FedSteer: Taming Extreme Gradient Staleness in Federated Learning with Corrective Projections and Caching

Haoran Zhang[*1] **Cainã F. Pereira**[3] **Marie Siew**[†4] **Xutong Liu**[5] **Carlee Joe-Wong**[2] **Rachid El-Azouzi**[3]

[1]Electrical and Computer Engineering Dept., The University of Texas at Austin, Austin, Texas, USA
[2]Electrical and Computer Engineering Dept., Carnegie Mellon University, Pittsburgh, Pennsylvania, USA
[3]CERI/LIA, University of Avignon, Avignon, France
[4]Information Systems Technology and Design Pillar, Singapore University of Technology and Design, Singapore
[5]Computer Science & Systems Dept., University of Washington, Tacoma, Washington, USA

Abstract

Federated learning (FL) is often subject to aggregation variance if clients do not consistently participate in training rounds. While reusing stale model updates from inactive clients is a common technique to reduce this variance, we find that with skewed client participation, the resulting update staleness can become severe enough to destabilize training. To remedy this, we propose `FedSteer`, a novel method that constructs a gradient subspace from a cache of recent client gradients to serve as a low-dimensional representation of the current optimization landscape. FedSteer projects an active client's true gradient onto this subspace to find a set of optimal coordinates. For an inactive client, FedSteer reuses these coordinates with the now-evolved subspace drifted by other active clients. This process effectively "steers" outdated gradients toward the current global objective. This is complemented by a selective caching strategy that identifies a representative client subset to form the subspace, reducing server memory. Experiments demonstrate that FedSteer significantly outperforms baselines, preventing performance collapse in challenging scenarios while delivering accuracy gains of over 7% in others. Code is available here.

1 INTRODUCTION

Federated Learning (FL) has become a popular distributed learning paradigm that enables model training on distributed data while ensuring the data remains on client devices [McMahan et al., 2017, Hard et al., 2018, Shah et al., 2020]. In a typical FL system, a central server coordinates training by aggregating model updates (e.g., gradients or weights) that clients compute on their local data.

In this work, we focus on FL applications where clients are edge devices, such as smartphones or Internet-of-Things (IoT) devices. These devices are often constrained by limited battery life, computational power, and network bandwidth, making it feasible for only a fraction of clients to participate in each training round [Ruan et al., 2020, Lan et al., 2023], a setting known as *partial client participation*. Compounding this challenge is the statistical heterogeneity of client data; since the data distributions are often non-IID across clients, their local model updates can vary significantly [Cho et al., 2020]. When combined with partial client participation, this heterogeneity means the aggregated update in any given round may not be representative of the true global data distribution, leading to high variance and instability in the training process [Chen et al., 2020, Rizk et al., 2021]. Furthermore, participation may be systematically biased, as clients with greater computational or network resources may participate more frequently, introducing a convergence bias [Rodio and Neglia, 2024]. While this bias can be mitigated by techniques like importance sampling [Chen et al., 2020]–which re-weights updates to ensure all clients contribute equally in expectation—such methods, despite ensuring unbiased convergence, often introduce even higher variance into the training process [Zhang et al., 2025, Wang et al., 2023].

To mitigate the training instability caused by partial client participation, several prior works reuse stale information to incorporate updates from inactive clients [Gu et al., 2021, Jhunjhunwala et al., 2022, Karimireddy et al., 2020, Rodio and Neglia, 2024]. These methods can be broadly categorized by how they leverage this stale information. *(1) Direct Reuse of Stale Updates:* One line of work directly reuses the last known update from an inactive client as a surrogate for its current one. MIFA [Gu et al., 2021] and FedVARP [Jhunjhunwala et al., 2022] both maintain the most recent update received from each client at the server. In each round, this

[*]The work was performed while the author was affiliated with Carnegie Mellon University.
[†]Corresponding author (marie_siew@sutd.edu.sg).

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:8090–8114.

memory is updated with fresh updates from participating clients, while the stale updates for non-participating clients are retained. MIFA then updates the global model by averaging all stored updates (fresh and stale), while FedVARP employs a SAGA-like variance reduction scheme that uses the stale updates as control variates [Jhunjhunwala et al., 2022, Defazio et al., 2014]. Similarly, SCAFFOLD [Karimireddy et al., 2020] corrects for client-drift using stateful client-specific control variates. When a client is inactive, its stale control variate is implicitly reused during the server-side aggregation, thus incorporating its past state into the global update. ConFREE [Zheng et al., 2025] mitigates gradient conflicts in personalized FL by projecting active clients' updates to remove opposing components, but it does not address the misalignment of stale gradients arising from inactive clients. *(2) Weighted Stale Updates:* Acknowledging that highly stale information can be detrimental, another line of work proposes down-weighting stale updates when aggregating them with fresh ones. FedStale [Rodio and Neglia, 2024] introduces a hyperparameter β to form a convex combination of an update using only fresh information (akin to FedAvg) and one using both fresh and stale information (akin to FedVARP). This allows the algorithm to control the influence of stale updates. This strategy is also common in asynchronous FL, where the updates from slower devices are reweighted to ensure training stability [Xie et al., 2019, Miao et al., 2023, Ma et al., 2024]. Although these methods can handle the variance from partial participation, their main limitation is that they treat stale gradients as fixed vectors. For example, for less active clients, these approaches ignore the fact that the direction of their gradients becomes increasingly misaligned as the global model evolves. This misalignment can destabilize the training process. Crucially, these approaches lack a mechanism to actively correct the stale gradient's direction by leveraging the more current information provided by active clients.

To address this issue, we propose FedSteer, a novel corrective mechanism that replaces a client's stale gradient with a more accurate estimate derived from an evolving gradient subspace. FedSteer constructs this subspace from a cache of recent gradients provided by a small core set of clients $\mathcal{X}$. Although each gradient vector exists in the high-dimensional space $\mathbb{R}^d$, the subspace's dimensionality is no more than the size of the core set $|\mathcal{X}|$. When a client (indexed by i) is active, we project its gradient onto the current subspace to find low-dimensional coordinates ($\mathbf{s}_i$). The coordinates $\mathbf{s}_i$ can be interpreted as stable similarity coefficients, which capture the relationship between client i and clients in the core set. When a client becomes inactive, FedSteer reuses these coordinates to the newly evolved subspace drifted by active clients from the core set to reconstruct a corrected gradient estimate. This process effectively "steers" the client's outdated information to approximate its true local gradient for the current model.

We complement FedSteer with a selective caching strategy that significantly reduces the server's memory overhead. The server only stores the gradient vectors (each in $\mathbb{R}^d$) for the small core set $\mathcal{X}$, and the low-dimensional coordinate vectors are cached for the entire client population (N). FedSteer's memory cost is therefore $O(|\mathcal{X}|d + N|\mathcal{X}|)$, whereas prior methods [Gu et al., 2021, Jhunjhunwala et al., 2022, Rodio and Neglia, 2024] that cache all stale gradients incur a much higher cost of $O(Nd)$.[1] To our knowledge, FedSteer is the first approach that corrects the direction of a client's stale update by leveraging collective, timely information from other clients in FL.

Our Contributions:

- We propose **FedSteer**, a novel algorithm that introduces a directional correction mechanism for stale updates. FedSteer reconstructs a corrected update gradient for inactive clients by applying a client's stable, cached projection coordinates to a dynamic gradient subspace that is drifted by other active clients. FedSteer effectively steers outdated gradients toward the current global objective. We prove that this corrective projection minimizes the global aggregation variance, providing theoretical support for its performance.
- We introduce a **selective client caching** strategy to make FedSteer memory-efficient. This method iteratively optimizes a small, representative core set of clients during a warm-up phase. The updated gradients from this core set are sufficient to construct an effective gradient subspace, significantly reducing the server's storage cost.
- We provide a rigorous **convergence analysis**, proving that FedSteer achieves a tighter convergence upper bound compared to prior methods that reuse stale information without a directional correction mechanism.
- Our **extensive empirical evaluation** on multiple datasets under different settings of data and system heterogeneity demonstrates that FedSteer significantly outperforms state-of-the-art baselines. It achieves an accuracy gain of at least 7.9% over the next-best method and prevents training collapse in the most challenging scenarios.

The rest of the paper is organized as follows. In Section 2, we formulate the problem of aggregation variance and introduce our proposed method, FedSteer, detailing its corrective projection mechanism and selective caching strategy. We then provide a convergence analysis in Section 3 and present extensive experimental results in Section 4. Finally, we conclude the paper in Section 5. Detailed proofs for the

[1] In experiments, we set $N = 100$, $|\mathcal{X}| = 10$, and the model dimension d exceeds 1.6×10^6. With these values, our method reduces the gradient-caching memory overhead by nearly an order of magnitude.

convergence analysis are provided in Supplementary Material A, along with additional experimental settings and results in Supplementary Material B.

2 METHODS

2.1 PROBLEM FORMULATION

We consider an FL system with a set of N clients, indexed by $i \in \mathcal{N} = \{1, \ldots, N\}$. The global objective is to minimize a weighted average of the clients' local loss functions:

$$\min_{\mathbf{w} \in \mathbb{R}^d} F(\mathbf{w}) \triangleq \sum_{i=1}^{N} d_i F_i(\mathbf{w}), \tag{1}$$

where $\mathbf{w} \in \mathbb{R}^d$ is the global model and $F_i(\mathbf{w}) \triangleq \mathbb{E}_{\xi_i \sim \mathcal{D}_i}[\ell(\mathbf{w}, \xi_i)]$ is the local objective for client i, representing the expected loss over its data distribution $\mathcal{D}_i$. Each client's contribution is scaled by a weight d_i, typically set in proportion to its dataset size, such that $d_i = \frac{|\mathcal{D}i|}{\sum_{j \in \mathcal{N}} |\mathcal{D}_j|}$. We model each client's participation as an independent Bernoulli trial with a client-specific participation probability p_i.

FL training process: In each global round $t = 1, 2, \ldots, T$, a subset of clients $\mathcal{A}_t$ is active, with client i participating with probability p_i. Active clients perform E local training epochs (e.g., using SGD) starting from $\mathbf{w}_i^0 = \mathbf{w}^t$. After training, each active client $i \in \mathcal{A}_t$ computes its total update $\mathbf{g}_i^t = \mathbf{w}_i^0 - \mathbf{w}_i^E$ and sends it to the server. Define the stale gradient of client i as: if $i \in \mathcal{A}_t$: $\mathbf{h}_i^{t+1} = \mathbf{g}_i^t$, otherwise: $\mathbf{h}_i^{t+1} = \mathbf{h}_i^t$.

2.2 CORRECTIVE PROJECTIONS FOR STALE GRADIENTS

To address the issue of extreme staleness, we propose **FedSteer**, a method that corrects a stale client update by projecting the new gradient onto a dynamically evolving gradient subspace, and reuse the subspace coordinates instead. As shown in Fig. 1, this process steers the outdated information toward the current optimization landscape, making it relevant for the global update.

2.2.1 The Dynamic Gradient Subspace

The foundation of our method is a low-dimensional gradient subspace constructed from recent client updates. The server maintains a cache of the most recent raw gradients, denoted as $\mathbf{h}_i^t \in \mathbb{R}^d$, from a small, representative subset of clients $\mathcal{X} \subseteq \mathcal{N}$, referred to as the **core set** (the selection process is described in Section 2.4). To ensure these gradients can form a well-defined basis, we make the following assumption:

Assumption 1 (Non-trivial gradients)**.** *For any cached gradient $\mathbf{h}_i^t, i \in \mathcal{X}$, its magnitude is strictly positive, i.e., $\|\mathbf{h}_i^t\|_2 > 0$.*

The assumption allows us to safely normalize each raw gradient, improving numerical stability. The normalized basis vector $\bar{\mathbf{h}}_i^t$ is defined as:

$$\bar{\mathbf{h}}_i^t = \frac{\mathbf{h}_i^t}{\|\mathbf{h}_i^t\|_2}. \tag{2}$$

The dynamic subspace is then formally defined by a matrix $\mathbf{Q}_t \in \mathbb{R}^{d \times |\mathcal{X}|}$, whose columns are these normalized basis vectors:

$$\mathbf{Q}_t = \left[\bar{\mathbf{h}}_{j_1}^t \quad \ldots \quad \bar{\mathbf{h}}_{j_k}^t \quad \ldots \quad \bar{\mathbf{h}}_{j_{|\mathcal{X}|}}^t \right]_{j_k \in \mathcal{X}}. \tag{3}$$

While the core set $\mathcal{X}$ can be fixed, the subspace itself is dynamic, as the basis vectors $\bar{\mathbf{h}}_j^t$ are updated whenever a client $j \in \mathcal{X}$ participates in a training round. Even if some core set clients are temporarily inactive, the subspace $\mathbf{Q}_t$ as a whole continues to evolve through the participation of its other members. Consequently, $\mathbf{Q}_t$ serves as a continuously updated, low-dimensional representation of the dominant directions of recent client update gradients. This set of dominant directions provides a basis for estimating any client's update. Since $\mathcal{X}$ is chosen to be representative, projecting a client's gradient onto $\mathbf{Q}_t$ effectively decomposes its update into a combination of these core directions. The resulting coordinates, which capture the client's similarity to the core set $\mathcal{X}$, are then reused to reconstruct an estimated gradient for any inactive client.

2.2.2 Corrective Projection via Cached Coordinates

Rather than directly reusing a stale gradient $\mathbf{h}_i^t$, which risks being outdated and destabilizing the training process, FedSteer computes a more relevant approximation, $\hat{\mathbf{g}}_i^t$, that lies within the dynamic subspace $\mathbf{Q}_t$. This approximation is defined as a linear combination of the subspace's basis vectors:

$$\hat{\mathbf{g}}_i^t = \mathbf{Q}_t \mathbf{s}_i. \tag{4}$$

Here, the vector $\mathbf{s}_i \in \mathbb{R}^{|\mathcal{X}|}$ denotes the coordinates of the gradient approximation within the subspace. The optimal coordinates are those that make $\hat{\mathbf{g}}_i^t$ the best possible estimate of the true gradient $\mathbf{g}_i^t$. A standard projection, however, can yield coordinates with large magnitudes that overfit to the specific basis vectors of $\mathbf{Q}_t$ at a single time step. To find a more stable and generalizable coordinate representation, we introduce a regularization term into the optimization.

For an active client $i \in \mathcal{A}_t$, the server computes its optimal coordinates $\mathbf{s}_i^*$ by solving the following regularized least-squares (Ridge Regression) problem:

$$\min_{\mathbf{s}_i \in \mathbb{R}^{|\mathcal{X}|}} \|\mathbf{g}_i^t - \mathbf{Q}_t \mathbf{s}_i\|^2 + \lambda \|\mathbf{s}_i\|^2, \forall\, i \in \mathcal{A}_t, \tag{5}$$

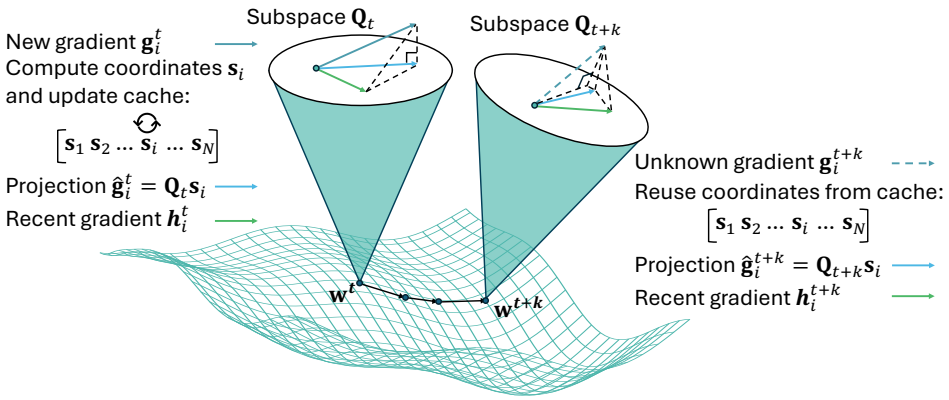


Figure 1: An overview of FedSteer's corrective mechanism. At round t, an active client i's gradient $\mathbf{g}_i^t$ is projected onto the subspace $\mathbf{Q}_t$ and the resulting low-dimensional coordinates $\mathbf{s}_i$ are cached. In a subsequent round $t+k$, as long as client i remains inactive, FedSteer reuses the cached coordinates $\mathbf{s}_i$ with the newly evolved subspace $\mathbf{Q}_{t+k}$ to form a corrected projection $\hat{\mathbf{g}}_i^{t+k} = \mathbf{Q}_{t+k}\mathbf{s}_i$. As illustrated, this "steered" projection can better estimate the true unknown gradient $\hat{\mathbf{g}}_i^{t+k}$ compared to its stale gradient $\mathbf{h}_i^{t+k}$.

Algorithm 1 FedSteer algorithm

Input: clients: $i \in \mathcal{N} = \{1, 2, \ldots, N\}$, client participation distribution $\{p_i\}_{i\in\mathcal{N}}$, core set $\mathcal{X}$, regularization factor λ, learning rate η_c and η_s.
Objective: $\min \sum_{i\in\mathcal{N}} d_i F_i(\mathbf{w})$
Initialization: The global model weights $\mathbf{w}^1$, $\mathbf{s}_i^{\text{cache}} = \mathbf{0}$ for all clients.
for global round $t = 1, \cdots, T$ **do**
  Determine the set of active clients $\mathcal{A}_t$ from $\{p_i\}_{i\in\mathcal{N}}$
  for each client $i \in \mathcal{A}_t$, in parallel **do**
    Local initialization $\mathbf{w}_i^0 = \mathbf{w}^t$
    SGD for E epochs, obtain $\mathbf{g}_i^t = \mathbf{w}_i^0 - \mathbf{w}_i^E$
    Send $\mathbf{g}_i^t$ to the server
  end for
  At server:
  Receive updates $\mathbf{g}_i^t$ from active clients
  Update gradient estimation $\hat{\mathbf{g}}_i^t = \mathbf{Q}_t\mathbf{s}_i^{\text{cache}}$
  Server aggregation: $\mathbf{w}^{t+1} = \mathbf{w}^t - \eta_s\Delta_t$
  Compute $\mathbf{s}_i^*$ for $i \in \mathcal{A}_t$ (Eq. 6), cache $\mathbf{s}_i^{\text{cache}} = \mathbf{s}_i^*$
  Update $\mathbf{h}_i^{t+1}$ and $\mathbf{Q}_{t+1}$ (Eq. (3))
  Broadcast $\mathbf{w}^{t+1}$ to clients
end for

where the regularization parameter $\lambda > 0$ penalizes large coefficient values. This prevents the projection from relying too heavily on any single cached gradient in the subspace, promoting a more robust representation. The closed-form solution is:

$$\mathbf{s}_i^* = (\mathbf{Q}_t^\top\mathbf{Q}_t + \lambda\mathbf{I})^{-1}\mathbf{Q}_t^\top\mathbf{g}_i^t. \tag{6}$$

Remark 1. *The matrix $(\mathbf{Q}_t^\top\mathbf{Q}_t + \lambda\mathbf{I})$ is guaranteed to be positive definite, and therefore invertible, for any $\lambda > 0$. This ensures the solution in Eq. (6) is always unique and well-defined, regardless of whether the columns of $\mathbf{Q}_t$ are linearly independent.*

Once computed, $\mathbf{s}_i^*$ are stored in the server's cache for client i: $\mathbf{s}_i^{\text{cache}} = \mathbf{s}_i^*$, and its gradient estimate is written as $\hat{\mathbf{g}}_i^t = \mathbf{Q}_t\mathbf{s}_i^{\text{cache}}$, which is used in global aggregation.

For an inactive client $i \notin \mathcal{A}_t$, its true gradient is unknown. FedSteer avoids directly reusing its stale gradient $\mathbf{h}_i^t$. Instead, it retrieves the client's cached coordinates $\mathbf{s}_i^{\text{cache}}$, which encode the stable relationship between client i's update and the core set's basis vectors. It then applies these coordinates to the *current* subspace $\mathbf{Q}_t$. As illustrated in Fig. 1, because this subspace has been updated by active clients, the resulting reconstruction $\hat{\mathbf{g}}_i^t = \mathbf{Q}_t\mathbf{s}_i^{\text{cache}}$ effectively "steers" the inactive client's historical information to align with the current optimization path.

2.2.3 The FedSteer Aggregation Rule

The final server update uses the projected gradients $\hat{\mathbf{g}}_i^t = \mathbf{Q}_t\mathbf{s}_i^{\text{cache}}$ as a baseline estimate for all clients and applies a variance reduction correction using the true gradients from the active set:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_s\Delta_t \tag{7}$$

$$\Delta_t = \underbrace{\sum_{i\in\mathcal{N}} d_i\hat{\mathbf{g}}_i^t}_{\text{Projection Baseline}} + \underbrace{\sum_{i\in\mathcal{A}_t}\frac{d_i}{p_i}(\mathbf{g}_i^t - \hat{\mathbf{g}}_i^t)}_{\text{Residual Correction}} \tag{8}$$

Proposition 1 (Unbiased Estimator)**.** *The FedSteer global update Δ_t is an unbiased estimator of the true global gradient update: $\mathbb{E}[\Delta_t] = \sum_{i\in\mathcal{N}} d_i\mathbf{g}_i^t$.*

Proof. Taking the expectation over client sampling ($\mathbb{E}[\mathbb{1}_{i\in\mathcal{A}_t}] = p_i$):

$$\mathbb{E}[\Delta_t] = \sum_{i\in\mathcal{N}} d_i\hat{\mathbf{g}}_i^t + \mathbb{E}\left[\sum_{i\in\mathcal{N}}\mathbb{1}_{i\in\mathcal{A}_t}\frac{d_i(\mathbf{g}_i^t - \hat{\mathbf{g}}_i^t)}{p_i}\right] = \sum_{i\in\mathcal{N}} d_i\mathbf{g}_i^t$$

The estimator is therefore unbiased. □

The unbiased estimator in Eq. (8) reduces variance by leveraging a baseline estimate ($\hat{\mathbf{g}}_i^t$) which allows the active clients to estimate only the small, low-variance residual error ($\mathbf{g}_i^t - \hat{\mathbf{g}}_i^t$). The pseudocode of FedSteer is provided in Algorithm 1.

Remark 2 (Reduction to prior work)**.** *FedSteer reduces to prior methods by selecting specific coordinates* $\{\mathbf{s}_i\}_{i\in\mathcal{N}}$*. Assuming* $\mathcal{X} = \mathcal{N}$*, FedSteer reduces to: (1) FedVARP [Jhunjhunwala et al., 2022], by setting coordinates to* $\mathbf{s}_i^{\text{cache}} = \|\mathbf{h}_i^t\|_2 \cdot \mathbf{e}_i$*, where* $\mathbf{e}_i$ *is a one-hot vector, to recover the exact gradient estimate* $\hat{\mathbf{g}}_i^t = \mathbf{h}_i^t$*; and (2) FedStale [Rodio and Neglia, 2024], by choosing coordinates* $\mathbf{s}_i^{\text{cache}} = \beta\|\mathbf{h}_i^t\|_2 \cdot \mathbf{e}_i$ *to yield the down-weighted estimate* $\hat{\mathbf{g}}_i^t = \beta\mathbf{h}_i^t$*.*

Complexity analysis: FedSteer incurs no additional communication cost over standard FL (e.g., FedAvg/FedVARP): only active clients transmit updates, and client-side computation remains unchanged. The server solves a small ridge regression with complexity $O(k^2d + k^3)$, which is modest since $k \ll N$ and comparable to standard aggregation. FedSteer is also more memory-efficient than stale-gradient methods (FedVARP/FedStale/MIFA), reducing storage from $O(Nd)$ to $O(Nk + kd)$ by keeping low-dimensional coordinates for all clients and full gradients only for a small core set. Detailed complexity analysis is provided in the Supplementary Material B.2.

2.3 VARIANCE MINIMIZATION VIA PROJECTION

We show that having each client i minimize its local projection error (Eq. (5) with $\lambda = 0$) over its own variable vector $\mathbf{s}_i$ is equivalent to minimizing the variance of the global update Δ_t over the collective set of variables $\{\mathbf{s}_i\}_{i\in\mathcal{N}}$.

Theorem 1 (Variance minimization)**.** *Given the true gradients* $\{\mathbf{g}_i^t\}_{i\in\mathcal{N}}$ *and the subspace matrix* $\mathbf{Q}_t$*, the variance of the global update,* $\text{Var}(\Delta_t)$*, is minimized when each client's coordinate vector* $\mathbf{s}_i$ *is the solution to the following least-squares problem (Eq.* (5) *with* $\lambda = 0$*) for each client* $i \in \mathcal{N}$*:*

$$\mathbf{s}_i = \underset{\mathbf{s}_i' \in \mathbb{R}^{|\mathcal{X}|}}{\text{argmin}} \|\mathbf{g}_i^t - \mathbf{Q}_t\mathbf{s}_i'\|^2. \tag{9}$$

Proof. The variance is taken over the random sampling of the active client set $\mathcal{A}_t$. Since the first term $\sum_{i\in\mathcal{N}} d_i\hat{\mathbf{g}}_i^t$ is constant with respect to this sampling, it does not contribute to the variance. The variance of the update is therefore: $\text{Var}(\Delta_t) = \text{Var}\left(\sum_{i\in\mathcal{N}} \mathbb{1}_{i\in\mathcal{A}_t}\frac{d_i}{p_i}(\mathbf{g}_i^t - \hat{\mathbf{g}}_i^t)\right)$, where $\mathbb{1}_{i\in\mathcal{A}_t}$ is the indicator variable defined in Proposition 1. Since clients are sampled independently, the variance of the sum is the sum of the variances: $\text{Var}(\Delta_t) = \sum_{i\in\mathcal{N}} \text{Var}\left(\mathbb{1}_{i\in\mathcal{A}_t}\frac{d_i}{p_i}(\mathbf{g}_i^t - \hat{\mathbf{g}}_i^t)\right)$. For each client i, we use $\text{Var}(\mathbf{X}) = \mathbb{E}[\|\mathbf{X}\|^2] - \|\mathbb{E}[\mathbf{X}]\|^2$. Let $\mathbf{r}_i = \mathbf{g}_i^t - \hat{\mathbf{g}}_i^t$ denote the residual vector. The variance for client i's term is:

$$\mathbb{E}\left[\left\|\mathbb{1}_{i\in\mathcal{A}_t}\frac{d_i}{p_i}\mathbf{r}_i\right\|^2\right] - \left\|\mathbb{E}\left[\mathbb{1}_{i\in\mathcal{A}_t}\frac{d_i}{p_i}\mathbf{r}_i\right]\right\|^2 = \frac{d_i^2(1-p_i)}{p_i}\|\mathbf{g}_i^t - \hat{\mathbf{g}}_i^t\|^2.$$

Therefore, minimizing the variance is equivalent to:

$$\min_{\{\mathbf{s}_i\}_{i\in\mathcal{N}}} \sum_{i\in\mathcal{N}} \frac{d_i^2(1-p_i)}{p_i}\|\mathbf{g}_i^t - \mathbf{Q}_t\mathbf{s}_i\|^2.$$

Since the objective is separable across clients, i.e., it consists of a sum of independent terms each involving only a single variable $\mathbf{s}_i$, the global optimization problem decouples into independent weighted least-squares problems for each client. □

Algorithm 2 Selective client caching

Input: Full client set $\mathcal{N}$, core set size $k \le |\mathcal{N}|$, number of selection cycles T_0, number of swap iterations I_{max} per cycle.
Initialization: Randomly initialize a core set $\mathcal{X} \subseteq \mathcal{N}$ of size k.
for selection cycle $t = 1, \ldots, T_0$ **do**
  Collect $\mathbf{g}_i^t$ from all clients $i \in \mathcal{N}$ asynchronously
  for $iter = 1, \ldots, I_{max}$ **do**
    Find the optimal swap (j_{out}^*, j_{in}^*) by solving:
    $\underset{j_{out}\in\mathcal{X},\, j_{in}\in\mathcal{N}\setminus\mathcal{X}}{\text{argmin}} J\big((\mathcal{X}\setminus\{j_{out}\})\cup\{j_{in}\}\big)$
    Let $\mathcal{X}_{\text{new}} = (\mathcal{X}\setminus\{j_{out}^*\})\cup\{j_{in}^*\}$
    if $J(\mathcal{X}_{\text{new}}) < J(\mathcal{X})$ **then**
      $\mathcal{X} = \mathcal{X}_{\text{new}}$ ▷ Update $\mathcal{X}$ with the best swap
    else
      break ▷ No further improvement
    end if
  end for
  Aggregation: $\mathbf{w}^{t+1} = \mathbf{w}^t - \eta_s\sum_{i\in\mathcal{N}} d_i\mathbf{g}_i^t$
end for
return the optimized core set $\mathcal{X}$

2.4 SELECTIVE CLIENT CACHING

We propose a method to select an effective **core set** $\mathcal{X}$ during an initial warm-up phase. After the warm-up phase, the optimized core set $\mathcal{X}$ is fixed for the formal training.[2] The warm-up phase consists of T_0 selection cycles. In each selection cycle, the server holds the global model constant and asynchronously collects available gradient from each client as they naturally become active. Once gradients from all N clients have been gathered, the server has a complete

[2]To ensure a fair comparison with baselines, the global model's weights are re-initialized before the formal training process begins.

snapshot of the gradient landscape and proceeds with the optimization step for $\mathcal{X}$. It then performs an aggregation including all clients to update the global model for the next selection cycle.

The core set selection within a selection cycle is performed via a **greedy local search** that iteratively improves an initially random subset $\mathcal{X}$. The objective is to find a core set whose gradient subspace can accurately reconstruct the gradients of the entire client population. Specifically, the algorithm seeks to find a set $\mathcal{X}$ that minimizes the total regularized projection error, $J(\mathcal{X})$, defined as:

$$J(\mathcal{X}) = \sum_{i \in \mathcal{N}} d_i \left(\|\mathbf{G}_i^t - \mathbf{Q}_t(\mathcal{X})\mathbf{s}_i\|_2^2 + \lambda \|\mathbf{s}_i\|_2^2 \right), \quad (10)$$

where $\mathbf{Q}_t(\mathcal{X})$ is the subspace matrix generated from the gradients of clients in $\mathcal{X}$, following the same definition in Eq. (3). Since an exhaustive search for the optimal $\mathcal{X}$ is computationally intractable, we refine the set through **single-client swaps**. In each iteration, it evaluates exchanging one client in the core set with one outside of it and executes the single swap that yields the greatest reduction in the objective $J(\mathcal{X})$. To manage the computational cost, we limit this search to a small number of iterations (e.g., at most five) per selection cycle. For each selection cycle, by updating the model and re-optimizing the core set, we ensure $\mathcal{X}$ is robust because its selection is based on performance across several different versions of the model. This prevents the choice of $\mathcal{X}$ from overfitting to a specific model state, ensuring $\mathcal{X}$ remains representative as the model continues to train. The pseudocode of the algorithm is provided in Algorithm 2.

Complexity analysis: The selective caching algorithm has complexity $O(T_0 I_{\max} N k)$, where T_0 is the number of selection cycles and $I_{\max}$ is the number of swap iterations per cycle. This cost is incurred only during a short warm-up phase on the server (typically the first $T_0 = 5$ rounds), after which the core set is fixed and no further selection overhead is introduced for the remaining training rounds. To scale to very large client populations N, we also employ a subsampled selection strategy, which achieves comparable performance (Supplementary Material B.3).

3 CONVERGENCE ANALYSIS

We make the following standard assumptions.

Assumption 2 (L-smoothness)**.** *Each F_i is L-smooth, and thus $F = \sum_{i \in \mathcal{N}} d_i F_i$ is also L-smooth.*

Assumption 3 (Bounded variance at client-level)**.** *The stochastic gradient at each client is an unbiased estimator of the local gradient: $\mathbb{E}_{\xi_i \sim \mathcal{D}_i}[\nabla F_i(\mathbf{w}, \xi_i)] = \nabla F_i(\mathbf{w})$, and its variance is bounded: $Var_{\xi_i \sim \mathcal{D}_i}(\nabla F_i(\mathbf{w}, \xi_i)) \leq \sigma$.*

Assumption 4 (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*

$$\|\nabla F_i(\mathbf{w}) - \nabla F(\mathbf{w})\|^2 \leq \sigma_g^2, \quad \forall \mathbf{w}, i.$$

Assumption 5 (Partial and heterogeneous client participation)**.** *In each round t, client i participates with a probability p_i, independently of previous rounds and other clients. p_i is bounded: $p_{min} < p_i < p_{max}$.*

Theorem 2 (Convergence analysis)**.** *Under Assumptions 2-5, $\eta_c \leq \frac{1}{4\sqrt{2E(E-1)}}$, and $\eta_s \leq \frac{1}{2L}$, the iterates $\{\mathbf{w}^t\}$ generated by FedSteer satisfy:*

$$\begin{aligned}
\min_{t \in [1,T]} \mathbb{E}\|\nabla F(\mathbf{w}^t)\|^2 \leq & \underbrace{\frac{4(F(\mathbf{w}^1) - F(\mathbf{w}^*))}{\eta_s T}}_{\textit{iterate initialization error}} \\
& + \underbrace{4\eta_s L \frac{\sum_{t=1}^T \mathbb{E}\|\Delta^t - \mathbb{E}[\Delta^t]\|^2}{T}}_{\textit{partial update variance error}} \\
& + \underbrace{\Gamma \sigma^2 [2\eta_c^2 L^2 (E-1) + \frac{1}{TE}]}_{\textit{stochastic gradient error}} \\
& + \underbrace{(2+\Gamma) 8 \eta_c^2 L^2 E (E-1) \sigma_g^2}_{\textit{error from data heterogeneity}} \\
& + \underbrace{\frac{1}{T} \frac{\Gamma(1-p_{min})}{p_{avg}} \frac{1}{N} \sum_{i=1}^N \|\nabla F_i(\mathbf{w}^1) - \mathbf{h}_i^1\|^2}_{\textit{memory initialization error}},
\end{aligned}$$

where $\Gamma = \frac{p_{min} p_{avg}}{\eta_s L (2 - p_{min})}$ with $p_{avg} = (1/N) \sum_{i=1}^N p_i$.

We provide the proof of Theorem 2 in the Supplementary Material A. The convergence bound consists of several terms. Some, like the *iterate initialization error*, vanish as the number of global rounds T grows. Others, including the *partial update variance error*, *stochastic gradient error*, and *error from data heterogeneity* contribute to a non-vanishing error floor, which is characteristic of stochastic, non-convex optimization in FL. As stated in Theorem 1, FedSteer can explicitly minimize the *partial update variance* term $\mathbb{E}\left[\|\Delta^t - \mathbb{E}[\Delta^t]\|^2\right]$. By reducing the magnitude of this dominant, non-vanishing term, FedSteer achieves a tighter convergence bound compared to prior methods including FedVARP [Jhunjhunwala et al., 2022] and FedStale [Rodio and Neglia, 2024]. As shown in Remark 2, methods like FedVARP and FedStale can be viewed as constrained versions of FedSteer where the coordinates $\mathbf{s}_i$ are restricted to be one-hot vectors. This suboptimal choice precludes the minimization of the partial update variance. Consequently, these methods, with a larger variance in the global update, result in a looser theoretical convergence bound.

Table 1: Final average test accuracy on EMNIST, Fashion-MNIST, and CIFAR-10 under moderate ($\gamma = 0.7$) and extreme ($\gamma = 0.9$) heterogeneity; best results are in bold. FedSteer consistently outperforms all baselines. Its advantage is most significant under extreme heterogeneity (EMNIST, $\gamma = 0.9$).

Methods	EMNIST		Fashion-MNIST		CIFAR-10	
	γ=0.9	γ=0.7	γ=0.9	γ=0.7	γ=0.9	γ=0.7
Full participation	$0.747_{\pm.023}$	$0.747_{\pm.023}$	$0.686_{\pm.006}$	$0.686_{\pm.005}$	$0.695_{\pm.004}$	$0.695_{\pm.004}$
FedAvg [McMahan et al., 2017]	$0.314_{\pm.015}$	$0.717_{\pm.027}$	$0.509_{\pm.010}$	$0.609_{\pm.010}$	$0.591_{\pm.010}$	$0.631_{\pm.016}$
FedVARP [Jhunjhunwala et al., 2022]	$0.281_{\pm.014}$	$0.731_{\pm.026}$	$0.486_{\pm.008}$	$0.588_{\pm.007}$	$0.569_{\pm.007}$	$0.604_{\pm.013}$
FedStale [Rodio and Neglia, 2024]	$0.309_{\pm.014}$	$0.734_{\pm.027}$	$0.497_{\pm.007}$	$0.596_{\pm.009}$	$0.585_{\pm.005}$	$0.638_{\pm.009}$
MIFA [Gu et al., 2021]	$0.243_{\pm.005}$	$0.287_{\pm.006}$	$0.461_{\pm.011}$	$0.500_{\pm.013}$	$0.530_{\pm.008}$	$0.551_{\pm.012}$
SCAFFOLD [Karimireddy et al., 2020]	$0.048_{\pm.00}$	$0.073_{\pm.00}$	$0.321_{\pm.004}$	$0.500_{\pm.007}$	$0.492_{\pm.006}$	$0.550_{\pm.010}$
FedProx [Li et al., 2020]	$0.051_{\pm.00}$	$0.093_{\pm.002}$	$0.351_{\pm.005}$	$0.597_{\pm.010}$	$0.509_{\pm.005}$	$0.553_{\pm.008}$
FedSteer	$0.551_{\pm.028}$	$0.740_{\pm.032}$	$\mathbf{0.554}_{\pm.010}$	$\mathbf{0.657}_{\pm.011}$	$0.599_{\pm.009}$	$0.656_{\pm.009}$
FedSteer (enforce=5)	$\mathbf{0.620}_{\pm.032}$	$\mathbf{0.740}_{\pm.029}$	$0.539_{\pm.008}$	$0.651_{\pm.009}$	$\mathbf{0.622}_{\pm.012}$	$\mathbf{0.660}_{\pm.015}$

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

FL system: We simulate an FL system with $N = 100$ clients under independent **system** and **data heterogeneity**. For system heterogeneity, we create a weak group (50 clients with participation probability $p_i = 0.04$) and a powerful group (50 clients with $p_i = 0.16$), resulting in a network-wide average participation rate of 0.1. For data heterogeneity, a γ fraction of clients (the common-label group) are assigned data from the first half of the L total labels, while the remaining $1 - \gamma$ fraction (rare-label group) receive data from the second half. To ensure a uniform global label distribution while maintaining an average of 100 datapoints per client, we allocate $\lfloor 50/\gamma \rfloor$ datapoints to each common client and $\lfloor 50/(1-\gamma) \rfloor$ to each rare client. We conduct experiments under moderate ($\gamma = 0.7$) and extreme ($\gamma = 0.9$) heterogeneity settings.

Datasets & Models: Fashion-MNIST, EMNIST, and CIFAR-10 datasets are used [Xiao et al., 2017, Cohen et al., 2017, Krizhevsky et al., 2009]. The model for Fashion-MNIST features two convolutional, two max-pooling, and two fully-connected layers, and the EMNIST model uses a deeper architecture with three of each layer type. We apply a PreAct ResNet-18 [He et al., 2016] for the CIFAR-10 task. Experiments are conducted on a GeForce 1080Ti with 8 different random seeds. We report the average accuracy and variance across these runs. We provide more details of experimental settings in the Supplementary Material B.

Baselines: We compare FedSteer against a suite of baseline methods that employ diverse strategies for handling client stale updates. 1) *FedAvg* serves as the fundamental baseline; it aggregates updates only from active clients in each round and discards any stale information. 2) *FedVARP* and *FedStale* both reuse stale updates using a SAGA-like aggregation rule for variance reduction. FedStale further introduces a down-weighting mechanism (β) to mitigate the negative impact of excessively old gradients. We report FedStale's best result from the selection of $\beta \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$. 3) *SCAFFOLD* corrects for client drift using control variates, which implicitly leverages historical information. 4) *MIFA* takes a direct approach by substituting updates from inactive clients with their most recent stale gradients. 5) *FedProx* ($\mu = 0.1$) applies the regularization to limit the local drift caused by client data heterogeneity. 6) We also benchmark against a *Full Participation* setting, where the participation rate is 1.0, to establish a performance upper bound for the given data distribution.

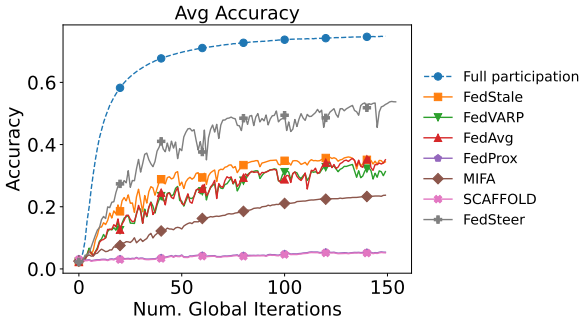


Figure 2: Comparison of FedSteer with other baselines ($\gamma = 0.9$). FedSteer (blue and orange lines) significantly outperforms all baselines. On EMNIST, FedSteer converges stably while baselines struggle, with SCAFFOLD and FedProx's performance collapsing entirely. Additional plots of Fashion-MNIST and CIFAR-10 are shown in Appendix B.4.

4.2 COMPARISON WITH BASELINE METHODS

The results in Fig. 2 and Table 1 are reported using a core set size of $|\mathcal{X}| = 10$ for EMNIST and CIFAR-10, and $|\mathcal{X}| = 40$ for Fashion-MNIST, with a regularization factor of $\lambda = 0.5$. The core set $\mathcal{X}$ was optimized for 5 selection

cycles ($T_0 = 5$). We also compare two schemes for updating $\mathbf{s}_i$: the default method (update only when active) and an enforced update every 5 rounds (`enforce=5`).

Our experiments reveal the vulnerability of existing baselines to highly stale updates under extreme heterogeneity; FedSteer, however, successfully overcomes this issue to deliver a significant performance advantage. As we mentioned, the experiment setting features a rare-label client group ($1 - \gamma$ fraction) that provides critical and unique information. However, half of these clients have a very low participation probability (weak group, $p_i = 0.04$), which introduces severely stale updates into the training process. As shown in Fig. 2 and quantified in Table 1, existing methods are highly susceptible to these highly stale updates, especially under the extreme heterogeneity setting ($\gamma = 0.9$) on EMNIST. For FedVARP, FedStale, and MIFA, the stale gradients fail to align with the current optimization objective, resulting in final accuracies (0.281, 0.309, and 0.243, respectively) that are on par with or worse than a standard FedAvg baseline (0.314). The failure of SCAFFOLD and FedProx is pronounced: their drift correction mechanisms are fundamentally compromised by extreme staleness, leading to a collapse of the training process and a final accuracy of just 0.048 and 0.051.

FedSteer, however, achieves stable convergence by avoiding the direct use of stale updates. Instead, it leverages their projection coordinates on a dynamic subspace Q_t, which is continuously steered by more active clients. On EMNIST ($\gamma = 0.9$), the enforced-update version of FedSteer achieves a final accuracy of 0.620, nearly doubling the performance of the best-performing baseline. On Fashion-MNIST ($\gamma = 0.9$), FedSteer provides an 8.8% higher accuracy (0.554) compared to the next-best method, FedAvg (0.509). Under moderate heterogeneity ($\gamma = 0.7$), where baselines are more competitive, FedSteer consistently exceeds the top-performing methods by 7.8%, approaching the ideal accuracy of full participation. FedSteer maintains this advantage on the larger and more complex CIFAR-10 dataset, achieving accuracies of 0.622 and 0.660 under $\gamma = 0.9$ and $\gamma = 0.7$, respectively.

4.3 IMPACT OF CORE SET SIZE AND REGULARIZATION

We evaluate the impact of the core set size $|\mathcal{X}|$ and regularization on FedSteer's performance on EMNIST, with results presented in Fig. 3. In this experiment, projection coordinates are recomputed every five rounds, which requires them to be robust enough to remain effective as the subspace evolves. The results reveal a critical trade-off between the representational capacity of the subspace and the coordinates' generalization ability. **For a small core sets** ($|\mathcal{X}| \leq 30$), the subspace provides a limited representation of the gradient landscape. Strong regularization is crucial here, as it prevents the coordinates from overfitting to the small basis, producing more robust estimates and leading to higher accuracy. **For medium core sets** ($40 \leq |\mathcal{X}| \leq 80$), as the core set grows, the subspace's capacity to represent client gradients increases. In this range, the unregularized approach starts to outperform the regularized one by leveraging the richer, more representative basis without the constraint of a penalty. **For large core sets** ($|\mathcal{X}| > 80$), the performance of both methods degrades. This decline can be attributed to the coordinates becoming over-specialized. When the basis becomes too large, the subspace's high dimensionality allows it to fit client gradients too closely at the moment of computation. These over-specialized coordinates are highly sensitive to the subspace's evolution across subsequent rounds; when they are reused, their effectiveness diminishes, leading to inaccurate gradient reconstructions and a drop in model accuracy.

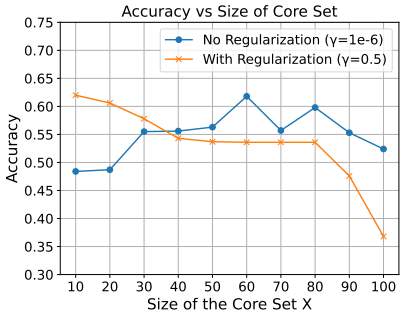


Figure 3: Final test accuracy on EMNIST versus core set size. The plot compares a strongly regularized ($\lambda = 0.5$) and a virtually unregularized ($\lambda = 1\text{e}{-6}$) FedSteer algorithm. Regularization proves crucial for small core sets to prevent overfitting. For large sets, performance degrades because the over-specialized coordinates can be sensitive to the evolving subspace when reused in later rounds.

In Appendix B, we provide the detailed experimental setup (B.1), complexity analysis against standard baselines (B.2), and additional results including a subsampled core set selection strategy for large-scale deployment with reduced complexity (B.3), convergence plots on Fashion-MNIST and CIFAR-10 (B.4), and the stability of coordinate reuse under evolving subspaces (B.5).

5 CONCLUSION

In this work, we introduce FedSteer, a novel algorithm that creates a dynamic, low-dimensional gradient subspace from a small client core set. It projects a client's gradient onto this subspace, computing and caching stable, low-dimensional coordinates for reuse. For inactive clients, these coordinates "steer" outdated information toward the current global objective by being applied to the newly evolved subspace. Complemented by a memory-efficient selective caching strategy,

this mechanism is proven to minimize global update variance and achieve a tighter convergence bound. Experiments confirm FedSteer significantly outperforms baselines, prevents training collapse under extreme heterogeneity, and delivers significant accuracy gains.

6 ACKNOWLEDGEMENTS

This work was supported by NSF CNS-2106891, TREES: ANR-24-TSIA-0004, and A*STAR under its IAF-ICP programme (H25-MCP3438).

References

Wenlin Chen, Samuel Horvath, and Peter Richtarik. Optimal client sampling for federated learning. arXiv preprint arXiv:2010.13723, 2020.

Yae Jee Cho, Jianyu Wang, and Gauri Joshi. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. arXiv preprint arXiv:2010.01243, 2020.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. Emnist: Extending mnist to handwritten letters. In 2017 international joint conference on neural networks (IJCNN), pages 2921–2926. IEEE, 2017.

Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. Advances in neural information processing systems, 27, 2014.

Xinran Gu, Kaixuan Huang, Jingzhao Zhang, and Longbo Huang. Fast federated learning in the presence of arbitrary device unavailability. In NeurIPS, 2021.

Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction. arXiv preprint arXiv:1811.03604, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In ECCV, 2016.

Divyansh Jhunjhunwala, Pranay Sharma, Aushim Nagarkatti, and Gauri Joshi. FedVARP: Tackling the variance due to partial client participation in federated learning. In UAI, 2022.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In ICML, pages 5132–5143. PMLR, 2020.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Guangchen Lan, Xiao-Yang Liu, Yijing Zhang, and Xiaodong Wang. Communication-efficient federated learning for resource-constrained edge devices. IEEE Transactions on Machine Learning in Communications and Networking, 1:210–224, 2023. doi: 10.1109/TMLCN.2023.3309773.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. Proceedings of Machine learning and systems, 2:429–450, 2020.

Jeffrey Ma, Alan Tu, Yiling Chen, and Vijay Janapa Reddi. Fedstaleweight: Buffered asynchronous federated learning with fair aggregation via staleness reweighting. arXiv preprint arXiv:2406.02877, 2024.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In AISTATS, pages 1273–1282. PMLR, 2017.

Yinbin Miao, Ziteng Liu, Xinghua Li, Meng Li, Hongwei Li, Kim-Kwang Raymond Choo, and Robert H Deng. Robust asynchronous federated learning with time-weighted and stale model aggregation. IEEE Transactions on Dependable and Secure Computing, 21(4):2361–2375, 2023.

Elsa Rizk, Stefan Vlaski, and Ali H Sayed. Optimal importance sampling for federated learning. In ICASSP, pages 3095–3099. IEEE, 2021.

Angelo Rodio and Giovanni Neglia. Fedstale: leveraging stale client updates in federated learning. arXiv preprint arXiv:2405.04171, 2024.

Yichen Ruan, Xiaoxi Zhang, Shu-Che Liang, and Carlee Joe-Wong. Towards flexible device participation in federated learning for non-iid data. arXiv preprint arXiv:2006.06954, 2020.

Aishanee Shah, Andrew Hard, Cameron Nguyen, Ignacio Lopez Moreno, Kurt Partridge, Niranjan Subrahmanya, Pai Zhu, and Rajiv Mathews. Training keyword spotting models on non-iid data with federated learning. Interspeech, 2020.

Lin Wang, Yongxin Guo, Tao Lin, and Xiaoying Tang. DELTA: Diverse client sampling for fasting federated learning. In NeurIPS, 2023.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747, 2017.

Cong Xie, Sanmi Koyejo, and Indranil Gupta. Asynchronous federated optimization. arXiv preprint arXiv:1903.03934, 2019.

Haoran Zhang, Zejun Gong, Zekai Li, Marie Siew, Carlee Joe-Wong, and Rachid El-Azouzi. Towards optimal heterogeneous client sampling in multi-model federated learning. arXiv preprint arXiv:2504.05138, 2025.

Hao Zheng, Zhigang Hu, Liu Yang, Meiguang Zheng, Aikun Xu, and Boyu Wang. Confree: Conflict-free client update aggregation for personalized federated learning. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 39, pages 22875–22883, 2025.

FedSteer: Taming Extreme Gradient Staleness in Federated Learning with Corrective Projections and Caching (Appendix)

Haoran Zhang[‡1] **Cainã F. Pereira**[3] **Marie Siew**[§4] **Xutong Liu**[5] **Carlee Joe-Wong**[2] **Rachid El-Azouzi**[3]

[1]Electrical and Computer Engineering Dept., The University of Texas at Austin, Austin, Texas, USA
[2]Electrical and Computer Engineering Dept., Carnegie Mellon University, Pittsburgh, Pennsylvania, USA
[3]CERI/LIA, University of Avignon, Avignon, France
[4]Information Systems Technology and Design Pillar, Singapore University of Technology and Design, Singapore
[5]Computer Science & Systems Dept., University of Washington, Tacoma, Washington, USA

A PROOFS

Assumption 1 (Non-trivial gradients)**.** *For any cached gradient $\mathbf{h}_i^t, i \in \mathcal{X}$, its magnitude is strictly positive, i.e., $\|\mathbf{h}_i^t\|_2 > 0$.*

Assumption 2 (L-smoothness)**.** *Each F_i is L-smooth, and thus $F = \sum_{i\in\mathcal{N}} d_i F_i$ is also L-smooth.*

Assumption 3 (Bounded variance at client-level)**.** *The stochastic gradient at each client is an unbiased estimator of the local gradient: $\mathbb{E}_{\xi_i\sim\mathcal{D}_i}[\nabla F_i(\mathbf{w},\xi_i)] = \nabla F_i(\mathbf{w})$, and its variance is bounded: $Var_{\xi_i\sim\mathcal{D}_i}(F_i(\mathbf{w},\xi_i)) \leq 0$.*

Assumption 4 (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*

$$\|\nabla F_i(\mathbf{w}) - \nabla F(\mathbf{w})\|^2 \leq \sigma_g^2, \quad \forall \mathbf{w}, i.$$

Assumption 5 (Partial and heterogeneous client participation)**.** *In each round t, client i participates with a probability p_i, independently of previous rounds and other clients. p_i is bounded: $p_{min} < p_i < p_{max}$.*

Theorem 2 (Convergence analysis)**.** *Under Assumptions 2-5, $\eta_c \leq \frac{1}{4\sqrt{2E(E-1)}}$, and $\eta_s \leq \frac{1}{2L}$, the iterates $\{\mathbf{w}^t\}$ generated by FedSteer satisfy:*

$$\min_{t\in[1,T]} \mathbb{E}\|\nabla F(\mathbf{w}^t)\|^2 \leq \underbrace{\frac{4(F(\mathbf{w}^1) - F(\mathbf{w}^*))}{\eta_s T}}_{\text{iterate initialization error}} + \underbrace{4\eta_s L \frac{\sum_{t=1}^T \mathbb{E}\|\Delta^t - \mathbb{E}[\Delta^t]\|^2}{T}}_{\text{partial update variance error}}$$

$$+ \underbrace{\Gamma\sigma^2[2\eta_c^2 L^2(E-1) + \frac{1}{TE}]}_{\text{stochastic gradient error}} + \underbrace{(2+\Gamma)8\eta_c^2 L^2 E(E-1)\sigma_g^2}_{\text{error from data heterogeneity}} + \underbrace{\frac{1}{T}\frac{\Gamma(1-p_{min})}{p_{avg}}\frac{1}{N}\sum_{i=1}^N \|\nabla F_i(\mathbf{w}^1) - \mathbf{h}_i^1\|^2}_{\text{memory initialization error}}$$

where $\Gamma = \frac{p_{min}p_{avg}}{\eta_s L(2-p_{min})}$.

To prove Theorem 1, we first formalize the sources of randomness inherent in the training process. **Source of Randomness:** The system's stochasticity arises from two primary sources in each global round t.

- **Client sampling:** A subset of clients, denoted by the random set $\mathcal{A}_t$ is selected to participate in the training. This selection is determined by the sampling distribution $\mathbf{p} = \{p_i\}_{i\in\mathcal{N}}$.

*The work was performed while the author was affiliated with Carnegie Mellon University.
†Corresponding author (marie_siew@sutd.edu.sg).
‡The work was performed while the author was affiliated with Carnegie Mellon University.
§Corresponding author (marie_siew@sutd.edu.sg).

- **Data sampling:** Each selected client $i \in \mathcal{A}_t$ computes stochastic gradients using mini-batches of data randomly sampled from its local dataset.

To formalize our analysis, we define the random variables corresponding to these processes. Let $\xi_i^{(t,k)}$ denote the mini-batch of data points sampled by client i at its k-th local step during round t.

To track the evolution of randomness across multiple rounds, we use the following notation:

- $\mathbb{1}_{i \in \mathcal{A}_t}$ is the indicator of client participation.
- $\mathcal{A}^{(s:q)} = \{\mathcal{A}_s, \mathcal{A}_{s+1}, \ldots, \mathcal{A}_q\}$ represents the sequence of client sets sampled from round s to round q.
- $\xi_i^t = \{\xi_i^{t,k}\}_{k=0}^{K-1}$ denotes the collection of all data mini-batches sampled by client i during its local training in round t.
- $\xi^t = \{\xi_i^t\}_{i \in \mathcal{A}_t}$ denotes the collection of all data mini-batches sampled by active client i at round t.
- $\xi^{(s:q)} = \{\xi^s, \ldots, \xi^q\}$ denotes the collection of all data mini-batches sampled by active clients from round s to round q.

The stochastic progression of the algorithm from the first round to the current round t can be expressed as:

$$\mathcal{H}^t = \{\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_{t-1}, \xi^1, \xi^2, \ldots, \xi^{t-1}\}. \tag{11}$$

We define the following pseudo-gradients for the proof:

$$\text{Local stochastic pseudo-gradient: } g_i^t = \frac{1}{E}\sum_{e=1}^{E-1} \nabla F_i(w_i^{t,e}, \xi_i^{t,e}), \tag{12}$$

$$\text{Local pseudo-gradient: } \overline{g}_i^t = \frac{1}{E}\sum_{e=1}^{E-1} \nabla F_i(w_i^{t,e}), \tag{13}$$

$$\text{Global stochastic pseudo-gradient: } g^t = \sum_{i \in \mathcal{N}} d_i g_i^t, \tag{14}$$

$$\text{Global pseudo-gradient: } \overline{g}^t = \sum_{i \in \mathcal{N}} d_i \overline{g}_i^t, \tag{15}$$

$$\text{Global stale pseudo-gradient: } h^t = \sum_{i \in \mathcal{N}} d_i h_i^t, \tag{16}$$

$$\text{Global estimate pseudo-gradient: } \hat{g}^t = \sum_{i \in \mathcal{N}} d_i \hat{g}_i^t, \tag{17}$$

where E is the number of local epochs, following the same definition of the paper. In the proof, we also use K to denote the number of local epochs, in the case where E may be misunderstood as the compute of expectation.

Lemma 1 (Descent lemma)**.** *Let $F : \mathbb{R}^d \to \mathbb{R}$ be an L-smooth function, optimized via the sequence of parameters $\{w^t\}$. At each iteration t, an SGD update is made according to a learning rate η_s and a stochastic gradient Δ^t. Let $\mathbb{E}_{\mathcal{A}_t, \xi^t | \mathcal{H}^t}[\Delta^t] = \bar{g}^t$. Then, the expected reduction in F after one iteration is bounded by:*

$$\begin{aligned}\mathbb{E}_{\mathcal{A}_t, \xi^t | \mathcal{H}^t}[F(w^{t+1})] \leq F(w^t) - \frac{\eta_s}{2}\big[\|\nabla F(w^t)\|^2 + \|\bar{g}^t\|^2 \\ -\|\bar{g}^t - \nabla F(w^t)\|^2\big] + \frac{\eta_s^2 L}{2}\mathbb{E}_{\mathcal{A}_t, \xi^t | \mathcal{H}^t}\|\Delta^t\|^2.\end{aligned} \tag{18}$$

Proof. By the L-smoothness of F, it follows that:

$$\begin{aligned}F(w^{t+1}) &\leq F(w^t) + \langle \nabla F(w^t), w^{t+1} - w^t \rangle + \frac{L}{2}\|w^{t+1} - w^t\|^2 \\ &\leq F(w^t) - \eta_s \langle \nabla F(w^t), \Delta^t \rangle + \frac{\eta_s^2 L}{2}\|\Delta^t\|^2,\end{aligned} \tag{19}$$

where Eq. (19) applies the update rule $w^{t+1} = w^t - \eta_s \Delta^t$.

Taking the expectation over the randomness at the t-th round, due to client participation (inherent in ξ^t) and stochastic gradients (inherent in $\xi^t := \{\xi_i^{(t,k)}\}_{i,k}$), yields:

$$\begin{aligned}
\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}[F(w^{t+1})] &\leq F(w^t) - \eta_s \mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\langle \nabla F(w^t), \Delta^t\rangle + \frac{\eta_s^2 L}{2}\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\|\Delta^t\|^2 \\
&\leq F(w^t) - \eta_s \left[\langle \nabla F(w^t), \bar{g}^t\rangle\right] + \frac{\eta_s^2 L}{2}\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\|\Delta^t\|^2 \\
&\leq F(w^t) - \frac{\eta_s}{2}\left[\|\nabla F(w^t)\|^2 + \|\bar{g}^t\|^2\right. \\
&\quad \left. -\|\bar{g}^t - \nabla F(w^t)\|^2\right] + \frac{\eta_s^2 L}{2}\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\|\Delta^t\|^2, \qquad (20)
\end{aligned}$$

where the second inequality uses $\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}[\Delta^t] = \bar{g}^t$ and the third inequality applies the identity $\|a-b\|^2 = \|a\|^2 + \|b\|^2 - 2\langle a, b\rangle$. □

Lemma 2 (Expected value of the local stochastic pseudo-gradients)**.** *If the stochastic gradients are unbiased (Assumption 3), the following identity holds:*

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left[g_i^t\right] = \bar{g}_i^t. \qquad (21)$$

Proof. We decompose the expected value of the gradient $\nabla F_i(w_i^{(t,k)}, \xi_i^{(t,k)})$ as:

$$\mathbb{E}_{\xi_i^{(t,0:k)}|\mathcal{H}^t}\left[\nabla F_i(w_i^{(t,k)}, \xi_i^{(t,k)})\right] = \mathbb{E}_{\xi_i^{(t,0:k-1)}|\mathcal{H}^t}\left[\mathbb{E}_{\xi_i^{(t,k)}|\xi_i^{(t,0:k-1)},H^t}\left[\nabla F_i(w_i^{(t,k)}, \xi_i^{(t,k)})\right]\right]. \qquad (22)$$

We finally use Assumption 3 to conclude that $\mathbb{E}_{\xi_i^{(t,k)}|\xi_i^{(t,0:k-1)},\mathcal{H}^t}\left[\nabla F_i(w_i^{(t,k)}, \xi_i^{(t,k)})\right] = \nabla F_i(w_i^{(t,k)})$.

Below, we present the detailed derivations of the proof.

$$\begin{aligned}
\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left[g_i^t\right] &= \frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left[\nabla F_i(w_i^{(t,k)}, \xi_i^{(t,k)})\right] && (23)\\
&= \frac{1}{K}\mathbb{E}_{\xi_i^{(t,0)}|\mathcal{H}^t}\left[\nabla F_i(w^t, \xi_i^{(t,0)})\right] + \frac{1}{K}\mathbb{E}_{\xi_i^{(t,0)},\xi_i^{(t,1)}|\mathcal{H}^t}\left[\nabla F_i(w_i^{(t,1)}, \xi_i^{(t,1)})\right] + \cdots && (24)\\
&\quad + \frac{1}{K}\mathbb{E}_{\xi_i^{(t,0:K-1)}|\mathcal{H}^t}\left[\nabla F_i(w_i^{(t,K-1)}, \xi_i^{(t,K-1)})\right] && (25)\\
&= \frac{1}{K}\nabla F_i(w^t) + \frac{1}{K}\mathbb{E}_{\xi_i^{(t,0)}|\mathcal{H}^t}\left[\mathbb{E}_{\xi_i^{(t,1)}|\xi_i^{(t,0)},\mathcal{H}^t}\left[\nabla F_i(w_i^{(t,1)}, \xi_i^{(t,1)})\right]\right] + \cdots && (26)\\
&\quad + \frac{1}{K}\mathbb{E}_{\xi_i^{(t,0:K-2)}|\mathcal{H}^t}\left[\mathbb{E}_{\xi_i^{(t,K-1)}|\xi_i^{(t,0:K-2)},\mathcal{H}^t}\left[\nabla F_i(w_i^{(t,K-1)}, \xi_i^{(t,K-1)})\right]\right] && (27)\\
&= \frac{1}{K}\left[\nabla F_i(w^t) + \mathbb{E}_{\xi_i^{(t,0)}|\mathcal{H}^t}\left[\nabla F_i(w_i^{(t,1)})\right] + \cdots + \mathbb{E}_{\xi_i^{(t,0:K-2)}|\mathcal{H}^t}\left[\nabla F_i(w_i^{(t,K-1)})\right]\right] && (28)\\
&= \frac{1}{K}\sum_{k=0}^{K-1}\nabla F_i(w_i^{(t,k)}) = \bar{g}_i^t, && (29)
\end{aligned}$$

where Eq. (23) uses the definition of g_i^t, Eq. (25) makes explicit the dependency of the iterate $w_i^{(t,k)}$ on the random batches $\xi_i^{(t,0:k-1)}$, Eq. (27) uses the law of total expectation given in (22), Eq. (28) applies the unbiasedness of the stochastic gradient (Assumption 3), and Eq. (29) uses the definition of $\bar{g}_i^t$. □

Lemma 3 (Variance of the local stochastic pseudo-gradients)**.** *If the variance of the local stochastic gradients is bounded by σ^2 (Assumption 3), the following inequality holds:*

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|g_i^t - \bar{g}_i^t\right\|^2 \leq \frac{\sigma^2}{K}. \qquad (30)$$

Proof. The proof builds on similar observations to those presented in Lemma 2, but additionally relies on the bounded variance of local stochastic gradients (Assumption 3).

Below, the detailed derivations.

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\|g_i^t - \bar{g}_i^t\|^2 \tag{31}$$

$$= \mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\frac{1}{K}\sum_{k=0}^{K-1}\left[\nabla F_i(\mathbf{w}_t^{(t,k)}, \xi_t^{(k)}) - \nabla F_i(\mathbf{w}_t)\right]^2\right\| \tag{32}$$

$$= \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\nabla F_i(\mathbf{w}_i^{(t,k)}, \xi_i^{(t,k)}) - \nabla F_i(\mathbf{w}_i^{(t,k)})\right\|^2 \tag{33}$$

$$+ \frac{1}{K^2}\sum_{k=0}^{K-1}\sum_{\substack{k'=0\\k'\neq k}}^{K-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\langle\nabla F_i(w_i^{(t,k)}, \xi_i^{(t,k)}) - \nabla F_i(w_i^{(t,k)}), \nabla F_i(w_i^{(t,k')}, \xi_i^{(t,k')}) - \nabla F_i(w_i^{(t,k')})\right\rangle, \tag{34}$$

where Eq. (32) applies the definitions for $\mathbf{g}^t$ and $\mathbf{g}^{(t,0)}$, and Eq. (34) expands the squared norm. To show that the second term in (34) is zero, we use the law of total expectation in a similar way as in (22). Indeed, denote $k'' = \max\{k, k'\}$. The following relation holds:

$$\mathbb{E}_{\xi_i^{(t,0:k'')}|\mathcal{H}^t}\left[\nabla F_i\left(\mathbf{w}_i^{(t,k'')}, \xi_i^{(t,k'')}\right) - \nabla F_i\left(\mathbf{w}_i^{(t,k'')}\right)\right] \tag{35}$$

$$= \mathbb{E}_{\xi_i^{(t,0:k''-1)}|\mathcal{H}^t}\left[\underbrace{\mathbb{E}_{\xi_i^{(t,k'')}|\xi_i^{(t,0:k''-1)},\mathcal{H}^t}\left[\nabla F_i\left(\mathbf{w}_i^{(t,k'')}, \xi_i^{(t,k'')}\right) - \nabla F_i\left(\mathbf{w}_i^{(t,k'')}\right)\right]}_{=0 \text{ by Assumption 3}}\right] \tag{36}$$

$$= 0 \tag{37}$$

Therefore, only the first term remains:

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\|g_i^t - \bar{g}_i^t\|^2 \tag{38}$$

$$= \frac{1}{K^2}\sum_{k=0}^{K-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\nabla F_i\left(\mathbf{w}_i^{(t,k)}, \xi_i^{(t,k)}\right) - \nabla F_i\left(\mathbf{w}_i^{(t,k)}\right)\right\|^2 \tag{39}$$

$$= \frac{1}{K^2}\Bigg[\mathbb{E}_{\xi_i^{(t,0)}}\|\nabla F_i(w^t, \xi_i^{(t,0)}) - \nabla F_i(w^t)\|^2 + \cdots \tag{40}$$

$$\mathbb{E}_{\xi_i^{(t,0:K-1)}|\mathcal{H}^t}\|\nabla F_i(w_i^{(t,K-1)}, \xi_i^{(t,K-1)}) - \nabla F_i(w_i^{(t,K-1)})\|^2\Bigg] \tag{41}$$

$$\leq \frac{1}{K^2}\sum_{k=1}^{K-1}\sigma^2 = \frac{\sigma^2}{K} \tag{42}$$

□

Lemma 4 (Variance of the global stochastic pseudo-gradient)**.** *Assuming that client participation outcomes ξ_i^t are decided by $\{p_i^t\}$, and that the variance of the local stochastic gradients is bounded by σ^2 (Assumption 3), the following inequality holds:*

$$\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^t}{p_i^t}\left(g_i^t - \bar{g}_i^t\right)\right\|^2 \leq \left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i^t}\right)\frac{\sigma^2}{NK}. \tag{43}$$

Proof. 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:

$$\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^t}{p_i^t}\left(g_i^t-\bar{g}_i^t\right)\right\|^2 \tag{44}$$

$$=\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left[\frac{1}{N^2}\sum_{i=1}^{N}\frac{(\xi_i^t)^2}{(p_i^t)^2}\left\|g_i^t-\bar{g}_i^t\right\|^2+\frac{1}{N^2}\sum_{i=1}^{N}\sum_{\substack{i'=1\\i'\neq i}}^{N}\frac{\xi_i^t\xi_{i'}^t}{p_i^tp_{i'}^t}\left\langle g_i^t-\bar{g}_i^t,g_{i'}^t-\bar{g}_{i'}^t\right\rangle\right] \tag{45}$$

We leverage the linearity of expectation, the independence of client participation (ξ^t) and batch sampling among clients (ξ_i^t and $\xi_{i'}^t$), and Lemma 2 to show that:

$$\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left[\frac{\xi_i^t\xi_{i'}^t}{p_i^tp_{i'}^t}\left\langle g_i^t-\bar{g}_i^t,g_{i'}^t-\bar{g}_{i'}^t\right\rangle\right] \tag{46}$$

$$=\frac{\mathbb{E}_{\xi^t|\mathcal{H}^t}[\xi_i^t\xi_{i'}^t]}{p_i^tp_{i'}^t}\mathbb{E}_{\xi^t|\mathcal{H}^t}\left[\left\langle g_i^t-\bar{g}_i^t,g_{i'}^t-\bar{g}_{i'}^t\right\rangle\right] \tag{47}$$

$$=\frac{\mathbb{E}_{\xi^t|\mathcal{H}^t}\left[\xi_i^t\xi_{i'}^t\right]}{p_i^tp_{i'}^t}\left\langle\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left[g_i^t-\bar{g}_i^t\right],\mathbb{E}_{\xi_{i'}^t|\mathcal{H}^t}\left[g_{i'}^t-\bar{g}_{i'}^t\right]\right\rangle=0, \tag{48}$$

Finally, we bound the remaining term using Lemma 3:

$$\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left\|\frac{1}{N}\sum_{i=1}^{N}\frac{\xi_i^t}{p_i}\left(g_i^t-\bar{g}_i^t\right)\right\|^2=\frac{1}{N^2}\sum_{i=1}^{N}\frac{\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left[\left(\xi_i^t\right)^2\right]}{(p_i^t)^2}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|g_i^t-\bar{g}_i^t\right\|^2 \tag{49}$$

$$\leq\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{p_i^t}\right)\frac{\sigma^2}{NK} \tag{50}$$

□

Lemma 5 (Client drift due to multiple local iterations)**.** *Under bounded local stochastic gradient variance (σ^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 ($\bar{g}_i^t$) from its local gradient ($\nabla F_i(w^t)$) is bounded as:*

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{g}_i^t-\nabla F_i(w^t)\right\|^2\leq 2\eta_c^2L^2K(K-1)\left[\frac{\sigma^2}{K}+2\left\|\nabla F_i(w^t)\right\|^2\right] \tag{51}$$

Additionally, if the variance of local gradients is uniformly bounded across clients (by σ_g^2, as per Assumption 4):

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{g}_i^t-\nabla F_i(w^t)\right\|^2\leq 2\eta_c^2L^2K(K-1)\left[\frac{\sigma^2}{K}+4\sigma_g^2+4\left\|\nabla F(w^t)\right\|^2\right]. \tag{52}$$

The bound in Eq. (56) captures that, when the number of local iterations K equals 1, $\bar{g}_i^t$ and $\nabla F_i(\mathbf{w}^t)$ become equivalent.

Proof.

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{g}_i^t - \nabla F_i(w^t)\right\|^2 = \mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\frac{1}{K}\sum_{k=0}^{K-1}\left(\nabla F_i(w_i^{(t,k)}) - \nabla F_i(w^t)\right)\right\|^2 \tag{53}$$

$$\leq \frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\nabla F_i(w_i^{(t,k)}) - \nabla F_i(w^t)\right\|^2 \tag{54}$$

$$\leq \frac{L^2}{K}\sum_{k=0}^{K-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|w_i^{(t,k)} - w^t\right\|^2 \tag{55}$$

The above inequalities use Jensen's inequality and L-smoothness assumption. Next, the individual difference is bounded as:

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\|w_i^{(t,k)} - w^t\|^2 \tag{56}$$

$$= \eta_c^2\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\|\sum_{k'=0}^{k-1}\nabla F_i(w_i^{(t,k')}, \xi_i^{(t,k')})\|^2 \tag{57}$$

$$= \eta_c^2\left[\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\|\sum_{k'=0}^{k-1}[\nabla F_i(w_i^{(t,k')}, \xi_i^{(t,k')}) - \nabla F_i(w_i^{(t,k')})]\|^2 + \mathbb{E}_{\xi_i^t|\mathcal{H}^t}\|\sum_{k'=0}^{k-1}\nabla F_i(w_i^{(t,k')})\|^2\right] \tag{58}$$

$$\leq \eta_c^2\left[\sum_{k'=0}^{k-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\nabla F_i\left(w_i^{(t,k')}, \xi_i^{(t,k')}\right) - \nabla F_i\left(w_i^{(t,k')}\right)\right\|^2 + k\sum_{k'=0}^{k-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\nabla F_i\left(w_i^{(t,k')}\right)\right\|^2\right] \tag{59}$$

$$\leq \eta_c^2\left[k\sigma^2 + k\sum_{k'=0}^{k-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\nabla F_i\left(w_i^{(t,k')}\right) - \nabla F_i\left(w^t\right) + \nabla F_i\left(w^t\right)\right\|^2\right] \tag{60}$$

$$\leq \eta_c^2\left[k\sigma^2 + 2k\sum_{k'=0}^{k-1}\left[L^2\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|w_i^{(t,k')} - w^t\right\|^2 + \left\|\nabla F_i(w^t)\right\|^2\right]\right] \tag{61}$$

where Eq. (57) applies the local update rule, Eq. (58) leverages the local stochastic gradient unbiasedness (as per Lemma 2) and its bias-variance decomposition. Eq. (59) involves squaring the former term, zeroing the cross terms, and applying Jensen's inequality to the latter term. Eq. (60) accounts for the bounded variance of local stochastic gradients in the former term (Lemma 3), and modifies the latter term by adding and subtracting the initial local gradient ($\nabla F_i(w^t)$); finally Eq. (61) uses the norm inequality and the L-smoothness of local objectives.

Summing over $k = 0, \ldots, K-1$, it yields:

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|w_i^{(t,k)} - w^t\right\|^2 \tag{62}$$

$$\leq \frac{\eta_c^2\sigma^2}{K}\sum_{k=0}^{K-1}k + \frac{2\eta_c^2L^2}{K}\sum_{k=0}^{K-1}k\sum_{k'=0}^{k-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|w_i^{(t,k')} - w^t\right\|^2 + \frac{2\eta_c^2}{K}\sum_{k=0}^{K-1}k\sum_{k'=0}^{k-1}\left\|\nabla F_i(w^t)\right\|^2 \tag{63}$$

$$\leq \eta_c^2(K-1)\sigma^2 + 2\eta_c^2L^2K(K-1)\left[\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|w_i^{(t,k)} - w^t\right\|^2\right] + 2\eta_c^2K(K-1)\left\|\nabla F_i(w^t)\right\|^2, \tag{64}$$

where Eq. (64) uses $\sum_{k'=0}^{k-1}\|w_i^{(t,k')} - w^t\|^2 \leq \sum_{k=0}^{K-1}\|w_i^{(t,k)} - w^t\|^2$, and $\sum_{k=0}^{K-1}k = \frac{1}{2}(K-1)K$.
Define $D := 2\eta_c^2L^2K(K-1)$. Choose η_c small enough such that $D \leq \frac{1}{2}$ so $\eta_c \leq \frac{1}{2LK}$. Rearranging the terms:

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|w_i^{(t,k)} - w^t\right\|^2 \leq \frac{\eta_c^2(K-1)\sigma^2}{1-D} + \frac{2\eta_c^2K(K-1)}{1-D}\left\|\nabla F_i(w^t)\right\|^2 \tag{65}$$

Substituting (65) back to (55):

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{g}_i^t - \nabla F_i(w^t)\right\|^2 \leq \frac{D}{2(1-D)}\frac{\sigma^2}{K} + \frac{D}{1-D}\left\|\nabla F_i(w^t)\right\|^2 \tag{66}$$

$$\leq D\frac{\sigma^2}{K} + 2D\left\|\nabla F_i(w^t)\right\|^2, \tag{67}$$

where Eq. (67) uses $D \leq \frac{1}{2}$. Replacing $D := 2\eta_c^2 L^2 K(K-1)$ into (67) completes the proof of Inequality (51). Additionally, inequality (52) removes the dependency on $\nabla F_i(w^t)$ by adding and subtracting $\nabla F(w^t)$ in the squared norm:

$$\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{g}_i^t - \nabla F_i(w^t)\right\|^2 \leq D\frac{\sigma^2}{K} + 2D\left\|\nabla F_i(w^t) - \nabla F(w^t) + \nabla F(w^t)\right\|^2 \tag{68}$$

$$\leq D\frac{\sigma^2}{K} + 4D\left\|\nabla F_i(w^t) - \nabla F(w^t)\right\|^2 + 4D\left\|\nabla F(w^t)\right\|^2 \tag{69}$$

$$\leq D\frac{\sigma^2}{K} + 4D\sigma_g^2 + 4D\left\|\nabla F(w^t)\right\|^2, \tag{70}$$

Replacing $D := 2\eta_c^2 L^2 K(K-1)$ into (70) concludes the proof of inequality (52). □

Lemma 6 (Bound on the memory term)**.** *Define H^t as the divergence between the local gradient and the historical pseudo-gradient at time t:*

$$H^t = \frac{1}{N}\sum_{i=1}^{N}\|\nabla F_i(w^{t-1}) - h_i^t\|^2 \tag{71}$$

The expected historical error H^{t+1} is recursively bounded as:

$$\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left[H^{t+1}\right] \leq \left(\frac{1}{N}\sum_{i=1}^{N}p_i^t\right)\frac{\sigma^2}{K} + \frac{1}{N}\sum_{i=1}^{N}p_i^t\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{g}_i^t - \nabla F_i(w^t)\right\|^2 \tag{72}$$

$$+ \eta_s^2 L^2\left(1+\frac{1}{C}\right)\left(1 - \frac{1}{N}\sum_{i=1}^{N}p_i^t\right)\left\|\Delta^{t-1}\right\|^2 + (1+C)(1-p_{min})H^t \tag{73}$$

Proof. the proof starts by definition of H^{t+1}:

$$\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left[H^{t+1}\right] \tag{74}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left[\mathbb{E}_{\xi_i^t|\xi_i^t,\mathcal{H}^t}\left\|\nabla F_i(\mathbf{w}^t) - \mathbf{h}_i^{t+1}\right\|^2\right] \tag{75}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\left[p_i^t\,\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\nabla F_i(\mathbf{w}^t) - \mathbf{g}_i^t\right\|^2 + (1-p_i^t)\left\|\nabla F_i(\mathbf{w}^t) - \mathbf{h}_i^t\right\|^2\right] \tag{76}$$

$$\leq \frac{1}{N}\sum_{i=1}^{N}p_i^t\,\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\mathbf{g}_i^t - \bar{\mathbf{g}}_i^t\right\|^2 + \frac{1}{N}\sum_{i=1}^{N}p_i^t\,\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{\mathbf{g}}_i^t - \nabla F_i(\mathbf{w}^t)\right\|^2 \tag{77}$$

$$+ \frac{\left(1+\frac{1}{C}\right)}{N}\sum_{i=1}^{N}(1-p_i^t)\left\|\nabla F_i(\mathbf{w}^t) - \nabla F_i(\mathbf{w}^{t-1})\right\|^2 + \frac{(1+C)}{N}\sum_{i=1}^{N}(1-p_i^t)\left\|\nabla F_i(\mathbf{w}^{t-1}) - \mathbf{h}_i^t\right\|^2 \tag{78}$$

$$\leq \left(\frac{1}{N}\sum_{i=1}^{N}p_i^t\right)\frac{\sigma^2}{K} + \frac{1}{N}\sum_{i=1}^{N}p_i^t\,\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{\mathbf{g}}_i^t - \nabla F_i(\mathbf{w}^t)\right\|^2 \tag{79}$$

$$+ \frac{\eta_s^2 L^2\left(1+\frac{1}{C}\right)}{N}\sum_{i=1}^{N}(1-p_i^t)\left\|\Delta^{t-1}\right\|^2 + \frac{(1+C)}{N}\sum_{i=1}^{N}(1-p_i^t)\left\|\nabla F_i(\mathbf{w}^{t-1}) - \mathbf{h}_i^t\right\|^2, \tag{80}$$

where Eq. (75) uses the law of total expectation to separate expectations on client participation (ξ_i^t) and batch sampling (ξ_i^t); Eq. (76) solves the inner expectation with respect to client participation; Eq. (78) manipulates the first term by adding

and subtracting $\bar{g}_i^t$, then leverages the bounded variance of the local stochastic pseudo-gradients, and similarly corrects the second term with $\nabla F_i(w^{t-1})$, then applies the norm inequality $\|a+b\|^2 \leq (1+\frac{1}{C})\|a\|^2 + (1+C)\|b\|^2$ for any positive C; Eq. (80) is derived from the L-smoothness property of local objectives.

The final expression in Eq. (73) 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 7 (Bound on the memory term - Initial condition)**.**

$$\mathbb{E}_{\mathbb{1}^1,\xi^1}\left[H^2\right] \leq \left(\frac{1}{N}\sum_{i=1}^N p_i^t\right)\frac{\sigma^2}{K} + \frac{1}{N}\sum_{i=1}^N p_i^t\,\mathbb{E}_{\xi_i^1}\left\|\nabla F_i(\mathbf{w}^1) - \bar{\mathbf{g}}_i^1\right\|^2 \tag{81}$$

$$+ (1-p_{\min})\frac{1}{N}\sum_{i=1}^N \left\|\nabla F_i(\mathbf{w}^1) - \mathbf{h}_i^1\right\|^2 \tag{82}$$

Proof. The proof starts with the definition of H^2:

$$\mathbb{E}_{1^1,\xi^1}\left[H^2\right] \tag{83}$$

$$= \frac{1}{N}\sum_{i=1}^N \mathbb{E}_{\xi_i^1}\left[\mathbb{E}_{\xi_i^1|\xi_i^1}\left\|\nabla F_i(\mathbf{w}^1) - \mathbf{h}_i^2\right\|^2\right] \tag{84}$$

$$= \frac{1}{N}\sum_{i=1}^N \left[p_i^t\,\mathbb{E}_{\xi_i^1}\left\|\nabla F_i(\mathbf{w}^1) - \mathbf{g}_i^1\right\|^2 + (1-p_i^t)\left\|\nabla F_i(\mathbf{w}^1) - \mathbf{h}_i^1\right\|^2\right] \tag{85}$$

$$= \frac{1}{N}\sum_{i=1}^N p_i^t\,\mathbb{E}_{\xi_i^1}\left\|\mathbf{g}_i^1 - \bar{\mathbf{g}}_i^1\right\|^2 + \frac{1}{N}\sum_{i=1}^N p_i^t\,\mathbb{E}_{\xi_i^1}\left\|\nabla F_i(\mathbf{w}^1) - \bar{\mathbf{g}}_i^1\right\|^2 + \frac{1}{N}\sum_{i=1}^N (1-p_i^t)\left\|\nabla F_i(\mathbf{w}^1) - \mathbf{h}_i^1\right\|^2 \tag{86}$$

where Eq. (84) uses the law of total expectation to separate expectations; Eq. (85) solves the inner expectation with respect to client participation; Eq. (86) adds and subtracts $\bar{g}_i^1$ to the first term, then leverages the local pseudo-gradients' unbiased property to separate the two components. □

Per Round Progress

Define Lyapunov function, for any $\frac{\eta_s^2 L}{2} < \delta < \frac{\eta_s}{2}$ and $\alpha \geq 0$:

$$\psi^{t+1} = F(w^{t+1}) + (\delta - \frac{\eta_s^2 L}{2})\|\Delta^t\|^2 + \alpha \underbrace{\frac{1}{N}\sum_{i=1}^N \|\nabla F_i(w^t) - h_i^{t+1}\|^2}_{H^{t+1}} \tag{87}$$

Considering expectation over the randomness at the t-th round and invoking the standard descent lemma for smooth

objective:

$$\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left[\psi^{t+1}\right] \tag{88}$$

$$= \mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left[F(w^{t+1}) + \left(\delta - \frac{\eta_s^2 L}{2}\right)\left\|\Delta^t\right\|^2 + \frac{\alpha}{N}\sum_{i=1}^N \left\|\nabla F_i(w^t) - h_i^{t+1}\right\|^2\right] \tag{89}$$

$$\leq F(w^t) - \frac{\eta_s}{2}\left\|\nabla F(w^t)\right\|^2 - \frac{\eta_s}{2}\,\mathbb{E}_{\xi^t|\mathcal{H}^t}\left\|\bar{g}^t\right\|^2 + \frac{\eta_s}{2}\,\mathbb{E}_{\xi^t|\mathcal{H}^t}\left\|\bar{g}^t - \nabla F(w^t)\right\|^2 \tag{90}$$

$$+ \frac{\eta_s^2 L}{2}\,\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left\|\Delta^t\right\|^2 + \left(\delta - \frac{\eta_s^2 L}{2}\right)\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left\|\Delta^t\right\|^2 + \frac{\alpha}{N}\sum_{i=1}^N \mathbb{E}_{\mathcal{A}_t,\xi_i^t|\mathcal{H}^t}\left\|\nabla F_i(w^t) - h_i^{t+1}\right\|^2 \tag{91}$$

$$\leq F(w^t) - \frac{\eta_s}{2}\left\|\nabla F(w^t)\right\|^2 - \frac{\eta_s}{2}\,\mathbb{E}_{\xi^t|\mathcal{H}^t}\left\|\bar{g}^t\right\|^2 + \frac{\eta_s}{2N}\sum_{i=1}^N \mathbb{1}_{i\in\mathcal{A}_t}\,\mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{g}_i^t - \nabla F_i(w^t)\right\|^2 \tag{92}$$

$$+ \delta\,\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left\|\Delta^t - \bar{g}^t + \bar{g}^t\right\|^2 + \frac{\alpha}{N}\sum_{i=1}^N \mathbb{E}_{\mathcal{A}_t,\xi_i^t|\mathcal{H}^t}\left\|\nabla F_i(w^t) - h_i^{t+1}\right\|^2 \tag{93}$$

$$\leq F(w^t) - \frac{\eta_s}{2}\left\|\nabla F(w^t)\right\|^2 + \left(\delta - \frac{\eta_s}{2}\right)\mathbb{E}_{\xi^t|\mathcal{H}^t}\left\|\bar{g}^t\right\|^2 + \frac{\eta_s}{2N}\sum_{i=1}^N \mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{g}_i^t - \nabla F_i(w^t)\right\|^2 \tag{94}$$

$$+ \delta\,\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left\|\Delta^t - \bar{g}^t\right\|^2 + \alpha\,\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left[\frac{1}{N}\sum_{i=1}^N \left\|\nabla F_i(w^t) - h_i^{t+1}\right\|^2\right] \tag{95}$$

$$\leq F(w^t) - \frac{\eta_s}{2}\left\|\nabla F(w^t)\right\|^2 + \frac{\eta_s}{2N}\sum_{i=1}^N \mathbb{E}_{\xi_i^t|\mathcal{H}^t}\left\|\bar{g}_i^t - \nabla F_i(w^t)\right\|^2 \tag{96}$$

$$+ \delta\,\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left\|\Delta^t - \bar{g}^t\right\|^2 + \alpha\,\mathbb{E}_{\mathcal{A}_t,\xi^t|\mathcal{H}^t}\left[H^{t+1}\right] \tag{97}$$

We simplify some notations:

$$\mathbb{E}[\psi^{t+1}] \leq F(w^t) - \frac{\eta_s}{2}\|\nabla F(w^t)\|^2 + \frac{\eta_s}{2N}\sum_{i=1}^N \mathbb{E}\|\bar{g}_i^t - \nabla F_i(w^t)\|^2 \tag{98}$$

$$+ \delta\underbrace{\mathbb{E}\|\Delta^t - \overline{g}^t\|^2}_{\text{variance of update}} + \alpha\mathbb{E}[H^{t+1}] \tag{99}$$

Next we apply Lemma 6,

$$\mathbb{E}[\psi^{t+1}] \leq F(w^t) - \frac{\eta_s}{2}\|\nabla F(w^t)\|^2 + \frac{\eta_s}{2N}\sum_{i=1}^N \mathbb{E}\|\bar{g}_i^t - \nabla F_i(w^t)\|^2 + \delta\mathbb{E}\|\Delta^t - \overline{g}^t\|^2 \tag{100}$$

$$+ \alpha[p_{avg}\frac{\sigma^2}{K} + \frac{1}{N}\sum_{i=1}^N p_i^t\mathbb{E}\|\bar{g}_i^t - \nabla F_i(w^t)\|^2 \tag{101}$$

$$+ \eta_s^2 L^2(1 + \frac{1}{C})(1 - p_{avg})\|\Delta^{t-1}\|^2 + (1 + C)(1 - p_{min})H^t] \tag{102}$$

$$= F(w^t) \tag{103}$$

$$+ \alpha\eta_s^2 L^2(1 + \frac{1}{C})(1 - p_{avg})\|\Delta^{t-1}\|^2 + \alpha(1 + C)(1 - p_{min})H^t \tag{104}$$

$$- \frac{\eta_s}{2}\|\nabla F(w^t)\|^2 + \frac{\eta_s}{2N}\sum_{i=1}^N \mathbb{E}\|\bar{g}_i^t - \nabla F_i(w^t)\|^2 + \delta\mathbb{E}\|\Delta^t - \overline{g}^t\|^2 \tag{105}$$

$$+ \alpha p_{avg}\frac{\sigma^2}{K} + \frac{\alpha}{N}\sum_{i=1}^N p_i^t\mathbb{E}\|\bar{g}_i^t - \nabla F_i(w^t)\|^2 \tag{106}$$

For bounding within the Lyapunov recursive framework, the conditions for this recursion step are:

$$\alpha\eta_s^2L^2(1+\frac{1}{C})(1-p_{avg}) \leq \delta - \frac{\eta_s^2 L}{2} \tag{107}$$

$$\alpha(1+C)(1-p_{min}) \leq \alpha \tag{108}$$

From Eq. (108), we know

$$C \leq \frac{p_{min}}{1-p_{min}} \tag{109}$$

We select $C = \frac{p_{min}}{2(1-p_{min})}$. From Eq. (107), we can have many selections of α, δ. For the convenience of writing, we select $\delta = \eta_s^2 L$. Then, α needs:

$$\alpha \leq \frac{1}{2L(1+\frac{1}{C})(1-p_{avg})} = \frac{p_{min}}{2L(2-p_{min})(1-p_{avg})} \tag{110}$$

We select $\alpha = \frac{p_{min}}{4L(2-p_{min})}$. Using the values of δ and α we select, Eq. (132) can be written as:

$$\mathbb{E}[\psi^{t+1}] \leq \psi^t - \frac{\eta_s}{2}\|\nabla F(w^t)\|^2 \tag{111}$$

$$+ \frac{\eta_s}{2N}\sum_{i=1}^{N}\mathbb{E}\|\overline{g}_i^t - \nabla F_i(w^t)\|^2 + \delta\mathbb{E}\|\Delta^t - \overline{g}^t\|^2 \tag{112}$$

$$+ \alpha p_{avg}\frac{\sigma^2}{K} + \frac{\alpha}{N}\sum_{i=1}^{N}p_i^t\mathbb{E}\|\overline{g}_i^t - \nabla F_i(w^t)\|^2 \tag{113}$$

$$= \psi^t - \frac{\eta_s}{2}\|\nabla F(w^t)\|^2 \tag{114}$$

$$+ \frac{\eta_s}{2N}\sum_{i=1}^{N}\mathbb{E}\|\overline{g}_i^t - \nabla F_i(w^t)\|^2 + \eta_s^2 L\mathbb{E}\|\Delta^t - \overline{g}^t\|^2 \tag{115}$$

$$+ \frac{p_{min}p_{avg}\sigma^2}{4L(2-p_{min})K} \tag{116}$$

$$+ \frac{p_{min}}{4L(2-p_{min})N}\sum_{i=1}^{N}p_i^t\mathbb{E}\|\overline{g}_i^t - \nabla F_i(w^t)\|^2 \tag{117}$$

Use Lemma 5 to bound $\mathbb{E}\|\overline{g}_i^t - \nabla F_i(w^t)\|^2$, we can further have:

$$\mathbb{E}[\psi^{t+1}] \leq \psi^t - \frac{\eta_s}{2}\|\nabla F(w^t)\|^2 \tag{118}$$

$$+ \frac{\eta_s}{2N}\sum_{i=1}^{N}\mathbb{E}\|\overline{g}_i^t - \nabla F_i(w^t)\|^2 + \eta_s^2 L\mathbb{E}\|\Delta^t - \overline{g}^t\|^2 \tag{119}$$

$$+ \frac{p_{min}p_{avg}\sigma^2}{4L(2-p_{min})K} \tag{120}$$

$$+ \frac{p_{min}}{4L(2-p_{min})N}\sum_{i=1}^{N}p_i^t\mathbb{E}\|\overline{g}_i^t - \nabla F_i(w^t)\|^2 \tag{121}$$

$$\leq \psi^t + \eta_s^2 L\mathbb{E}\|\Delta^t - \overline{g}^t\|^2 - \frac{\eta_s}{2}\|\nabla F(w^t)\|^2 \tag{122}$$

$$+ [\frac{\eta_s}{2} + \frac{p_{min}p_{avg}}{4L(2-p_{min})}]8\eta_c^2L^2K(K-1)\|\nabla F(w^t)\|^2 \tag{123}$$

$$+ \frac{p_{min}p_{avg}\sigma^2}{4L(2-p_{min})K} \tag{124}$$

$$+ [\frac{\eta_s}{2} + \frac{p_{min}p_{avg}}{4L(2-p_{min})}]2\eta_c^2L^2K(K-1)\frac{\sigma^2}{K} \tag{125}$$

$$+ [\frac{\eta_s}{2} + \frac{p_{min}p_{avg}}{4L(2-p_{min})}]8\eta_c^2L^2K(K-1)\sigma_g^2 \tag{126}$$

Here we want the coefficient for the gradient squared norm not exceed $\frac{-\eta_s}{4}$. To achieve this, we bound learning rates as:

$$8\eta_c^2 L^2 K(K-1) \leq \frac{1}{4} \tag{127}$$

$$\frac{p_{min}p_{avg}}{4L(2-p_{min})} \leq \frac{\eta_s}{2} \tag{128}$$

Requirements for learning rates can be easily derived here. Then, we have

$$\mathbb{E}[\psi^{t+1}] \leq \psi^t + \eta_s^2 L \mathbb{E}\|\Delta^t - \overline{g}^t\|^2 - \frac{\eta_s}{4}\|\nabla F(w^t)\|^2 \tag{129}$$

$$+ \frac{p_{min}p_{avg}\sigma^2}{4L(2-p_{min})K} \tag{130}$$

$$+ [\frac{\eta_s}{2} + \frac{p_{min}p_{avg}}{4L(2-p_{min})}]2\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \tag{131}$$

$$+ [\frac{\eta_s}{2} + \frac{p_{min}p_{avg}}{4L(2-p_{min})}]8\eta_c^2 L^2 K(K-1)\sigma_g^2 \tag{132}$$

Next, we provide a bound for the initial progress following similar steps before.

The initial progress can be bounded following similar steps above. Simlarly as Eq. (98),

$$\mathbb{E}[\psi^2] \leq F(w^1) - \frac{\eta_s}{2}\|\nabla F(w^1)\|^2 + \frac{\eta_s}{2N}\sum_{i=1}^{N}\mathbb{E}\|\overline{g}_i^t - \nabla F_i(w^1)\|^2 \tag{133}$$

$$+ \delta\mathbb{E}\|\Delta^1 - \overline{g}^1\|^2 + \alpha\mathbb{E}[H^2] \tag{134}$$

where $H^2 = \frac{1}{N}\sum_{i=1}^{N}\|\nabla F_i(w^1) - h_i^2\|^2$. Similarly, using Lemma 5, we can bound this as:

$$\mathbb{E}[\psi^2] \leq F(w^1) - \frac{\eta_s}{4}\|\nabla F(w^1)\|^2 + \eta_s^2 L \mathbb{E}\|\Delta^1 - \overline{g}^1\|^2 \tag{135}$$

$$+ \frac{p_{min}p_{avg}\sigma^2}{4L(2-p_{min})L} \tag{136}$$

$$+ [\frac{\eta_s}{2} + \frac{p_{min}p_{avg}}{4L(2-p_{min})}]2\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \tag{137}$$

$$+ [\frac{\eta_s}{2} + \frac{p_{min}p_{avg}}{4L(2-p_{min})}]8\eta_c^2 L^2 K(K-1)\sigma_g^2 \tag{138}$$

$$+ \frac{p_{min}(1-p_{min})}{4L(2-p_{min})}\frac{1}{N}\sum_{i=1}^{N}\|\nabla F_i(w^1) - h_i^1\|^2 \tag{139}$$

Finally, we can get the convergence conclusion. From Eq. (129), unfolding he recursion for $t = 2, 3, \cdots, T$, it yields:

$$\mathbb{E}[\psi^{t+1}] \leq \psi^2 + \eta_s^2 L \sum_{t=2}^{T}\mathbb{E}\|\Delta^t - \overline{g}^t\|^2 - \frac{\eta_s}{4}\sum_{t=2}^{T}\mathbb{E}\|\nabla F(w^t)\|^2 \tag{140}$$

$$+ \sum_{t=2}^{T}\frac{p_{min}p_{avg}\sigma^2}{4L(2-p_{min})K} \tag{141}$$

$$+ \sum_{t=2}^{T}[\frac{\eta_s}{2} + \frac{p_{min}p_{avg}}{4L(2-p_{min})}]2\eta_c^2 L^2 K(K-1)\frac{\sigma^2}{K} \tag{142}$$

$$+ \sum_{t=2}^{T}[\frac{\eta_s}{2} + \frac{p_{min}p_{avg}}{4L(2-p_{min})}]8\eta_c^2 L^2 K(K-1)\sigma_g^2 \tag{143}$$

ψ^2 can be bounded as Eq. (132), therefore, we can get

$$\mathbb{E}[\psi^{t+1}] \leq F(w^1) + \eta_s^2 L \sum_{t=1}^{T} \mathbb{E}\|\Delta^t - \overline{g}^t\|^2 - \frac{\eta_s}{4} \sum_{t=1}^{T} \mathbb{E}\|\nabla F(w^t)\|^2 \tag{144}$$

$$+ \sum_{t=1}^{T} \frac{p_{min} p_{avg} \sigma^2}{4L(2-p_{min})K} \tag{145}$$

$$+ \sum_{t=1}^{T} [\frac{\eta_s}{2} + \frac{p_{min} p_{avg}}{4L(2-p_{min})}] 2\eta_c^2 L^2 K(K-1) \frac{\sigma^2}{K} \tag{146}$$

$$+ \sum_{t=1}^{T} [\frac{\eta_s}{2} + \frac{p_{min} p_{avg}}{4L(2-p_{min})}] 8\eta_c^2 L^2 K(K-1) \sigma_g^2 \tag{147}$$

$$+ \frac{p_{min}(1-p_{min})}{4L(2-p_{min})} \frac{1}{N} \sum_{i=1}^{N} \|\nabla F_i(w^1) - h_i^1\|^2 \tag{148}$$

Notice that $\min_{t\in[1,T]} \mathbb{E}\|\nabla F(w^t)\|^2 \leq \frac{\sum_{t=1}^{T} \|\nabla F(w^t)\|^2}{T}$ Dividing both sides of Eq. (148) by T, after rearranging the terms, we can get

$$\min_{t\in[1,T]} \mathbb{E}\|\nabla F(w^t)\|^2 \leq \frac{4(F(w^1) - \mathbb{E}[\psi^T])}{\eta_s T} + 4\eta_s L \frac{\sum_{t=1}^{T} \mathbb{E}\|\Delta^t - \overline{g}^t\|^2}{T} \tag{149}$$

$$+ \frac{1}{T} \frac{p_{min} p_{avg} \sigma^2}{\eta_s L(2-p_{min})K} \tag{150}$$

$$+ [2 + \frac{p_{min} p_{avg}}{\eta_s L(2-p_{min})}] 2\eta_c^2 L^2 K(K-1) \frac{\sigma^2}{K} \tag{151}$$

$$+ [2 + \frac{p_{min} p_{avg}}{\eta_s L(2-p_{min})}] 8\eta_c^2 L^2 K(K-1) \sigma_g^2 \tag{152}$$

$$+ \frac{1}{T} \frac{p_{min}(1-p_{min})}{\eta_s L(2-p_{min})} \frac{1}{N} \sum_{i=1}^{N} \|\nabla F_i(w^1) - h_i^1\|^2 \tag{153}$$

Notice $\psi^T \geq F(w^T) \geq F(w^*)$. Therefore, we can have

$$\min_{t\in[1,T]} \mathbb{E}\|\nabla F(w^t)\|^2 \leq \underbrace{\frac{4(F(w^1) - F(w^*))}{\eta_s T}}_{\text{iterate initialization error}} + \underbrace{4\eta_s L \frac{\sum_{t=1}^{T} \mathbb{E}\|\Delta^t - \overline{g}^t\|^2}{T}}_{\text{partial update variance error}} \tag{154}$$

$$+ \underbrace{\frac{1}{T} \frac{p_{min} p_{avg} \sigma^2}{\eta_s L(2-p_{min})K}}_{\text{stochastic gradient error (will disappear)}} \tag{155}$$

$$+ \underbrace{[2 + \frac{p_{min} p_{avg}}{\eta_s L(2-p_{min})}] 2\eta_c^2 L^2 K(K-1) \frac{\sigma^2}{K}}_{\text{stochastic gradient error}} \tag{156}$$

$$+ \underbrace{[2 + \frac{p_{min} p_{avg}}{\eta_s L(2-p_{min})}] 8\eta_c^2 L^2 K(K-1) \sigma_g^2}_{\text{error from data heterogeneity (non-iid level)}} \tag{157}$$

$$+ \underbrace{\frac{1}{T} \frac{p_{min}(1-p_{min})}{\eta_s L(2-p_{min})} \frac{1}{N} \sum_{i=1}^{N} \|\nabla F_i(w^1) - h_i^1\|^2}_{\text{memory initialization error}} \tag{158}$$

B ADDITIONAL EXPERIMENTAL DETAILS

B.1 DETAILED HYPERPARAMETERS AND MODEL STRUCTURES

For reproducibility, we provide a detailed table (Table 2) specifying the exact model architectures and hyperparameters that we used in the experiments.

Table 2: Hyperparameter settings across datasets.

Hyperparameters	**EMNIST**	**Fashion-MNIST**	**CIFAR-10**
Total rounds	150	150	150
Local batch size	64	64	64
Local epochs	5	5	5
Local learning rate	0.5	0.01	0.3
Global learning rate	0.5	0.5	0.5
Model architecture	CNN (3 Conv: 16, 24, 32 filters, 5×5; 3 FC: 256, 84 units)	CNN (2 Conv: 64/128 filters, 3×3; 2 FC: 512 units)	PreAct ResNet-18 He et al. [2016]

B.2 DETAILED COMPARISON OF COMMUNICATION AND COMPUTE OVERHEAD

FedSteer is designed specifically for resource-constrained edge devices. We provide a detailed comparison below:

Communication Cost (Identical to FedVARP/FedAvg): FedSteer does not incur any additional communication costs compared to standard FL. Active clients send new updates; while inactive clients send nothing (they do not perform local training when they are inactive). No additional bits or control variates are transmitted during training. The dynamic subspace optimization happens on the server only.

Client Compute (Zero Additional Cost Compared to FedAvg): FedSteer's projection logic (solving s_i) happens entirely on the server. Clients only perform standard local training when it participates.

Server Compute (Negligible): The server solves a Ridge Regression problem for active clients (Eq. (5) in the paper) to derive the coordinates $\mathbf{s}_i$. The cost of directly computing the closed-form solution (Eq. (6)) is $O(k^2d + k^3)$, where d is the model dimension and $k = |\mathcal{X}|$ is the core set size. Compared to the cost of standard weight aggregation ($O(Nd)$), directly computing the closed-form may yield comparable cost (for example, N=100, k=10), normally the server is computationally powerful, so they can handle it. In the case that the server is also computation limited, it can alternatively solve it via gradient descent to avoid matrix operations entirely, as the problem is strictly convex. The core set selection (Algorithm 2) is restricted to a short "warm-up" phase, then the core set is fixed – this overhead drops to zero.

Server Memory (Superior Efficiency): FedSteer is significantly more memory-efficient than FedVARP/FedStale/MIFA. FedVARP/FedStale/MIFA must store full stale gradients for all N clients ($O(N \cdot d)$). FedSteer stores projection coordinates ($\mathbf{s}_i \in \mathbb{R}^k$) for all clients, and full stale gradients only for the small core set ($k \ll N$). Therefore the cost is $O((N \cdot k + k \cdot d)$. For large models ($d$) and many clients ($N$), this is a massive reduction. In experiments, we set $k = 10, N = 100$, resulting in reduced memory usage by around 10 times.

B.3 SCALABILITY VIA SUBSAMPLING

Table 3: Performance comparison under subsampled core-set selection on Fashion-MNIST.

Methods	$\gamma = 0.9$	$\gamma = 0.7$
FedAvg	0.509 ± 0.010	0.609 ± 0.010
FedSteer ($N_{\text{sub}} = 20$)	0.537 ± 0.013	0.644 ± 0.011
FedSteer ($N_{\text{sub}} = 100$)	0.554 ± 0.008	0.657 ± 0.011

To ensure scalability for very large N, we introduce a random subsampling strategy during core-set selection. Instead of searching over the entire client population, we randomly sample a subset $N_{\text{sub}} \ll N$ (e.g., 20 candidates out of 100) and perform the greedy swap search within this subset. This reduces the selection complexity to $O(T_0 I_{\max} N_{\text{sub}} k)$. Since N_{sub} is small and fixed (e.g., $N_{\text{sub}} = 20$), the computational cost becomes independent of the total population size N, enabling

scalability to large-scale FL systems. We further evaluate this subsampling strategy. As shown in Table 3, subsampling incurs only a minor accuracy drop (approximately 2.5% on average across two settings) while still outperforming all baselines.

B.4 ADDITIONAL EXPERIMENTAL RESULTS

Figures 4 and 5 empirically demonstrate the convergence behavior of FedSteer against multiple baseline algorithms over 150 global training iterations under extreme heterogeneity ($\gamma = 0.9$). Across both the Fashion-MNIST and CIFAR-10 tasks, FedSteer achieves stable convergence and significantly outperforms all evaluated baseline methods, obtaining a final accuracy superseded only by the theoretical "full participation" upper bound.

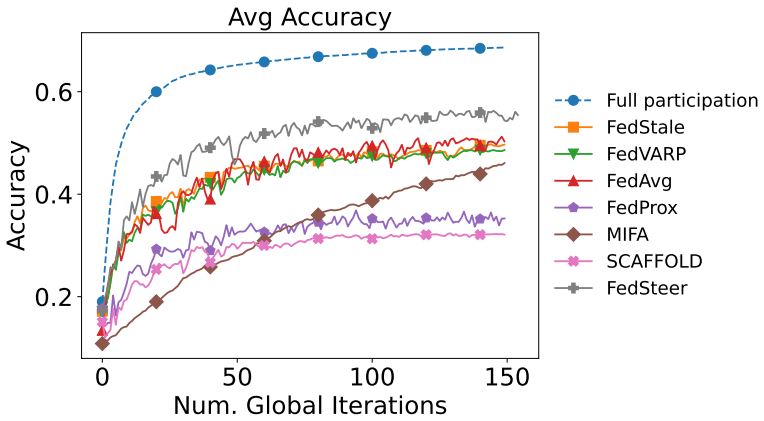


Figure 4: Comparison of average test accuracy against the number of global iterations on the Fashion-MNIST dataset under extreme data and system heterogeneity ($\gamma = 0.9$).

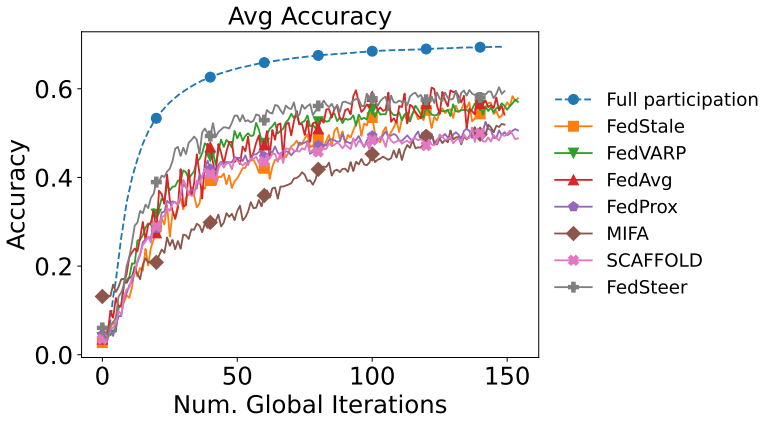
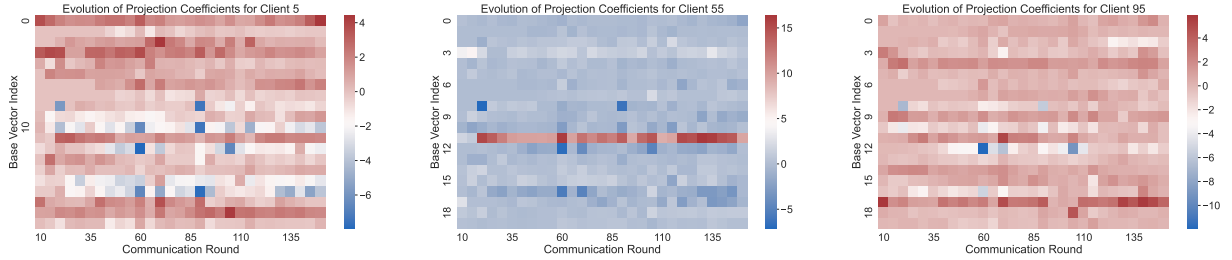


Figure 5: Comparison of average test accuracy against the number of global iterations on the CIFAR-10 dataset under extreme data and system heterogeneity ($\gamma = 0.9$).

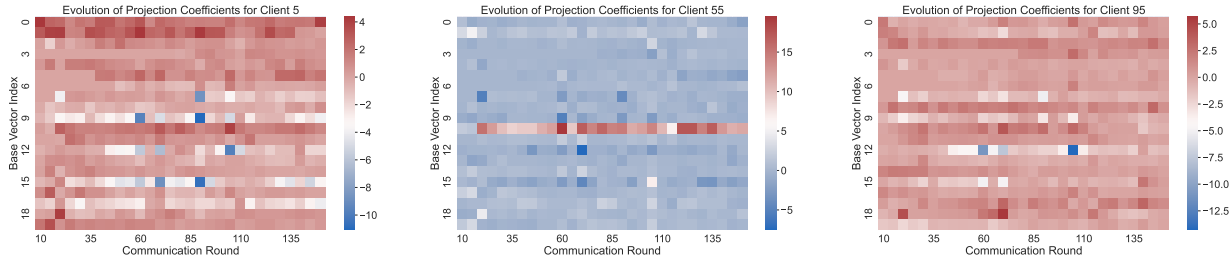
B.5 EFFECTIVENESS OF REUSING $\mathbf{s}_i$ ACROSS ROUNDS

Figure 6 illustrates the evolution of the projection coefficients, s_i, for three representative clients (5, 55, and 95) under different experimental conditions. We present a 2x2 comparison varying the regularization strength (λ) and the core set size ($|X|$). Each heatmap plots the coefficient vector s_i (y-axis) against the communication round (x-axis). The most prominent observation across all settings is the **stability of these coefficients over time**. The vertical patterns, which represent the distribution of s_i at a given round, exhibit slight changes as training progresses. This indicates that the optimal projection of a client's data onto the shared base vectors remains largely consistent. The result is conducted with `FedSteer` (`enforce=5`) under EMNIST. The observed stability supports FedSteer to reuse projection coefficients instead of the

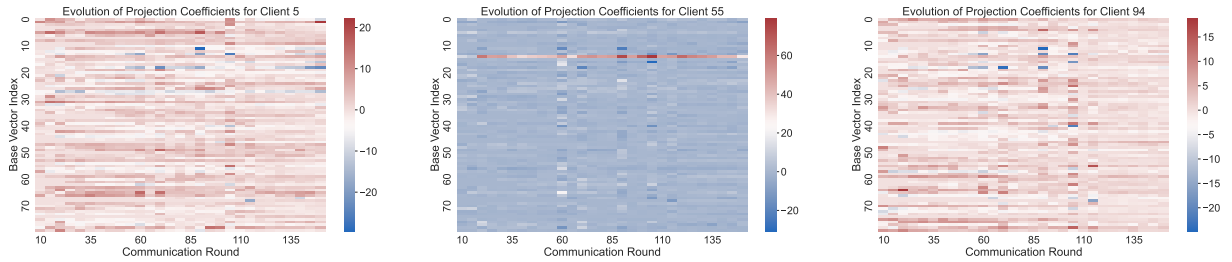
stale gradients, as the coefficients do not change significantly between rounds. Furthermore, the distinct patterns among clients (e.g., the contrast between client 55 and the others) highlight the method's ability to effectively capture stable, yet heterogeneous, client-specific representations.



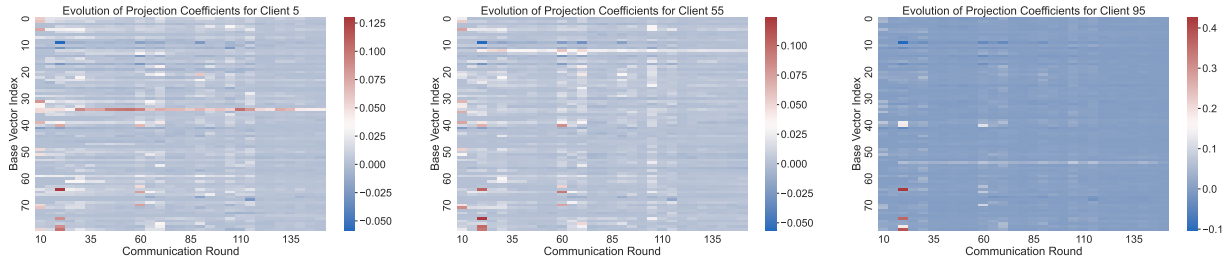
(a) Without regularization ($\lambda = 1\mathrm{e}{-6}$), $|\mathcal{X}| = 20$.



(b) With regularization ($\lambda = 0.5$), $|\mathcal{X}| = 20$.



(c) Without regularization ($\lambda = 1\mathrm{e}{-6}$), $|\mathcal{X}| = 80$.



(d) With regularization ($\lambda = 0.5$), $|\mathcal{X}| = 80$.

Figure 6: Comparison of projection coefficient evolution across different experimental settings. Each triplet of heatmaps shows results for clients 5, 55, and 95, respectively.