# Client heterogeneity in federated learning systems

Angelo Rodio

UNIVERSITÉ CÔTE D'AZUR | ÉCOLE DOCTORALE SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

# THÈSE DE DOCTORAT

# Hétérogénéité des Clients dans les Systèmes d'Apprentissage Fédérés


## Angelo RODIO

Équipe NEO, Centre Inria d'Université Côte d'Azur, 3IA Côte d'Azur


**Présentée en vue de l'obtention du grade de docteur en** Informatique **d'**Université Côte d'Azur

**Dirigée par :** Giovanni NEGLIA, Directeur de Recherche, Centre Inria d'Université Côte d'Azur

**Co-dirigée par :** Alain JEAN-MARIE, Directeur de Recherche, Centre Inria d'Université Côte d'Azur

**Soutenue le :** 03/07/2024

**Devant le jury, composé de :**

Walid DABBOUS, Directeur de recherche, Centre Inria d'Université Côte d'Azur
H. Vincent POOR, Professeur, Princeton University, United States
Michele ROSSI, Professeur, University of Padova, Italy
Erik G. LARSSON, Professeur, Linköping University, Sweden
Francesco MALANDRINO, Chargé de recherche, CNR-IEIIT, Italy

RÉPUBLIQUE FRANÇAISE
*Liberté Égalité Fraternité*

Inria

UNIVERSITÉ CÔTE D'AZUR | INTERDISCIPLINARY INSTITUTE FOR ARTIFICIAL INTELLIGENCE 3IA CÔTE D'AZUR

FRANCE 2030

# HÉTÉROGÉNÉITÉ DES CLIENTS DANS LES SYSTÈMES D'APPRENTISSAGE FÉDÉRÉS

---

*Client Heterogeneity in Federated Learning Systems*

**Angelo RODIO**

## Jury :

**Président du jury**
Walid DABBOUS, Directeur de recherche, Centre Inria d'Université Côte d'Azur

**Rapporteurs**
H. Vincent POOR, Professeur, Princeton University, United States
Michele ROSSI, Professeur, University of Padova, Italy

**Examinateurs**
Erik G. LARSSON, Professeur, Linköping University, Sweden
Francesco MALANDRINO, Chargé de recherche, CNR-IEIIT, Italy

**Directeur de thèse**
Giovanni NEGLIA, Directeur de Recherche, Centre Inria d'Université Côte d'Azur

**Co-directeur de thèse**
Alain JEAN-MARIE, Directeur de Recherche, Centre Inria d'Université Côte d'Azur

---

Université Côte d'Azur

Angelo RODIO

***Hétérogénéité des Clients dans les Systèmes d'Apprentissage Fédérés***

xiv+188 p.

Ce document a été préparé avec LaTeX2e et la classe `these-ISSS` version v. 2.10.

*Impression : output.tex – 28/8/2024 – 17:16*

*Révision pour la classe : these-ISSS.cls,v 2.10 2020/06/24 14:16:37 mpelleau*

*To my advisor, who shaped my academic journey, and to my family, for their unconditional support.*

# Hétérogénéité des Clients dans les Systèmes d'Apprentissage Fédérés


## Résumé

L'Apprentissage Fédéré (FL) est un cadre collaboratif où des clients, tels que des smartphones et des appareils IoT, entraînent un modèle de machine learning sous la coordination d'un serveur central sans partager leurs données. L'hétérogénéité des clients dans les systèmes FL résulte de l'hétérogénéité statistique des jeux de données locaux, des différences de spécifications matérielles (puissance du CPU, capacité de mémoire), des types de connectivité réseau (par exemple, WiFi ou 5G) et de disponibilité d'énergie (niveaux de batterie), échappant ainsi au contrôle du serveur. Cette thèse aborde les défis posés par cette hétérogénéité des clients, leur impact sur la convergence des algorithmes FL, et propose des solutions pratiques pour améliorer l'efficacité et l'utilisation des ressources du système. La première contribution traite le problème de la participation hétérogène des clients : ceux-ci participent à l'entraînement du modèle de manière occasionnelle et avec des fréquences variées, ce qui pose trois défis principaux. D'abord, les clients les plus actifs peuvent biaiser le modèle global en raison de l'hétérogénéité statistique de leurs données. Ensuite, compenser ce biais en surpondérant les clients moins actifs augmente la variance du processus d'apprentissage. Enfin, la participation des clients peut être corrélée temporellement et spatialement. Nous caractérisons le compromis biais-variance et analysons la convergence des algorithmes FL, en supposant que la participation suit un processus de Markov. Notre algorithme FL prenant en compte les corrélations, `CA-Fed`, est le premier à minimiser ce compromis et à accélérer la convergence des algorithmes FL lorsque la participation des clients est corrélée. La deuxième contribution traite de la variabilité du processus d'apprentissage introduite par la participation hétérogène des clients. Les méthodes de réduction de variance utilisant des mises à jour du modèle obsolètes pour les clients non participants présument une participation homogène. Avec une participation hétérogène, le serveur doit agréger des mises à jour du modèle d'obsolescence variable, un défi encore inexploré. Nous analysons la convergence de ces algorithmes et proposons `FedStale`, un algorithme FL qui combine de manière optimale mises à jour du modèle fraîches et obsolètes, performant dans divers contextes hétérogènes. La troisième contribution aborde l'hétérogénéité des ressources réseau : les clients rencontrent des canaux de communication avec des caractéristiques variées (par exemple, perte de paquets, interférences), ce qui dégrade les performances des algorithmes FL. Viser une haute fiabilité de transmission dans le FL est sous-optimal, et les stratégies d'atténuation des pertes (par exemple, les retransmissions) nécessitent plus de ressources et prolongent la durée de l'entraînement. Nous explorons des approches algorithmiques pour gérer les pertes pendant l'entraînement et présentons un algorithme FL tolérant aux pertes de paquets, `UPGA-PL`, offrant des performances comparables à celles des canaux sans perte au prix de quelques cycles de communication supplémentaires. La dernière contribution examine l'hétérogénéité des environnements matériels : des clients aux capacités de calcul variées (dispositifs périphériques, serveurs edge, et infrastructures cloud) peuvent coopérer pour apprendre un modèle commun ; cependant, cette hétérogénéité rend le déploiement uniforme du modèle infaisable à l'inférence. Les Systèmes d'Inférence Coopérative (CIS) permettent aux dispositifs moins performants de déléguer des tâches d'inférence à des dispositifs plus puissants avec des modèles plus volumineux ; cependant, l'entraînement FL ne tient pas compte de leur utilisation future à l'inférence. Notre algorithme FL optimisé pour l'inférence, `Fed-CIS`, est le premier à considérer la charge future des requêtes d'inférence pour chaque sous-modèle dès l'entraînement. Il permet aussi aux clients plus puissants de contribuer à l'entraînement des modèles pour les plus faibles. Les remarques finales de cette thèse abordent les défis ouverts rencontrés et esquissent les directions de recherche futures.



**Mots-clés :** Apprentissage Fédéré, Optimisation Distribuée, Chaîne de Markov, Réduction de la Variance

# Client Heterogeneity in Federated Learning Systems


## Abstract

Federated Learning (FL) is a collaborative framework where clients—typically smartphones and IoT devices—train a machine learning model under the orchestration of a central server without sharing their datasets. Client heterogeneity in FL systems stems from statistical heterogeneity of local datasets, different device capabilities in hardware specifications (CPU power, memory capacity), network connectivity types (e.g., 5G and WiFi), power availability (battery levels), and it is outside server control. This thesis tackles the challenges of client heterogeneity in FL systems, their impact on the convergence of FL algorithms, and presents practical solutions for enhanced system efficiency and resource use. The first contribution addresses the problem of heterogeneous client participation: clients partake in the model training only occasionally and with varying frequencies. Three primary challenges arise. First, the "more participating" clients may bias the global model due to statistical heterogeneity in the clients' datasets. Second, addressing this bias by overcompensating for "less participating" clients introduces a larger variance in the learning process. Third, client participation can be correlated, due to the clients' correlated participation dynamics across time and geographic distributions. We characterize the bias-variance trade-off resulting from heterogeneous client participation and analyze the convergence of FL algorithms, assuming that client participation follows a Markov process. Our correlation-aware FL algorithm, `CA-Fed`, is the first heuristic to minimize this bias-variance-correlation trade-off and thus achieve faster convergence. The second contribution addresses the large variability in the learning process introduced by heterogeneous client participation. Variance reduction methods that leverage stale model updates for non-participating clients only consider homogeneous client participation. When participation is heterogeneous, the server must aggregate client updates with varying staleness—a challenge that remained unexplored. We analyze the convergence of these algorithms under heterogeneous participation, examining the advantages and disadvantages of leveraging stale updates in such heterogeneous environments. Our Staleness-Aware FL algorithm, `FedStale`, opportunely aggregates fresh and stale updates and performs well across many heterogeneous settings. The third contribution tackles heterogeneity in network resources: clients experience lossy communication channels with diverse characteristics (e.g., path loss, interference), which degrade FL algorithms' performance. Targeting high transmission reliability in FL is suboptimal, and loss mitigation strategies (e.g., retransmissions) demand more resources and longer training durations. We investigate algorithmic approaches for handling losses during training and present a packet loss-aware FL algorithm, `UPGA-PL`, with comparable performance to ideal lossless channels at the cost of a few additional communication rounds. The last contribution investigates heterogeneity in hardware environments: clients with diverse computing capabilities (e.g., end-devices, edge servers, and cloud infrastructures) may cooperate to learn a common model; yet, client heterogeneity makes uniform model deployment infeasible at inference time. Cooperative Inference Systems (CISs) enable less-performing devices to offload parts of their inference tasks to more powerful devices with larger models within the network; however, FL training overlooks how these models will be used at inference time. Our inference-aware FL algorithm, `Fed-CIS`, is the first to consider the future inference request load for each sub-model at training time. It also enables computationally stronger clients to help train models for the weaker ones. The concluding remarks reflect on the open challenges encountered throughout this thesis and outline prospective research directions for future work.



**Keywords:** Federated Learning, Distributed Optimization, Markov chain, Variance Reduction

# Acknowledgements

I would like to thank Dr. Giovanni Neglia, my supervisor, for his persistent advice and constant support during this research. His scientific rigor and hard work have been truly inspirational. I have strived to learn as much as possible under his guidance, and I look forward to continuing this relationship in future endeavors.

I am also grateful to my thesis committee members, Walid Dabbous, Erik G. Larsson, Francesco Malandrino, H. Vincent Poor, and Michele Rossi, for their valuable feedback and participation in the defense. Engaging in discussions with such experts has been both challenging and rewarding.

The results in this thesis have been obtained in collaboration with many other people. I sincerely thank Francescomaria Faticanti, Othmane Marfoq, and Chuan Xu for their constant support and friendship during this journey. I am also grateful for the opportunity to have collaborated with Caelin Kaplan, Tareq Si Salem, Fabio Buscacca, Stefano Mangione, Francesco Restuccia, Ilenia Tinnirello, Sergio Palazzo, and Emilio Leonardi, as well as the master students I had the privilege of co-supervising: Albachiara Bellaroba, Maysae Hmamouchi, and Mohamed Salah Jebali. I have learned a lot from working with you, and hope you found our collaboration equally valuable.

Special thanks to all my colleagues at Inria Sophia Antipolis and the NEO team members, Ahmad Nasser, Alain Jean-Marie, Ashok Krishnan Komalan Sindhu, Caelin Kaplan, Charlotte Rodriguez, Cosimo Giani, Diego Goldsztajn, Eitan Altman, Emmanouil Athanasakos, Francesco Diana, Gabriele Castellano, Guilherme Iecker Ricardo, Hariprasad Manjunath, Ibtihal El Mimouni, Jake Clarkson, Jacopo Talpini, Jane Desplanques, Jose Francisco Daunas Torre, Kavitha Veeraruna, Ke Sun, Khushboo Agarwal, Konstantin Avrachenkov, Lotte Weedage, Louis Hauseux, Lucas Gamertsfelder, Lucas Siviero Sibemberg, Maximilien Dreveton, Mikhail Kamalov, Olga Chuchuk, Samir M. Perlaza, Sara Alouf, Sadaf Ul-Zuhra, Suhail Mohmad Shah, Tejas Pagare, Vinay Kumar B.R., Vijith Kumar, Xufeng Zhang, Xinying Zou, and Younes Ben Mazziane.

Finally, I am deeply grateful to my family—my parents, Mada and Piero; my girlfriend, Barbara; my grandmother, Benedetta; my cousins, Carlo and Francesco; and my friends, Luigi and Valerio—for their unwavering love and encouragement throughout this journey.

This research would not have been possible without the support of everyone mentioned above, along with many others who have contributed to this journey, for which I am truly grateful.

Sophia Antipolis, France
July 3rd, 2024

Angelo Rodio

# Contents

## Appendices

# CHAPTER 1

# Introduction

## 1.1 Context of the Thesis

The machine learning (ML) community has made tremendous improvements across many application domains by following a simple recipe: gather large training datasets, develop model architectures, and scale computing resources. This recipe helps to enhance user experience and engagement with ML technologies, and encourages further investment in ML research and development in both current and new application domains.

Breaking down the recipe, we first identify specific difficulties in improving model architectures. Although developing new models leads to major breakthroughs that improve applications, this process often depends on unreliable epiphanies. Research in this field often involves creative rethinking of the modeling problem and exhaustive hyper-parameter exploration, occasionally benefiting from serendipity.

As a complementary contribution to model architecture search, this thesis focuses on the other two recipe components: gathering large training data and scaling computational resources. In these fields, we have greater control and understanding over progress.

**The importance of data in machine learning.** Machine learning models are notoriously data-hungry, fed by massive volumes of data. It is well known that simply using more data to train larger models enhance their prediction accuracy (Polyzotis, Roy, Whang, & Zinkevich, 2017, 2018; Zheng, Li, Li, Shan, & Cheng, 2017; Roh, Heo, & Whang, 2021). The relationship between data volume and model accuracy has been observed in empirical learning curves generated across various real-world applications (Hestness et al., 2017). Figure 1.1 breaks down these empirical learning curves into three key phases:

- The curve begins in the "small data region," when models struggle to learn from a small number of training samples, and can only perform as well as "random" guessing.
- The middle area of learning curves is the "power-law region", where each new training sample adds knowledge that helps models improve predictions on previously unseen samples. The power-law exponent defines the steepness of this curve, and depends on the specific ML task. This power-law region exists across a large range of models, optimizers, regularizers, and loss functions.
- With sufficiently large training sets, models' accuracy is predicted to saturate in a region dominated by "irreducible error" (e.g., Bayes error). However, this region has still not been observed in real applications.

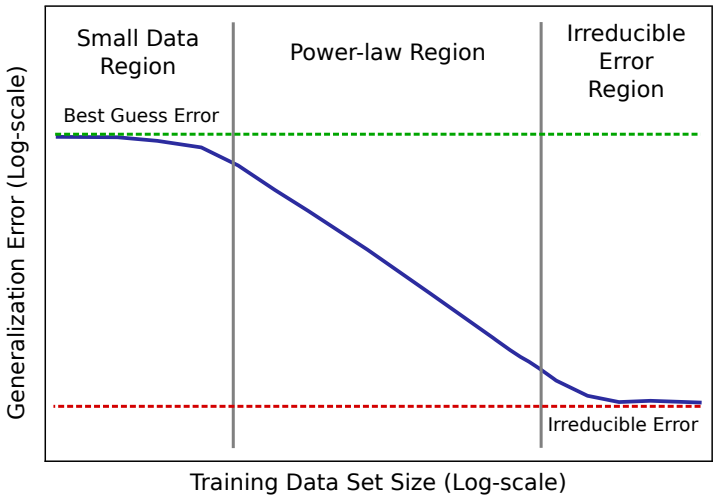



Figure 1.1: The empirical learning curve of real applications shows robust power-law regions: scaling the training data set is likely to improve the model's accuracy (Hestness et al., 2017).

**Massive data production on the end-device.** In the search for ever-increasing model accuracy, the data publicly available progressively becomes insufficient. A less exploited source of essential data is the massive volume generated at the edge of the network, notably through end-user devices such as smartphones and IoT devices (Uhlemann, Lehmann, & Steinhilper, 2017; Stoica & Shenker, 2021). These devices are not simply communication tools but prolific data factories, continuously generating detailed and diversified information. Smartphones, for instance, collect a plethora of data including images, geographic locations, health measurements, personal preferences, app usage statistics, and even ambient environmental data through sensors. Similarly, IoT devices generate a continual stream of real-time data about user interactions, system performance, and environmental variables. Leveraging this rich data from end-devices offers a critical resource, aligning with the expanding needs of more sophisticated machine learning architectures and representing a pivotal shift in machine learning training methodologies.

**Disadvantages of centralized training.** A standard approach to training machine learning models involves gathering data in a centralized server, often a cloud-based architecture. We formalize the problem as follows. Denote the parameter vector $\boldsymbol{w} \in \mathbb{R}^d$ of a machine learning (for instance, the vector $\boldsymbol{w}$ can represent the weights of a neural network architecture), and a finite set of devices $\mathcal{N} = 1, \ldots, N$, referred to as *clients*. Each client $i$ is equipped with a unique dataset $D_i$. In centralized training, the server aggregates the client datasets into a global dataset $D = \bigcup_{i=1}^{N} D_i$. Subsequently, the training process can be formulated as an optimization problem run by the server:

$$\min_{\boldsymbol{w} \in \mathbb{R}^d} \frac{1}{|D|} \sum_{z \in D} f(\boldsymbol{w}, z), \tag{1.1}$$

where each sample $z = (x, y)$ in dataset $D$ comprises of input features $x$ and the associated target output $y$. The loss function $f : \mathbb{R}^d \times (\mathcal{X} \times \mathcal{Y}) \to \mathbb{R}^+$ evaluates the model's performance, where $\mathcal{X}$ and $\mathcal{Y}$ denote the spaces of input features and output targets, respectively.

While the solution of Problem (1.1) has motivated a substantial body of research in optimization techniques, including the *gradient descent* algorithm (GD) and stochastic approximation methods such as *stochastic gradient descent* (SGD) (Robbins & Monro, 1951) and *minibatch-SGD*, it inherently requires the centralization of data. This requirement creates considerable limits. Firstly, the process of gathering data is communication-intensive and often cost-prohibitive. For example, transmitting video streams from a large number of cameras might quickly increase communication overhead and congest network resources. Secondly, data centralization might pose major privacy risks. Personal data produced by end-user devices is often sensitive, and regulatory frameworks like the General Data Protection Regulation (GDPR) in Europe or the California Consumer Privacy Act (CCPA) in California strictly control data collection procedures, making data aggregation unlawful. Additionally, concerns over data sovereignty—where data must remain inside the country of origin—further complicate the viability of centralizing data.

### 1.1.1 Why Federated Learning?

Federated Learning (FL) (Konečný et al., 2017; McMahan, Moore, Ramage, Hampson, & y Arcas, 2017) is a framework allowing geographically distributed clients to cooperatively learn machine learning (ML) models, without sharing their own local data. The term "Federated Learning" first appeared in 2016 in the seminal work McMahan et al. (2017)*: "We investigate a learning technique that allows users to collectively reap the benefits of shared models trained from this rich data, without the need to centrally store it. We term our approach *Federated Learning*, since the learning task is solved by a loose federation of participating devices (which we refer to as *clients*) which are coordinated by a *central server*." In this paradigm, a central server orchestrates the learning process across a federation of devices, or clients, ensuring that no sensitive data is transmitted, but rather model parameters are shared.

### 1.1.2 Main Motivations: Communication Efficiency and Data Privacy

Federated Learning is characterized by a distinctive feature: the local storing of data on each client's device. This method produces two substantial advantages, each with unique characteristics. The primary and first motivation for proposing this paradigm was to boost communication efficiency.† The federated learning framework avoids data transfer in the centralized cloud server, hence considerably lowering network bandwidth use. For example, transmitting model parameters rather than high-volume data streams may result in substantial network savings, decreasing both bandwidth demands and network congestion. Second, by confining data to its originating device, the federated learning framework fundamentally enhances privacy protection by minimizing data exposure to potential breaches and unauthorized access. This approach aligns with the principle of

*Reference McMahan et al. (2017) was uploaded on arXiv on February 17, 2016, before being published in the Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS) in 2017. Meanwhile, Reference Konečný et al. (2017) was initially published on arXiv on October 18, 2016, followed by an updated version on October 30, 2017.

†Supporting this statement, the title of the seminal paper McMahan et al. (2017) is "Communication-Efficient Learning of Deep Networks from Decentralized Data."

data minimization (*Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC*, 2016; *European Parliament legislative resolution of 13 March 2024 on the proposal for a regulation of the European Parliament and of the Council on laying down harmonised rules on Artificial Intelligence (Artificial Intelligence Act) and amending certain Union Legislative Acts*, 2016), substantially lowering the possible attack surface and the overall system's vulnerability.

### 1.1.3 Main Settings and Applications: Cross-Silo and Cross-Device

Federated Learning has several practical applications and two primary settings—*cross-device FL and cross-silo FL*—that significantly affect algorithm design. Cross-device FL focuses on machine learning across numerous mobile devices, whereas cross-silo FL promotes collaborative learning inside or across organizations. While both settings ensure local data storage, along with communication and privacy benefits, they also present distinct characteristics that affect algorithms implementation (Kairouz et al., 2021, Table 1).

- The *number of clients* in cross-device FL is generally much larger than in cross-silo FL. Cross-silo FL typically involves only a few to tens organizations—clients are powerful servers, clusters, or data centers. For instance, in inter-hospital training, a few clinics collaborate to enhance medical ML models without disclosing sensitive patient data (Silva, Altmann, Gutman, & Lorenzi, 2020). In contrast, cross-device FL may involve millions of end-devices, like for Google's keyboard, where Android smartphones cooperate to enhance Gboard's functionalities like next-word prediction, emoji suggestions, and out-of-vocabulary words discovery (Hard et al., 2018). The quantity of data per client is generally less in cross-device FL.

- *Client availability* vary substantially across cross-silo and cross-device settings. Cross-silo scenarios, characterized by robust computing and network resources, typically guarantee high client availability and consistent participation in training. In contrast, the enormous, sporadically available population in cross-device settings makes the client-server communication more unpredictable and less reliable. Additionally, in cross-device contexts, client participation may also correlate with local data distributions, generating diurnal or nocturnal, semi-cyclic variations that optimization methods must accommodate (Eichner, Koren, Mcmahan, Srebro, & Talwar, 2019; Zhu et al., 2021; Cho et al., 2023).

- *Computation and communication constraints* are more stringent in cross-device settings due to the limited capabilities of end-devices. Conversely, cross-silo clients in data centers typically have access to hardware accelerators and high-bandwidth connectivity.

- *Connection topology* also differs between settings. Mobile devices often support the client–server or hierarchical network topologies (Fercoq, Qu, Richtárik, & Takáč, 2014; Richtárik & Takáč, 2016; Briggs, Fan, & Andras, 2020; Wainakh, Guinea, Grube, & Mühlhäuser, 2020), whereas decentralized communications are feasible in cross-silo environments (Koloskova, Loizou, Boreiri, Jaggi, & Stich, 2020; Kovalev, Salim, & Richtarik, 2020; Kovalev, Koloskova, Jaggi, Richtarik, & Stich, 2021; Marfoq, Xu, Neglia, & Vidal, 2020).

In the corporate sector, large companies are recognizing the potential of federated learning for data-driven decision-making. Besides Google, Apple implemented this technology since iOS 13 for improvements in the QuickType keyboard and voice recognition for "Hey Siri" (Paulik et al., 2021). Additionally, federated learning is spreading its influence into several other sectors. In the financial industry, companies like WeBank are using it to detect money laundering operations (Liu, Kang, Xing, Chen, & Yang, 2020). The healthcare sector is also exploring its benefits: federated learning is being used to improve the understanding and treatment of complex diseases like breast cancer (du Terrail et al., 2021) and predicting clinical outcomes in patients with COVID-19 (specifically, to predict the future oxygen requirements and the need for mechanical ventilation treatment at 24h using data from 20 institutes across the globe) (Dayan et al., 2021).

## 1.2 A Typical Federated Learning System

In this section, we discuss the typical cross-device federated learning system introduced in McMahan et al. (2017). This system comprises several end-devices—denoted as *clients*—with limited computational capacity and slow internet connections, collaborating in a client-server architecture where mobiles communicate solely with the server.

### 1.2.1 Problem Formulation

The canonical federated learning formulation (McMahan et al., 2017; Reddi et al., 2021) involves a set of clients $\mathcal{N} = \{1, \ldots, N\}$, each client $i$ equipped with a dataset $D_i$ consisting of $|D_i|$ samples, collaboratively learning the parameters $\boldsymbol{w} \in \mathbb{R}^d$ of a global machine learning model (for instance, the weights of a neural network architecture). Orchestrated by a central server, these clients cooperate to minimize the *global* objective:

$$\min_{\boldsymbol{w}\in\mathbb{R}^d} F(\boldsymbol{w}) \triangleq \sum_{i=1}^{N} \alpha_i F_i(\boldsymbol{w}), \qquad \sum_{i=1}^{N} \alpha_i = 1, \tag{1.2}$$

where $F_i(\boldsymbol{w}) : \mathbb{R}^d \to \mathbb{R}$ denotes the local objective function at client $i$, that typically takes the form of an empirical risk minimization (ERM) objective function:

$$F_i(\boldsymbol{w}) \triangleq \frac{1}{|D_i|} \sum_{z_i \in D_i} f(\boldsymbol{w}, z_i), \tag{1.3}$$

and $f(\boldsymbol{w}, z_i)$ is the loss induced by model $\boldsymbol{w}$ on data sample $z_i \in D_i$.

Note that $\alpha_i$ is the relative weight of client $i$ and its choice affects the definition of the global objective $F(\boldsymbol{w})$ in Eq. (1.2). Typical choices are:

- Equal weights to all clients, i.e., $\alpha_i = \frac{1}{N}, \forall i$, also denoted as “per-client fairness” criterion;
- Weights proportional to the client’s number of samples, i.e., $\alpha_i = \frac{|D_i|}{|D|}$, where $D = \cup_{i=1}^{N} D_i$, also known as “per-sample fairness” criterion.

**Remark 1.2.1** (Equivalence of Problems (1.1) and (1.2) under "per-sample fairness" criterion)**.** When $\alpha_i = |D_i|/|D|$, the global objective $F(\boldsymbol{w})$ in Eq. (1.2) could be rewritten as follows:

$$F(\boldsymbol{w}) = \sum_{i=1}^{N} \frac{|D_i|}{|D|} F_i(\boldsymbol{w}) = \sum_{i=1}^{N} \frac{|D_i|}{|D|} \times \frac{1}{|D_i|} \sum_{z_i \in D_i} f(\boldsymbol{w}, z_i) = \frac{1}{|D|} \sum_{i=1}^{N} \sum_{z_i \in D_i} f(\boldsymbol{w}, z_i). \quad (1.4)$$

Setting $\alpha_i = |D_i|/|D|$ makes the objective function $F(\boldsymbol{w})$ in Eq. (1.2) equivalent to the ERM objective function in Eq. (1.1) that one would optimize centrally if we constructed a central training dataset from the union of their local datasets ($D = \cup_{i=1}^{N} D_i$).

Compared to the centralized training, we highlight unique properties of Eqs. (1.2) and (1.3):

- *Imbalanced and heterogeneous data*. The local datasets $D_i$'s can have varied distributions and sizes. As a consequence, the local objectives $F_i(\boldsymbol{w})$'s can be different. For example, they may have arbitrarily different local minima. Some assumptions on the local distributions are needed for federated learning to be possible, i.e., for each client to be able to take benefit of the data at other clients.
- *Communication and data privacy constraints*. The local datasets $D_i$ cannot be shared with the server or across clients. Therefore, the local objective function $F_i$ or its gradient $\nabla F_i$ can only be computed at the $i$-th client level, and later communicated to the server.
- *Partial and heterogeneous client availability*. In cross-device FL, the number of devices $N$ can be potentially large. At any given time, only a subset (typically fewer than 1%) of clients are available to connect with the server and participate in the training. Therefore, the server cannot always compute the global objective $F$ or its gradient $\nabla F$ because it only has access to a random subset $\mathcal{S} \subseteq \mathcal{N}$ of participating clients in each communication round. However, the objective function $F(\boldsymbol{w})$ can be employed as a mathematical object in the analysis of federated learning systems, or even approximated numerically in simulations as part of an empirical evaluation procedure (we will do this in Chapter 2).

### 1.2.2 A Typical Federated Learning Algorithm: `FedAvg`

A typical algorithm to solve Problem (1.2) is Federated Averaging (`FedAvg`) (McMahan et al., 2017). `FedAvg` (Algorithm 1) is an iterative algorithm that divides the training process into $T > 0$ communication rounds. At the beginning of the $t$-th communication round, only a subset of clients $\mathcal{S}^{(t)} \subseteq \mathcal{N}$ meeting eligibility rules participates. Specifically, for mobile phones, a device is typically considered eligible if it is currently plugged in, connected to an unmetered WiFi network, and idle, as detailed in (Kairouz et al., 2021, Section 1.1.2). Then, the server broadcasts the current model $\boldsymbol{w}^{(t)}$ to the participating clients $\mathcal{S}^{(t)}$ (Line 4). Upon receiving model $\boldsymbol{w}^{(t)}$, each client $i \in \mathcal{S}^{(t)}$ locally updates the model, usually through a finite number of local stochastic gradient descent (SGD) steps, using its local dataset $D_i$ (Line 6). Afterwards, the client sends-back its updated local model $\boldsymbol{w}_i^{(t,K)}$ to the server (Line 7). Finally, the server aggregates the local updated models $\boldsymbol{w}_i^{(t,K)}$ from all participating clients $i \in \mathcal{S}^{(t)}$ in order to produce a new global model $\boldsymbol{w}^{(t+1)}$ (Line 8).

The FedAvg algorithm can be extended to a versatile framework known as `FedOpt` (Reddi et al., 2021) (Algorithm 2), which grants the algorithm designer the flexibility to modify the client local update rule, the aggregation method, and the server global update rule. `FedOpt` maintains the

**Algorithm 1:** FedAvg (Federated Averaging McMahan et al. (2017))

1 **Input:** client datasets $\{D_i\}$; communication rounds $T$; local iterations $K$; learning rate $\eta$
2 **server** randomly initializes $\boldsymbol{w}^{(1)}$
3 **for** *global communication round* $t = 1, \ldots, T$ **do**
4  **server** broadcasts $\boldsymbol{w}^{(t)}$ to the participating clients $\mathcal{S}^{(t)}$
5  **for** ***participating client*** $i \in \mathcal{S}^{(t)}$, *in parallel* **do**
6  $\boldsymbol{w}_i^{(t,K)} \leftarrow \texttt{ClientUpdate}(\boldsymbol{w}^{(t)}, D_i, K, \eta)$
7  **client** sends $\boldsymbol{w}_i^{(t,K)}$ to the **server**
8  **server** aggregates $\boldsymbol{w}^{(t+1)} = \sum_{i \in \mathcal{S}^{(t)}} \frac{|D_i|}{|D|} \boldsymbol{w}_i^{(t,K)}$
9  **return** $\boldsymbol{w}^{(t+1)}$
10 **Procedure** ClientUpdate$(\boldsymbol{w}^{(t)}, D_i, K, \eta)$
11  **client** initializes $\boldsymbol{w}_i^{(t,0)} \leftarrow \boldsymbol{w}^{(t)}$
12  **for** *local iteration* $k = 0, \ldots, K-1$ **do**
13  **client** samples batch $\mathcal{B}_i^{(t,k)} \sim D_i$
14  **client** computes $\boldsymbol{w}_i^{(t,k+1)} \leftarrow \boldsymbol{w}_i^{(t,k)} - \eta_c \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})$
15  **return** $\boldsymbol{w}_i^{(t,K)}$

same core structure as FedAvg, but it incorporates two key distinctions. Firstly, each participating client $i \in \mathcal{S}^{(t)}$ communicates the local model update $\Delta_i^{(t)} \triangleq (\boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t)})$ to the server, as opposed to sending the model $\boldsymbol{w}_i^{(t,K)}$ itself (Line 7). Secondly, the server aggregates the client model updates $\{\Delta_i^{(t)}\}_{i \in \mathcal{S}^{(t)}}$ into a global model update, denoted as $\Delta^{(t)}$ (Line 8). Thirdly, the server leverages the negative of the global update, $-\Delta^{(t)}$ , as a pseudo-gradient and applies it to the global model, rather than directly aggregating the models (see Line 9).[‡]

**Remark 1.2.2** (The global update $\Delta^{(t)}$ can be considered as a global pseudo-gradient)**.** When ClientUpdate() and ServerUpdate() are chosen to be Stochastic Gradient Descent (SGD) and Aggregate() is the average operator, the negative of the global update $\Delta^{(t)}$ can be rewritten in the form of a *global* pseudo-gradient:

$$\Delta^{(t)} = \frac{1}{|\mathcal{S}^{(t)}|} \sum_{i \in \mathcal{S}^{(t)}} \left( \boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t)} \right) = -\frac{1}{|\mathcal{S}^{(t)}|} \sum_{i \in \mathcal{S}^{(t)}} \sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}), \tag{1.5}$$

where by *global pseudo-gradient* we mean the average gradient computed by the participating clients in $\mathcal{S}^{(t)}$ averaged over the local iterations $K$.

FedOpt is a widely used framework for describing and analyzing federated training processes, as illustrated in a recent survey by J. Wang et al. (2021).

[‡]The FedAvg algorithm (McMahan et al., 2017, Algorithm 1) can be derived by configuring ClientUpdate() and ServerUpdate() as Stochastic Gradient Descent (SGD) with a fixed server learning rate of $\eta_s = 1.0$.

**Algorithm 2:** `FedOpt` (Federated Optimization Reddi et al. (2021))

**Input:** client datasets $\{D_i\}$; global rounds $T$; local iterations $K$; client and server learning rates $\eta_c$ and $\eta_s$; `ClientUpdate()`; `ServerUpdate()`; `Aggregate()`
**server** randomly initializes $\boldsymbol{w}^{(1)}$
**for** *global communication round* $t = 1, \dots, T$ **do**
  **server** broadcasts $\boldsymbol{w}^{(t)}$ to the participating clients $\mathcal{S}^{(t)}$
  **for** ***participating client*** $i \in \mathcal{S}^{(t)}$, *in parallel* **do**
    $\boldsymbol{w}_i^{(t,K)} \leftarrow$ `ClientUpdate`$(\boldsymbol{w}^{(t)}, D_i, K, \eta_c)$
    **client** computes model update $\Delta_i^{(t)} \triangleq (\boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t)})$ and sends it to the **server**
  **server** aggregates clients' updates: $\Delta^{(t)} \leftarrow$ `Aggregate`$(\{\Delta_i^{(t)}\}_{i \in \mathcal{S}^{(t)}}, \dots)$
  **server** updates global model: $\boldsymbol{w}^{(t+1)} \leftarrow$ `ServerUpdate`$(\boldsymbol{w}^{(t)}, -\Delta^{(t)}, \eta_s)$
  **return** $\boldsymbol{w}^{(t+1)}$

## 1.3 Client Heterogeneity in Federated Learning

In this section, we examine client heterogeneity, a unique challenge that distinguishes federated learning from typical centralized model training. We focus on statistical and system heterogeneity as fundamental challenges of this thesis. This section explores the current methodologies for tackling these issues, and highlights the novel contributions provided by this manuscript to handle them.

### 1.3.1 Statistical Heterogeneity

In federated learning, client-generated data reflect diverse behaviors and preferences, often resulting in datasets that are not representative of the overall population distribution. This *statistical heterogeneity*, coupled with the uneven distribution of data points among devices, makes the data not identically distributed (non-IID). Statistical heterogeneity provides a dual challenge.

On the one hand, high statistical heterogeneity challenges the learning of a shared model suited for all clients. In fact, as observed in (Marfoq, Neglia, Bellet, Kameni, & Vidal, 2021), in presence of large statistical heterogeneity a global model may perform arbitrarily poorly for some clients.

On the other hand, statistical heterogeneity slows down the convergence of federated learning algorithms like `FedAvg`. While multiple local updates by each client are crucial for communication efficiency, these local iterations can negatively affect the training process, as, in presence of *statistical heterogeneity*, local client models progressively diverge towards client-specific local minimizers (Karimireddy et al., 2020; X. Li, Huang, Yang, Wang, & Zhang, 2020; Reddi et al., 2021; T. Li, Sahu, Zaheer, et al., 2020). To address this issue, the `SCAFFOLD` algorithm (Karimireddy et al., 2020) employs control variates to correct for the client drift in local updates. The `FedProx` algorithm (T. Li, Sahu, Zaheer, et al., 2020) enhances the stability of federated optimization by adding a proximal term to the local objectives. Both algorithms account for statistical heterogeneity.

While this manuscript does not specifically address the problem of statistical heterogeneity, our analyses of federated learning algorithms always deal with statistical heterogeneity and its effects

on convergence. Throughout this thesis, following common approaches (X. Li et al., 2020; T. Li, Sahu, Zaheer, et al., 2020; J. Wang, Liu, Liang, Joshi, & Poor, 2020; J. Wang et al., 2021; Karimireddy et al., 2020; Jhunjhunwala, Sharma, Nagarkatti, & Joshi, 2022), we use two equivalent measures of statistical heterogeneity: $\Gamma$ and $\sigma_g^2$.

**Definition 1.3.1** (Measure of Statistical Heterogeneity: $\Gamma$)**.** Following prior work (X. Li et al., 2020), the measure

$$\Gamma \triangleq F^* - \sum_{i=1}^{N} \alpha_i F_i^* \geq 0 \tag{1.6}$$

defines statistical heterogeneity among clients' local datasets in terms of the difference between the optimum of the global objective and the average of the local optima across all clients.

**Definition 1.3.2** (Measure of Statistical Heterogeneity: $\sigma_g^2$)**.** Following previous works (T. Li, Sahu, Zaheer, et al., 2020; J. Wang et al., 2020, 2021; Karimireddy et al., 2020; Jhunjhunwala et al., 2022), the measure

$$\sigma_g^2 \triangleq \sum_{i=1}^{N} \alpha_i \left\| \nabla F_i(\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*) \right\|^2 \geq 0 \tag{1.7}$$

defines statistical heterogeneity among clients' local datasets in terms of the variability of local and global gradients.

**Remark 1.3.1** (Equivalence of Heterogeneity Measures $\Gamma$ and $\sigma_g^2$)**.** Under the assumption of $L$-smoothness and $\mu$-strong convexity of local objectives, the two measures of heterogeneity—$\Gamma$ and $\sigma_g^2$—are equivalent as shown by the following relationships:

$$\Gamma \triangleq \sum_{i=1}^{N} \alpha_i \left[ F_i(\boldsymbol{w}^*) - F_i^* \right] \leq \frac{1}{2\mu} \sum_{i=1}^{N} \alpha_i \left\| \nabla F_i(\boldsymbol{w}^*) - \nabla F(\boldsymbol{w}^*) \right\|^2 \leq \frac{\sigma_g^2}{2\mu}, \tag{1.8}$$

and

$$\sigma_g^2 \triangleq \sum_{i=1}^{N} \alpha_i \left\| \nabla F_i(\boldsymbol{w}^*) \right\|^2 \leq 2L \sum_{i=1}^{N} \alpha_i \left[ F_i(\boldsymbol{w}^*) - F_i^* \right] \leq 2L\Gamma. \tag{1.9}$$

In Eq. (1.8), the first inequality follows from the Polyak-Lojasiewicz inequality and the second inequality uses Definition 1.3.2. In Eq. (1.9), the first inequality leverages the $L$-smoothness assumption and the second inequality uses Definition 1.3.1.

In both Eqs. (1.6) and (1.7), when the local datasets are identical among clients, the local functions $F_i$ coincide among them and with $F$; $\boldsymbol{w}^*$ is a minimizer of each local function, and $\Gamma = \sigma_g^2 = 0$. Generally, smaller values $\Gamma$ and $\sigma_g^2$ reflect more similar local data distributions and consequently a lower amount of statistical heterogeneity among clients.

### 1.3.2 System Heterogeneity

Clients in federated learning exhibit a varied range of characteristics, with differences in storage capacity, processing resources, and communication capabilities. These disparities originate from

variations in hardware specifications (CPU power, memory capacity), network connectivity types (3G, 4G, 5G, WiFi), and power availability (battery levels) (Verbraeken et al., 2020; Kairouz et al., 2021; Ludwig & Baracaldo, 2022). In this section, we highlight the challenges associated with system heterogeneity across three distinct scenarios: heterogeneity in client participation, in network resources, and in hardware configurations.

#### 1.3.2.1 Heterogeneous Client Availability

Systems constraints influence the clients' availability and activity in the cross-device setting. Client participation is limited by a variety of factors beyond the server control. These include different hardware resources, where CPU and memory usage may be shared among concurrent runtime processes, network constraints such as bandwidth limitations and packet losses, and power availability, with clients often willing to participate in model training only while charging to prevent battery drain (Bonawitz et al., 2019; J. Wang et al., 2021; H. Yang, Zhang, Khanduri, & Liu, 2022; Verbraeken et al., 2020; Kairouz et al., 2021; Ludwig & Baracaldo, 2022). Despite these evidences, a large portion of prior works assumes partial yet homogeneous client participation, neglecting the influence of such heterogeneity on the convergence of federated learning algorithms (X. Li et al., 2020; Karimireddy et al., 2020; T. Li, Sahu, Zaheer, et al., 2020; H. Yang, Fang, & Liu, 2020; Fraboni, Vidal, Kameni, & Lorenzi, 2021; W. Chen, Horváth, & Richtárik, 2022; Rizk, Vlaski, & Sayed, 2022; Cho et al., 2023). We identify and illustrate three main problems caused by heterogeneous client participation.

**Biased global model.** Heterogeneous participation can introduce statistical bias in the global model, favoring clients who participate more frequently. Intuitively, when certain clients participate more than others, the global model may disproportionately represent their local objectives, while disadvantaging clients who participate less. All our contributions, in line with other works (Tan et al., 2022; W. Chen et al., 2022; S. Wang & Ji, 2022; Ribero, Vikalo, & de Veciana, 2023; S. Wang & Ji, 2024), address this problem and, under different assumptions on the clients' participation, propose aggregation strategies that unbias the global update through an appropriate aggregation procedure (Algorithm 2, Line 8).

**Correlated client availability.** The devices eligibility criteria can correlate the client participation patterns over time and among clients. *Temporal correlation* can result from a smartphone that is charged for a few consecutive hours and then unavailable for the rest of the day. *Spatial correlation* refers instead to correlated participation dynamics among clients, which often arises from users' diverse geographical distribution. For example, clients in the same time zone often exhibit similar participation patterns, e.g., due to time-of-day effects. These temporal and spatial correlations negatively affect the convergence of federated learning algorithms, introducing problems such as catastrophic forgetting (Goodfellow, Mirza, Xiao, Courville, & Bengio, 2015; Kemker, McClure, Abitino, Hayes, & Kanan, 2018). However, most of the prior work disregarded the temporal and spatial correlations in the clients' availability patterns. We dedicate Chapter 2 to the analysis of a `FedAvg`-like algorithm under heterogeneous and correlated client availability. Our work leads to the design of the first Correlation-Aware FL algorithm: `CA-Fed`.

**Variance in model performance.** Even if the potential bias is addressed, partial and heterogeneous client participation exacerbates the variability of the learning process. Specifically, the large

weights assigned to the less participating client amplify variations in the magnitude of client updates, leading to increased variance in the learned model and slower convergence. Recent works on global variance reduction leverage, in each communication round, stale model updates for non-participating clients (Gu, Huang, Zhang, & Huang, 2021; H. Yang et al., 2022; Jhunjhunwala et al., 2022; Yan et al., 2024). However, their analysis is limited to homogeneous client participation. With varying client participation, global variance reduction methods must face the challenge of aggregating updates of varying staleness, a complex issue that remains unsolved. We explore this problem in Chapter 3, and design the first Staleness-Aware FL algorithm: `FedStale`.

**Definition 1.3.3.** Throughout this manuscript, following previous works (S. Wang & Ji, 2022, 2024), we model client participation heterogeneity through the *participation probability*:

$$p_i \triangleq \mathbb{E}_{\mathcal{S}^{(t)}} \left[ \mathbb{P}(i \in \mathcal{S}^{(t)}) \right]. \tag{1.10}$$

These probabilities indicate how frequently each client is expected to participate in training and can generally vary among clients.

#### 1.3.2.2 Heterogeneous Network Resources

Cross-device FL applications commonly involve models exchanged on wireless networks, which suffer from transmission losses. The performance of FL algorithms can be severely affected by lossy communication channels, which can exhibit varied characteristics among clients (e.g., path loss, multi-path fading, scattering, interference) (Eriş, Kantarci, & Oktug, 2021; Chandrasekaran et al., 2022; H. H. Yang, Liu, Quek, & Poor, 2020; Ye, Liang, & Li, 2022; Baccarelli, Scarpiniti, Momenzadeh, & Sarv Ahrabi, 2022; M. Chen et al., 2021). Methods like automatic repeat request and forward error correction (Wen, Li, Zeng, Ren, & Huang, 2019; M. Chen et al., 2021; Su, Zhou, Cui, & Liu, 2023) are inadequate in this context: first, packet losses can be unavoidable because out from the server's control; second, targeting high transmission reliability in FL-oriented applications may be sub-optimal as it demands longer training time and resource use. As a result, FL algorithms must deal with packet losses. Chapter 4 considers this scenario, where the losses can be in downlink, uplink, or both, and proposes an Unbiased Pseudo-Gradient Aggregation-based Packet Loss-aware FL algorithm: `UPGA-PL`.

**Further network considerations.** Federated learning faces challenges due to communication cost, which can hinder the scaling of distributed optimization algorithms. This has led to the development of compression schemes to reduce communication overhead (Haddadpour, Kamani, Mokhtari, & Mahdavi, 2021; Beznosikov, Horváth, Richtárik, & Safaryan, 2023; Philippenko & Dieuleveut, 2021; Koloskova, Stich, & Jaggi, 2019). However, these system characteristics also exacerbate challenges like straggler effect and fault tolerance. Techniques to mitigate stragglers include asynchronous communication (Lian, Zhang, Zhang, & Liu, 2018), importance client sampling (W. Chen et al., 2022), and dynamic backup workers (Xu, Neglia, & Sebastianelli, 2021).

#### 1.3.2.3 Heterogeneous Hardware Environments

In FL environments with highly heterogeneous hardware (such as smartphones, IoT devices, edge computing servers, and the cloud), every client would ideally learn a different model architecture, suited to its capabilities. Yet, `FedAvg`-based algorithms train and deploy the same model architecture among the network nodes. Recent algorithms have been designed enabling federated training

of models of different sizes. Federated distillation (Lin, Kong, Stich, & Jaggi, 2020) leverages the server to distill knowledge across a limited number of models, but requires access to a public dataset representative of the data distribution among clients. Alternative approaches are sub-model training (Horvath et al., 2021; Diao, Ding, & Tarokh, 2020), early exit networks (Nawar, Falavigna, & Brutti, 2023), or a combination of these techniques (Ilhan, Su, & Liu, 2023)—which entail collaboratively training models that share a subset of parameters. On the other hand, at inference time, Cooperative Inference Systems (CISs) (Salem, Castellano, Neglia, Pianese, & Araldo, 2023; Ren et al., 2023) enable less-performing devices to offload parts of their inference tasks to more powerful devices within the network, and therefore leverage their larger models to increase performance. However, current research mainly focuses on model placement optimization inside the network (E. Li, Zeng, Zhou, & Chen, 2019; Zeng, Li, Zhou, & Chen, 2019), and little effort has being done on improving training strategies for learning these models. Chapter 5 proposes a more systematic approach to FL heterogeneous models training in networks, taking into consideration the importance of each model at inference time, and introduces the first Inference-Aware FL algorithm: `Fed-CIS`.

## 1.4 Other Challenges

This section presents an overview of other challenges in federated learning beyond client heterogeneity and highlights ongoing efforts to address them, based on recent surveys T. Li, Sahu, Talwalkar, and Smith (2020); Kairouz et al. (2021); J. Wang et al. (2021). While addressing these issues is crucial for the development of efficient frameworks, the discussed solutions are often independent of the techniques proposed to mitigate client heterogeneity effects.

**Personalization.** Federated learning conventionally presupposes that all clients collaborate to train a shared, global ML model (McMahan et al., 2017; Konečný et al., 2017; Mohri, Sivek, & Suresh, 2019). However, as discussed in Section 1.3.1, the global model may not perform as well for certain clients in the presence of large statistical heterogeneity (Sattler, Müller, & Samek, 2021; Marfoq et al., 2021; Marfoq, Neglia, Vidal, & Kameni, 2022). Consider, for example, a language modeling task: given the sequence of tokens "I live in," the next word necessarily varies from one client to another. Thus, in some FL applications, learning a personalized model for every client might be necessary. On the other hand, when clients have little training data, the opposite extreme of training a local model for each client might also perform poorly. The limits of both purely local and global models raise a fundamental question: what is the best trade-off between personalization and federation, and how can this balance be achieved? FL literature provided several algorithms that compromise between the extremes of purely global and local training paradigms. Sattler et al. (2021); Mansour, Mohri, Ro, and Suresh (2020); Y. Deng, Kamani, and Mahdavi (2020a) group clients in clusters and train a FL model for each cluster. Collins et al. (Collins, Hassani, Mokhtari, & Shakkottai, 2021) examine this problem assuming that clients share a global feature representation. Even, Massoulié, and Scaman (2022); S. Ding and Wang (2022) introduce a personalized federated learning algorithm that identifies similarity in clients data. These approaches leverage prior knowledge regarding some measure of distance between local data distributions, tough to obtain due to client data confidentiality. In Chapters 2, 3 and 5, we address the training of purely global ML models. In Chapter 5, we focus on personalized model architectures for different clients, motivated by system heterogeneity (Section 1.3.2).

**Fairness.** Algorithmic fairness (Kearns & Roth, 2019; Pessach & Shmueli, 2022) is a research field that aims to mitigate the unintended effects of ML models on individuals or sensitive groups (e.g., ethnicities, genders, and religions). It is crucial to produce models that ensure fair performance beyond the average accuracy measure and maintain an adequate quality of service for all clients. Fairness issues may be exacerbated in federated learning due to statistical and system heterogeneity (Sections 1.3.1 and 1.3.2). Defining a notion of fairness is tough. By targeting uniformity of model performance distribution, some works propose min-max optimization to optimize model performance under worst-case data distributions (Hashimoto, Srivastava, Namkoong, & Liang, 2018; Y. Deng, Kamani, & Mahdavi, 2020b; Mohri et al., 2019), while others propose alternative objectives to the one in Eq. (1.2) that reweight local losses less aggressively (e.g., a hybrid between the "per-client" and "per-sample" fairness critera discussed in Section 1.2.1), allowing for a more flexible tradeoff between accuracy and fairness (Zhang, Li, Robles-Kelly, & Kankanhalli, 2020). Client selection algorithms, including our correlation-aware FL algorithm `CA-Fed` (Chapter 2, Algorithm 7), can mitigate unfairness and participation bias by selecting representative devices, which could produce more informative model updates and speed up convergence (W. Chen et al., 2022; Cho, Wang, & Joshi, 2020; Y. Deng et al., 2020b). However, finding the underlying data distributions among clients may be prohibitively expensive. Additionally, FL algorithms must be fair to hardware heterogeneity by guaranteeing adequate performance for "less performing" and "less participating" clients, particularly in cross-device environments where fairness concerns are prominent. Our `CA-Fed` carefully handles the participation of the "less avaiable" and "higly correlated" clients, targeting an overall greater accuracy for all clients. We discuss `CA-Fed`'s fairness in Appendix A. There are still several open problems for fairness in FL, such as understanding the connections between current FL fairness notions and studying the interactions between fairness and other objectives like personalization and privacy.

**Robustness and Privacy.** Machine learning is vulnerable to attacks and failures, including data distribution shifts, adversarial examples, and data poisoning (Athalye, Carlini, & Wagner, 2018). Federated learning, where millions of client devices participate in the training process, exposes the global model to new vulnerabilities (Kairouz et al., 2021). Robustness and privacy are crucial for the trustworthy, practical deployment of federated learning. Adversaries may target decreasing global accuracy or altering model behavior on a minority of clients. They may also act as eavesdroppers, attempting to obtain sensitive information about clients by simply querying the final model (Baruch, Baruch, & Goldberg, 2019). Known vulnerabilities include model inversion and membership inference attacks, which target the extraction of sensitive information about individual data contributors (Nasr, Shokri, & Houmansadr, 2019). Attacks are not always the result of explicit adversaries, but might originate from malfunctioning or excessively heterogeneous clients. In this context, statistical heterogeneity (Section 1.3.1) poses significant challenges, as it is hard to distinguish a client with unique data from an expert attacker. Defenses mechanisms are typically statistical-based. Differential privacy provides a mathematical foundation for tracking and limiting information leakage throughout federated learning communication rounds. Robust aggregation (Pillutla, Kakade, & Harchaoui, 2022) is used to mitigate the effect of corrupt client updates (e.g., by measuring gradient dissimilarities). As federated learning evolves, robustness and privacy challenges continue to be key research topics, to protect user data while leveraging decentralized machine learning advantages (Kairouz et al., 2021; J. Wang et al., 2021).

**Trade-off between Personalization, Fairness, Robustness and Privacy.** The balance between personalization, fairness, robustness, and privacy is crucial. Personalization targets ML models that perform best for individual clients, while differential privacy compromises performance to prevent the model from retaining client information. Fairness ensures the model performs well for clients with data distributions not aligned with the average population, while robustness prevents these clients from impacting the system's behavior. These constraints can potentially conflict, making future research an interesting area to optimize this delicate equilibrium.

## 1.5 Summary of the Thesis Contributions

Acknowledging the challenges presented in Sections 1.3 and 1.4, this section clarifies the main contributions of this thesis in addressing client heterogeneity.

### Contribution #1. Heterogeneous and Correlated Client Availability

**Problem**: Heterogeneous Client Availability; Correlated Client Availability (Section 1.3.2.1)

**Research Gap**: Previous work examined the impact of temporal correlation in the data sampling process both in centralized (Sun, Sun, & Yin, 2018; Doan, Nguyen, Pham, & Romberg, 2020a, 2020b) and distributed (Doan, 2020) settings, studying a variant of stochastic gradient descent where samples are drawn according to a Markov chain. In contrast, our work does not assume a correlation in the data sampling process but rather in the clients' availability. The heterogeneous client availability in federated learning, due to network and resource constraints, can bias the learning process.

**Main Contribution**:

- We formalize the optimization-bias trade-off induced by heterogeneous client participation by decomposing the convergence error into an optimization error (related to convergence speed) and a bias error (indicative of model quality). By minimizing the optimization error, we show that allocating larger aggregation weights to the "more participating" clients accelerates convergence.
- We explore the impact of temporal and spatial correlation on client availability, the former due to correlated client activity across time and the latter due to clients' correlated geographic distributions. By modeling client availability as a finite-state Markov chain, we prove that correlation has a detrimental effect on convergence. Specifically, it slows the convergence rate to within a logarithmic factor of the corresponding bound observed for independent client availability; this logarithmic factor is naturally related to the geometric mixing time of the Markov chain. We also find that lower aggregation weights for "highly correlated" clients accelerate convergence.
- We finally present the first Correlation-Aware FL algorithm, `CA-Fed`, to optimize the bias-variance trade-off and thus obtain faster convergence. `CA-Fed` dynamically adjusts the aggregation weight allocated to each client and selectively excludes clients with high temporal correlation and low availability. Experimental evaluations on varied datasets confirm the effectiveness of `CA-Fed` compared to state-of-the-art approaches.

**Publications**:

- **Angelo Rodio**, Francescomaria Faticanti, Othmane Marfoq, Giovanni Neglia, and Emilio Leonardi. "Federated Learning under Heterogeneous and Correlated Client Availability". In: *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications*. 2023, pp. 1–10. DOI: 10.1109/INFOCOM53939.2023.10228876. Online: https://ieeexplore.ieee.org/document/10228876
- **Angelo Rodio**, Francescomaria Faticanti, Othmane Marfoq, Giovanni Neglia, and Emilio Leonardi. "Federated Learning under Heterogeneous and Correlated Client Availability". In: *IEEE/ACM Transactions on Networking*. 2023, vol. 32, no. 2, pp. 1451-1460. DOI: 10.1109/TNET.2023.3324257. Online: https://ieeexplore.ieee.org/document/10292582. Supplementary material: https://ieeexplore.ieee.org/ielx7/90/10505042/10292582/supp1-3324257.pdf?arnumber=10292582

### Contribution #2. Leveraging Stale Updates for Non-Participating Clients

**Problem**: Heterogeneous Client Participation; Variance in Model Performance (Section 1.3.2.1)

**Research Gap**: Variance reduction methods that leverage stale model updates from non-participating clients, like `FedVARP`, are often expected to outperform simpler algorithms like `FedAvg` under heterogeneous client participation (Jhunjhunwala et al., 2022; S. Wang & Ji, 2024), yet theoretical support is limited to the homogeneous participation setting (Jhunjhunwala et al., 2022, Theorem 2), and empirical results lack definitive conclusions (S. Wang & Ji, 2024, Table 5). Under heterogeneous client participation, global variance reduction methods, including `FedVARP`, aggregate updates of varying staleness—a problem still unexplored.

**Main Contributions**:

- We analytically and experimentally reject the belief that `FedVARP` consistently outperforms `FedAvg`, suggesting that leveraging stale updates can be beneficial or detrimental depending on client data and participation heterogeneity.
- We propose the first Staleness-Aware FL algorithm, `FedStale`, that updates the global model using a convex, unbiased combination of fresh and stale updates, parameterized by a weight $\beta$. `FedStale` spans from `FedAvg` ($\beta = 0$, only fresh updates) to `FedVARP` ($\beta = 1$, equal weighting of fresh and stale updates). We provide guidelines for tuning the parameter $\beta$ to match specific data and participation heterogeneity conditions.
- We evaluate `FedAvg`, `FedVARP`, and `FedStale` across various client data and participation heterogeneity levels, finding that `FedStale` outperforms both alternatives in most settings.

**Publications**:

- **Angelo Rodio** and Giovanni Neglia. "FedStale: leveraging Stale Clients Updates in Federated Learning." *arXiv preprint*. 2024. Online: https://arxiv.org/abs/2405.04171
- **Angelo Rodio**. "The mupliple facets of Variance Reduction in Federated Learning". To appear in: *ACM SIGMETRICS Performance Evaluation Review*. 2024.

### Contribution #3. Federated Learning in Lossy Communication Channels

**Problem**: Heterogeneous Network Resources (Section 1.3.2.2)

**Research Gap**: Prior work has studied the convergence of FL algorithms under various channel assumptions (H. H. Yang et al., 2020; Ye et al., 2022; Baccarelli et al., 2022; M. Chen et al., 2021). Directly aggregating client models shared on wireless networks led to a non-vanishing error due to lossy channels, preventing convergence to the optimal model (M. Chen et al., 2021). To mitigate this error and control packet losses, strategies such as increasing transmission power, allocating more radio resource blocks, and implementing retransmission or error correction techniques have been proposed. losses (M. Chen et al., 2021; Wen et al., 2019; Su et al., 2023).

**Main Contribution**:

- Challenging standard mitigation strategies for packet losses, we verify that FL algorithms in heterogeneous, asymmetric lossy channels can achieve performance comparable to ideal, lossless environments in both theory and practice.
- Our Packet-Losses Aware FL algorithm, `UPGA-PL`, differs from `FedAvg` (Algorithm 1) by incorporating a pseudo-gradient step in place of direct model averaging and adjusting the aggregation weights to account for heterogeneous lossy channels.
- Empirical results indicate that `UPGA-PL`, under lossy channels, matches the ideal, lossless performance within a limited number of additional communication rounds.

**Publication**:

- **Angelo Rodio**, Giovanni Neglia, Fabio Busacca, Stefano Mangione, Sergio Palazzo, Francesco Restuccia, and Ilenia Tinnirello. "Federated Learning with Packet Losses." In: *IEEE 2023 26th International Symposium on Wireless Personal Multimedia Communications (WPMC)*. 2023, pp. 1-6. DOI: 10.1109/WPMC59531.2023.10338845. Online: https://ieeexplore.ieee.org/document/10338845

### Contribution #4. Cooperative Inference Systems: The Case of Early Exit Networks

**Problem**: Heterogeneous Hardware Environments (Section 1.3.2.3)

**Research Gap**: The literature on joint training of ML models of various sizes remains understudied, and existing training algorithms overlook how these models will be used at inference time (Horvath et al., 2021; Diao et al., 2020; Nawar et al., 2023). Additionally, research on Collaborative Inference Systems (CISs) often assumes that these models are pre-trained, focusing instead on optimizing their placement within the network (Salem et al., 2023; Ren et al., 2023).

**Main Contributions**:

- We reformulate the FL problem (Eq. 1.2) taking into account the inference requests for each sub-model inside the CIS. During training, we allocate larger weights to the models expected to handle a larger volume of inference requests.
- We present the first Inference-Aware FL algorithm, `Fed-CIS`, that enables computationally stronger clients to help weaker ones during training based on predefined probabilities.

Analyzing `Fed-CIS`'s convergence, we identify a trade-off between generalization, bias, and optimization errors, and we provide practical algorithmic configuration guidelines.[§]

- Experimental results show our algorithm outperforms state-of-the-art methods in scenarios with heterogeneous inference request rates or data availability among clients.

**Publication**:

- Caelin Kaplan, **Angelo Rodio**, Tareq Si Salem, Chuan Xu and Giovanni Neglia. "Federated Learning for Cooperative Inference Systems: The Case of Early Exit Networks." *arXiv preprint*. 2024. Online: https://arxiv.org/abs/2405.04249

## 1.6 Publications

### Published

- **Angelo Rodio**, Francescomaria Faticanti, Othmane Marfoq, Giovanni Neglia, and Emilio Leonardi. "Federated Learning under Heterogeneous and Correlated Client Availability". In: *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications*. 2023, pp. 1–10. DOI: 10.1109/INFOCOM53939.2023.10228876. Online: https://ieeexplore.ieee.org/document/10228876

- **Angelo Rodio**, Francescomaria Faticanti, Othmane Marfoq, Giovanni Neglia, and Emilio Leonardi. "Federated Learning under Heterogeneous and Correlated Client Availability". In: *IEEE/ACM Transactions on Networking*. 2023, vol. 32, no. 2, pp. 1451-1460. DOI: 10.1109/TNET.2023.3324257. Online: https://ieeexplore.ieee.org/document/10292582. Supplementary material: https://ieeexplore.ieee.org/ielx7/90/10505042/10292582/supp1-3324257.pdf?arnumber=10292582

- **Angelo Rodio**, Giovanni Neglia, Fabio Busacca, Stefano Mangione, Sergio Palazzo, Francesco Restuccia, and Ilenia Tinnirello. "Federated Learning with Packet Losses." In: *IEEE 2023 26th International Symposium on Wireless Personal Multimedia Communications (WPMC)*. 2023, pp. 1-6. DOI: 10.1109/WPMC59531.2023.10338845. Online: https://ieeexplore.ieee.org/document/10338845

- **Angelo Rodio**. "The mupliple facets of Variance Reduction in Federated Learning". To appear in: *ACM SIGMETRICS Performance Evaluation Review*. 2024.

### Submitted

- **Angelo Rodio** and Giovanni Neglia. "FedStale: leveraging Stale Clients Updates in Federated Learning." *arXiv preprint*. 2024. Online: https://arxiv.org/abs/2405.04171

- Caelin Kaplan, **Angelo Rodio**, Tareq Si Salem, Chuan Xu and Giovanni Neglia. "Federated Learning for Cooperative Inference Systems: The Case of Early Exit Networks." *arXiv preprint*. 2024. Online: https://arxiv.org/abs/2405.04249

[§]The author's contribution was primarily focused on the convergence analysis.

## 1.7 Thesis Outline

The remainder of this thesis is structured as follows. Chapters 2 and 3 examine the variance-bias trade-off that arises from heterogeneous client participation in federated learning. Chapter 2 identifies correlated client participation as a primary source of variance and introduces the first correlation-aware FL algorithm, `CA-Fed`, to address this problem. Chapter 3 shifts focus away from correlation to explore other variance reduction strategies: `FedStale` is our staleness-aware FL algorithm that leverages stale model updates for non-participating clients. Chapter 4 considers network heterogeneity by investigating the impact of clients' diverse communication channel characteristics in wireless networks. Finally, Chapter 5 tackles hardware heterogeneity and presents our `Fed-CIS` algorithm to train multiple models with varying architectural complexities. The concluding chapter includes the author's comments on interesting problems encountered during this thesis and outlines potential research directions for future work.

# CHAPTER 2

# Heterogeneous and Correlated Client Participation

This chapter is based on our works Rodio, Faticanti, Marfoq, Neglia, and Leonardi (2023a), published in IEEE Conference on Computer Communications (IEEE INFOCOM 2023), and Rodio, Faticanti, Marfoq, Neglia, and Leonardi (2023b), in IEEE/ACM Transactions on Networking.

## 2.1 Motivation

As observed in Chapter 1 (particularly in Section 1.3.2.1), the availability of clients in federated learning can be dictated by external circumstances that are outside the control of the orchestrating server and hard to predict. For instance, only smartphones that are idle, under charge, and connected to broadband networks are commonly allowed to participate in the training process (McMahan et al., 2017; Bonawitz et al., 2019). These eligibility requirements can make the availability of devices correlated over time and space (Eichner et al., 2019; Y. Ding et al., 2020; Zhu et al., 2021; Doan, 2020).

- *Temporal correlation* may origin, for example, from a smartphone being under charge for a few consecutive hours and then ineligible for the rest of the day. Similarly, the activity of a sensor powered by renewable energy may depend on natural phenomena intrinsically correlated over time (e.g., solar light).
- *Spatial correlation* refers instead to correlation across different clients, which often emerges as consequence of users' different geographical distribution. For instance, clients in the same time zone often exhibit similar availability patterns, e.g., due to time-of-day effects.

Temporal correlation in the data sampling procedure is known to negatively affect the performance of ML training even in the centralized setting (Doan et al., 2020a; Sun et al., 2018) and can potentially lead to *catastrophic forgetting*: the data used during the final training phases can have a disproportionate effect on the final model, "erasing" the memory of previously learned information (McCloskey & Cohen, 1989; Kirkpatrick et al., 2017). Catastrophic forgetting has also been observed in FL, where clients in the same geographical area have more similar local data distributions and clients' availability follows a cyclic daily pattern (leading also to spatial correlation) (Eichner et al., 2019; Y. Ding et al., 2020; Zhu et al., 2021; Tang et al., 2022). Despite

this evidence, a theoretical study of the convergence of FL algorithms under both temporally and spatially correlated client availability is still missing.

This chapter presents the first convergence analysis of `FedAvg` (McMahan et al., 2017) under heterogeneous and correlated client availability. We assume that the clients' availability follows an arbitrary finite-state Markov chain, modeling both temporal and spatial correlation while maintaining analytical tractability. Our theoretical analysis provides valuable insights by (i) quantitatively measuring the negative impact of correlation on the algorithm's convergence rate through a novel additional term that depends on the spectral properties of the Markov chain, and (ii) highlighting an important trade-off between two conflicting objectives: slow convergence to the optimal model and fast convergence to a biased model that minimizes a different objective function from the initial target. To leverage this trade-off, we propose `CA-Fed`, an algorithm which achieves an optimal balance between maximizing convergence speed and minimizing model bias through dynamic adjustment of aggregation weights assigned to clients. Depending on their contribution to the learning process, `CA-Fed` can decide to exclude clients exhibiting low availability and high temporal correlation. Our experimental results demonstrate that excluding such clients is a simple, but effective approach to handle the heterogeneous and correlated client availability in FL. Across synthetic and real datasets, `CA-Fed` consistently outperforms the state-of-the-art methods `F3AST` (Ribero et al., 2023) and `AdaFed` (Tan et al., 2022) in terms of test accuracy. These results underscore the importance of optimizing the training process to leverage available client resources effectively and mitigate the impact of less available and correlated clients, a task successfully accomplished by `CA-Fed`.

The remainder of this chapter is organized as follows. Section 2.2 introduces the problem of correlated client availability in FL and discusses the main related works. Section 2.3 provides a convergence analysis of `FedAvg` under heterogeneous and correlated client availability. `CA-Fed`, our correlation-aware FL algorithm, is presented in Section 2.4. We evaluate `CA-Fed` in Section 2.5, comparing it with state-of-the-art methods on synthetic and real-world data. Section 2.6 concludes the chapter. Appendix A provides detailed proofs and further discussions on `CA-Fed`.

## 2.2 Problem Description and Background

This chapter considers a similar optimization problem to the one presented in Chapter 1, Section 1.2.1. However, we constrain the search for the parameter vector $\boldsymbol{w}$ within the parameter space $W \subseteq \mathbb{R}^d$:

$$\min_{\boldsymbol{w} \in W \subseteq \mathbb{R}^d} \left[ F(\boldsymbol{w}) \triangleq \sum_{i=1}^{N} \alpha_i F_i(\boldsymbol{w}) \right]. \tag{2.1}$$

We recall that the coefficients $\boldsymbol{\alpha} = (\alpha_i)_{i \in \mathcal{N}}$ are positive numbers satisfying the condition $\sum_{i=1}^{N} \alpha_i = 1$. They represent the *target importance* that the central server assigns to each client $i$.

Under proper assumptions, precised in Section 2.3, Problem (2.1) admits a unique solution. We use $\boldsymbol{w}^*$ (respectively $F^*$) to denote the minimizer (respectively the minimum value) of $F$. Moreover, for $i \in \mathcal{N}$, $F_i$ admits a unique minimizer. We use $\boldsymbol{w}_i^*$ (respectively $F_i^*$) to denote the minimizer (respectively the minimum value) of $F_i$.

As observed in Chapter 1, Problem (2.1) is commonly solved through iterative algorithms (McMahan et al., 2017; J. Wang et al., 2021) requiring multiple communication rounds between the server and the clients. Here, we consider the `FedOpt` algorithm (Section 1.2.2, Algorithm 2) redefining the `ClientUpdate()`, `Aggregate()` and `ServerUpdate()` procedures as follows:

- Client $i \in \mathcal{S}^{(t)}$ updates the global model with its local data through $K \geq 1$ steps of local Stochastic Gradient Descent (SGD):

$$\boldsymbol{w}_i^{(t,k+1)} = \boldsymbol{w}_i^{(t,k)} - \eta_c \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) \quad k = 0, \dots, K-1, \tag{2.2}$$

  where $\eta_c > 0$ is an appropriately chosen learning rate, referred to as *local learning rate*; $\mathcal{B}_i^{(t,k)}$ is a random batch sampled from client-$i$'s local dataset at round $t$ and step $k$; $\nabla F_i(\cdot, \mathcal{B}) \triangleq \frac{1}{|\mathcal{B}|} \sum_{z \in \mathcal{B}} \nabla f(\cdot, z)$ is an unbiased estimator of the local gradient $\nabla F_i$.

- The server computes $\Delta^{(t)} \triangleq \sum_{i \in \mathcal{S}^{(t)}} q_i \cdot \Delta_i^{(t)}$, a weighted average of the clients' local updates with non-negative *aggregation weights* $\boldsymbol{q} = (q_i)_{i \in \mathcal{N}}$. The choice of the aggregation weights defines an aggregation strategy (we will discuss different aggregation strategies later).

- The aggregated update $\Delta^{(t)}$ can be interpreted as a proxy for $-\nabla F(\boldsymbol{w}^{(t,0)})$; the server applies it to the global model:

$$\boldsymbol{w}^{(t+1,0)} = \mathbf{Proj}_W(\boldsymbol{w}^{(t,0)} + \eta_s \cdot \Delta^{(t)}), \tag{2.3}$$

  where $\mathbf{Proj}_W(\cdot)$ denotes the projection over the set $W$, and $\eta_s > 0$ is an appropriately chosen learning rate, referred to as the *server learning rate*.*

The aggregate update $\Delta^{(t)}$ is generally a biased estimator of the pseudo-gradient $-\nabla F(\boldsymbol{w}^{(t,0)})$, to which each client $i$ contributes proportionally to its frequency of appearance in the set $\mathcal{S}^{(t)}$ and its aggregation weight $q_i$. More specifically, under proper assumptions specified in Section 2.3, we will prove in Theorem 2.3.3 that the update rule described by Eqs. (2.2) and (2.3) converges to the unique minimizer of a biased global objective $F_B$. This objective function depends both on the clients' participation (i.e., on the sequence $(\mathcal{S}^{(t)})_{t>0}$) and on the aggregation strategy (i.e., on $\boldsymbol{q} = (q_i)_{i \in \mathcal{N}}$):

$$F_B(\boldsymbol{w}) \triangleq \sum_{k=1}^{N} p_i F_i(\boldsymbol{w}), \text{ with } p_i \triangleq \frac{\pi_i q_i}{\sum_{h=1}^{N} \pi_h q_h}, \tag{2.4}$$

where $\pi_i$ represents the asymptotic participation of client $i$, defined as $\pi_i \triangleq \lim_{t \to +\infty} \mathbb{P}(i \in \mathcal{S}^{(t)})$. We denote $\boldsymbol{\pi} = (\pi_i)_{i \in \mathcal{N}}$. Moreover, the coefficients $\boldsymbol{p} = (p_i)_{i \in \mathcal{N}}$ in Eq. (2.4) can be interpreted as the *biased importance* the server is giving to each client $i$ during training, in general different from the *target importance* $\boldsymbol{\alpha}$. In what follows, $\boldsymbol{w}_B^*$ (respectively $F_B^*$) denotes the minimizer (respectively the minimum value) of $F_B$.

*The aggregation rule (2.3) has been considered also in other works, e.g., (Nichol, Achiam, & Schulman, 2018; Reddi et al., 2021; J. Wang et al., 2021). In other FL algorithms, e.g. `FedAvg` (Algorithm 1), the server computes an average of clients' local models. This aggregation rule can be obtained with minor changes to Eq. (2.3).

In some large-scale FL applications, like training Google keyboard next-word prediction models, each client participates in training at most for one round. The orchestrator usually selects a few hundred clients at each round for a few thousand rounds (e.g., see Kairouz et al. (2021, Table 2)), but the available set of clients may include hundreds of millions of Android devices. In this scenario, it is difficult to address the potential bias unless there is some a-priori information about each client's availability. Anyway, FL can be used by service providers with access to a much smaller set of clients (e.g., smartphone users that have installed a specific app). In this case, a client participates multiple times in training: the orchestrating server may keep track of each client's availability and try to compensate for the potentially dangerous heterogeneity in their availability.

Much previous effort on federated learning (McMahan et al., 2017; X. Li et al., 2020; T. Li, Sahu, Zaheer, et al., 2020; W. Chen et al., 2022; Fraboni et al., 2021; Tang et al., 2022; Tan et al., 2022; Ribero et al., 2023) considered this problem and, under different assumptions on the clients' availability (i.e., on $(\mathcal{S}^{(t)})_{t>0}$), designed aggregation strategies that unbias $\Delta^{(t)}$ through an appropriate choice of $\boldsymbol{q}$. Reference X. Li et al. (2020) provides the first analysis of `FedAvg` on non-iid data under partial client availability. Their analysis covers both the case when available clients are sampled uniformly at random without replacement from $\mathcal{N}$ and assigned aggregation weights equal to their target importance (as assumed in McMahan et al. (2017)), and the case when available clients are sampled iid with replacement from $\mathcal{N}$ with probabilities $\boldsymbol{\alpha}$ and assigned equal weights (as assumed in T. Li, Sahu, Zaheer, et al. (2020)). However, references (McMahan et al., 2017; X. Li et al., 2020; T. Li, Sahu, Zaheer, et al., 2020) do not address the variance induced by the clients stochastic availability. The authors of W. Chen et al. (2022) reduce such variance by considering only the clients with important updates, as measured by the value of their norm. References Tang et al. (2022) and Fraboni et al. (2021) reduce the aggregation variance through clustered and soft-clustered sampling, respectively.

Some recent works (Tan et al., 2022; Ribero et al., 2023; Jee Cho, Wang, & Joshi, 2022) do not actively pursue the optimization of the unbiased objective. Instead, they derive bounds for the convergence error and propose heuristics to minimize those bounds, potentially introducing some bias. Our work follows a similar development: we compare our algorithm with `F3AST` from Ribero et al. (2023) and `AdaFed` from Tan et al. (2022).

The novelty of our study is in considering the spatial and temporal correlation in clients' availability dynamics. As discussed in the introduction, such correlations are also introduced by clients' eligibility criteria, e.g., smartphones being under charge and connected to broadband networks. The effect of correlation has been ignored until now, probably due to the additional complexity in studying FL algorithms' convergence. To the best of our knowledge, the only exception is Ribero et al. (2023), which scratches the issue of spatial correlation by proposing two different algorithms for the case when clients' availabilities are uncorrelated and for the case when they are positively correlated (there is no smooth transition from one algorithm to the other as a function of the degree of correlation).

The effect of temporal correlation on *centralized* stochastic gradient methods has been addressed in Sun et al. (2018); Doan et al. (2020a, 2020b); Doan (2020): these works study a variant of stochastic gradient descent where samples are drawn according to a Markov chain. Reference Doan (2020) extends its analysis to a FL setting where each client draws samples according to a Markov chain. In contrast, our work does not assume a correlation in the data sampling but

rather in the client's availability. Nevertheless, some of our proof techniques are similar to those used in this line of work and, in particular, we rely on some results in Sun et al. (2018).

## 2.3 Convergence Analysis under Markov Availability Assumption

We consider a time-slotted system where a slot corresponds to a single FL communication round. We assume that clients' availability over the timeslots $t \in \mathbb{N}$ follows a discrete-time Markov chain $(\mathcal{S}^{(t)})_{t\geq 0}$.[†]

**Assumption 1.** *The Markov chain $(\mathcal{S}^{(t)})_{t\geq 0}$ on the $M$-finite state space $\mathcal{M}$ is time-homogeneous, irreducible, and aperiodic. It has transition matrix $\boldsymbol{P}$, stationary distribution $\boldsymbol{\rho}$, and has state distribution $\boldsymbol{\rho}$ at time $t = 0$.*

Markov chains have already been used in the literature to model the dynamics of stochastic networks where some nodes or edges in the graph can switch between active and inactive states (Meyers & Yang, 2021; Olle, Yuval, & Jeffrey, 1997). The previous Markovian assumption, while allowing a great degree of flexibility, still guarantees the analytical tractability of the system. The distance dynamics between the current and the stationary distributions of the Markov process can be characterized in terms of the spectral properties of its transition matrix $\boldsymbol{P}$ (Levin & Peres, 2017). Let $\bar{\lambda}_2(\boldsymbol{P})$ denote the the second largest module of the eigenvalues of $\boldsymbol{P}$. Previous work (Sun et al., 2018) has shown that:

$$\max_{i,j\in[M]} \left| [\boldsymbol{P}^t]_{i,j} - \rho_j \right| \leq C_P \cdot \lambda(\boldsymbol{P})^t, \qquad \text{for } t \geq T_P, \tag{2.5}$$

where the parameters $\lambda(\boldsymbol{P}) \triangleq (\bar{\lambda}_2(\boldsymbol{P}) + 1)/2$, $C_P$, and $T_P$ are positive constants whose values are defined in (Sun et al., 2018, Lemma 1) and reported for completeness in Appendix B.B, Lemma B.16.[‡] Note that $\lambda(\boldsymbol{P})$ quantifies the correlation of the Markov process $(\mathcal{S}^{(t)})_{t\geq 0}$: the closer $\lambda(\boldsymbol{P})$ is to one, the slower the Markov chain converges to its stationary distribution.

In our analysis, we make the following additional assumptions.

**Assumption 2.** *The hypothesis class $W$ is convex and compact with diameter $\mathrm{diam}(W)$, and contains the minimizers $\boldsymbol{w}^*, \boldsymbol{w}_B^*, \boldsymbol{w}_i^*$ in its interior.*

The following assumptions concern clients' local objective functions $\{F_i\}_{i\in\mathcal{N}}$. Assumptions 3 and 4 are standard in the literature on convex optimization (Bottou, Curtis, & Nocedal, 2018, Sections 4.1, 4.2). Assumption 5 is a standard hypothesis in the analysis of federated optimization algorithms (J. Wang et al., 2021, Section 6.1).

**Assumption 3** (L-smoothness)**.** *The local functions $\{F_i\}_{i=1}^N$ have L-Lipschitz continuous gradients: $F_i(\boldsymbol{v}) \leq F_i(\boldsymbol{w}) + \langle \nabla F_i(\boldsymbol{w}), \boldsymbol{v} - \boldsymbol{w}\rangle + \frac{L}{2}\|\boldsymbol{v} - \boldsymbol{w}\|_2^2, \forall \boldsymbol{v}, \boldsymbol{w} \in W$.*

**Assumption 4** (Strong convexity)**.** *The local functions $\{F_i\}_{i=1}^N$ are $\mu$-strongly convex: $F_i(\boldsymbol{v}) \geq F_i(\boldsymbol{w}) + \langle \nabla F_i(\boldsymbol{w}), \boldsymbol{v} - \boldsymbol{w}\rangle + \frac{\mu}{2}\|\boldsymbol{v} - \boldsymbol{w}\|_2^2, \forall \boldsymbol{v}, \boldsymbol{w} \in W$.*

[†] In Section 2.3.3.1 we will focus on the case where this chain is the superposition of $N$ independent Markov chains, one for each client.

[‡] Note that (2.5) holds for different definitions of $\lambda(\boldsymbol{P})$ as long as $\lambda(\boldsymbol{P}) \in (\bar{\lambda}_2(\boldsymbol{P}), 1)$. The specific choice for $\lambda(\boldsymbol{P})$ changes the values of $C_P$ and $T_P$.

**Assumption 5** (Bounded variance)**.** *The variance of stochastic gradients in each device is bounded:* $\mathbb{E}_{\mathcal{B}}\|\nabla F_i(\boldsymbol{w},\mathcal{B}) - \nabla F_i(\boldsymbol{w})\|^2 \leq \sigma_i^2,\ i = 1,\dots,N.$

Assumptions 2–5 imply the following properties for the local functions, described by Lemma 2.3.1 (proof in Appendix B).

**Lemma 2.3.1.** *Under Assumptions 2–5, there exist constants $D$, $G$, and $H > 0$, such that, for all $\boldsymbol{w} \in W$ and $i \in \mathcal{N}$, we have:*

$$\|\nabla F_i(\boldsymbol{w})\| \leq D, \tag{2.6}$$

$$\mathbb{E}_{\mathcal{B}}\|\nabla F_i(\boldsymbol{w},\mathcal{B})\|^2 \leq G^2, \tag{2.7}$$

$$|F_i(\boldsymbol{w}) - F_i(\boldsymbol{w}_B^*)| \leq H. \tag{2.8}$$

Similarly to other works (X. Li et al., 2020; T. Li, Sahu, Zaheer, et al., 2020; J. Wang et al., 2020, 2021), we introduce a metric to quantify the heterogeneity of clients' local datasets, typically referred to as *statistical heterogeneity*:

$$\Gamma \triangleq \max_{i\in\mathcal{N}}\{F_i(\boldsymbol{w}^*) - F_i^*\}. \tag{2.9}$$

If the local datasets are identical, the local functions $\{F_i\}_{i\in\mathcal{N}}$ coincide among them and with $F$, $\boldsymbol{w}^*$ is a minimizer of each local function, and $\Gamma = 0$. In general, $\Gamma$ is smaller the closer the distributions the local datasets are drawn from.

### 2.3.1 Optimization-Bias Error Decomposition

**Theorem 2.3.2** (Decomposing the total error)**.** *Let $\kappa \triangleq L/\mu$. Under Assumptions 2–4, the optimization error of the target global objective $\epsilon = F(\boldsymbol{w}) - F^*$ can be bounded as follows:*

$$\epsilon \leq 2\kappa^2(\underbrace{F_B(\boldsymbol{w}) - F_B^*}_{\triangleq\epsilon_{opt}} + \underbrace{F(\boldsymbol{w}_B^*) - F^*}_{\triangleq\epsilon_{bias}}). \tag{2.10}$$

*Moreover, let $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}} \triangleq \sum_{i=1}^N (\alpha_i - p_i)^2/p_i$. Then:*

$$\epsilon_{bias} \leq \kappa^2 \cdot \underbrace{\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}} \cdot \Gamma}_{\triangleq\bar{\epsilon}_{bias}}. \tag{2.11}$$

Theorem 2.3.2 (proof in Appendix A) decomposes the error of the target objective ($\epsilon$) as the sum of an optimization error for the biased objective ($\epsilon_{\text{opt}}$) and a bias error ($\epsilon_{\text{bias}}$). The term $\epsilon_{\text{opt}}$, evaluated on the trajectory determined by scheme (2.3), quantifies the optimization error associated with the biased objective $F_B$ and asymptotically vanishes (see Theorem 2.3.3 below). The non-vanishing bias error $\epsilon_{\text{bias}}$ captures the discrepancy between $F(\boldsymbol{w}_B^*)$ and $F^*$. This term is bounded by the chi-square divergence $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}}$ between the target and biased probability distributions $\boldsymbol{\alpha} = (\alpha_i)_{i\in\mathcal{N}}$ and $\boldsymbol{p} = (p_i)_{i\in\mathcal{N}}$, and by $\Gamma$, that quantifies the degree of heterogeneity of the local functions. When all local functions are identical ($\Gamma = 0$), the bias term $\epsilon_{\text{bias}}$ also vanishes. For $\Gamma > 0$, the bias error can still be controlled by the aggregation weights assigned to the devices. In particular, the bias term vanishes when $q_i \propto \alpha_i/\pi_i, \forall i \in \mathcal{N}$. Since it asymptotically cancels the bias error, we refer to this choice as *unbiased aggregation strategy*.

However, in practice, FL training is limited to a finite number of iterations $T$ (typically a few hundreds (Eichner et al., 2019; Kairouz et al., 2021)), and the previous asymptotic considerations may not apply. In this regime, the unbiased aggregation strategy can be sub-optimal, since the minimization of $\epsilon_{\text{bias}}$ not necessarily leads to the minimization of the total error $\epsilon \leq 2\kappa^2(\epsilon_{\text{opt}} + \epsilon_{\text{bias}})$. This motivates the analysis of the optimization error $\epsilon_{\text{opt}}$.

### 2.3.2 Convergence of the Optimization Error $\epsilon_{\text{opt}}$

**Theorem 2.3.3** (Convergence of the optimization error $\epsilon_{\text{opt}}$)**.** *Let Assumptions 1–5 hold and the constants $M, L, D, G, H, \Gamma, \sigma_i, C_P, T_P$, and $\lambda(\boldsymbol{P})$ defined above. Let $Q \triangleq \sum_{i\in\mathcal{N}} q_i$. We require a diminishing step-size $\eta_c^{(t)} > 0$ satisfying:*

$$\eta_1 \leq \frac{1}{2L(1+2KQ)}, \quad \sum_{t=1}^{+\infty} \eta_c^{(t)} = +\infty, \quad \sum_{t=1}^{+\infty} \ln(t) \cdot \left(\eta_c^{(t)}\right)^2 < +\infty. \tag{2.12}$$

*Let $T$ denote the total communication rounds. For $T \geq T_P$, the expected optimization error can be bounded as follows:*

$$\mathbb{E}\left[F_B(\bar{\boldsymbol{w}}^{(T,0)}) - F_B^*\right] \leq \underbrace{\frac{\frac{1}{2}\boldsymbol{q}^\intercal \boldsymbol{\Sigma} \boldsymbol{q} + \upsilon}{\boldsymbol{\pi}^\intercal \boldsymbol{q}} + \psi + \frac{\phi}{\ln(1/\lambda(\boldsymbol{P}))}}{\sum_{t=1}^{T} \eta_c^{(t)}}}_{\triangleq \bar{\epsilon}_{opt}}, \tag{2.13}$$

*where* $\bar{\boldsymbol{w}}^{(T,0)} \triangleq \frac{\sum_{t=1}^{T} \eta_c^{(t)} \boldsymbol{w}^{(t,0)}}{\sum_{t=1}^{T} \eta_c^{(t)}}$*, and*

$$\boldsymbol{\Sigma} \triangleq \operatorname{diag}\left(2(K+1)\pi_i\sigma_i^2 \sum_{t=1}^{+\infty} \left(\eta_c^{(t)}\right)^2\right), \tag{2.14}$$

$$\upsilon \triangleq \frac{2}{K}\operatorname{diam}(W)^2 + \frac{1}{4}MQ \sum_{t=1}^{+\infty} \left(\left(\eta_c^{(t)}\right)^2 + \frac{1}{t^2}\right), \tag{2.15}$$

$$\psi \triangleq \left(4L(1+KQ)\Gamma + 2K^2G^2\right) \sum_{t=1}^{+\infty} \left(\eta_c^{(t)}\right)^2 + H\left(\sum_{t=1}^{T_P-1} \eta_c^{(t)}\right), \tag{2.16}$$

$$\mathcal{J}^{(t)} \triangleq \min\left\{\max\left\{\left\lceil \frac{\ln(2C_P H t)}{\ln(1/\lambda(\boldsymbol{P}))} \right\rceil, T_P\right\}, t\right\}, \tag{2.17}$$

$$\phi \triangleq 2KDGQ \sum_{t=1}^{+\infty} \ln(2C_P H t) \left(\eta^{(t-\mathcal{J}^{(t)})}\right)^2. \tag{2.18}$$

Theorem 2.3.3 (proof in Appendix B) proves convergence of the expected biased objective $F_B$ to its minimum $F_B^*$ under correlated client availability. Our bound (2.13) captures the effect of correlation through the factor $\ln(1/\lambda(\boldsymbol{P}))$: a high correlation worsens the convergence rate. In particular, we found that the numerator of (2.13) has a quadratic-over-linear fractional dependence on $\boldsymbol{q}$. Minimizing $\bar{\epsilon}_{\text{opt}}$ leads, in general, to a different choice of $\boldsymbol{q}$ than minimizing $\bar{\epsilon}_{\text{bias}}$.

### 2.3.3 Minimizing the total error $\epsilon \leq 2\kappa^2(\bar{\epsilon}_{\text{opt}} + \bar{\epsilon}_{\text{bias}})$

Our analysis points out a trade-off between minimizing $\bar{\epsilon}_{\text{opt}}$ or $\bar{\epsilon}_{\text{bias}}$. Our goal is to find the optimal aggregation weights $\boldsymbol{q}^*$ that minimize the upper bound on total error $\epsilon(\boldsymbol{q})$ in (2.10):

$$\begin{aligned} \underset{\boldsymbol{q}}{\text{minimize}} \quad & \bar{\epsilon}_{\text{opt}}(\boldsymbol{q}) + \bar{\epsilon}_{\text{bias}}(\boldsymbol{q}); \\ \text{subject to} \quad & \boldsymbol{q} \geq 0, \\ & \|\boldsymbol{q}\|_1 = Q. \end{aligned} \tag{2.19}$$

In Appendix D we prove that (2.19) is a convex optimization problem, which can be solved with the method of Lagrange multipliers. However, its solution lacks practical utility because the constants in (2.10) and (2.13) (e.g., $L$, $\mu$, $\Gamma$, $C_P$) are in general problem-dependent and difficult to estimate during training. In particular, $\Gamma$ poses particular difficulties as it is defined in terms of the minimizer of the target objective $F$, but the FL algorithm generally minimizes the biased function $F_B$. Moreover, the bound in (2.10), as well as the bound in (J. Wang et al., 2020), diverges when setting some $q_i$ values equal to 0, but this divergence is merely an artifact of the proof technique. For more practical considerations, we present the following result (proof in Appendix C):

**Theorem 2.3.4** (An alternative bound on the bias error $\epsilon_{\text{bias}}$)**.** *Under the same assumptions of Theorem 2.3.2, define* $\Gamma' \triangleq \max_i\{F_i(\boldsymbol{w}_B^*) - F_i^*\}$. *The following result holds:*

$$\epsilon_{bias} \leq 4\kappa^2 \cdot \underbrace{d_{TV}^2(\boldsymbol{\alpha}, \boldsymbol{p}) \cdot \Gamma'}_{\triangleq \bar{\epsilon}'_{bias}}, \tag{2.20}$$

*where* $d_{TV}(\boldsymbol{\alpha}, \boldsymbol{p}) \triangleq \frac{1}{2}\sum_{i=1}^{N}|\alpha_i - p_i|$ *is the total variation distance between the probability distributions* $\boldsymbol{\alpha}$ *and* $\boldsymbol{p}$.

The new constant $\Gamma'$ is defined in terms of $\boldsymbol{w}_B^*$, and then it is easier to evaluate during training. However, $\Gamma'$ depends on $\boldsymbol{q}$, because it is evaluated at the point of minimum of $F_B$. This dependence makes the minimization of the right-hand side of (2.20) more challenging (for example, the corresponding problem is not convex). We study the minimization of the two terms $\bar{\epsilon}_{\text{opt}}$ and $\bar{\epsilon}'_{\text{bias}}$ separately and learn some insights, which we use to design the new FL algorithm `CA-Fed`.

#### 2.3.3.1 Minimizing $\bar{\epsilon}_{\text{opt}}$

The minimization of $\bar{\epsilon}_{\text{opt}}$ is still a convex optimization problem (Appendix E). In particular, at the optimum, non-negative weights are set accordingly to $q_i^* = a(\iota^*\pi_i - \theta^*)$ with $a$ and $\iota^*$ positive constants (Appendix E.B). It follows that clients with smaller availability get smaller weights in the aggregation. In particular, this suggests that clients with the smallest availability can be excluded from the aggregation, leading to the following guideline:

*Guideline A: to accelerate convergence, we can exclude clients with low availability* $\pi_i$ *by setting* $q_i^* = 0$.

This guideline can be justified intuitively: updates from clients with low availability may be too sporadic to allow the FL algorithm to keep track of their local objectives. Their updates act as a

noise slowing down the algorithm's convergence. It may then be advantageous to exclude these clients.

We observe that the choice of the aggregation weights $\boldsymbol{q}$ does not affect the clients' availability process and, in particular, $\lambda(\boldsymbol{P})$. However, if the algorithm excludes some clients, it is possible to consider the state space of the Markov chain that only specifies the availability state of the remaining clients, and this Markov chain may have different spectral properties. For the sake of concreteness, unless otherwise specified, we consider from now on the particular case when the availability of each client $i$ evolves according to a Markov chain $(\mathcal{S}_i^{(t)})_{t\geq 0}$ with transition probability matrix $\boldsymbol{P}_i$ and these Markov chains are all independent (Levin & Peres, 2017, Exercise 12.6). In this case, the aggregate process is described by the product Markov chain $(\mathcal{S}^{(t)})_{t\geq 0}$ with transition matrix $\boldsymbol{P} = \bigotimes_{i\in\mathcal{N}} \boldsymbol{P}_i$ and $\lambda(\boldsymbol{P}) = \max_{i\in\mathcal{N}} \lambda(\boldsymbol{P}_i)$, where $\boldsymbol{P}_i \bigotimes \boldsymbol{P}_j$ denotes the Kronecker product between matrices $\boldsymbol{P}_i$ and $\boldsymbol{P}_j$ (Appendix F.B). In this setting, it is possible to redefine the Markov chain $(\mathcal{S}^{(t)})_{t\geq 0}$ by taking into account the reduced state space defined by the clients with a non-null aggregation weight, i.e., $\boldsymbol{P'} = \bigotimes_{i'\in\mathcal{N}|q_{i'}>0} \boldsymbol{P}_{i'}$ and $\lambda(\boldsymbol{P'}) = \max_{i'\in\mathcal{N}|q_{i'}>0} \lambda(\boldsymbol{P}_{i'})$, which is potentially smaller w.r.t. the case when all clients participate to the aggregation. These considerations lead to the following guideline:

*Guideline B: to accelerate convergence, we can exclude clients with high correlation (high $\lambda(\boldsymbol{P}_i)$) by setting their $q_i^* = 0$.*

Intuition also supports this guideline. Clients with large $\lambda(\boldsymbol{P}_i)$ tend to be available or unavailable for long periods of time. Due to the well-known catastrophic forgetting problem affecting gradient methods (Goodfellow et al., 2015; Kemker et al., 2018), these clients may unfairly steer the algorithm toward their local objective when they appear at the final stages of the training period. Moreover, their participation in the early stages may be useless, as their contribution will be forgotten during their long absence. The FL algorithm may benefit from directly neglecting such clients.

We observe that Guideline B strictly applies to this specific setting where clients' dynamics are independent (and there is no spatial correlation). We do not provide a corresponding guideline for the case when clients are spatially correlated (we leave this task for future research). However, in this more general setting, it is possible to ignore Guideline B but still draw on Guidelines A and C, or still consider Guideline B if the spatially correlated clients can be grouped in clusters, each cluster evolving as an independent Markov chain (see Section 2.5, Paragraph 2.5.2).

#### 2.3.3.2 Minimizing $\bar{\epsilon}'_{\text{bias}}$

The bias error $\bar{\epsilon}'_{\text{bias}}$ in (2.20) vanishes when the total variation distance between the target importance $\boldsymbol{\alpha}$ and the biased importance $\boldsymbol{p}$ is zero, i.e., when $q_i \propto \alpha_i/\pi_i, \forall i \in \mathcal{N}$. Then, after excluding the clients that contribute the most to the optimization error and particularly slow down the convergence (Guidelines A and B), we can assign to the remaining clients an aggregation weight inversely proportional to their availability, such that the bias error $\bar{\epsilon}'_{\text{bias}}$ is minimized.

*Guideline C: to minimize the bias error, we assign $q_i^* \propto \alpha_i/\pi_i$ to the clients not excluded by the previous guidelines.*

## 2.4 Proposed Correlation-Aware FL Algorithm: `CA-Fed`

Guidelines A and B in Section 2.3 suggest that minimizing $\bar{\epsilon}_{\text{opt}}$ can lead to the exclusion of some available clients from the aggregation step (2.3), in particular those with low availability and/or high correlation. For the remaining clients, Guideline C proposes setting their aggregation weight inversely proportional to their availability to reduce the bias error $\bar{\epsilon}'_{\text{bias}}$. Motivated by these insights, we propose `CA-Fed`, a client aggregation strategy that considers the problem of correlated client availability in FL, described in Algorithm 3. `CA-Fed` learns during training which clients to exclude and how to set the aggregation weights of the remaining clients to achieve a good trade-off between $\bar{\epsilon}_{\text{opt}}$ and $\bar{\epsilon}'_{\text{bias}}$. While Guidelines A and B indicate which clients to remove, the exact number of clients to remove at round $t$ is identified by minimizing $\epsilon^{(t)}$ as a proxy for the bounds in (2.10) and (2.20):

$$\epsilon^{(t)} := \underbrace{F_B(\boldsymbol{w}^{(t,0)}) - F_B^*}_{\epsilon_{\text{opt}}} + 4\bar{\kappa}^2 \cdot \underbrace{d_{TV}^2(\boldsymbol{\alpha}, \boldsymbol{p})\Gamma'}_{\bar{\epsilon}'_{\text{bias}}}, \tag{2.21}$$

where $\bar{\kappa}^2 \geq 0$ is a hyper-parameter that weights the relative importance of the optimization and bias error (see Sec. 2.4.3).

**Algorithm 3:** `CA-Fed` (Correlation-Aware FL)

**Input :** $\boldsymbol{w}^{(0,0)}$, $\boldsymbol{\alpha}$, $\boldsymbol{q}^{(0)}$, $\{\eta_c^{(t)}\}_{t=1}^T$, $\eta_s$, $K$, $\bar{\kappa}^2$, $\beta$, $\tau$

Initialize $\hat{\boldsymbol{F}}^{(0)}$, $\hat{\boldsymbol{F}}^*$, $\hat{\Gamma}'^{(0)}$, $\hat{\boldsymbol{\pi}}^{(0)}$, and $\hat{\boldsymbol{\lambda}}^{(0)}$;

**for** $t = 1, \ldots, T$ **do**

  Receive set of active client $\mathcal{S}^{(t)}$, loss vector $\boldsymbol{F}^{(t)}$;

  Update $\hat{\boldsymbol{F}}^{(t)}$, $\hat{\Gamma}'^{(t)}$, $\hat{\boldsymbol{\pi}}^{(t)}$, and $\hat{\boldsymbol{\lambda}}^{(t)}$;

  Initialize $\boldsymbol{q}^{(t)} = \frac{\boldsymbol{\alpha}}{\hat{\boldsymbol{\pi}}^{(t)}}$;

  $\boldsymbol{q}^{(t)} \leftarrow$ `ComputeWeights`$(\boldsymbol{q}^{(t)}, \boldsymbol{\alpha}, \hat{\boldsymbol{F}}^{(t)}, \hat{\boldsymbol{F}}^*, \hat{\Gamma}'^{(t)}, \hat{\boldsymbol{\pi}}^{(t)}, \hat{\boldsymbol{\lambda}}^{(t)})$;

  $\boldsymbol{q}^{(t)} \leftarrow$ `ComputeWeights`$(\boldsymbol{q}^{(t)}, \boldsymbol{\alpha}, \hat{\boldsymbol{F}}^{(t)}, \hat{\boldsymbol{F}}^*, \hat{\Gamma}'^{(t)}, \hat{\boldsymbol{\pi}}^{(t)}, -\hat{\boldsymbol{\pi}}^{(t)})$;

  **for** *client* $\{i \in \mathcal{S}^{(t)};\ q_i^{(t)} > 0\}$, *in parallel* **do**

    **for** $k = 0, \ldots, K-1$ **do**

      $\boldsymbol{w}_i^{(t,k+1)} = \boldsymbol{w}_i^{(t,k)} - \eta_c^{(t)} \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})$;

    $\Delta_i^{(t)} \leftarrow \boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t,0)}$;

  $\boldsymbol{w}^{(t+1,0)} \leftarrow \mathbf{Proj}_W(\boldsymbol{w}^{(t,0)} + \eta_s \sum_{i \in \mathcal{S}^{(t)}} q_i^{(t)} \cdot \Delta_i^{(t)})$;

**Procedure** `ComputeWeights`$(\boldsymbol{q}, \boldsymbol{\alpha}, \boldsymbol{F}, \boldsymbol{F}^*, \Gamma, \boldsymbol{\pi}, \boldsymbol{\rho})$**:**

  Sort $\mathcal{N}$ by descending order in $\boldsymbol{\rho}$;

  $\hat{\epsilon} \leftarrow \langle \boldsymbol{F} - \boldsymbol{F}^*, \boldsymbol{\pi} \tilde{\odot} \boldsymbol{q} \rangle + 4\bar{\kappa}^2 \cdot d_{TV}^2(\boldsymbol{\alpha}, \boldsymbol{\pi} \tilde{\odot} \boldsymbol{q})\Gamma$;

  **for** $i \in \mathcal{N}$ **do**

    $q_i^+ \leftarrow 0$;

    $\hat{\epsilon}^+ \leftarrow \langle \boldsymbol{F} - \boldsymbol{F}^*, \boldsymbol{\pi} \tilde{\odot} \boldsymbol{q}^+ \rangle + 4\bar{\kappa}^2 \cdot d_{TV}^2(\boldsymbol{\alpha}, \boldsymbol{\pi} \tilde{\odot} \boldsymbol{q}^+)\Gamma$;

    **if** $\hat{\epsilon} - \hat{\epsilon}^+ \geq \tau$ **then**

      $\hat{\epsilon} \leftarrow \hat{\epsilon}^+$;

      $\boldsymbol{q} \leftarrow \boldsymbol{q}^+$;

  **return** $\boldsymbol{q}$

### 2.4.1 CA-Fed's core steps

At each communication round $t$, the server sends the current model $\boldsymbol{w}^{(t,0)}$ to all active clients and each client $i$ sends back a noisy estimate $F_i^{(t)}$ of the current loss computed on a batch of samples $\mathcal{B}_i^{(t,0)}$, i.e., $F_i^{(t)} = \frac{1}{|\mathcal{B}_i^{(t,0)}|} \sum_{z \in \mathcal{B}_i^{(t,0)}} f(\boldsymbol{w}^{(t,0)}, z)$ (line 3). The server uses these values and the information about the current set of available clients $\mathcal{S}^{(t)}$ to refine its own estimates of each client's loss ($\hat{\boldsymbol{F}}^{(t)} = (\hat{F}_i^{(t)})_{i \in \mathcal{N}}$), and each client's loss minimum value ($\hat{\boldsymbol{F}}^{*} = (\hat{F}_i^{*})_{i \in \mathcal{N}}$), as well as of $\Gamma'$, $\pi_i$, $\lambda(\boldsymbol{P}_i)$, and $\epsilon^{(t)}$, denoted as $\hat{\Gamma}'^{(t)}$, $\hat{\pi}_i^{(t)}$, $\hat{\lambda}_i^{(t)}$, and $\hat{\epsilon}^{(t)}$, respectively (possible estimators are described below) (line 4).

The server decides whether excluding clients whose availability pattern exhibits high correlation (high $\hat{\lambda}_i^{(t)}$) (line 6). First, the server considers all clients in descending order of $\hat{\boldsymbol{\lambda}}^{(t)}$ (line 14), and evaluates if, by excluding them (line 17), $\hat{\epsilon}^{(t)}$ appears to be decreasing by more than a threshold $\tau \geq 0$ (line 19). Then, the server considers clients in ascending order of $\hat{\boldsymbol{\pi}}^{(t)}$, and repeats the same procedure to possibly exclude some of the clients with low availability (low $\hat{\pi}_i^{(t)}$) (lines 7).

Once the participating clients (those with $q_i > 0$) have been selected, the server notifies them to proceed updating the current models (lines 9–10) according to (2.2), while the other available clients stay idle. Finally, model's updates are aggregated according to (2.3) (line 12).

### 2.4.2 Estimators

We now briefly discuss possible implementation of the estimators $\hat{F}_i^{(t)}$, $\hat{F}_i^{*}$, $\hat{\Gamma}'^{(t)}$, $\hat{\pi}_i^{(t)}$, and $\hat{\lambda}_i^{(t)}$. Server's estimates for the clients' local losses ($\hat{\boldsymbol{F}}^{(t)} = (\hat{F}_i^{(t)})_{i \in \mathcal{N}}$) can be obtained from the received active clients' losses ($\boldsymbol{F}^{(t)} = (F_i^{(t)})_{i \in \mathcal{S}^{(t)}}$) through an auto-regressive filter with parameter $\beta \in (0, 1]$:

$$\hat{\boldsymbol{F}}^{(t)} = (\mathbf{1} - \beta \mathbb{1}_{\mathcal{S}^{(t)}}) \odot \hat{\boldsymbol{F}}^{(t-1)} + \beta \mathbb{1}_{\mathcal{S}^{(t)}} \odot \boldsymbol{F}^{(t)}, \tag{2.22}$$

where $\odot$ denotes the component-wise multiplication between vectors, and $\mathbb{1}_{\mathcal{S}^{(t)}}$ is a $N$-dimensions binary vector whose $i$-th component equals 1 if and only if client $i$ is active at round $t$, i.e., $i \in \mathcal{S}^{(t)}$. The server can estimate client-$i$'s loss minimum value $F_i^{*}$ as $\hat{F}_i^{*} = \min_{s \in [0,t]} \hat{F}_i^{(s)}$. The values of $F_B(\boldsymbol{w}^{(t,0)})$, $F_B^{*}$, $\Gamma'$, and $\epsilon^{(t)}$ can be estimated as follows:

$$\hat{F}_B^{(t)} - \hat{F}_B^{*} = \langle \hat{\boldsymbol{F}}^{(t)} - \hat{\boldsymbol{F}}^{*}, \hat{\boldsymbol{\pi}}^{(t)} \tilde{\odot} \boldsymbol{q}^{(t)} \rangle, \tag{2.23}$$

$$\hat{\Gamma}'^{(t)} = \max_{i \in \mathcal{N}} (\hat{F}_i^{(t)} - \hat{F}_i^{*}), \tag{2.24}$$

$$\hat{\epsilon}^{(t)} = \hat{F}_B^{(t)} - \hat{F}_B^{*} + 4\bar{\kappa}^2 \cdot d_{TV}^2(\boldsymbol{\alpha}, \hat{\boldsymbol{\pi}}^{(t)} \tilde{\odot} \boldsymbol{q}^{(t)}) \hat{\Gamma}'^{(t)}. \tag{2.25}$$

where $\boldsymbol{\pi} \tilde{\odot} \boldsymbol{q} \in \mathbb{R}^N$, such that $(\boldsymbol{\pi} \tilde{\odot} \boldsymbol{q})_i \coloneqq \frac{\pi_i q_i}{\sum_{h=1}^{N} \pi_h q_h}$, $i \in \mathcal{N}$.

For $\hat{\pi}_i^{(t)}$, the server can simply keep track of the total number of times client $i$ was available up to time $t$ and compute $\hat{\pi}_i^{(t)}$ using a Bayesian estimator with beta prior, i.e., $\hat{\pi}_i^{(t)} = (\sum_{s \leq t} \mathbb{1}_{i \in \mathcal{S}^{(s)}} + n_i)/(t + n_i + m_i)$, where $n_i$ and $m_i$ are the initial parameters of the beta prior.

For $\hat{\lambda}_i^{(t)}$, the server can assume the client availability evolves according to a Markov chain with two states (active and inactive), track the corresponding number of state transitions, and estimate the

transition matrix $\hat{\boldsymbol{P}}_i^{(t)}$ through a Bayesian estimator similarly to what done for $\hat{\pi}_i^{(t)}$. Finally, $\hat{\lambda}_i^{(t)}$ is obtained computing the eigenvalues of $\hat{\boldsymbol{P}}_i^{(t)}$.

### 2.4.3 The role of the hyper-parameter $\bar{\kappa}^2$

Theorems 2.3.2 and 2.3.4 suggest that the condition number $\kappa^2$ has a significant impact on the minimization of the total error $\epsilon$. Our algorithm uses a proxy ($\epsilon^{(t)}$) for the total error (see (2.21)). To account for the effect of $\kappa^2$, we introduced the hyper-parameter $\bar{\kappa}^2 \geq 0$, which weights the relative importance of the optimization and bias error in (2.21). In practice, $\bar{\kappa}^2$ controls the number of excluded clients by `CA-Fed`. A small value of $\bar{\kappa}^2$ penalizes the bias term in favor of the optimization error, resulting in a larger number of excluded clients. Conversely, the bias term dominates for large values of $\bar{\kappa}^2$, and `CA-Fed` tends to include more clients. Asymptotically, for $\bar{\kappa}^2 \to \infty$, `CA-Fed` reduces to the *unbiased aggregation strategy*.

## 2.5 Experimental Evaluation

### 2.5.1 Experimental Setup

**Federated system simulator** In our experiments, we consider a population of $N = 100$ clients. We model the activity of each client $i \in \mathcal{N}$ as a two-state homogeneous Markov process with state space $\mathcal{S} = \{$"*active*", "*inactive*"$\}$, characterized by a transition matrix $\boldsymbol{P}_i$, a stationary distribution $\boldsymbol{\pi}^{(i)}$, and a second largest absolute eigenvalue $\bar{\lambda}_2(\boldsymbol{P}_i)$ (see Appendix F.C for details). Our goal is to simulate realistic dynamics of federated systems featuring varying levels of clients' availability and correlation. To introduce heterogeneity in clients' availability patterns, we divide the population in two equally-sized classes: the "more available" clients with a steady-state probability of being active $\pi_{i,\text{active}} = 1/2 + g$, and the "less available" clients with $\pi_{i,\text{active}} = 1/2 - g$. Here, the parameter $g \in (0, 1/2)$ controls the degree of heterogeneity in clients' availability. We furthermore divide each class of clients in two equally-sized sub-classes: clients exhibiting a largely correlated time behavior (in the following referred to as "correlated" clients) that tend to persist in the same state for rather long periods ($\lambda_i = \nu$ with values of $\nu$ close to 1), and clients exhibiting a weakly correlated time behavior (referred to as "weakly correlated" clients) that are almost as likely to keep as to change their state at every $t$ ($\lambda_i \sim \mathcal{N}(0, \varepsilon^2)$, with $\varepsilon$ close to 0). We use $g = 0.4$, $\nu = 0.9$, and $\varepsilon = 10^{-2}$.

**Datasets and models** We conduct experiments on the LEAF Synthetic dataset (Caldas et al., 2019), a benchmark for multinomial classification tasks, and on the real-world MNIST (L. Deng, 2012) and CIFAR-10 (Krizhevsky & Hinton, 2009) datasets, respectively for handwritten digits and image recognition tasks. To simulate the statistical heterogeneity present in the federated learning system, we use common approaches in the literature. For the Synthetic dataset, we tune the parameters $(\gamma, \delta)$, which control data heterogeneity among clients (X. Li et al., 2020). For MNIST and CIFAR-10, we distribute samples from the same class across the clients according to a symmetric Dirichlet distribution with parameter $\varsigma$, following the same approach as (H. Wang, Yurochkin, Sun, Papailiopoulos, & Khazaeni, 2020). Unless otherwise indicated, we set $\gamma = \delta = \varsigma = 0.5$. We use the original training/test data split of MNIST and reserve 20% of the training dataset as the validation dataset. For Synthetic and MNIST, we use a linear classifier with a ridge

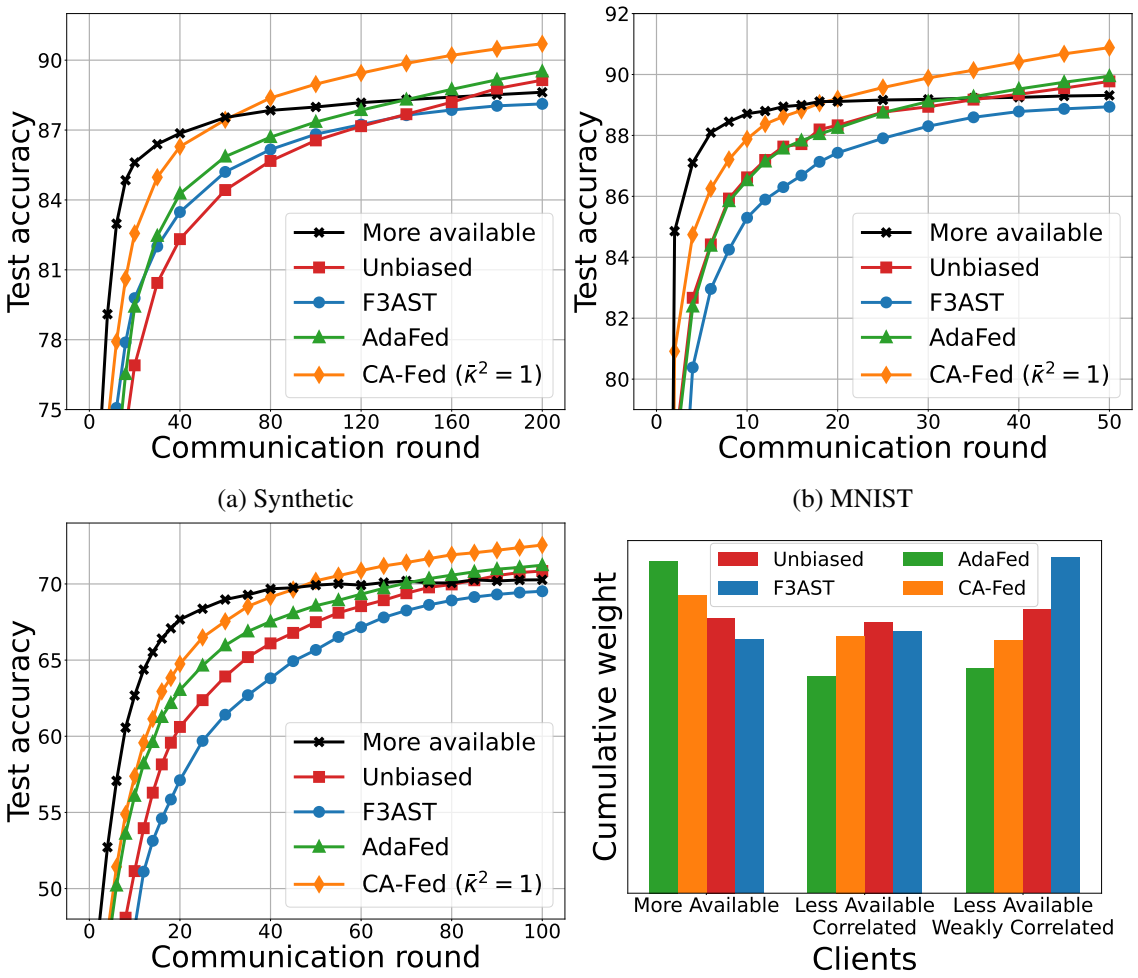
penalization of parameter $10^{-2}$, which corresponds to a strongly convex objective function. For CIFAR-10, we use a neural network with two convolutional and one fully connected layers.

**Benchmarks** We compare `CA-Fed`, defined in Algorithm 3, with four baselines including two state-of-the-art FL algorithms discussed in Section 2.2: 1) `Unbiased`, which aggregates the active clients $i \in \mathcal{S}^{(t)}$ with weights $q_i = \alpha_i/\pi_i$; 2) `More available`, which considers only the "more available" clients and always excludes the "less available" ones; 3) `AdaFed` (Tan et al., 2022), which, similarly to `Unbiased`, aggregates all active clients, but normalizes their aggregation weights (i.e., it considers $q_i = \frac{\alpha_i/\pi_i}{\sum_{i\in\mathcal{S}^{(t)}} \alpha_i/\pi_i}$); 4) `F3AST` (Ribero et al., 2023), which, oppositely to `More available`, favors the "less available" clients. For all algorithms, we tuned the learning rates $\eta_c$, $lr[s]$ via grid search. For `CA-Fed`, we use $\beta = \tau = 0$. Unless otherwise specified, we assume that the algorithms can access an oracle providing the true availability parameters for each client: in practice, all the algorithms rely on the exact knowledge of $\pi_{i,\text{active}}$; in addition, `CA-Fed` also receives $\lambda(\boldsymbol{P}_i)$. In Section 2.5.2, we will relax this assumption by considering the estimators $\hat{\pi}_i^{(t)}$ and $\hat{\lambda}_i^{(t)}$. The code for our experimental framework is available at: https://github.com/arodio/CA-Fed.

### 2.5.2 Experimental Results

**`CA-Fed` vs. baselines** Figure 2.1 compares the test accuracy achieved by `CA-Fed` ($\bar{\kappa}^2 = 1$) and the baselines on the Synthetic (Fig. 2.1a), MNIST (Fig. 2.1b), and CIFAR-10 (Fig. 2.1c) datasets over 10 different runs. Across all three datasets, `CA-Fed` consistently outperforms the baselines, achieving higher test accuracy (+1.56 pp on Synthetic; +0.94 pp on MNIST; +1.32 pp on CIFAR-10) compared to the second best performing method, `AdaFed`. These results demonstrate that `CA-Fed` achieves the best balance between convergence speed and test accuracy. For deeper insights into the algorithms' behavior, Figure 2.1d illustrates the cumulative aggregation weights $\{\frac{1}{T}\sum_{t=1}^{T} q_i^{(t)}\}_{i\in\mathcal{N}}$, representing the cumulative importance that the algorithms assigned to the clients at the end of the training. In Figure 2.1d, we grouped the clients into three categories: "more available", "less available, weakly correlated", and "less available, correlated". By setting the aggregation weights inversely proportional to the clients' availabilities, `Unbiased` equalizes the importance for all clients (see Fig. 2.1d), but achieves a slower convergence (as shown in Figs. 2.1a, 2.1b, and 2.1c). On the contrary, by excluding all the "less available" clients, `More available` achieves a faster convergence but introduces a non-vanishing bias error $\epsilon_{\text{bias}}$, which, in practice, leads to poor accuracy performance. The state-of-the-art algorithm `AdaFed`, similarly to `Unbiased`, considers all the active clients, but normalizes their aggregation weights at each communication round. As a result, similarly to `CA-Fed`, `AdaFed` indeed prioritizes the "more available" clients (as shown in Fig. 2.1d), and then a convergence speed-up could be expected. However, `AdaFed` does not exclude the "less available and correlated" clients, and therefore their presence causes a convergence slowdown. Finally, `F3AST` favors the "less available, correlated" clients and achieves a slower convergence with a non-vanishing bias error, which corresponds to lower accuracy performance. By opportunely excluding some of the "less available and correlated" clients, `CA-Fed` achieves the best test accuracy by the end of the training time.

**Convergence speed vs. Bias error** The trade-off between $\epsilon_{\text{opt}}$ or $\epsilon_{\text{bias}}$ discussed in Section 2.3 is visible in our experiments. In particular, Figure 2.2a compares the test accuracy achieved by




(a) Synthetic

(b) MNIST

(c) CIFAR-10

(d) Cumulative importance

Figure 2.1: Average test accuracy among $N = 100$ clients achieved by the algorithms on the Synthetic, MNIST, and CIFAR-10 datasets. Cumulative importance assigned by the algorithms to the clients after $T = 200$ rounds on the Synthetic dataset.

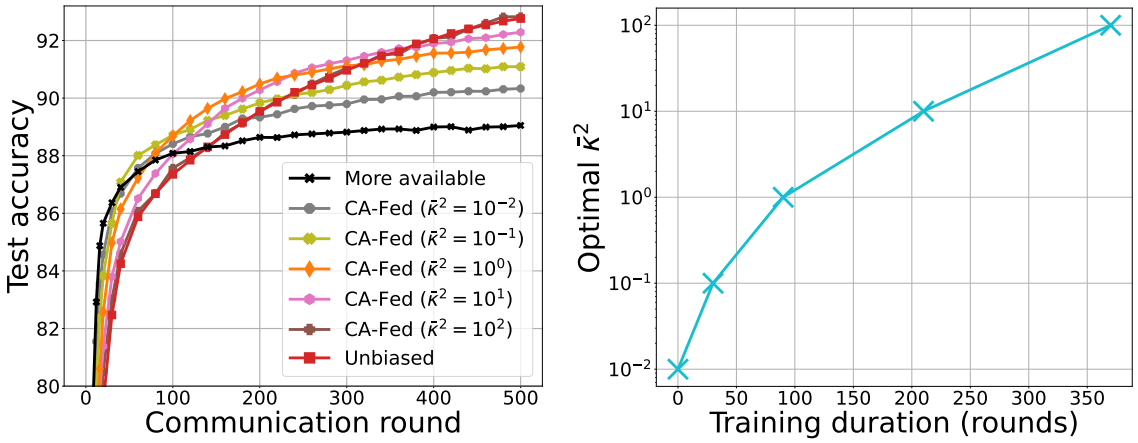
`More available`, `Unbiased`, and `CA-Fed` on the Synthetic dataset for $T = 500$ communication rounds. As expected, by targeting the minimization of $\epsilon_{\text{opt}}$ and thus excluding the "less available" clients, `More available` achieves the fastest convergence at the expense of a large non-vanishing bias error $\epsilon_{\text{bias}}$. On the other hand, by targeting the minimization of $\epsilon_{\text{bias}}$ and thus equalizing the clients' importance, `Unbiased` asymptotically removes this error and ultimately achieves the highest test accuracy at communication round $T = 500$, but suffers from slower convergence due to the presence of the "correlated" clients. Our algorithm, `CA-Fed`, leverages the trade-off between convergence speed and model bias and achieves fast convergence to the neighborhood of the target objective. To explore this trade-off, in Figure 2.2a, we varied the value of the hyper-parameter $\bar{\kappa}^2$ in the range $\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$. `CA-Fed` tends to exclude more clients for low values of $\bar{\kappa}^2$ and achieves a similar convergence rate as `More available` for $\bar{\kappa}^2 = 10^{-2}$. For intermediate values of $\bar{\kappa}^2$, `CA-Fed` trades a small accuracy decrease for faster

convergence (refer, for example, to the curves $\bar{\kappa}^2 = 10^0, 10^1$). For $\bar{\kappa}^2 = 10^2$, `CA-Fed` reduces to `Unbiased` (their curves overlap in Fig. 2.2a). Moreover, we observe that the optimal value of $\bar{\kappa}^2$ depends on the available time for training. Low values of $\bar{\kappa}^2$ speed-up convergence and then they can be beneficial for short training durations (e.g., `CA-Fed` ($\bar{\kappa} = 10^{-1}$) achieves a higher test accuracy of +2.8 pp with respect to `Unbiased` at communication round $t = 40$). For longer training periods, a larger value of $\bar{\kappa}^2$ may be preferable as it reduces the bias error and increases the test accuracy (e.g., `CA-Fed` ($\bar{\kappa} = 10^2$) improves of +3.8 pp with respect to `More available` at communication round $t = 500$). Figure 2.2b illustrates the optimal value of $\bar{\kappa}^2$ for different durations of the training period $T$.

**Effect of statistical heterogeneity** The bias error bounds $\bar{\epsilon}_{\text{bias}}$ and $\bar{\epsilon}'_{\text{bias}}$ in Theorems 2.3.2 and 2.3.4 are influenced by the degree of heterogeneity among local functions, commonly known as *statistical heterogeneity*, characterized by the constants $\Gamma$ and $\Gamma'$ in (2.11) and (2.20), respectively. To control statistical heterogeneity, we manipulate the dissimilarity among the clients' local datasets, specifically through the parameters $\gamma$ and $\delta$ in the case of the Synthetic dataset, as explained in Section 2.5.1. Figure 2.3 illustrates the impact of $\gamma$ and $\delta$ on the test accuracy achieved by `CA-Fed` after $T = 200$ communication rounds on the Synthetic dataset. As expected, in the extreme IID setting (when $\gamma = \delta = 0$), $\Gamma$ and $\Gamma'$ are small, and the bias error $\epsilon_{\text{bias}}$ is negligible. As a result, `More available` and `CA-Fed` ($\bar{\kappa}^2 = 10^{-2}$) reach the highest test accuracy, whereas `CA-Fed` ($\bar{\kappa}^2 = 10^2$) and `Unbiased` present slow convergence. Nevertheless, `More available` and `CA-Fed` ($\bar{\kappa}^2 = 10^{-2}$) perform poorly as the statistical heterogeneity increases (i.e., $\gamma = \delta \geq 0.25$). In the extreme non-IID setting (when $\gamma = \delta = 1$), $\Gamma$ and $\Gamma'$ are large, and $\epsilon_{\text{bias}}$ dominates. In this case, `CA-Fed` ($\bar{\kappa}^2 = 10^2$) and `Unbiased` should be preferred. For $\gamma = \delta = \{0.25, 0.5, 0.75\}$, `CA-Fed` (with $\bar{\kappa}^2 = 1$ or $\bar{\kappa}^2 = 10$) achieves the highest test accuracy (+1.6 pp, +1.2 pp, and +1.0 pp with respect to `Unbiased`).

**Estimation of the clients' availability and correlation** In this experiment, `CA-Fed` utilizes estimators $\hat{\pi}_i^{(t)}$ and $\hat{\lambda}_i^{(t)}$ to estimate the clients' $\pi_i$ and $\lambda_i$ values. We employ a Bayesian estimator with a beta prior to estimate $\hat{\boldsymbol{P}}_i^{(t)}$, which we generate by observing the evolution of the Markov chain defined by $\boldsymbol{P}_i$ over $t'$ time-steps. We compute $\hat{\pi}_i^{(t)}$ and $\hat{\lambda}_i^{(t)}$ analytically, following the methodology explained in Section 2.4.2 and described in detail in Appendix F.C. Figure 2.4a shows the estimation errors $\frac{1}{N}\sum_{i\in\mathcal{N}}|\hat{\pi}_i^{(t)} - \pi_i|$ and $\frac{1}{N}\sum_{i\in\mathcal{N}}|\hat{\lambda}_i^{(t)} - \lambda_i|$ as a function of the number of historical observations $t'$. As expected, both errors decrease with an increasing number of observations, and the estimation error for $\lambda_i$ is larger than that for $\pi_i$. Furthermore, Figure 2.4b compares the final test accuracy obtained by `CA-Fed` and the baselines for varying numbers of historical observations $t' \in \{10^1, 10^{1.5}, 10^2, 10^{2.5}, 10^3, 10^{3.5}, 10^4\}$ when training for $T = 50$ rounds on the MNIST dataset. In this setting, `CA-Fed` outperforms the baselines for $t' \geq 100$. This value is reasonable, because estimating $\lambda_i$ requires a number of observations comparable to the expected hitting time for the slowest Markov chain, which is given by $\max_{i\in\mathcal{N}} \frac{1}{(1-\lambda_i)\pi_i} = 100$.

**`CA-Fed` with Spatial Correlation** Although `CA-Fed` is primarily designed to handle temporal correlation (as discussed in Section 2.3.3.1), we also evaluate its performance in the presence of spatial correlation. In the considered spatially correlated scenario, clients are grouped into clusters, and each cluster $c \in \mathcal{C}$ is characterized by an underlying Markov chain that determines when




(a) Test accuracy (b) Optimal $\bar{\kappa}^2$

Figure 2.2: *Convergence speed vs. Model bias* trade-off for different values of $\bar{\kappa}^2$ on the Synthetic dataset, for $\gamma = \delta = 0.5$.

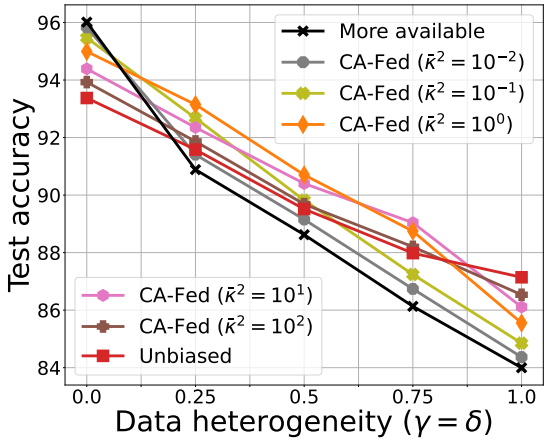
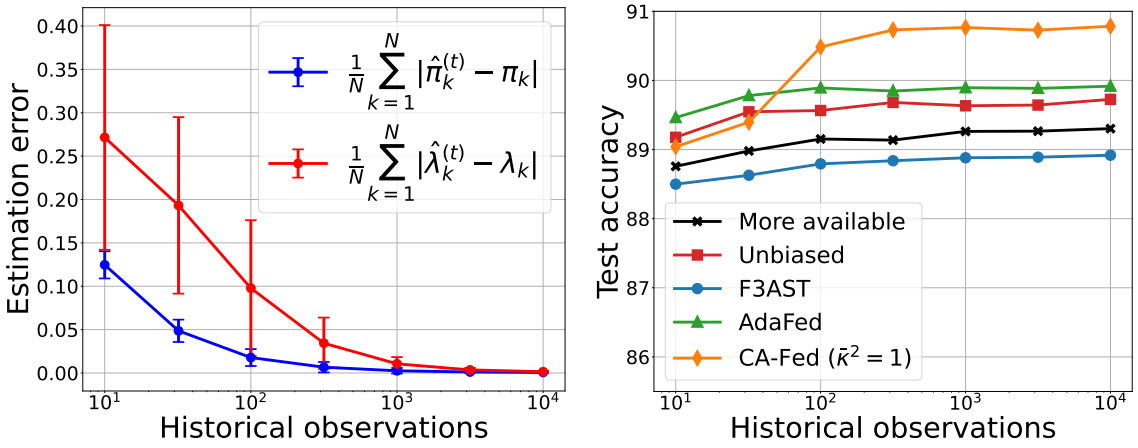
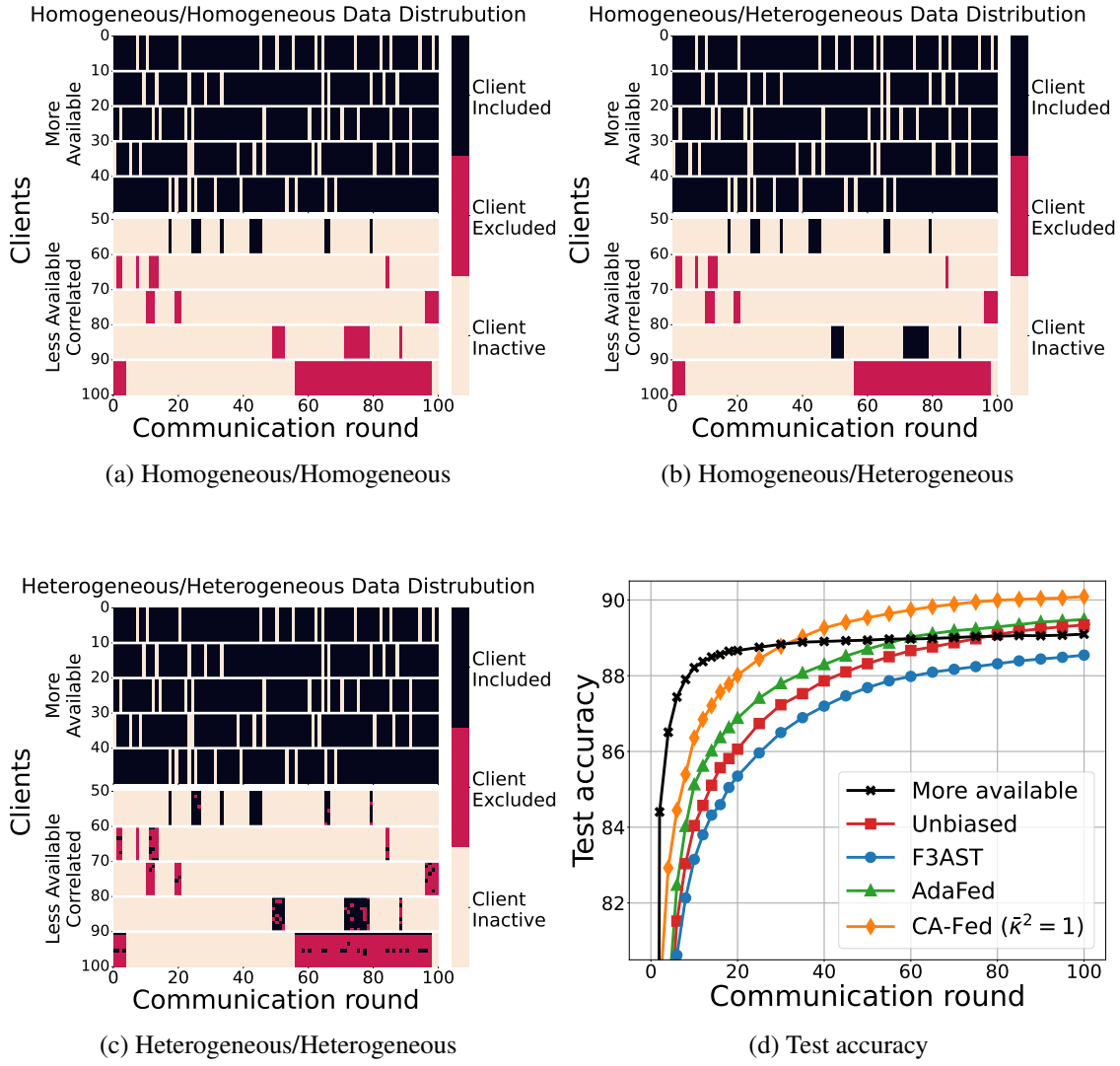



Figure 2.3: Effects of *data heterogeneity* on the Synthetic dataset after $T = 200$ rounds.




(a) Estimation error (b) Test accuracy

Figure 2.4: Estimation of the *clients' activities* $(\hat{\pi}_i^{(t)}, \hat{\lambda}_i^{(t)})$ for different priors $t \in \{10^1, 10^{1.5}, 10^2, 10^{2.5}, 10^3, 10^{3.5}, 10^4\}$ and test accuracy after $T = 50$ rounds on the MNIST dataset.




(a) Homogeneous/Homogeneous

(b) Homogeneous/Heterogeneous

(c) Heterogeneous/Heterogeneous

(d) Test accuracy

Figure 2.5: Clients' activities and `CA-Fed`'s inclusion/exclusion decisions in the presence of *spatial correlation* for different degrees of *intra-cluster/inter-cluster* data distributions. Average test accuracy after $T = 100$ rounds on the MNIST dataset.

all clients in the cluster are available or unavailable. The Markov chains of different clusters are independent. Let $\lambda_c$ denote the second-largest eigenvalue in magnitude of cluster $c$'s Markov chain. To reduce the eigenvalue of the aggregate Markov chain, `CA-Fed` needs to exclude all clients in the cluster $\bar{c} = \arg\max_{c \in \mathcal{C}} \lambda_c$. In this experiment, we consider a population of $N = 100$ clients grouped into $|\mathcal{C}| = 10$ clusters. We equally split the clients, or equivalently, the clusters, into two categories: "more available" with $\pi_c = 0.9$ and $\lambda_c = 0$ for $c = 0, \ldots, 4$, and "less available, correlated" with $\pi_c = 0.1$ and $\lambda_c = c/10$ for $c = 5, \ldots, 9$. In Figures 2.5a, 2.5b, and 2.5c, each pixel represents, for each client $i \in \mathcal{N}$ and for each communication round, the client's activity (active/inactive) and `CA-Fed`'s decision (included/excluded in training). From the experiments, we observe that `CA-Fed`'s decisions depend on the degree of statistical heterogeneity among clients within a cluster (i.e., *intra-cluster*) and among clusters (i.e., *inter-cluster*). When both the intra-cluster and inter-cluster clients' data distributions are homogeneous, `CA-Fed` starts considering the clients in cluster $\bar{c} = 9$ with $\lambda_{\bar{c}} = 0.9$, and sequentially excludes, in order, all clients from clusters $\{9, 8, 7, 6\}$ (as shown in Fig. 2.5a). When the clients' data distributions are homogeneous within clusters, but heterogeneous among clusters (Fig. 2.5b), `CA-Fed` still excludes all clients from clusters $c = \{9, 7, 6\}$, but decides to include clients from cluster $c = 8$. This is because these clients happen to have a lower value of $\hat{F}_i^{(t)} - \hat{F}_i^*$, and despite having a large $\lambda_c$, `CA-Fed` decides to include them. Finally, when both the intra-cluster and inter-cluster clients' data distributions are heterogeneous (Fig. 2.5c), `CA-Fed` can partially include clients from the more correlated clusters, even though their $\lambda_c$ is large. Figure 2.5d compares the test accuracy achieved by `CA-Fed` and the baselines with spatial correlation in the same setting as in Figure 2.5c. The experimental results show that `CA-Fed` can operate correctly in the presence of spatial correlation and still outperforms the baselines (+0.6 pp w.r.t. `AdaFed`).

## 2.6 Conclusion

This chapter presents the first convergence analysis of a `FedAvg`-like federated learning algorithm in presence of heterogeneous and correlated client availability. The analysis reveals the detrimental effect of correlation on the convergence rate and highlights a fundamental trade-off between convergence speed and model bias. To navigate this tradeoff, we introduce `CA-Fed`, a novel FL algorithm, which adaptively manages the conflicting aims of enhancing convergence speed and reducing model bias, with the ultimate objective of maximizing model quality within the constraints of the training time available. `CA-Fed` achieves this goal by dynamically excluding clients who exhibit high temporal correlation and limited availability, contingent on their data distributions. Indeed, model updates from such clients may act as noise, increasing variance and slowing down the algorithm's convergence. `CA-Fed` disregards such clients unless their local datasets notably enhance the quality of the final model. The experimental results validate the effectiveness of our strategy, demonstrating that `CA-Fed` is a versatile and resilient FL algorithm, well-suited to address real-world scenarios characterized by heterogeneous and correlated client availability. Further discussions on the computation and communication costs, and fairness of `CA-Fed` can be found in Appendix A.

# CHAPTER 3

# Variance Reduction: leveraging Stale Updates for Non-Participating Clients

This chapter is based on our works Rodio (2024), accepted for presentation in the ACM Student Research Competition (ACM SRC 2024) and forthcoming in the proceedings of ACM Sigmetrics IFIP Performance 2024, and Rodio and Neglia (2024), currently under peer review and available as a preprint at https://arxiv.org/abs/2405.04171.

## 3.1 Motivation

In Chapters 1 and 2, we extensively discussed the bias-variance trade-off that emerges from heterogeneous client participation. We observed how such heterogeneity risks biasing the global model in favor of the "more participating" clients (Chapter 2, Eq. (2.4)). To eliminate this bias, we reviewed previous literature (S. Wang & Ji, 2022, 2024) and contributed (Theorems 2.3.2 and 2.3.4) to propose an unbiased version of `FedAvg`, where client updates are scaled inversely to their participation frequency. However, such strategies, while removing bias, exacerbate the variability of the learning process. Specifically, the unbiased scaling process amplifies variations in the magnitude of client updates, leading to increased variance in the model and slower convergence.

Given these challenges, Chapter 2 focused on the impact of correlation in client participation—a factor previously unexplored—and introduced `CA-Fed` (Algorithm 7), which adaptively handles participation of the "less available" and "highly correlated" clients to better balance the bias-variance trade-off. In this chapter, we set aside the problem of correlation to explore other strategies that navigate the complexities of this trade-off.

To this purpose, global variance reduction methods (Gu et al., 2021; H. Yang et al., 2022; Jhunjhunwala et al., 2022; Yan et al., 2024) propose to leverage the most recent, albeit potentially stale, model updates in place of unavailable updates from non-participating clients. Among these algorithms, `FedVARP` (Federated VAriance Reduction for Partial client participation) (Jhunjhunwala et al., 2022)`FedVARP` has demonstrated, both theoretically and empirically, its capability to effectively lower variance and consistently outperform `FedAvg` in settings with partial yet homo-

geneous client participation. It is anticipated to perform similarly well even in heterogeneous settings (Jhunjhunwala et al., 2022). However, when client participation varies widely, global variance reduction methods, including `FedVARP`, must address the challenge of updates of varying staleness—a complex issue that remains unexplored and is the focus of this chapter.

This chapter specifically addresses the following questions:

1. *Is it really true that `FedVARP` outperforms the unbiased `FedAvg` under heterogeneous client participation?*
2. *Assuming that each method may be preferable in different settings, can we design an unbiased algorithm that combines fresh and stale updates and adapts to specific levels of participation heterogeneity?*

Addressing these questions is challenging and requires a deeper understanding of how stale client updates influence convergence.

**Our contributions.** We thoroughly analyze this problem and make the following novel contributions:

1. We analytically and experimentally refute the belief that `FedVARP` consistently outperforms `FedAvg`. Our convergence analysis reveals that leveraging stale updates can be either beneficial or detrimental, depending on the specific level of client data and participation heterogeneity.
2. We propose `FedStale` (Federated Averaging with Stale Updates), a novel FL algorithm that updates the global model through a convex, unbiased combination of fresh and stale updates, parameterized by a weight $\beta$. `FedStale` spans the spectrum from `FedAvg` ($\beta = 0$, exclusively fresh updates) to `FedVARP` ($\beta = 1$, equal weighting of fresh and stale updates). Our analysis provides guidelines to tune the parameter $\beta$ to match specific data and client participation heterogeneity scenarios.
3. We evaluate `FedAvg`, `FedVARP`, and `FedStale` across multiple levels of client data and participation heterogeneity. `FedStale` outperforms both `FedAvg` and `FedVARP` across the vast majority of heterogeneity levels examined.

The remainder of this chapter is organized as follows. Section 3.2 reviews the problem and related work. Section 3.3 introduces `FedStale`, our staleness-aware algorithm, through a motivating example. Section 3.4 provides a convergence analysis of `FedStale` under heterogeneous client participation. `FedStale` is extensively evaluated in Section 3.5, and Section 3.6 concludes the chapter. Detailed proofs are available in Appendix B.

## 3.2 Problem Description and Background

We consider here the same problem described in Section 1.2.1, focusing, for sake of concreteness, on the "per-client fairness" criterion, i.e., $\alpha_i = \frac{1}{N}, \forall i$:*

$$\min_{\boldsymbol{w}\in\mathbb{R}^d} F(\boldsymbol{w}) \triangleq \frac{1}{N}\sum_{i=1}^{N}\left[F_i(\boldsymbol{w}) \triangleq \frac{1}{n_i}\sum_{z_i\in D_i} f(\boldsymbol{w}, z_i)\right]. \tag{3.1}$$

In this chapter, we consider algorithms obeying the general operation in Algorithm 2, but differing in the `Aggregate()` procedure. We recall that, at round $t$, $\mathcal{S}^{(t)}$ denotes the random subset of participating clients, usually $|\mathcal{S}^{(t)}| \ll N$. Each client in $\mathcal{S}^{(t)}$ runs multiple ($K \geq 1$) iterations of local stochastic gradient descent (SGD) on its local dataset:

$$\boldsymbol{w}_i^{(t,k+1)} = \boldsymbol{w}_i^{(t,k)} - \eta_c \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) \quad \text{for } k = 0, \ldots, K-1$$

producing the local model $\boldsymbol{w}_i^{(t,K)}$, and the sends the model update $\Delta_i^{(t)} = (\boldsymbol{w}^{(t)} - \boldsymbol{w}_i^{(t,K)})$ to the server. The server aggregates these client updates into the *global* update:

$$\Delta_{\texttt{FedAvg}}^{(t)} = \frac{1}{|\mathcal{S}^{(t)}|}\sum_{i\in\mathcal{S}^{(t)}} \Delta_i^{(t)}, \tag{3.2}$$

and then applies this update to the previous global model in a manner similar to a gradient descent step to produce the new global model $\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} - \eta_s \Delta_{\texttt{FedAvg}}^{(t)}$.

Following standard assumptions (S. Wang & Ji, 2022; Rodio, Faticanti, et al., 2023b; S. Wang & Ji, 2024), in line with Section 1.3.2.1, we model client participation heterogeneity through the *participation probability* $p_i$:

$$p_i \triangleq \mathbb{E}_{\mathcal{S}^{(t)}}\left[\mathbb{P}(i \in \mathcal{S}^{(t)})\right]. \tag{3.3}$$

When client participation is *homogeneous* ($p_i = p, \forall i$), $\mathbb{E}_{\mathcal{S}^{(t)}}[\Delta_{\texttt{FedAvg}}^{(t)}] = \frac{1}{N}\sum_{i=1}^{N}\Delta_i^{(t)}$. Under this condition, Eq. (3.2) is then an unbiased estimator of the model update as if *all* clients were to participate (X. Li et al., 2020; Fraboni et al., 2021). This ensures that the final model fairly represents all clients.

Conversely, under *heterogeneous* participation, where probabilities $\{p_i\}$ vary among clients, Eq. (3.2) becomes a biased estimator of $\frac{1}{N}\sum_{i=1}^{N}\Delta_i^{(t)}$. This *bias* in the global update tends to overrepresent clients that participate more frequently, disadvantaging those that participate less. Participation heterogeneity can then lead to objective inconsistency, causing `FedAvg` to effectively minimize the *biased* objective:

$$F_B(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N}\frac{p_i}{\sum_{j=1}^{N} p_j} F_i(\boldsymbol{w}), \tag{3.4}$$

*The analysis in this chapter can be immediately extended to any weighted sum of local objectives (i.e, Problem (1.2)).

which may arbitrarily deviate from the global objective (3.1).

To effectively minimize objective (3.1) when client participation is heterogeneous, recent works (Fraboni et al., 2021; S. Wang & Ji, 2022; Fraboni, Vidal, Kameni, & Lorenzi, 2023; S. Wang & Ji, 2024), including our Rodio, Faticanti, et al. (2023b) (Chapter 2, Theorems 2.3.2 and 2.3.4), have discussed the need to debias $\Delta^{(t)}_{\texttt{FedAvg}}$. Specifically, Eq. (3.2) has been modified into Eq. (3.5), resulting in an unbiased version of `FedAvg`, denoted here as `U-FedAvg` (S. Wang & Ji, 2022; Rodio, Faticanti, et al., 2023b; S. Wang & Ji, 2024):

$$\Delta^{(t)}_{\texttt{U-FedAvg}} = \frac{1}{N} \sum_{i \in \mathcal{S}^{(t)}} \frac{\Delta_i^{(t)}}{p_i}. \tag{3.5}$$

Intuitively, reweighting each client update by $p_i^{-1}$ compensates for less participating clients by amplifying their update when they do participate. `U-FedAvg` naturally extends `FedAvg` to accommodate heterogeneous client participation—reducing to `FedAvg` when participation is uniform ($p_i = \frac{|\mathcal{S}^{(t)}|}{N}, \forall i$)—and effectively *unbiases* the global update ($\mathbb{E}_{\mathcal{S}^{(t)}}[\Delta^{(t)}_{\texttt{U-FedAvg}}] = \frac{1}{N}\sum_{i=1}^{N} \Delta_i^{(t)}$). However, it also introduces a drawback: the variance of each client updates is now proportional to $p_i^{-2}$. As participation probabilities decrease, this variance rapidly increases, becoming the dominant factor that slows down `U-FedAvg`'s convergence (Rodio, Faticanti, et al., 2023b; S. Wang & Ji, 2024).

A few recent works have addressed the variance introduced by partial client participation through global variance reduction, leveraging stale updates to compensate for non-participating clients (Gu et al., 2021; H. Yang et al., 2022; Jhunjhunwala et al., 2022; Yan et al., 2024). These methods were originally proposed for *homogeneous* participation and, if applied in their original form, would introduce a bias when client participation becomes heterogeneous. Fortunately, unbiasing them to work in *heterogeneous* participation scenarios is straightforward, similar to what was done for `FedAvg` in Eq. (3.5). We select `FedVARP` (Jhunjhunwala et al., 2022) as the representative algorithm and adapt it into `U-FedVARP` (Unbiased `FedVARP`).

In `U-FedVARP`, the server retains the most recent, though potentially stale, update for each client:

$$\boldsymbol{h}_i^{(t)} = \begin{cases} \Delta_i^{(t-1)} & \text{if } i \in \mathcal{S}^{(t-1)} \\ \boldsymbol{h}_i^{(t-1)} & \text{otherwise} \end{cases}, \tag{3.6}$$

and then uses these stale updates as proxies for missing contributions from non-participating clients in the current round:

$$\Delta^{(t)}_{\texttt{U-FedVARP}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{h}_i^{(t)} + \frac{1}{N} \sum_{i \in \mathcal{S}^{(t)}} \frac{\Delta_i^{(t)} - \boldsymbol{h}_i^{(t)}}{p_i}. \tag{3.7}$$

Unlike `U-FedAvg`, which essentially ignores non-participating clients, `U-FedVARP` utilizes their last updates, albeit stale, when they do not participate in the training process. When they participate again, `U-FedVARP` subtracts these stale updates to eliminate any inconsistency caused by using stale information, and applies the fresh update. Both corrections are reweighed by $p_i^{-1}$, similarly to `U-FedAvg`, ensuring that $\mathbb{E}\left[\Delta^{(t)}_{\texttt{U-FedVARP}}\right] = \frac{1}{N}\sum_{i=1}^{N} \Delta_i^{(t)}$. `U-FedVARP`'s aggregation (3.7) is then *unbiased*. Moreover, by leveraging stale updates for non-participating clients,

U-FedVARP acts as a SAGA-like (Defazio, Bach, & Lacoste-Julien, 2014) variance reduction method, aiming to reduce the variance caused by partial client participation. This strategy incurs an additional memory cost of $N \times d$, which must be allocated by the server.

Although variance reduction methods like FedVARP are often believed to outperform simpler algorithms like FedAvg under partial and heterogeneous client participation, as suggested for example in (Jhunjhunwala et al., 2022; S. Wang & Ji, 2024), theoretical support for this belief has been provided only for homogeneous participation scenarios (Jhunjhunwala et al., 2022, Theorem 2) and empirical results do not lead to definitive conclusions (S. Wang & Ji, 2024, Table 5).

This chapter challenges the presumed superiority of U-FedVARP under client participation heterogeneity. Both theoretical and experimental contributions indicate that the relative effectiveness of U-FedVARP and U-FedAvg varies depending on the specific levels of data heterogeneity and client participation heterogeneity.

In the remainder of the chapter, we focus on the unbiased versions of the two algorithms. However, for simplicity, we refer to them simply as FedVARP and FedAvg.

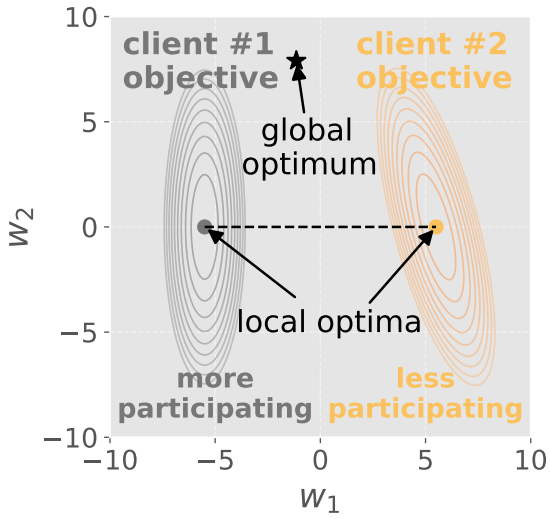
## 3.3 Proposed Staleness-Aware FL Algorithm: FedStale

We start questioning the expected superiority of FedVARP under client participation heterogeneity though the following illustrative example.

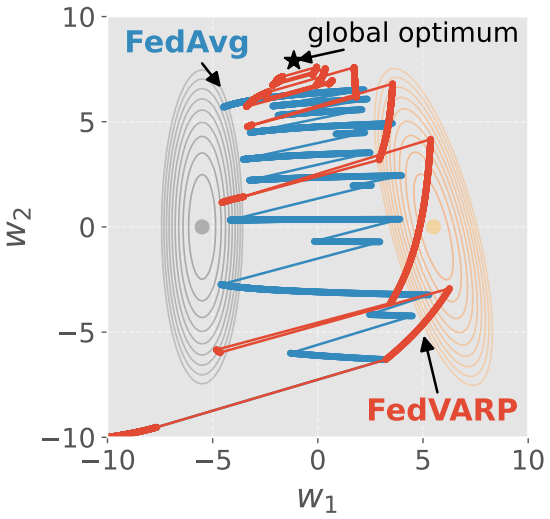
### 3.3.1 A Motivating Example

Figure 3.1a considers a two-clients scenario with quadratic bidimensional objectives $\{F_i(\boldsymbol{w}), i = 1, 2, \boldsymbol{w} \in \mathbb{R}^2\}$. The global optimum $\boldsymbol{w}^*$, minimizer of $F(\boldsymbol{w}) \triangleq \frac{1}{2}F_1(\boldsymbol{w}) + \frac{1}{2}F_2(\boldsymbol{w})$, does not align with the average of the local optima $\{\boldsymbol{w}_i^*, i = 1, 2\}$. Clients participate according to Bernoulli distributions with parameters $\{p_i, i = 1, 2\}$ and a skewed participation ratio $p_1/p_2 = 100$.

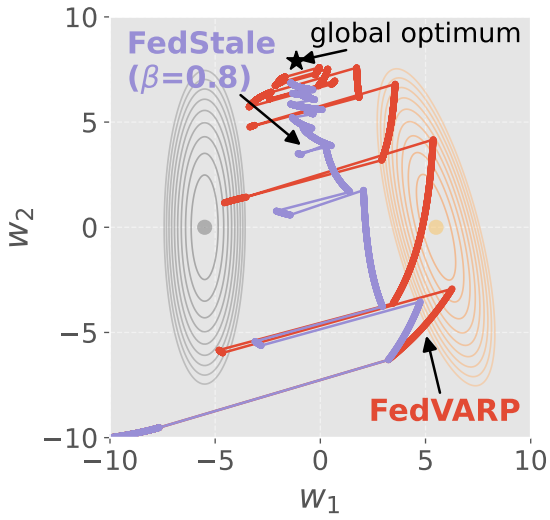
Figure 3.1b compares the model trajectories of FedAvg and FedVARP over $T = 4000$ rounds, starting from $\boldsymbol{w}^{(1)}$=(-10,-10) and running the experiments with same clients participation processes for comparability. Both algorithms initially share the same trajectory, driven solely by the participation of client 1, who targets $\boldsymbol{w}_1^*$. When client 2 first participates, the global update dramatically shifts towards $\boldsymbol{w}_2^*$ due to the reweighting factor $1/p_2$. As client 2 stops participating, the two trajectories diverge: FedAvg reverts to approaching $\boldsymbol{w}_1^*$, influenced only by the participating client 1, while FedVARP continues to factor in stale updates from client 2. Both algorithms eventually converge to the global optimum $\boldsymbol{w}^*$, consistently with the fact that both Eqs. (3.5) and (3.7) are *unbiased*. However, FedAvg suffers large variance and slow convergence due to significant shifts whenever client 2 participates, whereas FedVARP is affected by progressively more outdated updates from the less participating client, also resulting in suboptimal trajectories with abrupt corrections. Figure 3.1d compares the losses over these trajectories and confirms that both FedAvg and FedVARP exhibit high variability for distinct reasons. A hybrid approach that combines these two dynamics can potentially improve overall performance.



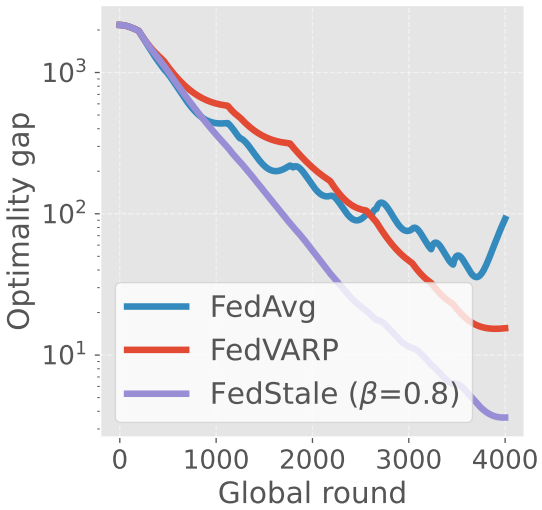

(a) Optimization objectives.




(b) Trajectories: FedAvg, FedVARP.




(c) Trajectories: FedVARP, FedStale.




(d) Convergence curves.

Figure 3.1: Comparison of FedAvg, FedVARP, and FedStale in a two-clients, 2D quadratic setting with *heterogeneous* client participation. **Fig. 3.1a:** Contour plots of client objectives, their local optima, and global optimum. Client participation ratio is $p_1/p_2 = 100$. **Fig. 3.1b:** Trajectories by FedAvg and FedVARP over T=4000 rounds with K=5 local iterations each. While both algorithms target the global optimum, FedAvg struggles with large variance and FedVARP follows suboptimal paths due to stale updates. **Fig. 3.1c:** FedStale ($\beta$=0.8) follow a more stable trajectory under heterogeneous client participation. **Fig. 3.1d:** Learning curves of FedAvg, FedVARP, and FedStale over 10 runs. With a lower weight on stale updates ($\beta$=0.8), FedStale achieves faster convergence to the global optimum.

**Algorithm 4:** `FedStale` (Federated Learning with Stale Client Updates)

**Input:** $\{\boldsymbol{h}_i^{(1)} = \mathbf{0}, p_i : \forall i\}$, $\beta$; **Output:** $\{\Delta^{(t)}_{\texttt{FedStale}} : \forall t\}$
**for** $t = 1, \dots, T$ **do**
  **Procedure** `Aggregate`$(\{\Delta_i^{(t)}\}_{i\in\mathcal{S}^{(t)}}, \beta)$:
  $\Delta^{(t)}_{\texttt{FedStale}} \leftarrow \frac{\beta}{N}\sum_{i=1}^{N} \boldsymbol{h}_i^{(t)} + \frac{1}{N}\sum_{i\in\mathcal{S}^{(t)}}(\Delta_i^{(t)} - \beta\boldsymbol{h}_i^{(t)})/p_i$
  **for** $i \in \mathcal{S}^{(t)}$ **do**
    $\boldsymbol{h}_i^{(t+1)} \leftarrow \Delta_i^{(t)}$ `// Update memory`

### 3.3.2 A Convex Combination of Fresh and Stale Updates

In Figs. 3.1c and 3.1d, a convex combination of `FedAvg` and `FedVARP` updates with a weighting parameter $\beta = 0.8$ results in a more stable trajectory and achieves faster convergence than either algorithm alone. This suggests that, in environments with heterogeneous client participation, parameterizing the weight to stale updates allows us to interpolate the two negative extremes of large variance (`FedAvg`) and outdated trajectories (`FedVARP`). Motivated by these observations, we propose `FedStale` (Federated Averaging with Stale Updates), outlined in Algorithm 4. In each round, `FedStale` updates the global model through a convex combination of fresh and stale updates, with parameter $\beta$ in the range $[0, 1]$:

$$\Delta^{(t)}_{\texttt{FedStale}} = (1-\beta)\Delta^{(t)}_{\texttt{FedAvg}} + \beta\Delta^{(t)}_{\texttt{FedVARP}} \tag{3.8}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\beta\boldsymbol{h}_i^{(t)} + \frac{1}{N}\sum_{i\in\mathcal{S}^{(t)}}\frac{\Delta_i^{(t)} - \beta\boldsymbol{h}_i^{(t)}}{p_i}. \tag{3.9}$$

`FedStale` interpolates between the behaviors of `FedAvg` when $\beta = 0$ and `FedVARP` when $\beta = 1$, merging the two algorithms into a single, versatile framework. Moreover, by adjusting $\beta$, `FedStale` can control the influence of stale updates, allowing for a continuum of behaviors that adapts with the specific level of client data and participation heterogeneity.

**Requirements.** In its operation, `FedStale` maintains the same computational and communication complexity as `FedVARP`, with tuning $\beta$ as the only additional requirement. Section 3.5 shows that a coarse adjustment of $\beta$ (e.g., $\beta \in 0, 0.2, 0.5, 0.8, 1$) provides reasonably good performance across varied settings, thus eliminating the need for fine-tuning.

As for storage requirements, `FedStale` mirrors `FedVARP` and other global variance reduction methods by storing stale updates from *all clients* at the server. Typically, servers possess more resources than clients, mitigating potential storage issues. Methods that avoid additional storage would otherwise escalate computational and communication demands on clients or necessitate *full client participation* in certain rounds—a requirement that may be overly demanding or even impractical, as will be discussed in the following section.

### 3.3.3 Comparison to Related Work

We discuss variance reduction methods emerged for centralized and distributed optimization. Some have already been adapted to federated learning, while others are discusses for potential applicability.

**FedLaAvg** (Yan et al., 2024), **MIFA** (Gu et al., 2021), **AFA-CD** and **AFA-CS** (H. Yang et al., 2022), similarly to FedVARP, address partial yet homogeneous client participation by storing the stale model updates for each client. However, their approach of uniformly weighting fresh and stale updates, through a SAG-based (Schmidt, Le Roux, & Bach, 2017) global variance reduction step, *biases* the global model leading to objective inconsistency.

**SVRG-based Variance Reduction Methods** (Johnson & Zhang, 2013; Lei, Ju, Chen, & Jordan, 2017; Nguyen, Liu, Scheinberg, & Takáč, 2017; Fang, Li, Lin, & Zhang, 2018) trade storage demands with computation needs by periodically calculating, in centralized settings, full or large-batch gradients. Although offering superior theoretical performance over SAGA-based (Defazio et al., 2014) variance reduction methods like FedVARP, their extension to FL settings is constrained by the impractical requirement for *all clients* to participate simultaneously during certain training rounds.

**SCAFFOLD** (Karimireddy et al., 2020) uses control variates to correct for data heterogeneity errors. Adapting this method to handle participation heterogeneity would require clients to perform local SAGA-like (Defazio et al., 2014) corrections, thereby *doubling the communication overhead* as clients must transmit both the model updates and correction vectors to the server. While this extension remains a topic for future research, we underscore the additional communication complexity involved.

In contrast to previous work, FedStale, much like FedVARP, performs corrections at the server level without involving clients in variance reduction, thus maintaining the same communication overhead as FedAvg and still matching SCAFFOLD's convergence rates.

## 3.4 Convergence Analysis

**Assumption 6** ($L$-smoothness)**.** *The local objective functions are $L$-smooth, i.e., $\|\nabla F_i(\boldsymbol{u}) - \nabla F_i(\boldsymbol{v})\| \leq L \|\boldsymbol{u} - \boldsymbol{v}\|$, $\forall \boldsymbol{u}, \boldsymbol{v}, i$.*

**Assumption 7** (Bounded variance at client level)**.** *The stochastic gradient at each client is an unbiased estimator of the local gradient: $\mathbb{E}_{\xi_i \sim \mathcal{D}_i}[\nabla F_i(\boldsymbol{w}, \xi_i)] = \nabla F_i(\boldsymbol{w})$, with bounded variance: $\mathbb{E}_{\xi_i \sim \mathcal{D}_i} \|\nabla F_i(\boldsymbol{w}, \xi_i) - \nabla F_i(\boldsymbol{w})\|^2 \leq \sigma^2$, $\forall \boldsymbol{w}, i$. The stochastic gradient noise is independent across clients, rounds, and local steps.*

**Assumption 8** (Bounded variance across clients)**.** *There exists a constant $\sigma_g^2 > 0$ such that the difference between the local gradient at the $i$-th client and the global gradient is bounded, that is $\|\nabla F_i(\boldsymbol{w}) - \nabla F(\boldsymbol{w})\|^2 \leq \sigma_g^2$, $\forall \boldsymbol{w}, i$.*

**Assumption 9** (Partial and heterogeneous client participation)**.** *In each round $t$, client $i$ participates with a probability $p_i$, independently of previous rounds and other clients.*

Assumptions 6–8 are standard in federated learning convergence analysis (H. Yang et al., 2020; S. Wang & Ji, 2022; Cho et al., 2023). The terms $\sigma^2$ and $\sigma_g^2$ denote the variances from *stochastic gradients* and *data heterogeneity*, respectively. Assumption 4, which models *client participation heterogeneity*, also appears in some prior works (S. Wang & Ji, 2022, 2024). Exploring more complex participation dynamics, following the methodologies in (S. Wang & Ji, 2022; Rodio, Faticanti, et al., 2023b), remains a task for future research.

### 3.4.1 FedStale, Upper Bound

We start by providing an upper bound for `FedStale`'s convergence. We defer detailed proofs to the Appendix B.

**Theorem 3.4.1** (Convergence of `FedStale`, upper bound)**.** *Under Assumptions 6–9, if the client and server learning rates, $\eta_c$ and $\eta_s$, are chosen such that $\eta_c \leq \frac{1}{8LK}$ and $\eta_s \leq \min\left\{\frac{Np_{var}}{12(1-\beta)^2}, \frac{p_{var}p_{min}}{3\beta^2 p_{avg}}\right\}$, the sequence of `FedStale` iterates satisfies*

$$\min_{t\in\{1,T\}} \mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)}_{\texttt{FedStale}})\right\|^2 \leq \underbrace{\mathcal{O}\left(\frac{F(\boldsymbol{w}^{(1)})-F^*}{\eta_s\eta_c KT}\right)}_{\text{iterate initialization error}} + \underbrace{\mathcal{O}\left(\frac{\beta^2\eta_s\eta_c LKH^{(1)}}{p_{var}p_{min}T}\right)}_{\text{memory initialization error}} + \underbrace{\mathcal{O}\left(\left[\frac{1}{N}+\beta^2\frac{p_{avg}}{p_{min}}\right]\frac{\eta_s\eta_c L\sigma^2}{p_{var}}\right)}_{\text{stochastic gradient error}} + \underbrace{\mathcal{O}\left(\left[\frac{(1-\beta)^2}{N}+\beta^2\eta_c^2L^2K(K-1)\frac{p_{avg}}{p_{min}}\right]\frac{\eta_s\eta_c LK\sigma_g^2}{p_{var}}\right)}_{\text{error from data heterogeneity}}, \tag{3.10}$$

*where $F^* \triangleq \min_{\boldsymbol{w}} F(\boldsymbol{w})$, $H^{(1)} \triangleq \frac{1}{N}\sum_{i=1}^N ||\nabla F_i(\boldsymbol{w}^{(1)}) - \boldsymbol{h}_i^{(1)}||^2$, $p_{var} \triangleq (\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i})^{-1}$, $p_{avg} \triangleq \frac{1}{N}\sum_{i=1}^N p_i$, and $p_{min} \triangleq \min_i p_i$.*

Theorem 3.4.1 relates `FedStale`'s convergence to the iterate and memory initial errors, and variances from stochastic gradients ($\sigma^2$) and data heterogeneity ($\sigma_g^2$). It also quantifies the impact of client participation heterogeneity through the terms $p_{\text{var}}$, $p_{\text{avg}}$, and $p_{\text{min}}$. By scaling the client learning rate as $\mathcal{O}(\frac{1}{\sqrt{T}})$, all error components asymptotically vanish, proving the *unbiasedness* of update (3.9).

Theorem 3.4.1 integrates `FedAvg` and `FedVARP` convergence analyses in a single framework, providing new insights on their different behaviors. First, for $\beta = 0$, the bound provides a convergence result for `FedAvg`.

**Corollary 3.4.2** (Convergence of `FedAvg`, upper bound)**.** *Under same assumptions as Theorem 3.4.1, the sequence of `FedAvg` iterates satisfies*

$$\min_{t\in\{1,T\}} \mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)}_{\texttt{FedAvg}})\right\|^2 \leq \underbrace{\mathcal{O}\left(\frac{F(\boldsymbol{w}^{(1)})-F^*}{\eta_s\eta_c KT}\right)}_{\text{iterate initialization error}} + \underbrace{\mathcal{O}\left(\frac{\eta_s\eta_c L\sigma^2}{Np_{var}}\right)}_{\text{stochastic gradient error}} + \underbrace{\mathcal{O}\left(\frac{\eta_s\eta_c LK\sigma_g^2}{Np_{var}}\right)}_{\text{error from data heterogeneity}}. \tag{3.11}$$

Corollary 3.4.2 shows that client participation heterogeneity only affects `FedAvg` convergence through the variance factor $1/p_{\text{var}}$. This term captures the variability of participation probabilities $p_i$ and is minimized—and equal to $(1 - p_{\text{avg}})/p_{\text{avg}}$—when client participation is homogeneous. Conversely, this variance term increases with larger participation heterogeneity, and may become the dominant factor in Eq. (3.11) that slows down `FedAvg` convergence. This justifies our observations for `FedAvg` in Figure 3.1b.

Second, for $\beta = 1$, Theorem 3.4.1 extends FedVARP known convergence results (Jhunjhunwala et al., 2022, Theorem 2) to heterogeneous client participation.

**Corollary 3.4.3** (Convergence of FedVARP, upper bound)**.** *Under the same assumptions as in Theorem 3.4.1, FedVARP's iterates satisfy*

$$\min_{t\in\{1,T\}} \mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)}_{\texttt{FedVARP}})\right\|^2 \leq \underbrace{\mathcal{O}\left(\frac{F(\boldsymbol{w}^{(1)})-F^*}{\eta_s\eta_c T}+\frac{\eta_s\eta_c H^{(1)}}{p_{var}p_{min}T}\right)}_{\text{iterate and memory initialization errors}} + \underbrace{\mathcal{O}\left(\frac{\eta_s\eta_c L p_{avg}\sigma^2}{p_{var}p_{min}}\right)}_{\text{stochastic gradient error}} + \underbrace{\mathcal{O}\left(\frac{\eta_s\eta_c^3 L^3K^2(K-1)p_{avg}\sigma_g^2}{p_{var}p_{min}}\right)}_{\text{error from data heterogeneity}}. \tag{3.12}$$

We highlight two differences with respect to FedAvg. First, FedVARP mitigates *data heterogeneity error*: by scaling the learning rate $\eta_c$ as $\mathcal{O}(T^{-1/2})$, the term in $\sigma_g^2$ decreases as $\mathcal{O}(T^{-3/2})$ versus $\mathcal{O}(T^{-1/2})$ for FedAvg in (3.11). However, FedVARP amplifies the stochastic gradient error through the ratio $p_{\text{avg}}/p_{\text{min}}$, and this terms may become dominant as *client participation* becomes more *heterogeneous*. This drawback, caused from stale updates, was not highlighted by earlier analyses, which considered only *homogeneous* client participation.

### 3.4.2 FedStale, Lower Bound

One may wonder whether the appearance of the factor $1/p_{\text{min}}$ in FedVARP bound may not be just an artifact of our proof technique. The following lower bound for FedVARP and FedStale convergence suggests that this is not the case.

**Theorem 3.4.4** (Convergence of FedStale, lower bound)**.** *Under Assumption 6, for any time horizon $T \leq \frac{d-1}{2}$, there exist $N$ local objectives $\{F_i(\boldsymbol{w}) : \mathbb{R}^d \to \mathbb{R}\}$ for which the iterates of any first-order black-box optimization procedure which leverages both fresh and stale updates satisfy*

$$\min_{t\in\{1,T\}} \mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)}_{\texttt{FedStale}})\right\|^2 \geq \Omega\left(\frac{F(\boldsymbol{w}^{(1)})-F^*}{p_{min}^3T^3+1}\right). \tag{3.13}$$

Theorem 3.4.4 proves that FedStale for any $\beta > 0$, and then FedVARP, requires at least $T \geq \Omega(1/p_{\text{min}})$ rounds to minimize objective (3.1).

### 3.4.3 Finding the optimal weight $\beta^*$

FedStale leverages the parameter $\beta$ to balance the multiple sources of variance in Theorem 3.4.1: stochastic gradients ($\sigma^2$), data heterogeneity ($\sigma_g^2$), and client participation heterogeneity (through the ratio $p_{\text{avg}}/p_{\text{min}}$).

The quadratic dependency on $\beta$ of the bound in Theorem 3.4.1, Eq. (3.10), guarantees a unique minimizer $\beta^* \in [0,1]$, generally different from the boundaries values of 0 and 1. The optimal $\beta^*$ is:

$$\beta^* = \frac{\sigma_g^2/N}{a_1\frac{p_{\text{avg}}}{p_{\text{min}}}\frac{\sigma^2}{K}+\left[\frac{1}{N}+a_2\frac{p_{\text{avg}}}{p_{\text{min}}}\eta_c^2L^2K(K-1)\right]\sigma_g^2}, \tag{3.14}$$

where $a_1$ and $a_2$ are positive numerical constants.

In practice, computing $\beta^*$ is challenging due to the unknowns $L$, $\sigma^2$, and $\sigma_g^2$ in Eqs. (3.10) and (3.14), which are difficult to estimate since they depend on the client objectives and on the specific heterogeneity setting. Moreover, Eq. (3.10) provides a worst-case upper bound for the gradient norm, but convergence may be significantly faster. For instance, the bound becomes vacuous as $p_{\min}$ approaches zero, yet, if all clients share the same local objective, convergence is unaffected by non-participating clients. Therefore, we primarily use Eq. (3.14) to derive *qualitative*, yet important guidelines.

The monotonically increasing behavior of $\beta^*$ with $\sigma_g^2$ in Eq. (3.14) suggests

*Guideline A: Increase the weight to stale updates, $\beta$, when data heterogeneity, $\sigma_g^2$, increases.*

Guideline A is in line with our previous comparison of Corollary 3.4.3 and Corollary 3.4.2. As we observed, stale updates become more beneficial when data heterogeneity ($\sigma_g^2$) is dominant. Conversely, as data heterogeneity decreases, the benefit from stale updates diminishes. This outcome is intuitive: in the extreme case where all clients share same datasets, each local objective aligns with the global objective. Relying solely on updates from participating clients is then optimal, as stale updates may only introduce unnecessary noise.

The monotonically decreasing behavior of $\beta^*$ with the ratio $p_{\text{avg}}/p_{\min}$ in Eq. (3.14) informs

*Guideline B: Decrease the weight to stale updates, $\beta$, as client participation heterogeneity, $p_{avg}/p_{min}$, increases.*

Also Guideline B is grounded in intuition: as client participation is more *heterogeneous* ($p_{\min} \ll p_{\text{avg}}$), the least participating clients refresh their stale update less frequently, leading to more *outdated* global updates: leveraging them may yield poor results. Conversely, when client participation is *homogeneous* ($p_{\min} \approx p_{\text{avg}}$), all clients *uniformly* refresh their update, and global variance reduction methods perform best.

## 3.5 Experimental Evaluation

We evaluate the performance of `FedStale` in experiments. The source code of our experimental framework is in the supplementary material and will made publicly available after publication.

### 3.5.1 Experimental setup

**System, Datasets, and Models.** We simulate a FL system with $N = 24$ clients. We consider two image classification tasks: handwritten digits recognition on MNIST (L. Deng, 2012) and natural image classification on CIFAR-10 (Krizhevsky & Hinton, 2009). Each dataset has 10 classes, or labels. We train two convolutional neural network (CNN) models with slightly different architectures. These models, with cross-entropy loss, define non-convex objectives (3.1).

**Participation heterogeneity.** Client participation follows a Bernoulli distribution, in line with Assumption 9. To simulate heterogeneity in client participation, we randomly divide clients into two groups based on their participation dynamics: one group of clients always participate, while the other, less participating group, have participation probabilities $p_{\min}$ varying in the range

$\{50, 20, 10, 5, 2, 1, 0.5, 0.2\}\%$. The ratio $p_{\text{avg}}/p_{\text{min}}$ specifies the degree of client participation heterogeneity.

**Data heterogeneity.** Following existing work (Sattler et al., 2021), we simulate data heterogeneity across clients' local datasets by: 1) randomly partitioning the dataset among clients; 2) swapping a fraction $\hat{\sigma}_g^2$ of two labels in the second group, with $\hat{\sigma}_g^2 \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$. The empirical parameter $\hat{\sigma}_g^2$ mirrors the theoretical variance $\sigma_g^2$ in Assumption 8, measuring the degree of data heterogeneity: $\hat{\sigma}_g^2 = 0$ represents homogeneous (IID) data distributions and $\hat{\sigma}_g^2 = 1$ indicates maximum heterogeneity among client datasets.

**Baselines.** We compare `FedAvg` ($\beta = 0$), `FedVARP` ($\beta = 1$), and `FedStale` (for $\beta \in \{0.2, 0.5, 0.8\}$) across diverse heterogeneity settings. Previous work (Jhunjhunwala et al., 2022) showed that, under partial client participation, `FedVARP` consistently outperformed both `MIFA` (Gu et al., 2021), due to its biased variance correction, and `SCAFFOLD` (Karimireddy et al., 2020), that also incurs higher communication costs. We benchmark all algorithms over a consistent time horizon, corresponding, on average, to the first ten participation instances by the least participating client. Clients perform $K = 5$ local iterations. We use a batch size of 128 in all experiments. For all algorithms, we fix the server learning rate $\eta_s$ to 1 and tune the client learning rate $\eta_c$ over the grid $\{10^{-2}, 10^{-2.5}, 10^{-3}, 10^{-3.5}, 10^{-4}\}$. While we initially assume all algorithms have exact knowledge of client participation probabilities, we relax this assumption in Section 3.5.3. We average results over three random seeds.

### 3.5.2 Existence of different regimes

In Figure 3.2, we show the empirical values of $\beta$ that yield the highest test accuracies among `FedAvg` ($\beta = 0$), `FedVARP` ($\beta = 1$), and `FedStale` ($\beta \in \{0.2, 0.5, 0.8\}$) across diverse heterogeneity settings on the MNIST dataset. We denote these values as $\beta_{\text{opt}}$.

The heatmap shows how $\beta_{\text{opt}}$ varies with client participation heterogeneity ($p_{\text{avg}}/p_{\text{min}}$, in the x-axis) and data heterogeneity ($\hat{\sigma}_g^2$, in the y-axis). Moreover, each cell reports the performance gains of the best setting for `FedStale`. $\Delta_0$ and $\Delta_1$ denote, respectively, the accuracy improvements of `FedStale`($\beta_{\text{opt}}$) over `FedAvg` ($\beta = 0$) and `FedVARP` ($\beta = 1$). This visualization aggregates results from 720 training runs, across 8 participation heterogeneity setups and 6 data heterogeneity setups, each comparing 5 algorithms for 3 independent seeds.

**Multiple regimes in heterogeneity settings.** No single algorithm consistently outperforms others across all settings. Instead, Figure 3.2 shows different zones where the best-performing algorithm depends on the interplay between data heterogeneity ($\hat{\sigma}_g^2$) and client participation heterogeneity ($p_{\text{avg}}/p_{\text{min}}$). The observed trends reflect our qualitative guidelines.

Specifically, Figure 3.2 identifies three distinct zones where specific patterns in performance emerge: *i)* `FedVARP` yields the best results for large data heterogeneity ($\hat{\sigma}_g^2 \geq 0.2$) and homogeneous client participation ($p_{\text{min}} \approx p_{\text{avg}}$), favoring larger weights to stale updates ($\beta_{\text{opt}} = 1$); *ii)* conversely, `FedAvg` best fits settings with low data heterogeneity ($\hat{\sigma}_g^2 \leq 0.2$) and large participation heterogeneity ($p_{\text{avg}} \geq 25 p_{\text{min}}$), where using stale updates overall reduces performance; *iii)* finally, a significant transitional zone exists where moderate heterogeneity levels ($3p_{\text{min}} \leq p_{\text{avg}} \leq 25p_{\text{min}}$) favor intermediate $\beta_{\text{opt}}$ values ($\beta_{\text{opt}} \in \{0.2, 0.5, 0.8\}$), which yield the best performance.

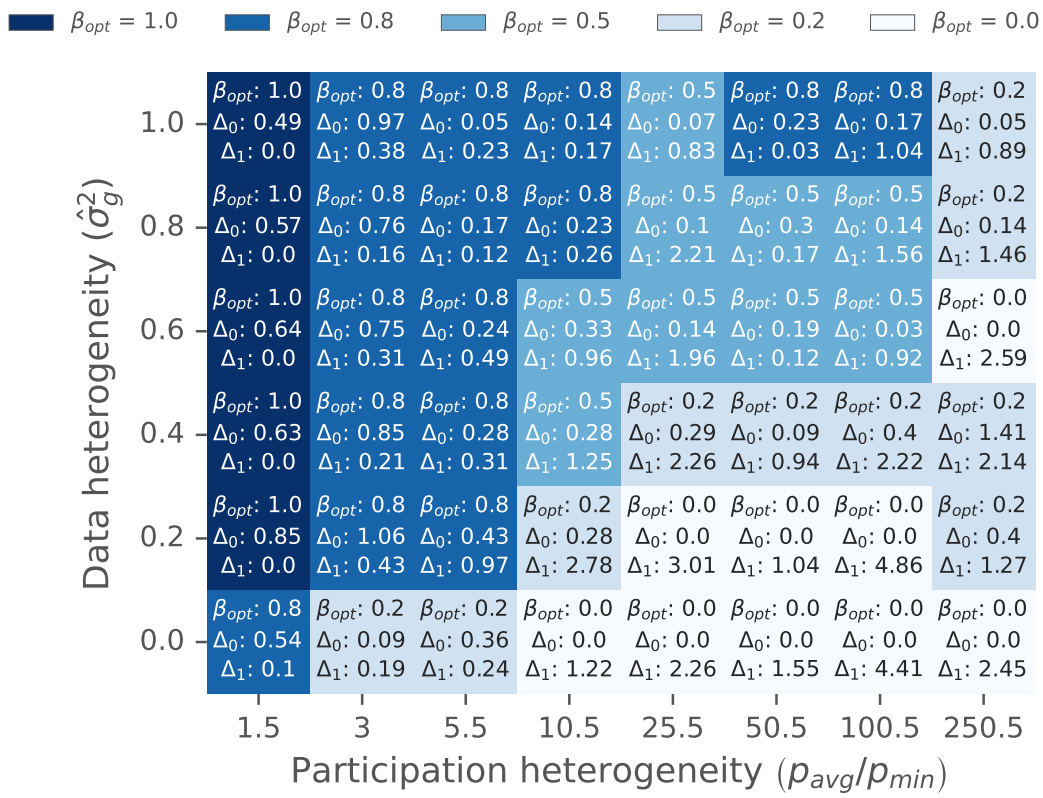
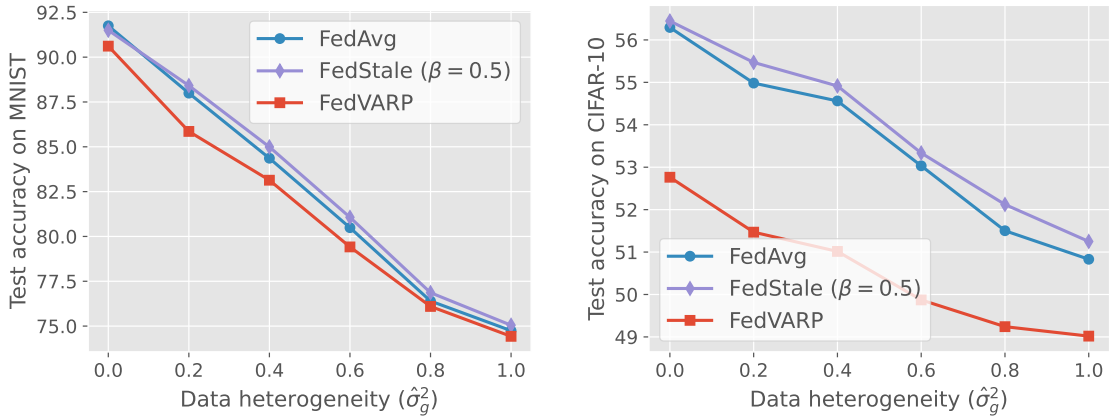



Figure 3.2: $\beta_{\text{opt}}$ values for FedAvg ($\beta$=0), FedVARP ($\beta$=1), and FedStale ($\beta \in \{0.2, 0.5, 0.8\}$) across 48 heterogeneity settings on the MNIST dataset. Color gradients range from lighter shades ($\beta_{\text{opt}}$=0) to darker shades ($\beta_{\text{opt}}$=1).

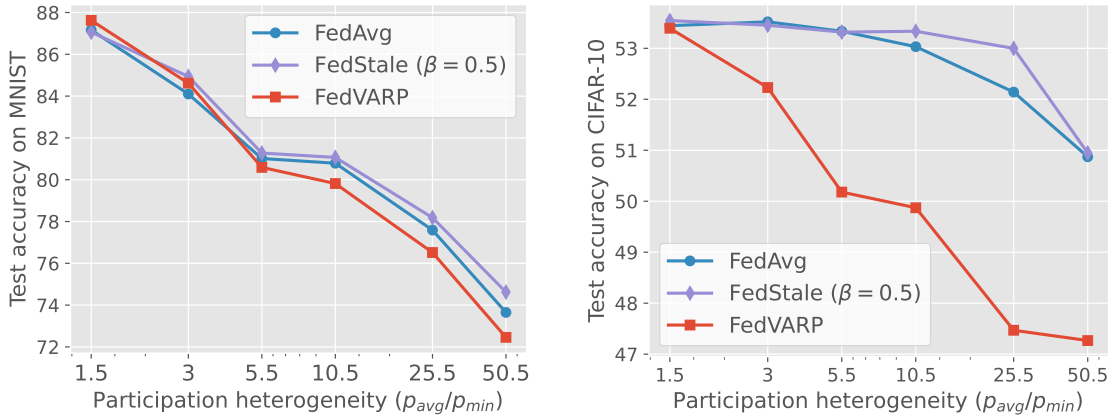
Overall, `FedStale` prevails in 72% of scenarios within our $6 \times 8$ grid, against `FedVARP`, 18%, and `FedAvg`, 10%. Therefore, `FedStale` plays a key role—we believe—in bridging the gaps posed by `FedAvg` and `FedVARP` in real-world federated settings, which often exhibit intermediate levels of client data and participation heterogeneity.

**Effect of data heterogeneity.** Figure 3.2 shows that $\beta_{\text{opt}}$ increases with data heterogeneity, in line with Guideline A. Figure 3.3 explores this trend in more detail, by holding participation heterogeneity constant at $p_{\text{avg}}/p_{\text{min}} = 10$ and varying data heterogeneity ($\hat{\sigma}_g^2$). For all algorithms, increased data heterogeneity corresponds to lower test accuracies. In Figures 3.3a and 3.3b, `FedStale` ($\beta = 0.5$), without particular fine-tuning, consistently outperforms `FedVARP` in settings of moderate participation heterogeneity and improves over `FedAvg` as client data become heterogeneous (already at $\hat{\sigma}_g^2 \geq 0.2$). Moreover, Figure 3.3b shows that `FedVARP`, despite its overall lower accuracy, proves to perform better in extremely heterogeneous data scenarios (when $\hat{\sigma}_g^2 \geq 0.8$).




(a) MNIST dataset (b) CIFAR-10 dataset

Figure 3.3: Test accuracy of FedAvg ($\beta$=0), FedVARP ($\beta$=1), and FedStale ($\beta$=0.5) varying data heterogeneity at fixed participation ratio $p_{\text{avg}}/p_{\text{min}} = 10$.




(a) MNIST dataset (b) CIFAR-10 dataset

Figure 3.4: Test accuracy of FedAvg ($\beta$=0), FedVARP ($\beta$=1), and FedStale ($\beta$=0.5) varying client participation ratio at fixed data heterogeneity $\hat{\sigma}_g^2 = 0.6$.

**Effect of participation heterogeneity.** Figure 3.2 shows that $\beta_{\text{opt}}$ decreases as the participation heterogeneity ($p_{\text{avg}}/p_{\text{min}}$) increases, in line with Guideline B. Figure 3.4 details this dynamic by fixing data heterogeneity at $\hat{\sigma}_g^2 = 0.6$ and only varying participation heterogeneity. In both Figures 3.4a and 3.4b, it is evident how `FedVARP` performs well when client participation is homogeneous ($p_{\text{min}} \approx p_{\text{avg}}$), yet struggles with increasing participation heterogeneity. `FedAvg` exhibits dual behavior, which confirms that the usefulness of stale updates progressively diminishes as participation heterogeneity increases (already at $p_{\text{avg}} \geq 3p_{\text{min}}$). Figure 3.4b also shows that `FedStale` ($\beta = 0.5$), without specific tuning, maintains robust performance across a wide range of participation levels (until $p_{\text{avg}} \approx 25p_{\text{min}}$), and only drops accuracy at $p_{\text{avg}} \approx 50p_{\text{min}}$.

### 3.5.3 Online estimation of participation probabilities

We evaluate `FedStale` with online estimation of client participation probabilities, to simulate scenarios where these probabilities are unknown before training (Ribero et al., 2023; Rodio, Faticanti, et al., 2023b; S. Wang & Ji, 2024). To this purpose, we integrate `FedStale` with `FedAU` (S. Wang & Ji, 2024), a state-of-the-art algorithm for tracking client participation dynamics, that balances bias and variance in the estimation through a cutoff mechanism.

Figure 3.5 shows that the integration of `FedStale` with `FedAU`'s estimation technique still aligns with our guidelines. Moreover, `FedVARP` performs significantly worse than `FedStale`($\beta_{\text{opt}}$) when client participation probabilities are estimated ($\Delta_1$ values in Fig. 3.5). Also, we observe overall lower $\beta_{\text{opt}}$ values in this scenario. These trends suggest that methods leveraging stale updates, like `FedVARP`, might be particularly sensitive to inaccurate $p_i$ estimations.

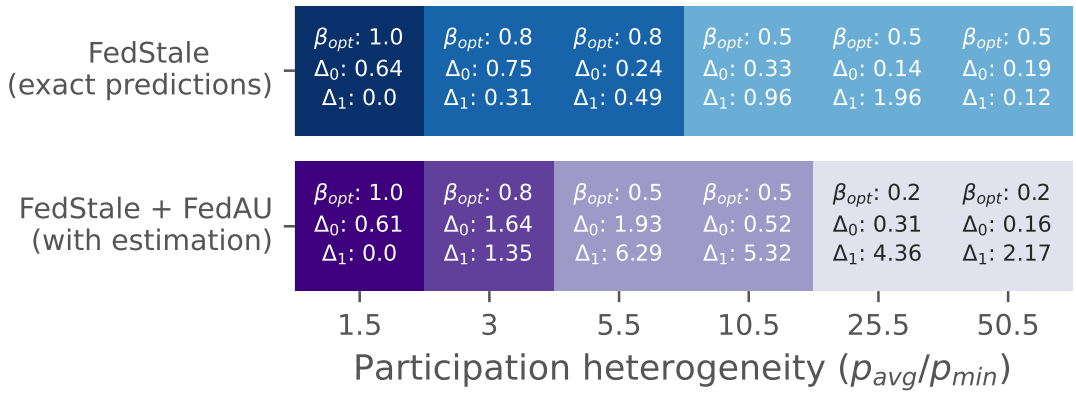



Figure 3.5: "Exact" vs. "Estimated" participation probabilities, $\hat{\sigma}_g^2 = 0.6$.

## 3.6 Conclusion

This chapter addresses global variance reduction in federated learning beyond the common assumption of homogeneous client participation. Unlike prior work, our research explores not only the advantages but also the challenges of leveraging stale client updates across varying heterogeneity scenarios. Our algorithm, `FedStale`, is equipped with guidelines: practitioners can decide whether storing stale updates is worthwhile or if solely relying on participating client updates

is more efficient. Exploring this tradeoff paves the way—we believe—for developing federated learning algorithms more attuned to the varied dynamics of client data and participation heterogeneity.

# CHAPTER 4

# Application to Wireless Networks with Lossy Communication Channels

This chapter is based on our work Rodio, Neglia, et al. (2023), published in 26th International Symposium on Wireless Personal Multimedia Communications (WPMC 2023).

## 4.1 Motivation

In many FL applications, such as training Google keyboard next-word prediction model, the clients are mobile devices such as smartphones or Internet of Things (IoT) devices, and the models are exchanged on wireless networks incurring potential transmission losses.

As observed in Chapter 1 (particularly in Section 1.3.2.2), lossy communication channels necessarily degrade the performance of FL training on wireless networks. Theoretical and experimental work (Eriş et al., 2021; Chandrasekaran et al., 2022; H. H. Yang et al., 2020; Ye et al., 2022; Baccarelli et al., 2022; M. Chen et al., 2021) has shown that packet losses affect the quality of the final model towards which the FL training algorithms converge as well as their convergence rate. Under medium/high network background traffic, the authors in (Eriş et al., 2021) measured a twofold training duration and a halved accuracy in the early stages of the training. Prior work (H. H. Yang et al., 2020; Ye et al., 2022; Baccarelli et al., 2022; M. Chen et al., 2021) analyzed the convergence of state-of-the-art FL algorithms under different channel assumptions. Specifically, the authors of (M. Chen et al., 2021) proved the existence of a non-vanishing error due to the lossy channels, which prevents the convergence of their direct model aggregation scheme to the optimal model, and proposed to reduce this error by opportunely allocating resources (e.g., transmission power, radio blocks) to control packet losses. Similar approaches to the one proposed in (M. Chen et al., 2021) have been considered to mitigate the effect of packet loss on wireless networks, relying on automatic repeat request (ARQ) and forward error correction (FEC) techniques (Wen et al., 2019; Su et al., 2023).

Despite these efforts, packet losses are typically caused by external factors beyond the control of the orchestrating server and can therefore be unavoidable (Section 1.3.2.2). Communication

protocols may define a maximum number of retransmissions, but these retransmissions can still fail. More importantly, we point out that targeting high transmission reliability in FL-oriented applications may be sub-optimal, as it usually comes at the detriment of training time and/or resource usage, e.g., in terms of wider sub-channel bandwidth, higher energy consumption, or both. These issues are even more exacerbated in resource-constrained scenarios, such as the IoT, where increasing communication reliability may result in a reduced device lifetime or may not be feasible. Moreover, the iterative nature of the gradient methods used for ML model training makes them robust against limited errors at intermediate calculations (Xing, Ho, Xie, & Wei, 2016).

For the aforementioned reasons, this chapter diverges from prior work (H. H. Yang et al., 2020; Ye et al., 2022; Chandrasekaran et al., 2022; Baccarelli et al., 2022; M. Chen et al., 2021; Wen et al., 2019; Su et al., 2023), which primarily focused on loss mitigation. Instead, we address the fundamental question of *whether FL algorithms can achieve optimal model convergence despite packet losses*. Our response is affirmative, necessitating only slight adjustments to the classic `FedAvg` (McMahan et al., 2017) algorithm.

More in detail, we consider a FL framework where losses can occur in the downlink, uplink, or both, and loss probabilities can differ among clients. Indeed, the channel quality in wireless networks can vary according to per-user characteristics, such as the relative positioning of the transmitter and the receiver, the device transmission power, the selected frequency channel. As a result, the clients will not participate evenly in the training process, potentially leading to learning a biased ML model (J. Wang et al., 2020; Rodio, Faticanti, et al., 2023a). Thus, the design of an aggregation strategy becomes critical to ensure the convergence of the FL model in the presence of packet losses. Previous works (M. Chen et al., 2021; Su et al., 2023) proposed direct model aggregation schemes, denoted in the following as DMA-PL, which stands for "Direct Model Aggregation with Packet Losses" and UDMA-PL, which stands for Unbiased DMA-PL. We will discuss and compare our approach to them in the rest of the chapter.

This work makes the following novel contributions:

- We propose `UPGA-PL`, a novel algorithm that aggregates pseudo-gradients, instead of models, and considers the client loss probabilities—therefore its name "Unbiased Pseudo-Gradient Aggregation with Packet Losses". Its complexity is comparable to `FedAvg`;
- We analytically prove `UPGA-PL`'s convergence to the optimal model, the same model which would have been learned over ideal, lossless channels. This result proves `UPGA-PL`'s ability to filter out the noise due to packet losses.
- We validate our analysis through numerical experiments. While losses largely affect the performance of the state-of-the-art algorithms (M. Chen et al., 2021; Su et al., 2023), `UPGA-PL` is robust to them and gains 5–10 percentage points on the test accuracy. Most importantly, even under severe losses, `UPGA-PL` achieves the same model's accuracy as `FedAvg` under lossless channels in less than 150 communication rounds.

## 4.2 Problem Description and Background

We consider the same optimization problem introduced in Section 1.2.1:

$$\min_{\boldsymbol{w}\in\mathbb{R}^d} \left[ F(\boldsymbol{w}) \triangleq \sum_{i=1}^{N} \alpha_i F_i(\boldsymbol{w}) \right], \tag{4.1}$$

recalling that $\{\alpha_i\}_{i=1}^N$ are positive coefficients, chosen by the server, such that $\sum_{i=1}^N \alpha_i = 1$. They represent the weight assigned to each client's objective function $F_i$. Typical choices, discussed in Section 1.2.1, are: 1) $\alpha_i = 1/N \; \forall i$, the server giving equal weight to all clients, 2) $\alpha_i = |D_i|/|D|$, with $D = \cup_{i=1}^N D_i$, the server giving equal weight to each data sample.

In the following, we recall two alternative algorithms to solve Problem (4.1). For both algorithms, the server receives either the $i$-th client's local model $\boldsymbol{w}_i^{(t,K)}$ or its model update $\Delta_i^{(t)} \triangleq \boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}_i^{(t,0)} = -\eta_c^{(t)} \sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})$. However, the server aggregation $\Delta^{(t)}$ and the new global model $\boldsymbol{w}^{(t+1)}$ differ in terms of the specific implementation for the `Aggregate()` and `ServerUpdate()` procedures, respectively:

- *Option 1)* The server can directly aggregate models computing $\boldsymbol{w}_{\text{DMA}}^{(t+1)} = \sum_{i=1}^N \alpha_i \boldsymbol{w}_i^{(t,K)}$: this scheme follows the `FedAvg` (McMahan et al., 2017) algorithm (Section 1.2.2, Algorithm 1), and we refer to it as Direct Model Aggregation (DMA).
- *Option 2)* The server can consider the model updates $\Delta_i^{(t)}$ received by the clients as *pseudo-gradients*, aggregate them as $\Delta^{(t)} = \sum_{i=1}^N \alpha_i \Delta_i^{(t)}$, and finally apply a global pseudo-SGD step as $\boldsymbol{w}_{\text{PGA}}^{(t+1)} = \boldsymbol{w}^{(t)} + \Delta^{(t)}$. This scheme corresponds to `FedOpt` (Reddi et al., 2021) (Section 1.2.2, Algorithm 2), with SGD used both as server and client optimizer. We denote it as Pseudo-Gradients Aggregation (PGA).

The DMA and PGA aggregation schemes are equivalent under lossless channels. However, in typical FL applications the information is transmitted over lossy channels, which ultimately affect the workflow of the considered FL algorithm.

We consider the same communication model as in M. Chen et al. (2021); Su et al. (2023): due to downlink losses, only a subset of clients $\mathcal{S}_{\text{DL}}^{(t)} \subseteq \mathcal{N}$ correctly receives the model $\boldsymbol{w}^{(t)}$ sent by the server and computes the local models $\{\boldsymbol{w}_i^{(t,K)}\}_{k\in\mathcal{S}_{\text{DL}}^{(t)}}$. On the other hand, due to losses in the upstream, the server gathers the updates (either the models $\boldsymbol{w}_i^{(t,K)}$ or the pseudo-gradients $\Delta_i^{(t)}$) only from a subset of clients $\mathcal{S}^{(t)} = \mathcal{S}_{\text{UL}}^{(t)} \subseteq \mathcal{S}_{\text{DL}}^{(t)}$.*

Since losses are random and can potentially differ among clients, the aggregation scheme plays an important role in the quality of the global ML model $\boldsymbol{w}^{(t+1)}$ learned by the FL training algorithm. Previous works (M. Chen et al., 2021; Su et al., 2023) considered the problem of FL training under lossy channels and proposed to generalize `FedAvg`'s DMA aggregation strategy by letting

*When transmission spans multiple packets, it is possible that only a subset of these packets may be affected by losses. Under these circumstances, the recipient could still leverage the partially received information effectively. Nonetheless, insufficient attention has been devoted to exploring this potential (H. H. Yang et al., 2020; Ye et al., 2022; M. Chen et al., 2021; Wen et al., 2019; Su et al., 2023). We identify a significant opportunity for future research, a topic we discuss more thoroughly in Section 6.1.

the server aggregate all received models, i.e., the models of all clients $i \in \mathcal{S}^{(t)}$:

$$\boldsymbol{w}_{\text{DMA-PL}}^{(t+1)} = \frac{\sum_{i\in\mathcal{S}^{(t)}} \alpha_i \boldsymbol{w}_i^{(t,K)}}{\sum_{i\in\mathcal{S}^{(t)}} \alpha_i}. \tag{4.2}$$

We refer to this strategy as Direct Model Aggregation with Packet Loss (DMA-PL). The authors of (M. Chen et al., 2021) also analyzed the convergence of the DMA-PL aggregation scheme under the effect of packet losses. They showed the existence of a generally non-vanishing error between the model trained under a non-zero loss rate and the optimal model towards which the training converges in the absence of losses:

$$\mathbb{E}\left[F(\boldsymbol{w}_{\text{DMA-PL}}^{(t+1)})\right] - F^* \leq \underbrace{A^t\left(F(\boldsymbol{w}^{(1)}) - F^*\right)}_{\text{vanishing term for small statistical heterogeneity}} + \underbrace{\frac{2\zeta_1}{L}\sum_{i=1}^{N}\alpha_i q_i \frac{1-A^t}{1-A}}_{\text{non-vanishing error due to statistical heterog. and packet loss}}, \tag{4.3}$$

where $q_i$ denotes the probability that the server does not receive client-$i$'s local model, $A = 1 - \frac{\mu}{L} + \frac{4\mu\zeta_2}{L}\sum_{i=1}^{N}\alpha_i q_i$, $L$ and $\mu$ are the $L$-smooth and $\mu$-strongly convex constants (they will be introduced in Assumptions 10, 11), and $\zeta_1, \zeta_2$ are parameters that quantify the *statistical heterogeneity* of the local datasets (the larger $\zeta_1$ and $\zeta_2$, the more heterogeneous the clients' data). We observe that, for non-zero loss probabilities and high statistical heterogeneity (large $\zeta_2$), it is possible that the bound does not guarantee convergence (when $A \geq 1$). On the contrary, for sufficiently small $\zeta_2$, (4.3) predicts linear convergence to a neighborhood of the optimal solution, whose size is proportional to the loss probabilities $\{q_i\}_{i=1}^N$. Motivated by these results, reference (M. Chen et al., 2021) focuses on resource allocation to reduce loss probabilities and minimize the non-vanishing term.

Moreover, due to losses, only a subset of the clients contributes to updating the new model at each round. Previous works (X. Li et al., 2020; J. Wang et al., 2020) have studied partial client participation due to client sampling, i.e., when the server samples at each round a subset of clients $\mathcal{S}^{(t)} \subseteq \mathcal{N}$. Convergence results in X. Li et al. (2020) require *unbiased* sampling for DMA to converge to the optimal model, i.e., the sampling scheme should satisfy $\mathbb{E}_{\mathcal{S}^{(t)}}[\boldsymbol{w}^{(t+1)}] = \sum_{i=1}^{N}\alpha_i \boldsymbol{w}_i^{(t,K)}$ (X. Li et al., 2020, Lemma 4), so that in expectation the $i$-th client contributes proportionally to its weight in the global objective (4.1). This observation suggests to unbias the DMA-PL scheme in (4.2) as follows:

$$\boldsymbol{w}_{\text{UDMA-PL}}^{(t+1)} = \sum_{i\in\mathcal{S}^{(t)}} \frac{\alpha_i}{1-q_i}\boldsymbol{w}_i^{(t,K)}, \tag{4.4}$$

so that the server counterbalances the more severe losses experienced by some clients with larger aggregation weights. We refer to this aggregation as Unbiased DMA-PL (UDMA-PL). However, by directly aggregating models, the UDMA-PL scheme suffers a possibly large variance due to the randomness in the set $\mathcal{S}^{(t)}$. Our analysis in Lemma A.1 confirms that this variance leads to a non-vanishing term, which prevents UDMA-PL from converging to the optimal model. Moreover, our experimental results in Section 4.4 confirm that UDMA-PL is not a practical solution.

In the next section, we present UPGA-PL, an unbiased aggregation scheme like UDMA-PL that filters out the noise due to losses and then succeeds in converging to the optimal model. To the best of our knowledge, only reference Ye et al. (2022) showed a similar result for a decentralized FL algorithm, but it required uplink and downlink channels to have the same loss probabilities, which is uncommon in wireless networks.

## 4.3 Proposed Packet Loss-Aware FL Algorithm: UPGA-PL

To solve the issues which characterized the DMA-PL and UDMA-PL aggregation schemes, we propose the Unbiased Pseudo-Gradient Aggregation strategy (UPGA-PL):

$$\boldsymbol{w}_{\text{UPGA-PL}}^{(t+1)} = \boldsymbol{w}^{(t)} + \sum_{i \in \mathcal{S}^{(t)}} \frac{\alpha_i}{1-q_i} \Delta_i^{(t)}. \tag{4.5}$$

Note that both UDMA-PL and UPGA-PL rely on the knowledge of the loss probabilities $\{q_i\}_{i=1}^N$. In practical scenarios, these probabilities can be estimated through channel measurements (Benko & Veres, 2002; Yajnik, Moon, Kurose, & Towsley, 1999).

As UDMA-PL, UPGA-PL is unbiased because it aggregates the pseudo-gradients with weights that compensate for clients' different loss probabilities. At the same time, by aggregating the pseudo-gradients $\{\Delta_i^{(t)}\}_{i \in \mathcal{S}^{(t)}}$ rather than the local models $\{\boldsymbol{w}_i^{(t,K)}\}_{i \in \mathcal{S}^{(t)}}$ (as UDMA-PL does), UPGA-PL can be seen as a stochastic approximation algorithm (Borkar, 2009) with stepsize $\eta_c^{(t)}$ (it is easy to verify that each $\Delta_i^{(t)}$ is proportional to $\eta_c^{(t)}$). Stochastic approximation theory suggests that convergence to the optimal model is guaranteed if $\eta_c^{(t)}$ decreases fast enough to filter out the noise due to the randomness in the set $\mathcal{S}^{(t)}$ (i.e., $\sum_t (\eta_c^{(t)})^2 < +\infty$), but also slow enough for the algorithm to be able to move from the initial tentative model ($\boldsymbol{w}^{(1)}$) to the optimal one (i.e., $\sum_t \eta_c^{(t)} = +\infty$). Our theoretical analysis below confirms these qualitative considerations: the UPGA-PL aggregation strategy enables the convergence of FL training algorithms to the optimal model even in the presence of lossy channels.

With abuse of language, we refer to the FL algorithm defined by the local update rule in (2.2) and the UPGA-PL aggregation scheme in (4.5) simply as UPGA-PL. The complete procedure is summarized in Algorithm 5. Similarly, we denote by DMA-PL and UDMA-PL the FL algorithms obtained replacing line 8 in Algorithm 5 with (4.2) and (4.4), respectively.

In the following, we analyze the convergence of UPGA-PL.

### 4.3.1 Convergence Analysis

For the analysis of the UPGA-PL algorithm, we make the following hypotheses. Assumptions 10 and 11 are standard in the literature on convex optimization (Bottou et al., 2018, Sections 4.1, 4.2). Assumptions 12 and 13 are standard hypothesis in the analysis of federated optimization algorithms (J. Wang et al., 2021; X. Li et al., 2020, Section 6.1).

**Assumption 10.** *$\{F_i\}_{i=1}^N$ are L-smooth: for all $\boldsymbol{v}$ and $\boldsymbol{w}$, $F_i(\boldsymbol{v}) \leq F_i(\boldsymbol{w}) + \langle \nabla F_i(\boldsymbol{w}), \boldsymbol{v} - \boldsymbol{w} \rangle + \frac{L}{2} \|\boldsymbol{v} - \boldsymbol{w}\|_2^2$.*

**Algorithm 5:** `UPGA-PL` (Unbiased Pseudo-Grad. Aggregation under Packet Loss)

**Input :** Initial model $\boldsymbol{w}^{(1)}$; Weights $\boldsymbol{\alpha} = \{\alpha_i\}_{i=1}^N$; Client loss probabilities $\boldsymbol{q} = \{q_i\}_{i=1}^N$; Learning rates $\{\eta_c^{(t)}\}_{t\in\mathcal{T}}$; Local steps $K$.

**for** *round* $t = 1, \ldots, T$ **do**
  **for** *client* $i = 1, \ldots, N$, *in parallel* **do**
    $\boldsymbol{w}_i^{(t,0)} = \boldsymbol{w}^{(t)}$;
    **for** $k = 0, \ldots, K-1$ **do**
      $\boldsymbol{w}_i^{(t,k+1)} = \boldsymbol{w}_i^{(t,k)} - \eta_c^{(t)} \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})$;
    $\Delta_i^{(t)} \leftarrow \boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}_i^{(t,0)}$;
  Receive $\{\Delta_i^{(t)}\}$ from a subset $\mathcal{S}^{(t)} \subseteq \mathcal{N}$ of clients;
  $\boldsymbol{w}_{\text{UPGA-PL}}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} + \sum_{i\in\mathcal{S}^{(t)}} \frac{\alpha_i}{1-q_i} \Delta_i^{(t)}$;

**Output:** Final model $\boldsymbol{w}_{\text{UPGA-PL}}^{(T+1)}$.

**Assumption 11.** *$\{F_i\}_{i=1}^N$ are $\mu$-strongly convex: for all $\boldsymbol{v}$ and $\boldsymbol{w}$, $F_i(\boldsymbol{v}) \geq F_i(\boldsymbol{w}) + \langle \nabla F_i(\boldsymbol{w}), \boldsymbol{v} - \boldsymbol{w} \rangle + \frac{\mu}{2} \|\boldsymbol{v} - \boldsymbol{w}\|_2^2$.*

**Assumption 12.** *Let $\mathcal{B}_i^{(t,k)}$ be a random batch sampled from the $i$-th device's local data uniformly at random. The variance of stochastic gradients in each device is bounded: $\mathbb{E}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) - \nabla F_i(\boldsymbol{w}_i^{(t,k)})\right\|^2 \leq \sigma_i^2$ for $i \in \mathcal{N}$.*

**Assumption 13.** *The expected squared norm of stochastic gradients is uniformly bounded, i.e., $\mathbb{E}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})\right\|^2 \leq G^2$ for $i \in \mathcal{N}$ and $t \in \mathcal{T}$, $j = 0, \ldots, E-1$.*

We use the indicator variable $\xi_i^{(t)}$ to denote the outcome of the $t$-th communication round between the server and the client $i$: $\xi_i^{(t)}$ equals one if and only if the server correctly receives client-$i$'s local model at round $t$.

**Assumption 14.** *At each round $t$, the communication outcomes $\{\xi_i^{(t)}\}_{i=1}^N$ are independent among clients. For each client $i$, the outcomes $\{\xi_i^{(t)}\}_{t\in\mathcal{T}}$ are independent and identically distributed (iid) over time with mean $\mathbb{E}[\xi_i^{(t)}] = 1 - q_i$.*

In Assumption 14, $q_i$ denotes the probability that the overall communication between the server and client $i$ fails either because client $i$ does not receive the global model $\boldsymbol{w}^{(t)}$ or because later the server does not receive client-$i$'s update $\Delta_i^{(t)}$. If these events are independent, and $q_{si}$ and $q_{is}$ denote the downlink and uplink loss probabilities, respectively, then $q_i = 1 - (1 - q_{si})(1 - q_{is})$. If ARQ or FEC techniques are employed, then $q_i$ can be interpreted as the residual loss probability experienced by the $i$-th client after potential retransmissions and/or error corrections, therefore our analysis remains agnostic to these methods.

Assumption 14 provides the flexibility for different loss probabilities across clients in the uplink and downlink transmissions, but does not consider spatial or temporal correlations, such as those arising from inter-channel interference or fading effects. We believe that results for Markov Chain gradient descent methods (where random samples are taken on the trajectory of a Markov chain)

could be used to study the convergence of UPGA-PL under correlated channels (Sun et al., 2018; Rodio, Faticanti, et al., 2023a). However, we defer this analysis to future work.

Convergence results for FL algorithms require to bound statistical heterogeneity in terms of some metric (e.g., X. Li et al. (2020); J. Wang et al. (2020); M. Chen et al. (2021)). We adopt the same metric introduced in X. Li et al. (2020):

**Definition 4.3.1.** Let $F^*$ and $F_i^*$ be the minimum values of $F$ and $F_i$, respectively. The parameter $\Gamma \triangleq F^* - \sum_{i=1}^N \alpha_i F_i^*$ quantifies the degree of data heterogeneity.

If the local datasets are identical, then the functions $\{F_i\}_{i=1}^N$ coincide and $\Gamma = 0$. In general, $\Gamma$ is larger the more heterogeneous the local data distributions are.

Theorem 4.3.1 (proof in Appendix A) establishes UPGA-PL convergence under lossy channels. It builds upon X. Li et al. (2020), which considers the ideal lossless scenario. Our primary technical contribution is captured in Lemma A.1 (Appendix A), with the additional term in (4.7) that accounts for the lossy channels.

**Theorem 4.3.1** (Convergence under lossy channels)**.** *Let Assumptions 10–14 hold and $L$, $\mu$, $\sigma_i$, $G$, $q_i$ defined therein. Choose diminishing learning rates as $\eta_c^{(t+1)} = \frac{2/\mu}{8\kappa+t}$, with $\kappa \triangleq L/\mu$. Then, for each $t \in \mathcal{T}$, UPGA-PL satisfies:*

$$\mathbb{E}\left[F(\boldsymbol{w}_{\textit{UPGA-PL}}^{(t+1)})\right] - F^* \leq \underbrace{\frac{\kappa}{8\kappa+t}\left(\frac{2KC}{\mu} + 4L\left\|\boldsymbol{w}^{(1)} - \boldsymbol{w}^*\right\|^2\right)}_{\textit{asymptotically vanishing term}}, \tag{4.6}$$

*where:*

$$C = \sum_{i=1}^N \alpha_i^2 \sigma_i^2 + 2(K-1)^2 G^2 + 6L\Gamma + \underbrace{KG^2 \sum_{i=1}^N \alpha_i^2 \frac{q_i}{1-q_i}}_{\textit{effect of lossy channels}}. \tag{4.7}$$

### 4.3.2 Discussion

**UPGA-PL enables convergence under lossy channels** Theorem 4.3.1 proves that the objective $F(\boldsymbol{w})$, evaluated on the sequence of models $\{\boldsymbol{w}^{(t)}\}_{t>0}$ computed by UPGA-PL, converges in expectation to its minimum value $F^*$. Moreover, as the function $F$ is strongly convex and then has a unique minimizer, the trained model converges also to the optimal one, i.e., $\lim_{t\to\infty} \mathbb{E}[\boldsymbol{w}_{\text{UPGA-PL}}^{(t)}] = \boldsymbol{w}^*$, where $\boldsymbol{w}^* \in \mathbb{R}^n \in \arg\min_{\boldsymbol{w}} F(\boldsymbol{w})$. The UPGA-PL aggregation strategy (with decreasing learning rates) does not suffer then from residual convergence errors as DMA-PL and UDMA-PL do.

**The effect of packet losses on the convergence** The constant $C$ (see (4.7)) quantifies the impact of lossy channels on the convergence in terms of the clients' loss probabilities $\{q_i\}_{i=1}^N$. As expected, the larger the loss probabilities, the larger is $C$ and the slower the convergence predicted by the bound in (4.6). Moreover, the convergence rate in Theorem 4.3.1 ($\mathcal{O}(1/t)$) is comparable to the convergence rate of FedAvg in absence of losses under the same assumptions ($\mathcal{O}(1/(Kt))$) (X. Li et al., 2020).

**Convergence speed vs. residual error** The bound in (4.3) suffers from a non-zero residual error but may achieve linear convergence ($A^t$ decreases exponentially fast); our bound in (4.6) removes such error at the cost of a sublinear convergence rate, i.e., of slower convergence. One may then think that for a short duration of the training period, `DMA-PL` is preferable to `UPGA-PL`. In reality, the bound (4.3) achieves such a rate requiring the use of full gradients at each client (i.e., $\sigma_i^2 = 0$) and a single local gradient update at each communication round (i.e., $K = 1$) (M. Chen et al., 2021); however, these assumptions do not correspond to FL practice (Kairouz et al., 2021).

## 4.4 Experimental Evaluation

### 4.4.1 Experimental setup

In the experiments, we consider a population with $N = 10$ clients. We split the population into two groups $G_i, i = 1, 2$, to which we associate different packet loss probabilities $q_i, i = 1, 2$. After evaluating different loss configurations, we present a challenging setting with $q_1 = 0.1$ and $q_2 = 0.9$.

We perform experiments on two datasets: the LEAF Synthetic Dataset for multinomial classification (Caldas et al., 2019) and the real-world MNIST dataset for handwritten digit recognition (L. Deng, 2012). To introduce statistical heterogeneity in the clients' datasets, we distribute the data among clients in a non-IID fashion. The LEAF Synthetic Dataset allows direct control of statistical heterogeneity through the parameters $\gamma$ and $\delta$: in our experiments, we set $\gamma = \delta = 1$. For MNIST, we generate a non-IID data distribution by splitting the labels among clients, with each client containing samples from only two classes (Achituve, Shamsian, Navon, Chechik, & Fetaya, 2021). We define Problem (4.1) with $\alpha_i = |D_i|/|D|, \forall i \in \mathcal{N}$.

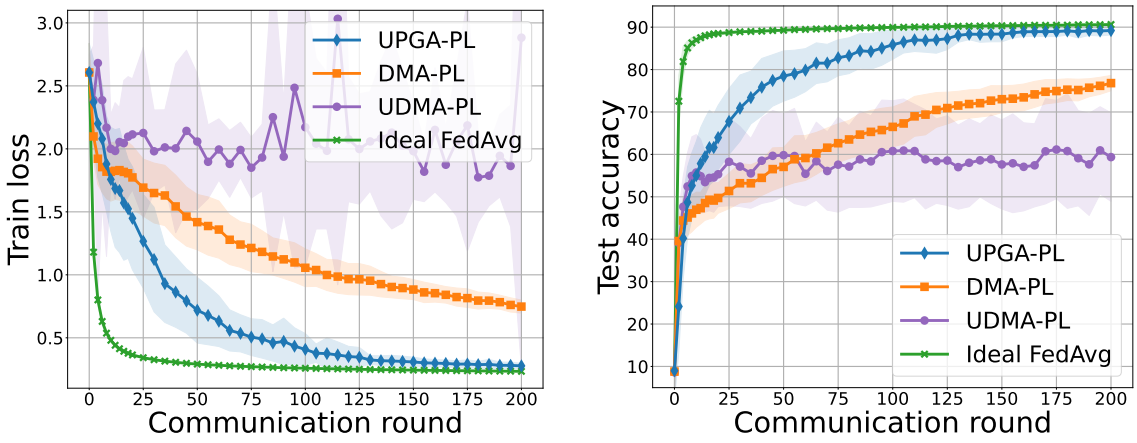
For the Synthetic LEAF dataset, we train a linear classifier with a ridge penalization of parameter $5 \times 10^{-4}$, which defines a strongly convex objective function that well aligns with our theoretical assumptions. As for MNIST, we use a CNN architecture with two convolutional and two fully connected layers, resulting in a non-convex objective function that introduces additional complexity to the learning process.

We compare `UPGA-PL`, in Algorithm 5, with `DMA-PL` (aggregation strategy in (4.2)), `UDMA-PL` (aggregation strategy in (4.4)), and an ideal lossless `FedAvg` (when $q_i = 0 \ \forall i \in \mathcal{N}$). In the experiments, `UDMA-PL` and `UPGA-PL` rely on the knowledge of $\{q_i\}_{i=1}^N$. For all algorithms, we tuned the learning rate $\eta_c = \{\eta_c^{(t)}\}_{t>0}$ via grid search with values $\eta = \{10^{-3}, 10^{-2.5}, 10^{-2}, 10^{-1.5}, 10^{-1}\}$. The reported results are averaged over 10 random seeds.

### 4.4.2 Experimental results

Figures 4.1–4.2 compare the train loss and test accuracy of `DMA-PL`, `UDMA-PL`, and `UPGA-PL` on the Synthetic LEAF and MNIST datasets. For both datasets, we include the reference performance of the ideal lossless `FedAvg`.

**`UPGA-PL` outperforms the baselines and converges to the optimal model** The experimental results unanimously confirm the advantages of the `UPGA-PL` aggregation strategy in terms of train loss and test accuracy on the two datasets. Indeed, `UPGA-PL` improves the state-of-the-art




(a) Train loss

(b) Test accuracy

Figure 4.1: Train loss/test accuracy on the Synthetic LEAF dataset.

solutions by 12 percentage points on the Synthetic LEAF dataset (Fig. 4.1b) and by 6 percentage points on the MNIST dataset (Fig. 4.2b). Moreover, `UPGA-PL` nearly attains the same performance as the `FedAvg` algorithm in lossless scenarios after around 150 communication rounds for both the Synthetic LEAF dataset (Fig. 4.1a) and the MNIST dataset (Fig. 4.2a).

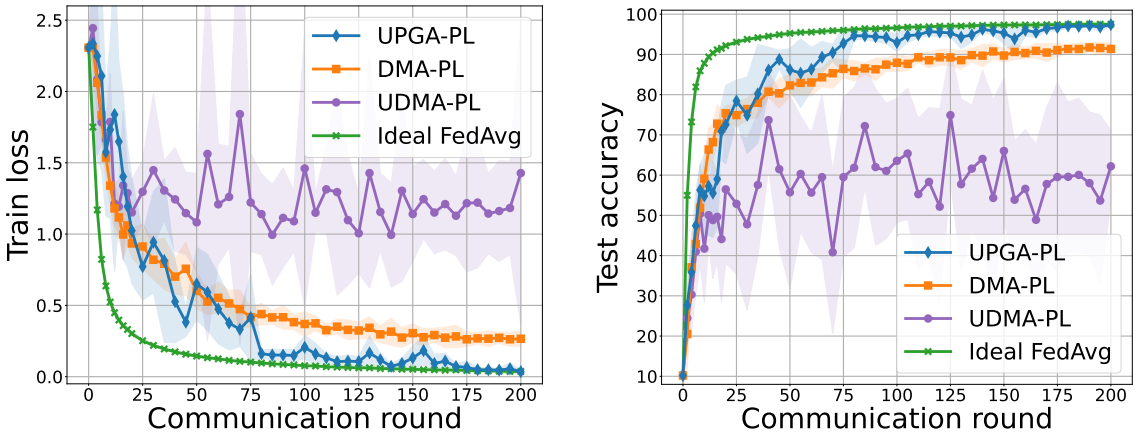
In line with our theoretical analysis, the numerical experiments also confirm the effects of lossy channels on the convergence captured by Theorem 4.3.1 and discussed in Section 4.3.2.

**The effects of packet losses on the convergence** FL algorithms perform best under the ideal scenario with lossless channels ($q_i = 0, \forall i \in \mathcal{N}$). Nevertheless, our experiments show that a severe amount of packet losses ($q_1 = 0.1, q_2 = 0.9$) slows down but does not prevent convergence to the optimal model, provided that `UPGA-PL` is used (`UPGA-PL` curve overlaps with `FedAvg` curve in the absence of losses).

**Residual errors** `DMA-PL` and `UDMA-PL` suffer from non-vanishing errors. The residual error of `DMA-PL` is evident in Figure 4.2a, where its loss curve reaches a plateau around the value 0.3, while `UPGA-PL` converges to the value 0.0, as the ideal `FedAvg`. On the other hand, the `UDMA-PL` aggregation strategy, which should, in theory, unbias the `DMA-PL` scheme, does not filter the variance introduced by the lossy channels and suffers a noisy convergence: the `UDMA-PL` performance dramatically oscillates in the experiments, and its mean accuracy lies around 50–70%.

## 4.5 Conclusion

This chapter addressed the problem of training FL algorithms over real-world wireless networks with lossy channels. We considered the presence of independent and identically distributed packet losses in the communication channels between the orchestrating server and the clients and we showed that the quality of the learned model is highly sensitive to the choice of the aggregation strategy. To mitigate the negative effects of packet losses, we proposed `UPGA-PL`, an algorithm




(a) Train loss

(b) Test accuracy

Figure 4.2: Train loss/test accuracy on the MNIST dataset.

that aggregates pseudo-gradients rather than models and that effectively converges to the optimal model under asymmetric lossy channels. While its complexity is comparable to `FedAvg`, under severe lossy settings `UPGA-PL` significantly outperformed the state-of-the-art solutions (M. Chen et al., 2021; Su et al., 2023) and attained very close performance to the optimal scenario with ideal, lossless channels at the cost of a slower convergence. Our work enabled optimal FL training under lossy channels, and—we believe—opened interesting research questions. For example, if losses affect only a part of the transmitted model, would it be possible for the clients or the server to take advantage of the partial information received instead of ignoring it (as `DMA-PL`, `UDMA-PL`, and `UPGA-PL` do)? What happens if the losses are correlated (e.g., due to inter-channel interference and/or fading)? What if they change over time?

## Remark

Subsequent to the publication of this research, we identified related work by Salehi and Hossain (2021), which also analyzes the convergence of FL algorithms under lossy communication channels. Although they propose a resource allocation strategy to control packet losses—specifically, they develop a scheduling policy for the allocation of radio blocks to each client, aligning with previous literature (Wen et al., 2019; M. Chen et al., 2021; Su et al., 2023), yet distinct from our `UPGA-PL` algorithm—their analysis already demonstrated convergence of FL algorithms under lossy communications challenges. We acknowledge the oversight of this seminal contribution and extend our sincerest apologies to the authors for this omission.

# CHAPTER 5

# Cooperative Inference Systems: The Case of Early Exit Networks

This chapter is based on our work Kaplan, Rodio, Salem, Xu, and Neglia (2024), currently under peer review and available as a preprint at https://arxiv.org/abs/2405.04249.

## 5.1 Motivation

After addressing client participation heterogeneity in Chapters 2 and 3, and heterogeneity in network resources in Chapter 4, we devote this chapter to hardware heterogeneity. Network nodes or clients (e.g., end-devices, edge servers, cloud infrastructures) often exhibit varying memory and computational capacities (Ren et al., 2023; Campolo, Iera, & Molinaro, 2023), as discussed in Section 1.3.2.3. This source of heterogeneity makes it infeasible to deploy a uniform ML model across all network nodes (Lim et al., 2020; Kairouz et al., 2021).

To overcome this problem, Cooperative Inference Systems (CISs) have been developed. These systems enable less capable devices to offload parts of their inference tasks to more powerful devices within the network, thereby leveraging their larger ML models to enhance overall performance (Salem et al., 2023; Ren et al., 2023).

Most research on CISs assumes the availability of such models and primarily focuses on either optimizing their placement within a network and/or identifying a beneficial cooperative serving policy (E. Li et al., 2019; Zeng et al., 2019; Salem et al., 2023; Ren et al., 2023). Significantly less research has focused on the training methodologies for deploying heterogeneous models within a CIS. This chapter specifically explores scenarios in which models are collaboratively trained using distributed datasets, hosted on the very clients that subsequently perform inference tasks.

Federated learning algorithms (McMahan et al., 2017; Kairouz et al., 2021; T. Li, Sahu, Zaheer, et al., 2020) typically involve the training of a single, common ML model, but can be extended to simultaneously train models of different sizes. Knowledge distillation (Lin et al., 2020; Mora, Tenison, Bellavista, & Rish, 2022) can enable knowledge transfer among heterogeneous models—but requires a public dataset. Alternative approaches involve the simultaneous training of models that share a subset of parameters.

Within the latter approach, prevalent methods include the joint training of Deep Neural Networks (DNNs) that either share entire layers or specific parameters within a layer. This can be achieved through techniques such as ordered dropout (Diao et al., 2020; Horvath et al., 2021) or early exits (Teerapittayanon, McDanel, & Kung, 2016, 2017). The final configuration of these shared parameters is influenced by the diverse and potentially conflicting requirements of the different models. For instance, parameters within a shared layer may learn to identify features of varying complexity depending on whether the layer is part of a deeper or shallower DNN.

Despite its importance, previous research has largely overlooked this unique challenge within CISs. Existing training methods, particularly those designed for distributed early exit networks (Teerapittayanon et al., 2017; Nawar et al., 2023; Ilhan et al., 2023), treat all models equally, failing to account for the heterogeneity in model capacities and performance. Only a few studies have empirically suggested assigning weights based on model complexity (Hu, Dey, Hebert, & Bagnell, 2019; Kaya, Hong, & Dumitras, 2019), but they still disregard the corresponding inference request rates.

To bridge this gap, this chapter introduces a first, theoretically grounded FL training algorithm specifically designed to improve the overall CIS inference performance.

**Our contributions.**

- We formalize the first inference-aware FL training framework for CISs, with the goal of maximizing overall inference accuracy. We reformulate the FL problem (Eq. 1.2) taking into account the inference requests for each sub-model inside the CIS. During training, we allocate larger weights to the models expected to handle a larger volume of inference requests.
- We present `Fed-CIS`, a novel and practical Inference-Aware FL algorithm designed for CISs. Two tuning parameters introduce flexibility in our algorithm: the weight assigned to each exit and the rates for the computationally stronger clients to assist the weaker ones during training.
- We analyze the impact of our tuning parameters on the convergence of `Fed-CIS` in terms of generalization error, optimization error, and bias error, providing a deeper understanding of how these parameters affect the overall training process and final inference performance. From this theoretical analysis, we derive practical configuration guidelines for our proposed training algorithm.
- We evaluate the effectiveness of our algorithm, `Fed-CIS`, showing that it significantly outperforms state-of-the-art methods, particularly in realistic scenarios where end devices handle higher inference request rates.

**Outline of chapter.** This chapter is organized as follows. Section 5.2 provides relevant background. Section 5.3 formalizes the model training problem in a CIS and introduces our novel FL algorithm. We present the theoretical guarantees of our approach in Section 5.3.3. Section 5.4 evaluates our algorithm against state-of-the-art (SOTA) training methods for early exit networks in a CIS setting. Finally, Section 5.5 concludes the chapter.

## 5.2 Background and Related Work

In this section, we discuss the relevant background necessary to understand Cooperative Inference Systems, Federated Learning, and Early Exit Networks.

### 5.2.1 Cooperative Inference Systems

Collaborative Inference Systems (CISs) (Ren et al., 2023), also known in the literature as Inference Delivery Networks (Salem et al., 2023), enable smaller devices to offload part of their inference tasks to more capable devices, and represent an active field of study. The scope of collaboration in these systems may vary, extending beyond the traditional device-cloud model, to include intermediate nodes such as edge servers, regional clouds, or a collective of devices within direct transmission range of each other (Teerapittayanon et al., 2017; E. Li et al., 2019; Zeng et al., 2019; Ren et al., 2023; Salem et al., 2023).

The form of collaboration within a CIS can also vary widely, from split computing frameworks, where a DNN is divided and processed across multiple nodes (Matsubara, Levorato, & Restuccia, 2022), to ensemble approaches that leverage multiple models working together for improved inference accuracy (Malka, Farhan, Morgenstern, & Shlezinger, 2022; Yilmaz, Hasırcıoğlu, & Gündüz, 2022). In this chapter, we consider a hierarchical structure of nodes, each equipped with increasingly complex models that collaborate by selectively forwarding inference requests to more powerful nodes within the network (Salem et al., 2023).

While much previous work has concentrated on optimizing the deployment and utilization of already trained model in a CIS (E. Li et al., 2019; Zeng et al., 2019; Salem et al., 2023), our research shifts focus to the less-explored challenge of *training* these models in a Federated Learning context. Here, each CIS node uses its local dataset to contribute to the training of the set of models intended for deployment across the network.

### 5.2.2 Federated Learning for a CIS

Traditional FL algorithms (e.g., FedAvg (McMahan et al., 2017), FedProx (T. Li, Sahu, Zaheer, et al., 2020)) typically assume that the participating nodes have similar storage and computation capacities, meaning that each node holds a DNN of the same architecture and can perform an equal amount of computation during training.

However, recent algorithms have been developed to efficiently train multiple models of different sizes within a network, with the most practical approach being the joint training of models that share a subset of parameters. For instance, FjORD (Horvath et al., 2021) introduces a framework where a DNN is pruned by channels to generate nested submodels of different sizes that can fit into heterogeneous nodes, following a mechanism known as ordered dropout. A similar idea is explored in HeteroFL (Diao et al., 2020). Alternative approaches involve the use of early exit networks (Nawar et al., 2023) or a combination of these two methods (Ilhan et al., 2023). While our algorithm and analysis apply to both pruning (e.g., ordered dropout) and early exit strategies, we focus on early exit networks for clarity and concreteness. Early exit networks also offer the clearest example of collaborative inference, as weaker nodes forward intermediate representations to more powerful nodes, unlike in FjORD/HeteroFL, where the input is forwarded.

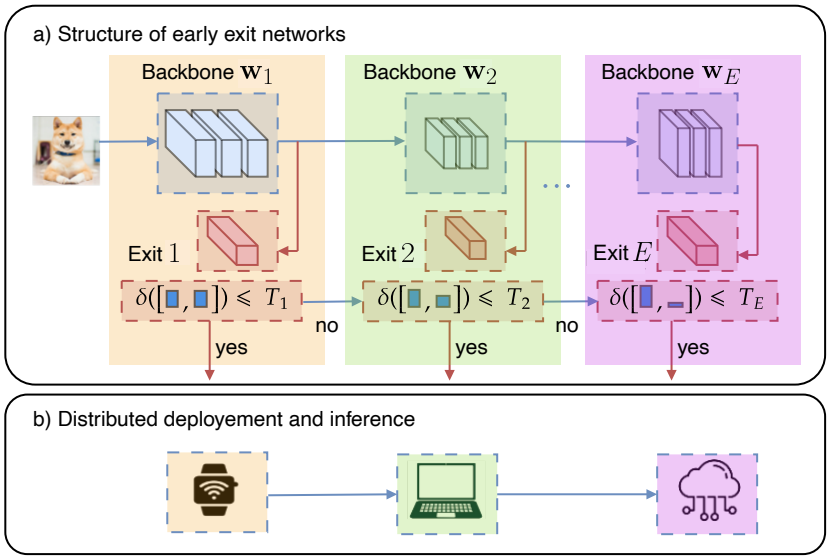



Figure 5.1: Early Exit Networks for Collaborative Inference System. An input sample is first passed through the initial layers of the DNN until it reaches Exit 1. If the measure of prediction uncertainty is below the threshold $T_1$, the prediction is served at the current node. Otherwise, the intermediate representation of the current input is transferred to a node with greater computational capacity, and inference continues. This process repeats until the prediction uncertainty is below $T_e$ or the final Exit $E$ is reached.

### 5.2.3 Early Exit Networks

Early Exit Networks (EENs), introduced initially as BranchyNets (Teerapittayanon et al., 2016), extend Deep Neural Networks (DNNs) by including additional classifiers (i.e., early exits) at intermediate layers (Teerapittayanon et al., 2016). For instance, implementing an early exit into a standard ResNet-34 architecture may involve adding a classifier after the 18th layer, enabling the original ResNet-34 to also serve as a ResNet-18 (He, Zhang, Ren, & Sun, 2016). The initial motivation for such a design is to allow for faster real-time inference with less computational cost, especially useful in computationally heavy computer vision and natural language processing tasks (Matsubara et al., 2022, Table 4). Fig. 5.1(a) details the inference process in standard EENs.

The typical training procedure involves minimizing the expected weighted loss across *all* exits (Teerapittayanon et al., 2016; Huang et al., 2018; H. Li, Zhang, Qi, Yang, & Huang, 2019; Hu et al., 2019; Kaya et al., 2019):

$$\min_{\boldsymbol{w}\in\mathbb{R}^d} \mathbb{E}_{z\sim\mathcal{D}}\left[\sum_{e\in\mathcal{E}} \alpha_e\, \ell^{(e)}(\boldsymbol{w}, z)\right], \tag{5.1}$$

where each data sample $z = (x, y)$ is drawn from the data distribution $\mathcal{D}$, with $x$ as the input features and $y$ the corresponding target, $\boldsymbol{w}$ represents the EEN parameters, $\mathcal{E} = \{1, \ldots, E\}$ is the set of early exits, $\ell^{(e)}$ the loss at the $e$-th exit, and $\alpha_e \in \mathbb{R}_{\geq 0}$ is the weight assigned to $e$-th exit's loss. In a CIS, this process extends to a distributed setting where each node holds a model with its assigned exit and all earlier exits. During inference, the intermediate representation can be sent to more powerful nodes, as shown in Fig. 5.1(b).

The weight coefficients $\alpha_e$ are crucial in determining each exit's ($e \in \mathcal{E}$) contribution to the overall model performance and can be assigned in various ways. Traditional training approaches generally assign equal weights to all exits (Teerapittayanon et al., 2017; Huang et al., 2018; Nawar et al., 2023; Ilhan et al., 2023). We refer to these methods collectively as the "Equal Weight" strategy. Alternatively, more complex approaches have been considered that allocate weights in proportion to each exit's computational complexity, often measured in FLOPS, which results in assigning more weight to later exits (Kaya et al., 2019; Hu et al., 2019, "Linear" baseline). We refer to these methods collectively as the "FLOPS Prop" strategy. These existing approaches overlook the fact that inference request rates can vary significantly across real-world networks, leading to accuracy drops in likely scenarios where end devices (e.g., smartphones) with shallow models handle most of the requests. Our proposed method addresses this issue by systematically incorporating these varying request rates into the training process.

## 5.3 Proposed Inference-Aware Algorithm: Fed-CIS

This section is organized as follows. In Section 5.3.1, we formalize the CIS and the training objective that takes into account inference requests. In Section 5.3.2, we present our algorithm for clients in a CIS to cooperatively train an EEN. In Section 5.3.3, we provide theoretical guarantees on the algorithm's convergence errors.

### 5.3.1 Problem Description

**Network Topology.** In a CIS, the network is composed of a set of nodes $\mathcal{N} = \{1, 2, \ldots, N\}$, organized in a hierarchical tree structure such as a cloud-edge-device model (Ren et al., 2023), where parent nodes possess greater computational resources and memory than their child nodes. While we consider a tree topology for presentation purposes, we stress that the proposed algorithm (Section 5.3.2) and its theoretical guarantees (Section 5.3.3) are broadly applicable to any directed acyclic graph, regardless of where the most powerful nodes are positioned within the network.

Leaf nodes, which have no children, are represented by the set $\mathcal{L} \subset \mathcal{N}$, and each node $i$ has a set of child nodes, denoted by $\mathcal{N}_i^-$. During *training*, each node $i$ holds multiple early exits, up to a maximum exit $E_i \leq E$, with the constraint that $E_i > E_j, \forall j \in \mathcal{N}_i^-$. However, during *inference*, node $i$ utilizes only its largest exit $E_i$ to ensure the most accurate prediction. The set $\mathcal{N}_e$ denotes nodes that use early exit $e$ for inference, i.e., $\mathcal{N}_e = \{i \in \mathcal{N} \mid E_i = e\}$.

**Real-time Inference Requests.** Local inference requests arrive at each node $i \in \mathcal{N}$ with an arrival rate $\lambda_i^l \in \mathbb{R}_{\geq 0}$. A child node $i$ can transfer inference requests to its parent node with a transfer rate $\lambda_i^t$. The total requests at node $i$ include both its local requests and those transferred from its children. Each node $i$ then serves a fraction $r_i \in [0, 1]$ of these requests locally using its largest exit $E_i$, resulting in a serving rate $\lambda_i^s$:

$$\lambda_i^s \triangleq \left(\lambda_i^l + \textstyle\sum_{j \in \mathcal{N}_i^-} \lambda_j^t\right) r_i, \tag{5.2}$$

while remaining requests are transferred to the parent node:

$$\lambda_i^t \triangleq \left(\lambda_i^l + \textstyle\sum_{j \in \mathcal{N}_i^-} \lambda_j^t\right) (1 - r_i). \tag{5.3}$$

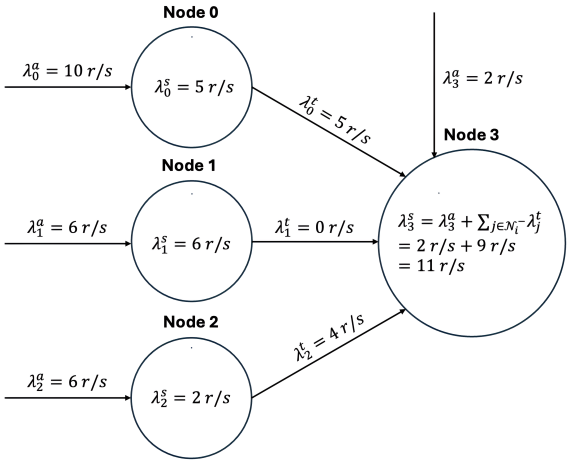



Figure 5.2: An example of a two-layer network with four nodes: Node 0, Node 1, and Node 2 each receive local requests, $\lambda_i^l$ (in requests per second, r/s), serve a portion locally, $\lambda_i^s$, and transfer the remainder, $\lambda_i^t$, to their parent. Node 3 receives requests both locally and from its children, and serves all requests as it has no parent.

Fig. 5.2 presents a straightforward numerical example illustrating how a CIS manages inference requests.

The transfer rate $\lambda_i^t$ is constrained by an upper limit $\mu_i^{\max}$, determined by the network's upstream bandwidth or the target inference delays. Each node is aware of its maximum transfer rate $\mu_i^{\max}$ and an estimate of its local arrival rate $\lambda_i^l$. Furthermore, nodes rank incoming samples by difficulty, allowing them to select the fraction $f_i$ of most favorable samples to serve locally (Teerapittayanon et al., 2016; Huang et al., 2018; Kaya et al., 2019). The data distribution of these served samples at node $i$ is $\mathcal{D}_i^s$.

**Training objective for CIS.** The primary goal of training in a CIS is to minimize the total loss across all served samples throughout the network, maximizing inference quality. We formalize this objective as the first *inference-aware* training framework for CISs using EENs, where the optimization problem is defined over the model parameters $\boldsymbol{w} \in \mathcal{W}$ and the serving fractions $\{r_i\}$ for each node:

$$\begin{aligned} \mathbb{P}_1: \quad & \min_{\boldsymbol{w}\in\mathcal{W},\{r_i\}} \sum_{i\in\mathcal{N}} \lambda_i^s \, \mathbb{E}_{z\sim\mathcal{D}_i^s} \left[ f^{(E_i)}(\boldsymbol{w}, z) \right], \\ \text{s.t.,} \quad & \lambda_i^t \le \mu_i^{\max}, \; f_i \in [0,1], \; \text{Eqs. 5.2 and 5.3}, \; \forall i \in \mathcal{N}. \end{aligned} \tag{5.4}$$

Building on existing research that shows deeper early exits typically yield higher inference accuracy (Teerapittayanon et al., 2016; Zeng et al., 2019; Baccarelli, Scardapane, Scarpiniti, Momenzadeh, & Uncini, 2020), we observe that smaller nodes should prioritize offloading requests to their parent nodes. We illustrate this point by contradiction: suppose in the optimal solution of $\mathbb{P}_1$, there exists a node with $r_i^* > 0$ and $\lambda_i^t < \mu_i^{\max}$. By decreasing $r_i^*$ to 0 or to a value that makes $\lambda_i^t = \mu_i^{\max}$ and adjusting $r_j^*$ s.t. parent node $j$ serves these additional requests locally, we achieve another feasible solution to $\mathbb{P}_1$ with a smaller loss. This allows us to simplify the optimization

problem $\mathbb{P}_1$ by restricting the search space to strategies that prioritize offloading, resulting in an equivalent optimization problem, $\mathbb{P}_2$, which focuses on minimizing losses at early exits:

$$\mathbb{P}_2: \qquad \min_{\boldsymbol{w}\in\mathcal{W}} \sum_{e\in\mathcal{E}} \Lambda_e \, \mathbb{E}_{z\sim\hat{\mathcal{D}}_e}\left[f^{(e)}(\boldsymbol{w}, z)\right], \tag{5.5}$$

where $\Lambda_e \triangleq \sum_{i\in\mathcal{N}_e}\left(\lambda_i^l + \sum_{j\in\mathcal{N}_i^-}\lambda_j^t - \lambda_i^t\right)$ is the total serving rate of all nodes using exit $e$ at inference time, and the data distribution of serving samples at early exit $e$ is $\hat{\mathcal{D}}_e$.*

In $\mathbb{P}_2$, the serving rates $\Lambda_e$ are constant, depending only on the arrival rates $\lambda_i^l$ and the maximum transfer rates $\mu_i^{\max}$. Before training begins, the cloud can collect this information from all nodes to compute the serving rates $\Lambda_e$.

### 5.3.2 The Fed-CIS Algorithm

We propose a FL algorithm that enables network nodes to collaboratively train an EEN for Problem $\mathbb{P}_2$ using their local datasets. Each node $i$ is designed to hold all exits up to its largest exit $E_i$. Although node $i$ only uses exit $E_i$ for inference, it can still play a crucial role in training smaller exits, particularly when it owns a substantial amount of data.

At each communication round, the server follows a two-step sampling process: first, it samples a set of nodes to participate in training, as in traditional FL algorithms; then, it selects a specific early exit for each chosen node to train. The probability that a node $i$ is selected to train a particular early exit $e$ is denoted by $p_{i,e}$, while $\boldsymbol{p} \in \mathbb{R}^{N\times E}$ represents the overall probability matrix. The set of nodes with a non-zero probability of training exit $e$ is $\mathcal{C}_e \triangleq \{i \in \mathcal{N} \mid p_{i,e} > 0\}$, and the set of all samples from nodes in $\mathcal{C}_e$ is $D_{e,\boldsymbol{p}} \triangleq \cup_{i\in\mathcal{C}_e} D_i$, where node $i$ holds samples $D_i$.

Our FL algorithm aims to minimize a proxy of the objective in $\mathbb{P}_2$, where the empirical loss at each early exit $e$ is replaced by the empirical loss computed on the dataset $D_{e,\boldsymbol{p}}$. Rather than strictly matching the weight $\tilde{\Lambda}_e$ to the expected inference request rate $\Lambda_e$, we adopt a more flexible training strategy that allows them to differ. This choice is supported by our theoretical results in Section 5.3.3. However, even without the analysis, it is evident that when exit $e$ has a high inference request rate $\Lambda_e$ but limited data $|D_{e,\boldsymbol{p}}|$, the empirical loss may be too noisy, making it preferable to set $\tilde{\Lambda}_e \ll \Lambda_e$.

Our algorithm works as follows: At each communication round $t$, the server samples nodes and their corresponding early exits based on the probability matrix $\boldsymbol{p}$ (Lines 3-4). The server then broadcasts the current global model to the sampled nodes (Line 5). Each node $i$ performs multiple steps of mini-batch gradient descent on the loss associated with its sampled early exit $e$, and returns the updated model to the server (Lines 6-10). TThe server aggregates these updates by computing a weighted sum of the pseudo-gradients from each node-exit pair $(i, e)$ (Line 11). Each pair's weight is determined by three key factors: (i) the importance $\tilde{\Lambda}_e$ assigned to exit $e$; (ii) the proportion of the dataset that node $i$ used to train relative to the total dataset used to train exit $e$ ($\frac{|S_i|}{|D_{e,\boldsymbol{p}}|}$); and (iii) the inverse of the probability that node $i$ was selected to train exit $e$ ($\frac{1}{p_{i,e}}$). The full procedure is summarized in Algorithm 6.

*In $\mathbb{P}_2$, the transfer rates are given by the recurrence relation $\lambda_i^t = \min\left\{\mu_i^{\max}, \lambda_i^l + \sum_{j\in\mathcal{N}_i^-}\lambda_j^t\right\}$.

**Algorithm 6:** `Fed-CIS` (Federated Learning for Cooperative Inference Systems)

**Input:** a randomized initial model $\boldsymbol{w}^{(1)}$, total communication rounds $T$, local steps $K$, global learning rate $\eta_s$, local learning rates $\{\eta_c^{(t,k)}\}$ at round $t$ and local step $k$, sampling matrix $\boldsymbol{p}$, aggregation weights $\tilde{\boldsymbol{\Lambda}}$.

**Output:** $\boldsymbol{w}^{(T)}$.

**for** $t = 1, \dots, T$ **do**

  Server samples the set $\mathcal{N}^{(t)}$ of node/exit pairs w.r.t. $\boldsymbol{p}$.

  Server broadcasts the model $\boldsymbol{w}^{(t)}$ to all nodes in $\mathcal{N}^{(t)}$.

  **for** $(i, e) \in \mathcal{N}^{(t)}$ *in parallel* **do**

    $\boldsymbol{w}_{i,e}^{(t,0)} \leftarrow \boldsymbol{w}^{(t)}$

    **for** $k = 0, \dots, K-1$ **do**

      Node $i$ selects a random batch $\mathcal{B}_i^{(t,k)}$

$$\boldsymbol{w}_{i,e}^{(t,k+1)} \leftarrow \boldsymbol{w}_{i,e}^{(t,k)} - \eta_c^{(t,k)} \frac{1}{|\mathcal{B}_i^{(t,k)}|} \sum_{z \in \mathcal{B}_i^{(t,k)}} \nabla f^{(e)}(\boldsymbol{w}_{i,e}^{(t,k)}, z)$$

    Node $i$ sends $\boldsymbol{w}_{i,e}^{(t,K)}$ to the server

  The server updates its global model:

$$\boldsymbol{w}^{(t+1)} \leftarrow \mathrm{Proj}\left( \boldsymbol{w}^{(t)} + \eta_s \sum_{(i,e) \in \mathcal{N}^{(t)}} \tilde{\Lambda}_e \frac{|D_i|}{|D_{e,\boldsymbol{p}}|} \frac{1}{p_{i,e}} (\boldsymbol{w}_{i,e}^{(t,K)} - \boldsymbol{w}^{(t)}) \right)$$

### 5.3.3 Generalization-Bias-Optimization Error Decomposition

Our analytical results assume that the serving distribution of every exit $e$ is the same, i.e., $\hat{\mathcal{D}}_e = \mathcal{D}, \forall e$. Let $\boldsymbol{w}^{(T)}$ be the output of our Algorithm 6, $F_{\mathcal{D},\boldsymbol{\Lambda}}(\boldsymbol{w}^{(T)})$ be the corresponding expected loss in Eq. (5.5) that we aim to minimize, and $F^\star_{\mathcal{D},\boldsymbol{\Lambda}}$ be its minimum value. In this section, we provide an upper-bound for the difference between $F_{\mathcal{D},\boldsymbol{\Lambda}}(\boldsymbol{w}^{(T)})$ and $F^\star_{\mathcal{D},\boldsymbol{\Lambda}}$.

More precisely, we investigate the *true error* of the algorithm:

$$\epsilon_{\text{true}} \triangleq \mathbb{E}_{D, A_{\tilde{\boldsymbol{\Lambda}}}} \left[ F_{\mathcal{D},\boldsymbol{\Lambda}}\left(\boldsymbol{w}^{(T)}\right) \right] - F^\star_{\mathcal{D},\boldsymbol{\Lambda}}, \tag{5.6}$$

where $A_{\tilde{\boldsymbol{\Lambda}}}$ is our algorithm and $D$ is the union of the nodes' datasets drawn from $\mathcal{D}$.

We first list the assumptions needed for our results, denoting node $i$'s empirical loss on early exit $e$ of model $\boldsymbol{w}$ as $F_{i,e}(\boldsymbol{w})$, i.e., $F_{i,e}(\boldsymbol{w}) \triangleq \frac{1}{|D_i|} \sum_{z \in D_i} \ell^{(e)}(\boldsymbol{w}, z)$. We can see from our aggregation rule that Algorithm 6 is minimizing $F_{D,\tilde{\boldsymbol{\Lambda}}}(\boldsymbol{w}) \triangleq \sum_{e \in E} \tilde{\Lambda}_e \sum_{i \in \mathcal{C}_e} \frac{|D_i|}{\sum_{i \in \mathcal{C}_e} |D_i|} F_{i,e}(\boldsymbol{w})$. Let $\boldsymbol{w}^\star_{i,e}$, $\boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}$, and $\boldsymbol{w}^\star_{\mathcal{D}}$ be the minimizers of $F_{i,e}$, $F_{D,\tilde{\boldsymbol{\Lambda}}}$, and $F_{\mathcal{D},\boldsymbol{\Lambda}}$, respectively.

**Assumption 15.** *The loss function is bounded:* $\forall \boldsymbol{w} \in \mathcal{W}$ *and* $z \in \mathcal{Z}$, $f(\boldsymbol{w}, z) \in [0, M]$.

**Assumption 16.** *The hypothesis space* $\mathcal{W} \subset \mathbb{R}^d$ *is convex and compact with diameter* $\mathrm{diam}(\mathcal{W})$*, and contains the minimizers* $\boldsymbol{w}^\star_{i,e}$*,* $\boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}$ *and* $\boldsymbol{w}^\star_{\mathcal{D}}$ *in its interior.*

**Assumption 17.** $\{F_{i,e}\}_{(i,e)\in\mathcal{N}\times\mathcal{E}}$ *are* $L$*-smooth: for all* $\boldsymbol{v}$ *and* $\boldsymbol{w}$ *in* $\mathcal{W}$*,* $\|\nabla F_{i,e}(\boldsymbol{v}) - \nabla F_{i,e}(\boldsymbol{w})\|_2 \leq L\|\boldsymbol{v}-\boldsymbol{w}\|_2$.

**Assumption 18.** $\{F_{i,e}\}_{(i,e)\in\mathcal{N}\times\mathcal{E}}$ *are* $\mu$*-strongly convex: for all* $\boldsymbol{v}$ *and* $\boldsymbol{w}$ *in* $\mathcal{W}$*,* $F_{i,e}(\boldsymbol{v}) \geq F_{i,e}(\boldsymbol{w}) + \langle \nabla F_{i,e}(\boldsymbol{w}), \boldsymbol{v}-\boldsymbol{w}\rangle + \frac{\mu}{2}\|\boldsymbol{v}-\boldsymbol{w}\|_2^2$.

**Assumption 19.** *Let* $\mathcal{B}_i$ *be a random batch sampled from the* $i$*-th node's local data uniformly at random. The variance of stochastic gradients in each node is bounded:* $\mathbb{E}\|\nabla F_{i,e}(\boldsymbol{w},\mathcal{B}_i) - \nabla F_{i,e}(\boldsymbol{w})\|^2 \leq \sigma_{i,e}^2$ *for all* $\boldsymbol{w}$ *in* $\mathcal{W}$ *and* $(i,e)\in\mathcal{N}\times\mathcal{E}$.

Assumption 15 is standard in statistical learning theory (e.g., Mohri, Rostamizadeh, and Talwalkar (2018); Shalev-Shwartz and Ben-David (2014)), while Assumptions 16–19 are standard in the analysis of federated optimization algorithms (e.g., J. Wang et al. (2021); X. Li et al. (2020); Rodio, Faticanti, et al. (2023b)). We observe that Assumptions 16, 17, and 19 jointly imply that the stochastic gradients are bounded. We denote this bound by $G$, i.e., $\mathbb{E}\|\nabla F_{i,e}(\boldsymbol{w},\mathcal{B}_i)\|^2 \leq G^2$ for $\boldsymbol{w}\in\mathcal{W}$ and $(i,e)\in\mathcal{N}\times\mathcal{E}$.

Theorem 5.3.1 provides an upper bound on the true error of our algorithm in terms of the sum of three components: a generalization error, a bias error (due to the mismatch between $F_{\mathcal{D},\tilde{\boldsymbol{\Lambda}}}$ and $F_{\mathcal{D},\boldsymbol{\Lambda}}$), and an optimization error. A detailed proof is available in the Appendix D.

**Theorem 5.3.1.** *Under Assumptions 15–19, the true error of the output* $\boldsymbol{w}^{(T)}$ *of Algorithm 6 with learning rate* $\eta_c^{(t,k)} = \frac{2}{\mu(\gamma+(t-1)K+j+1)}$ *and* $\gamma \triangleq \max\{8\kappa, K\} - 1$ *can be bounded as follows:*

$$\epsilon_{true} \leq \underbrace{\mathcal{O}\left(\sum_{e=1}^{E}\tilde{\Lambda}_e\sqrt{\frac{\mathrm{Pdim}(H_e)}{|D_{e,\boldsymbol{p}}|}}\right)}_{\epsilon_{gen}} + \underbrace{\mathcal{O}\left(\mathrm{dist}_{\mathrm{TV}}(\tilde{\boldsymbol{\Lambda}},\boldsymbol{\Lambda})\right)}_{\epsilon_{bias}} + \underbrace{\mathcal{O}\left(\frac{B(\tilde{\boldsymbol{\Lambda}},\boldsymbol{p},\boldsymbol{\sigma},\{|D_i|\}_{(i,e)\in\mathcal{N}\times\mathcal{E}})}{KT}\right)}_{\epsilon_{opt}}, \tag{5.7}$$

*where* $\kappa \triangleq \frac{L}{\mu}$, $Pdim(H_e)$ *represents the pseudo-dimension of the class of models for exit* $e$, $\mathrm{dist}_{\mathrm{TV}}$ *is the total variation distance,* $\boldsymbol{\Lambda} = (\Lambda_1,\ldots,\Lambda_E)$, $\tilde{\boldsymbol{\Lambda}} = (\tilde{\Lambda}_1,\ldots,\tilde{\Lambda}_E)$, *and the expression of* $B(\cdot)$ *is provided in Theorem D.4.*

### 5.3.4 Configuration Rules

Theorem 5.3.1 shows that the choice of aggregation weights $\tilde{\boldsymbol{\Lambda}}$ in Algorithm 6 (Line 11) affects all three error components: generalization error, optimization error, and bias error—each minimized by a different choice of $\tilde{\boldsymbol{\Lambda}}$.

The bias error $\epsilon_{\text{bias}}$ is dominant when each exit $e$ is trained on a large dataset $D_{e,\boldsymbol{p}}$ (making $\epsilon_{\text{gen}}$ small) and the number of communication rounds $T$ is high (making $\epsilon_{\text{opt}}$ small). In such settings, the optimal strategy sets the aggregation weights $\tilde{\boldsymbol{\Lambda}}$ equal to the expected serving rates $\boldsymbol{\Lambda}$. We refer to this configuration rule as "Serving Rate", which effectively eliminates the bias error, as $\mathrm{dist}_{\mathrm{TV}}(\tilde{\boldsymbol{\Lambda}},\boldsymbol{\Lambda}) = 0$. However, optimization and generalization errors can also play a significant role. In these cases, deviating $\tilde{\boldsymbol{\Lambda}}$ from $\boldsymbol{\Lambda}$ may reduce these errors, though it introduces a non-zero bias error.

The optimization error $\epsilon_{\text{opt}}$ is strongly influenced by the gradient variance $\sigma_{i,e}^2$ at each early exit $e$, as shown by the $B(\cdot)$ term, whose complete expression can be found in Theorem D.4. Empirical

evidence shows that gradient variance is significantly higher at the initial exits compared to the later ones, making the optimization error especially sensitive to the stochastic gradients produced at these early stages.[†] To reduce $\epsilon_{\text{opt}}$, earlier exits with higher variance should be assigned lower aggregation weights $\tilde{\Lambda}_e < \Lambda_e$ to lessen their impact during training. In scenarios where the optimization error dominates, it follows that minimizing $\epsilon_{\text{opt}}$ involves setting the aggregation weights $\tilde{\Lambda}_e$ inversely proportional to the gradient variance $\sigma^2_{i,e}$. We observe that this approach alters the weights in the same direction as the "FLOPS Prop" strategy (described in Section 5.2.3), which also assigns larger weights to more powerful models.

The generalization error, on the other hand, is affected by the ratio $\text{Pdim}(H_e)/|D_{e,\boldsymbol{p}}|$. In practice, $\text{Pdim}(H_e)$ acts as a proxy for the complexity of the model at exit $e$. It follows that exits with a larger model (i.e., larger $\text{Pdim}(H_e)$), and smaller dataset (i.e., smaller $|D_{e,\boldsymbol{p}}|$), contribute more to this error component, and thus reducing the aggregation weights associated to these exits minimizes $\epsilon_{\text{gen}}$. In extreme cases where the generalization error is dominant, the optimal strategy requires setting aggregation weights to zero for all exits except the one with the lowest complexity ratio $\text{Pdim}(H_e)/|D_{e,\boldsymbol{p}}|$. The probabilities $p_{i,e}$ can also play a role in further reducing the generalization error, whereby powerful nodes periodically train exit $e$, practically increasing the sample size $D_{e,\boldsymbol{p}}$ and leading to a reduced $\epsilon_{\text{gen}}$.

In many realistic scenarios, it is likely that no single error component is dominant, and one might consider configuring our FL algorithm by minimizing the entire bound in Theorem 5.3.1. However, this approach is often impractical due to the complexities involved in estimating theoretical parameters, such as the Lipschitz constant $L$ and the strong convexity constant $\mu$. To address this issue, our experimental findings suggest that a hybrid strategy, which balances the reduction of both bias and optimization errors, offers robust performance across many settings. For the remainder of this paper, we refer to this heuristic approach as "Balanced Adj", where the abbreviation "Adj" stands for Adjustment.

## 5.4 Experimental Evaluation

In this section, we present experimental results that validate our theoretical analysis in Section 5.3.3 and highlight the versatility of our algorithm across various CIS serving rate settings.

### 5.4.1 Training Details

We conduct experiments on the CIFAR10 and CIFAR100 datasets, employing the ResNet-18 model architecture (He et al., 2016). Both datasets and model are widely used to benchmark FL algorithms in the presence of device heterogeneity and EENs (H. Li et al., 2019; Hu et al., 2019; Kaya et al., 2019; Diao et al., 2020; Horvath et al., 2021; Ilhan et al., 2023). We insert early exits after the 2nd and 5th residual blocks for CIFAR10 and after the 5th and 7th residual blocks for CIFAR100. For reproducibility, all dataset details, training infrastructure, and hyperparameters are provided in Appendix F.

[†]Experimental evidence is provided in Appendix E.

### 5.4.2 Evaluation Methodology

**Baselines.** Our work represents the first attempt to develop a FL training algorithm for use within a CIS. Due to the lack of established baselines for direct comparison, we compare our approach to SOTA algorithms proposed to train traditional EENs, focusing on those that have a straightforward application to FL and CIS settings (see Section 5.2.3 for a comprehensive description of these methods). The two strategies in this category are: (i) "Equal Weight," which assigns equal weight to all early exits (Teerapittayanon et al., 2017; Huang et al., 2018), and (ii) "FLOPS Prop," which weights the exits according to their FLOPS (Kaya et al., 2019). While other centralized training methods, such as those proposed by Hu et al. (2019); H. Li et al. (2019), could potentially be adapted for our purposes, their extension is less straightforward and would require extra computation by the nodes. We also implement the (iii) "Serving Rate" and (iv) "Balanced Adj" strategies, both directly derived from our analysis in Section 5.3.3. The code for our experimental framework is publicly available.

**CIS Topology.** We utilize a hierarchical network topology as defined in Section 5.3.1 and considered in related works (Teerapittayanon et al., 2017; Ren et al., 2023) with seven nodes: four in the first layer, two in the second, and one in the third, each holding an increasing portion of the shared model according to their network layer.[‡]

In Section 5.4.3, we present results for two data partition settings: (a) "equal data partition," where data is evenly distributed across all network layers, and (b) "biased data partition," where data is heavily concentrated on the most powerful devices.

**Serving Rates.** We assume that all inference requests initially arrive at the leaf nodes ($\lambda_i^l = 0, \forall i \in \mathcal{N} \setminus \mathcal{L}$). During inference, each node $i$ assesses the confidence score of the incoming requests, serving the simplest ones based on its serving rate $\lambda_i^s$ and forwarding the remaining, more complex requests according to its transfer rate $\lambda_i^t$. We evaluate a wide range of serving rates $\mathbf{\Lambda}$, including scenarios where (i) the least powerful nodes serve most of the requests; (ii) request rates are evenly distributed across all layers; and (iii) the most powerful nodes serve most of the requests. To denote these serving rates, we use the notation x-y-z, where x, y, and z represent the percentage of inference requests served by nodes using Exits 1, 2, and 3, respectively.

### 5.4.3 Experimental Results

Table 5.1 presents our results on the CIFAR10 and CIFAR100 datasets under the "equal data partition" setting. On CIFAR10, our "Serving Rate" and "Balanced Adj" strategies consistently outperform the "Equal Weight" and "FLOPS Prop" methods across all CIS serving rate configurations, especially in scenarios where the smallest models handle most of the inference requests, such as in the 80-15-5, 60-30-10, and 45-35-20 settings. In these cases, both "Equal Weight" and "FLOPS Prop" perform poorly, as they fail to account for the actual distribution of serving rates. Specifically, in the 80-15-5 setting, "Serving Rate" outperforms "Equal Weight" by 4.3 percentage points (p.p.) and "FLOPS Prop" by 22.1 p.p., while in the 5-15-80 setting, "Balanced Adj" surpasses them by 2.5 p.p. and 1.3 p.p., respectively.

[‡]We conducted additional experiments with a larger network consisting of 17 total nodes, which confirmed the consistency of our results. Due to computational constraints, this larger network was not used for all experiments. Detailed results are in Appendix G, Table D.2.

Table 5.1: Experimental results for a variety of CIS serving rates on the CIFAR10 and CIFAR100 datasets using an "equal data partition" across the network layers. All reported accuracy values are the mean value over three independent random seeds.

| | | CIS Serving Rate Setting | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Dataset | Strategy | 80-15-5 | 60-30-10 | 45-35-20 | 33-33-33 | 20-35-45 | 10-30-60 | 5-15-80 |
| CIFAR10 | Equal Weight | 49.9 ± 1.2 | 60.6 ± 0.9 | 68.9 ± 0.6 | **74.9** ± 0.2 | 80.4 ± 0.2 | 83.8 ± 0.3 | 85.1 ± 0.4 |
| | FLOPS Prop | 32.1 ± 3.7 | 47.3 ± 2.9 | 58.5 ± 2.3 | 67.3 ± 1.7 | 76.4 ± 1.1 | 83.2 ± 0.7 | 86.3 ± 0.6 |
| | Serving Rate (ours) | **54.2** ± 2.4 | **62.6** ± 1.5 | **69.2** ± 1.3 | **74.9** ± 0.2 | **80.9** ± 0.2 | 84.6 ± 0.5 | 86.7 ± 0.5 |
| | Balanced Adj (ours) | 53.4 ± 2.2 | 61.1 ± 1.2 | **69.2** ± 0.7 | **74.9** ± 0.3 | 80.4 ± 0.5 | **85.0** ± 0.2 | **87.6** ± 0.5 |
| CIFAR100 | Equal Weight | 39.8 ± 1.2 | 45.9 ± 0.9 | 51.0 ± 0.6 | **55.2** ± 0.3 | **58.3** ± 0.2 | 60.3 ± 0.2 | 61.1 ± 0.3 |
| | FLOPS Prop | 30.4 ± 0.7 | 40.0 ± 0.6 | 48.3 ± 0.4 | 53.3 ± 0.0 | 58.0 ± 0.1 | **60.9** ± 0.1 | **62.1** ± 0.1 |
| | Serving Rate (ours) | 45.0 ± 0.7 | **50.2** ± 0.7 | **53.0** ± 0.8 | **55.2** ± 0.3 | 53.2 ± 1.0 | 56.2 ± 0.1 | 57.9 ± 0.5 |
| | Balanced Adj (ours) | **46.6** ± 1.2 | 49.5 ± 0.8 | 52.1 ± 0.6 | 54.8 ± 0.3 | 57.4 ± 0.2 | 59.3 ± 0.8 | 60.7 ± 0.2 |

To better understand these results, we analyze how different training strategies affect $\tilde{\mathbf{\Lambda}}$ and, in turn, the CIS test accuracy. First, setting $\tilde{\mathbf{\Lambda}}$ equal to the serving rate $\mathbf{\Lambda}$ minimizes the bias error $\epsilon_{\text{bias}}$, which is the objective of our "Serving Rate" strategy. On the CIFAR10 task, with $T = 100$ communication rounds and a sufficiently large dataset, $\epsilon_{\text{bias}}$ dominates, allowing "Serving Rate" to empirically minimize this term and perform well across various serving rate settings. In contrast, "Equal Weight" assigns equal weights to all exits, which can significantly increase $\epsilon_{\text{bias}}$ as serving rates become more uneven, likely leading to poor performance in scenarios with extreme serving rate imbalances.

On the CIFAR100 dataset, we observe performance trends similar to CIFAR10 across various device-biased CIS serving rate settings, including 80-15-5, 60-30-10, 45-35-20, and 33-33-33. However, when the largest models handle most of the requests, the "FLOPS Prop" baseline outperforms our "Serving Rate" strategy, likely due to the greater difficulty of the CIFAR100 task, which results in a larger optimization error. As noted in Section 5.3.4, strategies like "FLOPS Prop" are expected to perform well in these scenarios, though its performance drops significantly when the inference load shifts to the first-layer nodes. This shift increases the bias error because $\tilde{\mathbf{\Lambda}}$ diverges from $\mathbf{\Lambda}$, causing a significant drop in CIS accuracy. This is evident in the 80-15-5 configuration, where "Serving Rate" and "Balanced Adj" outperform "FLOPS Prop" by 14.6 and 16.2 p.p., respectively.

In Tables 5.2 and 5.3, we present results from experiments on CIFAR10 and CIFAR100 using the alternative "biased data partition" scheme, where nodes with greater memory and computational capacity are allocated more data. On CIFAR10, "Serving Rate" and "Balanced Adj" again consistently outperform other baselines across all CIS serving rate settings, while on CIFAR100, "Balanced Adj" remains strong in all scenarios, especially when the first-layer nodes handle most inference requests.

Combining these findings with those from the "equal data partition" experiments, our results show that the "Balanced Adj" strategy either leads or closely matches the performance of the best methods across various CIS configurations. Overall, these experiments reinforce the core insights from Section 5.3.4, highlighting the critical role of error decomposition in selecting aggregation weights $\tilde{\mathbf{\Lambda}}$. In particular, configuring $\tilde{\mathbf{\Lambda}}$ to minimize bias error $\epsilon_{\text{bias}}$ proves often beneficial. Additionally,

Table 5.2: Experimental results for a variety of CIS serving rates on the CIFAR10 and CIFAR100 datasets using the "biased data partition", where the networks layers hold 14.3%, 28.6%, and 57.1% of the data, respectively. All reported accuracy values are the mean value over three independent random seeds.

| | | CIS Serving Rate Setting | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Dataset | Strategy | 80-15-5 | 60-30-10 | 45-35-20 | 33-33-33 | 20-35-45 | 10-30-60 | 5-15-80 |
| CIFAR10 | Equal Weight | 47.4 ± 3.6 | 58.9 ± 3.5 | 67.8 ± 2.8 | **74.9** ± 2.2 | 80.9 ± 1.4 | 85.0 ± 0.7 | 86.7 ± 0.2 |
| | FLOPS Prop | 31.5 ± 3.2 | 47.2 ± 2.5 | 59.0 ± 1.8 | 68.4 ± 1.4 | 78.1 ± 0.8 | 85.4 ± 0.4 | 88.8 ± 0.4 |
| | Serving Rate (ours) | **53.1** ± 2.3 | **60.6** ± 0.7 | 66.4 ± 1.1 | **74.9** ± 2.2 | 81.7 ± 1.7 | 87.1 ± 0.7 | 89.2 ± 0.6 |
| | Balanced Adj (ours) | 51.0 ± 3.2 | 59.6 ± 4.2 | **68.0** ± 4.1 | 74.6 ± 2.2 | **82.1** ± 1.9 | **87.5** ± 0.7 | **90.3** ± 0.6 |
| CIFAR100 | Equal Weight | 37.3 ± 0.9 | 44.7 ± 0.6 | 50.7 ± 0.6 | **55.6** ± 0.3 | 59.8 ± 0.1 | 62.5 ± 0.1 | 63.6 ± 0.3 |
| | FLOPS Prop | 31.4 ± 0.6 | 40.0 ± 0.4 | 47.7 ± 0.4 | 54.1 ± 0.1 | **60.3** ± 0.1 | **64.3** ± 0.2 | **66.1** ± 0.3 |
| | Serving Rate (ours) | 38.1 ± 0.7 | 45.8 ± 0.4 | **51.5** ± 0.4 | **55.6** ± 0.3 | 55.5 ± 0.7 | 61.4 ± 0.8 | 64.9 ± 0.4 |
| | Balanced Adj (ours) | **42.8** ± 0.9 | **46.7** ± 0.3 | 51.1 ± 0.4 | 53.3 ± 1.1 | 56.9 ± 1.3 | 61.7 ± 0.8 | 64.3 ± 0.8 |

Table 5.3: Full experimental results for scenarios where nodes with the smallest models and datasets serve the majority of inference requests the CIFAR10 and CIFAR100 datasets. In this "highly biased data partition", networks layers hold 3.4%, 19.9%, and 76.7% of the data, respectively. All reported accuracy values are the mean value over three independent random seeds.

| | | CIS Serving Rate Setting | | |
|---|---|---|---|---|
| Dataset | Strategy | 80-15-5 | 60-30-10 | 45-35-20 |
| CIFAR10 | Equal Weight | 36.5 ± 4.0 | 45.5 ± 3.2 | 56.6 ± 3.0 |
| | Serving Rate (ours) | 41.3 ± 2.7 | 40.1 ± 2.2 | 55.3 ± 2.5 |
| | Serving Rate $p = 0.2$ (ours) | **49.2** ± 2.9 | **52.6** ± 3.1 | **56.8** ± 2.2 |
| CIFAR100 | Equal Weight | 10.0 ± 1.4 | 15.8 ± 3.4 | 24.6 ± 2.9 |
| | Serving Rate (ours) | 17.9 ± 1.0 | 20.6 ± 2.1 | 27.7 ± 3.9 |
| | Serving Rate $p = 0.2$ (ours) | **30.1** ± 2.1 | **30.8** ± 1.2 | **30.9** ± 0.9 |

incorporating adjustments to address the optimization error $\epsilon_{\text{opt}}$—as done by "Balanced Adj"—helps ensure that the resulting FL algorithm is robust across a wide range of serving rates, including the 80-15-5 and 5-15-80 settings.

**Enabling Node Collaboration through Probabilities $p$.** We conducted an ablation study to examine the impact of the hyperparameter $\boldsymbol{p}$, focusing on extreme scenarios where nodes with the smallest models and datasets serve the majority of inference requests. This scenario is especially relevant, as end devices typically have limited data storage compared to cloud servers or other more powerful nodes. Results for the most challenging configuration, the 80-15-5 serving rate setting, are presented in Table 5.4 for both the CIFAR10 and CIFAR100 datasets, where $p_{i,e} = p$ if $e < E_i$ and $p_{i,E_i} = 1-(E_i-1)p$. These experiments clearly show that increasing $\boldsymbol{p}$ significantly improves the overall inference accuracy by enabling stronger nodes to support weaker ones during training. Additional results on the impact of $\boldsymbol{p}$ can be found in Appendix G, Figure D.1.

Table 5.4: Results for the 80-15-5 serving rate setting using the highly biased data partition, where the network layers hold 3.4%, 19.9%, and 76.7% of the data, respectively.

| Dataset | Strategy | $p = 0$ | $p = 0.2$ |
|---|---|---|---|
| CIFAR10 | Equal Weight | 36.5 $\pm$ 4.0 | - |
| | Serving Rate (ours) | 41.3 $\pm$ 2.6 | **49.2** $\pm$ 2.9 |
| CIFAR100 | Equal Weight | 10.0 $\pm$ 1.4 | - |
| | Serving Rate (ours) | 17.9 $\pm$ 0.2 | **30.1** $\pm$ 2.1 |

## 5.5 Conclusion

We are the first to design an inference-aware FL training algorithm for CISs, demonstrating that inference serving rates influence all components of training error. When using our inference-aware configuration rules, which consider the error decomposition into the training process, our algorithm provides a significant advantage, particularly when inference request rates are unevenly distributed across the network. Moreover, our rigorous theoretical results are applicable to all approaches that jointly train models sharing a subset of parameters, including early exit networks, ordered dropout, pruning, and other nested training methodologies.

# CHAPTER 6

# Conclusion and Perspectives

In this manuscript, we performed a comprehensive examination of the several sources of client heterogeneity in federated learning, and we developed novel algorithms to mitigate their negative effects on these systems.

**Client participation heterogeneity.** Chapter 2 addressed the problem of heterogeneous and correlated client availability in federated learning, resulting from network and resource limitations as well as client eligibility requirements. Additionally, Chapter 3 studied the large variability introduced by such heterogeneity in the final model and presented a variance reduction strategy, proven effective in leveraging stale model updates for non-participating clients.

**Network resources heterogeneity.** Chapter 4 focused on tackling heterogeneity caused by clients' diverse communication channels—where each client experiences unique channel characteristics and packet losses—and presented a packet loss-aware FL algorithm.

**Hardware environments heterogeneity.** Finally, Chapter 5 was devoted to the training of heterogeneous-sized FL algorithms, suited for the clients diverse processing capacities, while taking into account their request rates and deployment at inference time.

In this chapter, we present a summary of the main contributions of this manuscript in Section 6.1, followed by an overview of prospective future research directions in Section 6.2. The manuscript closes with concluding remarks in Section 6.3.

## 6.1 Take-Home Lessons from the Main Contributions

### Federated Learning under Heterogeneous and Correlated Client Availability

Chapter 2 presented the first convergence analysis of a federated learning algorithm under heterogeneous and correlated client availability. It revealed the detrimental effect of correlation on convergence rate and highlighted a trade-off between convergence speed and model bias. It presented the first correlation-aware FL algorithm, `CA-Fed`, which adaptively handles the conflicting aims of boosting convergence speed and lowering model bias, by selectively excluding the less available and more correlated clients from training. Experimental results verify the efficiency of `CA-Fed`, making it suited for real-world applications with correlated client availability. Our key take-away from this chapter is that client sampling may be used to decrease the negative impact

of correlation in client availability. We identify the need for future research on designing such sampling strategies.

### Leveraging Stale Model Updates for Non-Participating Clients

Chapter 3 explored global variance reduction in federated learning beyond the typical assumption of homogeneous client participation. Our research underlined, unlike prior work, not only the advantages but also the disadvantages of exploiting stale client updates in different heterogeneity conditions. We hope that discussing this trade-off will promote the design of federated learning algorithms more responsive to the varied dynamics of client data and participation heterogeneity. We equipped our staleness-aware FL algorithm, `FedStale`, with guidelines: practitioners can decide if rely solely on participating client updates or whether it is worthwhile to store stale updates. However, a systematic approach to take this decision is currently lacking and it is subject for future research.

### Federated Learning in Lossy Communication Channels

Chapter 4 investigated the training of FL algorithms in real-world wireless networks with lossy communication channels. It presented a packet loss-aware FL algorithm, `UPGA-PL`, that achieves similar performance under asymmetric lossy channels to ideal, lossless scenarios. `UPGA-PL` outperforms state-of-the-art solutions and offers interesting research directions. This chapter's first conclusion is that the aggregation strategy considerably affects the quality of the learned model: for asymmetric lossy channels, pseudo-gradients aggregation significantly outperforms direct model aggregation. The second, interesting insight is that, if losses only impact a portion of the transmitted model, it is possible to extend our results to prove that training can still benefit from the partial information received. This study can also be extended to correlated losses resulting from, for instance, fading or inter-channel interference, similarly to what done in Chapter 2.

### Cooperative Inference Systems: The Case of Early Exit Networks

Chapter 5 presented the first inference-aware FL training algorithm for CISs, `Fed-CIS`, and demonstrated that the inference serving rates must be considered at training time if one wants to improve the overall performance of these systems. Our results apply to all hierarchical training systems, including early-exit networks, ordered dropout, pruning, and other nested training methods. By accounting for the expected serving rates at inference time, `Fed-CIS` provides significant gains, especially when end-devices serve the majority of requests. More complex algorithm design may be needed in some other settings, for example when devices store most of the training data and the cloud serves the majority of requests. These aspects deserve investigation in future research.

## 6.2 Perspectives and Future Research Directions

Although this manuscript presented novel algorithms to alleviate the negative impact of client heterogeneity in federated learning, it is crucial to acknowledge that some challenges still remain open. This section presents an overview of these problems, clarifies their complexities, and discusses possible solutions.

### Semi-Decentralized Topologies for Cross-Device FL

As an alternative to the standard client-server architecture, cross-device FL can leverage semi-decentralized topologies to address specific challenges of client partial and heterogeneous participation (Section 1.1.3). These topologies allow clients with limited connectivity capabilities to exchange model updates locally in a gossip-SGD fashion, decreasing the number of server communications (J. Wang, Sahu, Yang, Joshi, & Kar, 2019; Costantini, Neglia, & Spyropoulos, 2023). From the analysis perspective, the inter-client communications within these semi-decentralized frameworks produce time-varying graphs where a subset of nodes, representing the participating clients, only connects at certain rounds. To the best of our knowledge, existing decentralized SGD analyses require strong connectivity of the graph or symmetric adjacency matrices—assumptions not met when client participation is intermittent (Nedić & Olshevsky, 2015; Nedić, Olshevsky, & Shi, 2017; Assran, Loizou, Ballas, & Rabbat, 2019; Koloskova et al., 2020); nonetheless, FL analyses prove convergence with arbitrary small subsets of participating clients (X. Li et al., 2020; J. Wang et al., 2020, 2021). Future research will focus on bridging the gap between decentralized and federated learning analyses.

### Wireless Networks: Number of Retransmissions vs. Training Time

Previous research on wireless networks with lossy communication channels has focused on two extreme strategies for handling packet losses: comprehensive error correction and retransmissions until correct model receipt (M. Chen et al., 2021; Wen et al., 2019; Su et al., 2023), or loss-tolerant algorithms that enable convergence without any retransmission (Salehi & Hossain, 2021; Rodio, Neglia, et al., 2023). To the best of our knowledge, there is a significant gap in understanding the effect of retransmissions on the convergence of FL algorithms. It is expected that an optimal balance exists between the number of retransmissions and the overall training duration. Future research will focus on quantifying this trade-off and estimating the ideal number of retransmissions needed to minimize the overall training time. These estimation must account the influence of diverse channel characteristics, such as fading, interference, and path loss.

### Energy-Driven Correlated Client Participation in Cross-Silo FL

Another open research topic addresses algorithmic approaches to minimize energy consumption in cross-silo FL systems. Organizations in data centers exhibit heterogeneous participation patterns with seasonal and diurnal fluctuations, influenced by external factors beyond the server control, such as solar and wind activities (Eichner et al., 2019; Zhu et al., 2021; Cho et al., 2023). These variations, correlated across time and space, require dynamic and adaptive resource allocation strategies. We believe that consensus-based distributed optimization, which allows for variable communication topologies and partial participation of computing nodes, can play a role in lowering the energy footprint of cross-silo FL systems. This can be achieved by optimizing when and how nodes are activated based on real-time energy availability and training progress, and remains an open problem for future investigation.

### Incentivizing Client Participation in Federated Learning

One last research direction we discuss in this manuscript involves investigating federated learning in a setting where the clients can choose to join or leave the federation. Incentives may include

performance rewards for certain clients in the federation as well as financial support to encourage client participation. Game-theoretic stability analysis can provide a deeper understanding of this research area (Donahue & Kleinberg, 2021; Blum, Haghtalab, Phillips, & Shao, 2021; Tu et al., 2022). Moreover, economic incentives enable the system owner to actively control client participation (Kang et al., 2019; Cho, Jhunjhunwala, Li, Smith, & Joshi, 2022). Another appealing avenue for future work involves abandoning the idea of learning a shared, common ML model, and enabling clients to train personalized models more suitable to their local data distributions (Mansour et al., 2020; Y. Deng et al., 2020a; Grimberg, Hartley, Karimireddy, & Jaggi, 2021).

## 6.3 Concluding Reflections

In concluding this manuscript, the author acknowledges the modest contribution of this thesis in the vast research landscape. Future researchers will hopefully build upon this thesis, reflecting the collaborative nature of academic progress.

# References


Achituve, I., Shamsian, A., Navon, A., Chechik, G., & Fetaya, E. (2021). Personalized Federated Learning With Gaussian Processes. In *Advances in Neural Information Processing Systems.*

Assran, M., Loizou, N., Ballas, N., & Rabbat, M. (2019). Stochastic Gradient Push for Distributed Deep Learning. In *Proceedings of the 36th International Conference on Machine Learning* (pp. 344–353). PMLR.

Athalye, A., Carlini, N., & Wagner, D. (2018, July). Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples. In *Proceedings of the 35th International Conference on Machine Learning* (pp. 274–283). PMLR.

Baccarelli, E., Scardapane, S., Scarpiniti, M., Momenzadeh, A., & Uncini, A. (2020). Optimized training and scalable implementation of conditional deep neural networks with early exits for fog-supported IoT applications. *Information Sciences*, *521*, 107–143. doi: 10.1016/j.ins.2020.02.041

Baccarelli, E., Scarpiniti, M., Momenzadeh, A., & Sarv Ahrabi, S. (2022). AFAFed—Asynchronous Fair Adaptive Federated learning for IoT stream applications. *Computer Communications*, *195*, 376–402. doi: 10.1016/j.comcom.2022.09.016

Baruch, G., Baruch, M., & Goldberg, Y. (2019). A Little Is Enough: Circumventing Defenses For Distributed Learning. In *Advances in Neural Information Processing Systems* (Vol. 32). Curran Associates, Inc.

Benko, P., & Veres, A. (2002). A Passive Method for Estimating End-to-End TCP Packet Loss. In *IEEE Global Telecommunications Conference, 2002. GLOBECOM '02* (Vol. 3, p. 2609-2613 vol.3). doi: 10.1109/GLOCOM.2002.1189102

Beznosikov, A., Horváth, S., Richtárik, P., & Safaryan, M. (2023). On Biased Compression for Distributed Learning. *Journal of Machine Learning Research*, *24*(276), 1–50.

Blum, A., Haghtalab, N., Phillips, R. L., & Shao, H. (2021). One for One, or All for All: Equilibria and Optimality of Collaboration in Federated Learning. In *Proceedings of the 38th International Conference on Machine Learning* (pp. 1005–1014). PMLR.

Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., ... Roselander, J. (2019). Towards Federated Learning at Scale: System Design. *Proceedings of Machine Learning and Systems*, *1*, 374–388.

Borkar, V. S. (2009). *Stochastic Approximation: A Dynamical Systems Viewpoint* (Vol. 48). Springer. doi: 10.1007/978-93-86279-38-5

Bottou, L., Curtis, F. E., & Nocedal, J. (2018). Optimization Methods for Large-Scale Machine Learning. *SIAM Review*, *60*(2), 223–311. doi: 10.1137/16M1080173

Bousquet, O., Boucheron, S., & Lugosi, G. (2004-09-06, 2003). Introduction to statistical learning theory. In O. Bousquet, U. von Luxburg, & G. Rätsch (Eds.), *Advanced lectures on machine learning* (Vol. 3176, pp. 169–207). Springer.

Boyd, S., & Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press. doi: 10.1017/CBO9780511804441

Briggs, C., Fan, Z., & Andras, P. (2020). Federated learning with hierarchical clustering of local updates to improve training on non-IID data. In *2020 International Joint Conference on Neural Networks (IJCNN)* (pp. 1–9). doi: 10.1109/IJCNN48605.2020.9207469

Bubeck, S. (2015). Convex Optimization: Algorithms and Complexity. *Foundations and Trends® in Machine Learning*, *8*(3-4), 231–357. doi: 10.1561/2200000050

Caldas, S., Duddu, S. M. K., Wu, P., Li, T., Konečný, J., McMahan, H. B., … Talwalkar, A. (2019). LEAF: A Benchmark for Federated Settings. *arXiv:1812.01097 [cs, stat]*.

Campolo, C., Iera, A., & Molinaro, A. (2023). Network for distributed intelligence: A survey and future perspectives. *IEEE Access*, *11*, 52840–52861.

Chandrasekaran, R., Ergun, K., Lee, J., Nanjunda, D., Kang, J., & Rosing, T. (2022). FHDnn: Communication Efficient and Robust Federated Learning for AIoT Networks. In *Proceedings of the 59th ACM/IEEE Design Automation Conference* (pp. 37–42). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/3489517.3530394

Chen, M., Yang, Z., Saad, W., Yin, C., Poor, H. V., & Cui, S. (2021). A Joint Learning and Communications Framework for Federated Learning Over Wireless Networks. *IEEE Transactions on Wireless Communications*, *20*(1), 269–283. doi: 10.1109/TWC.2020.3024629

Chen, W., Horváth, S., & Richtárik, P. (2022). Optimal Client Sampling for Federated Learning. *Transactions on Machine Learning Research*.

Cho, Y. J., Jhunjhunwala, D., Li, T., Smith, V., & Joshi, G. (2022). To Federate or Not To Federate: Incentivizing Client Participation in Federated Learning. In *Workshop on Federated Learning: Recent Advances and New Challenges (in Conjunction with NeurIPS 2022)*.

Cho, Y. J., Sharma, P., Joshi, G., Xu, Z., Kale, S., & Zhang, T. (2023). On the Convergence of Federated Averaging with Cyclic Client Participation. In *Proceedings of the 40th International Conference on Machine Learning* (pp. 5677–5721). PMLR.

Cho, Y. J., Wang, J., & Joshi, G. (2020). Client Selection in Federated Learning: Convergence Analysis and Power-of-Choice Selection Strategies. *arXiv*(arXiv:2010.01243). doi: 10.48550/arXiv.2010.01243

Collins, L., Hassani, H., Mokhtari, A., & Shakkottai, S. (2021). Exploiting Shared Representations for Personalized Federated Learning. In *Proceedings of the 38th International Conference on Machine Learning* (pp. 2089–2099). PMLR.

Costantini, M., Neglia, G., & Spyropoulos, T. (2023). FedDec: Peer-to-peer Aided Federated Learning. *arXiv*(arXiv:2306.06715). doi: 10.48550/arXiv.2306.06715

Dayan, I., Roth, H. R., Zhong, A., Harouni, A., Gentili, A., Abidin, A. Z., ... Li, Q. (2021). Federated learning for predicting clinical outcomes in patients with COVID-19. *Nature Medicine*, *27*(10), 1735–1743. doi: 10.1038/s41591-021-01506-3

Defazio, A., Bach, F., & Lacoste-Julien, S. (2014). SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives. In *Advances in Neural Information Processing Systems* (Vol. 27). Curran Associates, Inc.

Deng, L. (2012). The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, *29*(6), 141–142.

Deng, Y., Kamani, M. M., & Mahdavi, M. (2020a). Adaptive Personalized Federated Learning. *arXiv*(arXiv:2003.13461). doi: 10.48550/arXiv.2003.13461

Deng, Y., Kamani, M. M., & Mahdavi, M. (2020b). Distributionally Robust Federated Averaging. In *Advances in Neural Information Processing Systems* (Vol. 33, pp. 15111–15122). Curran Associates, Inc.

Diao, E., Ding, J., & Tarokh, V. (2020). Heterofl: Computation and communication efficient federated learning for heterogeneous clients. *arXiv preprint arXiv:2010.01264*.

Ding, S., & Wang, W. (2022, December). Collaborative Learning by Detecting Collaboration Partners. *Advances in Neural Information Processing Systems*, *35*, 15629–15641.

Ding, Y., Niu, C., Yan, Y., Zheng, Z., Wu, F., Chen, G., ... Jia, R. (2020). Distributed Optimization over Block-Cyclic Data. *arXiv*. doi: 10.48550/arXiv.2002.07454

Doan, T. T. (2020). Local Stochastic Approximation: A Unified View of Federated Learning and Distributed Multi-Task Reinforcement Learning Algorithms. *arXiv*. doi: 10.48550/arXiv.2006.13460

Doan, T. T., Nguyen, L. M., Pham, N. H., & Romberg, J. (2020a). Convergence Rates of Accelerated Markov Gradient Descent with Applications in Reinforcement Learning. *arXiv*. doi: 10.48550/arXiv.2002.02873

Doan, T. T., Nguyen, L. M., Pham, N. H., & Romberg, J. (2020b). Finite-Time Analysis of Stochastic Gradient Descent under Markov Randomness. *arXiv*. doi: 10.48550/arXiv.2003.10973

Donahue, K., & Kleinberg, J. (2021). Model-sharing Games: Analyzing Federated Learning Under Voluntary Participation. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(6), 5303–5311. doi: 10.1609/aaai.v35i6.16669

du Terrail, J. O., Léopold, A., Joly, C., Beguier, C., Andreux, M., Maussion, C., ... Heudel, P.-E. (2021). Collaborative Federated Learning behind Hospitals' Firewalls for Predicting Histological Response to Neoadjuvant Chemotherapy in Triple-Negative Breast Cancer. *medRxiv*, 2021.10.27.21264834. doi: 10.1101/2021.10.27.21264834

Eichner, H., Koren, T., Mcmahan, B., Srebro, N., & Talwar, K. (2019). Semi-Cyclic Stochastic Gradient Descent. In *Proceedings of the 36th International Conference on Machine Learning* (pp. 1764–1773). PMLR.

Eriş, M. C., Kantarci, B., & Oktug, S. (2021). Unveiling the Wireless Network Limitations in Federated Learning. In *2021 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)* (pp. 262–267). doi: 10.1109/DySPAN53946.2021.9677285

*European Parliament legislative resolution of 13 March 2024 on the proposal for a regulation of the European Parliament and of the Council on laying down harmonised rules on Artificial Intelligence (Artificial Intelligence Act) and amending certain Union Legislative Acts.* (2016).

Even, M., Massoulié, L., & Scaman, K. (2022, December). On Sample Optimality in Personalized Collaborative and Federated Learning. *Advances in Neural Information Processing Systems*, *35*, 212–225.

Fang, C., Li, C. J., Lin, Z., & Zhang, T. (2018). SPIDER: Near-Optimal Non-Convex Optimization via Stochastic Path-Integrated Differential Estimator. In *Advances in Neural Information Processing Systems* (Vol. 31). Curran Associates, Inc.

Fercoq, O., Qu, Z., Richtárik, P., & Takáč, M. (2014). Fast distributed coordinate descent for non-strongly convex losses. In *2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)* (pp. 1–6). doi: 10.1109/MLSP.2014.6958862

Fraboni, Y., Vidal, R., Kameni, L., & Lorenzi, M. (2021). Clustered Sampling: Low-Variance and Improved Representativity for Clients Selection in Federated Learning. In *Proceedings of the 38th International Conference on Machine Learning* (pp. 3407–3416). PMLR.

Fraboni, Y., Vidal, R., Kameni, L., & Lorenzi, M. (2023). A General Theory for Federated Optimization with Asynchronous and Heterogeneous Clients Updates. *Journal of Machine Learning Research*, *24*(110), 1–43.

Goodfellow, I. J., Mirza, M., Xiao, D., Courville, A., & Bengio, Y. (2015). An Empirical Investigation of Catastrophic Forgetting in Gradient-Based Neural Networks. *arXiv*. doi: 10.48550/arXiv.1312.6211

Grimberg, F., Hartley, M.-A., Karimireddy, S. P., & Jaggi, M. (2021). Optimal Model Averaging: Towards Personalized Collaborative Learning. *arXiv*(arXiv:2110.12946). doi: 10.48550/arXiv.2110.12946

Gu, X., Huang, K., Zhang, J., & Huang, L. (2021). Fast Federated Learning in the Presence of Arbitrary Device Unavailability. In *Advances in Neural Information Processing Systems* (Vol. 34, pp. 12052–12064). Curran Associates, Inc.

Haddadpour, F., Kamani, M. M., Mokhtari, A., & Mahdavi, M. (2021). Federated Learning with Compression: Unified Analysis and Sharp Guarantees. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics* (pp. 2350–2358). PMLR.

Hard, A., Kiddon, C. M., Ramage, D., Beaufays, F., Eichner, H., Rao, K., ... Augenstein, S. (2018). *Federated learning for mobile keyboard prediction.* Retrieved from https://arxiv.org/abs/1811.03604

Hashimoto, T., Srivastava, M., Namkoong, H., & Liang, P. (2018). Fairness Without Demographics in Repeated Loss Minimization. In *Proceedings of the 35th International Conference on Machine Learning* (pp. 1929–1938). PMLR.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Hestness, J., Narang, S., Ardalani, N., Diamos, G., Jun, H., Kianinejad, H., ... Zhou, Y. (2017). Deep Learning Scaling is Predictable, Empirically. *arXiv*(arXiv:1712.00409). doi: 10.48550/arXiv.1712.00409

Horvath, S., Laskaridis, S., Almeida, M., Leontiadis, I., Venieris, S., & Lane, N. (2021). Fjord: Fair and accurate federated learning under heterogeneous targets with ordered dropout. *Advances in Neural Information Processing Systems*, *34*, 12876–12889.

Hu, H., Dey, D., Hebert, M., & Bagnell, J. A. (2019). Learning anytime predictions in neural networks via adaptive loss balancing. In *The thirty-third AAAI conference on artificial intelligence, AAAI 2019* (pp. 3812–3821). AAAI Press. doi: 10.1609/aaai.v33i01.33013812

Huang, G., Chen, D., Li, T., Wu, F., van der Maaten, L., & Weinberger, K. Q. (2018). Multi-scale dense networks for resource efficient image classification. In *6th international conference on learning representations, ICLR 2018, vancouver, BC, canada, april 30 - may 3, 2018, conference track proceedings*. OpenReview.net.

Ilhan, F., Su, G., & Liu, L. (2023). Scalefl: Resource-adaptive federated learning with heterogeneous clients. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 24532–24541).

Jee Cho, Y., Wang, J., & Joshi, G. (2022). Towards understanding biased client selection in federated learning. In *Proceedings of the 25th international conference on artificial intelligence and statistics* (pp. 10351–10375). PMLR.

Jhunjhunwala, D., Sharma, P., Nagarkatti, A., & Joshi, G. (2022). Fedvarp: Tackling the Variance due to Partial Client Participation in Federated Learning. In *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence* (pp. 906–916). PMLR.

Johnson, R., & Zhang, T. (2013). Accelerating Stochastic Gradient Descent using Predictive Variance Reduction. In *Advances in Neural Information Processing Systems* (Vol. 26). Curran Associates, Inc.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., ... Zhao, S. (2021). Advances and Open Problems in Federated Learning. *Foundations and Trends® in Machine Learning*, *14*(1–2), 1–210. doi: 10.1561/2200000083

Kang, J., Xiong, Z., Niyato, D., Yu, H., Liang, Y.-C., & Kim, D. I. (2019). Incentive Design for Efficient Federated Learning in Mobile Networks: A Contract Theory Approach. In *2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)* (pp. 1–5). doi: 10.1109/VTS-APWCS.2019.8851649

Kaplan, C., Rodio, A., Salem, T. S., Xu, C., & Neglia, G. (2024). Federated Learning for Cooperative Inference Systems: The Case of Early Exit Networks. *arXiv*(arXiv:2405.04249). doi: 10.48550/arXiv.2405.04249

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., & Suresh, A. T. (2020). SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. In *Proceedings of the 37th International Conference on Machine Learning* (pp. 5132–5143). PMLR.

Kaya, Y., Hong, S., & Dumitras, T. (2019). Shallow-deep networks: Understanding and mitigating network overthinking. In *International conference on machine learning* (pp. 3301–3310). PMLR.

Kearns, M., & Roth, A. (2019). *The ethical algorithm: The science of socially aware algorithm design*. Oxford University Press.

Kemker, R., McClure, M., Abitino, A., Hayes, T., & Kanan, C. (2018). Measuring Catastrophic Forgetting in Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1). doi: 10.1609/aaai.v32i1.11651

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., ... Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, *114*(13), 3521–3526. doi: 10.1073/pnas.1611835114

Koloskova, A., Loizou, N., Boreiri, S., Jaggi, M., & Stich, S. (2020). A Unified Theory of Decentralized SGD with Changing Topology and Local Updates. In *Proceedings of the 37th International Conference on Machine Learning* (pp. 5381–5393). PMLR.

Koloskova, A., Stich, S., & Jaggi, M. (2019). Decentralized Stochastic Optimization and Gossip Algorithms with Compressed Communication. In *Proceedings of the 36th International Conference on Machine Learning* (pp. 3478–3487). PMLR.

Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., & Bacon, D. (2017). Federated Learning: Strategies for Improving Communication Efficiency. *arXiv*(arXiv:1610.05492). doi: 10.48550/arXiv.1610.05492

Kovalev, D., Koloskova, A., Jaggi, M., Richtarik, P., & Stich, S. (2021). A Linearly Convergent Algorithm for Decentralized Optimization: Sending Less Bits for Free! In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics* (pp. 4087–4095). PMLR.

Kovalev, D., Salim, A., & Richtarik, P. (2020). Optimal and Practical Algorithms for Smooth and Strongly Convex Decentralized Optimization. In *Advances in Neural Information Processing Systems* (Vol. 33, pp. 18342–18352). Curran Associates, Inc.

Krizhevsky, A., & Hinton, G. (2009). *Learning multiple layers of features from tiny images* (Tech. Rep.). Toronto: Technical report, University of Toronto / University of Toronto.

Lei, L., Ju, C., Chen, J., & Jordan, M. I. (2017). Non-convex Finite-Sum Optimization Via SCSG Methods. In *Advances in Neural Information Processing Systems* (Vol. 30). Curran Associates, Inc.

Levin, D. A., & Peres, Y. (2017). *Markov Chains and Mixing Times: Second Edition* (Vol. 107). American Mathematical Soc.

Li, E., Zeng, L., Zhou, Z., & Chen, X. (2019). Edge AI: On-demand accelerating deep neural network inference via edge computing. *IEEE Transactions on Wireless Communications*, *19*(1), 447–457.

Li, H., Zhang, H., Qi, X., Yang, R., & Huang, G. (2019). Improved techniques for training adaptive deep networks. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 1891–1900).

Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated Learning: Challenges, Methods, and Future Directions. *IEEE Signal Processing Magazine*, *37*(3), 50–60. doi: 10.1109/MSP.2020.2975749

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., & Smith, V. (2020). Federated Optimization in Heterogeneous Networks. *Proceedings of Machine Learning and Systems*, *2*, 429–450.

Li, X., Huang, K., Yang, W., Wang, S., & Zhang, Z. (2020). On the Convergence of FedAvg on Non-IID Data. In *International Conference on Learning Representations.*

Lian, X., Zhang, W., Zhang, C., & Liu, J. (2018). Asynchronous Decentralized Parallel Stochastic Gradient Descent. In *Proceedings of the 35th International Conference on Machine Learning* (pp. 3043–3052). PMLR.

Lim, W. Y. B., Luong, N. C., Hoang, D. T., Jiao, Y., Liang, Y.-C., Yang, Q., ... Miao, C. (2020). Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, *22*(3), 2031–2063.

Lin, T., Kong, L., Stich, S. U., & Jaggi, M. (2020). Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, *33*, 2351–2363.

Liu, Y., Kang, Y., Xing, C., Chen, T., & Yang, Q. (2020). A Secure Federated Transfer Learning Framework. *IEEE Intelligent Systems*, *35*(4), 70–82. doi: 10.1109/MIS.2020.2988525

Livesay, M. (2017). *Chaining Method to Improve Rademacher Bound.* (Lecture Notes, https://therisingsea.org/notes/FoundationsForCategoryTheory.pdf)

Ludwig, H., & Baracaldo, N. (2022). *Federated Learning: A Comprehensive Overview of Methods and Applications*. Springer Cham. doi: 10.1007/978-3-030-96896-0

Malka, M., Farhan, E., Morgenstern, H., & Shlezinger, N. (2022). Decentralized low-latency collaborative inference via ensembles on the edge. *arXiv preprint arXiv:2206.03165*.

Mansour, Y., Mohri, M., Ro, J., & Suresh, A. T. (2020). Three Approaches for Personalization with Applications to Federated Learning. *arXiv*(arXiv:2002.10619). doi: 10.48550/arXiv.2002.10619

Marfoq, O., Neglia, G., Bellet, A., Kameni, L., & Vidal, R. (2021). Federated Multi-Task Learning under a Mixture of Distributions. In *Advances in Neural Information Processing Systems* (Vol. 34, pp. 15434–15447). Curran Associates, Inc.

Marfoq, O., Neglia, G., Kameni, L., & Vidal, R. (2023). Federated learning for data streams. In F. Ruiz, J. Dy, & J.-W. van de Meent (Eds.), *Proceedings of the 26th international conference on artificial intelligence and statistics* (Vol. 206, pp. 8889–8924). PMLR.

Marfoq, O., Neglia, G., Vidal, R., & Kameni, L. (2022). Personalized Federated Learning through Local Memorization. In *Proceedings of the 39th International Conference on Machine Learning* (pp. 15070–15092). PMLR.

Marfoq, O., Xu, Chuan., Neglia, G., & Vidal, R. (2020). Throughput-Optimal Topology Design for Cross-Silo Federated Learning. In *Advances in Neural Information Processing Systems* (Vol. 33, pp. 19478–19487). Curran Associates, Inc.

Matsubara, Y., Levorato, M., & Restuccia, F. (2022). Split computing and early exiting for deep learning applications: Survey and research challenges. *ACM Computing Surveys*, *55*(5), 1–30.

McCloskey, M., & Cohen, N. J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. In G. H. Bower (Ed.), *Psychology of learning and motivation* (Vol. 24, pp. 109–165). Academic Press.

McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* (pp. 1273–1282). PMLR.

Meyer, C. D. (2001). *Matrix analysis and applied linear algebra*. SIAM.

Meyers, A., & Yang, H. (2021). Markov chains for fault-tolerance modeling of stochastic networks. *IEEE Transactions on Automation Science and Engineering*.

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of machine learning* (2nd ed.). Cambridge, MA: MIT Press.

Mohri, M., Sivek, G., & Suresh, A. T. (2019). Agnostic Federated Learning. In *International conference on machine learning* (pp. 4615–4625). PMLR.

Mora, A., Tenison, I., Bellavista, P., & Rish, I. (2022). Knowledge distillation for federated learning: A practical guide. *arXiv preprint arXiv:2211.04742*.

Nasr, M., Shokri, R., & Houmansadr, A. (2019). Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning. In *2019 IEEE Symposium on Security and Privacy (SP)* (pp. 739–753). doi: 10.1109/SP.2019.00065

Nawar, M. N. A. M., Falavigna, D., & Brutti, A. (2023). Fed-EE: Federating heterogeneous ASR models using early-exit architectures. In *Proceedings of 3rd neurips workshop on efficient natural language and speech processing.*

Nedić, A., & Olshevsky, A. (2015). Distributed Optimization Over Time-Varying Directed Graphs. *IEEE Transactions on Automatic Control*, *60*(3), 601–615. doi: 10.1109/TAC.2014.2364096

Nedić, A., Olshevsky, A., & Shi, W. (2017). Achieving Geometric Convergence for Distributed Optimization Over Time-Varying Graphs. *SIAM Journal on Optimization*, *27*(4), 2597–2633. doi: 10.1137/16M1084316

Nesterov, Y. (2004). *Introductory Lectures on Convex Optimization* (Vol. 87; P. M. Pardalos & D. W. Hearn, Eds.). Boston, MA: Springer US. doi: 10.1007/978-1-4419-8853-9

Nguyen, L. M., Liu, J., Scheinberg, K., & Takáč, M. (2017). SARAH: A Novel Method for Machine Learning Problems Using Stochastic Recursive Gradient. In *Proceedings of the 34th International Conference on Machine Learning* (pp. 2613–2621). PMLR.

Nichol, A., Achiam, J., & Schulman, J. (2018). On First-Order Meta-Learning Algorithms. *arXiv*. doi: 10.48550/arXiv.1803.02999

Olle, H., Yuval, P., & Jeffrey, E. S. (1997). Dynamical Percolation. In *Annales de l'Institut henri poincare (B) probability and statistics* (Vol. 33, pp. 497–528). Elsevier.

Paulik, M., Seigel, M., Mason, H., Telaar, D., Kluivers, J., van Dalen, R., ... Hung, S. (2021). Federated Evaluation and Tuning for On-Device Personalization: System Design & Applications. *arXiv*(arXiv:2102.08503). doi: 10.48550/arXiv.2102.08503

Pessach, D., & Shmueli, E. (2022, February). A Review on Fairness in Machine Learning. *ACM Computing Surveys*, *55*(3), 51:1–51:44. doi: 10.1145/3494672

Philippenko, C., & Dieuleveut, A. (2021). Preserved central model for faster bidirectional compression in distributed settings. In *Advances in Neural Information Processing Systems* (Vol. 34, pp. 2387–2399). Curran Associates, Inc.

Pillutla, K., Kakade, S. M., & Harchaoui, Z. (2022). Robust Aggregation for Federated Learning. *IEEE Transactions on Signal Processing*, *70*, 1142–1154. doi: 10.1109/TSP.2022.3153135

Polyzotis, N., Roy, S., Whang, S. E., & Zinkevich, M. (2017). Data management challenges in production machine learning. In *Proceedings of the 2017 acm international conference on management of data* (p. 1723–1726). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3035918.3054782 doi: 10.1145/3035918.3054782

Polyzotis, N., Roy, S., Whang, S. E., & Zinkevich, M. (2018, dec). Data lifecycle challenges in production machine learning: A survey. *SIGMOD Rec.*, *47*(2), 17–28. Retrieved from https://doi.org/10.1145/3299887.3299891 doi: 10.1145/3299887.3299891

Reddi, S. J., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konečný, J., ... McMahan, H. B. (2021). Adaptive Federated Optimization. In *International Conference on Learning Representations.*

*Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC.* (2016).

Ren, W.-Q., Qu, Y.-B., Dong, C., Jing, Y.-Q., Sun, H., Wu, Q.-H., & Guo, S. (2023). A Survey on Collaborative DNN Inference for Edge Intelligence. *Machine Intelligence Research*, *20*(3), 370–395.

Ribero, M., Vikalo, H., & de Veciana, G. (2023). Federated Learning Under Intermittent Client Availability and Time-Varying Communication Constraints. *IEEE Journal of Selected Topics in Signal Processing*, *17*(1), 98–111. doi: 10.1109/JSTSP.2022.3224590

Richtárik, P., & Takáč, M. (2016). Distributed Coordinate Descent Method for Learning with Big Data. *Journal of Machine Learning Research*, *17*(75), 1–25.

Rizk, E., Vlaski, S., & Sayed, A. H. (2022). Federated Learning Under Importance Sampling. *IEEE Transactions on Signal Processing*, *70*, 5381–5396. doi: 10.1109/TSP.2022.3210365

Robbins, H., & Monro, S. (1951). A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, *22*(3), 400–407. doi: 10.1214/aoms/1177729586

Rodio, A. (2024). The multiple facets of Variance Reduction in Federated Learning. *ACM SIGMETRICS Performance Evaluation Review (To Appear)*.

Rodio, A., Faticanti, F., Marfoq, O., Neglia, G., & Leonardi, E. (2023a). Federated Learning under Heterogeneous and Correlated Client Availability. In *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications*.

Rodio, A., Faticanti, F., Marfoq, O., Neglia, G., & Leonardi, E. (2023b). Federated Learning under Heterogeneous and Correlated Client Availability. *IEEE/ACM Transactions on Networking*, 1–10. doi: 10.1109/TNET.2023.3324257

Rodio, A., & Neglia, G. (2024). FedStale: Leveraging stale client updates in federated learning. *arXiv*(arXiv:2405.04171). doi: 10.48550/arXiv.2405.04171

Rodio, A., Neglia, G., Busacca, F., Mangione, S., Palazzo, S., Restuccia, F., & Tinnirello, I. (2023). Federated Learning with Packet Losses. In *2023 26th international symposium on wireless personal multimedia communications (WPMC)* (pp. 1–6). doi: 10.1109/WPMC59531.2023.10338845

Roh, Y., Heo, G., & Whang, S. E. (2021, April). A Survey on Data Collection for Machine Learning: A Big Data - AI Integration Perspective. *IEEE Transactions on Knowledge and Data Engineering*, *33*(4), 1328–1347. doi: 10.1109/TKDE.2019.2946162

Salehi, M., & Hossain, E. (2021). Federated Learning in Unreliable and Resource-Constrained Cellular Wireless Networks. *IEEE Transactions on Communications*, *69*(8), 5136–5151. doi: 10.1109/TCOMM.2021.3081746

Salem, T. S., Castellano, G., Neglia, G., Pianese, F., & Araldo, A. (2023). Toward inference delivery networks: Distributing machine learning with optimality guarantees. *IEEE/ACM Transactions on Networking*.

Sattler, F., Müller, K.-R., & Samek, W. (2021). Clustered Federated Learning: Model-Agnostic Distributed Multitask Optimization Under Privacy Constraints. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(8), 3710–3722. doi: 10.1109/TNNLS.2020.3015958

Scaman, K., Bach, F., Bubeck, S., Lee, Y. T., & Massoulié, L. (2019). Optimal Convergence Rates for Convex Distributed Optimization in Networks. *Journal of Machine Learning Research*, *20*(159), 1–31.

Schmidt, M., Le Roux, N., & Bach, F. (2017). Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, *162*(1), 83–112. doi: 10.1007/s10107-016-1030-6

Shalev-Shwartz, S., & Ben-David, S. (2014). *Understanding machine learning - from theory to algorithms*. Cambridge University Press.

Silva, S., Altmann, A., Gutman, B., & Lorenzi, M. (2020). Fed-BioMed: A General Open-Source Frontend Framework for Federated Learning in Healthcare. In S. Albarqouni et al. (Eds.), *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning* (pp. 201–210). Cham: Springer International Publishing. doi: 10.1007/978-3-030-60548-3_20

Stoica, I., & Shenker, S. (2021). From cloud computing to sky computing. In *Proceedings of the Workshop on Hot Topics in Operating Systems* (pp. 26–32). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/3458336.3465301

Su, X., Zhou, Y., Cui, L., & Liu, J. (2023). On Model Transmission Strategies in Federated Learning With Lossy Communications. *IEEE Transactions on Parallel and Distributed Systems*, *34*(4), 1173–1185. doi: 10.1109/TPDS.2023.3240883

Sun, T., Sun, Y., & Yin, W. (2018). On Markov Chain Gradient Descent. In *Advances in Neural Information Processing Systems* (Vol. 31). Curran Associates, Inc.

Tan, L., Zhang, X., Zhou, Y., Che, X., Hu, M., Chen, X., & Wu, D. (2022). AdaFed: Optimizing Participation-Aware Federated Learning with Adaptive Aggregation Weights. *IEEE Transactions on Network Science and Engineering*.

Tang, M., Ning, X., Wang, Y., Sun, J., Wang, Y., Li, H., & Chen, Y. (2022). FedCor: Correlation-based active client selection strategy for heterogeneous federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.*

Teerapittayanon, S., McDanel, B., & Kung, H.-T. (2016). Branchynet: Fast inference via early exiting from deep neural networks. In *2016 23rd international conference on pattern recognition (ICPR)* (pp. 2464–2469). IEEE.

Teerapittayanon, S., McDanel, B., & Kung, H.-T. (2017). Distributed deep neural networks over the cloud, the edge and end devices. In *2017 IEEE 37th international conference on distributed computing systems (ICDCS)* (pp. 328–339). IEEE.

Tu, X., Zhu, K., Luong, N. C., Niyato, D., Zhang, Y., & Li, J. (2022). Incentive Mechanisms for Federated Learning: From Economic and Game Theoretic Perspective. *IEEE Transactions on Cognitive Communications and Networking*, *8*(3), 1566–1593. doi: 10.1109/TCCN.2022.3177522

Uhlemann, T. H.-J., Lehmann, C., & Steinhilper, R. (2017). The digital twin: Realizing the cyber-physical production system for industry 4.0. *Procedia CIRP*, *61*, 335–340. doi: 10.1016/j.procir.2016.11.152

Verbraeken, J., Wolting, M., Katzy, J., Kloppenburg, J., Verbelen, T., & Rellermeyer, J. S. (2020). A Survey on Distributed Machine Learning. *ACM Computing Surveys*, *53*(2), 30:1–30:33. doi: 10.1145/3377454

Wainakh, A., Guinea, A. S., Grube, T., & Mühlhäuser, M. (2020). Enhancing Privacy via Hierarchical Federated Learning. In *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)* (pp. 344–347). doi: 10.1109/EuroSPW51379.2020.00053

Wang, H., Yurochkin, M., Sun, Y., Papailiopoulos, D., & Khazaeni, Y. (2020). Federated Learning with Matched Averaging. In *International conference on learning representations.*

Wang, J., Charles, Z., Xu, Z., Joshi, G., McMahan, H. B., y Arcas, B. A., ... Zhu, W. (2021). A Field Guide to Federated Optimization. *arXiv*(arXiv:2107.06917). doi: 10.48550/arXiv.2107.06917

Wang, J., Liu, Q., Liang, H., Joshi, G., & Poor, H. V. (2020). Tackling the Objective Inconsistency Problem in Heterogeneous Federated Optimization. In *Advances in Neural Information Processing Systems* (Vol. 33, pp. 7611–7623). Curran Associates, Inc.

Wang, J., Sahu, A. K., Yang, Z., Joshi, G., & Kar, S. (2019). MATCHA: Speeding Up Decentralized SGD via Matching Decomposition Sampling. In *2019 Sixth Indian Control Conference (ICC)* (pp. 299–300). doi: 10.1109/ICC47138.2019.9123209

Wang, S., & Ji, M. (2022). A Unified Analysis of Federated Learning with Arbitrary Client Participation. *Advances in Neural Information Processing Systems*, *35*, 19124–19137.

Wang, S., & Ji, M. (2024). A Lightweight Method for Tackling Unknown Participation Statistics in Federated Averaging. *arXiv*(arXiv:2306.03401).

Wen, D., Li, X., Zeng, Q., Ren, J., & Huang, K. (2019). An Overview of Data-Importance Aware Radio Resource Management for Edge Machine Learning. *Journal of Communications and Information Networks*, *4*(4), 1–14.

Xing, E. P., Ho, Q., Xie, P., & Wei, D. (2016). Strategies and Principles of Distributed Machine Learning on Big Data. *Engineering*, *2*(2), 179–195. doi: 10.1016/J.ENG.2016.02.008

Xu, C., Neglia, G., & Sebastianelli, N. (2021). Dynamic backup workers for parallel machine learning. *Computer Networks*, *188*, 107846. doi: 10.1016/j.comnet.2021.107846

Yajnik, M., Moon, S., Kurose, J., & Towsley, D. (1999). Measurement and modelling of the temporal dependence in packet loss. In *IEEE INFOCOM '99. Conference on Computer Communications.* (Vol. 1, p. 345-352 vol.1). doi: 10.1109/INFCOM.1999.749301

Yan, Y., Niu, C., Ding, Y., Zheng, Z., Tang, S., Li, Q., ... Chen, G. (2024). Federated Optimization Under Intermittent Client Availability. *INFORMS Journal on Computing*, *36*(1), 185–202. doi: 10.1287/ijoc.2022.0057

Yang, H., Fang, M., & Liu, J. (2020). Achieving Linear Speedup with Partial Worker Participation in Non-IID Federated Learning. In *International Conference on Learning Representations.*

Yang, H., Zhang, X., Khanduri, P., & Liu, J. (2022). Anarchic Federated Learning. In *Proceedings of the 39th International Conference on Machine Learning* (pp. 25331–25363). PMLR.

Yang, H. H., Liu, Z., Quek, T. Q. S., & Poor, H. V. (2020). Scheduling Policies for Federated Learning in Wireless Networks. *IEEE Transactions on Communications*, *68*(1), 317–333. doi: 10.1109/TCOMM.2019.2944169

Ye, H., Liang, L., & Li, G. Y. (2022). Decentralized Federated Learning With Unreliable Communications. *IEEE Journal of Selected Topics in Signal Processing*, *16*(3), 487–500. doi: 10.1109/JSTSP.2022.3152445

Yilmaz, S. F., Hasırcıoğlu, B., & Gündüz, D. (2022). Over-the-air ensemble inference with model privacy. In *2022 IEEE international symposium on information theory (ISIT)* (pp. 1265–1270). IEEE.

Zeng, L., Li, E., Zhou, Z., & Chen, X. (2019). Boomerang: On-demand cooperative deep neural network inference for edge intelligence on the industrial Internet of Things. *IEEE Network*, *33*(5), 96–103.

Zhang, J., Li, C., Robles-Kelly, A., & Kankanhalli, M. (2020, May). Hierarchically Fair Federated Learning. *arXiv*(arXiv:2004.10386). doi: 10.48550/arXiv.2004.10386

Zheng, Y., Li, G., Li, Y., Shan, C., & Cheng, R. (2017, jan). Truth inference in crowdsourcing: is the problem solved? *Proc. VLDB Endow.*, *10*(5), 541–552. Retrieved from https://doi.org/10.14778/3055540.3055547 doi: 10.14778/3055540.3055547

Zhu, C., Xu, Z., Chen, M., Konečnỳ, J., Hard, A., & Goldstein, T. (2021). Diurnal or nocturnal? Federated learning from periodically shifting distributions. In *NeurIPS 2021 workshop on distribution shifts: Connecting methods and applications.*

# List of Figures

# List of Tables

# List of Theorems

# List of Algorithms

# Appendix

# APPENDIX A

# Heterogeneous and Correlated Client Participation

## A Proof of Theorem 2.3.2

**Theorem A.1** (Decomposing the total error)**.** *Let $\kappa \triangleq L/\mu$. Under Assumptions 2–4, the optimization error of the target global objective $\epsilon = F(\boldsymbol{w}) - F^*$ can be bounded as follows:*

$$\epsilon \leq 2\kappa^2 (\underbrace{F_B(\boldsymbol{w}) - F_B^*}_{\triangleq \epsilon_{opt}} + \underbrace{F(\boldsymbol{w}_B^*) - F^*}_{\triangleq \epsilon_{bias}}). \tag{2.10}$$

*Moreover, let $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}} \triangleq \sum_{i=1}^N (\alpha_i - p_i)^2/p_i$. Then:*

$$\epsilon_{bias} \leq \kappa^2 \cdot \underbrace{\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}} \cdot \Gamma}_{\triangleq \bar{\epsilon}_{bias}}. \tag{2.11}$$

The proof of Theorem A.1 employs well-established techniques from convex optimization. It is based on the proof presented in (J. Wang et al., 2020, Theorem 2).

*Proof of Theorem A.1.*

By leveraging the $L$-smoothness and $\mu$-strong convexity properties of $F$, we obtain:

$$\begin{aligned}
F(\boldsymbol{w}) - F^* &\leq \frac{1}{2\mu} \|\nabla F(\boldsymbol{w})\|^2 && \text{(A.1)}\\
&\leq \frac{L^2}{2\mu} \|\boldsymbol{w} - \boldsymbol{w}^*\|^2 && \text{(A.2)}\\
&\leq \frac{L^2}{\mu} (\|\boldsymbol{w} - \boldsymbol{w}_B^*\|^2 + \|\boldsymbol{w}_B^* - \boldsymbol{w}^*\|^2) && \text{(A.3)}\\
&\leq \frac{2L^2}{\mu^2} \Big( \underbrace{F_B(\boldsymbol{w}) - F_B^*}_{\triangleq \epsilon_{\text{opt}}} + \underbrace{F(\boldsymbol{w}_B^*) - F^*}_{\triangleq \epsilon_{\text{bias}}} \Big), && \text{(A.4)}
\end{aligned}$$

where the inequality in (A.1) follows from Assumption 4 and is commonly referred to as the *Polyak-Lojasiewicz inequality*; the inequality in (A.2) is derived using the fact that $\nabla F(\boldsymbol{w}^*) = 0$ (Assumption 2) and the definition of $L$-Lipschitz continuous gradient for $F$ (Assumption 3); the inequality in (A.3) is based on $(a+b)^2 \leq 2(a^2+b^2)$; lastly, the inequality in (A.4) follows from the $\mu$-strong convexity of both $F_B$ and $F$ (Assumptions 4), and uses $\nabla F_B(\boldsymbol{w}_B^*) = 0$ and $\nabla F(\boldsymbol{w}^*) = 0$ (Assumption 2). The obtained results complete the first part of the proof, establishing the bound in (2.10).

Next, to prove the relation in (2.11), we proceed by bounding the term $\epsilon_{\text{bias}}$ as follows:

$$\epsilon_{\text{bias}} \triangleq (F(\boldsymbol{w}_B^*) - F^*) \leq \frac{1}{2\mu} \left\|\nabla F(\boldsymbol{w}_B^*)\right\|^2, \tag{A.5}$$

where the inequality in (A.5) directly follows from the Polyak-Lojasiewicz inequality (Assumption 4).

Furthermore, we bound the term $\|\nabla F(\boldsymbol{w}_B^*)\|$ as follows:

$$\begin{aligned}
\|\nabla F(\boldsymbol{w}_B^*)\| &= \left\|\sum_{i=1}^{N}(\alpha_i - p_i)\nabla F_i(\boldsymbol{w}_B^*)\right\| && \text{(A.6)}\\
&\leq \sum_{i=1}^{N}|\alpha_i - p_i| \left\|\nabla F_i(\boldsymbol{w}_B^*)\right\| && \text{(A.7)}\\
&\leq L\sum_{i=1}^{N}|\alpha_i - p_i| \left\|\boldsymbol{w}_B^* - \boldsymbol{w}_i^*\right\| && \text{(A.8)}\\
&\leq L\sqrt{\frac{2}{\mu}}\sum_{i=1}^{N}|\alpha_i - p_i|\sqrt{(F_i(\boldsymbol{w}_B^*) - F_i^*)}, && \text{(A.9)}
\end{aligned}$$

where, in (A.6), we use $\nabla F_B(\boldsymbol{w}_B^*) = 0$ (Assumption 2) and apply the definitions of $F$ and $F_B$ given in (2.1) and (2.4), respectively. The bound in (A.7) follows from the triangle inequality. Next, the inequality in (A.8) uses $\nabla F_i(\boldsymbol{w}_i^*) = 0$ (Assumption 2) and the $L$-smoothness of $F_i$ (Assumption 3). Finally, the inequality in (A.9) leverages the $\mu$-strong convexity of $F_i$ (Assumption 4) and $\nabla F_i(\boldsymbol{w}_i^*) = 0$ (Assumption 2), and follows multiplying and dividing by $\sqrt{p_i}$.

By squaring both sides of Equation (A.9), we obtain:

$$\begin{aligned}
\|\nabla F(\boldsymbol{w}_B^*)\|^2 &\leq \frac{2L^2}{\mu}\left(\sum_{i=1}^{N}\frac{|\alpha_i - p_i|}{\sqrt{p_i}}\sqrt{p_i(F_i(\boldsymbol{w}_B^*) - F_i^*)}\right)^2 && \text{(A.10)}\\
&\leq \frac{2L^2}{\mu}\left(\sum_{i=1}^{N}\frac{(\alpha_i - p_i)^2}{p_i}\right)\left(\sum_{i=1}^{N}p_i(F_i(\boldsymbol{w}_B^*) - F_i^*)\right) && \text{(A.11)}\\
&\leq \frac{2L^2}{\mu}\cdot\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}}\cdot\Gamma, && \text{(A.12)}
\end{aligned}$$

where the inequality in (A.11) follows from the Cauchy-Schwarz inequality. Furthermore, the inequality in (A.12) holds because:

$$\sum_{i=1}^{N}p_i(F_i(\boldsymbol{w}_B^*) - F_i^*) = F_B^* - \sum_{i=1}^{N}p_iF_i^* \tag{A.13}$$

$$\leq F_B(\boldsymbol{w}^*) - \sum_{i=1}^{N} p_i F_i^* \tag{A.14}$$

$$= \sum_{i=1}^{N} p_i (F_i(\boldsymbol{w}^*) - F_i^*) \tag{A.15}$$

$$\leq \max_{i \in \mathcal{N}} \{F_i(\boldsymbol{w}^*) - F_i^*\} \triangleq \Gamma. \tag{A.16}$$

We remark that the inequality in (A.14) only holds if $\boldsymbol{w}_B^*$ is the global minimizer of $F_B$, as guaranteed by Assumption 2. By replacing (A.12) into (A.5), we have:

$$\epsilon_{\text{bias}} \leq \frac{1}{2\mu} \|\nabla F(\boldsymbol{w}_B^*)\|^2 \leq \frac{L^2}{\mu^2} \cdot \chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}} \cdot \Gamma, \tag{A.17}$$

which concludes the proof of Equation (2.11), and therefore, of Theorem A.1.

□

# B Proof of Theorem 2.3.3

## B.A Algorithm Overview and Supplementary Notation

Let $\boldsymbol{w}_i^{(t,k)}$ represent the model parameter maintained by the $i$-th client during the $t$-th global communication round and the $k$-th local step. The $t$-th global communication round can be described as follows: 1) The server broadcasts the model parameter $\boldsymbol{w}^{(t,0)}$ to the active clients, which adopt it as their local model, i.e., $\boldsymbol{w}_i^{(t,0)} = \boldsymbol{w}^{(t,0)}$ for $i \in \mathcal{S}^{(t)}$; 2) Each active client $i \in \mathcal{S}^{(t)}$ generates a sequence of local models $\{\boldsymbol{w}_i^{(t,k)}\}_{k=1}^{K}$ using the local-SGD update rule defined in (2.2); 3) The active clients send their model updates $\Delta_i^{(t)} \triangleq \boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t,0)}$ back to the server; 4) The server aggregates the model updates using the aggregation rule specified in (2.3), resulting in the new global model parameter $\boldsymbol{w}^{(t+1,0)}$.

$$\begin{cases} \boldsymbol{w}_{t,j+1}^{k} = \boldsymbol{w}_i^{(t,k)} - \eta_c^{(t)} \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) & \text{for } k = 0, \ldots, K-1; \quad (2.2) \\ \boldsymbol{w}^{(t+1,0)} = \mathbf{Proj}_W(\boldsymbol{w}^{(t,0)} + \sum_{i \in \mathcal{S}^{(t)}} q_i \left( \boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t,0)} \right)) & \text{for } k = K. \quad (2.3) \end{cases}$$

The projection operator in (2.3) ensures that the current iterate $\boldsymbol{w}^{(t+1,0)}$ in the optimization algorithm defined by (2.2) and (2.3) remains within the feasible region $W$.

### Sources of randomness

In the system, we model two sources of randomness. The first arises from the availability of random clients, which follows a Markov process as stated in Assumption 1. The second source of randomness originates from the random sampling of batches for computing stochastic gradients. Remember that $\mathcal{S}^{(t)}$ denotes the random set of clients available at the $t$-th communication round and that $\mathcal{B}_i^{(t,k)}$ denotes the random batch independently sampled from client-$i$'s local dataset at round $t$, local iteration $k$. For the analysis, we introduce the following additional notation:

$\mathcal{S}^{(i:j)} \triangleq \{\mathcal{S}^{(i)}, \ldots, \mathcal{S}^{(j)}\}$: the family of random sets of clients available from the $i$-th to the $j$-th communication rounds, $i < j$;

$\mathcal{B}_i^{(t)} \triangleq \{\mathcal{B}_i^{(t,k)}\}_{k=0}^{K-1}$: the set of random batches sampled by the $i$-th client at the $t$-th communication round;

$\mathcal{B}^{(t)} \triangleq \{\mathcal{B}_i^{(t)}\}_{i\in\mathcal{S}^{(t)}}$: the set of random batches sampled by the available clients ($\mathcal{S}^{(t)}$) in the $t$-th communication round;

$\mathcal{B}_i^{(t,i:j)} \triangleq \{\mathcal{B}_i^{(t,i)}, \ldots, \mathcal{B}_i^{(t,j)}\}$: the set of random batches sampled by the $i$-th client at the $t$-th communication round between the $i$-th and the $j$-th local iterations, $i < j$;

$\mathcal{B}^{(t,i:j)} \triangleq \{\mathcal{B}^{(i)}, \ldots, \mathcal{B}^{(j)}\}$: the set of random batches sampled by the available clients ($\mathcal{S}^{(i:j)}$) between the $i$-th and $j$-th communication rounds, $i < j$.

With this notation established, the randomness in the $t$-th communication round, which starts with the initial model $\boldsymbol{w}^{(t,0)}$ and yields the updated model $\boldsymbol{w}^{(t+1,0)}$, is fully determined by the sets $\mathcal{S}^{(t)}$ and $\mathcal{B}^{(t)}$. This implies that the evolution of the algorithm, governed by the update rules in (2.2) and (2.3), from round 0 to round $t$ can be completely described by the tuple:

$$\mathcal{H}^{(t)} \triangleq \left(\mathcal{S}^{(0)}, \ldots, \mathcal{S}^{(t-1)}; \mathcal{B}^{(0)}, \ldots, \mathcal{B}^{(t-1)}\right), \tag{A.18}$$

which represents the historical information up to the $t$-th communication round.

We introduce the following additional quantities for our analysis:

$$\boldsymbol{g}^{(t)}(\mathcal{S}^{(t)}, \mathcal{B}^{(t)}) \triangleq \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}), \tag{A.19}$$

and

$$\bar{\boldsymbol{g}}^{(t)}(\mathcal{S}^{(t)}, \mathcal{B}^{(t)}) \triangleq \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \tag{A.20}$$

where $\boldsymbol{g}^{(t)}(\mathcal{S}^{(t)}, \mathcal{B}^{(t)})$ denotes the global pseudo-gradient computed at communication round $t$, aggregated from the active clients in $\mathcal{S}^{(t)}$, and $\bar{\boldsymbol{g}}^{(t)}(\mathcal{S}^{(t)}, \mathcal{B}^{(t)})$ denotes its expected value with respect to the choices of the random batches $\mathcal{B}_i^{(t,k)}$, for all $k = 0, \ldots, K-1$ and $i \in \mathcal{S}^{(t)}$. With this notation established, the global update rule for the $t$-th communication round can be expressed as:

$$\boldsymbol{w}^{(t+1,0)} = \mathbf{Proj}_W(\boldsymbol{w}^{(t,0)} - \eta_c^{(t)} \boldsymbol{g}^{(t)}(\mathcal{S}^{(t)}, \mathcal{B}^{(t)})). \tag{A.21}$$

## B.B Supporting Lemmas

In this section, we introduce several lemmas that are instrumental in proving Theorem B.19. Firstly, we prove Lemma 2.3.1, introduced in Section 2.3. Its proof relies on the convexity and compactness of the hypothesis class $W$ (Assumption 2), on the $L$-smoothness of the functions $\{F_i\}_{i\in\mathcal{N}}$ (Assumption 3), and on the bounded variance of the stochastic gradients (Assumption 5).

**Lemma B.1.** *Under Assumptions 2, 3, and 5, there exist constants $D$, $G$, and $H > 0$, such that, for $\boldsymbol{w} \in W$ and $i \in \mathcal{N}$, we have:*

$$\|\nabla F_i(\boldsymbol{w})\| \leq D, \tag{2.6}$$

$$\mathbb{E}_{\mathcal{B}}\|\nabla F_i(\boldsymbol{w},\mathcal{B})\|^2 \leq G^2, \tag{2.7}$$
$$|F_i(\boldsymbol{w}) - F_i(\boldsymbol{w}_B^*)| \leq H. \tag{2.8}$$

*Proof of Lemma B.1.*

The boundedness of the hypothesis class $W$ (Assumption 2) provides a bound on the sequence $(\boldsymbol{w}^{(t,0)})_{t\geq 0}$ generated by the scheme defined in Equations (2.2) and (2.3). Moreover, since $\boldsymbol{w}_i^*$ minimizes $\nabla F_i(\boldsymbol{w})$, we have $\nabla F_i(\boldsymbol{w}_i^*) = 0$. Furthermore, the $L$-smoothness of $\{F_i\}_{i\in\mathcal{N}}$ (Assumption 3) leads to the following inequality:

$$\|\nabla F_i(\boldsymbol{w})\| = \|\nabla F_i(\boldsymbol{w}) - \nabla F_i(\boldsymbol{w}_i^*)\| \leq L\,\|\boldsymbol{w} - \boldsymbol{w}_i^*\| \triangleq D < +\infty. \tag{A.22}$$

The bound in (2.6) is directly derived from (A.22), while the bound in (2.8) follows from the continuity of $\{F_i\}_{i\in\mathcal{N}}$ over the compact set $W$ (Assumption 2). Finally, the inequality in (2.7) requires a bound on the variance of the stochastic gradients (Assumption 5). In particular, it holds that:

$$\mathbb{E}_{\mathcal{B}}\|\nabla F_i(\boldsymbol{w},\mathcal{B})\|^2 \leq D^2 + \max_{i\in\mathcal{N}}\{\sigma_i^2\} \triangleq G^2. \tag{A.23}$$

□

The following lemma proves that the global pseudo-gradient $\boldsymbol{g}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})$ is an unbiased estimator of $\bar{\boldsymbol{g}}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})$. A similar result has been used in previous works, specifically in (J. Wang et al., 2020, Appendix C1). Here, we provide a comprehensive proof for this result.

**Lemma B.2.** *Let $\boldsymbol{g}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})$ and $\bar{\boldsymbol{g}}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})$ be defined as in (A.19) and (A.20), respectively. The following equality holds:*

$$\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\boldsymbol{g}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})\right] = \mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\bar{\boldsymbol{g}}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})\right]. \tag{A.24}$$

*Proof of Lemma B.2.*

$$\begin{aligned}
&\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\boldsymbol{g}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})\right] \\
&\quad= \mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})\right] && \text{(A.25)}\\
&\quad= \sum_{i\in\mathcal{S}^{(t)}} q_i \mathbb{E}_{\mathcal{B}_i^{(t)}}\left[\sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})\right] && \text{(A.26)}\\
&\quad= \sum_{i\in\mathcal{S}^{(t)}} q_i \Big[\mathbb{E}_{\mathcal{B}_i^{(t,0)}}[\nabla F_i(\boldsymbol{w}^{(t,0)},\mathcal{B}_i^{(t,0)})] + \mathbb{E}_{\mathcal{B}_i^{(t,0)},\mathcal{B}_i^{(t,1)}}[\nabla F_i(\boldsymbol{w}_i^{(t,1)},\mathcal{B}_i^{(t,1)})] \\
&\qquad\qquad + \cdots + \mathbb{E}_{\mathcal{B}_i^{(t,0:K-1)}}[\nabla F_i(\boldsymbol{w}_i^{(t,K-1)},\mathcal{B}_i^{(t,K-1)})]\Big] && \text{(A.27)}\\
&\quad= \sum_{i\in\mathcal{S}^{(t)}} q_i \Big[\nabla F_i(\boldsymbol{w}^{(t,0)}) + \mathbb{E}_{\mathcal{B}_i^{(t,0)}}\left[\mathbb{E}_{\mathcal{B}_i^{(t,1)}|\mathcal{B}_i^{(t,0)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,1)},\mathcal{B}_i^{(t,1)})\right]\right]
\end{aligned}$$

$$+\cdots+\mathbb{E}_{\mathcal{B}_i^{(t,0:K-2)}}\left[\mathbb{E}_{\mathcal{B}_i^{(t,0:K-1)}|\mathcal{B}_i^{(t,0:K-2)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,K-1)},\mathcal{B}_i^{(t,K-1)})\right]\right]\Bigg] \tag{A.28}$$

$$=\sum_{i\in\mathcal{S}^{(t)}} q_i\left[\nabla F_i(\boldsymbol{w}^{(t,0)})+\mathbb{E}_{\mathcal{B}_i^{(t,0)}}[\nabla F_i(\boldsymbol{w}_i^{(t,1)})]+\cdots+\mathbb{E}_{\mathcal{B}_i^{(t,0:K-2)}}[\nabla F_i(\boldsymbol{w}_i^{(t,K-1)})]\right] \tag{A.29}$$

$$=\sum_{i\in\mathcal{S}^{(t)}} q_i\mathbb{E}_{\mathcal{B}_i^{(t,0:K-2)}}\left[\sum_{k=0}^{K-1}\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right] \tag{A.30}$$

$$=\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\sum_{k=0}^{K-1}\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right]=\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\bar{\boldsymbol{g}}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})\right], \tag{A.31}$$

where, in (A.26), we considered that both the evolution of the local models $\{\boldsymbol{w}_i^{(t,k)}\}_{k=0}^{K-1}$ and the choices of the random batches $\{\mathcal{B}_i^{(t,k)}\}_{k=0}^{K-1}$ are independent among different clients $i\in\mathcal{S}^{(t)}$ within the same communication round $t\in\mathcal{T}$.

□

For the sake of simplicity, we will henceforth denote $\boldsymbol{g}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})$ and $\bar{\boldsymbol{g}}^{(t)}(\mathcal{S}^{(t)},\mathcal{B}^{(t)})$ as $\boldsymbol{g}^{(t)}$ and $\bar{\boldsymbol{g}}^{(t)}$, respectively. The following lemma decomposes the optimization error into multiple components, which we will bound separately in subsequent lemmas.

**Lemma B.3** (Decomposition of the error in a global communication round)**.** *Let Assumption 2 hold. We have:*

$$\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left\|\boldsymbol{w}^{(t+1,0)}-\boldsymbol{w}_B^*\right\|^2\leq$$

$$\left\|\boldsymbol{w}^{(t,0)}-\boldsymbol{w}_B^*\right\|^2\underbrace{-2\eta_c^{(t)}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\langle\boldsymbol{w}^{(t,0)}-\boldsymbol{w}_B^*,\bar{\boldsymbol{g}}^{(t)}\rangle}_{\text{bounded in Lemma B.4}}+\underbrace{(\eta_c^{(t)})^2\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}^{(t)}\right\|^2}_{\text{bounded in Lemma B.5}}$$

$$+\underbrace{2\eta_c^{(t)}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\langle\boldsymbol{w}^{(T,0)}-\boldsymbol{w}_B^*-\eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)},\bar{\boldsymbol{g}}^{(t)}-\boldsymbol{g}^{(t)}\rangle}_{\text{bounded in Lemma B.6}}+\underbrace{(\eta_c^{(t)})^2\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left\|\boldsymbol{g}^{(t)}-\bar{\boldsymbol{g}}^{(t)}\right\|^2}_{\text{bounded in Lemma B.7}}. \tag{A.32}$$

*Proof of Lemma B.3.*

$$\left\|\boldsymbol{w}^{(t+1,0)}-\boldsymbol{w}_B^*\right\|^2=\left\|\mathbf{Proj}_W(\boldsymbol{w}^{(t,0)}-\eta_c^{(t)}\boldsymbol{g}^{(t)})-\mathbf{Proj}_W(\boldsymbol{w}_B^*)\right\|^2 \tag{A.33}$$

$$\leq\left\|\boldsymbol{w}^{(t,0)}-\eta_c^{(t)}\boldsymbol{g}^{(t)}-\boldsymbol{w}_B^*+\eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)}-\eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)}\right\|^2 \tag{A.34}$$

$$=\left\|\boldsymbol{w}^{(t,0)}-\boldsymbol{w}_B^*-\eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)}\right\|^2$$

$$+2\eta_c^{(t)}\langle\boldsymbol{w}^{(T,0)}-\boldsymbol{w}_B^*-\eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)},\bar{\boldsymbol{g}}^{(t)}-\boldsymbol{g}^{(t)}\rangle+(\eta_c^{(t)})^2\left\|\boldsymbol{g}^{(t)}-\bar{\boldsymbol{g}}^{(t)}\right\|^2 \tag{A.35}$$

$$=\left\|\boldsymbol{w}^{(t,0)}-\boldsymbol{w}_B^*\right\|^2-2\eta_c^{(t)}\langle\boldsymbol{w}^{(t,0)}-\boldsymbol{w}_B^*,\bar{\boldsymbol{g}}^{(t)}\rangle+(\eta_c^{(t)})^2\left\|\bar{\boldsymbol{g}}^{(t)}\right\|^2$$

$$+2\eta_c^{(t)}\langle\boldsymbol{w}^{(T,0)}-\boldsymbol{w}_B^*-\eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)},\bar{\boldsymbol{g}}^{(t)}-\boldsymbol{g}^{(t)}\rangle+(\eta_c^{(t)})^2\left\|\boldsymbol{g}^{(t)}-\bar{\boldsymbol{g}}^{(t)}\right\|^2, \tag{A.36}$$

where, in (A.33), we used Assumption 2; whereas, the inequality in (A.34) is due to the contracting property of projection. We observe that (A.34) does not hold in general if $\boldsymbol{w}_B^*\notin W$.

□

In what follows, we present a series of lemmas to establish bounds for the error in (A.32).

**Lemma B.4.** *Let Assumption 3 hold and the local functions $\{F_i\}_{i=1}^N$ be convex. We have:*

$$-2\eta_c^{(t)}\langle \boldsymbol{w}^{(t,0)} - \boldsymbol{w}_B^*, \bar{\boldsymbol{g}}^{(t)}\rangle \leq -2\eta_c^{(t)}(1-\eta_c^{(t)}L)\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right)$$
$$+ \underbrace{\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2}_{\text{bounded in Lemma B.9}} + 2(\eta_c^{(t)})^2 LK \underbrace{\sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right)}_{\text{bounded in Lemma B.10}}. \tag{A.37}$$

*Proof of Lemma B.4.*

We decompose the term $-2\eta_c^{(t)}\langle \boldsymbol{w}^{(t,0)} - \boldsymbol{w}_B^*, \bar{\boldsymbol{g}}^{(t)}\rangle$, by adding and subtracting $\boldsymbol{w}_i^{(t,k)}$:

$$-2\eta_c^{(t)}\langle \boldsymbol{w}^{(t,0)} - \boldsymbol{w}_B^*, \bar{\boldsymbol{g}}^{(t)}\rangle = \underbrace{-2\eta_c^{(t)}\langle \boldsymbol{w}^{(t,0)} - \boldsymbol{w}_i^{(t,k)}, \bar{\boldsymbol{g}}^{(t)}\rangle}_{\text{developed in Eq. (A.39)}} \underbrace{-2\eta_c^{(t)}\langle \boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}_B^*, \bar{\boldsymbol{g}}^{(t)}\rangle}_{\text{developed in Eq. (A.43)}}. \tag{A.38}$$

We bound the two terms separately. We bound the first term in (A.38) as:

$$-2\eta_c^{(t)}\langle \boldsymbol{w}^{(t,0)} - \boldsymbol{w}_i^{(t,k)}, \bar{\boldsymbol{g}}^{(t)}\rangle = -2\eta_c^{(t)}\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \boldsymbol{w}^{(t,0)} - \boldsymbol{w}_i^{(t,k)}\rangle \tag{A.39}$$
$$\leq (\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right\|^2 + \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2 \tag{A.40}$$
$$\leq 2(\eta_c^{(t)})^2 L \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i^*\right) + \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2 \tag{A.41}$$
$$= 2(\eta_c^{(t)})^2 L \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right) + \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2$$
$$+ 2(\eta_c^{(t)})^2 LK \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right), \tag{A.42}$$

where, in (A.40), we used $|\langle \boldsymbol{a}, \boldsymbol{b}\rangle| \leq \frac{1}{2}\|\boldsymbol{a}\|^2 + \frac{1}{2}\|\boldsymbol{b}\|^2$; in (A.41), we applied the $L$-smoothness of $\{F_i(\boldsymbol{w})\}_{i\in\mathcal{N}}$ (Assumption 3); in (A.42), we added and subtracted $F_i(\boldsymbol{w}_B^*)$.

We bound the second term in (A.38) as:

$$-2\eta_c^{(t)}\langle \boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}_B^*, \bar{\boldsymbol{g}}^{(t)}\rangle = -2\eta_c^{(t)}\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\langle \boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}_B^*, \nabla F_i(\boldsymbol{w}_i^{(t,k)})\rangle \tag{A.43}$$
$$\leq -2\eta_c^{(t)}\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right), \tag{A.44}$$

where, in (A.44), we use the convexity of $\{F_i(\boldsymbol{w})\}_{i\in\mathcal{N}}$.

By summing the bounds provided in (A.42) and (A.44), we conclude the proof.

□

**Lemma B.5** (Bound on the squared norm of a global gradient step)**.** *Let Assumption 3 hold. We have:*

$$(\eta_c^{(t)})^2 \left\|\bar{\boldsymbol{g}}^{(t)}\right\|^2 \leq 2(\eta_c^{(t)})^2 LKQ \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right) + 2(\eta_c^{(t)})^2 LK^2Q \underbrace{\sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right)}_{\text{bounded in Lemma B.10}}. \tag{A.45}$$

*Proof of Lemma B.5.*

$$(\eta_c^{(t)})^2 \left\|\bar{\boldsymbol{g}}^{(t)}\right\|^2 = (\eta_c^{(t)})^2 \left\| \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}) \right\|^2 \tag{A.46}$$

$$\leq (\eta_c^{(t)})^2 \sum_{i'\in\mathcal{S}^{(t)}} q_{k'} \sum_{i\in\mathcal{S}^{(t)}} q_i \left\| \sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}) \right\|^2 \tag{A.47}$$

$$\leq (\eta_c^{(t)})^2 QE \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left\|\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right\|^2 \tag{A.48}$$

$$\leq 2(\eta_c^{(t)})^2 QLK \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i^*\right) \tag{A.49}$$

$$= 2(\eta_c^{(t)})^2 LKQ \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right) + 2(\eta_c^{(t)})^2 LK^2Q \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right), \tag{A.50}$$

where, in (A.47) and in (A.48), we applied the Jensen's inequality; in (A.48), we also observed that $\sum_{i\in\mathcal{S}^{(t)}} q_i \leq \sum_{i\in\mathcal{N}} q_i \triangleq Q$; in (A.49), we used the $L$-smoothness of $\{F_i(\boldsymbol{w})\}_{i\in\mathcal{N}}$ (Assumption 3); in (A.50), we added and subtracted $F_i(\boldsymbol{w}_B^*)$ to the sum.

□

**Lemma B.6.** *Let Assumption 5 hold. We have:*

$$2\eta_c^{(t)} \mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[\langle \boldsymbol{w}^{(T,0)} - \boldsymbol{w}_B^* - \eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)}, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)}\rangle\right] \leq 2(\eta_c^{(t)})^2 LKQ \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=1}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right] + \frac{1}{2}(\eta_c^{(t)})^2 K(K-1) \sum_{i\in\mathcal{S}^{(t)}} q_i^2\sigma_i^2 + 2(\eta_c^{(t)})^2 LK^2Q \underbrace{\sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right)}_{\text{bounded in Lemma B.10}}. \tag{A.51}$$

*Proof of Lemma B.6.*

We decompose the term $\langle \boldsymbol{w}^{(T,0)} - \boldsymbol{w}_B^* - \eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)}, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)}\rangle$ in two parts:

$$\begin{aligned} &2\eta_c^{(t)}\langle \boldsymbol{w}^{(T,0)} - \boldsymbol{w}_B^* - \eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)}, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)}\rangle \\ &\quad = 2\eta_c^{(t)}\langle \boldsymbol{w}^{(T,0)} - \boldsymbol{w}_B^*, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)}\rangle - 2(\eta_c^{(t)})^2\langle \bar{\boldsymbol{g}}^{(t)}, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)}\rangle. \end{aligned} \tag{A.52}$$

From Lemma B.2, we conclude that $\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\langle \boldsymbol{w}^{(T,0)} - \boldsymbol{w}_B^*, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)}\rangle = 0$.

We now focus on:

$$-2(\eta_c^{(t)})^2\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\langle \bar{\boldsymbol{g}}^{(t)}, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)}\rangle\right] = \tag{A.53}$$

$$= -2(\eta_c^{(t)})^2\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}}\sum_{i'\in\mathcal{S}^{(t)}} q_i q_{k'}\sum_{k=0}^{K-1}\sum_{k'=0}^{K-1}\langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k')}) - \nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k')}, \mathcal{B}_{i'}^{(t,j')})\rangle\right] \tag{A.54}$$

$$\begin{aligned} &= -2(\eta_c^{(t)})^2\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i^2\sum_{k=0}^{K-1}\sum_{k'=0}^{K-1}\langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')})\rangle\right] \\ &\quad - 2(\eta_c^{(t)})^2\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}}\sum_{\substack{i'\in\mathcal{S}^{(t)}\\ i'\neq i}} q_i q_{k'}\sum_{k=0}^{K-1}\sum_{k'=0}^{K-1}\langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k')}) - \nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k')}, \mathcal{B}_{i'}^{(t,j')})\rangle\right] \end{aligned} \tag{A.55}$$

$$\begin{aligned} &= -2(\eta_c^{(t)})^2\sum_{i\in\mathcal{S}^{(t)}} q_i^2\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{k=0}^{K-1}\sum_{k'=0}^{K-1}\langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')})\rangle\right] \\ &\quad - 2(\eta_c^{(t)})^2\sum_{i\in\mathcal{S}^{(t)}}\sum_{\substack{i'\in\mathcal{S}^{(t)}\\ i'\neq i}} q_i q_{k'}\sum_{k=0}^{K-1}\sum_{k'=0}^{K-1}\langle \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right], \\ &\qquad \mathbb{E}_{\mathcal{B}_{i'}^{(t,0:k'-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\underbrace{\left[\mathbb{E}_{\mathcal{B}_{i'}^{(t,j')}|\mathcal{B}_{i'}^{(t,0:k'-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k')}) - \nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k')}, \mathcal{B}_{i'}^{(t,j')})\right]\right]}_{=0}\rangle, \end{aligned} \tag{A.56}$$

where, in (A.54), we replaced the definitions of $g_t$ and $\bar{\boldsymbol{g}}^{(t)}$ given in (A.19) and in (A.20), respectively; in (A.55), we consider the cases $k = k'$ and $k \neq k'$ separately; (A.56) follows from the consideration that local models of different clients evolve independently and then all the terms with $k' \neq k$ equal zero because $\nabla F_i(\boldsymbol{w}, \mathcal{B})$ is an unbiased estimator of $\nabla F_i(\boldsymbol{w})$. It follows that:

$$-2(\eta_c^{(t)})^2\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\langle \bar{\boldsymbol{g}}^{(t)}, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)}\rangle\right] = \tag{A.57}$$

$$= -2(\eta_c^{(t)})^2\sum_{i\in\mathcal{S}^{(t)}} q_i^2\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{k=0}^{K-1}\sum_{k'=0}^{K-1}\langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')})\rangle\right] \tag{A.58}$$

$$= -2(\eta_c^{(t)})^2\sum_{i\in\mathcal{S}^{(t)}} q_i^2\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{k=0}^{K-1}\sum_{\substack{k'=0\\ k'<k}}^{K-1}\langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')})\rangle\right]$$

$$- 2(\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[ \sum_{k=0}^{K-1} \sum_{\substack{k'=0 \\ k'\geq k}}^{K-1} \langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) \rangle \right] \tag{A.59}$$

$$= -2(\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=0}^{K-1} \sum_{\substack{k'=0 \\ k'<k}}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[ \langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) \rangle \right]$$

$$- 2(\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=0}^{K-1} \sum_{\substack{k'=0 \\ k'\geq k}}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t,0:k'-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \times$$

$$\times \left[ \mathbb{E}_{\mathcal{B}_i^{(t,k')}|\mathcal{B}_i^{(t,0:k'-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[ \langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) \rangle \right] \right] \tag{A.60}$$

$$= -2(\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=0}^{K-1} \sum_{\substack{k'=0 \\ k'<k}}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[ \langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) \rangle \right]$$

$$- 2(\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=0}^{K-1} \sum_{\substack{k'=0 \\ k'\geq k}}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t,0:k'-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \times$$

$$\times \left[ \langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \underbrace{\mathbb{E}_{\mathcal{B}_i^{(t,k')}|\mathcal{B}_i^{(t,0:k'-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[ \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) \right]}_{=0} \rangle \right], \tag{A.61}$$

where, in (A.59), we consider the cases $k' < k$ and $k' \geq k$ separately; then, in (A.60) and in (A.61), we use the law of total expectation.

Finally, we bound the remaining term in the right-hand side of (A.61) as follows:

$$- 2(\eta_c^{(t)})^2 \mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[ \langle \bar{\boldsymbol{g}}^{(t)}, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)} \rangle \right] = \tag{A.62}$$

$$= -2(\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=1}^{K-1} \sum_{k'<k} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) \rangle \tag{A.63}$$

$$= (\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=1}^{K-1} \sum_{k'<k} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[ \left\| \nabla F_i(\boldsymbol{w}_i^{(t,k)}) \right\|^2 + \left\| \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) \right\|^2 \right] \tag{A.64}$$

$$= (\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=1}^{K-1} \sum_{k'<k} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[ \left\| \nabla F_i(\boldsymbol{w}_i^{(t,k)}) \right\| \right] +$$

$$+ (\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=1}^{K-1} \sum_{k'<k} \mathbb{E}_{\mathcal{B}_i^{(t,0:j'-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \underbrace{\left[ \mathbb{E}_{\mathcal{B}_i^{(t,k')}|\mathcal{B}_i^{(t,0:j'-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\| \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) \right\|^2 \right]}_{\text{bounded with Assumption 5}} \tag{A.65}$$

$$\leq (\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=1}^{K-1} \sum_{k'<k} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\| \nabla F_i(\boldsymbol{w}_i^{(t,k)}) \right\|^2 + \frac{1}{2} (\eta_c^{(t)})^2 K(K-1) \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sigma_i^2 \tag{A.66}$$

$$\leq (\eta_c^{(t)})^2 L(K-1) \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=1}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[\left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i^*\right)\right] + \frac{1}{2}(\eta_c^{(t)})^2 K(K-1) \sum_{i\in\mathcal{S}^{(t)}} q_i^2\sigma_i^2 \tag{A.67}$$

$$= (\eta_c^{(t)})^2 L(K-1) \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=1}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[\left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right)\right]$$
$$+ (\eta_c^{(t)})^2 LK(K-1) \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right) + \frac{1}{2}(\eta_c^{(t)})^2 K(K-1) \sum_{i\in\mathcal{S}^{(t)}} q_i^2\sigma_i^2 \tag{A.68}$$

$$\leq (\eta_c^{(t)})^2 L(K-1)Q \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=1}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[\left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right)\right]$$
$$+ (\eta_c^{(t)})^2 LK(K-1)Q \underbrace{\sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right)}_{\text{bounded in Lemma B.10}} + \frac{1}{2}(\eta_c^{(t)})^2 K(K-1) \sum_{i\in\mathcal{S}^{(t)}} q_i^2\sigma_i^2, \tag{A.69}$$

where, in (A.64), we used $|\langle \boldsymbol{a}, \boldsymbol{b}\rangle| \leq \frac{1}{2}\|\boldsymbol{a}\|^2 + \frac{1}{2}\|\boldsymbol{b}\|^2$; in (A.66), we applied Assumption 5; in (A.67), we used the $L$-smoothness of $\{F_i(\boldsymbol{w})\}_{i\in\mathcal{N}}$; in (A.68), we added and subtracted $F_i(\boldsymbol{w}_B^*)$ from the sum; finally, in (A.69), we used $\sum_{i\in\mathcal{S}^{(t)}} q_i^2 f(i) \leq (\sum_{i\in\mathcal{S}^{(t)}} q_i)(\sum_{i\in\mathcal{S}^{(t)}} q_i f(i))$ and $\sum_{i\in\mathcal{S}^{(t)}} q_i \leq \sum_{i=1}^N q_i \triangleq Q$. Noting that $K-1 < 2K$ concludes the proof of Lemma B.6.

□

**Lemma B.7** (Bound on the variance of the stochastic gradients)**.** *Let Assumption 5 hold. Similarly to (X. Li et al., 2020, Lemma 2), we have:*

$$(\eta_c^{(t)})^2 \mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\|\boldsymbol{g}^{(t)} - \bar{\boldsymbol{g}}^{(t)}\right\|^2 \leq (\eta_c^{(t)})^2 E \sum_{i\in\mathcal{S}^{(t)}} q_i^2\sigma_i^2. \tag{A.70}$$

*Proof of Lemma B.7.*

$$\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\|\boldsymbol{g}^{(t)} - \bar{\boldsymbol{g}}^{(t)}\right\|^2 = \tag{A.71}$$

$$= \mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\| \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left(\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) - \nabla F_i(\boldsymbol{w}_i^{(t,k)})\right) \right\|^2 \tag{A.72}$$

$$= \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \sum_{k=0}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\|\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) - \nabla F_i(\boldsymbol{w}_i^{(t,k)})\right\|^2$$
$$+ \sum_{i\in\mathcal{S}^{(t)}} q_i^2 \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[\sum_{k=0}^{K-1} \sum_{\substack{k'=0\\k'\neq k}}^{K-1} \langle \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) - \nabla F_i(\boldsymbol{w}_i^{(t,k)}), \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')})\rangle\right]$$
$$+ \sum_{i\in\mathcal{S}^{(t)}} \sum_{\substack{i'\in\mathcal{S}^{(t)}\\i'\neq i}} q_i q_{k'} \sum_{k=0}^{K-1} \langle \mathbb{E}_{\mathcal{B}_i^{(t,0:k-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \underbrace{\left[\mathbb{E}_{\mathcal{B}_i^{(t,k)}|\mathcal{B}_i^{(t,0:k-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left[\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) - \nabla F_i(\boldsymbol{w}_i^{(t,k)})\right]\right]}_{=0},$$

$$\mathbb{E}_{\mathcal{B}_{i'}^{(t,0:k-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\underbrace{\left[\mathbb{E}_{\mathcal{B}_{i'}^{(t,k)}|\mathcal{B}_{i'}^{(t,0:k-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k)},\mathcal{B}_{i'}^{(t,k)})-\nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k)})\right]\right]}_{=0}\rangle$$

$$+\sum_{i\in\mathcal{S}^{(t)}}\sum_{\substack{i'\in\mathcal{S}^{(t)}\\ i'\neq i}}q_iq_{k'}\sum_{k=0}^{K-1}\sum_{\substack{k'=0\\ k'\neq k}}^{K-1}\langle\mathbb{E}_{\mathcal{B}_i^{(t,0:k-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\underbrace{\left[\mathbb{E}_{\mathcal{B}_i^{(t,k)}|\mathcal{B}_i^{(t,0:k-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})-\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right]\right]}_{=0},$$

$$\mathbb{E}_{\mathcal{B}_{i'}^{(t,0:k'-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\underbrace{\left[\mathbb{E}_{\mathcal{B}_{i'}^{(t,j')}|\mathcal{B}_{i'}^{(t,0:k'-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k')},\mathcal{B}_{i'}^{(t,j')})-\nabla F_{i'}(\boldsymbol{w}_{i'}^{(t,k')})\right]\right]}_{=0}\rangle \tag{A.73}$$

$$=\sum_{i\in\mathcal{S}^{(t)}}q_i^2\sum_{k=0}^{K-1}\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t,k)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})-\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right\|^2}_{\text{bounded with Assumption 5}}$$

$$+\sum_{i\in\mathcal{S}^{(t)}}q_i^2\sum_{k=0}^{K-1}\sum_{\substack{k'=0\\ k'<k}}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t,0:k-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\mathbb{E}_{\mathcal{B}_i^{(t,k)}|\mathcal{B}_i^{(t,0:k-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\langle\nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})-\nabla F_i(\boldsymbol{w}_i^{(t,k)}),\right.\right.$$

$$\left.\left.\nabla F_i(\boldsymbol{w}_i^{(t,k')},\mathcal{B}_i^{(t,k')})-\nabla F_i(\boldsymbol{w}_i^{(t,k')})\rangle\right]\right]$$

$$+\sum_{i\in\mathcal{S}^{(t)}}q_i^2\sum_{k=0}^{K-1}\sum_{\substack{k'=0\\ j'>j}}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t,0:k'-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\mathbb{E}_{\mathcal{B}_i^{(t,k')}|\mathcal{B}_i^{(t,0:j'-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\langle\nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})-\nabla F_i(\boldsymbol{w}_i^{(t,k)}),\right.\right.$$

$$\left.\left.\nabla F_i(\boldsymbol{w}_i^{(t,k')},\mathcal{B}_i^{(t,k')})-\nabla F_i(\boldsymbol{w}_i^{(t,k')})\rangle\right]\right] \tag{A.74}$$

$$=\sum_{i\in\mathcal{S}^{(t)}}q_i^2\sum_{k=0}^{K-1}\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t,k)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})-\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right\|^2}_{\text{bounded with Assumption 5}}$$

$$+\sum_{i\in\mathcal{S}^{(t)}}q_i^2\sum_{k=0}^{K-1}\sum_{\substack{k'=0\\ k'<k}}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t,0:k-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\langle\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t,k)}|\mathcal{B}_i^{(t,0:k-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})-\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right]}_{=0},\right.$$

$$\left.\nabla F_i(\boldsymbol{w}_i^{(t,k')},\mathcal{B}_i^{(t,k')})-\nabla F_i(\boldsymbol{w}_i^{(t,k')})\rangle\right]$$

$$+\sum_{i\in\mathcal{S}^{(t)}}q_i^2\sum_{k=0}^{K-1}\sum_{\substack{k'=0\\ j'>j}}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t,0:k'-1)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\langle\nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})-\nabla F_i(\boldsymbol{w}_i^{(t,k)}),\right.$$

$$\left.\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t,k')}|\mathcal{B}_i^{(t,0:j'-1)},\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,k')},\mathcal{B}_i^{(t,k')})-\nabla F_i(\boldsymbol{w}_i^{(t,k')})\right]}_{=0}\rangle\right] \tag{A.75}$$

$$\leq E\sum_{i\in\mathcal{S}^{(t)}}q_i^2\sigma_i^2, \tag{A.76}$$

where, in (A.73), (A.74), and (A.75), we used the law of total expectation; in (A.76), we applied Assumption 5. Multiplying both sides of (A.76) by $(\eta_c^{(t)})^2$ completes the proof of Lemma B.7.

□

**Lemma B.8.** *Let Assumption 3 hold and let the local functions $\{F_i\}_{i=1}^N$ be convex. Define $\gamma_t \triangleq 2\eta_c^{(t)}(1-\eta_c^{(t)}L(1+2EQ))$.*
*For a diminishing step-size $0 < \eta_c^{(t)} \leq \frac{1}{2L(1+2KQ)}$, satisfying $\gamma_t > 0$, we have:*

$$-\gamma_t \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right) \leq -\frac{1}{2}\eta_c^{(t)} E \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}^{(t,0)}) - F_i(\boldsymbol{w}_B^*)\right)$$
$$+ \underbrace{\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2}_{\text{bounded in Lemma B.9}} + 2(\eta_c^{(t)})^2 LK \underbrace{\sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right)}_{\text{bounded in Lemma B.10}}, \tag{A.77}$$

*Proof of Lemma B.8.*

In the following, we require $\gamma_t > 0$.

$$-\gamma_t \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right) \tag{A.78}$$
$$= -\gamma_t \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}^{(t,0)})\right) - \gamma_t \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left(F_i(\boldsymbol{w}^{(t,0)}) - F_i(\boldsymbol{w}_B^*)\right) \tag{A.79}$$
$$\leq -\gamma_t \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \langle \nabla F_i(\boldsymbol{w}^{(t,0)}), \boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\rangle - \gamma_t E \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}^{(t,0)}) - F_i(\boldsymbol{w}_B^*)\right) \tag{A.80}$$
$$\leq \gamma_t \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \frac{1}{2}\left[\eta_c^{(t)} \left\|\nabla F_i(\boldsymbol{w}^{(t,0)})\right\|^2 + \frac{1}{\eta_c^{(t)}} \left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2\right]$$
$$- \gamma_t E \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}^{(t,0)}) - F_i(\boldsymbol{w}_B^*)\right) \tag{A.81}$$
$$\leq \gamma_t \eta_c^{(t)} LK \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}^{(t,0)}) - F_i^*\right) + \frac{\gamma_t}{2\eta_c^{(t)}} \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2$$
$$- \gamma_t E \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}^{(t,0)}) - F_i(\boldsymbol{w}_B^*)\right) \tag{A.82}$$
$$\leq -\gamma_t E(1-\eta_c^{(t)}L) \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}^{(t,0)}) - F_i(\boldsymbol{w}_B^*)\right) + \frac{\gamma_t}{2\eta_c^{(t)}} \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2$$
$$+ \gamma_t \eta_c^{(t)} LK \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right) \tag{A.83}$$

where, in (A.79), we added and subtracted $F_i(\boldsymbol{w}^{(t,0)})$ to the sum; in (A.80), we used the convexity of $\{F_i(\boldsymbol{w})\}_{i\in\mathcal{N}}$; note that (A.80) also requires $\gamma_t > 0$; in (A.81), we used the inequality $|\langle \boldsymbol{a}, \boldsymbol{b}\rangle| \leq \frac{1}{2}\|\boldsymbol{a}\|^2 + \frac{1}{2}\|\boldsymbol{b}\|^2$; in (A.82), we applied the $L$-smoothness of $\{F_i(\boldsymbol{w})\}_{i\in\mathcal{N}}$ (Assumption 3); finally, in (A.83), we added and subtracted $F_i(\boldsymbol{w}_B^*)$ to the sum.

In particular, for $\gamma_t \triangleq 2\eta_c^{(t)}(1-\eta_c^{(t)}L(1+2EQ)) > 0$, since $0 < \eta_c^{(t)} \leq \frac{1}{2L(1+2KQ)}$, we further obtain:

$$
\begin{aligned}
&-\gamma_t \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right) \\
&\quad \leq -\frac{1}{2}\eta_c^{(t)} E \sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}^{(t,0)}) - F_i(\boldsymbol{w}_B^*)\right) \\
&\qquad + \underbrace{\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2}_{\text{bounded in Lemma B.9}} + 2(\eta_c^{(t)})^2 LK \underbrace{\sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right)}_{\text{bounded in Lemma B.10}},
\end{aligned} \tag{A.84}
$$

where, in (A.84), we used $0 < \eta_c^{(t)} \leq \frac{1}{2L(1+2KQ)}$, which gives $-\gamma_t E(1-\eta_c^{(t)}L) = -2\eta_c^{(t)}E\,(1-\eta_c^{(t)}L(1+2KQ))\,(1-\eta_c^{(t)}L) \leq -\frac{1}{2}\eta_c^{(t)}E$. Moreover, since $\gamma_t \leq 2\eta_c^{(t)}$, we also used $\gamma_t\eta_c^{(t)} \leq 2(\eta_c^{(t)})^2$, and $\frac{\gamma_t}{2\eta_c^{(t)}} \leq 1$.

□

**Lemma B.9** (Bound on the divergence of local models)**.** *Let Assumption 2, 3, and 5 hold, the local functions $\{F_i\}_{i=1}^N$ be convex and $G$ be defined as in Lemma B.1, Equation (2.7). Similarly to (X. Li et al., 2020, Lemma 3), we obtain the following inequality:*

$$
\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2\right] \leq \frac{1}{2}(\eta_c^{(t)})^2 E^3 G^2 \left(\sum_{i\in\mathcal{S}^{(t)}} q_i\right). \tag{A.85}
$$

*Proof of Lemma B.9.*

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2\right] \\
&= \mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=1}^{K-1} (\eta_c^{(t)})^2 \left\|\sum_{k'=0}^{j-1} \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')})\right\|^2\right] && \text{(A.86)} \\
&\leq (\eta_c^{(t)})^2 \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=1}^{K-1} j \sum_{k'=0}^{j-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\left\|\nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')})\right\|^2\right] && \text{(A.87)} \\
&\leq (\eta_c^{(t)})^2 G^2 \left(\sum_{k=1}^{K-1} j^2\right)\left(\sum_{i\in\mathcal{S}^{(t)}} q_i\right) && \text{(A.88)} \\
&= \frac{1}{6}(\eta_c^{(t)})^2 K(K-1)(2K-1)G^2 \left(\sum_{i\in\mathcal{S}^{(t)}} q_i\right), && \text{(A.89)}
\end{aligned}
$$

where, in (A.87), we used the triangle and the Jensen's inequalities; in (A.88), we applied the bound in Lemma 2.3.1, Equation (2.7); finally, in (A.89), we developed the sum of sequence of squares $\sum_{k=1}^{K-1} j^2 = \frac{1}{6}K(K-1)(2K-1) \leq \frac{1}{2}E^3$ since $E \geq 1$.

□

**Lemma B.10** (Bound on the dissimilarity of local functions)**.** *Let Assumption 1 hold and* $\left(\mathcal{S}^{(t)}\right)_{t\geq 0}$ *defined therein. We have:*

$$\mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\left(F_i(\boldsymbol{w}_B^*)-F_i^*\right)\right] \leq \left(\sum_{i=1}^N \pi_i q_i\right)\Gamma, \tag{A.90}$$

*where* $\Gamma$ *is defined in* (2.9)*.*

*Proof of Lemma B.10.*

$$\mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\left(F_i(\boldsymbol{w}_B^*)-F_i^*\right)\right] = \sum_{i=1}^N \pi_i q_i\left(F_i(\boldsymbol{w}_B^*)-F_i^*\right) \tag{A.91}$$

$$= \left(\sum_{k'=1}^N \pi_{k'}q_{k'}\right)\sum_{i=1}^N p_i\left(F_i(\boldsymbol{w}_B^*)-F_i^*\right) \tag{A.92}$$

$$\leq \left(\sum_{k'=1}^N \pi_{k'}q_{k'}\right)\sum_{i=1}^N p_i\left(F_i(\boldsymbol{w}^*)-F_i^*\right) \tag{A.93}$$

$$\leq \left(\sum_{k'=1}^N \pi_{k'}q_{k'}\right)\underbrace{\max_{i\in\mathcal{N}}\{(F_i(\boldsymbol{w}^*)-F_i^*)\}}_{\triangleq\Gamma} = \left(\sum_{i=1}^N \pi_i q_i\right)\Gamma, \tag{A.94}$$

where, in (A.91), we solved the total expectation, observing that $\mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i f(i)\right] = \sum_{i=1}^N \pi_i q_i f(i)$ (Assumption 1); in (A.92), we applied $p_i \triangleq \frac{\pi_i q_i}{\sum_{k'=1}^N \pi_{k'}q_{k'}}$; in (A.93), we used $F_B(\boldsymbol{w}) \triangleq \sum_{i=1}^N p_i F_i(\boldsymbol{w})$ and we observed $F_B(\boldsymbol{w}_B^*) \leq F_B(\boldsymbol{w}^*)$; finally, in (A.94), we used $\sum_{i=1}^N p_i = 1$ and $\Gamma \triangleq \max_{i\in\mathcal{N}}\{(F_i(\boldsymbol{w}^*)-F_i^*)\}$.

□

**Lemma B.11** (Convergence results under heterogeneous client availability)**.** *Let Assumptions 1–3 and 5 hold and the functions* $\{F_i\}_{i=1}^N$ *be convex. For a diminishing step-size* $0 < \eta_c^{(t)} \leq \frac{1}{2L(1+2KQ)}$ *satisfying* $\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2 < +\infty$*, for any* $t_0 \leq T$*, we have:*

$$\sum_{t=t_0}^T \eta_c^{(t)}\mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\left(F_i(\boldsymbol{w}^{(t,0)})-F_i(\boldsymbol{w}_B^*)\right)\right] \leq \frac{2}{E}\operatorname{diam}(W)^2 + (E+1)\left(\sum_{i=1}^N \pi_i q_i^2\sigma_i^2\right)\left(\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right)$$
$$+ 2E^2G^2\left(\sum_{i=1}^N \pi_i q_i\right)\left(\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right)$$
$$+ 4L(1+KQ)\Gamma\left(\sum_{i=1}^N \pi_i q_i\right)\left(\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right)$$
$$:= C_0 < +\infty. \tag{A.95}$$

*Proof of Lemma B.11.*

We take expectation over $\mathcal{B}^{(t)} \mid \mathcal{S}^{(t)}, \mathcal{H}^{(t)}$ on Lemma B.3:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\|\boldsymbol{w}^{(t+1,0)} - \boldsymbol{w}_B^*\right\|^2 \leq \\
&\left\|\boldsymbol{w}^{(t,0)} - \boldsymbol{w}_B^*\right\|^2 \underbrace{-2\eta_c^{(t)}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\langle \boldsymbol{w}^{(t,0)} - \boldsymbol{w}_B^*, \bar{\boldsymbol{g}}^{(t)}\rangle}_{\text{bounded in Lemma B.4}} + \underbrace{(\eta_c^{(t)})^2\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\|\bar{\boldsymbol{g}}^{(t)}\right\|^2}_{\text{bounded in Lemma B.5}} \\
&\underbrace{+ 2\eta_c^{(t)}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\langle \boldsymbol{w}^{(T,0)} - \boldsymbol{w}_B^* - \eta_c^{(t)}\bar{\boldsymbol{g}}^{(t)}, \bar{\boldsymbol{g}}^{(t)} - \boldsymbol{g}^{(t)}\rangle}_{\text{bounded in Lemma B.6}} + \underbrace{(\eta_c^{(t)})^2\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\|\boldsymbol{g}^{(t)} - \bar{\boldsymbol{g}}^{(t)}\right\|^2}_{\text{bounded in Lemma B.7}}.
\end{aligned} \tag{A.96}$$

Replacing Lemmas B.4–B.7 in (A.96), we obtain:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\|\boldsymbol{w}^{(t+1,0)} - \boldsymbol{w}_B^*\right\|^2 \\
&\leq \left\|\boldsymbol{w}^{(t,0)} - \boldsymbol{w}_B^*\right\|^2 + 2(\eta_c^{(t)})^2 LK(1+2KQ)\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i \left(F_i(\boldsymbol{w}_B^*) - F_i^*\right)\right] \\
&\underbrace{- \underbrace{2\eta_c^{(t)}(1-\eta_c^{(t)}L(1+2EQ))}_{\gamma_t}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=1}^{K-1}\left(F_i(\boldsymbol{w}_i^{(t,k)}) - F_i(\boldsymbol{w}_B^*)\right)\right]}_{\text{bounded in Lemma B.8}} \\
&+ \frac{1}{2}(\eta_c^{(t)})^2 E(E+1)\sum_{i\in\mathcal{S}^{(t)}} q_i^2\sigma_i^2 + \underbrace{\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1}\left\|\boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t,0)}\right\|^2\right]}_{\text{bounded in Lemma B.9}}
\end{aligned} \tag{A.97}$$

We apply Lemmas B.8 and B.9 to (A.97) with $\gamma_t \triangleq 2\eta_c^{(t)}(1-\eta_c^{(t)}L(1+2EQ))$. We observe that $\gamma_t > 0$ because:

$$0 \leq \eta_c^{(t)} \leq \frac{1}{2L(1+2KQ)}. \tag{A.98}$$

We obtain:

$$\begin{aligned}
\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}} \left\|\boldsymbol{w}^{(t+1,0)} - \boldsymbol{w}_B^*\right\|^2 \leq \left\|\boldsymbol{w}^{(t,0)} - \boldsymbol{w}_B^*\right\|^2 &- \frac{1}{2}\eta_c^{(t)} E\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{S}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\left(F_i(\boldsymbol{w}^{(t,0)}) - F_i(\boldsymbol{w}_B^*)\right)\right] \\
&+ \frac{1}{2}(\eta_c^{(t)})^2 E(E+1)\sum_{i\in\mathcal{S}^{(t)}} q_i^2\sigma_i^2 + (\eta_c^{(t)})^2 E^3 G^2 \sum_{i\in\mathcal{S}^{(t)}} q_i \\
&+ 4(\eta_c^{(t)})^2 LK(1+KQ)\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\left(F_i(\boldsymbol{w}_B^*) - F_i^*\right)\right].
\end{aligned} \tag{A.99}$$

Computing the total expectation on (A.99), we have:

$$\mathbb{E}_{\mathcal{S}^{(t)},\mathcal{B}^{(t)},\mathcal{H}^{(t)}} \left\|\boldsymbol{w}^{(t+1,0)} - \boldsymbol{w}_B^*\right\|^2$$

$$\leq \mathbb{E}_{\mathcal{H}_t}\left\|\boldsymbol{w}^{(t,0)}-\boldsymbol{w}_B^*\right\|^2-\frac{1}{2}\eta_c^{(t)}E\mathbb{E}_{\mathcal{S}^{(t)},\mathcal{B}^{(t)},\mathcal{H}^{(t)}}\left[\sum_{i\in\mathcal{S}^{(t)}}q_i\left(F_i(\boldsymbol{w}^{(t,0)})-F_i(\boldsymbol{w}_B^*)\right)\right]$$
$$+\frac{1}{2}(\eta_c^{(t)})^2E(E+1)\mathbb{E}_{\mathcal{S}^{(t)},\mathcal{H}_t}\left[\sum_{i\in\mathcal{S}^{(t)}}q_i^2\sigma_i^2\right]+(\eta_c^{(t)})^2E^3G^2\mathbb{E}_{\mathcal{S}^{(t)},\mathcal{H}_t}\left[\sum_{i\in\mathcal{S}^{(t)}}q_i\right]$$
$$+4(\eta_c^{(t)})^2LK(1+KQ)\underbrace{\mathbb{E}_{\mathcal{S}^{(t)},\mathcal{H}_t}\left[\sum_{i\in\mathcal{S}^{(t)}}q_i\left(F_i(\boldsymbol{w}_B^*)-F_i^*\right)\right]}_{\text{bounded in Lemma B.10}} \tag{A.100}$$

Applying Lemma B.10 to (A.100) and considering $\mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}}a_i\right]=\sum_{i=1}^N\pi_ia_i$ (Assumption 1), the following inequality holds:

$$\mathbb{E}\left\|\boldsymbol{w}^{(t+1,0)}-\boldsymbol{w}_B^*\right\|^2\leq\mathbb{E}\left\|\boldsymbol{w}^{(t,0)}-\boldsymbol{w}_B^*\right\|^2-\frac{1}{2}\eta_c^{(t)}E\mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}}q_i\left(F_i(\boldsymbol{w}^{(t,0)})-F_i(\boldsymbol{w}_B^*)\right)\right]$$
$$+\frac{1}{2}(\eta_c^{(t)})^2E(E+1)\left(\sum_{i=1}^N\pi_iq_i^2\sigma_i^2\right)+(\eta_c^{(t)})^2E^3G^2\left(\sum_{i=1}^N\pi_iq_i\right)$$
$$+4(\eta_c^{(t)})^2LK(1+KQ)\Gamma\left(\sum_{i=1}^N\pi_iq_i\right). \tag{A.101}$$

Rearranging and summing over $t=t_0,\ldots,T$, we obtain the following inequality:

$$\sum_{t=t_0}^T\eta_c^{(t)}\mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}}q_i(F_i(\boldsymbol{w}^{(t,0)})-F_i(\boldsymbol{w}_B^*))\right]\leq\frac{2}{E}\sum_{t=t_0}^T\mathbb{E}\left[\left(\left\|\boldsymbol{w}^{(t,0)}-\boldsymbol{w}_B^*\right\|^2-\left\|\boldsymbol{w}^{(t+1,0)}-\boldsymbol{w}_B^*\right\|^2\right)\right]$$
$$+(E+1)\left(\sum_{i=1}^N\pi_iq_i^2\sigma_i^2\right)\left(\sum_{t=t_0}^T(\eta_c^{(t)})^2\right)$$
$$+2E^2G^2\left(\sum_{i=1}^N\pi_iq_i\right)\left(\sum_{t=t_0}^T(\eta_c^{(t)})^2\right)$$
$$+4L(1+KQ)\Gamma\left(\sum_{i=1}^N\pi_iq_i\right)\left(\sum_{t=t_0}^T(\eta_c^{(t)})^2\right). \tag{A.102}$$

The first term in the right-hand side of (A.102) is a telescoping sum and we remove the negative term $-\mathbb{E}\left\|\boldsymbol{w}^{(t+1,0)}-\boldsymbol{w}_B^*\right\|^2$:

$$\sum_{t=t_0}^T\eta_c^{(t)}\mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}}q_i(F_i(\boldsymbol{w}^{(t,0)})-F_i(\boldsymbol{w}_B^*))\right]\leq\frac{2}{E}\mathbb{E}\left\|\boldsymbol{w}_{t_0,0}-\boldsymbol{w}_B^*\right\|^2+(E+1)\left(\sum_{i=1}^N\pi_iq_i^2\sigma_i^2\right)\left(\sum_{t=t_0}^T(\eta_c^{(t)})^2\right)$$
$$+2E^2G^2\left(\sum_{i=1}^N\pi_iq_i\right)\left(\sum_{t=t_0}^T(\eta_c^{(t)})^2\right)$$

$$+ 4L(1+KQ)\Gamma \left(\sum_{i=1}^{N} \pi_i q_i\right) \left(\sum_{t=t_0}^{T} (\eta_c^{(t)})^2\right). \tag{A.103}$$

Finally, by noting that $\|\boldsymbol{w}_{t_0,0} - \boldsymbol{w}_B^*\| \leq \operatorname{diam}(W)$ and $\sum_{t=t_0}^{T} (\eta_c^{(t)})^2 \leq \sum_{t=1}^{+\infty} (\eta_c^{(t)})^2 < +\infty$, we complete the proof of Lemma B.11.

□

**Lemma B.12.** *Let Assumptions 2 and 3 hold, and the local functions $\{F_i\}_{i=1}^N$ be convex. We have:*

$$|F_i(\boldsymbol{v}) - F_i(\boldsymbol{w})| \leq D \cdot \|\boldsymbol{v} - \boldsymbol{w}\|, \ \forall \boldsymbol{v}, \boldsymbol{w} \in W \tag{A.104}$$

*Proof of Lemma B.12.*

In Lemma B.1, under Assumptions 2 and 3, we have already proved that:

$$\|\nabla F_i(\boldsymbol{w})\| \leq D. \tag{2.6}$$

Moreover, from the convexity of $\{F_i\}_{i\in\mathcal{N}}$, it follows that:

$$\langle \nabla F_i(\boldsymbol{v}), \boldsymbol{v} - \boldsymbol{w} \rangle \leq F_i(\boldsymbol{v}) - F_i(\boldsymbol{w}) \leq \langle \nabla F_i(\boldsymbol{w}), \boldsymbol{v} - \boldsymbol{w} \rangle. \tag{A.105}$$

The Cauchy–Schwarz inequality completes the proof of Lemma B.12:

$$|F_i(\boldsymbol{v}) - F_i(\boldsymbol{w})| \leq \max\{\|\nabla F_i(\boldsymbol{v})\|, \|\nabla F_i(\boldsymbol{w})\|\} \cdot \|\boldsymbol{v} - \boldsymbol{w}\| \leq D \cdot \|\boldsymbol{v} - \boldsymbol{w}\|. \tag{A.106}$$

□

**Lemma B.13.** *Let Assumptions 2, 3, and 5 hold. We have:*

$$\mathbb{E}_{\mathcal{B}_t|\mathcal{S}^{(t)},\mathcal{H}_t} \left\| \boldsymbol{w}^{(t+1,0)} - \boldsymbol{w}^{(t,0)} \right\| \leq \eta_c^{(t)} EG \left( \sum_{i\in\mathcal{S}^{(t)}} q_i \right). \tag{A.107}$$

*Proof of Lemma B.13.*

The proof is based on (Sun et al., 2018, Proposition 1.4).

$$\mathbb{E}_{\mathcal{B}_t|\mathcal{S}^{(t)},\mathcal{H}_t} \left\| \boldsymbol{w}^{(t+1,0)} - \boldsymbol{w}^{(t,0)} \right\| = \mathbb{E}_{\mathcal{B}_t|\mathcal{S}^{(t)},\mathcal{H}_t} \left\| -\eta_c^{(t)} \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) \right\| \tag{A.108}$$

$$\leq \eta_c^{(t)} \sum_{i\in\mathcal{S}^{(t)}} q_i \sum_{k=0}^{K-1} \mathbb{E}_{\mathcal{B}_{t,0:j-1}^k|\mathcal{S}^{(t)},\mathcal{H}_t} \left[ \mathbb{E}_{\mathcal{B}_i^{(t,k)}|\mathcal{B}_{t,0:j-1}^k,\mathcal{S}^{(t)},\mathcal{H}_t} \left[ \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) \right] \right] \tag{A.109}$$

$$\leq \eta_c^{(t)} EG \left( \sum_{i\in\mathcal{S}^{(t)}} q_i \right), \tag{A.110}$$

where, in (A.109), we used the triangle inequality and the law of total expectation; in (A.110), we applied Lemma 2.3.1, Equation (2.7).

□

Similarly to (Sun et al., 2018, Theorem 1), we provide the following definition.

**Definition B.1.** For communication round $t \geq 1$, denote the positive integer $\mathcal{J}^{(t)}$ as follows:

$$\mathcal{J}^{(t)} \triangleq \min\left\{\max\left\{\left\lceil \frac{\ln\left(2C_P H t\right)}{\ln\left(1/\lambda(\boldsymbol{P})\right)}\right\rceil, T_P\right\}, t\right\}. \tag{A.111}$$

The parameter $\mathcal{J}^{(t)}$ is crucial in our analysis: it represents the communication rounds needed to bound the stationary distribution convergence of the Markov process $(\mathcal{S}^{(t)})_{t>0}$. It will play a key role in Lemmas B.14–B.18 and in the proof of Theorem B.19. We remark that, by definition: $T_P \leq \mathcal{J}^{(t)} \leq t$.

Our definition of $\mathcal{J}^{(t)}$ corrects a typo in (Sun et al., 2018, (6.27)), which considered $\ln\left(t/(2C_P H)\right)$ rather than $\ln\left(2C_P H t\right)$. In fact, we observe that (Sun et al., 2018, (6.28)) and consequently (Sun et al., 2018, (6.35)) do not hold when $\mathcal{J}^{(t)}$ is defined as in (Sun et al., 2018, (6.27)).

**Lemma B.14** (Convergence results under heterogeneous and correlated client availability after $\mathcal{J}^{(t)}$ communication rounds)**.** *Let Assumptions 1–3, and 5 hold, the local functions $\{F_i\}_{i=1}^N$ be convex, and the parameter $\mathcal{J}^{(t)} \leq t$ be as in Definition B.1. For a diminishing step-size $\{\eta_c^{(t)}\}_{t\geq 1}$ satisfying $\sum_{t=1}^{+\infty} \ln(t)\cdot(\eta^{(t)})^2$, for any $t_0 \leq T$, we have:*

$$\sum_{t=t_0}^{T} \eta_c^{(t)} \mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\left(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}^{(t,0)})\right)\right] \leq \frac{C_1}{\ln(1/\lambda(\boldsymbol{P}))} < +\infty, \tag{A.112}$$

*where:*

$$C_1 \triangleq EDGQ\left(\sum_{i=1}^{N}\pi_i q_i\right)\left(\sum_{t=1}^{+\infty}\ln\left(2C_P H t\right)\left(\eta^{(t-\mathcal{J}^{(t)})}\right)^2\right). \tag{A.113}$$

*Proof of Lemma B.14.*

This proof is based on (Sun et al., 2018, Equation (6.31)).

$$\begin{aligned}
&\sum_{t=t_0}^{T} \eta_c^{(t)} \mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\left(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}^{(t,0)})\right)\right] \\
&\quad\leq Q\sum_{t=t_0}^{T}\eta_c^{(t)}\mathbb{E}\left[\max_{i\in\mathcal{N}}\left\{F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}^{(t,0)})\right\}\right] &\text{(A.114)}\\
&\quad\leq DQ\sum_{t=t_0}^{T}\eta_c^{(t)}\mathbb{E}\left\|\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)} - \boldsymbol{w}^{(t,0)}\right\| &\text{(A.115)}\\
&\quad\leq DQ\sum_{t=t_0}^{T}\eta_c^{(t)}\sum_{d=t-\mathcal{J}^{(t)}}^{t-1}\mathbb{E}_{\mathcal{S}^{(d)},\mathcal{H}^{(d)}}\left[\mathbb{E}_{\mathcal{B}^{(d)}|\mathcal{S}^{(d)},\mathcal{H}^{(d)}}\left\|\boldsymbol{w}^{(d,0)} - \boldsymbol{w}^{(d+1,0)}\right\|\right] &\text{(A.116)}\\
&\quad\leq EDGQ\sum_{t=t_0}^{T}\sum_{d=t-\mathcal{J}^{(t)}}^{t-1}\eta_c^{(t)}\eta^{(d)}\mathbb{E}\left[\sum_{k\in\mathcal{S}^{(d)}} q_i\right] &\text{(A.117)}
\end{aligned}$$

$$\leq EDGQ\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\sum_{d=t-\mathcal{J}^{(t)}}^{t-1}\eta_c^{(t)}\eta^{(d)} \tag{A.118}$$

$$\leq \frac{EDGQ}{2}\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\sum_{d=t-\mathcal{J}^{(t)}}^{t-1}\left((\eta_c^{(t)})^2+(\eta^{(d)})^2\right) \tag{A.119}$$

$$\leq EDGQ\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\mathcal{J}^{(t)}(\eta^{(t-\mathcal{J}^{(t)})})^2, \tag{A.120}$$

where, in (A.114), we used $\sum_{i\in\mathcal{S}^{(t)}} q_i a_i \leq \sum_{i=1}^{N} q_i a_i \leq (\sum_{i=1}^{N} q_i)\cdot\max_{i\in\mathcal{N}}\{a_i\} = Q\cdot\max_{i\in\mathcal{N}}\{a_i\}$; in (A.115), we applied Lemma B.12; in (A.116), we used the triangle inequality and the law of total expectation; in (A.117), we applied Lemma B.13 and again the law of total expectation; in (A.118), we observed that $\mathbb{E}\left[\sum_{k\in\mathcal{S}^{(d)}} q_i\right] = \sum_{i=1}^{N}\pi_i q_i$ (Assumption 1); in (A.119), we used $2ab \leq a^2+b^2$; finally, in (A.120), we applied $\eta_c^{(t)} < \eta^{(d)} \leq \eta^{(t-\mathcal{J}^{(t)})}$ due to the diminishing learning rate.

We apply then the definition of $\mathcal{J}^{(t)}$ in (A.111) and we observe that $\sum_{t=t_0}^{T}\ln(t)(\eta^{(t-\mathcal{J}^{(t)})})^2 \leq \sum_{t=1}^{+\infty}\ln(t)(\eta^{(t-\mathcal{J}^{(t)})})^2$:

$$\sum_{t=t_0}^{T}\eta_c^{(t)}\mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)})-F_i(\boldsymbol{w}^{(t,0)}))\right] \leq EDGQ\left(\sum_{i=1}^{N}\pi_i q_i\right)\left(\sum_{t=t_0}^{T}\frac{\ln(2C_P Ht)}{\ln(1/\lambda(\boldsymbol{P}))}(\eta^{(t-\mathcal{J}^{(t)})})^2\right) \tag{A.121}$$

$$\leq EDGQ\left(\sum_{i=1}^{N}\pi_i q_i\right)\left(\sum_{t=1}^{+\infty}\frac{\ln(2C_P Ht)}{\ln(1/\lambda(\boldsymbol{P}))}(\eta^{(t-\mathcal{J}^{(t)})})^2\right) \tag{A.122}$$

$$= \frac{C_1}{\ln(1/\lambda(\boldsymbol{P}))}. \tag{A.123}$$

Finally, we conclude that $C_1$ is finite. To this purpose, we observe that $\mathcal{J}^{(t)} \leq a\ln(t)+b$, for opportune positive values $a$ and $b$. Let $t'$ be a positive integer such that $t \geq a\ln(t)+b$ for any $t \geq t'$. Then:

$$\sum_{t=t'}^{T}\ln(t)\cdot(\eta^{(t-\mathcal{J}^{(t)})})^2 = \sum_{t=t'-\mathcal{J}^{(t)}}^{T-\mathcal{J}^{(t)}}\ln(t+\mathcal{J}^{(t)})\cdot(\eta^{(t)})^2 \tag{A.124}$$

$$\leq \sum_{t=1}^{+\infty}\ln(t+a\ln t+b)\cdot(\eta^{(t)})^2 \tag{A.125}$$

$$\leq \sum_{t=1}^{+\infty}\ln\left((1+a+b)\,t\right)\cdot(\eta^{(t)})^2 < +\infty. \tag{A.126}$$

□

**Lemma B.15.** *Let Assumptions 2, 3 and 5 hold, the local functions $\{F_i\}_{i=1}^{N}$ be convex, and $\mathcal{J}^{(t)} \leq t$ be as in Definition B.1. Let the step-size be decreasing and satisfy: $\sum_{t=1}^{+\infty}\ln(t)\cdot(\eta^{(t)})^2 < +\infty$. For any $t_0 \leq T$, we have:*

$$\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\eta_c^{(t)}\mathbb{E}\left[F_B(\boldsymbol{w}^{(t,0)})-F_B(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)})\right] \leq \frac{C_1}{\ln(1/\lambda(\boldsymbol{P}))} < +\infty, \tag{A.127}$$

*where:*

$$C_1 \triangleq EDGQ\left(\sum_{i=1}^{N}\pi_i q_i\right)\left(\sum_{t=1}^{+\infty}\ln\left(2C_P H t\right)(\eta^{(t-\mathcal{J}^{(t)})})^2\right). \tag{A.128}$$

*Proof of Lemma B.15.*

This proof is based on (Sun et al., 2018, Equation (6.38)).

$$\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\eta_c^{(t)}\mathbb{E}\left[F_B(\boldsymbol{w}^{(t,0)})-F_B(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)})\right]$$

$$=\sum_{t=t_0}^{T}\eta_c^{(t)}\sum_{i=1}^{N}\pi_i q_i\mathbb{E}\left[F_i(\boldsymbol{w}^{(t,0)})-F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)})\right] \tag{A.129}$$

$$\leq D\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\eta_c^{(t)}\mathbb{E}\left\|\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}-\boldsymbol{w}^{(t,0)}\right\| \tag{A.130}$$

$$\leq D\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\eta_c^{(t)}\sum_{d=t-\mathcal{J}^{(t)}}^{t-1}\mathbb{E}_{\mathcal{S}^{(d)},\mathcal{H}^{(d)}}\left[\mathbb{E}_{\mathcal{B}^{(d)}|\mathcal{S}^{(d)},\mathcal{H}^{(d)}}\left\|\boldsymbol{w}^{(d,0)}-\boldsymbol{w}^{(d+1,0)}\right\|\right] \tag{A.131}$$

$$\leq DEGQ\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\sum_{d=t-\mathcal{J}^{(t)}}^{t-1}\eta_c^{(t)}\eta^{(d)} \tag{A.132}$$

$$\leq \frac{DEGQ}{2}\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\sum_{d=t-\mathcal{J}^{(t)}}^{t-1}\left((\eta_c^{(t)})^2+(\eta^{(d)})^2\right) \tag{A.133}$$

$$\leq DEGQ\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=t_0}^{T}\mathcal{J}^{(t)}\cdot(\eta^{(t-\mathcal{J}^{(t)})})^2 \tag{A.134}$$

$$\leq EDGQ\left(\sum_{i=1}^{N}\pi_i q_i\right)\left(\sum_{t=1}^{+\infty}\frac{\ln\left(2C_P H t\right)}{\ln(1/\lambda(\boldsymbol{P}))}(\eta^{(t-\mathcal{J}^{(t)})})^2\right)=\frac{C_1}{\ln(1/\lambda(\boldsymbol{P}))}, \tag{A.135}$$

where, in (A.129), we applied $F_B(\boldsymbol{w})=\sum_{i=1}^{N}p_i F_i(\boldsymbol{w})$, where $p_i=\frac{\pi_i q_i}{\sum_{h=1}^{N}\pi_h q_h}$; in (A.130), we applied Lemma B.12; in (A.131), we applied the triangle inequality and the law of total expectation; in (A.132), we applied Lemma B.13; in (A.133), we used $2ab\leq a^2+b^2$; in (A.134), we observed that $(\eta_c^{(t)})^2+(\eta^{(d)})^2\leq 2(\eta^{(t-\mathcal{J}^{(t)})})^2$ due to the diminishing learning rate; finally, in (A.135), we applied the definition of $\mathcal{J}^{(t)}$ given in (A.111) and we observed that $\sum_{t=t_0}^{T}\ln(t)(\eta^{(t-\mathcal{J}^{(t)})})^2\leq\sum_{t=1}^{+\infty}\ln(t)(\eta^{(t-\mathcal{J}^{(t)})})^2<+\infty$ and then $C_1<+\infty$.

□

**Lemma B.16** (Bound on the distance dynamics between the current and the stationary distributions of the Markov process)**.** *Let Assumption 1 hold, and $\boldsymbol{P}$, $\boldsymbol{\rho}$ defined therein. The following inequality holds:*

$$\max_{i,j\in[M]}\left|[\boldsymbol{P}^t]_{i,j}-\rho_j\right|\leq C_P\cdot\lambda(\boldsymbol{P})^t,\quad \textit{for } t\geq T_P, \tag{5}$$

*where $C_P$ and $T_P$ are positive constants defined as:*

$$C_P\triangleq\left(\sum_{i=2}^{d}n_i^2\right)^{\frac{1}{2}}\cdot\|\boldsymbol{U}\|_F\|\boldsymbol{U}^{-1}\|_F, \tag{A.136}$$

$$T_P \triangleq \max\left\{\max_{1\le i\le d}\left\{\left\lceil\frac{2n_i(n_i-1)(\ln(\frac{2n_i}{\ln\lambda(\boldsymbol{P})/|\bar{\lambda}_2(\boldsymbol{P})|})-1)}{(n_i+1)\ln(\lambda(\boldsymbol{P})/|\bar{\lambda}_2(\boldsymbol{P})|)}\right\rceil\right\},\ 0\right\}. \tag{A.137}$$

*Here, $d$, $n_i$, and $\boldsymbol{U}$ are quantities related to the Jordan canonical form of $\boldsymbol{P}$. Specifically, $\boldsymbol{P}=\boldsymbol{U}\boldsymbol{J}\boldsymbol{U}^{-1}$, where $\boldsymbol{J}$ denotes the Jordan $M\times M$ matrix with $d$ blocks $\boldsymbol{J}_i, i=2,\dots,d$. Each block $\boldsymbol{J}_i$, $i=2,3,\dots,d$, has a dimension $n_i\ge 1$, and $\sum_{i=1}^{d} n_i = M$. Moreover, $|\boldsymbol{U}|_F$ denotes the Frobenius norm of the matrix $\boldsymbol{U}$.*

*Furthermore, let Assumptions 2 and 3 hold, $H$ be defined as in Lemma B.1, Equation (2.8), and $T_P \le \mathcal{J}^{(t)} \le t$ be defined in (A.111). We obtain the additional inequality:*

$$\left|[\boldsymbol{P}^{\mathcal{J}^{(t)}}]_{i,j}-\rho_j\right| \le C_P\cdot\lambda(\boldsymbol{P})^t \le C_P\lambda(\boldsymbol{P})^{\mathcal{J}^{(t)}} = \frac{1}{2Ht}, \quad \forall i,j\in[M] \text{ and } \forall t\ge T_P. \tag{A.138}$$

*Proof of Lemma B.16.*

The inequality in (2.5) is proven in(Sun et al., 2018, Lemma 1) and holds for any $t\ge T_P$. Here, $T_P$ is a constant dependent on the transition matrix $\boldsymbol{P}$ of the Markov chain $(\mathcal{S}^{(t)})_{t\ge 0}$ defined in Assumption 1. To prove (A.138), we further observe that $0<\lambda(\boldsymbol{P})\le 1$ and $T_P\le\mathcal{J}^{(t)}\le t$. The last inequality in (A.138) follows from the definition of $\mathcal{J}^{(t)}$ in (A.111).

□

We remark that the bounds in (Sun et al., 2018, Lemma 1), and consequently our (A.138), require $t\ge T_P$. Therefore, the derivations in (Sun et al., 2018, (6.28)) and (Sun et al., 2018, (6.35)–(6.37)) are not accurate, since they hold for $t\ge T_P$. We address this problem with Lemmas B.17 and B.18.

**Lemma B.17.** *Let Assumptions 1–3 hold, and $T_P$ be defined as in (A.137). The following inequality holds:*

$$\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=1}^{T_P-1}\eta_c^{(t)}\mathbb{E}\left[F_B(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)})-F_B^*\right]\le C_2<+\infty, \tag{A.139}$$

*where:*

$$C_2 \triangleq H\left(\sum_{t=1}^{T_P-1}\eta_c^{(t)}\right)\left(\sum_{i=1}^{N}\pi_i q_i\right)<+\infty. \tag{A.140}$$

*Proof of Lemma B.17.*

$$\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=1}^{T_P-1}\eta_c^{(t)}\mathbb{E}\left[F_B(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)})-F_B^*\right] = \sum_{t=1}^{T_P-1}\eta_c^{(t)}\sum_{i=1}^{N}\pi_i q_i\mathbb{E}\left[F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)})-F_i(\boldsymbol{w}_B^*)\right] \tag{A.141}$$

$$\le H\left(\sum_{t=1}^{T_P-1}\eta_c^{(t)}\right)\left(\sum_{i=1}^{N}\pi_i q_i\right)\triangleq C_2<+\infty, \tag{A.142}$$

where, in (A.141), we used the definition of $F_B$ from (2.4), and in (A.142), we applied Lemma 2.3.1, Equation (2.8), which holds for any $\boldsymbol{w}\in W$. Lastly, it is worth noting that $C_2$ is a sum of finite elements, and is therefore finite.

□

**Lemma B.18.** *Let Assumptions 1–3 and 5 hold, and $\{F_i\}_{i=1}^N$ be convex. Recall the definitions of $\mathcal{J}^{(t)}$ and $T_P$ in (A.111) and in (A.137), respectively. Let the step-size $(\eta_c^{(t)})_{t\geq 1}$ decrease and satisfy $\eta_1 \leq \frac{1}{2L(1+2KQ)}$, $\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2 < +\infty$, and $\sum_{t=1}^{+\infty}\ln(t)\cdot(\eta^{(t)})^2 < +\infty$. For $t \geq T_P$, we have:*

$$\left(\sum_{i=1}^N \pi_i q_i\right) \sum_{t=T_P}^{T} \eta_c^{(t)} \mathbb{E}\left[F_B(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_B^*\right] \leq \frac{C_1}{\ln(1/\lambda(\boldsymbol{P}))} + C_3 < +\infty, \tag{A.143}$$

*where:*

$$C_1 \triangleq EDGQ\left(\sum_{i=1}^N \pi_i q_i\right)\left(\sum_{t=1}^{+\infty} \ln(2C_P H t)\cdot(\eta^{(t-\mathcal{J}^{(t)})})^2\right) < +\infty. \tag{A.144}$$

$$C_3 \triangleq C_0 + \frac{MQ}{4}\sum_{t=1}^{+\infty}\left((\eta_c^{(t)})^2 + \frac{1}{t^2}\right) < +\infty; \tag{A.145}$$

*Proof of Lemma B.18.*

Assume $t \geq T_P$. With a similar proof technique to (Sun et al., 2018, (6.35)), we derive the following lower bound:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}^{(t)}|\mathcal{S}^{(t-\mathcal{J}^{(t)})},\mathcal{H}^{(t-\mathcal{J}^{(t)})}}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\left(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\right)\right] = \\
&= \sum_{a\in\mathcal{M}} \mathbb{P}(\mathcal{S}^{(t)} = a \mid \mathcal{S}^{(t-\mathcal{J}^{(t)})}, \mathcal{H}^{(t-\mathcal{J}^{(t)})}) \sum_{k\in a} q_i\left(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\right) && \text{(A.146)}\\
&= \sum_{a\in\mathcal{M}} \left[\boldsymbol{P}^{\mathcal{J}^{(t)}}\right]_{\mathcal{S}^{t-\mathcal{J}^{(t)}},a} \sum_{k\in a} q_i\left(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\right) && \text{(A.147)}\\
&\geq \sum_{a\in\mathcal{M}} \left(\rho_a - \frac{1}{2Ht}\right) \sum_{k\in a} q_i\left(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\right) && \text{(A.148)}\\
&= \sum_{i=1}^N \mathbb{E}\left[\mathbb{1}_{i\in\mathcal{S}^{(t)}}\right] q_i\left(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\right) - \frac{1}{2Ht}\sum_{a\in\mathcal{M}}\sum_{k\in a} q_i\left(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\right) && \text{(A.149)}\\
&\geq \sum_{i=1}^N \pi_i q_i\left(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\right) - \frac{MQ}{2Ht}\max_{i\in\mathcal{N}}\left\{F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\right\} && \text{(A.150)}\\
&\geq \left(\sum_{i=1}^N \pi_i q_i\right)\cdot\left(F_B(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_B^*\right) - \frac{MQ}{2t}, && \text{(A.151)}
\end{aligned}$$

where, in (A.146), we applied the definition of expected value to the random variable $\mathcal{S}^{(t)}$, with $a$ representing a realization of $\mathcal{S}^{(t)}$, that is a state in the state space $\mathcal{M}$, and $\mathbb{P}(\mathcal{S}^{(t)} = a \mid \mathcal{S}^{(t-\mathcal{J}^{(t)})}, \mathcal{H}^{(t-\mathcal{J}^{(t)})})$ denoting the conditional probability of the event $\mathcal{S}^{(t)} = a$ given $(\mathcal{S}^{(t-\mathcal{J}^{(t)})}, \mathcal{H}^{(t-\mathcal{J}^{(t)})})$; in (A.147), we applied the Markov property (Assumption 1), observing that $\mathbb{P}(\mathcal{S}^{(t)} = a \mid \mathcal{S}^{(t-\mathcal{J}^{(t)})}) = [\boldsymbol{P}^{\mathcal{J}^{(t)}}]_{\mathcal{S}^{t-\mathcal{J}^{(t)}},a}$, where $[\boldsymbol{P}^k]_{i,j}$ denotes the $(i,j)$-th element of the $i$-th power of the transition matrix $\boldsymbol{P}$; in (A.148), we applied Lemma B.16, Equation (A.138); for the first term in (A.149), we used $\sum_{a\in\mathcal{M}}\rho_a\sum_{k\in a} f(i) = \sum_{a\in\mathcal{M}}\rho_a\sum_{i=1}^N \mathbb{1}_{\{k\in a\}} f(i) = \sum_{i=1}^N f(i)\sum_{a\in\mathcal{M}}\rho_a \mathbb{1}_{k\in a} = \sum_{i=1}^N f(i)\mathbb{E}\left[\mathbb{1}_{i\in\mathcal{S}^{(t)}}\right]$, where $\mathbb{1}_{i\in\mathcal{S}^{(t)}}$ is the indicator function that equals 1 if and only

if $i \in \mathcal{S}^{(t)}$; in (A.150), we used $\mathbb{E}\left[\mathbb{1}_{i\in\mathcal{S}^{(t)}}\right] = \mathbb{P}(i \in \mathcal{S}^{(t)}) \triangleq \pi_i$ for the first term, and $\sum_{k\in a} q_i f(i) \leq \sum_{i=1}^{N} q_i f(i) \leq (\sum_{i=1}^{N} q_i)(\max_{i\in\mathcal{N}} f(i)) = Q \max_{i\in\mathcal{N}} f(i)$ and $\sum_{a\in\mathcal{M}} 1 = M$ for the second term; finally, in (A.151), we used the definition of $F_B$ in (2.4) for the first term, and we used Lemma 2.3.1, Equation (2.8) for the second term.

Our derivations in (A.150) and (A.151) correct a typo in (Sun et al., 2018, (6.35)), which considered $Q/(2t)$ instead of $(MQ)/(2t)$. In (A.151), the dimension ($M$) of the state space ($\mathcal{M}$) of the Markov chain $\left(\mathcal{S}^{(t)}\right)_{t\geq 0}$ appears in the numerator of the second term.

Note that the steps in (A.148)–(A.151) require $t \geq T_P$. Multiplying by $\eta_c^{(t)}$ and summing for $t = T_P, \ldots, T$, rearranging, and computing the total expectation, we obtain the following inequality:

$$\begin{aligned}
&\left(\sum_{i=1}^{N} \pi_i q_i\right) \sum_{t=T_P}^{T} \eta_c^{(t)} \mathbb{E}\left[F_B(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_B^*\right] \\
&\leq \sum_{t=T_P}^{T} \eta_c^{(t)} \mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\big(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\big)\right] + \frac{MQ}{2} \sum_{t=T_P}^{T} \frac{\eta_c^{(t)}}{t} \qquad \text{(A.152)} \\
&\leq \underbrace{\sum_{t=T_P}^{T} \eta_c^{(t)} \mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\big(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\big)\right]}_{\text{bounded with Lemma B.11 + Lemma B.14}} + \frac{MQ}{4} \sum_{t=1}^{T} \left((\eta_c^{(t)})^2 + \frac{1}{t^2}\right), \qquad \text{(A.153)}
\end{aligned}$$

where, in (A.153), we used $2ab \leq a^2 + b^2$ and we observed that $\sum_{t=T_P}^{T} \left((\eta_c^{(t)})^2 + \frac{1}{t^2}\right) \leq \sum_{t=1}^{T} \left((\eta_c^{(t)})^2 + \frac{1}{t^2}\right)$ since $t > 0$ and $\eta_c^{(t)} > 0$.

Moreover, if the step-size $(\eta_c^{(t)})_{t\geq 1}$ decreases and satisfies $\eta_1 \leq \frac{1}{2L(1+2KQ)}$, $\sum_{t=1}^{+\infty} (\eta_c^{(t)})^2 < +\infty$, and $\sum_{t=1}^{+\infty} \ln(t) \cdot (\eta^{(t)})^2 < +\infty$, we can further bound the first term in (A.153) by combining Lemma B.11 and Lemma B.14 for $t_0 = T_P$, and we obtain:

$$\sum_{t=T_P}^{T} \eta_c^{(t)} \mathbb{E}\left[\sum_{i\in\mathcal{S}^{(t)}} q_i\big(F_i(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)}) - F_i(\boldsymbol{w}_B^*)\big)\right] \leq C_0 + \frac{C_1}{\ln(1/\lambda(\boldsymbol{P}))} < +\infty, \qquad \text{(A.154)}$$

where:

$$\begin{aligned}
C_0 \triangleq\ & \frac{2}{E} \operatorname{diam}(W)^2 + (E+1) \left(\sum_{i=1}^{N} \pi_i q_i^2 \sigma_i^2\right) \left(\sum_{t=1}^{+\infty} (\eta_c^{(t)})^2\right) \\
&+ 2E^2 G^2 \left(\sum_{i=1}^{N} \pi_i q_i\right) \left(\sum_{t=1}^{+\infty} (\eta_c^{(t)})^2\right) \\
&+ 4L(1+KQ)\Gamma \left(\sum_{i=1}^{N} \pi_i q_i\right) \left(\sum_{t=1}^{+\infty} (\eta_c^{(t)})^2\right). \qquad \text{(A.155)}
\end{aligned}$$

Finally, plugging (A.154) into (A.153), observing that $\sum_{t=1}^{T} \left((\eta_c^{(t)})^2 + \frac{1}{t^2}\right) \leq \sum_{t=1}^{+\infty} \left((\eta_c^{(t)})^2 + \frac{1}{t^2}\right) < +\infty$ because $\sum_{t=1}^{+\infty} (\eta_c^{(t)})^2 < +\infty$ and $\sum_{t=1}^{+\infty} \frac{1}{t^2} = \frac{\pi}{6} < +\infty$, and denoting $C_3 \triangleq C_0 + \frac{MQ}{4} \sum_{t=1}^{+\infty} \left((\eta_c^{(t)})^2 + \frac{1}{t^2}\right) < +\infty$, we conclude the proof of Lemma B.18.

□

## B.C Proof of Theorem 2.3.3

**Theorem B.19** (Convergence of the optimization error $\epsilon_{\text{opt}}$)**.** *Let Assumptions 1–3 and 5 hold and the functions $\{F_i\}_{i=1}^N$ be convex. Recall the constants $M, L, D, G, H, \Gamma$, $\sigma_i, C_P, T_P, \mathcal{J}^{(t)}$, and $\lambda(\boldsymbol{P})$ defined above. Let $Q = \sum_{i\in\mathcal{N}} q_i$.*
*Let the step-size $\eta_c^{(t)} > 0$ decrease and satisfy:*

$$\eta_1 \leq \frac{1}{2L(1+2KQ)}, \qquad \sum_{t=1}^{+\infty} \eta_t = +\infty, \qquad \sum_{t=1}^{+\infty} \ln(t)\cdot(\eta_c^{(t)})^2 < +\infty. \tag{2.12}$$

*Let $T$ denote the total communication rounds.*
*For $T \geq T_P$, the expected optimization error $\mathbb{E}[F_B(\bar{\boldsymbol{w}}^{(T,0)}) - F_B^*]$ can be bounded as follows:*

$$\mathbb{E}[F_B(\bar{\boldsymbol{w}}^{(T,0)}) - F_B^*] \leq \frac{\frac{1}{2}\boldsymbol{q}^\intercal\boldsymbol{\Sigma}\boldsymbol{q}+\upsilon}{\boldsymbol{\pi}^\intercal\boldsymbol{q}} + \psi + \frac{\phi}{\ln(1/\lambda(\boldsymbol{P}))}}{(\sum_{t=1}^T \eta_c^{(t)})}, \tag{2.13}$$

*where $\bar{\boldsymbol{w}}^{(T,0)} = \frac{\sum_{t=1}^T \eta_c^{(t)}\boldsymbol{w}^{(t,0)}}{\sum_{t=1}^T \eta_c^{(t)}}$, and:*

$$\boldsymbol{\Sigma} \triangleq \operatorname{diag}\left(2\left(E+1\right)\pi_i\sigma_i^2\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right); \tag{A.156}$$

$$\upsilon \triangleq \frac{2}{E}\operatorname{diam}(W)^2 + \frac{MQ}{4}\sum_{t=1}^{+\infty}\left((\eta_c^{(t)})^2 + \frac{1}{t^2}\right); \tag{A.157}$$

$$\psi \triangleq 4L(1+KQ)\Gamma\left(\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right) + 2E^2G^2\left(\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right) + H\left(\sum_{t=1}^{T_P-1}\eta_c^{(t)}\right); \tag{A.158}$$

$$\phi \triangleq 2EDGQ\left(\sum_{t=1}^{+\infty}\ln(2C_PHt)\cdot(\eta^{(t-\mathcal{J}^{(t)})})^2\right). \tag{A.159}$$

*Proof of Theorem B.19.*

The proof involves three main steps.

### Step 1

From Lemma B.15, observe that:

$$\left(\sum_{i=1}^N \pi_i q_i\right)\sum_{t=1}^T \eta_c^{(t)}\mathbb{E}[F_B(\boldsymbol{w}^{(t,0)}) - F_B(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)})] \leq \frac{C_1}{\ln(1/\lambda(\boldsymbol{P}))} < +\infty, \tag{A.160}$$

where:

$$C_1 \triangleq EDGQ\left(\sum_{i=1}^N \pi_i q_i\right)\left(\sum_{t=1}^{+\infty}\ln\left(2C_PHt\right)\cdot(\eta^{(t-\mathcal{J}^{(t)})})^2\right) < +\infty. \tag{A.161}$$

**Step 2**

By combining Lemma B.17 and Lemma B.18, we obtain:

$$\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=1}^{T}\eta_c^{(t)}\mathbb{E}[F_B(\boldsymbol{w}^{(t-\mathcal{J}^{(t)},0)})-F_B^*)] \leq \frac{C_1}{\ln(1/\lambda(\boldsymbol{P}))}+C_2+C_3<+\infty, \tag{A.162}$$

where $C_1$ is defined in (A.161), and:

$$C_2 \triangleq H\left(\sum_{t=1}^{T_P-1}\eta_c^{(t)}\right)\left(\sum_{i=1}^{N}\pi_i q_i\right)<+\infty; \tag{A.163}$$

$$\begin{aligned} C_3 \triangleq\ & \frac{2}{E}\operatorname{diam}(W)^2+(E+1)\left(\sum_{i=1}^{N}\pi_i q_i^2\sigma_i^2\right)\left(\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right) \\ & +2E^2G^2\left(\sum_{i=1}^{N}\pi_i q_i\right)\left(\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right) \\ & +4L(1+KQ)\Gamma\left(\sum_{i=1}^{N}\pi_i q_i\right)\left(\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right)+\frac{MQ}{4}\sum_{t=1}^{+\infty}\left((\eta_c^{(t)})^2+\frac{1}{t^2}\right)<+\infty. \end{aligned} \tag{A.164}$$

**Step 3**

By summing the results from Steps 1 and 2, given in (A.160) and (A.162), respectively, we have:

$$\left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=1}^{T}\eta_c^{(t)}\mathbb{E}[F_B(\boldsymbol{w}^{(t,0)})-F_B^*] \leq \frac{2C_1}{\ln(1/\lambda(\boldsymbol{P}))}+C_2+C_3<+\infty. \tag{A.165}$$

With the convexity of $F_B(\cdot)$, applying the Jensen's inequality, we complete Step 3:

$$\left(\sum_{t=1}^{T}\eta_c^{(t)}\right)\left(\sum_{i=1}^{N}\pi_i q_i\right)\mathbb{E}[F_B(\bar{\boldsymbol{w}}^{(T,0)})-F_B^*] \leq \left(\sum_{i=1}^{N}\pi_i q_i\right)\sum_{t=1}^{T}\eta_c^{(t)}\mathbb{E}[F_B(\boldsymbol{w}^{(t,0)})-F_B^*] \tag{A.166}$$

$$\leq \frac{2C_1}{\ln(1/\lambda(\boldsymbol{P}))}+C_2+C_3<+\infty, \tag{A.167}$$

where $\bar{\boldsymbol{w}}^{(T,0)} := \frac{\sum_{t=1}^{T}\eta_c^{(t)}\boldsymbol{w}^{(t,0)}}{\sum_{t=1}^{T}\eta_c^{(t)}}$, and the constants $C_1$, $C_2$, and $C_3$ are defined in (A.161), (A.163), and (A.164), respectively.

By dividing (A.166) and (A.167) by $\left(\sum_{t=1}^{T}\eta_c^{(t)}\right)\cdot\left(\sum_{i=1}^{N}\pi_i q_i\right)$, we obtain the expression for Theorem B.19 given in (2.13).

□

# C Proof of Theorem 2.3.4

**Theorem C.1** (An alternative bound on the bias error $\epsilon_{\text{bias}}$)**.** *Under the same assumptions of Theorem 2.3.2, define* $\Gamma' \triangleq \max_i\{F_i(\boldsymbol{w}_B^*) - F_i^*\}$. *The following result holds:*

$$\epsilon_{bias} \leq 4\kappa^2 \cdot \underbrace{d_{TV}^2(\boldsymbol{\alpha}, \boldsymbol{p}) \cdot \Gamma'}_{\triangleq \bar{\epsilon}'_{bias}}, \tag{2.20}$$

*where* $d_{TV}(\boldsymbol{\alpha}, \boldsymbol{p}) \triangleq \frac{1}{2}\sum_{i=1}^N |\alpha_i - p_i|$ *denotes the total variation distance between the probability distributions* $\boldsymbol{\alpha}$ *and* $\boldsymbol{p}$.

*Proof of Theorem C.1.*

The proof follows the same steps as in Theorem A.1, proceeding from (A.9) as follows:

$$\|\nabla F(\boldsymbol{w}_B^*)\| \leq L\sqrt{\frac{2}{\mu}}\sum_{i=1}^N |\alpha_i - p_i|\sqrt{(F_i(\boldsymbol{w}_B^*) - F_i^*)} \tag{A.9}$$

$$\leq 2L\sqrt{\frac{2}{\mu}} d_{TV}(\boldsymbol{\alpha}, \boldsymbol{p})\sqrt{\Gamma'}, \tag{A.168}$$

where, in (A.168), we applied the definitions of $d_{TV}(\boldsymbol{\alpha}, \boldsymbol{p}) \triangleq \frac{1}{2}\sum_{i=1}^N |\alpha_i - p_i|$ and $\Gamma' \triangleq \max_i\{F_i(\boldsymbol{w}_B^*) - F_i^*\}$.

Squaring (A.168), we obtain the following expression:

$$\|\nabla F(\boldsymbol{w}_B^*)\|^2 \leq \frac{8L^2}{\mu} d_{TV}^2(\boldsymbol{\alpha}, \boldsymbol{p})\Gamma'. \tag{A.169}$$

Then, replacing (A.169) in (A.5), we obtain:

$$\epsilon_{\text{bias}} \triangleq (F(\boldsymbol{w}_B^*) - F^*) \leq \frac{1}{2\mu}\|\nabla F(\boldsymbol{w}_B^*)\|^2 \leq 4\frac{L^2}{\mu^2}\underbrace{d_{TV}^2(\boldsymbol{\alpha}, \boldsymbol{p})\Gamma'}_{\triangleq \bar{\epsilon}'_{\text{bias}}}, \tag{A.170}$$

which concludes the proof of Theorem C.1.

□

# D Convexity of $\bar{\epsilon}_{\text{opt}} + \bar{\epsilon}_{\text{bias}}$

For the proof of the convexity of $\bar{\epsilon}_{\text{opt}}(\boldsymbol{q})$, please refer to Appendix E.A. To prove that $\bar{\epsilon}_{\text{bias}}(\boldsymbol{q})$ is also convex, we need to study the convexity of $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}} \triangleq \sum_{i=1}^N (\alpha_i - p_i)^2/p_i$ in $\boldsymbol{q} \in \{q_i > 0\ \forall k, \|\boldsymbol{q}\|_1 = Q > 0\}$. To this purpose, we define the following functions:

$$h_i : \mathbb{R}_{\geq 0}^N \setminus \{\boldsymbol{0}\} \to \mathbb{R}_{\geq 0}, \qquad h_i(\boldsymbol{q}) \triangleq \frac{\pi_i q_i}{\sum_{k'=1}^N \pi_{k'} q_{k'}}; \tag{A.171}$$

$$g_i : \mathbb{R}_{>0} \to \mathbb{R}_{\geq 0}, \qquad g_i(p_i) \triangleq \frac{(p_i - \alpha_i)^2}{p_i}. \tag{A.172}$$

Finally, we write the chi-square divergence $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}}$ between the target and biased probability distributions $\boldsymbol{\alpha}$ and $\boldsymbol{p}$ as:

$$\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}}(\boldsymbol{q}) = \sum_{i=1}^{N} (g_i \circ h_i)(\boldsymbol{q}) = \sum_{i=1}^{N} g_i(h_i(\boldsymbol{q})). \tag{A.173}$$

We observe that:

$h_i(\boldsymbol{q})$ is a particular case of linear-fractional functions (Boyd & Vandenberghe, 2004, Example 3.32, p. 97);

$g_i(\cdot)$ is a convex in $p_i$ over $\mathbb{R}_{>0}$ because sum of convex functions;

each $g_i \circ h_i$ is quasi-convex in $\boldsymbol{q} \in \mathbb{R}^N_{>0}$ because composition of a convex function ($g_i$) and a linear-fractional function ($h_i$) (Boyd & Vandenberghe, 2004, p. 102).

However, note that the sum of quasi-convex functions is not necessarily quasi-convex.

**Proposition D.1.** *The function $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}}(\boldsymbol{q})$ is not convex over $\mathbb{R}^N_{>0}$.*

*Proof of Proposition D.1.*

To analyze the convexity of $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}}(\boldsymbol{q}) = \sum_{i=1}^{N} (g_i \circ h_i)(\boldsymbol{q})$ over $\mathbb{R}^N_{>0}$, a possible approach is to check whether each function $(g_i \circ h_i)(\boldsymbol{q})$ is convex over $\mathbb{R}^N_{>0}$. In what follows, we show that $(g_i \circ h_i)$ is not convex over $\mathbb{R}^N_{>0}$.

Consider the case when $\pi_i = 1 \; \forall i \in \mathcal{N}$. We can rewrite $(g_i \circ h_i)(\boldsymbol{q})$ as follows:

$$(g_i \circ h_i)(\boldsymbol{q}) = \frac{\left(\frac{q_i}{\|\boldsymbol{q}\|_1} - \alpha_i\right)^2}{\frac{q_i}{\|\boldsymbol{q}\|_1}}. \tag{A.174}$$

We show that this function fails to satisfy the definition of convexity, i.e., $\exists\, \boldsymbol{q}, \boldsymbol{q'} \in \mathbb{R}^N_{>0}, \zeta \in [0,1]$ such that:

$$(g_i \circ h_i)\left(\zeta \boldsymbol{q} + (1-\zeta)\boldsymbol{q'}\right) > \zeta\,(g_i \circ h_i)\,(\boldsymbol{q}) + (1-\zeta)\,(g_i \circ h_i)\,(\boldsymbol{q'}). \tag{A.175}$$

The left-hand side (LHS) of (A.175) is:

$$(g_i \circ h_i)\left(\zeta \boldsymbol{q} + (1-\zeta)\boldsymbol{q'}\right) = \frac{\left(\frac{\zeta q_i + (1-\zeta) q'_i}{\zeta \|\boldsymbol{q}\|_1 + (1-\zeta)\|\boldsymbol{q'}\|_1} - \alpha_i\right)^2}{\frac{\zeta q_i + (1-\zeta) q'_i}{\zeta \|\boldsymbol{q}\|_1 + (1-\zeta)\|\boldsymbol{q'}\|_1}}. \tag{A.176}$$

If we take $\boldsymbol{q} : \|\boldsymbol{q}\|_1 = 1$, $q_i = \alpha_i$, $\zeta = \frac{1}{2}$, $\boldsymbol{q'} = \frac{Q}{N}\mathbf{1}$, and we let $Q \to +\infty$, then the LHS in (A.176) converges to:

$$\lim_{Q \to +\infty} \frac{\left(\frac{\frac{1}{2}\alpha_i + \frac{1}{2}\frac{Q}{N}}{\frac{1}{2}1 + \frac{1}{2}Q} - \alpha_i\right)^2}{\frac{\frac{1}{2}\alpha_i + \frac{1}{2}\frac{Q}{N}}{\frac{1}{2}1 + \frac{1}{2}Q}} = \frac{\left(\frac{1}{N} - \alpha_i\right)^2}{\frac{1}{N}}. \tag{A.177}$$

On the other hand, for the same choices of $q_i$, $\boldsymbol{q}$, $\boldsymbol{q'}$, and $\zeta$, and if we let $Q \to +\infty$, the right-hand side (RHS) of (A.175) is:

$$\zeta\,(g_i \circ h_i)\,(\boldsymbol{q}) + (1-\zeta)\,(g_i \circ h_i)\,(\boldsymbol{q'}) = 0 + \frac{1}{2}\frac{\left(\frac{1}{N} - \alpha_i\right)^2}{\frac{1}{N}}. \tag{A.178}$$

Finally, comparing (A.177) and (A.178), we conclude that, for $Q$ large enough, the LHS in (A.175) is larger than the RHS.

□

**Proposition D.2.** *The function $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}}(\boldsymbol{q})$ is convex over $\mathbb{R}^N_{>0} \cap \{\boldsymbol{q} : \|\boldsymbol{q}\|_1 = Q > 0\}$.*

*Proof of Proposition D.2.*

To verify the convexity of $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}}(\boldsymbol{q}) = \sum_{i=1}^N (g_i \circ h_i)(\boldsymbol{q})$ over $\mathbb{R}^N_{>0} \cap \{\boldsymbol{q} : \|\boldsymbol{q}\|_1 = Q > 0\}$, one possible approach is to demonstrate the convexity of each function $(g_i \circ h_i)(\boldsymbol{q})$ over the set $\mathbb{R}^N_{>0} \cap \{\boldsymbol{q} : \|\boldsymbol{q}\|_1 = Q > 0\}$.

We prove this result for a more general case. We show that, if

$$\tilde{g} \text{ is a convex function over its domain } \mathcal{D}_g \tag{A.179}$$

and

$$\tilde{h}(\boldsymbol{q}) = \frac{\boldsymbol{A}\boldsymbol{q} + b}{\boldsymbol{c}^\intercal \boldsymbol{q} + d}, \tag{A.180}$$

then

$$\tilde{g} \circ \tilde{h} \text{ is convex over } \mathcal{D} = \mathbb{R}^N_{>0} \cap \{\boldsymbol{q} : \boldsymbol{c}^\intercal \boldsymbol{q} + d = Q > 0, \frac{\boldsymbol{A}\boldsymbol{q} + b}{\boldsymbol{c}^\intercal \boldsymbol{q} + d} \in \mathcal{D}_g\}. \tag{A.181}$$

It is then sufficient to apply this result to each pair $(g_i, h_i)$ to conclude that $(g_i \circ h_i)$ is convex and then $\chi^2_{\boldsymbol{\alpha}\|\boldsymbol{p}}(\boldsymbol{q})$ is convex.

By direct inspection, for all $\boldsymbol{q}, \boldsymbol{q'} \in \mathcal{D}$, $\forall\, \zeta \in [0, 1]$, the following equality holds:

$$\left(\tilde{g} \circ \tilde{h}\right)(\zeta \boldsymbol{q} + (1-\zeta)\boldsymbol{q'}) = \tilde{g}\left(\tilde{h}\left(\zeta \boldsymbol{q} + (1-\zeta)\boldsymbol{q'}\right)\right) = \tilde{g}\left(\zeta' \frac{\boldsymbol{A}\boldsymbol{q} + b}{\boldsymbol{c}^\intercal \boldsymbol{q} + d} + (1-\zeta')\frac{\boldsymbol{A}\boldsymbol{q'} + b}{\boldsymbol{c}^\intercal \boldsymbol{q'} + d}\right), \tag{A.182}$$

where:

$$\zeta' = \frac{\zeta\left(\boldsymbol{c}^\intercal \boldsymbol{q} + d\right)}{\zeta\left(\boldsymbol{c}^\intercal \boldsymbol{q} + d\right) + (1-\zeta)\left(\boldsymbol{c}^\intercal \boldsymbol{q'} + d\right)} \in [0, 1]. \tag{A.183}$$

Applying the convexity of $\tilde{g}$, we bound Equation (A.182) as follows:

$$\tilde{g}\left(\zeta' \frac{\boldsymbol{A}\boldsymbol{q} + b}{\boldsymbol{c}^\intercal \boldsymbol{q} + d} + (1-\zeta')\frac{\boldsymbol{A}\boldsymbol{q'} + b}{\boldsymbol{c}^\intercal \boldsymbol{q'} + d}\right) \overset{\text{convexity of } \tilde{g}}{\leq} \zeta' \tilde{g}\left(\frac{\boldsymbol{A}\boldsymbol{q} + b}{\boldsymbol{c}^\intercal \boldsymbol{q} + d}\right) + (1-\zeta')\tilde{g}\left(\frac{\boldsymbol{A}\boldsymbol{q'} + b}{\boldsymbol{c}^\intercal \boldsymbol{q'} + d}\right) \tag{A.184}$$

$$= \quad \zeta'\left(\tilde{g} \circ \tilde{h}\right)(\boldsymbol{q}) + (1-\zeta')\left(\tilde{g} \circ \tilde{h}\right)(\boldsymbol{q'}). \tag{A.185}$$

Finally, to conclude the proof, we show that $\zeta' = \zeta$. This is true because, for any $\boldsymbol{q}$ and $\boldsymbol{q'} \in \mathcal{D}$, $\boldsymbol{c}^\intercal \boldsymbol{q} + d = \boldsymbol{c}^\intercal \boldsymbol{q'} + d = Q > 0$. In fact, by using this condition in Equation (A.183), we have that:

$$\zeta' = \frac{\zeta Q}{\zeta Q + (1-\zeta)Q} = \zeta, \tag{A.186}$$

which establishes the convexity of $\tilde{g} \circ \tilde{h}$ by definition.

□

# E Minimizing $\bar{\epsilon}_{\text{opt}}$

Equation (2.13) can be rewritten as:

$$\left(\sum_{t=1}^{T}\eta_c^{(t)}\right)\mathbb{E}\left[F_B(\bar{\boldsymbol{w}}^{(T,0)})-F_B^*\right]\leq\frac{\frac{1}{2}\boldsymbol{q}^\intercal\boldsymbol{\Sigma}\boldsymbol{q}+\upsilon}{\boldsymbol{\pi}^\intercal\boldsymbol{q}}+\psi+\frac{\phi}{\ln(1/\lambda(\boldsymbol{P}))} \tag{A.187}$$

$$=\frac{\frac{1}{2}\boldsymbol{q}^\intercal\boldsymbol{A}\boldsymbol{q}+B}{\boldsymbol{\pi}^\intercal\boldsymbol{q}}+C\triangleq J(\boldsymbol{q}), \tag{A.188}$$

where:

$$\boldsymbol{A}\triangleq\boldsymbol{\Sigma}=\operatorname{diag}\left(2\left(E+1\right)\pi_i\sigma_i^2\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right); \tag{A.189}$$

$$B\triangleq\upsilon=\frac{2}{E}\operatorname{diam}(W)^2+\frac{MQ}{4}\sum_{t=1}^{+\infty}\left((\eta_c^{(t)})^2+\frac{1}{t^2}\right); \tag{A.190}$$

$$C\triangleq\psi+\frac{\phi}{\ln(1/\lambda(\boldsymbol{P}))}$$

$$=\left(4L(1+KQ)\Gamma+2E^2G^2\right)\left(\sum_{t=1}^{+\infty}(\eta_c^{(t)})^2\right)+2EDGQ\left(\sum_{t=1}^{+\infty}\mathcal{J}^{(t)}\cdot(\eta^{(t-\mathcal{J}^{(t)})})^2\right)+H\left(\sum_{t=1}^{T_P-1}\eta_c^{(t)}\right). \tag{A.191}$$

The minimization of (A.188), defines the following optimization problem:

$$\underset{\boldsymbol{q}}{\text{minimize}}\quad J(\boldsymbol{q})\triangleq\frac{\frac{1}{2}\boldsymbol{q}^\intercal\boldsymbol{A}\boldsymbol{q}+B}{\boldsymbol{\pi}^\intercal\boldsymbol{q}}+C; \tag{A.192a}$$

$$\text{subject to}\quad \boldsymbol{q}\geq 0, \tag{A.192b}$$

$$\boldsymbol{\pi}^\intercal\boldsymbol{q}>0, \tag{A.192c}$$

$$\|\boldsymbol{q}\|_1=Q. \tag{A.192d}$$

**Remark E.1.** In Problem (A.192a)–(A.192d), when setting some $q_i$ to zero, we do not consider the possibility of redefining the Markov chain $(\mathcal{S}^{(t)})_{t\geq 0}$ in Assumption 1 by considering the reduced state space of clients with $q_i>0$. In this case, the redefined Markov chain would have a different transition matrix $\boldsymbol{P'}\neq\boldsymbol{P}$ with $\lambda(\boldsymbol{P'})\neq\lambda(\boldsymbol{P})$, resulting in $C$ no longer being constant.

## E.A The optimization problem in (A.192a)–(A.192d) is convex

Let us rewrite the problem by adding a variable $s\triangleq 1/\boldsymbol{\pi}^\intercal\boldsymbol{q}$ and then replacing $\boldsymbol{y}\triangleq s\boldsymbol{q}$. We have:

$$J(\boldsymbol{y},s)=s\left(\frac{1}{2}\frac{\boldsymbol{y}^\intercal}{s}\boldsymbol{A}\frac{\boldsymbol{y}}{s}+B\right)+C=s\cdot K\left(\frac{\boldsymbol{y}}{s}\right)+C, \tag{A.193}$$

where $K:\mathbb{R}^N\to\mathbb{R}$, $K(\boldsymbol{q})\triangleq\frac{1}{2}\boldsymbol{q}^\intercal\boldsymbol{A}\boldsymbol{q}+B$ is a (strictly) convex function, and:

$$\underset{\boldsymbol{y},\,s}{\text{minimize}}\quad J(\boldsymbol{y},s)=\frac{1}{2s}\boldsymbol{y}^\intercal\boldsymbol{A}\boldsymbol{y}+Bs+C \tag{A.194a}$$

$$\text{subject to}\quad \boldsymbol{y}\geq 0, \tag{A.194b}$$

$$s > 0, \tag{A.194c}$$
$$\boldsymbol{\pi}^\intercal \boldsymbol{y} = 1, \tag{A.194d}$$
$$\|\boldsymbol{y}\|_1 = Qs. \tag{A.194e}$$

Note that the objective function $J(\boldsymbol{y}, s) : \mathbb{R}^{N+1} \to \mathbb{R}$, $J(\boldsymbol{y}, s) = s \cdot K(\boldsymbol{y}/s) + C$ in (A.193) is the perspective of the convex function $K(\boldsymbol{q}) + C$, and is therefore convex (Boyd & Vandenberghe, 2004, pp. 89–90). Moreover, the constraints in (A.194b)–(A.194e) define a convex set, and then the optimization problem defined by (A.194a)–(A.194e) is convex. We solve it with the method of Lagrange multipliers.

## E.B Support for Guideline A (Section 2.3)

The Lagrangian function $\mathcal{L}$ is as follows:

$$\mathcal{L}(\boldsymbol{y}, s, \iota, \theta, \boldsymbol{\omega}) = \frac{1}{2s}\boldsymbol{y}^\intercal \boldsymbol{A}\boldsymbol{y} + Bs + C + \iota(1 - \boldsymbol{\pi}^\intercal \boldsymbol{y}) + \theta(\|\boldsymbol{y}\|_1 - Qs) - \boldsymbol{\omega}^\intercal \boldsymbol{y}. \tag{A.195}$$

Since the constraint $s > 0$ defines an open set, the set defined by the constraints in (A.194b)–(A.194e) is not closed. However, the solution of the optimization problem defined by (A.194a)–(A.194e) is never on the boundary $s = 0$ because $\mathcal{L} \to +\infty$ as $s \to 0^+$, therefore we can consider $s \geq 0$. Moreover, strong duality holds for the Slater's constraint qualification for convex problems.

The KKT conditions read:

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial s}(\boldsymbol{y}^*, s^*, \iota^*, \theta^*, \boldsymbol{\omega}^*) = 0, & \text{(A.196)} \\ \nabla_{\boldsymbol{y}} \mathcal{L}(\boldsymbol{y}^*, s^*, \iota^*, \theta^*, \boldsymbol{\omega}^*) = 0, & \text{(A.197)} \\ \boldsymbol{\pi}^\intercal \boldsymbol{y}^* - 1 = 0, & \text{(A.198)} \\ \|\boldsymbol{y}^*\|_1 - Qs = 0, & \text{(A.199)} \\ \boldsymbol{\omega}^{*\intercal} \boldsymbol{y}^* = 0, & \text{(A.200)} \\ \boldsymbol{y}^*, \boldsymbol{\omega}^* \geq 0. & \text{(A.201)} \end{cases}$$

In particular, the KKT condition for $\boldsymbol{y}^*$ read:

$$\nabla_{\boldsymbol{y}} \mathcal{L}(\boldsymbol{y}^*, s^*, \iota^*, \theta^*, \boldsymbol{\omega}^*) = \frac{1}{s^*}\boldsymbol{A}\boldsymbol{y}^* - \iota^* \boldsymbol{\pi} + \theta^* \mathbf{1} - \boldsymbol{\omega}^* = 0, \tag{A.202}$$

which is satisfied when:

$$\frac{\partial \mathcal{L}}{\partial y_i^*} = \frac{1}{s^*} A_{kk} y_i^* - \iota^* \pi_i + \theta^* - \omega_i^* = 0, \ \forall i \in \mathcal{N}, \tag{A.203}$$

where $A_{ij}$ denotes the element on the $i$-th row and the $k$-th column of matrix $\boldsymbol{A}$.

Furthermore, the Complementary Slackness conditions in (A.200) and (A.201) present two cases:

1. If $y_i^* > 0$ (and $q_i^* > 0$), then $\omega_i^* = 0$ and:

$$y_i^* = \frac{s^*}{A_{kk}}(\iota^* \pi_i - \theta^*), \quad q_i^* = \frac{1}{A_{kk}}(\iota^* \pi_i - \theta^*); \tag{A.204}$$

2. $y_i^* = q_i^* = 0$ otherwise.

By replacing the equality constraint (A.194d) in Problem (A.194a)–(A.194e) with the inequality constraint $\boldsymbol{\pi}^\intercal \boldsymbol{y} \geq 1$, we establish an equivalent optimization problem. The equivalence holds because, for any feasible solution $\boldsymbol{y}'$ with $\boldsymbol{\pi}^\intercal \boldsymbol{y}' > 1$, we can consider the solution $\boldsymbol{y}'' = \frac{\boldsymbol{y}'}{\boldsymbol{\pi}^\intercal \boldsymbol{y}'} < \boldsymbol{y}'$, leading to a lower objective function value. Additionally, the new problem states that the Lagrange multiplier ($\iota^*$) associated with the inequality constraint must be non-negative. By considering $A_{kk} \geq 0$ and $\iota^* \geq 0$ in Equation (A.204), we conclude that $q_i^*$ increases with $\pi_i$, providing analytical support for Guideline A.

### E.C Closed-form solution of the optimization problem in (A.192a)–(A.192d)

The solution of the optimization problem in (A.192a)–(A.192d) is not of practical utility because its constants (e.g., $L$, $\omega$, $\Gamma$, $C_P$) are in general problem-dependent and difficult to estimate during training. In particular, $\Gamma$ poses particular difficulties as it is defined in terms of the minimizer of the target objective $F$, but the FL algorithm generally minimizes the biased function $F_B$. Nevertheless, we include the closed-formed solution of the optimization problem in (A.192a)–(A.192d) for completeness.

We use the active-set method: let $\mathcal{X}$ be the set of coordinates corresponding to the active inequalities, i.e., $\mathcal{X} = \{k \mid y_i^* = 0\}$.

From the KKT condition in (A.198), we derive a relation between $\iota^*$ and $\theta^*$:

$$\boldsymbol{\pi}^\intercal \boldsymbol{y}^* = \sum_{k \notin \mathcal{X}} \pi_i y_i^* = \sum_{k \notin \mathcal{X}} \pi_i \frac{s^*}{\boldsymbol{A}_{kk}} (\iota^* \pi_i - \theta^*) = \iota^* s^* \sum_{k \notin \mathcal{X}} \frac{\pi_i^2}{\boldsymbol{A}_{kk}} - \theta^* s^* \sum_{k \notin \mathcal{X}} \frac{\pi_i}{\boldsymbol{A}_{kk}} = 1. \tag{A.205}$$

We use the KKT condition in (A.199) to derive another relation between $\iota^*$ and $\theta^*$:

$$\|\boldsymbol{y}^*\|_1 = \sum_{k \notin \mathcal{X}} y_i^* = \sum_{k \notin \mathcal{X}} \frac{s^*}{\boldsymbol{A}_{kk}} (\iota^* \pi_i - \theta^*) = Qs \quad \Leftrightarrow \quad \iota^* = \frac{Q + \theta^* \sum_{k \notin \mathcal{X}} \frac{1}{\boldsymbol{A}_{kk}}}{\sum_{k \notin \mathcal{X}} \frac{\pi_i}{\boldsymbol{A}_{kk}}}, \tag{A.206}$$

and, replacing (A.206) in (A.205), we derive the closed-form solution for $\theta^*$:

$$\theta^* = \frac{\sum_{k \notin \mathcal{X}} \frac{\pi_i}{\boldsymbol{A}_{kk}} - Qs^* \sum_{k \notin \mathcal{X}} \frac{\pi_i^2}{\boldsymbol{A}_{kk}}}{s^* \left[ \left( \sum_{k \notin \mathcal{X}} \frac{1}{\boldsymbol{A}_{kk}} \right) \cdot \left( \sum_{k \notin \mathcal{X}} \frac{\pi_i^2}{\boldsymbol{A}_{kk}} \right) - \left( \sum_{k \notin \mathcal{X}} \frac{\pi_i}{\boldsymbol{A}_{kk}} \right)^2 \right]}. \tag{A.207}$$

## F Background on Markov Chains

### F.A Markov Chain for the Analysis (Section 2.3)

We recall some existing results (Levin & Peres, 2017; Sun et al., 2018) for the Markov chain $(\mathcal{S}^{(t)})_{t \geq 0}$ used in our analysis (Assumption 1).

**Assumption 1.** *The Markov chain $(\mathcal{S}^{(t)})_{t \geq 0}$ on the $M$-finite state space $\mathcal{M}$ is time-homogeneous, irreducible, and aperiodic. It has transition matrix $\boldsymbol{P}$, stationary distribution $\boldsymbol{\rho}$, and has state distribution $\boldsymbol{\rho}$ at time $t = 0$.*

Let $\boldsymbol{\rho}^{(t)} = [\rho_1^{(t)}, \rho_2^{(t)}, \ldots, \rho_M^{(t)}], \sum_{i=1}^M \rho_i^{(t)} = 1$ be the state probability distribution on the Markov chain $(\mathcal{S}^{(t)})_{t \geq 0}$ at time step $t$. Assumption 1 guarantees the existence of a stationary distribution $\boldsymbol{\rho} = \lim_{t \to +\infty} \boldsymbol{\rho}^{(t)} = [\rho_1, \rho_2, \ldots, \rho_M]$

with $\min_i\{\rho_i\} > 0$ and $\boldsymbol{\rho}^\intercal \boldsymbol{P} = \boldsymbol{\rho}^\intercal$. Then $\boldsymbol{\rho}$ is a left eigenvector relative to the eigenvalue 1, which is the largest eigenvalue of the matrix $\boldsymbol{P}$.

For the transition matrix $\boldsymbol{P}$, we label its eigenvalues in decreasing order:

$$1 = \lambda_1(\boldsymbol{P}) > \lambda_2(\boldsymbol{P}) \geq \cdots \geq \lambda_M(\boldsymbol{P}). \tag{A.208}$$

We define:

$$\bar{\lambda}_2(\boldsymbol{P}) := \max\{|\lambda_2(\boldsymbol{P})|, |\lambda_M(\boldsymbol{P})|\} \quad \text{and} \quad \lambda(\boldsymbol{P}) := \frac{\bar{\lambda}_2(\boldsymbol{P}) + 1}{2}. \tag{A.209}$$

The second largest absolute eigenvalue $\bar{\lambda}_2(\boldsymbol{P})$ of the transition matrix $\boldsymbol{P}$ characterizes the mixing time of a Markov chain. The absolute spectral gap $\gamma := 1 - \bar{\lambda}_2(\boldsymbol{P})$ and its reciprocal, the relaxation time $t_{\text{rel}} := \frac{1}{\gamma}$, play a role in this relationship. To quantify the convergence of the Markov chain towards stationarity, we use the parameter $d(t) \triangleq \max_{a\in\mathcal{M}} \|[\boldsymbol{P}^t]_{a,\cdot} - \boldsymbol{\rho}\|_{TV}$, which measures the maximum distance between the distribution $[\boldsymbol{P}^t]_{a,\cdot}$ and the stationary distribution $\boldsymbol{\rho}$ for all initial states $a \in \mathcal{M}$. The mixing time $t_{\text{mix}}(\varepsilon)$ is defined as the minimum time at which the distance $d(t)$ becomes less than or equal to a given threshold $\varepsilon$: $t_{\text{mix}}(\varepsilon) := \min\{t : d(t) \leq \varepsilon\}$. Upper and lower bounds exist for the mixing time based on the relaxation time and the stationary distribution: $(t_{\text{rel}} - 1)\log\left(\frac{1}{2\varepsilon}\right) \leq t_{\text{mix}}(\varepsilon) \leq \log\left(\frac{1}{\varepsilon\rho_{\min}}\right) t_{\text{rel}}$, where $\rho_{\min} := \min_{a\in\mathcal{M}} \rho_a$ (Levin & Peres, 2017, pp. 154–156).

## F.B Markov Chain for Guideline B (Section 2.4)

In Section 2.3.3.1 (Guideline B), we examine a specific scenario where the availability of each client $i$ follows an independent Markov chain $(\mathcal{S}_i^{(t)})_{t\geq 0}$ with transition probability matrix $\boldsymbol{P}_i$. This setup allows us to model the aggregate process as a product of independent Markov chains, known as a Product Chain (Levin & Peres, 2017, Section 12.4).

**Definition F.1** (Product Chain)**.** Let $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ be transition matrices on state spaces $\mathcal{M}_1$ and $\mathcal{M}_2$ respectively, with corresponding stationary distributions $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$. We consider a Markov Chain on the state space $\mathcal{M}_1 \times \mathcal{M}_2$ that moves independently in the first and second coordinates according to $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ respectively. The transition matrix of this Markov Chain is the Kronecker product $\widetilde{\boldsymbol{P}} = \boldsymbol{P}_1 \otimes \boldsymbol{P}_2$, defined as:

$$\widetilde{\boldsymbol{P}}((x, y), (z, w)) = \boldsymbol{P}_1(x, z)\boldsymbol{P}_2(y, w). \tag{A.210}$$

**Proposition F.1.** *The stationary distribution of the Markov chain defined by $\widetilde{\boldsymbol{P}} = \boldsymbol{P}_1 \otimes \boldsymbol{P}_2$ is the Kronecker product $\widetilde{\boldsymbol{\rho}} = \boldsymbol{\pi}_1 \otimes \boldsymbol{\pi}_2$.*

*Proof.*

We can observe the following:

$$\widetilde{\boldsymbol{\rho}}^\intercal \widetilde{\boldsymbol{P}} = (\boldsymbol{\pi}_1 \otimes \boldsymbol{\pi}_2)^\intercal \cdot (\boldsymbol{P}_1 \otimes \boldsymbol{P}_2) = (\boldsymbol{\pi}_1^\intercal \boldsymbol{P}_1) \otimes (\boldsymbol{\pi}_2^\intercal \boldsymbol{P}_2) = \boldsymbol{\pi}_1^\intercal \otimes \boldsymbol{\pi}_2^\intercal = \widetilde{\boldsymbol{\rho}}^\intercal, \tag{A.211}$$

where, in (A.211), we used the mixed-product property of the Kronecker product in the second step, and in the third step, we noted that $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$ are the stationary distributions for $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$, respectively. For a comprehensive list of properties that the Kronecker product satisfies, please refer to (Meyer, 2001, p. 597).

□

**Proposition F.2** ((Levin & Peres, 2017, Exercise 12.6)). *Let $\boldsymbol{u}$ and $\boldsymbol{v}$ be eigenvectors of $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$, respectively, with eigenvalues $\lambda$ and $\mu$. Then $\boldsymbol{u} \otimes \boldsymbol{v}$ is an eigenvector of $\boldsymbol{P}_1 \otimes \boldsymbol{P}_2$ with eigenvalue $\lambda\mu$.*

*Proof.*

We can verify the following:

$$(\boldsymbol{u} \otimes \boldsymbol{v})^\intercal(\boldsymbol{P}_1 \otimes \boldsymbol{P}_2) = (\boldsymbol{u}^\intercal \boldsymbol{P}_1) \otimes (\boldsymbol{v}^\intercal \boldsymbol{P}_2) = (\lambda \boldsymbol{u}^\intercal) \otimes (\mu \boldsymbol{v}^\intercal) = \lambda\mu(\boldsymbol{u} \otimes \boldsymbol{v})^\intercal. \tag{A.212}$$

In (A.212), we used the mixed-product property and the associativity of the scalar multiplication with the Kronecker product.

□

In general, let $\boldsymbol{P}_1$ be a $m \times m$ matrix with eigenvalues $\lambda_1, ..., \lambda_m$, and $\boldsymbol{P}_2$ be a $n \times n$ matrix with eigenvalues $\mu_1, ..., \mu_n$. The complete eigen-decomposition of $\boldsymbol{P}_1 \otimes \boldsymbol{P}_2$ depends on the Kronecker product structure and involves combinations of the eigenvalues and eigenvectors of $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$.

**Proposition F.3** (Spectrum of the Kronecker product, (Meyer, 2001, Exercise 7.8.11)). *Let the eigenvalues of $\boldsymbol{P}_1 \in \mathbb{R}^{m\times m}$ be denoted by $\lambda_i$ and let the eigenvalues of $\boldsymbol{P}_2 \in \mathbb{R}^{n\times n}$ be denoted by $\mu_j$. The eigenvalues of $\boldsymbol{P}_1 \otimes \boldsymbol{P}_2$ are the $mn$ numbers $\{\lambda_i\mu_j\}_{i=1,k=1}^{m,n}$.*

*Proof.*

Let $\boldsymbol{J}_1 = \boldsymbol{A}_1^{-1}\boldsymbol{P}_1\boldsymbol{A}_1$ and $\boldsymbol{J}_2 = \boldsymbol{A}_2^{-1}\boldsymbol{P}_2\boldsymbol{A}_2$ be the respective Jordan forms for $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$. We use the mixed-product property and the inverse property of the Kronecker product to show that $\boldsymbol{P}_1 \otimes \boldsymbol{P}_2$ is similar to $\boldsymbol{J}_1 \otimes \boldsymbol{J}_2$:

$$\begin{aligned}\boldsymbol{J}_1 \otimes \boldsymbol{J}_2 &= (\boldsymbol{A}_1^{-1}\boldsymbol{P}_1\boldsymbol{A}_1) \otimes (\boldsymbol{A}_2^{-1}\boldsymbol{P}_2\boldsymbol{A}_2) \\ &= (\boldsymbol{A}_1^{-1} \otimes \boldsymbol{A}_2^{-1})(\boldsymbol{P}_1 \otimes \boldsymbol{P}_2)(\boldsymbol{A}_1 \otimes \boldsymbol{A}_2) \\ &= (\boldsymbol{A}_1 \otimes \boldsymbol{A}_2)^{-1}(\boldsymbol{P}_1 \otimes \boldsymbol{P}_2)(\boldsymbol{A}_1 \otimes \boldsymbol{A}_2).\end{aligned} \tag{A.213}$$

Consequently, the eigenvalues of $\boldsymbol{P}_1 \otimes \boldsymbol{P}_2$ coincide with those of $\boldsymbol{J}_1 \otimes \boldsymbol{J}_2$. Since $\boldsymbol{J}_1$ and $\boldsymbol{J}_2$ are upper triangular with $\{\lambda_i\}_{i=1}^m$ and $\{\mu_j\}_{k=1}^n$ on the diagonals, respectively, $\boldsymbol{J}_1 \otimes \boldsymbol{J}_2$ is also upper triangular with diagonal entries given by $\{\lambda_i\mu_j\}_{i=1,k=1}^{m,n}$.

□

**Proposition F.4.** *Let $\bar{\lambda}_2(\boldsymbol{P}_i)$ denote the second largest eigenvalue in absolute value of the transition matrix $\boldsymbol{P}_i$ associated with the $i$-th client, and define $\lambda(\boldsymbol{P}_i) \triangleq \frac{\bar{\lambda}_2(\boldsymbol{P}_i)+1}{2}$. For the product chain defined by $\boldsymbol{P} = \bigotimes_{k\in\mathcal{K}} \boldsymbol{P}_i$, the second largest eigenvalue in absolute value $\bar{\lambda}_2(\boldsymbol{P})$ and $\lambda(\boldsymbol{P}) \triangleq \frac{\bar{\lambda}_2(\boldsymbol{P})+1}{2}$ satisfy:*

$$\bar{\lambda}_2(\boldsymbol{P}) = \max_{k\in\mathcal{K}} \bar{\lambda}_2(\boldsymbol{P}_i) \quad \textit{and} \quad \lambda(\boldsymbol{P}) = \max_{k\in\mathcal{K}} \lambda(\boldsymbol{P}_i). \tag{A.214}$$

The proof of Proposition F.4 follows a similar structure to the one in (Levin & Peres, 2017, Corollary 12.13). *Proof.*

From Proposition F.3, we know that the eigenvalues of $\boldsymbol{P} = \bigotimes_{k\in\mathcal{K}} \boldsymbol{P}_i$ are given by:

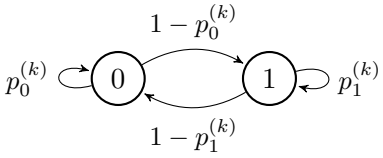
$$\left\{ \prod_{i\in\mathcal{N}} \lambda_i(\boldsymbol{P}_i) : \lambda_i(\boldsymbol{P}_i) \text{ an eigenvalue of } \boldsymbol{P}_i \right\}. \tag{A.215}$$

Recall that $\bar{\lambda}_2(\boldsymbol{P}_i)$ is the second largest eigenvalue of $\boldsymbol{P}_i$ in absolute value. If $k^*$ denotes the index such that $\bar{\lambda}_2(\boldsymbol{P}_{k^*}) = \max_{k\in\mathcal{K}} \bar{\lambda}_2(\boldsymbol{P}_i)$, the second largest eigenvalue in module of $\boldsymbol{P}$ is the product of $\bar{\lambda}_2(\boldsymbol{P}_{k^*})$ for the $k^*$-th client and $\lambda_1(\boldsymbol{P}_j) = 1$ for the remaining clients $j \neq k^*$. The second result in (A.214) follows from the definitions of $\lambda(\boldsymbol{P})$ and $\lambda(\boldsymbol{P}_i)$.

□

### F.C Markov Chain for the Experiments (Section 2.5)

In the experiments (Section 2.5.1), we consider a scenario where the activity of each client $i \in \mathcal{N}$ follows a two-state homogeneous Markov process. The state space $\mathcal{M}$ consists of two states: "inactive" (with value 0) and "active" (with value 1):




We provide detailed expressions of the transition matrix $\boldsymbol{P}_i$, stationary distribution $\boldsymbol{\pi}^{(k)}$, and the second eigenvalue $\lambda_2(\boldsymbol{P}_i)$ used in the experiments for each client $i \in \mathcal{N}$:

$$\boldsymbol{P}_i = \begin{bmatrix} p_0^{(k)} & 1-p_0^{(k)} \\ 1-p_1^{(k)} & p_1^{(k)} \end{bmatrix} = \begin{bmatrix} 1-(1-\lambda_2(\boldsymbol{P}_i))\pi_i & (1-\lambda_2(\boldsymbol{P}_i))\pi_i \\ (1-\lambda_2(\boldsymbol{P}_i))(1-\pi_i) & \lambda_2(\boldsymbol{P}_i)+(1-\lambda_2(\boldsymbol{P}_i))\pi_i \end{bmatrix}. \tag{A.216}$$

$$\boldsymbol{\pi}^{(k)} = [1-\pi_i, \pi_i] = \left[\frac{1-p_1^{(k)}}{2-p_0^{(k)}-p_1^{(k)}}, \frac{1-p_0^{(k)}}{2-p_0^{(k)}-p_1^{(k)}}\right]. \tag{A.217}$$

$$\lambda_2(\boldsymbol{P}_i) = p_0^{(k)} + p_1^{(k)} - 1. \tag{A.218}$$

## G Experimental Evaluation

### G.A Details on Experimental Setup

#### G.A.1 Datasets and Models

In this section, we provide a detailed description of the datasets and models used in our experiments. We considered a total of $N = 100$ clients. We tested `CA-Fed` on the benchmark synthetic LEAF dataset (Caldas et al., 2019) for regularized logistic regression tasks, which satisfy Assumptions 3-4. Additionally, we incorporated two "real-world" datasets: MNIST (L. Deng, 2012) for handwritten digit recognition and CIFAR-10 (Krizhevsky & Hinton, 2009) for image recognition. Detailed descriptions of the datasets and the models used for each of them are provided below.

**Synthetic LEAF dataset** Synthetic data provides us with precise control over heterogeneity. The Synthetic LEAF dataset achieves this by using parameters $\gamma$ and $\delta$, where $\gamma$ determines the degree of variation among local models and $\delta$ determines the variability in the local data across different devices. The generation process follows the setup described in (T. Li, Sahu, Zaheer, et al., 2020; X. Li et al., 2020):

Table A.1: Average computation time and used CPU/GPU for each dataset.

| Dataset | CPU/GPU | Simulation time |
|---|---|---|
| Binary Synthetic | Intel(R) Xeon(R) CPU | 10min |
| Synthetic LEAF | Intel(R) Xeon(R) CPU | 6min |
| MNIST (L. Deng, 2012) | GeForce GTX 1080 Ti | 42min |
| CIFAR10 (Krizhevsky & Hinton, 2009) | GeForce GTX 1080 Ti | 2h37min |

Table A.2: Learning rates $\eta$ and $\bar{\eta}$ used for the experiments in Figure 2.1.

| Dataset | `Unbiased` | `More available` | `CA-Fed` ($\bar{\kappa} = 1$) | `AdaFed` (Tan et al., 2022) | `F3AST` (Ribero et al., 2023) |
|---|---|---|---|---|---|
| Synthetic LEAF | 2.0/2.0 | 1.0/7.0 | 2.0/3.0 | 1.0/1.0 | 2.0/2.0 |
| MNIST | 0.03/1.0 | 0.1/4.0 | 0.1/1.0 | 0.03/1.0 | 0.1/0.3 |
| CIFAR10 | 0.03/1.0 | 0.03/3.0 | 0.03/1.0 | 0.03/1.0 | 0.03/0.3 |

1. For each client $i \in \mathcal{N}$, sample the model parameters $\boldsymbol{W}_i \in \mathbb{R}^{10\times 60}$ and $\boldsymbol{b}_i \in \mathbb{R}^{10}$ from a normal distribution with mean $\mu_i$ and standard deviation 1, where $\mu_i$ is sampled from $\mathcal{N}(0, \gamma)$.
2. For each client $i \in \mathcal{N}$, generate the client's input data $\boldsymbol{X}_i \in \mathbb{R}^{n_i\times 60}$ as follows: sample each element $(x_i)_j$ from a normal distribution with mean $v_i$ and standard deviation $\frac{1}{j^{1.2}}$, where $v_i$ is sampled from $\mathcal{N}(B_i, 1)$ and $B_i$ is sampled from $\mathcal{N}(0, \delta)$.
3. Generate synthetic samples $(\boldsymbol{X}_i, \boldsymbol{Y}_i)$, where $\boldsymbol{Y}_i \in \mathbb{R}^{n_i}$, according to the model $y = \arg\max(\text{softmax}(\boldsymbol{W}_i\boldsymbol{x} + \boldsymbol{b}_i))$, where $\boldsymbol{x} \in \mathbb{R}^{60}$.

The distribution of samples $n_i = |D_i|$ among the clients follows a power law, resulting in an imbalanced data distribution. We refer to the synthetic dataset with parameters $\gamma$ and $\delta$ as synthetic$(\gamma, \delta)$. We set $(\gamma, \delta)$ values to $(0, 0)$, $(0.25, 0.25)$, $(0.5, 0.5)$, $(0.75, 0.75)$, and $(1, 1)$ to investigate various levels of heterogeneity in the data.

**MNIST** To classify handwritten digits in the MNIST dataset, we employ multinomial logistic regression. The model takes a flattened 784-dimensional (28 × 28) image as input and predicts a class label from 0 to 9 as output. To introduce heterogeneity in the data distribution, we distribute the dataset among $N = 100$ clients using a Dirichlet allocation method (H. Wang et al., 2020) with parameter $\varsigma$. This allocation scheme allows for varying proportions of the dataset to be assigned to each client, contributing to the heterogeneous nature of our experimental setting.

**CIFAR-10** The CIFAR-10 dataset consists of 60,000 input images, sourced from a collection of 80 million tiny images, with 10 distinct labels. To partition the CIFAR-10 dataset among $N = 100$ clients, we employ a Dirichlet allocation (H. Wang et al., 2020) with parameter $\varsigma$. For this particular dataset, we train a shallow neural network comprising two convolutional layers followed by one fully connected layer. This network architecture is designed to capture relevant features from the CIFAR-10 images and facilitate accurate classification.

### G.A.2 Implementation Details

**Machines** The experiments were conducted on a CPU/GPU cluster, utilizing various available GPUs such as Nvidia Tesla V100, GeForce GTX 1080 Ti, and Quadro RTX 8000. The majority of experiments involving Synthetic datasets were executed on an Intel(R) Xeon(R) CPU E5-1660 v3 @ 3.00GHz. On the other hand, experiments involving MNIST and CIFAR-10 datasets were performed using GeForce GTX 1080 Ti cards. For each dataset, we conducted approximately 50 experiments, excluding the time dedicated to development and debugging. Due to the usage of a train batch size of 32 samples, the experiments with MNIST and CIFAR-10 datasets exhibited slower execution times. Table A.1 provides the average duration required to execute one simulation for each dataset. The authors are grateful to the OPAL infrastructure from Université Côte d'Azur for providing resources and support.

**Libraries** We extensively employed the PyTorch deep learning framework throughout our experiments. PyTorch provided us with a comprehensive set of tools and functionalities for model construction, training, and evaluation. It allowed us to efficiently implement and optimize various neural network architectures, including the multinomial logistic regression model for the MNIST dataset and the shallow neural network for the CIFAR-10 dataset. To simplify the data preparation process, we utilized Torchvision, a PyTorch package designed for computer vision tasks. Torchvision facilitated seamless dataset management, including the download and pre-processing of MNIST and CIFAR-10, enabling us to transform the raw image data into a suitable format for training and evaluation.

**Hyper-parameters** For each method and task, we performed a grid search to determine the optimal learning rates $\eta$ and $\bar{\eta}$. For the MNIST and CIFAR-10 datasets, we explored the grids $\eta = \{2.0, 1.0, 0.3, 0.1, 0.03, 0.01\}$ and $\bar{\eta} = \{5.0, 4.0, 3.0, 2.0, 1.0, 0.3, 0.1\}$. For the Synthetic LEAF dataset, we shifted the grid to $\bar{\eta} = \{8.0, 7.0, 6.0, 5.0, 4.0, 3.0, 2.0, 1.0\}$. Table A.2 reports the learning rates $\eta$ and $\bar{\eta}$ corresponding to the results in Figure 2.1 for each dataset and method. For `CA-Fed`, we use the hyper-parameters $\beta = \tau = 0$. In the case of `AdaFed`, we set full device participation, where the parameter server samples all active clients ($|\mathcal{S}_t| = |\mathcal{S}^{(t)}|$). To ensure a fair comparison, we set the number of clients sampled by `F3AST` to the average number of clients included by `CA-Fed`, which is 45 on average. Furthermore, we set the smoothness parameter $\beta$ of `F3AST` to be $\mathcal{O}(1/T)$, as suggested by the authors in (Ribero et al., 2023, Appendix D).

# H Further Discussion about CA-Fed

## H.A CA-Fed's computation/communication cost

CA-Fed aims to improve training convergence and not to reduce its computation and communication overhead. Nevertheless, excluding some available clients reduces the overall training cost, as we will discuss in this section referring, for the sake of concreteness, to neural networks' training.

In terms of computation, the available clients not selected for training are only requested to evaluate their local loss on the current model once on a single batch instead than performing $E$ gradient updates, which would require roughly $2 \times E - 1$ more calculations (because of the forward and backward pass). The selected clients have no extra computation cost as computing the loss corresponds to the forward pass they should, in any case, perform during the first local gradient update.

In terms of communication, the excluded clients only transmit the loss, a single scalar, much smaller than the model update. Conversely, participating clients transmit the local loss and the model update. Still, this additional overhead is negligible and likely fully compensated by the communication savings for the excluded clients.

## H.B CA-Fed and Client Sampling

In cross-device FL, a common practice is to employ client sampling, where a small subset of clients (denoted as $\mathcal{S}_t$) is uniformly selected at random from the set of active clients ($\mathcal{S}^{(t)}$) during each communication round of model training. This is primarily done to mitigate communication overhead and enhance scalability.

In our analysis, based on Assumption 1, we assume that spatial and temporal correlations primarily concern clients' availability dynamics and we consider, for simplicity, $\mathcal{S}_t = \mathcal{S}^{(t)}$. However, our findings have a noteworthy implication: while the set of available clients $\mathcal{S}^{(t)}$ exhibits correlation, the client sampling in $\mathcal{S}_t$ can be designed to make clients' participation dynamics independent over time and among clients. A promising direction for future research is to extend our work in this context and derive a refined bound similar to our result in Theorem B.19 which quantifies the impact of client sampling on $\lambda(\boldsymbol{P})$.

Consistent with our analysis, we have designed our algorithm to align with the assumption $\mathcal{S}_t = \mathcal{S}^{(t)}$. By design, CA-Fed excludes clients with large temporal correlation and low availability and activates, in each communication round, only clients satisfying $\{i \in \mathcal{S}^{(t)};\ q_i^{(t)} > 0\}$ (line 8 in Algorithm 3). However, when only a small fraction of clients is excluded, CA-Fed seamlessly integrates with client sampling. This only involves replacing $\mathcal{S}^{(t)}$ with $\mathcal{S}_t$ in Equation (2.22) and Algorithm 3 (server estimates for clients' local losses ($\hat{\boldsymbol{F}}^{(t)} = (\hat{F}_i^{(t)})_{i\in\mathcal{N}}$) are now updated from the sampled clients' losses ($\boldsymbol{F}^{(t)} = (F_i^{(t)})_{k\in\mathcal{S}_t}$)).

## H.C About CA-Fed's fairness

Strategies that exclude clients from the training phase, such as CA-Fed, may raise concerns about fairness. The concept of *fairness* in federated learning does not have a unified definition in the literature (Ludwig & Baracaldo, 2022, Chapter 8). Fairness goals can be established by appropriately selecting the target weights $\boldsymbol{\alpha} = \{\alpha_i\}_{i\in\mathcal{N}}$ in the definition of the global target objective (2.1). For instance, *per-client fairness* can be achieved by setting $\alpha_i$ to be equal for every client (i.e., $\alpha_i = 1/N$), while *per-sample fairness* can be accomplished by setting $\alpha_i$ proportional to the local dataset size $|D_i|$ (i.e, $\alpha_i = |D_i|/|D|$).

Assuming that the global objective in (2.1) truly reflects fairness concerns, then CA-Fed can be considered intrinsically fair. This is because CA-Fed continually focuses on minimizing the total error $\epsilon \triangleq F(\boldsymbol{w}_T) - F^*$, which

guarantees that the performance objective of the learned model is as close as possible to its optimal value at every time. Although `CA-Fed` occasionally excludes clients with low availability and high temporal correlation, the optimization problem (2.1) is carefully designed to ensure that the learned model performs well for these clients. As a result, `CA-Fed` effectively learns a model that is consistently accurate and fair across all clients, regardless of their availability or temporal correlation.

# APPENDIX B

# Variance Reduction: leveraging Stale Updates for Non-Participating Clients

## A FedStale, Upper bound

### A.A Preliminaries

In this section, we provide an overview of the `FedStale` algorithm and establish the necessary notation used throughout this supplementary material.

#### Algorithm Description

The algorithm's structure, as outlined in Algorithm 7, is detailed below:

We detail some modifications from the main text, introduced to streamline notation for this proof:

1. In Algorithm 7, we assume that *all* clients partake in the local optimization step and compute $\boldsymbol{g}_i^{(t)}$. However, the server aggregates only the model updates from participating clients (where $\xi_i^{(t)} = 1$). This assumption simplifies notation and is equivalent to a scenario where only participating clients return their model updates to the server.
2. To condense notation, we normalize the client update $\Delta_i^{(t)}$ by the client learning rate $\eta_c$ and the number of local iterations $K$. This results in rescaling the client update $\Delta_i^{(t)} = (\boldsymbol{w}_i^{(t,0)} - \boldsymbol{w}_i^{(t,E)})$ by $\eta_c$ and $K$. The server update step is then rewritten as $\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} - \eta\Delta^{(t)}$, where $\eta = \eta_s\eta_c K$ represents an "equivalent" learning rate at the server level.
3. We explicitly write the participation indicator function $\xi_i^{(t)}$ in the server update $\Delta^{(t)}$. This formulation not only brings transparency to the notation but also allows for a more clear understanding of the `FedAvg`, `FedVARP`, and `FedStale` updates:

$$\Delta_{\texttt{FedAvg}}^{(t)} = \frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(t)}}{p_i}\boldsymbol{g}_i^{(t)} \tag{B.1}$$

**Algorithm 7:** `FedStale` (Federated Learning with Stale Client Updates) – Appendix

**for** *each round* $t = 1, \dots, T$ **do**
  **for** *all clients* $i = 1, \dots, N$, *in parallel* **do**
    Initialize $\boldsymbol{w}_i^{(t,0)} = \boldsymbol{w}^{(t)}$
    **for** *local iterations* $k = 0, \dots, K-1$ **do**
      Sample data $\mathcal{B}_i^{(t,k)} \overset{\text{iid}}{\sim} \mathcal{D}_i$
      Compute stochastic gradient $\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})$
      Update $\boldsymbol{w}_i^{(t,k+1)} = \boldsymbol{w}_i^{(t,k)} - \eta_c \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})$
    Compute and return $\boldsymbol{g}_i^{(t)} = \frac{1}{\eta_c K}(\boldsymbol{w}_i^{(t,0)} - \boldsymbol{w}_i^{(t,E)})$ to the server
  Aggregate client updates $\Delta^{(t)} = \frac{1}{N}\sum_{i=1}^N \beta \boldsymbol{h}_i^{(t)} + \frac{1}{N}\sum_{i=1}^N \frac{\xi_i^{(t)}}{p_i}(\boldsymbol{g}_i^{(t)} - \beta \boldsymbol{h}_i^{(t)})$
  Update global model $\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} - \eta \Delta^{(t)}_{\texttt{FedStale}}$, $\eta = \eta_s \eta_c K$
  At the server level, update memory $\forall i, \boldsymbol{h}_i^{(t+1)} = \begin{cases} \boldsymbol{g}_i^{(t)} & \text{if } \xi_i^{(t)} = 1 \\ \boldsymbol{h}_i^{(t)} & \text{otherwise} \end{cases}$

$$\Delta^{(t)}_{\texttt{FedVARP}} = \frac{1}{N}\sum_{i=1}^N \boldsymbol{h}_i^{(t)} + \frac{1}{N}\sum_{i=1}^N \frac{\xi_i^{(t)}}{p_i}\left(\boldsymbol{g}_i^{(t)} - \boldsymbol{h}_i^{(t)}\right) \tag{B.2}$$

$$\Delta^{(t)}_{\texttt{FedStale}} = (1-\beta)\Delta^{(t)}_{\texttt{FedAvg}} + \beta \Delta^{(t)}_{\texttt{FedVARP}} \tag{B.3}$$

$$= \frac{1}{N}\sum_{i=1}^N \beta \boldsymbol{h}_i^{(t)} + \frac{1}{N}\sum_{i=1}^N \frac{\xi_i^{(t)}}{p_i}\left(\boldsymbol{g}_i^{(t)} - \beta \boldsymbol{h}_i^{(t)}\right) \tag{B.4}$$

$$= \frac{1}{N}\sum_{i=1}^N \frac{\xi_i^{(t)}}{p_i}\boldsymbol{g}_i^{(t)} - \frac{\beta}{N}\sum_{i=1}^N \left(\frac{\xi_i^{(t)}}{p_i} - 1\right)\boldsymbol{h}_i^{(t)} \tag{B.5}$$

The comparison of Equations (B.1)–(B.5) allows for the following considerations:

1. `FedVARP`'s update (Eq. (B.2)) recovers `FedAvg`'s update (Eq. (B.1)) when:
   (a) All clients participate in the current round ($\xi_i^{(t)} = 1, \forall i$), or
   (b) All memory terms are set to zero ($\boldsymbol{h}_i^{(t)} = 0, \forall i$).
2. `FedStale`'s update can be rewritten in three different forms (Equations (B.3)–(B.5)), each offering a unique perspective:
   (a) Eq. (B.3) interprets `FedStale`'s update as a convex combination of `FedAvg`'s update (Eq. (B.1)) and `FedVARP`'s update (Eq. (B.2)), where $\beta$ is the parameter of the convex combination;
   (b) Eq. (B.4) relates `FedStale`'s update to `FedVARP` (Eq. (B.2)), where $\beta$ acts as a weight for the memory terms $\{\boldsymbol{h}_i^{(t)}, \forall i\}$;

(c) Eq. (B.5) frames `FedStale`'s in relation to `FedAvg`(Eq. (B.1)), introducing the memory term $\boldsymbol{h}_i^{(t)}$ whenever client $i$ does not participate and subtracting cumulative memory terms $\boldsymbol{h}_i^{(t)}/p_i$ when client $i$ does participate again.

The normalized client update $\boldsymbol{g}_i^{(t)}$ is the average of $K$ local stochastic gradients computed by client $i$ during round $t$. We denote it as *local stochastic pseudo-gradient*:

**Remark A.1.** The local update $\boldsymbol{g}_i^{(t)}$ can be considered as a local stochastic pseudo-gradient:

$$\boldsymbol{g}_i^{(t)} = \frac{1}{\eta_c K}\Delta_i^{(t)} = \frac{1}{\eta_c K}(\boldsymbol{w}_i^{(t,0)} - \boldsymbol{w}_i^{(t,E)}) = \frac{1}{K}\sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}). \tag{B.6}$$

*Proof.*

Unroll the recursion $\boldsymbol{w}_i^{(t,k+1)} = \boldsymbol{w}_i^{(t,k)} - \eta_c \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})$ for $k = 0, \ldots, K-1$.

□

**Additional Notation**

$$\text{Local Stochastic Pseudo-Gradient:} \quad \boldsymbol{g}_i^{(t)} = \frac{1}{K}\sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}); \tag{B.7}$$

$$\text{Local Pseudo-Gradient:} \quad \bar{\boldsymbol{g}}_i^{(t)} = \frac{1}{K}\sum_{k=0}^{K-1} \nabla F_i(\boldsymbol{w}_i^{(t,k)}); \tag{B.8}$$

$$\text{Global Stochastic Pseudo-Gradient:} \quad \boldsymbol{g}^{(t)} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{g}_i^{(t)}; \tag{B.9}$$

$$\text{Global Pseudo-Gradient:} \quad \bar{\boldsymbol{g}}^{(t)} = \frac{1}{N}\sum_{i=1}^{N} \bar{\boldsymbol{g}}_i^{(t)}; \tag{B.10}$$

$$\text{Global Stale Pseudo-Gradient:} \quad \boldsymbol{h}^{(t)} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{h}_i^{(t)}. \tag{B.11}$$

**Main Assumptions**

**Assumption 2.** *The gradients of $F_i(\boldsymbol{w})$ are L-Lipschitz continuous, $\forall \boldsymbol{w}, i$.*

**Assumption 3.** *The stochastic gradients are unbiased: $\mathbb{E}_{\mathcal{B}\sim\mathcal{D}_i}[\nabla F_i(\boldsymbol{w}, \mathcal{B})] = \nabla F_i(\boldsymbol{w})$ with bounded variance: $\mathbb{E}_{\mathcal{B}\sim\mathcal{D}_i} \|\nabla F_i(\boldsymbol{w}, \mathcal{B}) - \nabla F_i(\boldsymbol{w})\|^2 \leq \sigma^2$, $\forall \boldsymbol{w}, i$.*

**Assumption 4.** *The divergence between local and global gradients is uniformly bounded: $\|\nabla F_i(\boldsymbol{w}) - \nabla F(\boldsymbol{w})\|^2 \leq \sigma_g^2$, $\forall \boldsymbol{w}, i$.*

**Assumption 5.** *The client participation outcomes follow a Bernoulli distribution with parameter $p_i$, i.e., $\xi_i^{(t)} \sim$ Bern$(p_i)$.*

**Sources of Randomness**

In this system, we model two sources of randomness. The first arises from the partial and heterogeneous participation of clients, which follows a Bernoulli distribution as stated in Assumption 5. The second source of randomness originates from the random sampling of data points for stochastic gradients computation. Recall that $\mathbb{1}^{(t)}$ denotes the random set of clients participating at the $t$-th communication round and that $\xi_i^{(t,k)}$ denotes the random data point independently sampled from client-$i$'s local dataset at round $t$, local iteration $k$. For the analysis, we introduce the following additional notation:

$\mathbb{1}^{(s:q)} := \{\mathbb{1}^{(s)}, \ldots, \mathbb{1}^{(q)}\}$: the random set of clients participating from the $s$-th to the $q$-th communication rounds, $s<q$;

$\xi_i^{(t)} := \{\xi_i^{(t,k)}\}_{k=0}^{K-1}$: the set of random batches sampled by the $i$-th client at the $t$-th communication round;

$\xi^{(t)} := \{\xi_i^{(t)}\}_{i\in\mathbb{1}^{(t)}}$: the set of random batches sampled by the participating clients ($\mathbb{1}^{(t)}$) in the $t$-th round;

$\xi_i^{(t,s:q)} := \{\xi_i^{(t,s)}, \ldots, \xi_i^{(t,q)}\}$: the set of random batches sampled by the $i$-th client at the $t$-th communication round between the $s$-th and the $q$-th local iterations, $s < q$;

$\xi^{(s:q)} := \{\xi^{(s)}, \ldots, \xi^{(q)}\}$: the set of random batches sampled by the available clients ($\mathbb{1}^{(s:q)}$) between the $s$-th and $q$-th communication rounds, $s < q$.

With this notation established, the randomness in the $t$-th communication round, which starts with the initial model $\boldsymbol{w}^{(t)}$ and yields the updated model $\boldsymbol{w}^{(t+1)}$, is fully captured by the sets $\mathbb{1}^{(t)}$ and $\xi^{(t)}$. Thus, the stochastic progression of our algorithm from the first round to round $t$ can be comprehensively described by the tuple:

$$\mathcal{H}^{(t)} := \left(\mathbb{1}^{(1)}, \ldots, \mathbb{1}^{(t-1)}; \mathcal{B}^{(1)}, \ldots, \mathcal{B}^{(t-1)}\right), \tag{B.12}$$

which represents the historical information up to the $t$-th communication round.

**Remark A.2.** For any algorithm, $\texttt{Algm} \in [\texttt{FedAvg}, \texttt{FedVARP}, \texttt{FedStale}]$, the global pseudo-gradient $\Delta_{\texttt{Algm}}^{(t)}$ is unbiased with respect to both sources of randomness—client participation and stochastic gradients:

$$\mathbb{E}_{\mathbb{1}^{(t)}|\mathcal{B}^{(t)},\mathcal{H}^{(t)}}[\Delta_{\texttt{Algm}}^{(t)}] = \boldsymbol{g}^{(t)},$$
$$\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}[\Delta_{\texttt{Algm}}^{(t)}] = \bar{\boldsymbol{g}}^{(t)},$$

and consequently, the global model $\boldsymbol{w}^{(t+1)}$ is also unbiased:

$$\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}[\boldsymbol{w}^{(t+1)}] = \boldsymbol{w}^{(t)} - \eta\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}[\Delta_{\texttt{Algm}}^{(t)}] = \boldsymbol{w}^{(t)} - \eta\bar{\boldsymbol{g}}^{(t)}.$$

## A.B Supporting Lemmas

In this section, we present key lemmas that underpin the theoretical analysis and facilitate the proof of Theorem A.12.

**Lemma A.1** (Descent lemma)**.** *Let $F : \mathbb{R}^d \to \mathbb{R}$ be an $L$-smooth function (Assumption 2), optimized via the sequence of parameters $\{\boldsymbol{w}^{(t)}\}$. At each iteration $t$, an SGD update is made according to a learning rate $\eta$ and a stochastic gradient $\Delta^{(t)}$. Let $\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[\Delta^{(t)}\right] = \bar{\boldsymbol{g}}^{(t)}$. Then, the expected reduction in $F$ after one iteration is bounded by:*

$$\begin{aligned}&\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[F(\boldsymbol{w}^{(t+1)})\right]\\ &\leq F(\boldsymbol{w}^{(t)}) - \frac{\eta}{2}\left[\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 + \left\|\bar{\boldsymbol{g}}^{(t)}\right\|^2 - \left\|\bar{\boldsymbol{g}}^{(t)} - \nabla F(\boldsymbol{w}^{(t)})\right\|^2\right] + \frac{\eta^2 L}{2}\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\Delta^{(t)}\right\|^2.\end{aligned} \tag{B.13}$$

*Proof of Lemma A.1.*

By the $L$-smoothness of $F$, it follows that:

$$F(\boldsymbol{w}^{(t+1)}) \leq F(\boldsymbol{w}^{(t)}) + \langle \nabla F(\boldsymbol{w}^{(t)}), \boldsymbol{w}^{(t+1)} - \boldsymbol{w}^{(t)} \rangle + \frac{L}{2} \left\| \boldsymbol{w}^{(t+1)} - \boldsymbol{w}^{(t)} \right\|^2 \tag{B.14}$$

$$\leq F(\boldsymbol{w}^{(t)}) - \eta \langle \nabla F(\boldsymbol{w}^{(t)}), \Delta^{(t)} \rangle + \frac{\eta^2 L}{2} \left\| \Delta^{(t)} \right\|^2, \tag{B.15}$$

where Eq. (B.15) applies the update rule $\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} - \eta \Delta^{(t)}$.

Taking expectation over the randomness at the $t$-th round, due to client participation (inherent in $\xi^{(t)}$) and to stochastic gradients (inherent in $\mathcal{B}^{(t)} := \{\mathcal{B}_i^{(t,k)}\}_{i,k}$), yields:

$$\mathbb{E}_{\mathbb{1}^{(t)}, \mathcal{B}^{(t)} | \mathcal{H}^{(t)}} \left[ F(\boldsymbol{w}^{(t+1)}) \right]$$

$$\leq F(\boldsymbol{w}^{(t)}) - \eta \mathbb{E}_{\mathbb{1}^{(t)}, \mathcal{B}^{(t)} | \mathcal{H}^{(t)}} \langle \nabla F(\boldsymbol{w}^{(t)}), \Delta^{(t)} \rangle + \frac{\eta^2 L}{2} \mathbb{E}_{\mathbb{1}^{(t)}, \mathcal{B}^{(t)} | \mathcal{H}^{(t)}} \left\| \Delta^{(t)} \right\|^2 \tag{B.16}$$

$$\leq F(\boldsymbol{w}^{(t)}) - \eta \left[ \langle \nabla F(\boldsymbol{w}^{(t)}), \bar{\boldsymbol{g}}^{(t)} \rangle \right] + \frac{\eta^2 L}{2} \mathbb{E}_{\mathbb{1}^{(t)}, \mathcal{B}^{(t)} | \mathcal{H}^{(t)}} \left\| \Delta^{(t)} \right\|^2 \tag{B.17}$$

$$\leq F(\boldsymbol{w}^{(t)}) - \frac{\eta}{2} \left[ \left\| \nabla F(\boldsymbol{w}^{(t)}) \right\|^2 + \left\| \bar{\boldsymbol{g}}^{(t)} \right\|^2 - \left\| \bar{\boldsymbol{g}}^{(t)} - \nabla F(\boldsymbol{w}^{(t)}) \right\|^2 \right] + \frac{\eta^2 L}{2} \mathbb{E}_{\mathbb{1}^{(t)}, \mathcal{B}^{(t)} | \mathcal{H}^{(t)}} \left\| \Delta^{(t)} \right\|^2, \tag{B.18}$$

where Eq. (B.17) uses $\mathbb{E}_{\mathbb{1}^{(t)}, \mathcal{B}^{(t)} | \mathcal{H}^{(t)}} \left[ \Delta^{(t)} \right] = \bar{\boldsymbol{g}}^{(t)}$ and Eq. (B.18) applies the identity $\|\boldsymbol{a} - \boldsymbol{b}\|^2 = \|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2 - 2\langle \boldsymbol{a}, \boldsymbol{b} \rangle$.

□

**Lemma A.2** (Expected value of the local stochastic pseudo-gradients)**.** *Let $\boldsymbol{g}_i^{(t)}$ and $\bar{\boldsymbol{g}}_i^{(t)}$ be defined in Eqs. (B.7) and (B.8), respectively. If the stochastic gradients are unbiased (Assumption 3), the following identity holds:*

$$\mathbb{E}_{\mathcal{B}_i^{(t)} | \mathcal{H}^{(t)}} \left[ \boldsymbol{g}_i^{(t)} \right] = \bar{\boldsymbol{g}}_i^{(t)}. \tag{B.19}$$

*Proof of Lemma A.2.*

*We observe that the randomness in the iterate for a specific client, $\boldsymbol{w}_i^{(t,k)}$, is influenced both by the sequence of events up to time $t$ (denoted as $\mathcal{H}^{(t)}$) and by the random batches used for training up to the $k$-th iteration ($\mathcal{B}_i^{(t,0:k-1)}$). We then rely on a fundamental property of expectations to decompose the expected value of the gradient $\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})$ as:*

$$\mathbb{E}_{\mathcal{B}_i^{(t,0:k)} | \mathcal{H}^{(t)}} \left[ \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) \right] = \mathbb{E}_{\mathcal{B}_i^{(t,0:k-1)} | \mathcal{H}^{(t)}} \left[ \mathbb{E}_{\mathcal{B}_i^{(t,k)} | \mathcal{B}_i^{(t,0:k-1)}, \mathcal{H}^{(t)}} \left[ \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) \right] \right]. \tag{B.20}$$

*We finally use Assumption 3 to conclude that $\mathbb{E}_{\mathcal{B}_i^{(t,k)} | \mathcal{B}_i^{(t,0:k-1)}, \mathcal{H}^{(t)}} [\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)})] = \nabla F_i(\boldsymbol{w}_i^{(t,k)})$.*

*Below, we present the detailed derivations of the proof.*

$$\mathbb{E}_{\mathcal{B}_i^{(t)} | \mathcal{H}^{(t)}} \left[ \boldsymbol{g}_i^{(t)} \right] = \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)} | \mathcal{H}^{(t)}} \left[ \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) \right] \tag{B.21}$$

$$= \frac{1}{K}\Bigg[\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t,0)}|\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}^{(t)}, \mathcal{B}_i^{(t,0)})\right]}_{\text{bounded by Assumption 3}} + \mathbb{E}_{\mathcal{B}_i^{(t,0)},\mathcal{B}_i^{(t,1)}|\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,1)}, \mathcal{B}_i^{(t,1)})\right] + \cdots +$$

$$+ \cdots + \mathbb{E}_{\mathcal{B}_i^{(t,0:K-1)}|\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,K-1)}, \mathcal{B}_i^{(t,K-1)})\right]\Bigg] \tag{B.22}$$

$$= \frac{1}{K}\Bigg[\nabla F_i(\boldsymbol{w}^{(t)}) + \mathbb{E}_{\mathcal{B}_i^{(t,0)}|\mathcal{H}^{(t)}}\underbrace{\left[\mathbb{E}_{\mathcal{B}_i^{(t,1)}|\mathcal{B}_i^{(t,0)},\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,1)}, \mathcal{B}_i^{(t,1)})\right]\right]}_{\text{bounded by Assumption 3}} + \cdots +$$

$$+ \cdots + \mathbb{E}_{\mathcal{B}_i^{(t,0:K-2)}|\mathcal{H}^{(t)}}\underbrace{\left[\mathbb{E}_{\mathcal{B}_i^{(t,K-1)}|\mathcal{B}_i^{(t,0:K-2)},\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,K-1)}, \mathcal{B}_i^{(t,K-1)})\right]\right]}_{\text{bounded by Assumption 3}}\Bigg] \tag{B.23}$$

$$= \frac{1}{K}\left[\nabla F_i(\boldsymbol{w}^{(t)}) + \mathbb{E}_{\mathcal{B}_i^{(t,0)}|\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,1)})\right] + \cdots + \mathbb{E}_{\mathcal{B}_i^{(t,0:K-2)}|\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,K-1)})\right]\right] \tag{B.24}$$

$$= \frac{1}{K}\sum_{k=0}^{K-1}\nabla F_i(\boldsymbol{w}_i^{(t,k)}) = \bar{\boldsymbol{g}}_i^{(t)}, \tag{B.25}$$

*where Eq. (B.21) uses the definition of $\boldsymbol{g}_i^{(t)}$ given in (B.7), Eq. (B.22) makes explicit the dependency of the iterate $\boldsymbol{w}_i^{(t,k)}$ on the random batches $\mathcal{B}_i^{(t,0:k-1)}$, Eq. (B.23) uses the law of total expectation given in (B.20), Eq. (B.24) applies the unbiasedness of the stochastic gradient (Assumption 3), and Eq. (B.25) uses the definition of $\bar{\boldsymbol{g}}_i^{(t)}$ given in (B.8).*

□

**Lemma A.3** (Variance of the local stochastic pseudo-gradients)**.** *Let $\boldsymbol{g}_i^{(t)}$ and $\bar{\boldsymbol{g}}_i^{(t)}$ be defined as in Eqs. (B.7) and (B.8), respectively. If the variance of the local stochastic gradients is bounded by $\sigma^2$ (Assumption 3), the following inequality holds:*

$$\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{g}_i^{(t)} - \bar{\boldsymbol{g}}_i^{(t)}\right\|^2 \leq \frac{\sigma^2}{K}. \tag{B.26}$$

*Proof of Lemma A.3.*

*The proof builds on similar observations to those presented in Lemma A.2, but additionally relies on the bounded variance of local stochastic gradients (Assumption 3).*

*Below, the detailed derivations.*

$$\begin{aligned}
&\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{g}_i^{(t)} - \bar{\boldsymbol{g}}_i^{(t)}\right\|^2 \\
&= \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\frac{1}{K}\sum_{k=0}^{K-1}\left[\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) - \nabla F_i(\boldsymbol{w}_i^{(t,k)})\right]\right\|^2 \\
&= \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) - \nabla F_i(\boldsymbol{w}_i^{(t,k)})\right\|^2
\end{aligned} \tag{B.27}$$

$$+\frac{1}{K^2}\sum_{k=0}^{K-1}\sum_{\substack{k'=0\\k'\neq k}}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\langle\nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})-\nabla F_i(\boldsymbol{w}_i^{(t,k)}),\nabla F_i(\boldsymbol{w}_i^{(t,k')},\mathcal{B}_i^{(t,k')})-\nabla F_i(\boldsymbol{w}_i^{(t,k')})\rangle, \quad \text{(B.28)}$$

*where Eq. (B.27) applies the definitions for $\boldsymbol{g}_i^{(t)}$ and $\bar{\boldsymbol{g}}_i^{(t)}$ given in (B.7) and (B.8), and Eq. (B.28) expands the squared norm. To show that the second term in (B.28) is zero, we use the law of total expectation in a similar way as in (B.20). Indeed, denote $k''=\max\{k,k'\}$. The following relation holds:*

$$\begin{aligned}&\mathbb{E}_{\mathcal{B}_i^{(t,0:k'')}|\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,k'')},\mathcal{B}_i^{(t,k'')})-\nabla F_i(\boldsymbol{w}_i^{(t,k'')})\right]\\&=\mathbb{E}_{\mathcal{B}_i^{(t,0:k''-1)}|\mathcal{H}^{(t)}}\Bigg[\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t,k'')}|\mathcal{B}_i^{(t,0:k''-1)},\mathcal{H}^{(t)}}\left[\nabla F_i(\boldsymbol{w}_i^{(t,k'')},\mathcal{B}_i^{(t,k'')})-\nabla F_i(\boldsymbol{w}_i^{(t,k'')})\right]}_{=0 \text{ by Assumption 3}}\Bigg]=0.\end{aligned} \quad \text{(B.29)}$$

*Therefore, only the first term remains:*

$$\begin{aligned}&\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)}\right\|^2\\&=\frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,k)},\mathcal{B}_i^{(t,k)})-\nabla F_i(\boldsymbol{w}_i^{(t,k)})\right\|^2 \quad \text{(B.30)}\\&=\frac{1}{K^2}\Bigg[\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t,0)}|\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}^{(t)},\mathcal{B}_i^{(t,0)})-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2}_{\leq\sigma^2 \text{ by Assumption 3}}+\mathbb{E}_{\mathcal{B}_i^{(t,0)},\mathcal{B}_i^{(t,1)}|\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,1)},\mathcal{B}_i^{(t,1)})-\nabla F_i(\boldsymbol{w}_i^{(t,1)})\right\|^2\\&\quad+\cdots+\mathbb{E}_{\mathcal{B}_i^{(t,0:K-1)}|\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,K-1)},\mathcal{B}_i^{(t,K-1)})-\nabla F_i(\boldsymbol{w}_i^{(t,K-1)})\right\|^2\Bigg] \quad \text{(B.31)}\\&\leq\frac{1}{K^2}\Bigg[\sigma^2+\mathbb{E}_{\mathcal{B}_i^{(t,0)}|\mathcal{H}^{(t)}}\underbrace{\left[\mathbb{E}_{\mathcal{B}_i^{(t,1)}|\mathcal{B}_i^{(t,0)},\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,1)},\mathcal{B}_i^{(t,1)})-\nabla F_i(\boldsymbol{w}_i^{(t,1)})\right\|^2\right]}_{\leq\sigma^2 \text{ by Assumption 3}}\\&\quad+\cdots+\mathbb{E}_{\mathcal{B}_i^{(t,0:K-2)}|\mathcal{H}^{(t)}}\underbrace{\left[\mathbb{E}_{\mathcal{B}_i^{(t,K-1)}|\mathcal{B}_i^{(t,0:K-2)},\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}_i^{(t,K-1)},\mathcal{B}_i^{(t,K-1)})-\nabla F_i(\boldsymbol{w}_i^{(t,K-1)})\right\|^2\right]}_{\leq\sigma^2 \text{ by Assumption 3}}\Bigg] \quad \text{(B.32)}\\&\leq\frac{1}{K^2}\sum_{k=0}^{K-1}\sigma^2=\frac{\sigma^2}{K}, \quad \text{(B.33)}\end{aligned}$$

*where Eq. (B.30) uses (B.29), Eq. (B.31) explicits the dependency of the iterate $\boldsymbol{w}_i^{(t,k)}$ on the random batches $\mathcal{B}_i^{(t,0:k-1)}$, Eq. (B.32) applies the law of total expectation given in (B.20), and Eq. (B.33) uses the uniform bound on the variance of local stochastic gradients (Assumption 3).*

□

**Lemma A.4** (Variance of the global stochastic pseudo-gradient)**.** *Let $\boldsymbol{g}_i^{(t)}$ and $\bar{\boldsymbol{g}}_i^{(t)}$ be defined as in Eqs. (B.7) and (B.8), respectively. Assuming that client participation outcomes ($\xi_i^{(t)}$) are Bernoulli-distributed with parameter $p_i$, and that the variance of the local stochastic gradients is bounded by $\sigma^2$ (Assumption 3), the following inequality*

*holds:*

$$\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(t)}}{p_i}\left(\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)}\right)\right\|^2\leq\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)\frac{\sigma^2}{NK}. \tag{B.34}$$

*Proof of Lemma A.4.*

The proof starts by expanding the squared norm of the average stochastic gradient deviations into a variance term accounting for individual client gradients and a covariance term between gradients from different clients:

$$\begin{aligned}&\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(t)}}{p_i}\left(\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)}\right)\right\|^2\\&=\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[\frac{1}{N^2}\sum_{i=1}^{N}\frac{\left[\xi_i^{(t)}\right]^2}{p_i^2}\left\|\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)}\right\|^2+\frac{1}{N^2}\sum_{i=1}^{N}\sum_{\substack{i'=1\\i'\neq i}}^{N}\frac{\xi_i^{(t)}\xi_{i'}^{(t)}}{p_ip_{i'}}\langle\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)},\boldsymbol{g}_{i'}^{(t)}-\bar{\boldsymbol{g}}_{i'}^{(t)}\rangle\right].\end{aligned} \tag{B.35}$$

We leverage the linearity of expectation, the independence of client participation ($\xi^{(t)}$) and batch sampling ($\mathcal{B}^{(t)}$), the independence of batch sampling among clients ($\mathcal{B}_i^{(t)}$ and $\mathcal{B}_{i'}^{(t)}$), and Lemma A.2 to show that:

$$\begin{aligned}&\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[\frac{\xi_i^{(t)}\xi_{i'}^{(t)}}{p_ip_{i'}}\langle\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)},\boldsymbol{g}_{i'}^{(t)}-\bar{\boldsymbol{g}}_{i'}^{(t)}\rangle\right]\\&=\frac{\mathbb{E}_{\xi^{(t)}|\mathcal{H}^{(t)}}[\xi_i^{(t)}\xi_{i'}^{(t)}]}{p_ip_{i'}}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[\langle\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)},\boldsymbol{g}_{i'}^{(t)}-\bar{\boldsymbol{g}}_{i'}^{(t)}\rangle\right] &\text{(B.36)}\\&=\frac{\mathbb{E}_{\xi^{(t)}|\mathcal{H}^{(t)}}[\xi_i^{(t)}\xi_{i'}^{(t)}]}{p_ip_{i'}}\langle\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}[\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)}]}_{=0\text{ by Lemma A.2}},\underbrace{\mathbb{E}_{\mathcal{B}_{i'}^{(t)}|\mathcal{H}^{(t)}}[\boldsymbol{g}_{i'}^{(t)}-\bar{\boldsymbol{g}}_{i'}^{(t)}]}_{=0\text{ by Lemma A.2}}\rangle=0. &\text{(B.37)}\end{aligned}$$

Finally, we bound the remaining term using Lemma A.3:

$$\begin{aligned}\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(t)}}{p_i}\left(\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)}\right)\right\|^2&=\frac{1}{N^2}\sum_{i=1}^{N}\frac{\mathbb{E}_{\xi_i^{(t)}|\mathcal{H}^{(t)}}\left[\left(\xi_i^{(t)}\right)^2\right]}{p_i^2}\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)}\right\|^2}_{\text{bounded in Lemma A.3}} &\text{(B.38)}\\&\leq\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)\frac{\sigma^2}{NK}, &\text{(B.39)}\end{aligned}$$

where Equation (B.38) derives from (B.37) and requires the independence of client participation ($\xi^{(t)}$) and batch sampling ($\mathcal{B}^{(t)}$); Equation (B.39) replaces the Bernoulli's second-order moment ($p_i$) and applies Lemma A.3.

□

**Lemma A.5** (Client drift due to multiple local iterations)**.** *Under bounded local stochastic gradient variance ($\sigma^2$, as per Assumption 3) and the client learning rate $\eta_c\leq\frac{1}{2LK}$, the expected squared deviation of a client's pseudo-gradient ($\bar{\boldsymbol{g}}_i^{(t)}$) from its local gradient ($\nabla F_i(\boldsymbol{w}^{(t)})$) is bounded as:*

$$\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)}-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2\leq 2\eta_c^2L^2K(K-1)\left[\frac{\sigma^2}{K}+2\left\|\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2\right]. \tag{B.40}$$

*Additionally, if the variance of local gradients is uniformly bounded across clients (by $\sigma_g^2$, as per Assumption 4):*

$$\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)}) \right\|^2 \leq 2\eta_c^2 L^2 K(K-1) \left[ \frac{\sigma^2}{K} + 4\sigma_g^2 + 4 \left\| \nabla F(\boldsymbol{w}^{(t)}) \right\|^2 \right]. \tag{B.41}$$

The bound in Eq. (B.41) captures that, when the number of local iterations $K$ equals 1, $\bar{\boldsymbol{g}}_i^{(t)}$ and $\nabla F_i(\boldsymbol{w}^{(t)})$ become equivalent. *Proof of Lemma A.5.*

This proof is borrowed from (J. Wang et al., 2020), (Jhunjhunwala et al., 2022, Lemma 6). It is included for completeness.

The proof starts by replacing the definition of $\bar{\boldsymbol{g}}_i^{(t)}$ given in (B.8):

$$\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)}) \right\|^2 = \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \frac{1}{K} \sum_{k=0}^{K-1} \left( \nabla F_i(\boldsymbol{w}_i^{(t,k)}) - \nabla F_i(\boldsymbol{w}^{(t)}) \right) \right\|^2 \tag{B.42}$$

$$\leq \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \nabla F_i(\boldsymbol{w}_i^{(t,k)}) - \nabla F_i(\boldsymbol{w}^{(t)}) \right\|^2 \tag{B.43}$$

$$\leq \frac{L^2}{K} \sum_{k=0}^{K-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t)} \right\|^2, \tag{B.44}$$

where Eq. (B.43) follows from the Jensen's inequality; Eq. B.44 uses the $L$-smoothness of local gradients (Assumption 2).

Next, the individual difference is bounded as:

$$\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \boldsymbol{w}_i^{(t,k)} - \boldsymbol{w}^{(t)} \right\|^2$$

$$= \eta_c^2 \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \sum_{k'=0}^{k-1} \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) \right\|^2 \tag{B.45}$$

$$= \eta_c^2 \left[ \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \sum_{k'=0}^{k-1} \left[ \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}) \right] \right\|^2 + \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \sum_{k'=0}^{k-1} \nabla F_i(\boldsymbol{w}_i^{(t,k')}) \right\|^2 \right] \tag{B.46}$$

$$\leq \eta_c^2 \Bigg[ \underbrace{\sum_{k'=0}^{k-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}_i^{(t,k')}) \right\|^2}_{\text{bounded by Assumption 3, in a similar way as Lemma A.3}} + k \sum_{k'=0}^{k-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \nabla F_i(\boldsymbol{w}_i^{(t,k')}) \right\|^2 \Bigg] \tag{B.47}$$

$$\leq \eta_c^2 \left[ k\sigma^2 + k \sum_{k'=0}^{k-1} \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \nabla F_i(\boldsymbol{w}_i^{(t,k')}) - \nabla F_i(\boldsymbol{w}^{(t)}) + \nabla F_i(\boldsymbol{w}^{(t)}) \right\|^2 \right] \tag{B.48}$$

$$\leq \eta_c^2 \left[ k\sigma^2 + 2k \sum_{k'=0}^{k-1} \left[ L^2 \mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}} \left\| \boldsymbol{w}_i^{(t,k')} - \boldsymbol{w}^{(t)} \right\|^2 + \left\| \nabla F_i(\boldsymbol{w}^{(t)}) \right\|^2 \right] \right], \tag{B.49}$$

where Eq. (B.45) applies the local update rule $\boldsymbol{w}_i^{(t,k)} = \boldsymbol{w}^{(t)} - \eta_c \sum_{k'=0}^{k-1} \nabla F_i(\boldsymbol{w}_i^{(t,k')}, \mathcal{B}_i^{(t,k')})$; Eq. (B.46) leverages the local stochastic gradient unbiasedness (as per Lemma A.2) and its bias-variance decomposition; Eq. (B.47) involves squaring the former term, zeroing the cross terms as per (B.29), and applying Jensen's inequality to the latter term; Eq. (B.48) accounts for the bounded variance of local stochastic gradients in the former term (Assumption 3,

as in Lemma A.3), and modifies the latter term by adding and subtracting the initial local gradient ($\nabla F_i(\boldsymbol{w}^{(t)})$); finally, Eq. (B.49) uses the norm inequality ($\|\boldsymbol{a}+\boldsymbol{b}\|^2 \leq 2\|\boldsymbol{a}\|^2+2\|\boldsymbol{b}\|^2$) and the $L$-smoothness of local objectives (Assumption 2).

Summing over $k=0,\ldots,K-1$, it yields:

$$\begin{aligned}
&\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{w}_i^{(t,k)}-\boldsymbol{w}^{(t)}\right\|^2 \\
&\leq \frac{\eta_c^2\sigma^2}{K}\sum_{k=0}^{K-1}k+\frac{2\eta_c^2L^2}{K}\sum_{k=0}^{K-1}k\sum_{k'=0}^{k-1}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{w}_i^{(t,k')}-\boldsymbol{w}^{(t)}\right\|^2+\frac{2\eta_c^2}{K}\sum_{k=0}^{K-1}k\sum_{k'=0}^{k-1}\left\|\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2 \qquad \text{(B.50)}\\
&\leq \eta_c^2(K-1)\sigma^2+2\eta_c^2L^2K(K-1)\left[\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{w}_i^{(t,k)}-\boldsymbol{w}^{(t)}\right\|^2\right]+2\eta_c^2K(K-1)\left\|\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2, \qquad \text{(B.51)}
\end{aligned}$$

where Eq. (B.51) uses $\sum_{k'=0}^{k-1}\left\|\boldsymbol{w}_i^{(t,k')}-\boldsymbol{w}^{(t)}\right\|^2 \leq \sum_{k=0}^{K-1}\left\|\boldsymbol{w}_i^{(t,k)}-\boldsymbol{w}^{(t)}\right\|^2$ and $\sum_{k=0}^{K-1}k=\frac{1}{2}(K-1)K$.

Define $D \coloneqq 2\eta_c^2L^2K(K-1)$. Choose $\eta_c$ small enough such that $D \leq 1/2$ ($\Rightarrow \eta_c \leq \frac{1}{2LK}$). Rearranging the terms:

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{w}_i^{(t,k)}-\boldsymbol{w}^{(t)}\right\|^2 \leq \frac{\eta_c^2(K-1)\sigma^2}{1-D}+\frac{2\eta_c^2K(K-1)}{1-D}\left\|\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2 \qquad \text{(B.52)}$$

Substituting (B.52) back into (B.44):

$$\begin{aligned}
\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)}-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2 &\leq \frac{D}{2(1-D)}\frac{\sigma^2}{K}+\frac{D}{1-D}\left\|\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2 \qquad \text{(B.53)}\\
&\leq D\frac{\sigma^2}{K}+2D\left\|\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2, \qquad \text{(B.54)}
\end{aligned}$$

where Eq. (B.54) uses $D \leq 1/2$. Replacing $D \coloneqq 2\eta_c^2L^2K(K-1)$ into (B.54) completes the proof of Inequality (B.40).
Additionally, inequality (B.41) removes the dependency on $\nabla F_i(\boldsymbol{w}^{(t)})$ by adding and subtracting $\nabla F(\boldsymbol{w}^{(t)})$ in the squared norm:

$$\begin{aligned}
\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)}-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2 &\leq D\frac{\sigma^2}{K}+2D\left\|\nabla F_i(\boldsymbol{w}^{(t)})-\nabla F(\boldsymbol{w}^{(t)})+\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \qquad \text{(B.55)}\\
&\leq D\frac{\sigma^2}{K}+4D\left\|\nabla F_i(\boldsymbol{w}^{(t)})-\nabla F(\boldsymbol{w}^{(t)})\right\|^2+4D\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \qquad \text{(B.56)}\\
&\leq D\frac{\sigma^2}{K}+4D\sigma_g^2+4D\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2, \qquad \text{(B.57)}
\end{aligned}$$

where Eq. (B.56) uses the norm inequality ($\|\boldsymbol{a}+\boldsymbol{b}\|^2 \leq 2\|\boldsymbol{a}\|^2+2\|\boldsymbol{b}\|^2$) and Eq. (B.57) leverages the uniform variance bound of local gradients across clients ($\sigma_g^2$, from Assumption 4).

Replacing $D \coloneqq 2\eta_c^2L^2K(K-1)$ into (B.57) concludes the proof of inequality (B.41).

□

**Lemma A.6** (Variance of `FedStale`'s update)**.** *Let $\Delta^{(t)}$ denote `FedStale`'s global update with randomness from client participation ($\xi^{(t)}$) and batch sampling ($\mathcal{B}$), and $\bar{\boldsymbol{g}}^{(t)}$ its unbiased counterpart, as per Eqs. (B.4) and (B.10), respectively. Under Assumptions 2–5, we bound the variance of `FedStale`'s pseudo-gradient, due to partial participation and batch sampling, as:*

$$
\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}} \left\|\Delta^{(t)} - \bar{\boldsymbol{g}}^{(t)}\right\|^2 \\
&\leq \frac{6(1-\beta)^2}{N}\left[\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2}_{\text{uniformly bounded in Lemma A.5}} + \left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2 + \left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2\right] \\
&+ \frac{6\beta^2}{N}\left[\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2}_{\text{uniformly bounded in Lemma A.5}} + \eta^2 L^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left\|\Delta^{(t-1)}\right\|^2 + N\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)H^{(t)}\right] \\
&+ \left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)\frac{\sigma^2}{NK}.
\end{aligned} \tag{B.58}
$$

*Proof of Lemma A.6.*

The proof starts by substituting the definitions for $\Delta^{(t)}$ and $\bar{\boldsymbol{g}}^{(t)}$, given in (B.4) and (B.10):

$$
\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}} \left\|\Delta^{(t)} - \bar{\boldsymbol{g}}^{(t)}\right\|^2 \\
&= \mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}} \left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(t)}}{p_i}\left(\boldsymbol{g}_i^{(t)} - \beta\boldsymbol{h}_i^{(t)}\right) - \frac{1}{N}\sum_{i=1}^{N}\left(\bar{\boldsymbol{g}}_i^{(t)} - \beta\boldsymbol{h}_i^{(t)}\right)\right\|^2 && \text{(B.59)} \\
&= \mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}} \left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(t)}}{p_i}\left(\boldsymbol{g}_i^{(t)} - \bar{\boldsymbol{g}}_i^{(t)} + \bar{\boldsymbol{g}}_i^{(t)} - \beta\boldsymbol{h}_i^{(t)}\right) - \frac{1}{N}\sum_{i=1}^{N}\left(\bar{\boldsymbol{g}}_i^{(t)} - \beta\boldsymbol{h}_i^{(t)}\right)\right\|^2 && \text{(B.60)} \\
&= \mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}} \left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(t)}}{p_i}\left(\bar{\boldsymbol{g}}_i^{(t)} - \beta\boldsymbol{h}_i^{(t)}\right) - \frac{1}{N}\sum_{i=1}^{N}\left(\bar{\boldsymbol{g}}_i^{(t)} - \beta\boldsymbol{h}_i^{(t)}\right)\right\|^2 \\
&\quad + \underbrace{\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}} \left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(t)}}{p_i}\left(\boldsymbol{g}_i^{(t)} - \bar{\boldsymbol{g}}_i^{(t)}\right)\right\|^2}_{\text{bounded by Lemma A.4}}, && \text{(B.61)}
\end{aligned}
$$

where Eq. (B.59) uses definitions (B.4) and (B.10); Eq. (B.60) involves adding and subtracting $\bar{\boldsymbol{g}}_i^{(t)}$ within the squared norm; Eq. (B.61) is based on $\bar{\boldsymbol{g}}_i^{(t)}$ being an unbiased estimator of $\boldsymbol{g}_i^{(t)}$ (Lemma A.2). The latter term is bounded by Lemma A.4.

Conditioning on $\mathcal{H}^{(t)}$, $\boldsymbol{h}_i^{(t)}$ is constant, and the first term in (B.61) represents a variance, due to client participation ($\xi^{(t)}$) and stochastic gradients ($\mathcal{B}^{(t)}$). Moreover, conditioning on $\mathcal{B}^{(t)}$, the randomness in Eq. (B.62) is only due to client participation ($\xi^{(t)}$):

$$
\mathbb{V}\mathrm{ar}_{\xi^{(t)}|\mathcal{B}^{(t)},\mathcal{H}^{(t)}}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(t)}}{p_i}\left(\bar{\boldsymbol{g}}_i^{(t)} - \beta\boldsymbol{h}_i^{(t)}\right)\right)
$$

$$= \mathbb{V}\mathrm{ar}_{\xi^{(t)}|\mathcal{B}^{(t)},\mathcal{H}^{(t)}}\left(\frac{1}{N}\frac{\xi_i^{(t)}}{p_i}\left[(1-\beta)\bar{\boldsymbol{g}}_i^{(t)} + \beta\left(\bar{\boldsymbol{g}}_i^{(t)} - \boldsymbol{h}_i^{(t)}\right)\right]\right) \tag{B.62}$$

$$= \frac{1}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|(1-\beta)\bar{\boldsymbol{g}}_i^{(t)} + \beta\left(\bar{\boldsymbol{g}}_i^{(t)} - \boldsymbol{h}_i^{(t)}\right)\right\|^2 \tag{B.63}$$

$$\leq \underbrace{\frac{2(1-\beta)^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\bar{\boldsymbol{g}}_i^{(t)}\right\|^2}_{T_1} + \underbrace{\frac{2\beta^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \boldsymbol{h}_i^{(t)}\right\|^2}_{T_2}, \tag{B.64}$$

where Eq. (B.62) adds and subtracts $\beta\bar{\boldsymbol{g}}_i^{(t)}$, then rearranges terms for $\beta$; Eq. (B.63) derives from the Bernoulli variance ($\mathrm{Var}(\xi_i^{(t)}) = p_i(1-p_i)$), under Assumption 5; Eq. (B.64) leverages the norm inequality $\|\boldsymbol{a}+\boldsymbol{b}\|^2 \leq 2\|\boldsymbol{a}\|^2 + 2\|\boldsymbol{b}\|^2$.

We proceed by bounding the first term of Eq. (B.64) as follows:

$$T_1 = \frac{2(1-\beta)^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\bar{\boldsymbol{g}}_i^{(t)}\right\|^2$$

$$= \frac{2(1-\beta)^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)}) + \nabla F_i(\boldsymbol{w}^{(t)}) - \nabla F(\boldsymbol{w}^{(t)}) + \nabla F(\boldsymbol{w}^{(t)})\right\|^2 \tag{B.65}$$

$$\leq \frac{6(1-\beta)^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left[\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2 + \underbrace{\left\|\nabla F_i(\boldsymbol{w}^{(t)}) - \nabla F(\boldsymbol{w}^{(t)})\right\|^2}_{\leq \sigma_g^2 \text{ by Assumption 4}} + \left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2\right] \tag{B.66}$$

$$\leq \frac{6(1-\beta)^2}{N}\left[\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\underbrace{\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2}_{\text{bounded in expectation by Lemma A.5}} + \left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2 + \left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2\right], \tag{B.67}$$

where Eq. (B.65) adds and subtracts the local gradient ($\nabla F_i(\boldsymbol{w}^{(t)})$) and the global gradient ($\nabla F_i(\boldsymbol{w}^{(t)})$) within the squared norm; Eq. (B.66) leverages the norm inequality $\|\boldsymbol{a}+\boldsymbol{b}+\boldsymbol{c}\|^2 \leq 3\|\boldsymbol{a}\|^2 + 3\|\boldsymbol{b}\|^2 + 3\|\boldsymbol{c}\|^2$; Eq. (B.67) applies Assumption 4.

We separately bound the second term of (B.64) as follows:

$$T_2 = \frac{2\beta^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \boldsymbol{h}_i^{(t)}\right\|^2$$

$$= \frac{2\beta^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)}) + \nabla F_i(\boldsymbol{w}^{(t)}) - \nabla F_i(\boldsymbol{w}^{(t-1)}) + \nabla F_i(\boldsymbol{w}^{(t-1)}) - \boldsymbol{h}_i^{(t)}\right\|^2 \tag{B.68}$$

$$= \frac{6\beta^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left[\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2 + \left\|\nabla F_i(\boldsymbol{w}^{(t)}) - \nabla F_i(\boldsymbol{w}^{(t-1)})\right\|^2 + \left\|\nabla F_i(\boldsymbol{w}^{(t-1)}) - \boldsymbol{h}_i^{(t)}\right\|^2\right] \tag{B.69}$$

$$\leq \frac{6\beta^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2 + \frac{6\beta^2 L^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left\|\boldsymbol{w}^{(t)} - \boldsymbol{w}^{(t-1)}\right\|^2$$
$$+ \frac{6\beta^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\nabla F_i(\boldsymbol{w}^{(t-1)}) - \boldsymbol{h}_i^{(t)}\right\|^2 \tag{B.70}$$

$$\leq \frac{6\beta^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\bar{\boldsymbol{g}}_i^{(t)}-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2+\frac{6\beta^2\eta^2L^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left\|\Delta^{(t-1)}\right\|^2$$
$$+6\beta^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\underbrace{\left(\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(t-1)})-\boldsymbol{h}_i^{(t)}\right\|^2\right)}_{\triangleq H^{(t)}} \tag{B.71}$$

$$\leq \frac{6\beta^2}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\underbrace{\left\|\bar{\boldsymbol{g}}_i^{(t)}-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2}_{\text{bounded in expectation by Lemma A.5}}+\frac{6\beta^2\eta^2L^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left\|\Delta^{(t-1)}\right\|^2$$
$$+6\beta^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)H^{(t)}, \tag{B.72}$$

where Eqs. (B.68) and (B.69) follow the steps of Eqs. (B.65) and (B.66); Eq. (B.70) is based on the $L$-smoothness of local objectives (Assumption 2); Eq. (B.71) applies the inequality $\sum_{i=1}^{N}a_ib_i\leq(\sum_{i=1}^{N}a_i)(\sum_{i=1}^{N}b_i)$ for positive $a_i$ and $b_i$; Eq. (B.72) defines

$$H^{(t)}\triangleq\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(t-1)})-\boldsymbol{h}_i^{(t)}\right\|^2. \tag{B.73}$$

Finally, the bound in Eq. (B.58) combines Eqs. (B.61), (B.67), and (B.72).

□

**Lemma A.7** (Bound on the memory term)**.** *Let $H^{(t)}$, the divergence between the local gradient and the historical pseudo-gradient at time t, be defined in Eq. (B.73). Under Assumptions 2, 3, and 5, the expected historical error $H^{(t+1)}$ is recursively bounded as:*

$$\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[H^{(t+1)}\right]\leq\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\frac{\sigma^2}{K}+\frac{1}{N}\sum_{i=1}^{N}p_i\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)}-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2$$
$$+\eta^2L^2\left(1+\frac{1}{C}\right)\left(1-\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left\|\Delta^{(t-1)}\right\|^2+(1+C)\left(1-p_{min}\right)H^{(t)}. \tag{B.74}$$

*Proof of Lemma A.7.*

The proof starts by definition of $H^{(t+1)}$:

$$\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[H^{(t+1)}\right]$$
$$=\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left[\mathbb{E}_{\xi_i^{(t)}|\mathcal{B}_i^{(t)},\mathcal{H}^{(t)}}\left[\left\|\nabla F_i(\boldsymbol{w}^{(t)})-\boldsymbol{h}_i^{(t+1)}\right\|^2\right]\right] \tag{B.75}$$
$$=\frac{1}{N}\sum_{i=1}^{N}\left[p_i\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}^{(t)})-\boldsymbol{g}_i^{(t)}\right\|^2+(1-p_i)\left\|\nabla F_i(\boldsymbol{w}^{(t)})-\boldsymbol{h}_i^{(t)}\right\|^2\right] \tag{B.76}$$
$$\leq\frac{1}{N}\sum_{i=1}^{N}p_i\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{g}_i^{(t)}-\bar{\boldsymbol{g}}_i^{(t)}\right\|^2}_{\text{bounded by Lemma A.3}}+\frac{1}{N}\sum_{i=1}^{N}p_i\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)}-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2$$

$$+\frac{\left(1+\frac{1}{C}\right)}{N}\sum_{i=1}^{N}(1-p_i)\left\|\nabla F_i(\boldsymbol{w}^{(t)})-\nabla F_i(\boldsymbol{w}^{(t-1)})\right\|^2+\frac{(1+C)}{N}\sum_{i=1}^{N}(1-p_i)\left\|\nabla F_i(\boldsymbol{w}^{(t-1)})-\boldsymbol{h}_i^{(t)}\right\|^2 \tag{B.77}$$

$$\leq\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\frac{\sigma^2}{K}+\frac{1}{N}\sum_{i=1}^{N}p_i\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)}-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2}_{\text{uniformly bounded by Lemma A.5}}$$

$$+\frac{\eta^2L^2\left(1+\frac{1}{C}\right)}{N}\sum_{i=1}^{N}(1-p_i)\left\|\Delta^{(t-1)}\right\|^2+\frac{(1+C)}{N}\sum_{i=1}^{N}(1-p_i)\left\|\nabla F_i(\boldsymbol{w}^{(t-1)})-\boldsymbol{h}_i^{(t)}\right\|^2, \tag{B.78}$$

where Eq. (B.75) uses the law of total expectation to separate expectations on client participation ($\xi_i^{(t)}$) and batch sampling ($\mathcal{B}_i^{(t)}$); Eq. (B.76) solves the inner expectation with respect to client participation ($\xi_i^{(t)}$); Eq. (B.77) manipulates the first term by adding and subtracting $\bar{\boldsymbol{g}}_i^{(t)}$, then leverages the bounded variance of the local stochastic pseudo-gradients (Lemma A.3, Assumption 3), and similarly corrects the second term with $\nabla F_i(\boldsymbol{w}^{(t-1)})$, then applies the norm inequality $\|\boldsymbol{a}+\boldsymbol{b}\|^2\leq(1+\frac{1}{C})\|\boldsymbol{a}\|^2+(1+C)\|\boldsymbol{b}\|^2$ for any positive $C$; Eq. (B.78) is derived from the $L$-smoothness property of local objectives (Assumption 2).

The final expression in Eq. (B.74) is derived by observing that $\sum_{i=1}^{N}(1-a_i)b_i\leq(1-a_{\min})\sum_{i=1}^{N}b_i$.

□

**Lemma A.8** (Variance of `FedStale`'s update - Initial condition)**.** *Denote $\Delta^{(1)}$ and $\bar{\boldsymbol{g}}^{(1)}$ in Eq. (B.4) and (B.10), respectively. Under Assumptions 3–5, we bound the initial variance of `FedStale` update, due to partial participation and batch sampling, as:*

$$\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left\|\Delta^{(1)}-\bar{\boldsymbol{g}}^{(1)}\right\|^2\leq\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)\frac{\sigma^2}{NK}$$

$$+\frac{3}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\underbrace{\mathbb{E}_{\mathcal{B}^{(1)}}\left\|\bar{\boldsymbol{g}}_i^{(1)}-\nabla F_i(\boldsymbol{w}^{(1)})\right\|^2}_{\textit{uniformly bounded by Lemma A.5}}+\frac{3}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2+\frac{3}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2. \tag{B.79}$$

*Proof of Lemma A.8.*

Similarly to Lemma A.6, we start by the definitions of $\Delta^{(1)}$ and $\bar{\boldsymbol{g}}^{(1)}$ given in Eqs. (B.4) and (B.10):

$$\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left\|\Delta^{(1)}-\bar{\boldsymbol{g}}^{(1)}\right\|^2$$

$$=\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(1)}}{p_i}\left(\boldsymbol{g}_i^{(1)}-\beta\boldsymbol{h}_i^{(1)}\right)-\frac{1}{N}\sum_{i=1}^{N}\left(\bar{\boldsymbol{g}}_i^{(1)}-\beta\boldsymbol{h}_i^{(1)}\right)\right\|^2 \tag{B.80}$$

$$=\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(1)}}{p_i}\bar{\boldsymbol{g}}_i^{(1)}-\frac{1}{N}\sum_{i=1}^{N}\bar{\boldsymbol{g}}_i^{(1)}\right\|^2+\underbrace{\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(1)}}{p_i}\left(\boldsymbol{g}_i^{(1)}-\bar{\boldsymbol{g}}_i^{(1)}\right)\right\|^2}_{\text{bounded by Lemma A.3}}, \tag{B.81}$$

where Eq. (B.80) adds and subtracts $\bar{\boldsymbol{g}}_i^{(1)}$ within the squared norm; Eq. (B.81) is based on $\bar{\boldsymbol{g}}_i^{(1)}$ being an unbiased estimator of $\boldsymbol{g}_i^{(1)}$ (Lemma A.2, Assumption 3). The latter term is bounded by Lemma A.3.

Similarly to Eq. (B.62), we observe that the first term in Eq. (B.81) represents a variance. Conditioning on the first batch sample ($\mathcal{B}^{(1)}$), we solve the expectation with respect to initial client participation ($\mathbb{1}^{(1)}$):

$$\mathbb{V}\mathrm{ar}_{\mathbb{1}^{(1)}|\mathcal{B}^{(1)}}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^{(1)}}{p_i}\bar{\boldsymbol{g}}_i^{(1)}\right)$$

$$= \frac{1}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left\|\bar{\boldsymbol{g}}_i^{(1)}\right\|^2 \tag{B.82}$$

$$\leq \frac{3}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\left[\left\|\bar{\boldsymbol{g}}_i^{(1)}-\nabla F_i(\boldsymbol{w}^{(1)})\right\|^2+\underbrace{\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\nabla F(\boldsymbol{w}^{(1)})\right\|^2}_{\leq\sigma_g^2 \text{ by Assumption 4}}+\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2\right] \tag{B.83}$$

$$\leq \frac{3}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\underbrace{\left\|\bar{\boldsymbol{g}}_i^{(1)}-\nabla F_i(\boldsymbol{w}^{(1)})\right\|^2}_{\text{bounded in expectation by Lemma A.5}}+\frac{3}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2+\frac{3}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2, \tag{B.84}$$

where Eq. (B.82) derives from the Bernoulli variance ($\mathrm{Var}(\xi_i^{(1)})=p_i(1-p_i)$), under Assumption 5; Eq. (B.83) adds and subtracts the local and global initial gradients ($\nabla F_i(\boldsymbol{w}^{(1)})$ and $\nabla F(\boldsymbol{w}^{(1)})$), then leverages the norm inequality $\|\boldsymbol{a}+\boldsymbol{b}+\boldsymbol{c}\|^2\leq 3\|\boldsymbol{a}\|^2+3\|\boldsymbol{b}\|^2+3\|\boldsymbol{b}\|^2$; finally, Eq. (B.84) uses Assumption 4.

□

**Lemma A.9** (Bound on the memory term - Initial condition)**.** *Let $H^{(1)}$, the initial error due to the historical pseudo-gradients, be defined in Eq.* (B.73)*. Under Assumptions 3 and 5, the expected error $H^{(2)}$ is bounded as:*

$$\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left[H^{(2)}\right]\leq\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\frac{\sigma^2}{K}+\frac{1}{N}\sum_{i=1}^{N}p_i\underbrace{\mathbb{E}_{\mathcal{B}_i^{(1)}}\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\bar{\boldsymbol{g}}_i^{(1)}\right\|^2}_{\textit{uniformly bounded by Lemma A.5}}+(1-p_{min})\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\boldsymbol{h}_i^{(1)}\right\|^2. \tag{B.85}$$

*Proof of Lemma A.9.*

Similarly to Lemma A.7, the proof starts with the definition of $H^{(2)}$:

$$\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left[H^{(2)}\right]$$

$$= \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{\mathcal{B}_i^{(1)}}\left[\mathbb{E}_{\xi_i^{(1)}|\mathcal{B}_i^{(1)}}\left[\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\boldsymbol{h}_i^{(2)}\right\|^2\right]\right] \tag{B.86}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\left[p_i\mathbb{E}_{\mathcal{B}_i^{(1)}}\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\boldsymbol{g}_i^{(1)}\right\|^2+(1-p_i)\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\boldsymbol{h}_i^{(1)}\right\|^2\right] \tag{B.87}$$

$$= \frac{1}{N}\sum_{i=1}^{N}p_i\underbrace{\mathbb{E}_{\mathcal{B}_i^{(1)}}\left\|\boldsymbol{g}_i^{(1)}-\bar{\boldsymbol{g}}_i^{(1)}\right\|^2}_{\text{bounded by Lemma A.3}}+\frac{1}{N}\sum_{i=1}^{N}p_i\underbrace{\mathbb{E}_{\mathcal{B}_i^{(1)}}\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\bar{\boldsymbol{g}}_i^{(1)}\right\|^2}_{\text{uniformly bounded by Lemma A.5}}+\frac{1}{N}\sum_{i=1}^{N}(1-p_i)\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\boldsymbol{h}_i^{(1)}\right\|^2, \tag{B.88}$$

where Eq. (B.86) uses the law of total expectation to separate expectations on client participation ($\xi_i^{(1)}$) and batch sampling ($\mathcal{B}_i^{(1)}$); Eq. (B.87) solves the inner expectation with respect to client participation ($\xi_i^{(1)}$); Eq. (B.88) adds and subtracts $\bar{\boldsymbol{g}}_i^{(1)}$ to the first term, then leverages the local pseudo-gradients' unbiased property (Lemma A.2, Assumption 3) to separate the two components.

The expression in Eq. (B.85) is finally achieved by observing that $\sum_{i=1}^N (1-a_i)b_i \leq (1-a_{\min})\sum_{i=1}^N b_i$.

□

**Lemma A.10** (`FedStale`: Per Round Progress)**.** *Under Assumptions 2–5, and appropriate client and server learning rates*

$$\eta_c \leq \frac{1}{8LK} \;\wedge\; \eta_s \leq \min\left\{ \frac{N}{12(1-\beta)^2\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right)}, \frac{2N}{3\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right)}, \frac{1}{3\beta^2\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right)\left(\frac{p_{avg}}{p_{min}}\right)} \right\}, \tag{B.89}$$

*define the following Lyapunov function including the objective value, squared global pseudo-gradient, and historical error term:*

$$\psi^{(t+1)} \coloneqq F(\boldsymbol{w}^{(t+1)}) + \frac{\eta^2 L}{2}\left\|\Delta^{(t)}\right\|^2 + 12\eta^2 L\beta^2\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right)\left(\frac{1}{p_{min}}\right)H^{(t+1)}. \tag{B.90}$$

*We bound `FedStale`'s per-round performance into a progress term—accounting for the decrement in the objective value—and a deviation term—from the stochastic gradient noise and data heterogeneity:*

$$\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}[\psi^{(t+1)}] \\
&\leq \psi^{(t)} - \frac{\eta}{4}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \\
&\quad + \left[\frac{\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^N \frac{1}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^N p_i\right)\left(\frac{1}{p_{min}}\right)\right]\frac{\sigma^2}{K} \\
&\quad + \left[\frac{\eta}{2} + \frac{6\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^N p_i\right)\left(\frac{1}{p_{min}}\right)\right]2\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \\
&\quad + \frac{6\eta^2 L(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right)\sigma_g^2 \\
&\quad + \left[\frac{\eta}{2} + \frac{6\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^N \frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^N p_i\right)\left(\frac{1}{p_{min}}\right)\right]8\eta_c^2 L^2 K(K-1)\sigma_g^2.
\end{aligned} \tag{B.91}$$

*Proof of Lemma A.10.*

We introduce the following Lyapunov function, also adopted by (Jhunjhunwala et al., 2022), for any $\frac{\eta^2 L}{2} < \delta \leq \frac{\eta}{2}$ and $\alpha \geq 0$:

$$\psi^{(t+1)} \coloneqq F(\boldsymbol{w}^{(t+1)}) + \left(\delta - \frac{\eta^2 L}{2}\right)\left\|\Delta^{(t)}\right\|^2 + \alpha \underbrace{\frac{1}{N}\sum_{i=1}^N \left\|\nabla F_i(\boldsymbol{w}^{(t)}) - \boldsymbol{h}_i^{(t+1)}\right\|^2}_{\triangleq H^{(t+1)}}. \tag{B.92}$$

Considering expectation over the randomness at the $t$-th round and invoking the standard descent lemma for smooth objectives (Assumption 2 and Lemma A.1):

$$\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}[\psi^{(t+1)}]$$

$$= \mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[F(\boldsymbol{w}^{(t+1)}) + \left(\delta - \frac{\eta^2 L}{2}\right)\left\|\Delta^{(t)}\right\|^2 + \frac{\alpha}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(t)}) - \boldsymbol{h}_i^{(t+1)}\right\|^2\right] \tag{B.93}$$

$$\leq F(\boldsymbol{w}^{(t)}) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 - \frac{\eta}{2}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}^{(t)}\right\|^2 + \frac{\eta}{2}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}^{(t)} - \nabla F(\boldsymbol{w}^{(t)})\right\|^2$$
$$+ \frac{\eta^2 L}{2}\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\Delta^{(t)}\right\|^2 + \left(\delta - \frac{\eta^2 L}{2}\right)\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\Delta^{(t)}\right\|^2 + \frac{\alpha}{N}\sum_{i=1}^{N}\mathbb{E}_{\xi_i^{(t)},\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}^{(t)}) - \boldsymbol{h}_i^{(t+1)}\right\|^2 \tag{B.94}$$

$$\leq F(\boldsymbol{w}^{(t)}) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 - \frac{\eta}{2}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}^{(t)}\right\|^2 + \frac{\eta}{2N}\sum_{i=1}^{N}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2$$
$$+ \delta\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\Delta^{(t)} - \bar{\boldsymbol{g}}^{(t)} + \bar{\boldsymbol{g}}^{(t)}\right\|^2 + \frac{\alpha}{N}\sum_{i=1}^{N}\mathbb{E}_{\xi_i^{(t)},\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\nabla F_i(\boldsymbol{w}^{(t)}) - \boldsymbol{h}_i^{(t+1)}\right\|^2 \tag{B.95}$$

$$\leq F(\boldsymbol{w}^{(t)}) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 + \left(\delta - \frac{\eta}{2}\right)\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}^{(t)}\right\|^2 + \frac{\eta}{2N}\sum_{i=1}^{N}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2$$
$$+ \delta\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\Delta^{(t)} - \bar{\boldsymbol{g}}^{(t)}\right\|^2 + \alpha\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(t)}) - \boldsymbol{h}_i^{(t+1)}\right\|^2\right] \tag{B.96}$$

$$\leq F(\boldsymbol{w}^{(t)}) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 + \frac{\eta}{2N}\sum_{i=1}^{N}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2$$
$$+ \delta\underbrace{\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\Delta^{(t)} - \bar{\boldsymbol{g}}^{(t)}\right\|^2}_{\text{bounded by Lemma A.6}} + \alpha\underbrace{\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left[H^{(t+1)}\right]}_{\text{bounded by Lemma A.7}}, \tag{B.97}$$

where Eq. (B.94) applies Lemma A.1; Eq. (B.95) follows from Jensen's inequality, and introduces the global pseudo-gradient $\bar{\boldsymbol{g}}^{(t)}$; Eq. (B.96) requires $\Delta^{(t)}$ as an unbiased estimator for $\bar{\boldsymbol{g}}^{(t)}$; and Eq. (B.97) holds for $\delta \leq \frac{\eta}{2}$.

Next, we apply Lemmas A.6 and A.7 into Eq. (B.97):

$$\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}[\psi^{(t+1)}] \\
&\leq F(\boldsymbol{w}^{(t)}) \\
&\quad + \left[\frac{6\delta\beta^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + \alpha\left(1+\frac{1}{C}\right)\left(1-\frac{1}{N}\sum_{i=1}^{N}p_i\right)\right]\eta^2 L^2\left\|\Delta^{(t-1)}\right\|^2 \\
&\quad + \left[6\delta\beta^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + \alpha\left(1+C\right)\left(1-p_{\min}\right)\right]H^{(t)} \\
&\quad - \frac{\eta}{2}\left[1 - \frac{12\delta(1-\beta)^2}{\eta N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\right]\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \\
&\quad + \frac{\eta}{2N}\sum_{i=1}^{N}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2 + \frac{6\delta}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2
\end{aligned}$$

$$+\frac{\alpha}{N}\sum_{i=1}^{N}p_i\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)}-\nabla F_i(\boldsymbol{w}^{(t)})\right\|^2$$
$$+\left[\frac{\delta}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)+\alpha\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\right]\frac{\sigma^2}{K}$$
$$+\frac{6\delta(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2, \tag{B.98}$$

where Eq. (B.98) is derived by straightforward reordering of terms.

The initial segment of Eq. (B.98)—comprising the objective value, squared global pseudo-gradient norm, and historical error at round $t$—qualifies for bounding within the Lyapunov recursive framework. The conditions for this recursion step are:

$$\left[\frac{6\delta\beta^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)+\alpha\left(1+\frac{1}{C}\right)\left(1-\frac{1}{N}\sum_{i=1}^{N}p_i\right)\right]\eta^2L^2\leq\delta-\frac{\eta^2L}{2}; \tag{B.99}$$

$$6\delta\beta^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)+\alpha\left(1+C\right)\left(1-p_{\min}\right)\leq\alpha. \tag{B.100}$$

Reversing Eq. (B.100), the resulting condition on $\alpha$ is:

$$\alpha\geq\frac{6\delta\beta^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)}{\left[1-\left(1+C\right)\left(1-p_{\min}\right)\right]}. \tag{B.101}$$

To ensure Eq. (B.101) is positive ($\alpha>0$), a suitable choice for $C$ must satisfy $[1-(1+C)(1-p_{\min})]>0$:

$$C<\frac{p_{\min}}{1-p_{\min}}\quad\Longrightarrow\quad\text{choose}\quad C=\frac{p_{\min}}{2(1-p_{\min})}.$$

It follows that $1+C=\frac{2-p_{\min}}{2(1-p_{\min})}$, and

$$\alpha=12\delta\beta^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{p_{\min}}\right). \tag{B.102}$$

Reversing Eq. (B.99), the resulting condition on $\delta$ is:

$$\left[\frac{6\delta\beta^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)+\alpha\left(1+\frac{1}{C}\right)\left(1-\frac{1}{N}\sum_{i=1}^{N}p_i\right)\right]\eta^2L^2\leq\delta-\frac{\eta^2L}{2}. \tag{B.103}$$

By replacing $\alpha$ (as per Eq. (B.102)) and $1+\frac{1}{C}=\frac{2-p_{\min}}{p_{\min}}$ into (B.103), we have:

$$\frac{6\delta\beta^2\eta^2L^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)+\frac{12\delta\beta^2\eta^2L^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(1-\frac{1}{N}\sum_{i=1}^{N}p_i\right)(2-p_{\min})}{(p_{\min})^2}\leq\delta-\frac{\eta^2L}{2}, \tag{B.104}$$

therefore:

$$\delta\geq\frac{\frac{\eta^2L}{2}}{1-\frac{6\beta^2\eta^2L^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)-\frac{12\beta^2\eta^2L^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(1-\frac{1}{N}\sum_{i=1}^{N}p_i\right)(2-p_{\min})}{(p_{\min})^2}}. \tag{B.105}$$

Choosing $\delta = \eta^2 L$ requires the following constraints on the step-size:

$$\frac{6\beta^2\eta^2L^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) \leq \frac{1}{4} \Longrightarrow \eta^2 \leq \frac{N}{24\beta^2L^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)}; \tag{B.106}$$

$$\begin{aligned}&\frac{12\beta^2\eta^2L^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)(1-p_{\text{avg}})(2-p_{\min})}{(p_{\min})^2} \leq \frac{1}{4} \iff \\ &\iff \eta^2 \leq \frac{(p_{\min})^2}{48\beta^2L^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)(1-p_{\text{avg}})(2-p_{\min})}.\end{aligned} \tag{B.107}$$

Under these requirements, we are able to pick:

$$\delta = \eta^2 L; \tag{B.108}$$

$$\alpha = 12\eta^2L\beta^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{p_{\min}}\right). \tag{B.109}$$

Replacing the selected values for $\delta$ and $\alpha$ into Eq. (B.98) (as per Eq. (B.108) and (B.109), respectively), yields:

$$\begin{aligned}&\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}[\psi^{(t+1)}] \\ &\leq \psi^{(t)} - \frac{\eta}{2}\left[1 - \frac{12\eta L(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\right]\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \\ &\quad + \frac{\eta}{2N}\sum_{i=1}^{N}\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2}_{\text{uniformly bounded by Lemma A.5}} + \frac{6\eta^2L}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2}_{\text{uniformly bounded by Lemma A.5}} \\ &\quad + 12\beta^2\eta^2L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{p_{\min}}\right)\frac{1}{N}\sum_{i=1}^{N}p_i\underbrace{\mathbb{E}_{\mathcal{B}_i^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{g}}_i^{(t)} - \nabla F_i(\boldsymbol{w}^{(t)})\right\|^2}_{\text{uniformly bounded by Lemma A.5}} \\ &\quad + \left[\frac{\eta^2L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right) + 12\beta^2\eta^2L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\frac{\sigma^2}{K} \\ &\quad + \frac{6\eta^2L(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2.\end{aligned} \tag{B.110}$$

Next, applying Lemma A.5 into Eq. (B.110):

$$\begin{aligned}&\mathbb{E}_{\mathbb{1}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}[\psi^{(t+1)}] \\ &\leq \psi^{(t)} - \frac{\eta}{2}\left[1 - \frac{12\eta L(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\right]\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \\ &\quad + \frac{\eta}{2}\left[1 + \frac{12\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + 24\beta^2\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\times \\ &\quad \times 8\eta_c^2L^2K(K-1)\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2\end{aligned}$$

$$
\begin{aligned}
&+\left[\frac{\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)+12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\frac{\sigma^2}{K}\\
&+\left[\frac{\eta}{2}+\frac{6\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)+12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]2\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K}\\
&+\frac{6\eta^2 L(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2\\
&+\left[\frac{\eta}{2}+\frac{6\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)+12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]8\eta_c^2 L^2 K(K-1)\sigma_g^2,
\end{aligned}
\tag{B.111}
$$

where Eq. (B.111) requires minor rearrangements.

Achieving Eq. (B.91)'s per-round progress requires the gradient squared norm's coefficient should not exceed $-\frac{\eta}{4}$. This leads to the following step-size requirements:

$$
8\eta_c^2 L^2 K(K-1)\le\frac{1}{8}\iff\eta_c^2\le\frac{1}{64L^2K^2};
\tag{B.112}
$$

$$
\frac{12\eta L(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\le\frac{1}{8}\iff\eta\le\frac{N}{96(1-\beta)^2L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)};
\tag{B.113}
$$

$$
\frac{96\eta\eta_c^2 L^3K(K-1)}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\le\frac{1}{8}\iff\eta\le\frac{N}{12L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)};
\tag{B.114}
$$

$$
192\beta^2\eta\eta_c^2 L^3K(K-1)\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{p_{\text{avg}}}{p_{\min}}\right)\le\frac{1}{8}\iff\eta\le\frac{1}{24\beta^2L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{p_{\text{avg}}}{p_{\min}}\right)}.
\tag{B.115}
$$

In summary, combining conditions (B.106), (B.107), and (B.112)–(B.115), the necessary step-size requirements are:

$$
\eta_c\le\frac{1}{8LK};
\tag{B.116}
$$

$$
\eta_s\le\min\left\{\frac{N}{12(1-\beta)^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)},\frac{2N}{3\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)},\frac{1}{3\beta^2\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{p_{\text{avg}}}{p_{\min}}\right)}\right\}.
\tag{B.117}
$$

Under these conditions, we bound the gradient squared norm's coefficient with $-\frac{\eta}{4}$, thus deriving Eq. (B.91).

□

**Lemma A.11** (`FedStale`: Initial progress)**.** *Under Assumptions 2–5 and the specified client-server learning rates (Eq. (B.89)), recall the definition of Lyapunov function $\psi^{(t)}$ in Eq. (B.90). Define the initial error resulting from the memory term's initialization as:*

$$
H^{(1)}:=\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\boldsymbol{h}_i^{(1)}\right\|^2.
\tag{B.118}
$$

*We decompose* `FedStale`*'s initial progress into three main terms: the objective's initial decrease, the initial memory error, and* `FedAvg`*'s error from stochastic gradients and data heterogeneity—which* `FedStale` *also encounters upon memory initialization:*

$$
\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left[\psi^{(2)}\right] \\
&\leq F(\boldsymbol{w}^{(1)}) - \frac{\eta}{4}\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2 \\
&\quad + \left[\frac{\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{min}}\right)\right]\frac{\sigma^2}{K} \\
&\quad + \left[\frac{\eta}{2} + \frac{6\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{min}}\right)\right]2\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \\
&\quad + \frac{3\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2 \\
&\quad + \left[\frac{\eta}{2} + \frac{3\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{min}}\right)\right]8\eta_c^2 L^2 K(K-1)\sigma_g^2 \\
&\quad + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1-p_{min}}{p_{min}}\right)\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)}) - \boldsymbol{h}_i^{(1)}\right\|^2. \qquad \text{(B.119)}
\end{aligned}
$$

*Proof of Lemma A.11.*

We bound $\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}[\psi^{(2)}]$ following Lemma A.10's methodology, starting with $\psi^{(t)}$'s definition from Eq. (B.90):

$$
\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left[\psi^{(2)}\right] \\
&= \mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left[F(\boldsymbol{w}^{(2)})\right] + \left(\delta - \frac{\eta^2 L}{2}\right)\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left\|\Delta^{(1)}\right\|^2 + \alpha\underbrace{\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left[\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)}) - \boldsymbol{h}_i^{(2)}\right\|^2\right]}_{\triangleq H^{(2)}} \qquad \text{(B.120)} \\
&\leq F(\boldsymbol{w}^{(1)}) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2 + \frac{\eta}{2N}\sum_{i=1}^{N}\mathbb{E}_{\mathcal{B}_i^{(1)}}\left\|\nabla F_i(\boldsymbol{w}^{(1)}) - \bar{\boldsymbol{g}}_i^{(1)}\right\|^2 \\
&\quad + \delta\underbrace{\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left\|\Delta^{(1)} - \bar{\boldsymbol{g}}^{(1)}\right\|^2}_{\text{bounded by Lemma A.8}} + \alpha\underbrace{\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left[\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)}) - \boldsymbol{h}_i^{(2)}\right\|^2\right]}_{\text{bounded by Lemma A.9}}, \qquad \text{(B.121)}
\end{aligned}
$$

where Eq.(B.121) leverages LemmaA.1 and Jensen's inequality, with $\Delta^{(t)}$ as an unbiased estimator of $\bar{\boldsymbol{g}}^{(t)}$ and $\delta \leq \frac{\eta}{2}$, following Eqs. (B.94)–(B.97). Next, we apply Lemmas A.8 and A.9 into Eq. (B.121):

$$
\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left[\psi^{(2)}\right] \\
&\leq F(\boldsymbol{w}^{(1)}) - \frac{\eta}{2}\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2 + \frac{\eta}{2N}\sum_{i=1}^{N}\underbrace{\mathbb{E}_{\mathcal{B}_i^{(1)}}\left\|\nabla F_i(\boldsymbol{w}^{(1)}) - \bar{\boldsymbol{g}}_i^{(1)}\right\|^2}_{\text{uniformly bounded by Lemma A.5}}
\end{aligned}
$$

$$+\frac{\delta}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)\frac{\sigma^2}{K}+\frac{3\delta}{N^2}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\underbrace{\mathbb{E}_{\mathcal{B}_i^{(1)}}\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\bar{\boldsymbol{g}}_i^{(1)}\right\|^2}_{\text{uniformly bounded by Lemma A.5}}$$
$$+\frac{3\delta}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2+\frac{3\delta}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2$$
$$+\alpha\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\frac{\sigma^2}{K}+\alpha\frac{1}{N}\sum_{i=1}^{N}p_i\underbrace{\mathbb{E}_{\mathcal{B}_i^{(1)}}\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\bar{\boldsymbol{g}}_i^{(1)}\right\|^2}_{\text{uniformly bounded by Lemma A.5}}+\alpha(1-p_{\min})\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\boldsymbol{h}_i^{(1)}\right\|^2. \tag{B.122}$$

Then, we invoke Lemma A.5:

$$\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(1)},\mathcal{B}^{(1)}}\left[\psi^{(2)}\right]\\
&\leq F(\boldsymbol{w}^{(1)})-\frac{\eta}{2}\left[1-\frac{6\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\right]\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2\\
&\quad+\frac{\eta}{2}\left[1+\frac{6\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)+24\beta^2\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]8\eta_c^2L^2K(K-1)\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2\\
&\quad+\left[\frac{\eta^2L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)+12\beta^2\eta^2L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\frac{\sigma^2}{K}\\
&\quad+\left[\frac{\eta}{2}+\frac{6\eta^2L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)+12\beta^2\eta^2L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]2\eta_c^2L^2K(K-1)\frac{\sigma^2}{K}\\
&\quad+\frac{3\eta^2L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2\\
&\quad+\left[\frac{\eta}{2}+\frac{3\eta^2L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)+12\beta^2\eta^2L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]8\eta_c^2L^2K(K-1)\sigma_g^2\\
&\quad+12\beta^2\eta^2L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1-p_{\min}}{p_{\min}}\right)\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)})-\boldsymbol{h}_i^{(1)}\right\|^2,
\end{aligned} \tag{B.123}$$

where Eq. (B.123) results from rearrangement of terms.

Finally, applying the step-size criteria in Eq. (B.89), Eq. (B.119) achieves `FedStale`'s per-round progress of $-\frac{\eta}{4}\left\|\nabla F(\boldsymbol{w}^{(1)})\right\|^2$.

□

## A.C Proof of Theorem A.12

**Theorem A.12** (`FedStale`'s Convergence)**.** *Within Assumptions 2–5 and specified learning rates (Eq. (B.89)), `FedStale`'s expected squared gradient norm over $T$ rounds is influenced by the initial errors (related to $\boldsymbol{w}^{(1)}$ and $H^{(1)}$), deviations from stochastic gradient variance ($\sigma^2$) and data heterogeneity ($\sigma_g^2$), and the critical hyper-parameter*

*$\beta$ controlling stale updates influence:*

$$
\begin{aligned}
&\min_{t\in[1,T]} \mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \\
&\leq \frac{4\left(F(\boldsymbol{w}^{(1)}) - \mathbb{E}[\psi^{(T)}]\right)}{\eta T} + \beta^2\left[\frac{48\eta L}{T}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1-p_{min}}{p_{min}}\right)\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)}) - \boldsymbol{h}_i^{(1)}\right\|^2\right] \\
&\quad + \frac{4\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)\frac{\sigma^2}{K} + 4\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} + \frac{48\eta\eta_c^2 L^3 K(K-1)}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\frac{\sigma^2}{K} \\
&\quad + \beta^2\left[48\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{min}}\right)\right]\frac{\sigma^2}{K} \\
&\quad + \beta^2\left[96\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{min}}\right)\right]\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \\
&\quad + 16\eta_c^2 L^2 K(K-1)\sigma_g^2 + \frac{192\eta\eta_c^2 L^3 K(K-1)}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2 \\
&\quad + (1-\beta)^2\left[\frac{24\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\right]\sigma_g^2 \\
&\quad + \beta^2\left[384\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{min}}\right)\right]\eta_c^2 L^2 K(K-1)\sigma_g^2 + \frac{12\eta L}{NT}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2.
\end{aligned}
\tag{B.124}
$$

*Proof of Theorem A.12.*

The proof relies on Lemmas A.10 and A.11, under Assumptions 2–5 and the specified learning rates (Eq. (B.89)).

From Lemma A.10, unfolding the recursion for $t = 2, \ldots, T$ through the law of total expectation, it yields:

$$
\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(2:T)},\mathcal{B}^{(2:T)}|\mathcal{H}^{(1)}}\left[\psi^{(T)}\right] \\
&\leq \underbrace{\psi^{(2)}}_{\text{bounded in expectation by Lemma A.11}} + \sum_{t=2}^{T}\Bigg(-\frac{\eta}{4}\mathbb{E}_{\mathbb{1}^{(2:T)},\mathcal{B}^{(2:T)}|\mathcal{H}^{(1)}}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \\
&\quad + \left[\frac{\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\frac{\sigma^2}{K} \\
&\quad + \left[\frac{\eta}{2} + \frac{6\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]2\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \\
&\quad + \frac{6\eta^2 L(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2 \\
&\quad + \left[\frac{\eta}{2} + \frac{6\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]8\eta_c^2 L^2 K(K-1)\sigma_g^2\Bigg).
\end{aligned}
\tag{B.125}
$$

Invoking Lemma A.11 into Eq. (B.125) and taking the total expectation:

$$
\begin{aligned}
&\mathbb{E}_{\mathbb{1}^{(1:T)},\mathcal{B}^{(1:T)}}\left[\psi^{(T)}\right] \\
&\leq F(\boldsymbol{w}^{(1)}) + \sum_{t=1}^{T}\Bigg(-\frac{\eta}{4}\mathbb{E}_{\mathbb{1}^{(1:T)},\mathcal{B}^{(1:T)}}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \\
&\quad + \left[\frac{\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\frac{\sigma^2}{K} \\
&\quad + \left[\frac{\eta}{2} + \frac{6\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]2\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \\
&\quad + \left[\frac{\eta}{2} + \frac{6\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]8\eta_c^2 L^2 K(K-1)\sigma_g^2\Bigg) \\
&\quad + \sum_{t=2}^{T}\left[\frac{6\eta^2 L(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2\right] + \frac{3\eta^2 L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2 \\
&\quad + 12\beta^2\eta^2 L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1-p_{\min}}{p_{\min}}\right)\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)}) - \boldsymbol{h}_i^{(1)}\right\|^2. \qquad \text{(B.126)}
\end{aligned}
$$

Dividing both sides of Eq. (B.126) by $T$ and rearranging the terms:

$$
\begin{aligned}
&\min_{t\in[1,T]}\mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \\
&\leq \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 \qquad \text{(B.127)} \\
&\leq \frac{4\left(F(\boldsymbol{w}^{(1)}) - \mathbb{E}[\psi^{(T)}]\right)}{\eta T} + \left[\frac{4\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right) + 48\beta^2\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\frac{\sigma^2}{K} \\
&\quad + \left[4 + \frac{48\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + 96\beta^2\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \\
&\quad + \left[16 + \frac{192\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right) + 384\beta^2\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\eta_c^2 L^2 K(K-1)\sigma_g^2 \\
&\quad + \frac{T-1}{T}\left[\frac{24\eta L(1-\beta)^2}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2\right] + \frac{1}{T}\left[\frac{12\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2\right] \\
&\quad + \frac{1}{T}\left[48\beta^2\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1-p_{\min}}{p_{\min}}\right)\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)}) - \boldsymbol{h}_i^{(1)}\right\|^2\right], \qquad \text{(B.128)}
\end{aligned}
$$

where Eq. (B.127) follows comparing the minimum expected squared gradient norm across iterations to the average. Observing that $\psi^{(T)} \geq F(\boldsymbol{w}^{(T)})$, we group errors into common, $(1-\beta^2)$-specific, and $\beta^2$-specific terms:

$$
\min_{t\in[1,T]}\mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2
$$

$$
\begin{aligned}
&\leq \frac{4\left(F(\boldsymbol{w}^{(1)}) - \mathbb{E}[F(\boldsymbol{w}^{(T)})]\right)}{\eta T} \\
&+ \frac{4\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i}\right)\frac{\sigma^2}{K} + 4\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} + \frac{48\eta\eta_c^2 L^3 K(K-1)}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\frac{\sigma^2}{K} \\
&+ 16\eta_c^2 L^2 K(K-1)\sigma_g^2 + \frac{192\eta\eta_c^2 L^3 K(K-1)}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2 + \frac{12\eta L}{NT}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\sigma_g^2 \\
&+ (1-\beta)^2\left[\frac{24\eta L}{N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\right]\sigma_g^2 \\
&+ \beta^2\left[48\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\frac{\sigma^2}{K} \\
&+ \beta^2\left[96\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \\
&+ \beta^2\left[384\eta L\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1}{N}\sum_{i=1}^{N}p_i\right)\left(\frac{1}{p_{\min}}\right)\right]\eta_c^2 L^2 K(K-1)\sigma_g^2 \\
&+ \beta^2\left[\frac{48\eta L}{T}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1-p_i}{p_i}\right)\left(\frac{1-p_{\min}}{p_{\min}}\right)\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla F_i(\boldsymbol{w}^{(1)}) - \boldsymbol{h}_i^{(1)}\right\|^2\right]. \qquad \text{(B.129)}
\end{aligned}
$$

Finally, organizing errors into initial, stochastic gradient-specific, and data heterogeneity-specific terms yields Eq. (B.124).

□

# B FedStale, Lower bound

To prove oracle complexities lower bounds for smooth objectives, we consider the function used by Nesterov (Nesterov, 2004; Bubeck, 2015)

$$
\begin{aligned}
F(\boldsymbol{w}) &= \frac{L}{8}\left(\boldsymbol{w}^\top A_{2t+1}\boldsymbol{w} - 2\boldsymbol{w}^\top e_1\right) && \text{(B.130)} \\
&= \frac{L}{8}\left[(\boldsymbol{w})_1^2 + \sum_{j=1}^{2t}\left((\boldsymbol{w})_j - (\boldsymbol{w})_{j+1}\right)^2 + (\boldsymbol{w})_{2t+1}^2 - 2(\boldsymbol{w})_1\right] && \text{(B.131)}
\end{aligned}
$$

where, for $t \leq \frac{d-1}{2}$, $A_t \in \mathbb{R}^{d\times d}$ is a symmetric and tridiagonal matrix defined as

$$
(A_t)_{ij} = \begin{cases} 2, & i = j, i \leq t \\ -1, & |i-j| = 1, i \leq t, j \neq t+1 \\ 0, & \text{otherwise.} \end{cases} \qquad \text{(B.132)}
$$

Following the methodology introduced in (Scaman, Bach, Bubeck, Lee, & Massoulié, 2019), our approach relies on distributing the objective function $F(\boldsymbol{w})$ across two clients. This split is designed such that most components of

the parameter vector $\boldsymbol{w}^{(t)}$ remain zero. Local gradient computations increase the number of non-zero components by at most one whenever a client becomes active, without any additional component revealed until the other client participates. More rigorously, let $i_0, i_1 \in \mathcal{N}$ denote two clients. For every client $i \in \mathcal{N}$, we define the objective functions $F_i(\boldsymbol{w}) : \mathbb{R}^d \to \mathbb{R}$ as follows:

$$F_i(\boldsymbol{w}) = \begin{cases} \frac{NL}{8}\left[(\boldsymbol{w})_1^2 + \sum_{j=1}^{t}\left((\boldsymbol{w})_{2j} - (\boldsymbol{w})_{2j+1}\right)^2 - 2(\boldsymbol{w})_1\right] & \text{if } i = i_0 \\ \frac{NL}{8}\left[\sum_{j=1}^{t}\left((\boldsymbol{w})_{2j-1} - (\boldsymbol{w})_{2j}\right)^2 + (\boldsymbol{w})_{2t+1}^2\right] & \text{if } i = i_1 \\ 0 & \text{otherwise.} \end{cases} \tag{B.133}$$

It is easy to verify that $F(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N} F_i(\boldsymbol{w})$.

## B.A Upper bound on $k^{(t)}$

Our objective is to establish an upper bound for the maximum non-zero index of the parameter vector $\boldsymbol{w}^{(t)}$, which minimizes $F(\boldsymbol{w})$ at any given time $t$. We introduce the notations

$$k_i^{(t)} = \max\{k \in \mathbb{N}, k \leq d \mid \exists \boldsymbol{w}_i^{(t)} \in \mathbb{R}^d \text{ such that } (\boldsymbol{w}_i^{(t)})_k \neq 0\}$$

and

$$k^{(t)} = \max\{k \in \mathbb{N}, k \leq d \mid \exists \boldsymbol{w}^{(t)} \in \mathbb{R}^d \text{ such that } (\boldsymbol{w}^{(t)})_k \neq 0\}$$

to respectively represent the largest index of non-zero components in $\boldsymbol{w}_i^{(t)}$ and $\boldsymbol{w}^{(t)}$.

At the beginning of each round $t$, the server initializes $k_i^{(t)}$ to $k^{(t-1)}$ for any participating client $i \in \mathcal{S}^{(t)}$. Subsequently, after the local computations conclude, the server updates $k^{(t)}$ to the maximum of all $k_i^{(t)}$ values among participating clients, which corresponds to the largest index of non-zero components discovered up to time $t$.

Extending the results of (Scaman et al., 2019, Lemma 23) to scenarios with partial client participation, it can be easily shown that:

$$k_i^{(t+1)} \leq \begin{cases} k_i^{(t)} + \mathbb{1}\{k_i^{(t)} \equiv 0 \bmod 2\} & \text{if } i = i_0 \wedge i_0 \text{ is active,} \\ k_i^{(t)} + \mathbb{1}\{k_i^{(t)} \equiv 1 \bmod 2\} & \text{if } i = i_1 \wedge i_1 \text{ is active,} \\ k_i^{(t)} & \text{otherwise.} \end{cases} \tag{B.134}$$

Due to the stochastic nature of client participation, the sequences $k_i^{(t)}$ and $k^{(t)}$, for $i \in \mathcal{N}$ and $t \in \mathcal{T}$, must be treated as stochastic processes. Initially, to facilitate the analysis, we assume a deterministic model for client participation. Subsequently, by leveraging Jensen's inequality, we generalize our results to account for the expected stochastic dynamics of client participation.

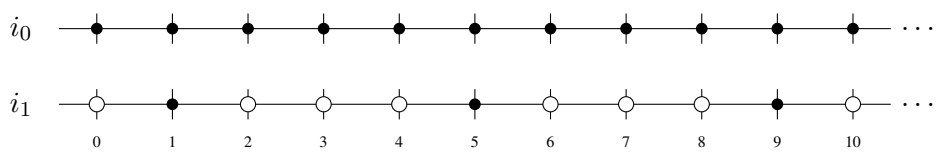
In what follows, we consider, without loss of generality, that client $i_0$ is always active ($p_0 = 1$), and client $i_1$ exhibits the minimum availability ($p_1 = \min_i p_i = p$). In terms of upper bound on $k^{(t)}$, this assumption represents a boundary condition on how fast $k^{(t)}$ can grow in a framework characterized by minimum client participation $p$.

From the two previous assumptions, we observe that client $i_1$ exhibits a deterministic activity pattern, participating in every $\tau = 1/p > 1$ communication cycles. We can straightforwardly derive from (B.134) the subsequent update rule for $k^{(t)}$:

$$k^{(t+1)} = k^{(t)} + \mathbb{1}\{t \bmod \tau \in \{0, 1\}\}, \quad \forall t \in \mathcal{T} \tag{B.135}$$

In the example below, we clarify the notation and illustrate the increment process of $k^{(t)}$.

**Example B.1.** We set $p = 1/4$, implying $\tau = 4$. Moreover, we assume $i_1$ is active at rounds $t = \{1, 5, 9, \dots\}$.




Given the initialization parameters $\boldsymbol{w}^{(0)}$ with $k^{(0)} = 0$, we observe:

$t = 0$ : Client $i_0$ is active and increases $k_0^{(0)} = 1$ ($k^{(0)}$ is even). $\underline{k^{(0)} = 1}$.

$t = 1$ : Client $i_0$ is active but does not increase $k_0^{(1)}$ ($k^{(0)}$ is odd). Client $i_1$ is active, $k_1^{(1)} = 2$ (since $k^{(0)}$ is odd). $\underline{k^{(1)} = 2}$.

$t = 2$ : Client $i_0$ increases $k_0^{(2)}$ to 3 (since $k^{(1)}$ is even). Client $i_1$ is inactive. $\underline{k^{(2)} = 3}$.

$t = 3$ : Client $i_0$ does not increase $k_0^{(3)}$ (since $k^{(2)}$ is odd). Client $i_1$ is inactive. $\underline{k^{(3)} = 3}$.

$t = 4$ : Client $i_1$ is inactive. $\underline{k^{(4)} = 3}$.

$t = 5$ : Client $i_1$ is active and increases $k_1^{(5)}$ to 4. $\underline{k^{(5)} = 4}$.

$t = 6$ : Client $i_0$ increases $k_0^{(6)}$ to 5 (since $k^{(5)}$ is even). Client $i_1$ is inactive. $\underline{k^{(6)} = 5}$.

$t = 7$ : Client $i_0$ is active but does not increase $k_0^{(7)}$ ($k^{(6)}$ is odd). Client $i_1$ is inactive. $\underline{k^{(7)} = 5}$.

$t = 8$ : Client $i_1$ is inactive. $\underline{k^{(8)} = 5}$.

$t = 9$ : Client $i_1$ is active and increases $k_1^{(9)}$ to 6. $\underline{k^{(9)} = 6}$.

$t = 10$ : Client $i_0$ increases $k_0^{(10)}$ to 7 (since $k^{(9)}$ is even). Client $i_1$ is inactive. $\underline{k^{(10)} = 7}$.

These observations verify $k^{(t+1)} = k^{(t)} + \mathbb{1}\{t \bmod 4 \in \{0, 1\}\}$, for all $t \in \mathcal{T}$.

Moreover, Example B.1 motivates two further observations:

1. Client $i_1$'s initial activation (say $j$) produces $\tau$ distinct temporal sequences (say $k^{(t,j)}$), each delayed from the reference sequence $k^{(t,1)}$ up to $\tau - 1$ time steps. Specifically, the following relationship holds:

$$k^{(t,j)} = k^{(t-((j+\tau-1) \bmod \tau),1)}, \quad \text{for } j = 0, \dots, \tau - 1.$$

   In other words, all sequences $k^{(t,j)}$, for $j \neq 1$, progress at a slower pace than $k^{(t,1)}$; thus, the sequence $k^{(t,1)}$ represents the fastest evolution among them.

2. Given the objectives partition among clients as detailed in Eq. (B.133)—with clients $i_0$ and $i_1$ optimizing even and odd components, respectively—the optimization process in Example B.1 initiates with client $i_0$ targeting the first (even) component at $t = 0$. Should we invert their objectives in Eq. (B.133), then client $i_1$ would commence at $t = 0$. This switch initiates $\tau$ new sequences, $\hat{k}^{(t,j)}$, distinct from the original $k^{(t,j)}$ sequences. For $j = 1, \dots, \tau - 1$, each $\hat{k}^{(t,j)}$ is one value below its $k^{(t,j)}$ counterpart except for $j = 0$, where:

$$(k^{(t,1)})_i = \begin{cases} (\hat{k}^{(t,0)})_i, & \text{if } i \bmod \tau \in \{0, 1\}, \\ (\hat{k}^{(t,0)})_i + 1, & \text{otherwise.} \end{cases}$$

In any case, $k^{(t,1)}$ remains consistently the fastest sequence among $\hat{k}^{(t,j)}$ for $j = 0, \ldots, \tau - 1$.

For notational brevity, we denote $k^{(t,1)}$ simply as $k^{(t)}$ in subsequent discussions. The following lemma proves that at least $\mathcal{O}(\tau)$ communication rounds are necessary in-between every gradient computation, in order to optimize the global objective.

**Lemma B.1.** *Given a set $\mathcal{N}$ of $N$ clients, with $i_0$ (having $p_0 = 1$) always participating, and $i_1$ (with $p_1 = \min_{i \in \mathcal{N}} p_i = p$) being the least participating client at a period $\tau = 1/p > 1$, let $k^{(t)} = \max\{k \in \mathbb{N}, k \leq d \mid \exists \boldsymbol{w}^{(t)} \in \mathbb{R}^d$ such that $(\boldsymbol{w}^{(t)})_k \neq 0\}$ represent the largest index of non-zero components in any parameter vector $\boldsymbol{w}^{(t)} \in \mathbb{R}^d$. Assuming the global objective $F(\boldsymbol{w})$ is partitioned among clients as specified in Eq. (B.133), the upper bound for the index $k^{(t)}$ at any time $t \geq 0$ is given by:*

$$k^{(t)} \leq 1 + \left\lfloor \frac{t + \tau - 2}{\tau} \right\rfloor + \left\lfloor \frac{t + \tau - 1}{\tau} \right\rfloor. \tag{B.136}$$

*Proof of Lemma B.1.*

The proof proceeds by induction on the time step $t$.

*Base case.* At $t = 0$, the initial condition yields:

$$k^{(0)} \leq 1 + \left\lfloor \frac{\tau - 2}{\tau} \right\rfloor + \left\lfloor \frac{\tau - 1}{\tau} \right\rfloor = 1, \tag{B.137}$$

since, for $\tau \geq 1$, both the floor terms $\left\lfloor \frac{\tau-2}{\tau} \right\rfloor$ and $\left\lfloor \frac{\tau-1}{\tau} \right\rfloor$ evaluate to zero.

*Inductive step.* Assume Eq. (B.136) is valid for an arbitrary $t \geq 0$. Our goal is to show that the relationship holds for $t + 1$ as well. From the induction hypothesis for $k^{(t)}$, we have:

$$k^{(t+1)} \leq 1 + \left\lfloor \frac{t + \tau - 1}{\tau} \right\rfloor + \left\lfloor \frac{t + \tau}{\tau} \right\rfloor \tag{B.138}$$

$$= k^{(t)} + \left\lfloor \frac{t}{\tau} + 1 \right\rfloor - \left\lfloor \frac{t - 2}{\tau} + 1 \right\rfloor, \tag{B.139}$$

where, in (B.139), we applied the definition of $k^{(t)}$ from Eq. (B.136).

Next, we observe that the difference $\left\lfloor \frac{t+\tau}{\tau} \right\rfloor - \left\lfloor \frac{t+\tau-2}{\tau} \right\rfloor$ only depends on the congruence class of $t \bmod \tau$, as the $\tau$ term simplifies in the subtraction. Specifically,

The expression $\left\lfloor \frac{t \bmod \tau}{\tau} + 1 \right\rfloor$ consistently equals one, since $0 \leq \frac{t \bmod \tau}{\tau} < 1$.

Conversely, $\left\lfloor \frac{t \bmod \tau - 2}{\tau} + 1 \right\rfloor$ is one for $t \bmod \tau \geq 2$, and zero for $t \bmod \tau \in \{0, 1\}$.

Thus, the difference $\left\lfloor \frac{t}{\tau} + 1 \right\rfloor - \left\lfloor \frac{t-2}{\tau} + 1 \right\rfloor$ equals one for $t \equiv 0, 1 \pmod{\tau}$, and zero otherwise. This observation aligns with the incremental rule of Eq. (B.135), thus concluding the proof.

□

## B.B Lower bound on $||\nabla F(\boldsymbol{w}^{(t)})||^2$

**Lemma B.2.** *For any time step $t \leq \frac{d-1}{2}$ in a $d$-dimensional space, and Lipschitz constant $L > 0$, there exists an $L$-smooth convex function $F : \mathbb{R}^d \to \mathbb{R}$, for which the minimum squared norm of the gradient, evaluated at any point within the first $t$ steps of any first-order black-box optimization procedure, satisfies:*

$$\min_{1\leq s\leq t} \left\|\nabla F(\boldsymbol{w}^{(s)})\right\|^2 \geq \frac{6L\left(F(\boldsymbol{w}^{(1)}) - F^*\right)}{t(2t+1)^2}, \tag{B.140}$$

*where $\boldsymbol{w}^{(s)}$ represents the parameter vector at step $s$, and $F^*$ denotes the minimum value of $F$.*

*Proof of Lemma B.2.*

We begin by recalling the global objective function (Nesterov, 2004; Bubeck, 2015)

$$F(\boldsymbol{w}) = \frac{L}{8}\boldsymbol{w}^\top A_{2t+1}\boldsymbol{w} - \frac{L}{4}\boldsymbol{w}^\top e_1,$$

where $A_t \in \mathbb{R}^{d\times d}$ is the symmetric, tridiagonal matrix, defined for $t \leq \frac{d-1}{2}$ as:

$$(A_t)_{ij} = \begin{cases} 2, & \text{if } i = j \text{ and } i \leq t, \\ -1, & \text{if } |i-j| = 1 \text{ and } i \leq t, j \neq t+1, \\ 0, & \text{otherwise.} \end{cases}$$

**Proposition B.3.** *The function $F(\boldsymbol{w})$ satisfies Assumption 6.*

*Proof.*

The proof follows directly from (Bubeck, 2015, Theorem 3.14).

□

Our objective is to derive a lower bound on the squared norm of the gradient $\nabla F(\boldsymbol{w}^{(t)})$, specifically, the minimum gradient norm observed up to and including time step $t$.

Given the black-box procedure's assumption, we note that $\boldsymbol{w}^{(t)}$ is restricted to the linear span of $\mathbf{e}_1, \ldots, \mathbf{e}_{t-1}$, implying:

$$\boldsymbol{w}^{(t)} = \left((\boldsymbol{w}^{(t)})_1, \ldots, (\boldsymbol{w}^{(t)})_{t-1}, 0, \ldots, 0\right).$$

We define $\boldsymbol{w}^{(t,*)} = \underset{\boldsymbol{w}\in \text{Span}(\mathbf{e}_1,\ldots,\mathbf{e}_{t-1})}{\arg\min} \|\nabla F(\boldsymbol{w})\|^2$, and $\boldsymbol{w}^{(*)} = \underset{\boldsymbol{w}\in\mathbb{R}^d}{\arg\min} \|\nabla F(\boldsymbol{w})\|^2$. The following inequality holds:

$$\min_{1\leq s\leq t} \left\|\nabla F(\boldsymbol{w}^{(s)})\right\|^2 \geq \left\|\nabla F(\boldsymbol{w}^{(t,*)})\right\|^2 \geq \left\|\nabla F(\boldsymbol{w}^{(*)})\right\|^2$$

Moving forward in the derivation of the lower bound, our focus shifts towards evaluating $\left\|\nabla F(\boldsymbol{w}^{(t,*)})\right\|^2$. The first step involves identifying $\boldsymbol{w}^{(t,*)}$, the parameter vector that minimizes the squared norm of the gradient within the span of $\{\mathbf{e}_1, \ldots, \mathbf{e}_{t-1}\}$.

### B.B.1 Finding $\boldsymbol{w}^{(t,*)}$

Considering $\boldsymbol{w}^{(t)} \in \mathrm{Span}(\mathbf{e}_1, \ldots, \mathbf{e}_{t-1})$, we calculate the gradient of $F$ at $\boldsymbol{w}^{(t)}$ as follows:

$$\frac{\partial F(\boldsymbol{w}^{(t)})}{\partial (\boldsymbol{w}^{(t)})_i} = \begin{cases} \frac{L}{4}\left[-1 + 2(\boldsymbol{w}^{(t)})_i - (\boldsymbol{w}^{(t)})_{i+1}\right] & \text{for } i = 1, \\ \frac{L}{4}\left[-(\boldsymbol{w}^{(t)})_{i-1} + 2(\boldsymbol{w}^{(t)})_i - (\boldsymbol{w}^{(t)})_{i+1}\right] & \text{for } i = 2, \ldots, t-2, \\ \frac{L}{4}\left[-(\boldsymbol{w}^{(t)})_{i-1} + 2(\boldsymbol{w}^{(t)})_i\right] & \text{for } i = t-1, \\ \frac{L}{4}\left[-(\boldsymbol{w}^{(t)})_i\right] & \text{for } i = t, \\ 0 & \text{for } i = t+2, \ldots, 2t+1, \end{cases}$$

where the gradient evaluations explicitly reflect their dependence on adjacent components $i-1$, $i$, and $i+1$, as a consequence of the structural properties of the symmetric, tridiagonal matrix $A$, and the boundary conditions imposed on the vector $\boldsymbol{w}^{(t)}$.
The squared gradient norm, $\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2$, is then given by:

$$\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2 = \frac{L^2}{16}\Bigg[\left(2(\boldsymbol{w}^{(t)})_1 - (\boldsymbol{w}^{(t)})_2 - 1\right)^2 + \sum_{i=2}^{t-2}\left((\boldsymbol{w}^{(t)})_{i-1} + 2(\boldsymbol{w}^{(t)})_i - (\boldsymbol{w}^{(t)})_{i+1}\right)^2 + \left((\boldsymbol{w}^{(t)})_{t-2} + 2(\boldsymbol{w}^{(t)})_{t-1}\right)^2 + (\boldsymbol{w}^{(t)})_{t-1}^2\Bigg].$$

Minimizing this expression with respect to $\boldsymbol{w}^{(t)}$ involves setting its partial derivatives to zero, leading to the system of equations:

$$\frac{\partial \left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2}{\partial (\boldsymbol{w}^{(t)})_i} = \begin{cases} \frac{L^2}{16}\left[-4 + 10(\boldsymbol{w}^{(t)})_i - 8(\boldsymbol{w}^{(t)})_{i+1} + 2(\boldsymbol{w}^{(t)})_{i+2}\right] & \text{for } i = 1, \\ \frac{L^2}{16}\left[2 - 8(\boldsymbol{w}^{(t)})_{i-1} + 12(\boldsymbol{w}^{(t)})_i - 8(\boldsymbol{w}^{(t)})_{i+1} + 2(\boldsymbol{w}^{(t)})_{i+2}\right] & \text{for } i = 2, \\ \frac{L^2}{16}\left[2(\boldsymbol{w}^{(t)})_{i-2} - 8(\boldsymbol{w}^{(t)})_{i-1} + 12(\boldsymbol{w}^{(t)})_i - 8(\boldsymbol{w}^{(t)})_{i+1} + 2(\boldsymbol{w}^{(t)})_{i+2}\right] & \text{for } i = 3, \ldots, t-3, \\ \frac{L^2}{16}\left[2(\boldsymbol{w}^{(t)})_{i-2} - 8(\boldsymbol{w}^{(t)})_{i-1} + 12(\boldsymbol{w}^{(t)})_i - 8(\boldsymbol{w}^{(t)})_{i+1}\right] & \text{for } i = t-2, \\ \frac{L^2}{16}\left[2(\boldsymbol{w}^{(t)})_{i-2} - 8(\boldsymbol{w}^{(t)})_{i-1} + 12(\boldsymbol{w}^{(t)})_i\right] & \text{for } i = t-1. \end{cases}$$

The minimizer of $\left\|\nabla F(\boldsymbol{w}^{(t)})\right\|^2$ now depends on adjacent indices $i-2$, $i-1$, $i$, $i+1$, and $i+2$:

$$(\boldsymbol{w}^{(t,*)})_i = \begin{cases} \frac{2}{5} + \frac{4}{5}(\boldsymbol{w}^{(t,*)})_{i+1} - \frac{1}{5}(\boldsymbol{w}^{(t,*)})_{i+2} & \text{for } i = 1, \\ -\frac{1}{6} + \frac{2}{3}(\boldsymbol{w}^{(t,*)})_{i-1} + \frac{2}{3}(\boldsymbol{w}^{(t,*)})_{i+1} - \frac{1}{6}(\boldsymbol{w}^{(t,*)})_{i+2} & \text{for } i = 2, \\ -\frac{1}{6}(\boldsymbol{w}^{(t,*)})_{i-2} + \frac{2}{3}(\boldsymbol{w}^{(t,*)})_{i-1} + \frac{2}{3}(\boldsymbol{w}^{(t,*)})_{i+1} - \frac{1}{6}(\boldsymbol{w}^{(t,*)})_{i+2} & \text{for } i = 3, \ldots, t-3, \\ -\frac{1}{6}(\boldsymbol{w}^{(t,*)})_{i-2} + \frac{2}{3}(\boldsymbol{w}^{(t,*)})_{i-1} + \frac{2}{3}(\boldsymbol{w}^{(t,*)})_{i+1} & \text{for } i = t-2, \\ -\frac{1}{6}(\boldsymbol{w}^{(t,*)})_{i-2} + \frac{2}{3}(\boldsymbol{w}^{(t,*)})_{i-1} & \text{for } i = t-1. \end{cases}$$

The optimal vector $\boldsymbol{w}^{(t,*)}$ emerges as solution of this system, relating the $i$-th component directly to $(\boldsymbol{w}^{(t,*)})_1$ and $(\boldsymbol{w}^{(t,*)})_2$. This yields the following recursive formula for $i = 3, \ldots, t-1$:

$$(\boldsymbol{w}^{(t,*)})_i = \frac{1}{6}(i^3 - i)(\boldsymbol{w}^{(t,*)})_2 - \frac{1}{3}(i^2 - 4)(\boldsymbol{w}^{(t,*)})_1 + \frac{1}{6}(i^3 - 7i + 6).$$

Finally, leveraging the boundary conditions on $(\boldsymbol{w}^{(t,*)})_{t-2}$ and $(\boldsymbol{w}^{(t,*)})_{t-1}$, we solve for the initial components $(\boldsymbol{w}^{(t,*)})_1$ and $(\boldsymbol{w}^{(t,*)})_2$. The generalized expression for $\boldsymbol{w}^{(t,*)}$ is:

$$(\boldsymbol{w}^{(t,*)})_i = \begin{cases} \frac{2t^3-3(i-1)t^2-(3i-1)t+i^3-i}{t(t+1)(2t+1)} & \text{for } i = 1, \dots, t-1, \\ 0 & \text{otherwise,} \end{cases}$$

within the linear span of $\mathbf{e}_1, \dots, \mathbf{e}_{t-1}$.

### B.B.2 Evaluating $\left\|\nabla F(\boldsymbol{w}^{(t,*)})\right\|^2$

To complete the lower bound, we first derive the explicit form of the gradient of $F$ at $\boldsymbol{w}^{(t,*)}$:

$$\frac{\partial F(\boldsymbol{w}^{(t,*)})}{\partial (\boldsymbol{w}^{(t,*)})_i} = \begin{cases} \frac{L}{4}\left[2(\boldsymbol{w}^{(t,*)})_i - (\boldsymbol{w}^{(t,*)})_{i+1} - 1\right] & \text{for } i = 1, \\ \frac{L}{4}\left[-(\boldsymbol{w}^{(t,*)})_{i-1} + 2(\boldsymbol{w}^{(t,*)})_i - (\boldsymbol{w}^{(t,*)})_{i+1}\right] & \text{for } i = 2, \dots, t-2, \\ \frac{L}{4}\left[-(\boldsymbol{w}^{(t,*)})_{i-1} + 2(\boldsymbol{w}^{(t,*)})_i\right] & \text{for } i = t-1, \\ \frac{L}{4}\left[-(\boldsymbol{w}^{(t,*)})_{i-1}\right] & \text{for } i = t, \\ 0 & \text{for } i = t+1, \dots, 2t+1. \end{cases}$$

This yields the gradient's $i$-th component as:

$$\frac{\partial F(\boldsymbol{w}^{(t,*)})}{\partial (\boldsymbol{w}^{(t,*)})_i} = \begin{cases} -\frac{3Li}{2t(t+1)(2t+1)} & \text{for } i = 1, \dots, t \\ 0 & \text{for } i > t. \end{cases}$$

Subsequently, the squared norm of the gradient, $\left\|\nabla F(\boldsymbol{w}^{(t,*)})\right\|^2$, is calculated as follows:

$$\begin{aligned} \left\|\nabla F(\boldsymbol{w}^{(t,*)})\right\|^2 &= \sum_{i=1}^{t} \left(-\frac{3Li}{2t(t+1)(2t+1)}\right)^2 \\ &= \frac{9L^2}{4t^2(t+1)^2(2t+1)^2} \sum_{i=1}^{t} i^2 \\ &= \frac{3L^2}{8t(t+1)(2t+1)}, \end{aligned}$$

by summing the squares of the gradient components and observing that $\sum_{i=1}^{t} i^2 = \frac{1}{6}t(t+1)(2t+1)$.

Additionally, considering the initial error:

$$F(\boldsymbol{w}^{(1)}) - F^* = 0 - \frac{L}{8}\left(1 - \frac{1}{2t+2}\right) = \frac{L(2t+1)}{16(t+1)},$$

we derive the final expression:

$$\min_{1 \le s \le t} \left\|\nabla F(\boldsymbol{w}^{(s)})\right\|^2 \ge \left\|\nabla F(\boldsymbol{w}^{(t,*)})\right\|^2 \ge \frac{6L\left(F(\boldsymbol{w}^{(1)}) - F^*\right)}{t(2t+1)^2},$$

thus concluding the proof.

□

## B.C Proof of Theorem B.4

**Theorem B.4.** *In a federated learning setting involving a set of $N$ clients ($\mathcal{N} = \{1, \ldots, N\}$), where each client $i$ is associated with a participation probability $p_i$, let $p_{min}$ be the minimum participation probability among these clients, i.e., $p_{min} \coloneqq \min_{i \in \mathcal{N}} p_i$. There exists $N$ local functions $F_i : \mathbb{R}^d \to \mathbb{R}$, contributing to the global objective function $F$, such that:*

1. *The function $F : \mathbb{R}^d \to \mathbb{R}$ is $L$-smooth;*
2. *Under any first-order black-box optimization procedure up to time-step $t$, $t \leq \frac{d-1}{2}$, the minimum squared norm of the gradient of $F$ evaluated at any parameter vector $\boldsymbol{w}^{(s)}$ within this time interval satisfies:*

$$\min_{1 \leq s \leq t} \mathbb{E} \left\| \nabla F(\boldsymbol{w}^{(s)}) \right\|^2 \geq \frac{3L \left( F(\boldsymbol{w}^{(1)}) - F^* \right)}{(p_{min} t + 2)(4 p_{min} t + 9)^2}, \tag{B.141}$$

*where $\boldsymbol{w}^{(s)}$ denotes the parameter vector at step $s$, and $F^*$ represents the minimum value of $F$.*

*Proof of Theorem B.4.*

Given the linear span restriction of $\boldsymbol{w}^{(t)}$ to the first $k^{(t)}$ basis vectors, we have

$$\boldsymbol{w}^{(t)} = \left( (\boldsymbol{w}^{(t)})_1, \ldots, (\boldsymbol{w}^{(t)})_{k^{(t)}}, 0, \ldots, 0 \right).$$

From our previous result (Lemma B.2), the minimum squared gradient norm is bounded below by:

$$\min_{1 \leq s \leq t} \left\| \nabla F(\boldsymbol{w}^{(s)}) \right\|^2 \geq \frac{6L \left( F(\boldsymbol{w}^{(1)}) - F^* \right)}{(k^{(t)} + 1)(2k^{(t)} + 3)^2}. \tag{B.142}$$

Considering $k^{(t)}$ as a random variable depending on the geometric distribution of success time with expected value $\tau = 1/p_{\min}$, and leveraging the upper bound established in Lemma B.1, we have:

$$\mathbb{E}[k^{(t)}] \leq 3(1 - p_{\min}) + 2p_{\min} t \leq 3 + 2p_{\min} t. \tag{B.143}$$

Application of Jensen's inequality to the function $f(x) = \frac{1}{(x+1)(2x+3)^2}$, which is convex over $x \geq 0$, completes the proof:

$$\min_{1 \leq s \leq t} \mathbb{E} \left\| \nabla F(\boldsymbol{w}^{(s)}) \right\|^2 \geq \frac{6L \left( F(\boldsymbol{w}^{(1)}) - F^* \right)}{(\mathbb{E}[k^{(t)}] + 1)(2\mathbb{E}[k^{(t)}] + 3)^2} \geq \frac{3L \left( F(\boldsymbol{w}^{(1)}) - F^* \right)}{(p_{\min} t + 2)(4p_{\min} t + 9)^2}. \tag{B.144}$$

□

# APPENDIX C

# Application to Wireless Networks with Lossy Communication Channels

## A Proof of Theorem 4.3.1

For the proof, we define the sequence $\bar{\boldsymbol{w}}^{(t,k)}=\sum_{i\in\mathcal{N}}\alpha_i\boldsymbol{w}_i^{(t,k)}$.

We denote:

$$\mathcal{B}^{(t)}=\{\mathcal{B}^{(t,0)},\mathcal{B}^{(t,1)},\dots,\mathcal{B}^{(t,K-1)}\}; \tag{C.1}$$

$$\mathcal{H}^{(t)}=\{\mathcal{B}^{(1)},\mathcal{S}^{(1)},\mathcal{B}^{(2)},\mathcal{S}^{(2)},\dots,\mathcal{B}^{(t-1)},\mathcal{S}^{(t-1)}\}, \tag{C.2}$$

where $\mathcal{B}^{(t,k)}=\{\mathcal{B}_i^{(t,k)}\}_{i\in\mathcal{N}}$ is the set of random batches sampled at time $(t,k)$ and $\mathcal{H}^{(t)}$ includes all history up to the $t$-th round.

**Lemma A.1.** *Let Assumptions 13–14 hold, and $\boldsymbol{w}^{(t)}=\boldsymbol{w}^{(t)}_{\text{UPGA-PL}}$. Then:*

$$\mathbb{E}_{\mathcal{S}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{w}^{(t+1)}-\bar{\boldsymbol{w}}^{(t,K)}\right\|^2\leq(\eta^{(t)})^2K^2G^2\sum_{i\in\mathcal{N}}\alpha_i^2\frac{q_i}{1-q_i}. \tag{C.3}$$

*Conversely, for $\boldsymbol{w}^{(t)}=\boldsymbol{w}^{(t)}_{\text{UDMA-PL}}$, we have:*

$$\mathbb{E}_{\mathcal{S}^{(t)}|\mathcal{H}^{(t)},\mathcal{B}^{(t)}}\left\|\boldsymbol{w}^{(t+1)}-\bar{\boldsymbol{w}}^{(t,K)}\right\|^2=\sum_{i\in\mathcal{N}}\alpha_i^2\frac{q_i}{1-q_i}\left\|\boldsymbol{w}_i^{(t,K)}\right\|^2. \tag{C.4}$$

*Proof of Lemma A.1.*

$$\mathbb{E}_{\mathcal{S}^{(t)}|\mathcal{H}^{(t)},\mathcal{B}^{(t)}}\left\|\boldsymbol{w}^{(t)}+\sum_{i\in\mathcal{N}}\frac{\alpha_i}{1-q_i}\left(\boldsymbol{w}_i^{(t,K)}-\boldsymbol{w}^{(t)}\right)\mathbb{1}_i^{(t)}-\bar{\boldsymbol{w}}^{(t,K)}\right\|^2$$

$$= \mathbb{V}\mathrm{ar}\left(\sum_{i\in\mathcal{N}} \frac{\alpha_i}{1-q_i}\left(\boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t)}\right)\mathbb{1}_i^{(t)}\right) =$$
$$= \sum_{i\in\mathcal{N}} \mathbb{V}\mathrm{ar}\left(\frac{\alpha_i}{1-q_i}\left(\boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t)}\right)\mathbb{1}_i^{(t)}\right) =$$
$$= \sum_{i\in\mathcal{N}} \frac{{\alpha_i}^2}{(1-q_i)^2}\left\|\boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t)}\right\|^2 \mathrm{Var}\left(\mathbb{1}_i^{(t)}\right) =$$
$$= \sum_{i\in\mathcal{N}} \alpha_i^2 \frac{q_i}{1-q_i}\left\|\boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t)}\right\|^2. \tag{C.5}$$

Finally:

$$\mathbb{E}_{\mathcal{S}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{w}^{(t+1)} - \bar{\boldsymbol{w}}^{(t,K)}\right\|^2 = \sum_{i\in\mathcal{N}} \alpha_i^2 \frac{q_i}{1-q_i}\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{w}_i^{(t,K)} - \boldsymbol{w}^{(t)}\right\|^2$$
$$\leq \sum_{i\in\mathcal{N}} \alpha_i^2 \frac{q_i}{1-q_i}(\eta^{(t)})^2 K^2 G^2. \tag{C.6}$$

Conversely, for $\boldsymbol{w}^{(t)} = \boldsymbol{w}^{(t)}_{\text{UDMA-PL}}$, the same proof technique leads to the bound in (C.4), but the steps in (C.6) do not hold.

□

*Proof of Theorem 4.3.1.*

$$\mathbb{E}_{\mathcal{S}^{(t)}|\mathcal{H}^{(t)},\mathcal{B}^{(t)}}\left\|\boldsymbol{w}^{(t+1)} - \boldsymbol{w}^*\right\|^2 =$$
$$= \mathbb{E}_{\mathcal{S}^{(t)}|\mathcal{H}^{(t)},\mathcal{B}^{(t)}}\left\|\boldsymbol{w}^{(t+1)} - \bar{\boldsymbol{w}}^{(t,K)}\right\|^2 + \left\|\bar{\boldsymbol{w}}^{(t,K)} - \boldsymbol{w}^*\right\|^2. \tag{C.7}$$

From (X. Li et al., 2020, Lemma 1), (X. Li et al., 2020, Lemma 2), and (X. Li et al., 2020, Lemma 3), recursively:

$$\mathbb{E}_{\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\bar{\boldsymbol{w}}^{(t,K)} - \boldsymbol{w}^*\right\|^2 \leq$$
$$\leq (1-\eta^{(t)}\mu)^K\left\|\bar{\boldsymbol{w}}^{(t)} - \boldsymbol{w}^*\right\|^2 + (\eta^{(t)})^2 B\sum_{k=0}^{K-1}(1-\eta^{(t)}\mu)^j \tag{C.8}$$
$$\leq (1-\eta^{(t)}\mu)\left\|\bar{\boldsymbol{w}}^{(t)} - \boldsymbol{w}^*\right\|^2 + (\eta^{(t)})^2 KB, \tag{C.9}$$

where $B = \sum_{i\in\mathcal{N}} \alpha_i^2\sigma_i^2 + 6L\Gamma + 2(K-1)^2G^2$.

Combining (C.7) and (C.9), and applying Lemma A.1, we have:

$$\mathbb{E}_{\mathcal{S}^{(t)},\mathcal{B}^{(t)}|\mathcal{H}^{(t)}}\left\|\boldsymbol{w}^{(t+1)} - \boldsymbol{w}^*\right\|^2 \leq (1-\eta^{(t)}\mu)\left\|\bar{\boldsymbol{w}}^{(t)} - \boldsymbol{w}^*\right\|^2 + (\eta^{(t)})^2 KC. \tag{C.10}$$

The conclusion of the proof follows similar steps as (X. Li et al., 2020, Theorem 1). We require a learning rate $\eta^{(t)} \leq (\frac{1}{\mu}, \frac{1}{4L}) = \frac{1}{4L}$. Set $\eta^{(t+1)} \leq \frac{2/\mu}{8\kappa+t}$, with $\kappa := L/\mu$, such that $\eta^{(1)} = \frac{1}{4L}$. Then:

$$\mathbb{E}\left[F(\boldsymbol{w}^{(t+1)})\right] - F^* \leq \frac{\kappa}{8\kappa+t}\left(\frac{2KC}{\mu} + 4L\left\|\boldsymbol{w}^{(1)} - \boldsymbol{w}^*\right\|^2\right). \tag{C.11}$$

□

# APPENDIX D

# Cooperative Inference Systems: The Case of Early Exit Networks

## A Error Decomposition

We start upperbounding the true error by three terms: a generalization error, a bias error (due to the mismatch between $F_{\mathcal{D},\tilde{\boldsymbol{\Lambda}}}$ and $F_{\mathcal{D},\boldsymbol{\Lambda}}$), and an optimization error:

$$\epsilon_{\text{true}} \leq 2\mathbb{E}_D\left[\sup_{\boldsymbol{w}}\left|F_{D,\tilde{\boldsymbol{\Lambda}}}(\boldsymbol{w}) - F_{\mathcal{D},\boldsymbol{\Lambda}}(\boldsymbol{w})\right|\right] + \mathbb{E}_{S,A_{\tilde{\boldsymbol{\Lambda}}}}\left[F_{D,\tilde{\boldsymbol{\Lambda}}}(\boldsymbol{w}^{(T)}) - F^{\star}_{D,\tilde{\boldsymbol{\Lambda}}}\right] \tag{D.1}$$

$$\leq \underbrace{2\,\mathbb{E}_D\left[\sup_{\boldsymbol{w}}\left|F_{D,\tilde{\boldsymbol{\Lambda}}}(\boldsymbol{w}) - F_{\mathcal{D},\tilde{\boldsymbol{\Lambda}}}(\boldsymbol{w})\right|\right]}_{\epsilon_{\text{gen}}} + \underbrace{2\,\mathbb{E}_D\left[\sup_{\boldsymbol{w}}\left|F_{\mathcal{D},\tilde{\boldsymbol{\Lambda}}}(\boldsymbol{w}) - F_{\mathcal{D},\boldsymbol{\Lambda}}(\boldsymbol{w})\right|\right]}_{\epsilon_{\text{bias}}} + \underbrace{\mathbb{E}_{S,A_{\tilde{\boldsymbol{\Lambda}}}}\left[F_{D,\tilde{\boldsymbol{\Lambda}}}(\boldsymbol{w}^{(T)}) - F^{\star}_{D,\tilde{\boldsymbol{\Lambda}}}\right]}_{\epsilon_{\text{opt}}}, \tag{D.2}$$

where the first inequality is quite standard (e.g., (Marfoq, Neglia, Kameni, & Vidal, 2023, Eq. 9). We obtain the final result by bounding each term.

## B Generalization Error

For the generalization term, let $F_{\mathcal{D},e}(\boldsymbol{w}) \triangleq \mathbb{E}_{z\sim\mathcal{D}}[f^{(e)}(\boldsymbol{w},z)]$, we observe that

$$\begin{aligned}\epsilon_{\text{gen}} &\leq \sum_{e=1}^{E}\tilde{\Lambda}_e\mathbb{E}_D\left[\sup_{\boldsymbol{w}}\left|\left(\sum_{i\in\mathcal{C}_e}\frac{|D_i|}{|D_{e,\boldsymbol{p}}|}F_{i,e}(\boldsymbol{w})\right) - F_{\mathcal{D},e}(\boldsymbol{w})\right|\right] \\ &= \sum_{e=1}^{E}\tilde{\Lambda}_e\mathbb{E}_D\left[\sup_{\boldsymbol{w}}\left|F_{D_{e,\boldsymbol{p}}}(\boldsymbol{w}) - F_{\mathcal{D},e}(\boldsymbol{w})\right|\right].\end{aligned} \tag{D.3}$$

We can then bound the (expected) representativity $\mathbb{E}_D\left[\sup_{\boldsymbol{w}}\left|F_{D_{e,\boldsymbol{p}}}(\boldsymbol{w}) - F_{\mathcal{D},e}(\boldsymbol{w})\right|\right]$ for each exit $e$. Our task is not necessarily a binary classification task, but its representativity can be bounded by the representativity of an opportune classification task with the 0-1 loss and set of classifiers $H'_e = \{h_{\boldsymbol{w},t}(z), \boldsymbol{w}\in\mathcal{W}, t\in\mathbb{R}^+\}$, where $h_{\boldsymbol{w},t}(z) = \mathbb{1}_{f^{(e)}(\boldsymbol{w},z)>t}$ (Mohri et al., 2018, Sec. 11.2.3). In particular, let $\mathcal{R}_D(H)$ denote the Rademacher complexity of class $H$

on dataset $D$ and let $F'_{D_{e,\boldsymbol{p}}}(\boldsymbol{w},t)$ and $F'_{\mathcal{D},e}(\boldsymbol{w},t)$ denote the empirical loss and the expected loss for such classification problem, respectively. The analysis is then quite standard:

$$\begin{aligned}
&\mathbb{E}_D\left[\sup_{\boldsymbol{w}}\left|F_{D_{e,\boldsymbol{p}}}(\boldsymbol{w})-F_{\mathcal{D},e}(\boldsymbol{w})\right|\right]\\
&\quad\leq M\mathbb{E}_D\left[\sup_{\boldsymbol{w}}\left|F'_{D_{e,\boldsymbol{p}}}(\boldsymbol{w},t)-F'_{\mathcal{D},e}(\boldsymbol{w},t)\right|\right] &\text{(D.4)}\\
&\quad\leq 2M\mathbb{E}_{D_{e,\boldsymbol{p}}}\left[\mathcal{R}_{D_{e,\boldsymbol{p}}}(H'_e)\right] &\text{(D.5)}\\
&\quad\leq MC\sqrt{\frac{\mathrm{VCdim}(H'_e)}{|D_{e,\boldsymbol{p}}|}} &\text{(D.6)}\\
&\quad= MC\sqrt{\frac{\mathrm{Pdim}(H_e)}{|D_{e,\boldsymbol{p}}|}}. &\text{(D.7)}
\end{aligned}$$

For a proof of the three inequalities the reader can refer to (Mohri et al., 2018, Thm. 11.8), (Shalev-Shwartz & Ben-David, 2014, Lm. 26.2), (Bousquet, Boucheron, & Lugosi, 2004-09-06, 2003, Sec. 5), respectively (the constant $C$ can be selected to be 320 (Livesay, 2017, Cor. 6.4)). The final equality follows from the definition of pseudo-dimension.

# C Bias Error

For the bias term $\epsilon_{\text{bias}}$, it is sufficient to observe that

$$\begin{aligned}
\epsilon_{\text{bias}}&\leq\mathbb{E}_D\left[\sup_{\boldsymbol{w}}\left|\sum_{e=1}^{E}\left(\tilde{\Lambda}_e-\Lambda_e\right)F_{\mathcal{D},e}(\boldsymbol{w})\right|\right] &\text{(D.8)}\\
&\leq 2M\,\mathrm{dist}_{\mathrm{TV}}(\tilde{\boldsymbol{\Lambda}},\boldsymbol{\Lambda}). &\text{(D.9)}
\end{aligned}$$

Finally, for the optimization term, we can consider the pair $(c,e)$ to be a fictitious client in a usual FL system and adapt the proofs in (Salehi & Hossain, 2021; Rodio, Neglia, et al., 2023) to take into account 1) negative-correlation across fictitious clients' participation (every client $c$ only trains one exit at each round) and 2) the projection step.

# D Optimization Error

Our proof is similar to the proofs in (Salehi & Hossain, 2021; Rodio, Neglia, et al., 2023). We adapt our notation to follow more closely that in those papers.

Let us consider the node update rule and the server aggregation rule in our algorithm:

$$\boldsymbol{w}_{i,e}^{(t,j+1)}=\boldsymbol{w}_{t,j}^{(i,e)}-\eta_{t,j}\frac{1}{\left|\mathcal{B}_{i,e}^{(t,j)}\right|}\sum_{z\in\mathcal{B}_{i,e}^{(t,j)}}\nabla f_e(\boldsymbol{w}_{t,j}^{(i,e)},z),\ \text{for } j=0,\dots,J-1 \tag{D.10}$$

$$\boldsymbol{w}^{(t+1)}=\boldsymbol{w}^{(t)}+\eta_s\sum_{e\in E}\tilde{\Lambda}_e\sum_{i\in\mathcal{N}_{t,e}}\frac{|D_i|}{|D_{e,\boldsymbol{p}}|}\frac{1}{p_{i,e}}(\boldsymbol{w}_{i,e}^{(t,J)}-\boldsymbol{w}^{(t)}),\ \text{for } t=1,\dots T. \tag{D.11}$$

We consider that a node corresponds to the pair $k\triangleq(i,e)\in\mathcal{K}$, where $\mathcal{K}\triangleq\mathcal{N}\times\mathcal{E}, i\in\mathcal{N},\ e\in\mathcal{E}$, and we define $\alpha_k\triangleq\alpha_{i,e}\triangleq\eta_s\tilde{\Lambda}_e\frac{|D_i|}{|D_{e,\boldsymbol{p}}|}$, $\xi_k^{(t)}\triangleq\xi_{i,e}^{(t)}\triangleq\mathbb{1}_{i\in\mathcal{N}_{t,e}}$, and $\nabla F_k(\boldsymbol{w}_{t,j}^{(i,e)},\mathcal{B}_k^{(\tau)})\triangleq\frac{1}{|\mathcal{B}_{i,e}^{(t,j)}|}\sum_{z\in\mathcal{B}_{i,e}^{(t,j)}}\nabla f_e(\boldsymbol{w}_{t,j}^{(i,e)},z)$.

Moreover, we count gradient steps at nodes and aggregation steps at the server using the same time sequence $(\tau = J(t-1)+j)_{t=1,\dots,T, j=0,\dots,J-1}$. The set of values $\mathcal{I}^{(J)} = \{Jt, t = 1,\dots,T\}$ corresponds to the aggregation steps. The equations above can then be rewritten as follows in terms of two new virtual sequences:

$$\boldsymbol{v}_k^{(\tau+1)} = \boldsymbol{w}_k^{(\tau)} - \eta_\tau \nabla F_k(\boldsymbol{w}_k^{(\tau)}, \mathcal{B}_k^{(\tau)}) \tag{D.12}$$

$$\boldsymbol{w}_k^{(\tau+1)} = \begin{cases} \boldsymbol{w}^{(1)} & \text{for } \tau+1 = 0, \\ \Pi_{\mathcal{W}}\left(\boldsymbol{w}_k^{(\tau+1-J)} + \sum_{k\in\mathcal{K}} \frac{\alpha_k \xi_k^{(\tau+1-J)}}{p_k}(\boldsymbol{v}_k^{(\tau+1)} - \boldsymbol{w}_k^{(\tau+1-J)})\right) & \text{for } \tau+1 \in \mathcal{I}^{(J)}, \\ \boldsymbol{v}_k^{(\tau+1)} & \text{otherwise.} \end{cases} \tag{D.13}$$

$\boldsymbol{v}_k^{(J(t-1)+j)}$ coincides then with the local model $\boldsymbol{w}_{t,j}^{(i,e)}$ and $\boldsymbol{w}_k^{(J(t-1))}$ coincides with the global model $\boldsymbol{w}^{(t)}$.

We observe that, for $\tau+1 \in \mathcal{I}^{(J)}$, $\boldsymbol{w}_k^{(\tau+1)} = \boldsymbol{w}_{k'}^{(\tau+1)}$ for any $k$ and $k'$, and that, for $\tau+1 \notin \mathcal{I}^{(J)}$, $\boldsymbol{v}_k^{(\tau+1)} = \boldsymbol{w}_k^{(\tau+1)}$. Moreover, define the average sequences $\bar{\boldsymbol{v}}^{(\tau+1)} = \sum_{k\in\mathcal{K}} \alpha_k \boldsymbol{v}_k^{(\tau+1)}$ and $\bar{\boldsymbol{w}}^{(\tau+1)} = \sum_{k\in\mathcal{K}} \alpha_k \boldsymbol{w}_k^{(\tau+1)}$ and similarly the average gradients $\boldsymbol{g}^{(\tau)} = \sum_{k\in\mathcal{K}} \alpha_k \nabla F_k(\boldsymbol{w}_k^{(\tau)}, \mathcal{B}_k^{(t)})$ and $\bar{\boldsymbol{g}}^{(\tau)} = \sum_{k\in\mathcal{K}} \alpha_k \nabla F_k(\boldsymbol{w}_k^{(\tau)})$. We also define the sequence

$$\bar{\boldsymbol{w}}^{(\tau+1)\dagger} = \begin{cases} \boldsymbol{w}_k^{(\tau+1-J)} + \sum_{k\in\mathcal{K}} \frac{\alpha_k \xi_k^{(\tau+1-J)}}{p_k}(\boldsymbol{v}_k^{(\tau+1)} - \boldsymbol{w}_k^{(\tau+1-J)}), & \text{for } \tau+1 \in \mathcal{I}^{(J)} \\ \bar{\boldsymbol{w}}^{(\tau+1)}, & \text{otherwise.} \end{cases} \tag{D.14}$$

We note that $\bar{\boldsymbol{w}}^{(\tau+1)} = \Pi_{\mathcal{W}}\left(\bar{\boldsymbol{w}}^{(\tau+1)\dagger}\right)$ for $\tau+1 \in \mathcal{I}^{(J)}$ and coincide otherwise.

We denote by $\mathcal{B}^{(\tau)} = (\mathcal{B}_k^{(\tau)})_{k\in\mathcal{K}}$ and $\xi^{(\tau)} = (\xi_k^{(\tau)})_{k\in\mathcal{K}}$, the set of batches and the set of indicator variables for node participation at instant $\tau$. The history of the system at time $\tau$ is made by the values of the random variables until that time and it can be defined by recursion as follows: $\mathcal{H}^{(1)} = \emptyset$, $\mathcal{H}^{(\tau+1)} = \{\xi^{(\tau+1)}, \mathcal{B}^{(\tau)}, \mathcal{H}^{(\tau)}\}$ if $\tau+1 \in \mathcal{I}^{(J)}$ and $\mathcal{H}^{(\tau+1)} = \{\mathcal{B}^{(\tau)}, \mathcal{H}^{(\tau)}\}$, otherwise.

We define $G_{i,e} \triangleq \sigma_{i,e}^2 + (L \operatorname{diam}(\mathcal{W}))^2$ and observe that it bounds the second moment of the stochastic gradient at $(i,e)$:

$$\mathbb{E}\left[\|\nabla F_{i,e}(\boldsymbol{w}, \mathcal{B})\|^2\right] = \mathbb{E}\left[\|\nabla F_{i,e}(\boldsymbol{w}, \mathcal{B}) - \nabla F_{i,e}(\boldsymbol{w})\|^2\right] + \|\nabla F_{i,e}(\boldsymbol{w})\|^2 \tag{D.15}$$

$$\leq \sigma_{i,e}^2 + L^2 \left\|\boldsymbol{w} - \boldsymbol{w}_{i,e}^*\right\|^2 \tag{D.16}$$

$$\leq \sigma_{i,e}^2 + L^2 \operatorname{diam}(\mathcal{W})^2 \tag{D.17}$$

$$= G_{i,e}, \tag{D.18}$$

where we have used Assumption 16. We also define a uniform bound over all nodes and all exits: $G \triangleq \max_{(i,e)\in\mathcal{N}\times\mathcal{E}} G_{i,e}$.

Similarly to other works (X. Li et al., 2020; T. Li, Sahu, Zaheer, et al., 2020; J. Wang et al., 2020, 2021), we introduce a metric to quantify the heterogeneity of nodes' local datasets, typically referred to as *statistical heterogeneity*:

$$\Gamma \triangleq \max_{(i,e)\in\mathcal{N}\times\mathcal{E}} F_{i,e}(\boldsymbol{w}_{\boldsymbol{\Lambda}}^\star) - F_{i,e}^\star. \tag{D.19}$$

Finally, we define $h(\tau) \triangleq \max\{\tau' \in \mathcal{I}^{(J)} : \tau' \leq \tau\}$. Then $h(\tau)$ indicates the time of the last server update before $\tau$.

The following lemma corresponds to (Salehi & Hossain, 2021, Lemma 4).

**Lemma D.1.**

$$\mathbb{E}_{\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}}\left\|\bar{\boldsymbol{v}}^{(\tau+1)}-\boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\| \leq (1-\eta_\tau\mu)\mathbb{E}_{\mathcal{B}^{(\tau-1)},\ldots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}}\left\|\bar{\boldsymbol{w}}^{(\tau)}-\boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\| + \eta_\tau^2\left(\sum_{k\in\mathcal{K}}\alpha_k^2\sigma_k^2+6L\Gamma+8(J-1)^2G^2\right). \tag{D.20}$$

*Proof.*

From (X. Li et al., 2020, Lemma 1):

$$\mathbb{E}_{\mathcal{B}^{(\tau)}|\mathcal{H}^{(\tau)}}\left\|\bar{\boldsymbol{v}}^{(\tau+1)}-\boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\| \leq (1-\eta_\tau\mu)\left\|\bar{\boldsymbol{w}}^{(\tau)}-\boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\| + \eta_\tau^2\mathbb{E}_{\mathcal{B}^{(\tau)}|\mathcal{H}^{(\tau)}}\left\|\boldsymbol{g}^{(\tau)}-\bar{\boldsymbol{g}}^{(\tau)}\right\|^2 + \eta_\tau^2 6L\Gamma + 2\sum_{k\in\mathcal{K}}\alpha_k\left\|\boldsymbol{w}_k^{(\tau)}-\bar{\boldsymbol{w}}^{(\tau)}\right\|^2. \tag{D.21}$$

From (X. Li et al., 2020, Lemma 2):

$$\mathbb{E}_{\mathcal{B}^{(\tau)}|\mathcal{H}^{(\tau)}}\left\|\boldsymbol{g}^{(\tau)}-\bar{\boldsymbol{g}}^{(\tau)}\right\|^2 \leq \mathbb{E}_{\mathcal{B}^{(\tau)}|\mathcal{H}^{(\tau)}}\left\|\sum_{k\in\mathcal{K}}\alpha_k\left(\nabla F_k(\boldsymbol{w}_k^{(t)},\mathcal{B}_k^{(\tau)})-\nabla F_k(\boldsymbol{w}_k^{(t)})\right)\right\|^2 \tag{D.22}$$

$$= \sum_{k\in\mathcal{K}}\alpha_k^2\mathbb{E}_{\mathcal{B}_k^{(\tau)}|\mathcal{H}^{(t)}}\left\|\nabla F_k(\boldsymbol{w}_k^{(t)},\mathcal{B}_k^{(\tau)})-\nabla F_k(\boldsymbol{w}_k^{(t)})\right\|^2 \tag{D.23}$$

$$\leq \sum_{k\in\mathcal{K}}\alpha_k^2\sigma_k^2. \tag{D.24}$$

Combining the two inequalities above:

$$\mathbb{E}_{\mathcal{B}^{(\tau)}|\mathcal{H}^{(\tau)}}\left\|\bar{\boldsymbol{v}}^{(\tau+1)}-\boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\| \leq (1-\eta_\tau\mu)\left\|\bar{\boldsymbol{w}}^{(\tau)}-\boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\| + \eta_\tau^2\sum_{k\in\mathcal{K}}\alpha_k^2\sigma_k^2 + \eta_\tau^2 6L\Gamma + 2\sum_{k\in\mathcal{K}}\alpha_k\left\|\boldsymbol{w}_k^{(\tau)}-\bar{\boldsymbol{w}}^{(\tau)}\right\|^2. \tag{D.25}$$

By definition of $h(\tau)$, we observe that $0 \leq \tau - h(\tau) \leq J-1$ and $\mathcal{H}^{(\tau)} = \{\mathcal{B}^{(\tau-1)}, B^{(\tau-2)},\ldots,B^{(h(\tau))},\mathcal{H}^{(h(\tau))}\}$.

From (X. Li et al., 2020, Lemma 3):

$$\sum_{k\in\mathcal{K}}\alpha_k\mathbb{E}_{\mathcal{B}^{(\tau-1)},\ldots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}}\left\|\boldsymbol{w}_k^{(\tau)}-\bar{\boldsymbol{w}}^{(\tau)}\right\|^2$$

$$= \sum_{k\in\mathcal{K}}\alpha_k\mathbb{E}_{\mathcal{B}^{(\tau-1)},\ldots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}}\left\|(\boldsymbol{w}_k^{(\tau)}-\bar{\boldsymbol{w}}^{(h(\tau))})-(\bar{\boldsymbol{w}}^{(\tau)}-\bar{\boldsymbol{w}}^{(h(\tau))})\right\|^2 \tag{D.26}$$

$$\leq \sum_{k\in\mathcal{K}}\alpha_k\mathbb{E}_{\mathcal{B}^{(\tau-1)},\ldots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}}\left\|\boldsymbol{w}_k^{(\tau)}-\bar{\boldsymbol{w}}^{(h(\tau))}\right\|^2 \tag{D.27}$$

$$= \sum_{k\in\mathcal{K}}\alpha_k\mathbb{E}_{\mathcal{B}^{(\tau-1)},\ldots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}}\left\|\sum_{i=h(\tau)}^{t-1}\eta_i\nabla F_k(\boldsymbol{w}_k^{(i)},\mathcal{B}_k^{(i)})\right\|^2 \tag{D.28}$$

$$\leq \sum_{k\in\mathcal{K}} \alpha_k(\tau - h(\tau))\mathbb{E}_{\mathcal{B}^{(\tau-1)},\dots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left[\sum_{i=h(\tau)}^{t-1} \eta_i^2 \left\|\nabla F_k(\boldsymbol{w}_k^{(i)}, \mathcal{B}_k^{(i)})\right\|^2\right] \tag{D.29}$$

$$\leq \eta_{h(\tau)}^2 (t - h(\tau))^2 G^2 \tag{D.30}$$

$$\leq 4\eta_\tau^2 (J-1)^2 G^2. \tag{D.31}$$

By repeatedly computing expectations over the previous batch conditioned on the previous history and combining the inequalities above, we obtain:

$$\mathbb{E}_{\mathcal{B}^{(\tau)},\dots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left\|\bar{\boldsymbol{v}}^{(\tau+1)} - \boldsymbol{w}_{\tilde{\boldsymbol{\Lambda}}}^\star\right\| \leq (1 - \eta_\tau\mu)\mathbb{E}_{\mathcal{B}^{(\tau-1)},\dots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left\|\bar{\boldsymbol{w}}^{(\tau)} - \boldsymbol{w}_{\tilde{\boldsymbol{\Lambda}}}^\star\right\| + \eta_\tau^2 \left(\sum_{k\in\mathcal{K}} \alpha_k^2\sigma_k^2 + 6L\Gamma + 8(J-1)^2G^2\right). \tag{D.32}$$

□

The following lemma corresponds to (Salehi & Hossain, 2021, Lemma 2), but it needs to be adapted to take into account the projection.

**Lemma D.2.**

$$\mathbb{E}_{\xi^{(h(\tau))}|\mathcal{B}^{(\tau)},\dots,\mathcal{B}^{(h(\tau)+1)},\mathcal{H}^{(h(\tau))}}[\bar{\boldsymbol{w}}^{(\tau+1)\dagger}] = \bar{\boldsymbol{v}}^{(\tau+1)}. \tag{D.33}$$

*Proof.*

First, we observe that $\bar{\boldsymbol{w}}^{(\tau+1)\dagger} = \bar{\boldsymbol{w}}^{(\tau+1)} = \bar{\boldsymbol{v}}^{(\tau+1)}$ for $\tau + 1 \notin \mathcal{I}^{(J)}$. For $\tau + 1 \in \mathcal{I}^{(J)}$, $h(\tau) = \tau + 1 - J$ and

$$\mathbb{E}_{\xi^{(\tau+1-J)}|\mathcal{B}^{(\tau)},\dots,\mathcal{B}^{(\tau+1-J)},\mathcal{H}^{(\tau+1-J)}}[\bar{\boldsymbol{w}}^{(\tau+1)\dagger}] =$$

$$= \bar{\boldsymbol{w}}^{(\tau+1-J)} - \sum_{k\in\mathcal{K}} \frac{\alpha_k \mathbb{E}[\xi_k^{(\tau+1-J)}]}{p_k} \sum_{j=0}^{J-1} \eta_{\tau+1-J+j} \nabla F_k(\boldsymbol{w}_k^{(\tau+1-J+j)}, \mathcal{B}_k^{(\tau+1-J+j)}) \tag{D.34}$$

$$= \bar{\boldsymbol{w}}^{(\tau+1-J)} - \sum_{k\in\mathcal{K}} \alpha_k \sum_{j=0}^{J-1} \eta_{\tau+1-J+j} \nabla F_k(\boldsymbol{w}_k^{(\tau+1-J+j)}, \mathcal{B}_k^{(\tau+1-J+j)}) \tag{D.35}$$

$$= \bar{\boldsymbol{v}}^{(\tau+1)}. \tag{D.36}$$

□

The following lemma corresponds to (Salehi & Hossain, 2021, Lemma 3). We modify the proof to take into account the correlation in the participation of the fictitious nodes in $\mathcal{K}$. Indeed, each node $i$ selects a single exit to train and then the random variables $\{\xi^{(h(\tau))}\}_{e\in E}$ are (negatively) correlated.

**Lemma D.3.**

$$\mathbb{E}_{\mathcal{B}^{(\tau)},\dots,\mathcal{B}^{(h(\tau))},\xi^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left\|\bar{\boldsymbol{w}}^{(\tau+1)\dagger} - \bar{\boldsymbol{v}}^{(\tau+1)}\right\|^2 \leq 4\eta_\tau^2 J^2 G^2 \sum_{i=1}^{N} \left(\sum_{e\in E_i} \frac{\alpha_{i,e}^2}{p_{i,e}} - \left(\sum_{e\in E_i} \alpha_{i,e}\right)^2\right). \tag{D.37}$$

*Proof.*

We have a tighter bound ($\alpha_k^2$ instead of $\alpha_k$), observing that $\mathrm{Var}(X) = \mathbb{E}[X - \mathbb{E}[X]]^2$. Let $\boldsymbol{X}$ be a d-dimensional random variable, we define its variance as follows: $\mathbb{V}\mathrm{ar}(\boldsymbol{X}) \triangleq \sum_{i=1}^{d} \mathrm{Var}(X_i)$. We also denote by $E_i$ the set of exits node $i$ may train, i.e., $E_i \triangleq \{e : p_{i,e} > 0, e = 1, \ldots, E\}$.

In order to keep the following calculations simpler to follow, we denote by $\boldsymbol{U}_{i,e} = \sum_{j=0}^{\tau-h(\tau)} \eta_{h(\tau)+j} \nabla F_{i,e}(\boldsymbol{w}_{i,e}^{(h(\tau)+j)}, \mathcal{B}_{i,e}^{(h(\tau)+j)})$.

$$\mathbb{E}_{\xi^{(h(\tau))}|\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))},\mathcal{H}^{(h(\tau))}} \left\| \bar{\boldsymbol{w}}^{(\tau+1)\dagger} - \bar{\boldsymbol{v}}^{(\tau+1)} \right\|^2$$

$$= \mathbb{V}\mathrm{ar}_{\xi^{(h(\tau))}|\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))},\mathcal{H}^{(h(\tau))}} \left( \sum_{k\in\mathcal{K}} \frac{\alpha_k \xi_k^{(h(\tau))}}{p_k} \sum_{j=0}^{\tau-h(\tau)} \eta_{h(\tau)+j} \nabla F_k(\boldsymbol{w}_k^{(h(\tau)+j)}, \mathcal{B}_k^{(h(\tau)+j)}) \right) \tag{D.38}$$

$$= \mathbb{V}\mathrm{ar}_{\xi^{(h(\tau))}|\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))},\mathcal{H}^{(h(\tau))}} \left( \sum_{i=1}^{N} \sum_{e\in E_i} \frac{\alpha_{i,e} \xi_{i,e}^{(h(\tau))}}{p_{i,e}} \boldsymbol{U}_{i,e} \right) \tag{D.39}$$

$$= \sum_{i=1}^{N} \mathbb{V}\mathrm{ar}_{\xi^{(h(\tau))}|\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))},\mathcal{H}^{(h(\tau))}} \left( \sum_{e\in E_i} \frac{\alpha_{i,e} \xi_{i,e}^{(h(\tau))}}{p_{i,e}} \boldsymbol{U}_{i,e} \right) \tag{D.40}$$

$$\leq \sum_{i=1}^{N} \sum_{e\in E_i} \mathbb{V}\mathrm{ar}_{\xi^{(h(\tau))}|\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))},\mathcal{H}^{(h(\tau))}} \left( \frac{\alpha_{i,e} \xi_{i,e}^{(h(\tau))}}{p_{i,e}} \boldsymbol{U}_{i,e} \right) \tag{D.41}$$

$$= \sum_{i=1}^{N} \sum_{e\in E_i} \mathrm{Var} \left( \frac{\alpha_{i,e} \xi_{i,e}^{(h(\tau))}}{p_{i,e}} \right) \left\| \boldsymbol{U}_{i,e} \right\|^2 \tag{D.42}$$

$$= \sum_{i=1}^{N} \sum_{e\in E_i} \alpha_{i,e}^2 \frac{1 - p_{i,e}}{p_{i,e}} \left\| \boldsymbol{U}_{i,e} \right\|^2 . \tag{D.43}$$

where (D.41) takes into account that $\xi_{i,e}^{(\tau+1-J)} \xi_{i,e'}^{(\tau+1-J)} = 0$ for $e \neq e'$ because each node selects a single exit to train.

Then, the expectation over the random batches is computed

$$\mathbb{E}_{\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))},\xi^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left\| \bar{\boldsymbol{w}}^{(\tau+1)\dagger} - \bar{\boldsymbol{v}}^{(\tau+1)} \right\|^2$$

$$\leq \sum_{i=1}^{N} \sum_{e\in E_i} \alpha_{i,e}^2 \frac{1 - p_{i,e}}{p_{i,e}} \mathbb{E}_{\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left[ \left\| \boldsymbol{U}_{i,e} \right\|^2 \right] \tag{D.44}$$

$$\leq \sum_{i=1}^{N} \sum_{e\in E_i} \alpha_{i,e}^2 \frac{1 - p_{i,e}}{p_{i,e}} \mathbb{E}_{\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left[ \left\| \sum_{j=0}^{\tau-h(\tau)} \eta_{h(\tau)+j} \nabla F_{i,e}(\boldsymbol{w}_{i,e}^{(h(\tau)+j)}, \mathcal{B}_{i,e}^{(h(\tau)+j)}) \right\|^2 \right] \tag{D.45}$$

$$\leq \eta_{h(\tau)+j}^2 J \sum_{i=1}^{N} \sum_{e\in E_i} \alpha_{i,e}^2 \frac{1 - p_{i,e}}{p_{i,e}} \sum_{j=0}^{\tau-h(\tau)} \mathbb{E}_{\mathcal{B}^{(\tau)},\ldots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left[ \left\| \nabla F_{i,e}(\boldsymbol{w}_{i,e}^{(h(\tau)+j)}, \mathcal{B}_{i,e}^{(h(\tau)+j)}) \right\|^2 \right] \tag{D.46}$$

$$\leq \eta_{h(\tau)+j}^2 J^2 \sum_{i=1}^{N} \sum_{e\in E_i} \alpha_{i,e}^2 \frac{1 - p_{i,e}}{p_{i,e}} G_{i,e} \tag{D.47}$$

$$\leq 4\eta_\tau^2 J^2 \sum_{i=1}^{N} \sum_{e\in E_i} \alpha_{i,e}^2 \frac{1-p_{i,e}}{p_{i,e}} G_{i,e}, \tag{D.48}$$

where (D.48) uses $\eta_{h(\tau)+j} \leq \eta_{\tau-J} \leq 2\eta_\tau$.

□

**Theorem D.4.** *Under Assumptions 16–19, the optimization error of Algorithm 6 with learning rate* $\eta_{t,j} = \frac{2}{\mu(\gamma+(t-1)J+j+1)}$ *and* $\gamma \triangleq \max\{8\kappa, J\} - 1$ *can be bounded as follows:*

$$\mathbb{E}\left[F_{D,\tilde{\boldsymbol{\Lambda}}}(\boldsymbol{w}^{(T)})\right] - F^\star_{D,\tilde{\boldsymbol{\Lambda}}} = \frac{\kappa}{\gamma+JT}\left(\frac{2B}{\mu} + \frac{\mu(\gamma+1)}{2}\mathbb{E}\left[\boldsymbol{w}^{(1)} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right]\right), \tag{D.49}$$

*where*

$$B \triangleq \sum_{(i,e)\in\mathcal{N}\times\mathcal{E}} \alpha_{i,e}^2\sigma_{i,e}^2 + 6L\Gamma + 8(J-1)^2G^2 + 4J^2\sum_{i=1}^{N}\sum_{e\in E_i}\alpha_{i,e}^2\frac{1-p_{i,e}}{p_{i,e}}G_{i,e}, \tag{D.50}$$

$$G_{i,e} \triangleq \sigma_{i,e}^2 + (L\operatorname{diam}(\mathcal{W}))^2, \tag{D.51}$$

$$G \triangleq \max_{(i,e)\in\mathcal{N}\times\mathcal{E}} G_{i,e}, \tag{D.52}$$

$$\alpha_{i,e} \triangleq \eta_s\tilde{\Lambda}_e\frac{|D_i|}{|D_{e,\boldsymbol{p}}|}. \tag{D.53}$$

*Proof.*

As we mention at the beginning of this appendix, we count gradient steps at nodes and aggregation steps at the server using the same time sequence $(\tau = J(t-1)+j)_{t=1,\dots,T,j=0,\dots,J-1}$. The set of values $\mathcal{I}^{(J)} = \{Jt, t = 1,\dots,T\}$ corresponds to the aggregation steps.

We have

$$\left\|\bar{\boldsymbol{w}}^{(\tau+1)} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 \leq \left\|\bar{\boldsymbol{w}}^{(\tau+1)\dagger} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 \tag{D.54}$$

$$= \left\|\bar{\boldsymbol{w}}^{(\tau+1)\dagger} - \bar{\boldsymbol{v}}^{(\tau+1)} + \bar{\boldsymbol{v}}^{(\tau+1)} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 \tag{D.55}$$

$$= \left\|\bar{\boldsymbol{w}}^{(\tau+1)\dagger} - \bar{\boldsymbol{v}}^{(\tau+1)}\right\|^2 + \left\|\bar{\boldsymbol{v}}^{(\tau+1)} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 + 2\langle\bar{\boldsymbol{w}}^{(\tau+1)\dagger} - \bar{\boldsymbol{v}}^{(\tau+1)}, \bar{\boldsymbol{v}}^{(\tau+1)} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\rangle, \tag{D.56}$$

where the first inequality is trivially true for $\tau+1 \notin \mathcal{I}^{(J)}$ because $\bar{\boldsymbol{w}}^{(\tau+1)} = \bar{\boldsymbol{w}}^{(\tau+1)\dagger}$, while for $\tau+1 \notin \mathcal{I}^{(J)}$, it follows from Assumption 16 and $\left\|\bar{\boldsymbol{w}}^{(\tau+1)} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 = \left\|\Pi_{\mathcal{W}}(\bar{\boldsymbol{w}}^{(\tau+1)\dagger}) - \Pi_{\mathcal{W}}(\boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}})\right\|^2 \leq \left\|\bar{\boldsymbol{w}}^{(\tau+1)\dagger} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right\|^2$.

We take expectation over nodes' participation

$$\mathbb{E}_{\xi^{(h(\tau))}|\mathcal{B}^{(\tau)},\dots,\mathcal{B}^{(h(\tau))},\mathcal{H}^{(h(\tau))}}\left\|\bar{\boldsymbol{w}}^{(\tau+1)} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 \leq \left\|\bar{\boldsymbol{v}}^{(\tau+1)} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 + \mathbb{E}_{\xi^{(h(\tau))}|\mathcal{B}^{(\tau)},\dots,\mathcal{B}^{(h(\tau))},\mathcal{H}^{(h(\tau))}}\left\|\bar{\boldsymbol{w}}^{(\tau+1)\dagger} - \bar{\boldsymbol{v}}^{(\tau+1)}\right\|^2 \tag{D.57}$$

$$\leq \left\|\bar{\boldsymbol{v}}^{(\tau+1)} - \boldsymbol{w}^\star_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 + 4\eta_\tau^2J^2\sum_{i=1}^{N}\sum_{e\in E_i}\alpha_{i,e}^2\frac{1-p_{i,e}}{p_{i,e}}G_{i,e}, \tag{D.58}$$

where the equality derives from Lemma D.2 and the inequality from Lemma D.3. We take then expectation over the random batches

$$\mathbb{E}_{\mathcal{B}^{(\tau)},\dots,\mathcal{B}^{(h(\tau))},\xi^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left\|\bar{\boldsymbol{w}}^{(\tau+1)} - \boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\|^2$$

$$\leq \mathbb{E}_{\mathcal{B}^{(\tau)},\dots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left\|\bar{\boldsymbol{v}}^{(\tau+1)} - \boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 + 4\eta_\tau^2 J^2 \sum_{i=1}^{N} \sum_{e\in E_i} \alpha_{i,e}^2 \frac{1-p_{i,e}}{p_{i,e}} G_{i,e} \quad \text{(D.59)}$$

$$\leq (1-\eta_\tau\mu)\mathbb{E}_{\mathcal{B}^{(\tau-1)},\dots,\mathcal{B}^{(h(\tau))}|\mathcal{H}^{(h(\tau))}} \left\|\bar{\boldsymbol{w}}^{(\tau)} - \boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\| + \eta_\tau^2 \left(\sum_{k\in\mathcal{K}} \alpha_k^2\sigma_k^2 + 6L\Gamma + 8(J-1)^2G^2\right)$$

$$+ 4\eta_\tau^2 J^2 \sum_{i=1}^{N} \sum_{e\in E_i} \alpha_{i,e}^2 \frac{1-p_{i,e}}{p_{i,e}} G_{i,e}, \quad \text{(D.60)}$$

where the last inequality follows from Lemma D.1 observing that if $\tau+1 \in \mathcal{I}^{(J)}$, then $h(\tau) = \tau+1-J$.

Finally, we take total expectation

$$\mathbb{E}\left\|\bar{\boldsymbol{w}}^{(\tau+1)} - \boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\|^2 \leq (1-\eta_\tau\mu)\mathbb{E}\left\|\bar{\boldsymbol{w}}^{(\tau)} - \boldsymbol{w}^{\star}_{\tilde{\boldsymbol{\Lambda}}}\right\| + \eta_\tau^2 \left(\sum_{k\in\mathcal{K}} \alpha_k^2\sigma_k^2 + 6L\Gamma + 8(J-1)^2G^2 + 4J^2 \sum_{i=1}^{N} \sum_{e\in E_i} \alpha_{i,e}^2 \frac{1-p_{i,e}}{p_{i,e}} G_{i,e}\right). \quad \text{(D.61)}$$

This leads to a recurrence relation of the form $\Delta^{(\tau+1)} \leq (1-\eta_\tau\mu)\Delta^{(\tau)} + \eta_\tau^2 B$, and the result is obtained following the same steps in the proof of (X. Li et al., 2020, Thm. 1).

□

# E Gradient Variance Analysis

As discussed in Section 5.3.4, we observed empirical evidence showing that gradient variance is significantly higher at the initial exits compared to the later ones, making the optimization error especially sensitive to the stochastic gradients produced at these early stages. We conducted the following experiment: (1) Instantiate an Early Exit Network, e.g., a ResNet-18 with early exits after the 2nd and 5th residual blocks for CIFAR10 and after the 5th and 7th residual blocks for CIFAR100; (2) Iterate over the training data in mini-batches and calculate the gradient of the loss w.r.t. the weights at each exit; (3) Calculate the point-wise mean of each gradient over the mini-batches for each exit; (4) Take the mean of the gradient variance mean's to get a single value representing the average point-wise gradient variance per exit. We present below the empirical values from conducting this experiment:

Table D.1: Average Point-wise gradient variances per-exit.

| Dataset | Exit 1 | Exit 2 | Exit 3 |
|---|---|---|---|
| CIFAR10 | 0.00374 | 0.00224 | 0.00101 |
| CIFAR100 | 0.00216 | 0.00126 | 0.00101 |

# F Training Details

**Datasets.** We use the CIFAR10 and CIFAR100 datasets, which are commonly used to benchmark FL algorithms and early exit networks (Horvath et al., 2021; Diao et al., 2020; H. Li et al., 2019; Hu et al., 2019; Kaya et al., 2019; Ilhan et al., 2023). CIFAR10 and CIFAR100 each contain 60,000 total images composed of 32 x 32 colored pixels, with 10 and 100 classes, respectively. In our experiments, we use 45,000 images for training data, 5,000 images for validation data, and 10,000 images for test data.

**Model Architecture and Hyperparameters.** We conduct our experiments using a ResNet-18 model architecture (He et al., 2016), which has been widely used to study early exit networks and device heterogeneity in FL (Horvath et al., 2021; Diao et al., 2020; H. Li et al., 2019; Hu et al., 2019; Kaya et al., 2019; Ilhan et al., 2023). We insert early exits after the 2nd and 5th residual blocks for CIFAR10 and after the 5th and 7th residual blocks for CIFAR100. The training takes place for 100 outer epochs and the number of local epochs per node is scaled such that each node does the same number of gradient updates. We use mini-batch SGD with a starting learning rate of 0.1 and a cosine annealing schedule, a batch size of 128, weight decay of $5 \times 10^{-4}$, and momentum of 0.9. These hyperparameter values were selected based on empirically observing convergence during training for several basic CIS configurations, e.g., equal data partition and 33-33-33 serving rate setting. The same values are used for all experiments, i.e., all training data partitions, CIS serving rate setting, and training strategy configurations. All presented results are the mean value over three random seeds: 9, 42, and 67.

**Training Infrastructure.** We conducted our experiments on a computing node equipped with 3 x Nvidia A40 PCIe GPUs, each providing 10,752 CUDA cores, 336 tensor cores, and 48 GB of RAM. The node is powered by 2 x AMD EPYC 7282 processors running at 2.8 GHz, with 256 GB of system RAM. The operating system used was a Linux-based environment (e.g., Ubuntu 20.04), and the experiments were implemented using Python 3.8, CUDA 11.4, and cuDNN 8.2.

# G Additional Experiments

Table D.2: Experimental results for a CIS with 17 nodes (12 in the first layer, 4 in the second, and 1 in the third) for several CIS serving rates on the CIFAR10 dataset using an equal data partition across the network layers. All reported accuracy values are the mean value over three independent random seeds. The performance of the strategies for each serving rate setting follows the exact same order as in Table 1, indicating that our experimental setup with seven nodes is adequate for capturing CIS dynamics observed at larger scales.

| | CIS Serving Rate Setting | |
|---|---|---|
| Strategy | 60-30-10 | 10-30-60 |
| Equal Weight | 58.9 ± 3.9 | 83.5 ± 0.6 |
| FLOPS Prop | 44.6 ± 1.5 | 82.4 ± 0.5 |
| Serving Rate (ours) | **62.1** ± 1.7 | 84.3 ± 1.1 |
| Balanced Adj (ours) | 60.2 ± 3.1 | **84.7** ± 1.0 |

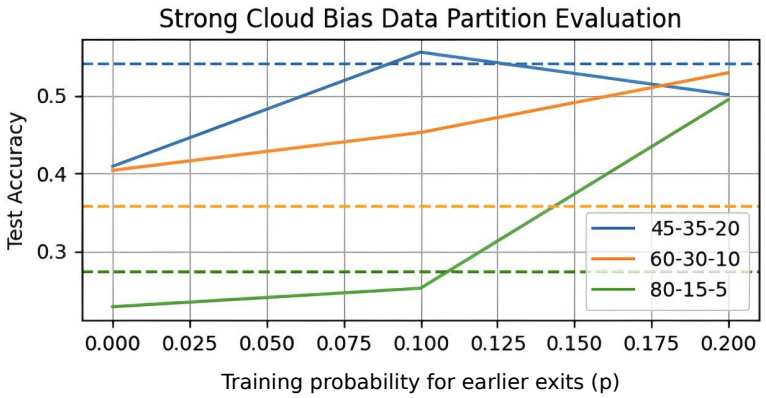



Figure D.1: Evaluating the impact of $p$ on the test accuracy for the strong cloud bias training data partition using the "Serving Rate ($p = k$)" strategy. The dashed lines represent the "Equal Weight" strategy's test accuracy for each serving rate setting. We remind the reader that $p$ stands for the probability that a given client will train each of its smaller exits.

# Hétérogénéité des Clients dans les Systèmes d'Apprentissage Fédérés

Angelo RODIO

**Abstract**

Federated Learning (FL) is a collaborative framework where clients—typically smartphones and IoT devices—train a machine learning model under the orchestration of a central server without sharing their datasets. Client heterogeneity in FL systems stems from statistical heterogeneity of local datasets, different device capabilities in hardware specifications (CPU power, memory capacity), network connectivity types (e.g., 5G and WiFi), power availability (battery levels), and it is outside server control. This thesis tackles the challenges of client heterogeneity in FL systems, their impact on the convergence of FL algorithms, and presents practical solutions for enhanced system efficiency and resource use. The first contribution addresses the problem of heterogeneous client participation: clients partake in the model training only occasionally and with varying frequencies. Three primary challenges arise. First, the "more participating" clients may bias the global model due to statistical heterogeneity in the clients' datasets. Second, addressing this bias by overcompensating for "less participating" clients introduces a larger variance in the learning process. Third, client participation can be correlated, due to the clients' correlated participation dynamics across time and geographic distributions. We characterize the bias-variance trade-off resulting from heterogeneous client participation and analyze the convergence of FL algorithms, assuming that client participation follows a Markov process. Our correlation-aware FL algorithm, `CA-Fed`, is the first heuristic to minimize this bias-variance-correlation trade-off and thus achieve faster convergence. The second contribution addresses the large variability in the learning process introduced by heterogeneous client participation. Variance reduction methods that leverage stale model updates for non-participating clients only consider homogeneous client participation. When participation is heterogeneous, the server must aggregate client updates with varying staleness—a challenge that remained unexplored. We analyze the convergence of these algorithms under heterogeneous participation, examining the advantages and disadvantages of leveraging stale updates in such heterogeneous environments. Our Staleness-Aware FL algorithm, `FedStale`, opportunely aggregates fresh and stale updates and performs well across many heterogeneous settings. The third contribution tackles heterogeneity in network resources: clients experience lossy communication channels with diverse characteristics (e.g., path loss, interference), which degrade FL algorithms' performance. Targeting high transmission reliability in FL is suboptimal, and loss mitigation strategies (e.g., retransmissions) demand more resources and longer training durations. We investigate algorithmic approaches for handling losses during training and present a packet loss-aware FL algorithm, `UPGA-PL`, with comparable performance to ideal lossless channels at the cost of a few additional communication rounds. The last contribution investigates heterogeneity in hardware environments: clients with diverse computing capabilities (e.g., end-devices, edge servers, and cloud infrastructures) may cooperate to learn a common model; yet, client heterogeneity makes uniform model deployment infeasible at inference time. Cooperative Inference Systems (CISs) enable less-performing devices to offload parts of their inference tasks to more powerful devices with larger models within the network; however, FL training overlooks how these models will be used at inference time. Our inference-aware FL algorithm, `Fed-CIS`, is the first to consider the future inference request load for each sub-model at training time. It also enables computationally stronger clients to help train models for the weaker ones. The concluding remarks reflect on the open challenges encountered throughout this thesis and outline prospective research directions for future work.

**Keywords:** Federated Learning, Distributed Optimization, Markov chain, Variance Reduction