FedStale: leveraging Stale Updates in Federated Learning

Angelo Rodio, Giovanni Neglia

FedStale: leveraging Stale Updates in Federated Learning

Angelo Rodio[a] **and Giovanni Neglia**[a]

[a]Centre Inria d'Université Côte d'Azur, France. Email: {firstname.lastname}@inria.fr

Abstract. Federated learning algorithms, such as FedAvg, are negatively affected by data heterogeneity and partial client participation. To mitigate the latter problem, global variance reduction methods, like FedVARP, leverage stale model updates for non-participating clients. These methods are effective under homogeneous client participation. Yet, this paper shows that, when some clients participate much less than others, aggregating updates with different levels of staleness can detrimentally affect the training process. Motivated by this observation, we introduce FedStale, a novel algorithm that updates the global model in each round through a convex combination of "fresh" updates from participating clients and "stale" updates from non-participating ones. By adjusting the weight in the convex combination, FedStale interpolates between FedAvg, which only uses fresh updates, and FedVARP, which treats fresh and stale updates equally. Our analysis of FedStale convergence yields novel findings: *i)* it integrates and extends previous FedAvg and FedVARP analyses to heterogeneous client participation; *ii)* it underscores how the least participating client influences convergence error; *iii)* it provides practical guidelines to best exploit stale updates, showing that their usefulness diminishes as data heterogeneity decreases and participation heterogeneity increases. Extensive experiments featuring diverse levels of client data and participation heterogeneity not only confirm these findings but also show that FedStale outperforms both FedAvg and FedVARP in many settings.[1]

1 Introduction

Edge devices generate critical data for training machine learning models. However, centralizing this data is often impractical due to substantial communication overhead or simply impossible due to privacy regulations. Federated Learning (FL) [22, 16] offers a solution. In this paradigm, edge devices—also referred to as clients—collaborate to train a shared machine learning model. This collaboration, coordinated by a central server, maintains data decentralized, effectively addressing privacy and communication challenges.

In Federated Averaging (FedAvg) [22] and similar FL algorithms [19, 25, 1, 15], clients perform multiple stochastic gradient descent (SGD) steps on their local datasets and then transmit their updated models to the central server. The server aggregates these client models to form a new global model, which is subsequently disseminated to the clients for further iterations.

The *multiple* local updates performed by each client are crucial for enhancing communication efficiency. However, these updates can negatively impact the training process, as local client models progressively diverge towards client-specific local minimizers due to *data heterogeneity* [20, 15].

Another significant source of heterogeneity stems from varying levels of client participation in the training process. This *participation heterogeneity* is driven by factors beyond server control [2, 35, 40], such as diverse hardware specifications (CPU power, memory), network connectivity (WiFi, 5G), and power availability (e.g., clients may only participate when charging to prevent battery drain) [33, 14, 21]. Despite this, much of the prior research assumes partial yet homogeneous client participation [20, 15, 19, 39, 9, 4, 27, 5], overlooking the impact of such heterogeneity on the convergence of FedAvg-like algorithms. We identify and illustrate two main problems caused by the heterogeneous client participation.

First, heterogeneous participation risks biasing the global model in favor of clients that participate more frequently. Intuitively, when some clients participate more often than others, the global model may disproportionately reflect the local objectives of these more participating clients, thereby disadvantaging those who participate less. To counteract this bias, recent studies [36, 37] propose an unbiased version of FedAvg, which scales clients' model updates inversely with their participation frequency. By assigning greater weight to less participating clients, this approach ensures that the global model fairly represents all clients.

Second, even if the potential bias is mitigated, partial and heterogeneous client participation still exacerbates the variability of the learning process. The unbiased scaling amplifies variations in the magnitude of client updates, leading to increased variance in the learned model and slower convergence. Although a few recent works focus on global variance reduction [11, 40, 12, 38], they are limited to scenarios involving homogeneous client participation. Specifically, FedVARP (Federated VAriance Reduction for Partial client participation) [12] leverages the most recent, albeit potentially stale, model updates in place of unavailable updates from non-participating clients. FedVARP has demonstrated, both theoretically and empirically, its capability to effectively lower variance and consistently outperform FedAvg in settings with partial yet homogeneous client participation. It is anticipated to perform similarly well even in heterogeneous settings [12]. 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 paper.

This paper 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

[1] The full paper, including supplementary material, is available [28].

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 β. `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 β 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 paper is organized as follows. Section 2 reviews the problem and related work. Section 3 introduces `FedStale`, our staleness-aware algorithm, through a motivating example. Section 4 provides a convergence analysis of `FedStale` under heterogeneous client participation. `FedStale` is extensively evaluated in Section 5, and Section 6 concludes the paper. Proof outlines are included in the Appendix, while detailed proofs are available in the supplementary material [28].

2 Problem Definition and Background

We consider a FL setting where N clients, each client i equipped with a dataset $\mathcal{D}_i$ consisting of n_i samples, collaboratively learn the parameters $\boldsymbol{w} \in \mathbb{R}^d$ of a global ML model (e.g., the weights of a neural network). Orchestrated by a central server, these clients cooperate to minimize the *global* objective:[2]

$$\min_{\boldsymbol{w}\in\mathbb{R}^d} F(\boldsymbol{w}) \triangleq \frac{1}{N}\sum_{i=1}^{N}\left[F_i(\boldsymbol{w}) \triangleq \frac{1}{n_i}\sum_{\xi_i\in\mathcal{D}_i} f(\boldsymbol{w},\xi_i)\right], \quad (1)$$

where client i has *local* objective $F_i(\boldsymbol{w})$ and $f(\boldsymbol{w},\xi_i)$ is the loss function evaluating model performance on data sample $\xi_i \in \mathcal{D}_i$.

In this paper, we consider algorithms obeying the general operation in Algorithm 1, differing in the `ComputeUpdate()` procedure.

`FedAvg` iteratively solves Problem (1) while maintaining data decentralization. Model training involves T rounds of communication between server and clients: at the beginning of each round $t > 0$, the server sends the current global model, $\boldsymbol{w}^{(t)}$, to a random subset of participating clients $\mathcal{S}^{(t)}$, 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:

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

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:

$$\Delta^{(t)}_{\texttt{FedAvg}} = \frac{1}{|\mathcal{S}^{(t)}|}\sum_{i\in\mathcal{S}^{(t)}} \Delta_i^{(t)}, \quad (2)$$

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 $\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} - \eta_s \Delta^{(t)}_{\texttt{FedAvg}}$.

[2] Objective (1) corresponds to a "per-client fairness" criterion. Another common choice is to weight each local objective proportionally to the client's number of samples (per-sample fairness). We consider objective (1) for the sake of concreteness, but the analysis in this paper can be immediately extended to any weighted sum of local objectives.

Algorithm 1: FL algorithm with pluggable global update

Input: $\boldsymbol{w}^{(1)}$, K, η_s, η_c ; **Output:** $\{\boldsymbol{w}^{(t)} : \forall t\}$
for $t = 1,\dots,T$ **do**
  for $i \in \mathcal{S}^{(t)}$, *in parallel* **do**
    $\boldsymbol{w}_i^{(t,0)} \leftarrow \boldsymbol{w}^{(t)}$
    for $k = 0, 1 \dots, K-1$ **do**
      $\boldsymbol{w}_i^{(t,k+1)} \leftarrow \boldsymbol{w}_i^{(t,k)} - \eta_c \nabla F_i(\boldsymbol{w}_i^{(t,k)}, \xi_i^{(t,k)})$
    $\Delta_i^{(t)} \leftarrow (\boldsymbol{w}^{(t)} - \boldsymbol{w}_i^{(t,K)})$
  $\Delta^{(t)} \leftarrow \texttt{ComputeUpdate}(\{\Delta_i^{(t)}\}_{i\in\mathcal{S}^{(t)}}, \dots)$
  $\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} - \eta_s \Delta^{(t)}$

Following standard assumptions [36, 29, 37], we model client participation heterogeneity through the *participation probability* p_i:

$$p_i \triangleq \mathop{\mathbb{E}}_{\mathcal{S}^{(t)}}\left[\mathbb{P}(i \in \mathcal{S}^{(t)})\right]. \quad (3)$$

When client participation is *homogeneous* ($p_i = p, \forall i$), $\mathbb{E}_{\mathcal{S}^{(t)}}[\Delta^{(t)}_{\texttt{FedAvg}}] = \frac{1}{N}\sum_{i=1}^N \Delta_i^{(t)}$. Under this condition, Eq. (2) is then an unbiased estimator of the model update as if *all* clients were to participate [20, 9]. This ensures that the final model fairly represents all clients.

Conversely, under *heterogeneous* participation, where probabilities $\{p_i\}$ vary among clients, Eq. (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:

$$\widetilde{F}(\boldsymbol{w}) = \frac{1}{N}\sum_{i=1}^{N}\frac{p_i}{\sum_{j=1}^N p_j} F_i(\boldsymbol{w}), \quad (4)$$

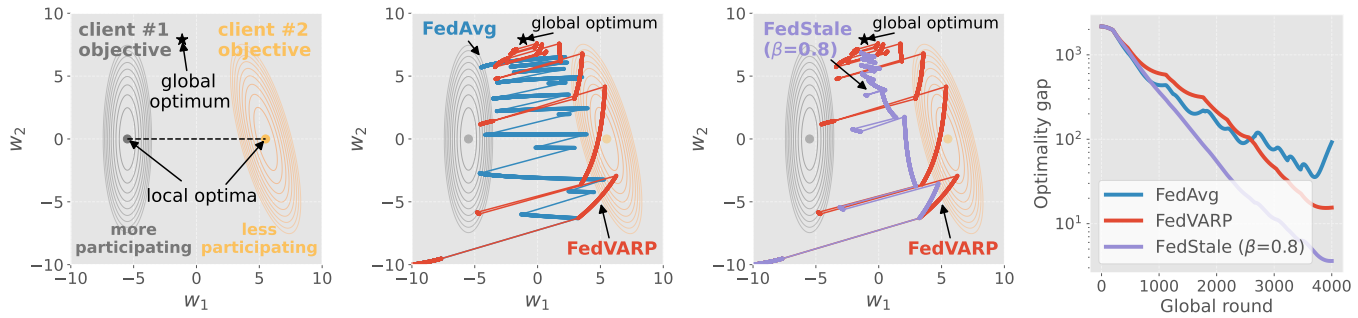
which may arbitrarily deviate from the global objective (1).

To effectively minimize objective (1) when client participation is heterogeneous, recent works [9, 36, 10, 29, 37] have discussed the need to debias $\Delta^{(t)}_{\texttt{FedAvg}}$. Specifically, Eq. (2) has been modified into Eq. (5), resulting in an unbiased version of `FedAvg`, denoted here as `U-FedAvg` [36, 29, 37]:

$$\Delta^{(t)}_{\texttt{U-FedAvg}} = \frac{1}{N}\sum_{i\in\mathcal{S}^{(t)}}\frac{\Delta_i^{(t)}}{p_i}. \quad (5)$$

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 [29, 37].

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 [11, 40, 12, 38]. 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



(a) Optimization objectives. (b) Trajectories: FedAvg, FedVARP. (c) Trajectories: FedVARP, FedStale. (d) Convergence curves.

Figure 1: Comparison of FedAvg, FedVARP, and FedStale in a two-clients, 2D quadratic setting with *heterogeneous* client participation. **Fig. 1a:** Contour plots of client objectives, their local optima, and global optimum. Client participation ratio is $p_1/p_2 = 100$. **Fig. 1b:** Trajectories by FedAvg and FedVARP over T=4000 rounds with K=5 local iterations. While both algorithms target the global optimum, FedAvg struggles with large variance and FedVARP follows suboptimal paths due to stale updates. **Fig. 1c:** FedStale (β=0.8) follow a more stable trajectory under heterogeneous client participation. **Fig. 1d:** Learning curves of FedAvg, FedVARP, and FedStale over 10 runs. With a lower weight on stale updates (β=0.8), FedStale converges faster to the global optimum.

straightforward, similar to what was done for `FedAvg` in Eq. (5). We select `FedVARP` [12] 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:

$$\boldsymbol{h}_i^{(t)} = \begin{cases} \Delta_i^{(t-1)} & \text{if } i \in \mathcal{S}^{(t-1)} \\ \boldsymbol{h}_i^{(t-1)} & \text{otherwise} \end{cases}, \tag{6}$$

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

$$\Delta_{\texttt{U-FedVARP}}^{(t)} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{h}_i^{(t)} + \frac{1}{N}\sum_{i\in\mathcal{S}^{(t)}} \frac{\Delta_i^{(t)} - \boldsymbol{h}_i^{(t)}}{p_i}. \tag{7}$$

Unlike `U-FedAvg`, which essentially ignores non-participating clients, `U-FedVARP` leverages their last updates, albeit stale, when they do not participate in training. When they participate again, `U-FedVARP` subtracts these stale updates to eliminate any inconsistency caused by leveraging stale information, and then applies the fresh update. Both corrections are reweighed by p_i^{-1}, similarly to `U-FedAvg`, ensuring that $\mathbb{E}\left[\Delta_{\texttt{U-FedVARP}}^{(t)}\right] = \frac{1}{N}\sum_{i=1}^{N}\Delta_i^{(t)}$, making `U-FedVARP`'s aggregation (7) *unbiased*. Moreover, by leveraging stale updates for non-participating clients, `U-FedVARP` acts as a `SAGA`-like [6] 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 the server must allocate.

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 [12, 37], theoretical support for this belief has been provided only for homogeneous participation scenarios [12, Theorem 2] and empirical results do not lead to definitive conclusions [37, Table 5].

This paper 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 paper, we focus on the unbiased versions of the two algorithms. However, for brevity, we refer to them simply as `FedVARP` and `FedAvg`.

3 The FedStale Algorithm

We challenge `FedVARP`'s expected superiority under client participation heterogeneity through the following illustrative example.

3.1 A motivating example

Figure 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 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. (5) and (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 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.

3.2 A convex combination of fresh and stale updates

In Figs. 1c and 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`)

Algorithm 2: Global update computation in FedStale

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

and outdated trajectories (FedVARP). Motivated by these observations, we propose FedStale (Federated Averaging with Stale Updates), outlined in Algorithm 2. In each round, FedStale updates the global model through a convex combination of fresh and stale updates, with parameter β in the range $[0, 1]$:

$$\Delta_{\texttt{FedStale}}^{(t)} = (1-\beta)\Delta_{\texttt{FedAvg}}^{(t)} + \beta \Delta_{\texttt{FedVARP}}^{(t)} \tag{8}$$

$$= \frac{1}{N}\sum_{i=1}^{N} \beta \boldsymbol{h}_i^{(t)} + \frac{1}{N} \sum_{i \in \mathcal{S}^{(t)}} \frac{\Delta_i^{(t)} - \beta \boldsymbol{h}_i^{(t)}}{p_i}. \tag{9}$$

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 β, 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 β as the only additional requirement. Section 5 shows that a coarse adjustment of β (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 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 [38], MIFA [11], AFA-CD and AFA-CS [40], 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 [32] global variance reduction step, *biases* the global model leading to objective inconsistency.

SVRG-based Variance Reduction Methods [13, 18, 24, 8] 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 [6] 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 [15] uses control variates to correct for data heterogeneity errors. Adapting this method to handle participation heterogeneity would require clients to perform local SAGA-like [6] 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.

4 Convergence Analysis

Assumption 1 (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 2 (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 3 (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 4 (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 1–3 are standard in federated learning convergence analysis [39, 36, 5]. The terms σ^2 and σ_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 [36, 37]. Exploring more complex participation dynamics, following the methodologies in [36, 29], remains a task for future research.

We first provide an upper bound for FedStale's convergence. To focus the discussion on our main results, we defer proof outlines to the appendix and detailed proofs to the supplementary material [28].

Theorem 1 (Convergence of FedStale, upper bound)**.** *Under Assumptions 1–4, if the client and server learning rates, η_c and η_s, are chosen such that $\eta_c \leq \frac{1}{8LK}$ and $\eta_s \leq \min\left\{\frac{N p_{var}}{12(1-\beta)^2}, \frac{p_{var} p_{min}}{3\beta^2 p_{avg}}\right\}$, the sequence of FedStale iterates satisfies*

$$\min_{t \in \{1,T\}} \mathbb{E}\left\|\nabla F(\boldsymbol{w}_{\texttt{FedStale}}^{(t)})\right\|^2 \leq \underbrace{\mathcal{O}\left(\frac{F(\boldsymbol{w}^{(1)}) - F^*}{\eta_s \eta_c K T}\right)}_{\text{iterate initialization error}} \tag{10}$$

$$+ \underbrace{\mathcal{O}\left(\frac{\beta^2 \eta_s \eta_c L K H^{(1)}}{p_{var} p_{min} T}\right)}_{\text{memory initialization error}} + \underbrace{\mathcal{O}\left(\left[\frac{1}{N} + \beta^2 \frac{p_{avg}}{p_{min}}\right] \frac{\eta_s \eta_c L \sigma^2}{p_{var}}\right)}_{\text{stochastic gradient error}}$$

$$+ \underbrace{\mathcal{O}\left(\left[\frac{(1-\beta)^2}{N} + \beta^2 \eta_c^2 L^2 K (K-1) \frac{p_{avg}}{p_{min}}\right] \frac{\eta_s \eta_c L K \sigma_g^2}{p_{var}}\right)}_{\text{error from data heterogeneity}},$$

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 1 relates FedStale's convergence to the iterate and memory initial errors, and variances from stochastic gradients (σ^2) and data heterogeneity (σ_g^2). It also quantifies the impact of client

participation heterogeneity through the terms p_{var}, p_{avg}, and $p_{\min}$. By scaling the client learning rate as $\mathcal{O}(\frac{1}{\sqrt{T}})$, all error components asymptotically vanish, proving the *unbiasedness* of update (9).

Theorem 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 2 (Convergence of FedAvg, upper bound)**.** *Under same assumptions as Theorem 1, the sequence of* FedAvg *iterates satisfies*

$$\min_{t\in\{1,T\}} \mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)}_{FedAvg})\right\|^2 \leq \tag{11}$$
$$\underbrace{\mathcal{O}\left(\frac{F(\boldsymbol{w}^{(1)})-F^*}{\eta_s\eta_c KT}\right)}_{\text{iterate initialization error}} + \underbrace{\mathcal{O}\left(\frac{\eta_s\eta_c L\sigma^2}{Np_{var}}\right)}_{\text{stochastic gradient error}} + \underbrace{\mathcal{O}\left(\frac{\eta_s\eta_c LK\sigma_g^2}{Np_{var}}\right)}_{\text{error from data heterogeneity}}.$$

Corollary 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. (11) that slows down FedAvg convergence. This justifies our observations for FedAvg in Figure 1b.

Second, for $\beta = 1$, Theorem 1 extends FedVARP known convergence results [12, Theorem 2] to heterogeneous client participation.

Corollary 3 (Convergence of FedVARP, upper bound)**.** *Under the same assumptions as in Theorem 1,* FedVARP*'s iterates satisfy*

$$\min_{t\in\{1,T\}} \mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)}_{FedVARP})\right\|^2 \leq \underbrace{\mathcal{O}\left(\frac{F(\boldsymbol{w}^{(1)})-F^*}{\eta_s\eta_c T} + \frac{\eta_s\eta_c H^{(1)}}{p_{var}p_{min}T}\right)}_{\text{iterate and memory initialization errors}}$$
$$+ \underbrace{\mathcal{O}\left(\frac{\eta_s\eta_c Lp_{avg}\sigma^2}{p_{var}p_{min}}\right)}_{\text{stochastic gradient error}} + \underbrace{\mathcal{O}\left(\frac{\eta_s\eta_c^3L^3K^2(K-1)p_{avg}\sigma_g^2}{p_{var}p_{min}}\right)}_{\text{error from data heterogeneity}}. \tag{12}$$

We highlight two differences with respect to FedAvg. First, FedVARP mitigates *data heterogeneity error*: by scaling the learning rate η_c as $\mathcal{O}(T^{-1/2})$, the term in σ_g^2 decreases as $\mathcal{O}(T^{-3/2})$ versus $\mathcal{O}(T^{-1/2})$ for FedAvg in (11). However, FedVARP amplifies the stochastic gradient error through the ratio $p_{\text{avg}}/p_{\min}$, 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.

One may wonder whether the appearance of the factor $1/p_{\min}$ 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 4 (Convergence of FedStale, lower bound)**.** *Under Assumption 1, 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*

$$\min_{t\in\{1,T\}} \mathbb{E}\left\|\nabla F(\boldsymbol{w}^{(t)}_{FedStale})\right\|^2 \geq \Omega\left(\frac{F(\boldsymbol{w}^{(1)})-F^*}{p_{min}^3T^3+1}\right). \tag{13}$$

Theorem 4 proves that FedStale, for any $\beta > 0$, and FedVARP require at least $T \geq \Omega(1/p_{\min})$ rounds to minimize objective (1).

4.1 Finding the optimal weight β^*

FedStale leverages the parameter β to balance the multiple sources of variance in Theorem 1: stochastic gradients (σ^2), data heterogeneity (σ_g^2), and client participation heterogeneity (through the ratio $p_{\text{avg}}/p_{\min}$).

The quadratic dependency on β of the bound in Theorem 1, Eq. (10), guarantees a unique minimizer $\beta^* \in [0,1]$, generally different from the boundaries values of 0 and 1. The optimal β^* is:

$$\beta^* = \frac{\sigma_g^2/N}{a_1\frac{p_{\text{avg}}}{p_{\min}}\frac{\sigma^2}{K} + \left[\frac{1}{N} + a_2\frac{p_{\text{avg}}}{p_{\min}}\eta_c^2L^2K(K-1)\right]\sigma_g^2}, \tag{14}$$

where a_1 and a_2 are positive constants.

In practice, computing β^* is challenging due to the unknowns L, σ^2, and σ_g^2 in Eqs. (10) and (14), which are difficult to estimate since they depend on the client objectives and on the specific heterogeneity setting. Moreover, Eq. (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. (14) to derive *qualitative*, yet important guidelines.

The monotonically increasing behavior of β^* with σ_g^2 in Eq. (14) suggests *Guideline A: Increase the weight to stale updates, β, when data heterogeneity, σ_g^2, increases.*

Guideline A is in line with our previous comparison of Corollary 2 and Corollary 3. As we observed, stale updates become more beneficial when data heterogeneity (σ_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 β^* with $p_{\text{avg}}/p_{\min}$ in Eq. (14) leads to *Guideline B: Decrease the weight to stale updates, β, 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.

5 Experimental Results

We evaluate the performance of FedStale through experiments. The source code for our experimental framework is publicly accessible at https://github.com/arodio/FedStale.

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 [7] and natural image classification on CIFAR-10 [17]. 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 (1).

Participation heterogeneity. Client participation follows a Bernoulli distribution, in line with Assumption 4. To simulate

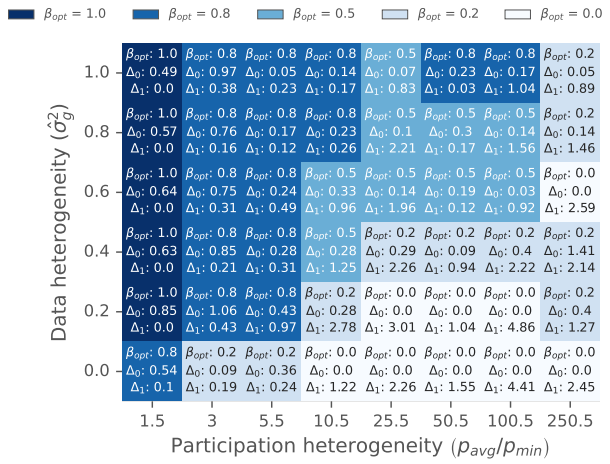


Figure 2: β_{opt} values for FedAvg (β=0), FedVARP (β=1), and FedStale ($\beta \in \{0.2, 0.5, 0.8\}$) across 48 heterogeneity settings on the MNIST dataset. Color gradients range from lighter shades (β_{opt}=0) to darker shades (β_{opt}=1).

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 [30], 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 σ_g^2 in Assumption 3, 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 [12] showed that, under partial client participation, `FedVARP` consistently outperformed both `MIFA` [11], due to its biased variance correction, and `SCAFFOLD` [15], 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 η_s to 1 and tune the client learning rate η_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 5.3. We average results over three random seeds.

5.2 *Existence of different regimes*

In Figure 2, we show the empirical values of β 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 β_{opt}.

The heatmap shows how β_{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`. Δ_0 and Δ_1 denote, respectively, the accuracy improvements of `FedStale`(β_{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 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 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 ($p_{\text{avg}} \geq 25p_{\text{min}}$), 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 β_{opt} values ($\beta_{\text{opt}} \in \{0.2, 0.5, 0.8\}$), which yield the best performance.

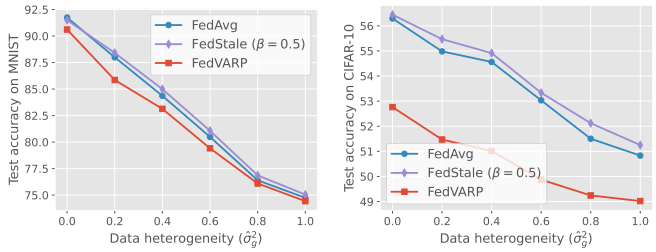
Overall, `FedStale` prevails in 72% of scenarios within our 6×8 grid, against `FedVARP`, 18%, and `FedAvg`, 10%. Therefore, `FedStale` plays a key role—we believe—in bridging the gaps posed by `FedAvg` and `FedVARP` in real-world federated settings, which often exhibit intermediate levels of client data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

Effect of data heterogeneity. Figure 2 shows that β_{opt} increases with data heterogeneity, in line with Guideline A. Figure 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 3a and 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 3b shows that `FedVARP`, despite its overall lower accuracy, proves to perform better in extremely heterogeneous data scenarios (when $\hat{\sigma}_g^2 \geq 0.8$).

Effect of participation heterogeneity. Figure 2 shows that β_{opt} decreases as the participation heterogeneity ($p_{\text{avg}}/p_{\text{min}}$) increases, in line with Guideline B. Figure 4 details this dynamic by fixing data heterogeneity at $\hat{\sigma}_g^2 = 0.6$ and only varying participation heterogeneity. In both Figures 4a and 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 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 $p_{\text{avg}} \approx 50p_{\text{min}}$.

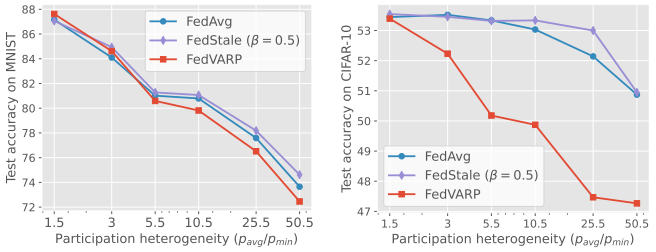
5.3 *Online estimation of participation probabilities*

We evaluate `FedStale` with online estimation of client participation probabilities, to simulate scenarios where these probabilities are unknown before training [26, 29, 37]. To this purpose, we integrate `FedStale` with `FedAU` [37], a state-of-the-art algorithm for tracking client participation dynamics, that balances bias and variance in



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

Figure 3: Test accuracy of FedAvg (β=0), FedVARP (β=1), and FedStale (β=0.5) varying data heterogeneity at fixed participation ratio $p_{\text{avg}}/p_{\text{min}} = 10$.



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

Figure 4: Test accuracy of FedAvg (β=0), FedVARP (β=1), and FedStale (β=0.5) varying client participation ratio at fixed data heterogeneity $\hat{\sigma}_g^2 = 0.6$.

the estimation through a cutoff mechanism.

Figure 5 shows that the integration of `FedStale` with `FedAU`'s estimation technique still aligns with our guidelines. Moreover, `FedVARP` performs significantly worse than `FedStale`(β_{opt}) when client participation probabilities are estimated (Δ_1 values in Fig. 5). Also, we observe overall lower β_{opt} values in this scenario. These trends suggest that methods leveraging stale updates, like `FedVARP`, might be particularly sensitive to inaccurate p_i estimations.

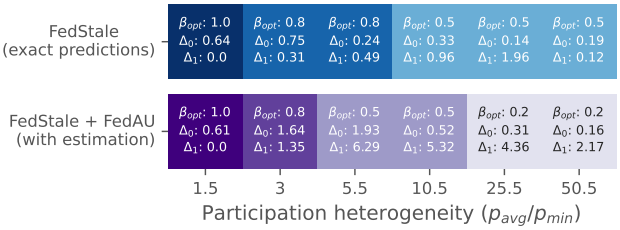


Figure 5: "Exact" vs. "Estimated" participation probabilities, $\hat{\sigma}_g^2 = 0.6$.

6 Conclusion

This paper 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.

A Appendix

A.1 Proof sketch, Theorem 1

Previous work analyzed convergence of FL algorithms in various settings. Closest to our setting are Wang et al. [34] and Jhunjhunwala et al. [12], that analyzed `FedAvg` and `FedVARP`, respectively, under non-iid data and partial yet *homogeneous* client participation.

Our analysis in Theorem 1 builds on [12] and relies on a similar Lyapunov optimization function as in [12, Appendix C.2, Eq. (20)]:

$$\psi^{(t)} \triangleq F(\boldsymbol{w}^{(t)}) + \delta \left\| \Delta^{(t)}_{\texttt{FedStale}} \right\|^2 + \gamma H^{(t)}, \quad \delta, \gamma > 0, \qquad (15)$$

where $H^{(t)} \triangleq \frac{1}{N} \sum_{i=1}^{N} \left\| \nabla F_i(\boldsymbol{w}^{(t)}) - \boldsymbol{h}_i^{(t)} \right\|^2$ quantifies the deviation of stale client updates from the "true" local gradients.

This section provides the proof outline for Theorem 1, focusing mostly on the novel contributions of our analysis:

- We apply the standard descent lemma [34], for smooth and non-convex objectives, to Eq. (15) [Supplementary [28], Lemma 1];
- Under Assumptions 1–4, the variance of the global update $\Delta^{(t)}_{\texttt{FedStale}}$ is bounded by variances from stochastic gradients (σ^2) and data heterogeneity (σ_g^2), by the square norm of the previous global update $\Delta^{(t-1)}_{\texttt{FedStale}}$, and by the memory error $H^{(t)}$. The last two terms, emerging from stale updates for non-participating clients, are multiplied by β^2 and contribute to both optimization and error. Additionally, the client participation variance $1/p_{\text{var}}$, consequence of the Bernoulli assumption on the client participation (Assumption 4), equally impacts all these terms. More details are provided in the Supplementary [28], Lemmas 6 and 8;
- Under Assumptions 1–4, the error from stale updates ($H^{(t)}$) is also bounded by the variances σ^2 and σ_g^2, the square norm of the previous global update $\Delta^{(t-1)}_{\texttt{FedStale}}$, and the previous memory error $H^{(t-1)}$. This error, under Assumption 4, depends on the participation probability of the least participating client (p_{min}), is consistently weighted by β^2, and does not affect `FedAvg`. More details in the Supplementary [28], Lemmas 7 and 9;
- Through the Lyapunov recursion, the dependency on p_{min} remains consistent across all β^2-weighted terms. This suggests that the influence of p_{min} stems from stale updates and can be balanced by controlling β. More details in Supplementary [28], Theorem 1.

A.2 Proof sketch, Theorem 4

Our proof builds upon Nesterov [23] and Bubeck [3], who established lower bounds in *centralized* settings, and Scaman et al. [31], for general *decentralized* setting. We adapt the analysis to non-convex federated settings with heterogeneous client participation:

- We split Nesterov's function for *centralized* optimization [23, 3] between the most and least participating clients (p_{max} and p_{min});
- Most dimensions of the parameters $\boldsymbol{w}^{(t)}_{\texttt{FedStale}}$ remains zero, and (fresh or stale) client updates only increase non-zero dimensions once every $1/p_{\text{min}}$ steps on average;
- We standardize the lower bound measure to squared gradient norms for direct comparison with non-convex counterparts (Theorem 1), in expectation over the randomness in client participation.

Acknowledgements

This research was supported in part by the French government through the 3IA Côte d'Azur Investments in the Future project, managed by the National Research Agency (ANR) under reference number ANR-19-P3IA-0002, in part by the European Network of Excellence dAIEDGE under Grant Agreement Nr. 101120726, and in part by Groupe La Poste, sponsor of the Inria Foundation, in the framework of the FedMalin Inria Challenge.

References

[1] D. A. E. Acar, Y. Zhao, R. Matas, M. Mattina, P. Whatmough, and V. Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2021.

[2] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečný, S. Mazzocchi, B. McMahan, T. Van Overveldt, D. Petrou, D. Ramage, and J. Roselander. Towards Federated Learning at Scale: System Design. *Proceedings of Machine Learning and Systems*, 1:374–388, Apr. 2019.

[3] S. Bubeck. Convex Optimization: Algorithms and Complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, Nov. 2015. ISSN 1935-8237, 1935-8245. doi: 10.1561/2200000050.

[4] W. Chen, S. Horváth, and P. Richtárik. Optimal Client Sampling for Federated Learning. *Transactions on Machine Learning Research*, Aug. 2022. ISSN 2835-8856.

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

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

[7] L. Deng. The MNIST Database of Handwritten Digit Images for Machine Learning Research. *IEEE Signal Processing Magazine*, 2012.

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

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

[10] Y. Fraboni, R. Vidal, L. Kameni, and M. Lorenzi. A General Theory for Federated Optimization with Asynchronous and Heterogeneous Clients Updates. *Journal of Machine Learning Research*, 24(110):1–43, 2023. ISSN 1533-7928.

[11] X. Gu, K. Huang, J. Zhang, and L. Huang. Fast Federated Learning in the Presence of Arbitrary Device Unavailability. In *Advances in Neural Information Processing Systems*, volume 34, pages 12052–12064. Curran Associates, Inc., 2021.

[12] D. Jhunjhunwala, P. Sharma, A. Nagarkatti, and G. Joshi. Fedvarp: Tackling the variance due to partial client participation in federated learning. In *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, pages 906–916. PMLR, Aug. 2022.

[13] R. Johnson and T. Zhang. Accelerating Stochastic Gradient Descent using Predictive Variance Reduction. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

[14] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D'Oliveira, H. Eichner, S. E. Rouayheb, D. Evans, J. Gardner, Z. Garrett, A. Gascón, B. Ghazi, P. B. Gibbons, M. Gruteser, Z. Harchaoui, C. He, L. He, Z. Huo, B. Hutchinson, J. Hsu, M. Jaggi, T. Javidi, G. Joshi, M. Khodak, J. Konecný, A. Korolova, F. Koushanfar, S. Koyejo, T. Lepoint, Y. Liu, P. Mittal, M. Mohri, R. Nock, A. Özgür, R. Pagh, H. Qi, D. Ramage, R. Raskar, M. Raykova, D. Song, W. Song, S. U. Stich, Z. Sun, A. T. Suresh, F. Tramèr, P. Vepakomma, J. Wang, L. Xiong, Z. Xu, Q. Yang, F. X. Yu, H. Yu, and S. Zhao. Advances and Open Problems in Federated Learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, June 2021. ISSN 1935-8237, 1935-8245. doi: 10.1561/2200000083.

[15] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. In *Proceedings of the 37th International Conference on Machine Learning*, pages 5132–5143. PMLR, Nov. 2020.

[16] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon. Federated Learning: Strategies for Improving Communication Efficiency, Oct. 2017.

[17] A. Krizhevsky and G. Hinton. Learning Multiple Layers of Features from Tiny Images. Technical report, University of Toronto, Toronto, 2009.

[18] L. Lei, C. Ju, J. Chen, and M. I. Jordan. Non-convex Finite-Sum Optimization Via SCSG Methods. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[19] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith. Federated Optimization in Heterogeneous Networks. *Proceedings of Machine Learning and Systems*, 2:429–450, Mar. 2020.

[20] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang. On the Convergence of FedAvg on Non-IID Data. In *International Conference on Learning Representations*, Sept. 2019.

[21] H. Ludwig and N. Baracaldo. *Federated Learning: A Comprehensive Overview of Methods and Applications*. Springer Cham, 2022. doi: https://doi.org/10.1007/978-3-030-96896-0.

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

[23] Y. Nesterov. *Introductory Lectures on Convex Optimization*, volume 87 of *Applied Optimization*. Springer US, Boston, MA, 2004. ISBN 978-1-4613-4691-3. doi: 10.1007/978-1-4419-8853-9.

[24] L. M. Nguyen, J. Liu, K. Scheinberg, and M. Takáč. SARAH: A Novel Method for Machine Learning Problems Using Stochastic Recursive Gradient. In *Proceedings of the 34th International Conference on Machine Learning*, pages 2613–2621. PMLR, July 2017.

[25] S. J. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečný, S. Kumar, and H. B. McMahan. Adaptive Federated Optimization. In *International Conference on Learning Representations*, Apr. 2023.

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

[27] E. Rizk, S. Vlaski, and A. H. Sayed. Federated Learning Under Importance Sampling. *IEEE Transactions on Signal Processing*, 70:5381–5396, 2022. ISSN 1941-0476. doi: 10.1109/TSP.2022.3210365.

[28] A. Rodio and G. Neglia. FedStale: leveraging Stale Client Updates in Federated Learning. *arXiv:2405.04171 [cs]*, May 2024.

[29] A. Rodio, F. Faticanti, O. Marfoq, G. Neglia, and E. Leonardi. Federated Learning Under Heterogeneous and Correlated Client Availability. *IEEE/ACM Transactions on Networking*, pages 1–10, 2023. doi: 10.1109/TNET.2023.3324257.

[30] F. Sattler, K.-R. Müller, and W. Samek. Clustered Federated Learning: Model-Agnostic Distributed Multitask Optimization Under Privacy Constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8):3710–3722, Aug. 2021. ISSN 2162-2388. doi: 10.1109/TNNLS.2020.3015958.

[31] K. Scaman, F. Bach, S. Bubeck, Y. T. Lee, and L. Massoulié. Optimal Convergence Rates for Convex Distributed Optimization in Networks. *Journal of Machine Learning Research*, 20(159):1–31, 2019. ISSN 1533-7928.

[32] M. Schmidt, N. Le Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1):83–112, Mar. 2017. ISSN 1436-4646. doi: 10.1007/s10107-016-1030-6.

[33] J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer. A Survey on Distributed Machine Learning. *ACM Computing Surveys*, 53(2):30:1–30:33, Mar. 2020. ISSN 0360-0300. doi: 10.1145/3377454.

[34] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor. Tackling the Objective Inconsistency Problem in Heterogeneous Federated Optimization. In *Advances in Neural Information Processing Systems*, volume 33, pages 7611–7623. Curran Associates, Inc., 2020.

[35] J. Wang, Z. Charles, Z. Xu, G. Joshi, H. B. McMahan, B. A. y Arcas, M. Al-Shedivat, G. Andrew, S. Avestimehr, K. Daly, D. Data, S. Diggavi, H. Eichner, A. Gadhikar, Z. Garrett, A. M. Girgis, F. Hanzely, A. Hard, C. He, S. Horvath, Z. Huo, A. Ingerman, M. Jaggi, T. Javidi, P. Kairouz, S. Kale, S. P. Karimireddy, J. Konecny, S. Koyejo, T. Li, L. Liu, M. Mohri, H. Qi, S. J. Reddi, P. Richtarik, K. Singhal, V. Smith, M. Soltanolkotabi, W. Song, A. T. Suresh, S. U. Stich, A. Talwalkar, H. Wang, B. Woodworth, S. Wu, F. X. Yu, H. Yuan, M. Zaheer, M. Zhang, T. Zhang, C. Zheng, C. Zhu, and W. Zhu. A Field Guide to Federated Optimization. *arXiv:2107.06917 [cs]*, July 2021.

[36] S. Wang and M. Ji. A Unified Analysis of Federated Learning with Arbitrary Client Participation. *Advances in Neural Information Processing Systems*, 35:19124–19137, Dec. 2022.

[37] S. Wang and M. Ji. A Lightweight Method for Tackling Unknown Participation Statistics in Federated Averaging, Jan. 2024.

[38] Y. Yan, C. Niu, Y. Ding, Z. Zheng, S. Tang, Q. Li, F. Wu, C. Lyu, Y. Feng, and G. Chen. Federated Optimization Under Intermittent Client Availability. *INFORMS Journal on Computing*, 36(1):185–202, Jan. 2024. ISSN 1091-9856. doi: 10.1287/ijoc.2022.0057.

[39] H. Yang, M. Fang, and J. Liu. Achieving Linear Speedup with Partial Worker Participation in Non-IID Federated Learning. In *International Conference on Learning Representations*, Oct. 2020.

[40] H. Yang, X. Zhang, P. Khanduri, and J. Liu. Anarchic Federated Learning. In *Proceedings of the 39th International Conference on Machine Learning*, pages 25331–25363. PMLR, June 2022.