} = -\eta\sum_{k=1}^{K}q^{(i,k)} + \eta^2\mathbf{r}^{(i,0)}, \tag{4.7}$$

$$\mathbf{r}^{(i,0)} = \sum_{k=1}^{K}\sum_{j=1}^{k-1}S^{(i,k)}\left(\prod_{j'=j+1}^{k-1}(I-\eta S^{(i,j')})\right)q^{(i,j)}. \tag{4.8}$$

$$q^{(i,k)} = \sum_{m\in\sigma(k)}\frac{|B_{m,k}|}{|B_m|}F_{m,k}(\mathbf{w}^{(i,0)}), \quad S^{(i,k)} = \sum_{m\in\sigma(k)}\frac{|B_{m,k}|}{|B_m|}\mathbf{H}_{m,k}^{(i,k)}, \tag{4.9}$$

$$\tag{4.10}$$

and we apply the convention

$$\prod_{j'=a+1}^{a}(I-\eta S^{(i,j')}) = 1, \quad a\in\mathbb{N}_{>0}. \tag{4.11}$$

Lemma 2. Suppose Assumptions 1 and 3 hold. Then,

$$\|\mathbf{r}^{(i,0)}\| \leq K \sum_{k=1}^{k-1} \sum_{j=1}^{k-1} \left(L \sum_{m \in \sigma(k)} \frac{|B_{m,k}|}{|B_m|} \right) \left(\prod_{j'=j+1}^{k-1} \left(1 + \eta L \sum_{m \in \sigma(j')} \frac{|B_{m,j'}|}{|B_m|} \right) \right) \cdot \left(C \sum_{m \in \sigma(j)} \left(1 - \frac{|B_{m,j}|}{|B_m|} \right) |\alpha + \sigma(j)| + \sigma(j) \|\nabla \mathbf{F}(\mathbf{w}^{(i,0)})\| \right). \quad (4.12)$$

Lemma 3. Suppose Assumptions 1 and 3 hold. Then,

$$\|\mathbf{r}^{(i,0)}\|^2 \leq K \sum_{k=1} (k-1) \sum_{j=1}^{k-1} L^2 \left(\sum_{m \in \sigma(k)} \frac{|B_{m,k}|}{|B_m|} \right)^2 \left(\prod_{j'=j+1}^{k-1} \left(1 + \eta L \sum_{m \in \sigma(j')} \frac{|B_{m,j'}|}{|B_m|} \right) \right)^2 \cdot \left(2\sigma(j) \sum_{m \in \sigma(j)} \left(1 - \frac{|B_{m,j}|}{|B_m|} \right)^2 |\alpha^2 + \|\nabla \mathbf{F}(\mathbf{w}^{(i,0)})\|^2| \right). \quad (4.13)$$

Proof. A detailed proof is provided in the Appendix in the supplementary materials. As $F(\mathbf{w})$ is L-smooth,

$$F(\mathbf{w}^{(i+1,0)}) - F(\mathbf{w}^{(i,0)}) \leq \langle \nabla F(\mathbf{w}^{(i,0)}), \mathbf{w}^{(i+1,0)} - \mathbf{w}^{(i,0)} \rangle + \frac{L}{2} \|\mathbf{w}^{(i+1,0)} - \mathbf{w}^{(i,0)}\|^2. \quad (19)$$

By Lemma 1,

$$\begin{aligned} \langle \nabla F(\mathbf{w}^{(i,0)}), \mathbf{w}^{(i+1,0)} - \mathbf{w}^{(i,0)} \rangle &\leq \langle \nabla F(\mathbf{w}^{(i,0)}), -\eta M \nabla F(\mathbf{w}^{(i,0)}) + \eta^2 \mathbf{r}^{(i,0)} \rangle \\ &\leq -\eta M \|\nabla F(\mathbf{w}^{(i,0)})\|^2 + \eta^2 \|\nabla F(\mathbf{w}^{(i,0)})\| \|\mathbf{r}^{(i,0)}\|. \end{aligned} \quad (20)$$

Write

$$\begin{aligned} \|\mathbf{r}^{(i,0)}\| &= U_1 + U_2 \|\nabla F(\mathbf{w}^{(i,0)})\|, \\ \|\mathbf{r}^{(i,0)}\|^2 &= V_1 + V_2 \|\nabla F(\mathbf{w}^{(i,0)})\|^2. \end{aligned} \quad (21)$$

Hence,

$$\begin{aligned} \langle \nabla F(\mathbf{w}^{(i,0)}), \mathbf{w}^{(i+1,0)} - \mathbf{w}^{(i,0)} \rangle &\leq -\eta M \|\nabla F(\mathbf{w}^{(i,0)})\|^2 + \eta^2 \|\nabla F(\mathbf{w}^{(i,0)})\| (U_1 + U_2 \|\nabla F(\mathbf{w}^{(i,0)})\|) \\ &\leq -\left(\eta M - \eta^2 U_2 - \frac{\eta^2}{2} \right) \|\nabla F(\mathbf{w}^{(i,0)})\|^2 + \frac{\eta^2 U_2^2}{2}. \end{aligned} \quad (22)$$

We also have

$$\frac{L}{2}\|\mathbf{w}^{(i+1,0)} - \mathbf{w}^{(i,0)}\|^2 \leq L\eta^4 M^2 \|\nabla F(\mathbf{w}^{(i,0)})\|^2 + L\eta^4(V_1 + \|\nabla F(\mathbf{w}^{(i,0)})\|^2 V_2). \tag{23}$$

As such,

$$F(\mathbf{w}^{(i+1,0)}) - F(\mathbf{w}^{(i,0)}) \leq \left(\frac{\eta^2}{2} + \eta^2 U_2 - \eta M + L\eta^4 V_2\right)\|\nabla F(\mathbf{w}^{(i,0)})\|^2 + \frac{\eta^2 U_2^2}{2} + L\eta^4 V_1. \tag{24}$$

Note that for η sufficiently small,

$$-\eta M + \frac{\eta^2}{2} + \eta^4 U_2 + L\eta^4 M^2 + L\eta^2 V_2 \leq -\frac{1}{2}\eta M. \tag{25}$$

This implies that

$$F(\mathbf{w}^{(i+1,0)}) - F(\mathbf{w}^{(i,0)}) \leq -\eta M \mu (F(\mathbf{w}^{(i,0)}) - F^*) + \frac{\eta^2 U_2^2}{2} + L\eta^4 V_1. \tag{26}$$

Unrolling, we obtain

$$F(\mathbf{w}^{(K,0)}) - F^* \leq (F(\mathbf{w}^{(0,0)}) - F^*)(1 - \eta M \mu)^K + \frac{1}{\eta M \mu}\left(\frac{\eta^2 U_2^2}{2} + L\eta^4 V_1\right), \tag{27}$$

as required.

4.2 Impact of Partial Participation Strategies

For sufficiently small $\eta > 0$, the dominating term in Theorem 1 depending on the partial participation strategy is proportional to U_1^2. As a consequence, we now investigate the impact of the partial participation strategy on U_1. To gain insight, we consider the following four scenarios:

Scenario 1:

- [**Number of Clients:**] M,
- [**Number of Groups:**] K,

- [**Clients per Group:**] M/K,
- [**Partial Participation Strategy:**] Each client is active in exactly one group.

In this scenario, U_1 in Theorem 1 is given by

$$U_1^{(1)} = \sum_{k=1}^{K} \sum_{j=1}^{k-1} \frac{ML}{K} \left(\prod_{j'=j+1}^{K} \left(1 + \eta \frac{LM}{K} \right) \right) \frac{M\alpha}{K} \tag{4.14}$$

Scenario 2:

- [**Number of Clients:**] M,
- [**Number of Groups:**] K,
- [**Clients per Group:**] M/K clients in group 1, $M/K + 1$ in group $j \neq 1$,
- [**Partial Participation Strategy:**] A client in group K is present in all groups. As such, the size of the groups $j \neq K$ is $M/K + 1$ clients.

In this scenario, U_1 in Theorem 1 is given by

$$\begin{aligned} U_1^{(2)} = & \sum_{k=1}^{K-1} \sum_{j=1}^{k-1} L \left(\frac{M}{K} + \frac{1}{K} \right) \left(\prod_{j'=j+1}^{K} \left(1 + \eta L \left(\frac{M}{K} + \frac{1}{K} \right) \right) \right) \\ & \cdot \left(C \left(1 - \frac{1}{K} \right) + \alpha \left(\frac{M}{K} + 1 \right) \right) \\ & + \sum_{j=1}^{K-1} \left(\frac{LM}{K} - 1 + \frac{1}{K} \right) \left(\prod_{j'=j+1}^{K-1} \left(1 + \eta L \left(\frac{M}{K} + \frac{1}{K} \right) \right) \right) \\ & \cdot \left(C \left(1 - \frac{1}{K} \right) + \left(\frac{M}{K} + 1 \right) \alpha \right) \end{aligned} \tag{4.15}$$

Scenario 3:

- [**Number of Clients:**] M,
- [**Number of Groups:**] $K + 1$,
- [**Clients per Group:**] M/K,

- [**Partial Participation Strategy:**] A set of K groups are first constructed as in Scenario 1. In group k, M/K^2 clients from group $k-1$ are also associated (M/K^2 clients from group K are associated with group 1). In order to ensure there are M/K clients per group, M/K^2 clients from each group are removed and associated instead to a new group with index $K+1$. Except for clients in group $K+1$, each client is therefore associated with two groups.

In this scenario, U_1 in Theorem 1 is given by

$$U_1^{(3)} = \sum_{j=1}^{K} L\left(\frac{M}{K} - \frac{M}{2K^2}\right)\left(\prod_{j'=j+1}^{K}\left(1+\eta L\left(\frac{M}{K}-\frac{M}{2K^2}\right)\right)\right)\left(\frac{CM}{2K^2}+\frac{M}{K}\alpha\right) + \sum_{k=1}^{K}\sum_{j=1}^{k-1}\frac{LM}{K}\left(\prod_{j'=j+1}^{k-1}\left(1+\eta\left(\frac{M}{K}-\frac{M}{2K^2}\right)\right)\right)\left(\frac{CM}{2K^2}+\frac{M}{K}\alpha\right) \quad (4.16)$$

Scenario 4:

- [**Number of Clients:**] M,
- [**Number of Groups:**] $K+1$,
- [**Clients per Group:**] M/K,
- [**Partial Participation Strategy:**] A set of K groups are first constructed as in Scenario 1. In group k, M/K^2 common clients from group 1 are also associated. To ensure M/K clients per group, in groups $k=2,\ldots,K$, M/K^2 clients are removed and instead associated with group indexed by $K+1$. The common clients in groups $k=1,\ldots,K$ are then also associated with group $K+1$, yielding M/K clients for each group.

In this scenario, U_1 in Theorem 1 is given by

$$U_1^{(4)} = \sum_{k=1}^{K+1}\sum_{j=1}^{k-1} L\left(\frac{M}{K}-\frac{M}{K^2}+\frac{M}{(K+1)K^2}\right)\left(\prod_{j'=j+1}^{k-1}\left(1+\eta L\left(\frac{M}{K}-\frac{M}{K^2}+\frac{M}{(K+1)K^2}\right)\right)\right) \cdot\left(\frac{CM}{K^2}\left(1-\frac{1}{K+1}\right)+\frac{M}{K}\alpha\right) \quad (4.17)$$

An illustrative comparison of U_1 in each scenario is given in Fig. 4.1. The scenarios considered in the study are summarised in Table. 4.1. An inspection of the quantities $U_1^{(j)},\ j=1,\ldots,4$ and Fig. 4.1 yields the following observations:

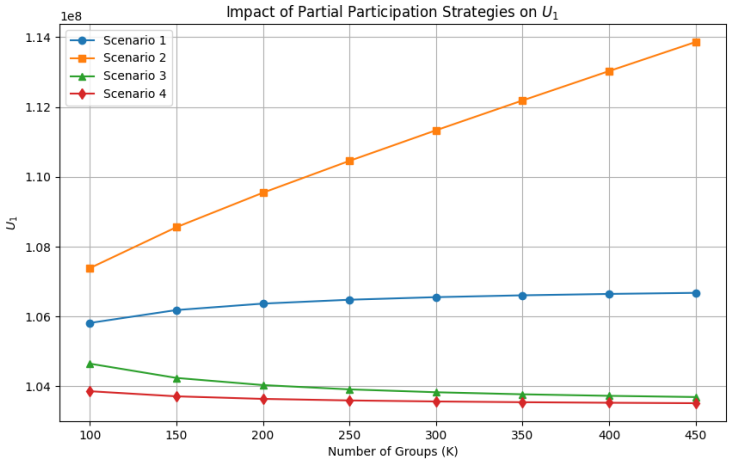


Figure 4.1: Plot of U_1 for varying number of groups K in Scenario 1. Parameters: $M = 10000$, $\alpha = 2$, $C = 1$, $L = 1$, $\eta = 0.00001$.

(i) **Adding additional clients to groups reduces performance.** Observe that $U_1^{(1)}$ (Scenario 1) is strictly smaller than $U_1^{(2)}$ (Scenario 2).

(ii) **For a small number of groups K, associating clients to multiple groups reduces performance.**

(iii) **For a sufficiently large number of groups K, associating clients to multiple groups does not lead to performance degradation.**

(iv) **Associating a client with $K+1$ groups improves performance over associating a client with 2 groups.** Observe in Fig. 4.1 that $U_1^{(4)}$ is strictly smaller than $U_1^{(3)}$.

Scenario	Number of Clients	Number of Groups	Clients per Group
Scenario 1	M	K	M/K
Scenario 2	M	K	M/K in group 1, $M/K+1$ in group $j \neq 1$
Scenario 3	M	$K+1$	M/K
Scenario 4	M	$K+1$	M/K

Table 4.1: Summary of Scenarios

Chapter 5

Experimental Results

5.1 Implementation Setup

We evaluated Algorithm 1 (Option ShuffledSGD) by implementing various partial participation strategies in a standard image classification task, specifically training the CNN ResNet-18 model on the CIFAR-10 and CIFAR-100 datasets. These datasets include 50,000 training images and 10,000 test images, which were distributed among clients according to the Dirichlet distribution DirK (α). The parameter α defines the heterogeneity of data among clients, ranging from α = 0.1 (high heterogeneity) to α = 100 (homogeneity). The entire dataset, both training and testing images, was partitioned among clients using the Dirichlet distribution. Each client's data subset was further divided into training and testing sets in an 80:20 ratio.

Client-side training utilized the shuffledSGD with momentum optimization. The learning rate on the client side was systematically reduced, being halved after every one-fifth of the remaining training rounds. Hyper-parameter selections were based on exhaustive grid search experiments.

Given the balanced nature of both local and global datasets, model performance was assessed using top-1 test accuracy. The test accuracy for each cycle was determined by averaging the individual accuracies of all models trained in that cycle.

5.2 Impact of Data Splitting on Global Model Accuracy

In Section 2, the problem formulation describes a client affiliated with multiple groups updating based on only a subset of the local data. To validate this approach, we compared its performance against a scenario where the client updates using the entire local dataset.

figure 5.3 display the top-1 test accuracy for both data splitting (using a subset of the local dataset) and the complete local dataset (full data) for the CIFAR datasets. This experiment was conducted across various levels of client heterogeneity, defined by α values ranging from 0.1 (highly heterogeneous) to 100 (homogeneous). The numbers in parentheses indicate the count of clients shared between consecutive groups.

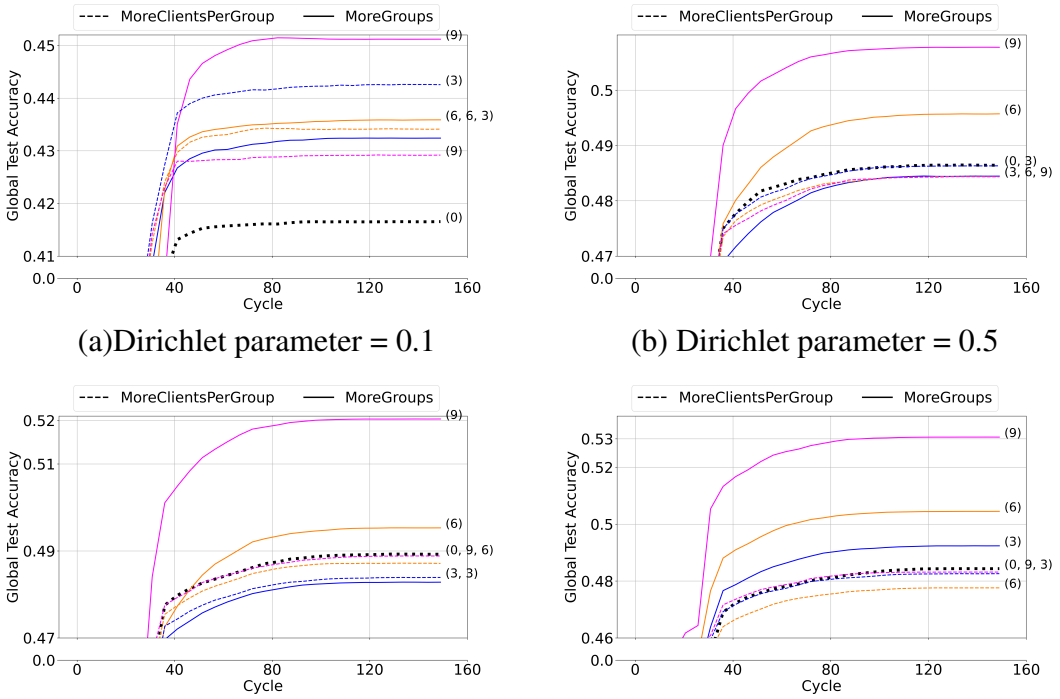
The results show that the performance difference between data splitting and full data updates is minimal, with a maximum variance of 1.5% across experiments. This indicates that data splitting is an effective strategy for reducing computational demands on clients. Additionally, increasing the number of clients associated with multiple groups (i.e., enhancing overlaps between client groups) can lead to substantial performance gains, reaching up to 4%.

5.2.1 Partial Participation Strategies

Fig. 5.1 and Fig. 5.2 plots the global test accuracy for different partiap participation strategies. We focus on Scenario 2 and Scenario 4 detailed in Chapter 4, corresponding to introducing overlapping groups via increasing the group size (Scenario 2) and increasing the number of groups while maintaining the group size (Scenario 4).

We observe that introducing overlapping groups by increasing the number of clients per group leads to performance improvements only when the client data sets are highly homogeneous. In the case of heterogeneous local data, a performance reduction leads to a reduction in the accuracy.

In the case where the number of groups is increased, we observe that the test accuracy is consistently improved. This observation holds both for heterogeneous and homogeneous client data sets. We also observe that increasing the number of groups leads to improved performance over increasing the number of clients per group.



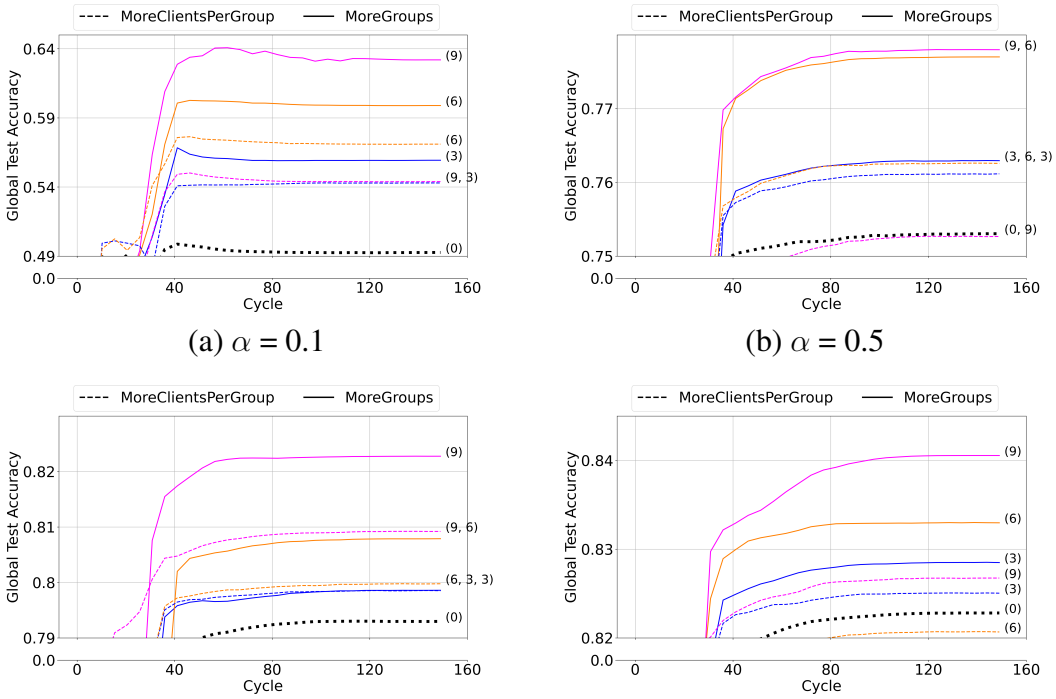
(a)Dirichlet parameter = 0.1

(b) Dirichlet parameter = 0.5

(c) Dirichlet parameter = 1.0

(d) Dirichlet parameter = 100.0

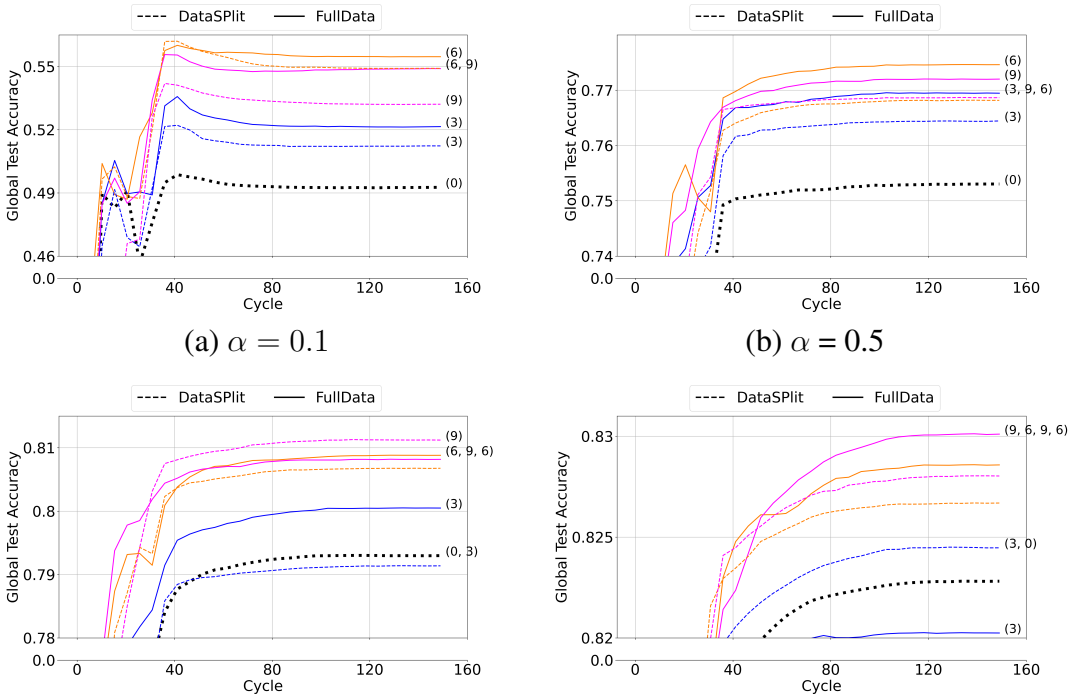
Figure 5.1: Performance of the global model subject to the proposed grouping strategies vs the number of cycles elapsed, with varying degrees of overlaps and data heterogeneity. The solid lines denote the strategy wherein we increase the number of groups to accommodate the common clients while maintaining a constant group size. The dashed lines correspond to the other proposed strategy, where the common clients are simply injected into every other group. The values in the brackets depict the number of clients that are common between the first group and every other group.



(a) $\alpha = 0.1$ (b) $\alpha = 0.5$

(c) $\alpha = 1.0$ (d) $\alpha = 100.0$

Figure 5.2: Performance of the global model subject to the proposed grouping strategies vs the number of cycles elapsed, with varying degrees of overlaps and data heterogeneity for Resnet-18 model trained on CIFAR-10 dataset. The solid lines denote the strategy wherein we increase the number of groups to accommodate the common clients while maintaining a constant group size. The dashed lines correspond to the other proposed strategy, where the common clients are simply injected into every other group.



(a) $\alpha = 0.1$ (b) $\alpha = 0.5$

(c) $\alpha = 1.0$ (d) $\alpha = 100.0$

Figure 5.3: The figures show the top-1 test accuracy of the global model vs the number of cycles elapsed for Resnet-18 model trained on CIFAR-10 dataset. The solid curves depict the runs when clients participate with their full dataset, while for the runs corresponding to the dashed curves, the dataset of the common clients was partitioned such that every client goes through their entire dataset exactly once in the course of one cycle. The values in the brackets correspond to the number of clients common between adjacent client groups.

Chapter 6

Concluson

In this paper, we explored the impact of client heterogeneity in communication capabilities on federated learning (FL) performance. We specifically focused on strategies for grouping clients and the effects of allowing clients to participate in multiple groups. Our findings indicate that, while traditional methods often assume single-group participation per client, leveraging multi-group participation can lead to substantial accuracy improvements in cross-device FL scenarios.

Our theoretical analysis provided insights into the convergence behavior of the proposed strategies, establishing the conditions under which multi-group participation can be beneficial. Experimentation on standard image classification tasks using the CIFAR-10 and CIFAR-100 datasets demonstrated that clients updating on subsets of their data within multiple groups outperformed those updating on their entire dataset. These results held across varying degrees of data heterogeneity, reinforcing the viability of our approach.

We identified two key questions and provided answers through empirical and theoretical analysis:

1) Clients participating in multiple groups should train on subsets of their data, as opposed to the entire dataset, to avoid performance degradation.

2) Increasing the number of groups while allowing clients to participate in multiple groups can improve performance, especially when the client data is heterogeneous.

In cross-device FL, devices may have significantly different communication re- sources. This can arise either from the devices themselves or their location rel- ative to the server. In this case, it may be desirable for some clients to commu- nicate updates more frequently than others. In this paper, we studied partial participation strategies where clients can be associated with multiple groups, and groups are activated in a cyclic fashion. We showed that in many scenar- ios increasing the number of clients associated with multiple groups can lead to significant performance improvements.

Chapter 7

Future Scope

The scope for future work in this domain is vast, with several promising directions to further enhance the effectiveness of federated learning in heterogeneous communication environments. Some potential areas of exploration include:

1) Dynamic Group Formation: Investigate adaptive strategies for dynamically forming and reconfiguring client groups based on real-time communication capabilities and data characteristics. This can help optimize resource usage and improve model convergence rates.

2) Advanced Data Partitioning: Explore more sophisticated data partitioning techniques that account for data distribution and relevance, potentially leveraging clustering and sampling methods to enhance local training efficiency and global model accuracy.

3) Exploring the impact of Federated Learning Algorithms that mitigate data heterogeneity: Overlapping groups in a way try to mitigate data heterogeneity by reducing the inter-group heterogeneity. Several algorithms have been proposed to mitigate the problems arising from data heterogeneity in FL, such as [7, 8], and it would be interesting to see how they play with our setup.

4) Security and Privacy: Investigate the implications of multi-group participation on data privacy and security. Develop robust mechanisms to ensure that the proposed strategies do not inadvertently compromise client data or make the system more vulnerable to attacks.

5) Real-world Applications: Validate the proposed strategies in real-world scenarios, such as smart cities, healthcare, and IoT networks, where communication capabilities and data distributions are inherently heterogeneous. This can provide practical insights and further refine the strategies for specific application domains.

6) Scalability: Assess the scalability of the proposed methods in large-scale federated learning deployments with thousands or millions of clients. Address challenges related to communication overhead, computational load balancing, and efficient coordination among clients and the server.

By addressing these areas, future research can build on the foundations laid in this paper, pushing the boundaries of what is possible with federated learning in heterogeneous environments and paving the way for more robust, efficient, and practical deployment of FL systems.

Bibliography

[1] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. pages 1273–1282.

[2] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[3] Fei-Yue Wang, Rui Qin, Juanjuan Li, Xiao Wang, Hongwei Qi, Xiaofeng Jia, and Bin Hu. Federated management: Toward federated services and federated security in federated ecology. *IEEE Transactions on Computational Social Systems*, 8(6):1283–1290, 2021.

[4] Kallista Bonawitz, Peter Kairouz, Brendan McMahan, and Daniel Ramage. Federated learning and privacy: Building privacy-preserving systems for machine learning and data science on decentralized data. *Queue*, 19(5):87–114, oct 2021.

[5] Hubert Eichner, Tomer Koren, Brendan Mcmahan, Nathan Srebro, and Kunal Talwar. Semi-cyclic stochastic gradient descent. pages 1764–1773.

[6] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2):1–19, jan 2019.

[7] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450.

[8] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. pages 5132–5143.

[9] Yae Jee Cho, Pranay Sharma, Gauri Joshi, Zheng Xu, Satyen Kale, and Tong Zhang. On the convergence of federated averaging with cyclic client participation. pages 5677–5721.

[10] Mónica Ribero, Haris Vikalo, and Gustavo de Veciana. Federated learning under intermittent client availability and time-varying communication constraints. *IEEE Journal of Selected Topics in Signal Processing*, 17(1):98–111, 2023.

[11] Chen Zhu, Zheng Xu, Mingqing Chen, Jakub Konečný, Andrew Hard, and Tom Goldstein. Diurnal or nocturnal.

[12] Zhe Qu, Xingyu Li, Jie Xu, Bo Tang, Zhuo Lu, and Yao Liu. On the convergence of multi-server federated learning with overlapping area. 2022.

[13] Eunjeong Jeong, Seungeun Oh, Hyesung Kim, Jihong Park, Mehdi Bennis, and Seong-Lyun Kim. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data. 2018.

[14] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. 2019.

[15] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. 2018.

[16] Michael R. Sprague, Amir Jalalirad, Marco Scavuzzo, Catalin Capota, Moritz Neun, Lyman Do, and Michael Kopp. *Asynchronous Federated Learning for Geospatial Applications*, page 21–28. Springer International Publishing, 2019.

[17] Accelerating federated learning by selecting beneficial herd of local gradients. https://arxiv.org/html/2403.16557v1, 2024. Accessed: 2024-06-13.

[18] Takayuki Nishio and Ryo Yonetani. Client selection for federated learning with heterogeneous resources in mobile edge. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*. IEEE, may 2019.