FEDECHO: ECHOES OF THE PAST GUIDE FUTURE STARTS IN FEDERATED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated learning typically uses the current global model both as the shared state for continual aggregation and as the common local training starting point for all participating clients, although the latter itself constitutes an independently designable degree of freedom in optimization. Prior work adjusts local initialization using historical model differences or random perturbations, but how to use the historical responses of repeatedly participating clients relative to the concurrent collaborative process to predict their relative optimization trends at the next participation, and accordingly coordinate starting-point dispatch, remains underexplored. We observe that client relative responses, measured relative to the same-round weighted-average displacement and along the concurrent global trajectory, exhibit short-term cross-participation persistence that decays as the participation gap increases. This suggests that what transfers across participations is not a stale gradient or a fixed parameter direction, but a client's response tendency relative to the collaborative process. Building on this observation, we propose **FedEcho**, which recasts training starting-point dispatch as a jointly coordinated, low-dimensional optimization control among clients participating in the same round. The server predicts each client's relative optimization trend from its historical relative response and jointly determines client-specific training starting points to precompensate for the predictable relative component of local-path divergence. The dispatch guarantees that the weighted-average starting point across participating clients equals the current global model, while leaving the original local training and server aggregation rules unchanged. Under smoothness conditions, local-path divergence upper-bounds the corresponding aggregated gradient mismatch, providing an optimization interpretation of this design. Experimental results across three datasets, three network architectures, and three data distributions show that FedEcho achieves the highest test accuracy in all 27 settings, outperforming the strongest baseline by 2.80 percentage points on average.

1 INTRODUCTION

Federated learning (FL) collaboratively trains a shared model through model dispatch, local training, and server aggregation without centralizing raw data (McMahan et al., 2017; Huang et al., 2024; Bao et al., 2024). In synchronous training protocols such as FedAvg, the current global model serves two roles: it is both the shared state for continual aggregation and the common local training starting point for all participating clients (Wang & Ji, 2022; Cho et al., 2023; Sun et al., 2025). The latter is typically treated as a fixed protocol step, although it itself constitutes an independently designable degree of freedom in optimization. Under non-IID data, clients starting from the same point may evolve along different local optimization paths, thereby affecting the aggregated update (Karimireddy et al., 2020; Sun et al., 2024; Patel et al., 2025). For repeatedly participating clients, the server has already observed local updates from their previous participations, which contain information about their responses relative to the concurrent collaborative process. Standard dispatch, however, does not use such longitudinal information to determine their next training starting points. This motivates our central question: *Can a client's historical relative responses be used to predict its relative optimization trend at the next participation and, in turn, jointly coordinate the training starting points of the clients participating in the current round?*

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

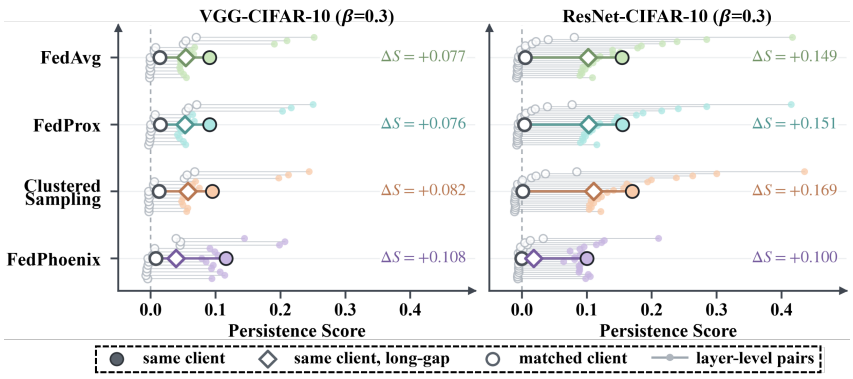

Figure 1: **Cross-participation persistence of client relative responses.** Persistence is the cosine similarity between relative response coordinates across participations. Filled circles, diamonds, and open circles denote same-client, long-gap same-client, and matched different-client pairs, respectively. Small dots show layer-level scores; ΔS is the same-client score minus its matched control. Persistence is higher for the same client and decreases with participation gap.

Existing methods for heterogeneous federated optimization coordinate client training by modifying local objectives, correcting updates, or compensating for parameter drift (Li et al., 2020; Acar et al., 2021; Karimireddy et al., 2020; 2021; Gao et al., 2022). Historical models, global trajectories, and local training signals have also been used to improve the current optimization process (Li et al., 2021; Fan et al., 2024; Li et al., 2025; Tang et al., 2024; Xu et al., 2025). More directly related to our work, several methods instead intervene at local initialization: FedInit adjusts the starting point based on the parameter difference between a historical local model and the current global model (Sun et al., 2023c), while FedMut and FedPhoenix generate differentiated dispatch models through random mutation or parameter resetting (Hu et al., 2024; Wu et al., 2025). These methods show that the shared aggregation state and the local training starting point can be designed separately. This paper instead focuses on a different basis for dispatch: a client's longitudinal response relative to the concurrent collaborative process.

To examine whether historical responses contain client-specific information that transfers across participations, we measure each client's deviation from the same-round weighted-average displacement using the concurrent global trajectory as the reference scale. As shown in Fig. 1, across multiple federated protocols and model architectures, the cross-participation persistence score of a client's relative response exceeds that of matched controls from different clients, and this gap decreases as the participation gap increases. This indicates that, even as the direction of the global trajectory and the composition of participating clients change, a client's relative response retains a short-term predictable component. What transfers across participations is therefore the relative response tendency, rather than a stale gradient or a fixed parameter direction. The local training starting point need not remain a common protocol default; it can instead be conditioned on longitudinal information.

Building on this observation, we propose **FedEcho**, which models server-to-client training starting-point dispatch as a low-dimensional optimization control driven by longitudinal response prediction and jointly coordinated among clients participating in the same round (Fig. 2). The server predicts each client's relative optimization trend for the current round from its historical relative responses and uses these predictions to jointly determine client-specific training starting points within a low-dimensional space spanned by the current global trajectory, while ensuring that their weighted average equals the current global model. The local training and aggregation rules remain unchanged, and the overall process continues to learn a single shared global model. From an optimization perspective, local-path divergence upper-bounds the corresponding aggregated gradient mismatch, and FedEcho's starting-point dispatch can be viewed as pre-compensating for its predictable relative component.

Our main contributions are as follows:

- We observe that client relative responses with respect to the concurrent collaborative process exhibit short-term cross-participation persistence that decays as the participation gap

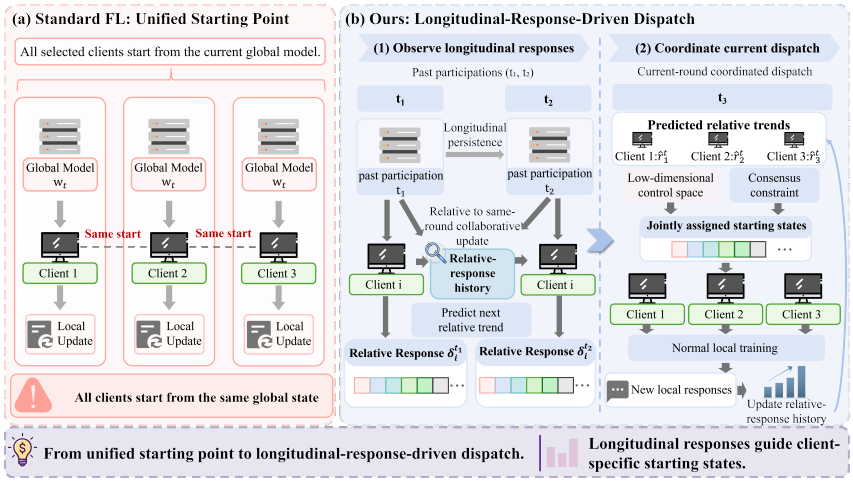

Figure 2: **From a common starting point to longitudinal-response-driven model-state dispatch.** (a) Standard dispatch uses the current global model as the common local training starting point for all participating clients. (b) FedEcho jointly assigns client-specific starting points within a low-dimensional space spanned by the current global trajectory based on historical relative-response prediction, while keeping their weighted average equal to the current global model.

increases, providing empirical support for using longitudinal information to predict subsequent relative optimization trends.

- We propose **FedEcho**, which recasts training starting-point dispatch as a low-dimensional control driven by longitudinal response prediction and jointly coordinated among clients participating in the same round. FedEcho assigns client-specific starting points while anchoring their weighted average at the current global model, and we provide an optimization interpretation based on local-path divergence.
- Across 27 settings spanning three image classification datasets, three network architectures, and three data distributions, FedEcho achieves the highest test accuracy in every setting, outperforming the strongest baseline by 2.80 percentage points on average.

2 Related Work

2.1 Heterogeneous Federated Optimization and Cross-Round Information

Heterogeneous federated optimization commonly coordinates client training by adjusting local objectives, gradient estimates, or update dynamics (Qu et al., 2022; Sun et al., 2023b;a; Lee et al., 2024). FedProx and FedDyn modify local objectives through proximal constraints and dynamic regularization (Li et al., 2020; Acar et al., 2021); SCAFFOLD and Mime correct local updates using control variates or server-side optimization statistics (Karimireddy et al., 2020; 2021); and FedDC maintains local drift variables to compensate for parameter deviations (Gao et al., 2022). Recent work further improves local training signals through feature distributions, sample selection, and gradient correction under system heterogeneity (Tang et al., 2024; Xu et al., 2025; Hu et al., 2026). These methods primarily modify local objectives, training signals, or stepwise update dynamics. FedEcho instead leaves the local training process unchanged and moves the intervention point to model-state dispatch before local training begins.

Cross-round history and the global trajectory have also been used to condition the current round's optimization. MOON constructs representation-level contrastive constraints using historical local models and the global model (Li et al., 2021). FedLESAM estimates a perturbation direction from the difference between the global model from a client's previous participation and the one received in the current round, while FedGMT uses the global model trajectory to construct perturbations for sharpness-aware optimization (Fan et al., 2024; Li et al., 2025). These methods primarily use historical information to shape constraints or perturbations during local training. FedEcho instead

uses a client's longitudinal relative responses to the concurrent collaborative process to predict its subsequent relative optimization trend and determine starting-point dispatch accordingly. The current global trajectory spans a low-dimensional model-state control space rather than serving as a perturbation direction for stepwise local updates.

2.2 Local Initialization and Model-State Dispatch

The local training starting point itself can also serve as a degree of freedom for coordinating client optimization. FedInit constructs a relaxed initialization from the parameter difference between a client's most recent local model and the current global model (Sun et al., 2023c). For personalized federated learning, FedALA uses local data to learn fusion weights between the global model and a retained local model to form the local initialization (Zhang et al., 2023), while FedLFH exploits cross-round information by combining historical tracking variables with global and personalized feature extractors (Fu et al., 2026). In contrast, FedEcho determines training starting points from historical relative-response predictions rather than historical parameter differences or retained local models, and it continues to optimize a single shared global model rather than personalized models.

The closest methods to FedEcho are server-side approaches for differentiated model-state dispatch. FedMut constructs randomly mutated models from the global update and dispatches them to clients (Hu et al., 2024), while FedPhoenix generates different training starting points by randomly resetting selected convolutional kernels in copies of the global model (Wu et al., 2025). Like FedEcho, these methods allow clients to begin local training from differentiated states while continuing to learn a single shared global model. Their basis for dispatch, however, differs: FedMut and FedPhoenix primarily introduce starting-point differences through random mutation or random resetting, whereas FedEcho drives starting-point adjustments with client-specific longitudinal response predictions and jointly determines them among clients participating in the same round, keeping the weighted-average starting point anchored at the current global model.

3 FedEcho: Longitudinal-Response-Driven Model-State Dispatch

Let $\mathbf{w}_t$ denote the global model at round t, $\mathcal{S}_t$ the set of participating clients, and $\mathcal{D}_i$ the local dataset of client i containing n_i samples. The local objective is $F_i(\mathbf{w}) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i}\ell(f_{\mathbf{w}}(\mathbf{x}), y)$, the aggregation weight is $p_i^t = n_i / \sum_{j\in\mathcal{S}_t} n_j$, and local training is denoted by $\mathcal{U}_i(\cdot;\mathcal{D}_i)$. FedEcho leaves the standard FedAvg local training and server aggregation rules unchanged, maintains $\mathbf{w}_t$ as the single shared global model, and uses the historical relative responses of repeatedly participating clients to predict their relative optimization trends at the next participation, based on which it jointly assigns local training starting points (Fig. 3).

3.1 Longitudinal Relative Response and Prediction

FedEcho restricts client starting-point control to a low-dimensional space constructed from the server's recent global displacement. Let $\mathbf{m}_t = \mathbf{w}_t - \mathbf{w}_{t-1}$ ($\mathbf{m}_0 = \mathbf{0}$). In its basic form, the control space is the one-dimensional subspace $\mathcal{V}_t = \mathrm{span}\{\mathbf{m}_t\}$; if $\mathbf{m}_t = \mathbf{0}$, then $\mathcal{V}_t = \{\mathbf{0}\}$. Client i performs local training from the server-dispatched starting point $\mathbf{x}_i^t$ (see §3.2 for the dispatch rule):

$$\mathbf{y}_i^t = \mathcal{U}_i(\mathbf{x}_i^t; \mathcal{D}_i), \qquad \mathbf{u}_i^t = \mathbf{y}_i^t - \mathbf{x}_i^t, \qquad \bar{\mathbf{u}}^t = \sum_{j\in\mathcal{S}_t} p_j^t \mathbf{u}_j^t, \tag{1}$$

where $\mathbf{u}_i^t$ is the full local training displacement. Throughout the main text, we assume that the same client follows a consistent local training protocol across participations. When $\|\mathbf{m}_t\|$ satisfies the numerical validity condition, the server projects the client's deviation from the same-round weighted-average displacement onto the current global trajectory:

$$\delta_i^t = \frac{\langle \mathbf{u}_i^t - \bar{\mathbf{u}}^t, \mathbf{m}_t \rangle}{\|\mathbf{m}_t\|^2}. \tag{2}$$

The response coordinate δ_i^t, scaled by the current global step length, characterizes the client's response along this trajectory relative to the clients participating in the same round. Since $\mathbf{u}_i^t$ results from the actual dispatched starting point, δ_i^t is a closed-loop response under the current dispatch policy. For each client, the server stores its most recent valid response and the round in which it was observed, (δ_i, τ_i).

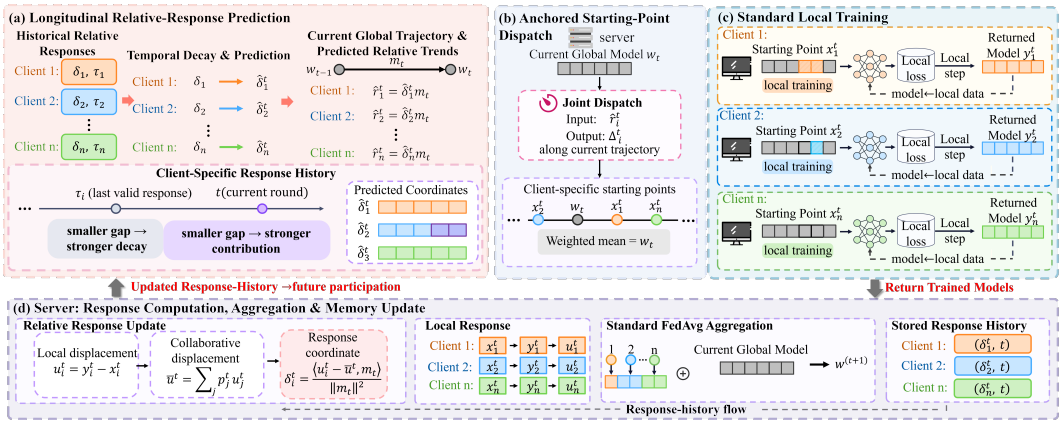

Figure 3: **Overview of the FedEcho framework.** (a) The server predicts each client's relative optimization trend from its longitudinal relative response: for each client, only the single most recent valid response (δ_i, τ_i) is stored, decayed according to the participation gap $t - \tau_i$ (larger gap $\rightarrow$ weaker decay; smaller gap $\rightarrow$ stronger decay), and projected onto the current global trajectory $\mathbf{m}_t$ to obtain the predicted response $\hat{\mathbf{r}}_i^t$. (b) These predictions serve as input to a consensus-preserving starting-point dispatch: the server jointly determines client-specific offsets $\mathbf{\Delta}_i^t$ along the current trajectory, under the constraint that their weighted mean equals the current global model $\mathbf{w}_t$. (c) Each client performs standard local training from its dispatched starting point $\mathbf{x}_i^t$ and returns the trained model $\mathbf{y}_i^t$. (d) The server aggregates the returned models via standard FedAvg weighted aggregation, computes each client's updated response coordinate δ_i^t, and stores it for use in future participation, closing the loop between response history and subsequent dispatch.

Longitudinal transferability assumption. The historical coordinate δ_i is measured along direction $\mathbf{m}_{\tau_i}$, whereas the next control action is applied along the current direction $\mathbf{m}_t$. FedEcho assumes that, even as the specific trajectory direction and the composition of participating clients change, a client's relative response to the concurrent collaborative process, scaled by the step length of the concurrent global trajectory, retains short-term longitudinal persistence (see Fig. 1 for empirical evidence). What is transferred under this assumption is the response coordinate, rather than a stale gradient or a previous parameter direction. Accordingly, the server predicts

$$\hat{\delta}_i^t = \hat{\rho}_t^{\,t-\tau_i}\delta_i, \qquad \hat{\mathbf{r}}_i^t = \hat{\delta}_i^t \mathbf{m}_t, \tag{3}$$

where $\hat{\rho}_t \in [0,1]$ is estimated by aggregating, over all clients, their pairs of adjacent valid responses observed before round t together with the corresponding participation gaps. For a first-time participant or a client without valid history, we set $\hat{\delta}_i^t = 0$.

3.2 Consensus-Preserving Starting-Point Dispatch

The server dispatches $\mathbf{x}_i^t = \mathbf{w}_t + \mathbf{\Delta}_i^t$ to client i. Let $\kappa \in [0,1]$ denote normalized local training progress. FedEcho treats $\hat{\mathbf{r}}_i^t$ as a given predicted response for the current dispatch decision and uses $\mathbf{\Delta}_i^t + \kappa\hat{\mathbf{r}}_i^t$ as a linear proxy for the predictable relative component of the local path. It then jointly solves

$$\min_{\{\mathbf{\Delta}_i^t\}} \sum_{i\in\mathcal{S}_t} p_i^t \int_0^1 \|\mathbf{\Delta}_i^t + \kappa\hat{\mathbf{r}}_i^t\|^2 \, d\kappa \quad \text{s.t.} \quad \mathbf{\Delta}_i^t \in \mathcal{V}_t,\ \sum_{i\in\mathcal{S}_t} p_i^t \mathbf{\Delta}_i^t = \mathbf{0},\ \|\mathbf{\Delta}_i^t\| \le \|\mathbf{m}_t\|. \tag{4}$$

The zero-sum constraint guarantees $\sum_{i\in\mathcal{S}_t} p_i^t \mathbf{x}_i^t = \mathbf{w}_t$, so the weighted-average starting point remains strictly anchored at the current shared global model. The trust region bounds each client's starting-point offset to the scale of the most recent global step.

Let $\bar{\mathbf{r}}^t = \sum_{j\in\mathcal{S}_t} p_j^t \hat{\mathbf{r}}_j^t$. When the trust-region constraint is inactive, the solution to Eq. equation 4 is

$$\mathbf{\Delta}_i^{t,*} = -\frac{1}{2}\left(\hat{\mathbf{r}}_i^t - \bar{\mathbf{r}}^t\right). \tag{5}$$

The centering arises from the zero-sum constraint, while the factor $1/2$ results from uniformly integrating the linear predicted path over normalized local training progress.

Structured control space. The one-dimensional construction above extends to a structured low-dimensional control space. Using a fixed rule, we decompose $\mathbf{m}_t$ into pairwise orthogonal structured components satisfying $\sum_{a=1}^{r} \mathbf{d}_{t,a} = \mathbf{m}_t$, and set $\mathcal{V}_t = \mathrm{span}\{\mathbf{d}_{t,a}\}_{a=1}^{r}$. For each numerically valid component, the server separately maintains response coordinates that transfer across participations, predicts them using a shared temporal decay parameter, and then solves Eq. equation 4 within this low-dimensional space. The fixed construction gives the same component index a consistent structural meaning across rounds; for CNNs, we use a rank-2 instance.

3.3 Local Training, Global Aggregation, and Optimization Connection

Once $\mathbf{\Delta}_i^t$ is determined, each client performs the original local training according to Eq. equation 1, and the server continues to use FedAvg weighted aggregation. By the zero-sum constraint,

$$\mathbf{w}_{t+1} = \sum_{i \in \mathcal{S}_t} p_i^t \mathbf{y}_i^t = \mathbf{w}_t + \bar{\mathbf{u}}^t. \tag{6}$$

This equality shows that the zero-sum constraint removes the direct average contribution of the starting-point offsets from weighted aggregation. However, because $\mathbf{u}_i^t$ depends on the actual dispatched starting point, FedEcho's global update generally differs from that of FedAvg with a common starting point. After aggregation, the server updates the response history according to Eq. equation 2, forming the closed loop $(\delta_i, \tau_i) \to \hat{\delta}_i^t \to \mathbf{\Delta}_i^t \to \mathbf{x}_i^t \to \mathbf{u}_i^t \to \delta_i^t$.

The dispatch objective can be connected to aggregated gradient mismatch through local-path divergence. Consider the case where all clients use the same number of local steps and the same optimizer. Let $\mathbf{y}_{i,0}^t = \mathbf{x}_i^t$, let $\mathbf{y}_{i,k}^t$ denote the model of client i after its k-th local update, and define the consensus path as $\bar{\mathbf{y}}_k^t = \sum_{i \in \mathcal{S}_t} p_i^t \mathbf{y}_{i,k}^t$. By the zero-sum constraint, $\bar{\mathbf{y}}_0^t = \mathbf{w}_t$, so the initial path divergence for client i is $\mathbf{\Delta}_i^t$. If each F_i is L-smooth, then

$$\left\| \sum_{i \in \mathcal{S}_t} p_i^t \left[\nabla F_i(\mathbf{y}_{i,k}^t) - \nabla F_i(\bar{\mathbf{y}}_k^t) \right] \right\|^2 \leq L^2 \sum_{i \in \mathcal{S}_t} p_i^t \|\mathbf{y}_{i,k}^t - \bar{\mathbf{y}}_k^t\|^2. \tag{7}$$

Thus, the local-path divergence on the right-hand side upper-bounds the corresponding aggregated gradient mismatch. Equation equation 4 does not directly minimize the full true path divergence. Instead, it uses historical responses to construct a proxy for its predictable low-dimensional relative component and pre-compensates for this component through starting-point dispatch. Its practical effect depends on the quality of response prediction and the sensitivity of local optimization to changes in the starting point.

The numerical validity condition, estimation of $\hat{\rho}_t$, and the complete procedure (Algorithm 1) are provided in Appendix B; the constrained solution when the trust region is active is given in Appendix C; and the proof of Eq. equation 7, the connection between the discrete path and the integral proxy, and the specific construction of structured components are given in Appendix D.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate FedEcho on CIFAR-10, CIFAR-100 (Krizhevsky, 2009), and Tiny-ImageNet (Le & Yang, 2015). Data partitioning, client population, and participation follow FedPhoenix (Wu et al., 2025). For each dataset, we construct a federated system with $N = 100$ clients and partition the training data either uniformly under IID or non-IID using a Dirichlet distribution (Hsu et al., 2019), $\mathrm{Dir}(\beta)$, with $\beta \in \{0.6, 0.3\}$, where a smaller β indicates stronger data heterogeneity. By default, $K = 10$ clients are sampled uniformly at random in each round with participation rate $f = 0.1$. ClusteredSampling selects clients according to its own sampling algorithm while keeping the same number of participating clients per round.

Table 1: **Test accuracy comparison under IID and non-IID settings across three network architectures.** Mean ± sample standard deviation over three seeds.

Model	Dataset	Setting	Test Accuracy (%)						
			FedAvg	FedProx	FedMut	ClusteredSampling	FedGen	FedPhoenix	**FedEcho**
VGG-16	CIFAR-10	$\beta = 0.3$	76.71±1.07	77.11±0.12	78.72±0.44	76.43±0.15	77.82±0.57	82.61±0.68	**86.85±0.49**
		$\beta = 0.6$	78.73±0.08	79.00±0.07	81.14±0.21	79.18±0.34	80.01±0.06	84.56±0.33	**88.04±0.24**
		IID	80.19±0.02	80.25±0.01	82.23±0.02	80.47±0.01	81.20±0.01	87.06±0.04	**90.34±0.07**
	CIFAR-100	$\beta = 0.3$	54.73±0.06	54.61±0.41	56.34±0.46	54.18±0.57	54.27±0.68	61.72±0.19	**64.86±0.28**
		$\beta = 0.6$	56.08±0.11	56.55±0.47	57.42±0.36	56.37±0.61	55.16±0.12	62.81±0.23	**65.19±0.21**
		IID	57.82±0.02	57.87±0.02	59.26±0.02	57.63±0.09	56.24±0.07	64.53±0.11	**66.59±0.15**
	Tiny-ImageNet	$\beta = 0.3$	51.36±0.07	51.62±0.10	52.23±0.05	51.25±0.07	52.43±0.12	57.48±0.18	**59.21±0.26**
		$\beta = 0.6$	52.04±0.04	52.31±0.13	53.12±0.07	53.88±0.03	52.67±0.10	57.91±0.12	**59.68±0.19**
		IID	52.18±0.02	52.09±0.09	53.00±0.03	54.72±0.18	53.06±0.10	59.10±0.06	**60.50±0.09**
ResNet-18	CIFAR-10	$\beta = 0.3$	59.13±0.35	60.20±0.36	64.34±0.70	59.22±0.59	61.24±0.21	82.11±0.59	**86.69±0.46**
		$\beta = 0.6$	65.09±0.14	65.39±0.05	68.01±0.10	65.51±0.22	66.89±0.13	82.67±0.76	**87.44±0.28**
		IID	65.03±0.04	65.03±0.15	66.54±0.03	65.08±0.02	65.63±0.11	85.54±0.16	**89.69±0.08**
	CIFAR-100	$\beta = 0.3$	42.16±0.15	40.91±0.38	45.12±0.09	40.37±0.17	41.53±0.22	57.64±0.21	**60.18±0.31**
		$\beta = 0.6$	43.72±0.09	44.56±0.06	47.20±0.03	44.49±0.11	42.81±0.10	59.72±0.23	**62.03±0.24**
		IID	43.36±0.04	43.22±0.17	45.65±0.27	43.41±0.13	43.38±0.18	62.09±0.06	**64.79±0.17**
	Tiny-ImageNet	$\beta = 0.3$	49.27±0.09	49.28±0.13	50.11±0.05	49.35±0.10	49.04±0.07	59.27±0.34	**61.15±0.25**
		$\beta = 0.6$	49.59±0.11	49.54±0.09	50.70±0.06	50.19±0.12	49.67±0.04	59.96±0.18	**61.68±0.24**
		IID	50.02±0.06	49.85±0.04	50.27±0.04	50.30±0.05	49.26±0.03	61.12±0.11	**62.98±0.14**
MobileNet-V1	CIFAR-10	$\beta = 0.3$	54.75±3.62	54.49±1.55	57.15±1.18	54.83±1.71	56.03±1.08	75.02±0.89	**80.18±0.92**
		$\beta = 0.6$	60.61±1.12	60.25±0.58	61.73±0.09	59.74±0.44	62.05±0.69	77.37±0.31	**81.41±0.18**
		IID	62.32±0.12	61.75±0.16	62.62±0.07	62.63±0.14	62.82±0.02	81.91±0.11	**86.08±0.07**
	CIFAR-100	$\beta = 0.3$	36.21±0.28	36.74±0.20	39.12±0.16	36.68±0.29	36.57±0.17	53.24±0.94	**55.68±0.69**
		$\beta = 0.6$	40.01±0.06	40.36±0.09	41.47±0.07	40.02±0.12	39.14±0.11	54.72±0.35	**57.11±0.19**
		IID	41.97±0.20	41.73±0.16	43.28±0.19	42.73±0.07	40.86±0.10	59.41±0.09	**61.51±0.12**
	Tiny-ImageNet	$\beta = 0.3$	39.38±0.07	39.11±0.06	42.30±0.06	40.15±0.04	40.59±0.08	54.24±0.15	**56.12±0.20**
		$\beta = 0.6$	40.49±0.05	41.18±0.07	43.55±0.03	41.50±0.04	41.38±0.06	55.04±0.09	**56.89±0.14**
		IID	40.84±0.02	41.06±0.04	43.48±0.03	40.55±0.06	40.90±0.05	57.29±0.08	**58.94±0.06**

Models and baselines. We use VGG-16 (Simonyan & Zisserman, 2015), ResNet-18 (He et al., 2016), and MobileNet-V1 (Howard et al., 2017) as backbone architectures, and compare FedEcho with FedAvg (McMahan et al., 2017), FedProx (Li et al., 2020), FedGen (Zhu et al., 2021), ClusteredSampling (Fraboni et al., 2021), FedMut (Hu et al., 2024), and FedPhoenix (Wu et al., 2025). All methods are rerun within the same framework using identical general training settings.

Implementation details. The client-side classification models of all methods are optimized with SGD using a local learning rate of 0.01, momentum of 0.5, and batch size of 50. Each selected client performs $E = 5$ local epochs, and training lasts for $T = 1000$ communication rounds. The overall experimental protocol follows FedPhoenix (Wu et al., 2025). FedEcho is implemented using the default settings described in Section 3 and the appendix, without introducing additional tuned client-side optimization hyperparameters. Experiments are implemented in PyTorch 2.7.1 and run on an NVIDIA RTX 4090 GPU. All main experiments use three random seeds, and we report the mean ± sample standard deviation of test accuracy. The same metric computation is used for all methods. The best and second-best results in each table are marked in bold and underlined, respectively.

4.2 Performance Comparison

Table 1 reports the test accuracy of all methods across 27 settings spanning three datasets, three network architectures, and three data partitions. FedEcho achieves the highest mean test accuracy in every setting, with FedPhoenix being the second-best method throughout. Averaged across settings, FedEcho improves over the strongest baseline in each setting by 2.80 percentage points. The gains are largest on CIFAR-10 and narrower on CIFAR-100 and Tiny-ImageNet, and are also observed under IID partitioning as well as under both non-IID Dirichlet splits ($\beta = 0.6$ and $\beta = 0.3$), indicating that the improvement is observable across the tested model architectures, datasets, and data partitions rather than being confined to a specific configuration.

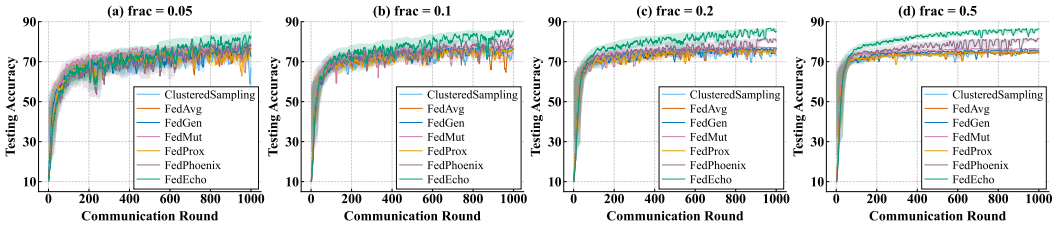

Figure 4: Learning curves on CIFAR-10 using VGG with Dirichlet $\beta = 0.3$ under different client participation rates.

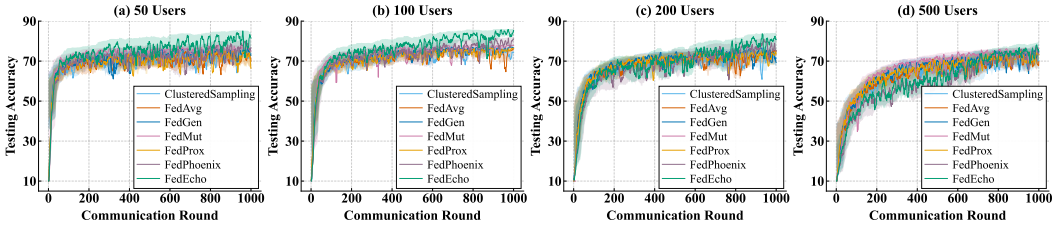

Figure 5: Learning curves on CIFAR-10 using VGG with Dirichlet $\beta = 0.3$ under different total numbers of clients.

4.3 Effect of Participation Settings

Both studies below use VGG-16 on CIFAR-10 with $\beta = 0.3$.

Client participation rate. Figure 4 fixes the total number of clients at 100 and compares learning curves under participation rates $f \in \{0.05, 0.1, 0.2, 0.5\}$. FedEcho achieves the highest test accuracy at all four participation rates, with its advantage emerging earlier at higher participation rates. For $f = 0.2$ and $f = 0.5$, FedEcho separates from the baselines early in training. At $f = 0.05$, the curves overlap substantially during the early stage, and FedEcho's advantage emerges only later in training.

Total number of clients. Figure 5 fixes the number of selected clients per round at 10 and compares learning curves for $N \in \{50, 100, 200, 500\}$. FedEcho achieves the highest test accuracy in all four settings, but its advantage becomes smaller and emerges later as N increases. At $N = 500$, FedEcho trails most baselines during the early and middle stages of training and only overtakes them later. As the client pool grows, repeated participation by the same client becomes sparser, slowing the accumulation and updating of historical responses. This may explain the later emergence of FedEcho's advantage.

4.4 Ablation Study

Table 2 compares FedEcho with its ablated variants using VGG-16 on CIFAR-10 with $\beta = 0.3$. Shuffling the client-history correspondence causes the largest drop, supporting the role of preserving client-specific historical correspondence. Replacing relative responses with absolute historical updates reduces accuracy to 83.91%, supporting the use of a response representation defined relative to the concurrent collaborative process. Removing temporal decay, weighted centering, or the trust region decreases accuracy by 0.89, 1.38, and 1.57 percentage points, respectively, indicating that each design contributes under this setting.

Computational overhead. Table 3 compares the additional per-round computation of each method relative to FedAvg. FedEcho introduces additional computation only on the server, comprising response-coordinate extraction and starting-point dispatch, without adding client-side for-

Table 2: **FedEcho ablation** on CIFAR-10 with VGG-16 ($\beta = 0.3$). Mean $\pm$ sample standard deviation over three seeds. "Hist.": client-specific history correspondence; "Rel.": cohort-relative historical response.

Method	Hist.	Rel.	Decay	Center	Trust	Acc. (%)
FedAvg	–	–	–	–	–	76.71 ± 1.07
Shuffled History	×	✓	✓	✓	✓	81.42 ± 0.58
w/o Relative Historical Response	✓	×	✓	✓	✓	83.91 ± 0.43
w/o Temporal Decay	✓	✓	×	✓	✓	85.96 ± 0.29
w/o Weighted Centering	✓	✓	✓	×	✓	85.47 ± 0.36
w/o Trust Region	✓	✓	✓	✓	×	85.28 ± 0.47
FedEcho	✓	✓	✓	✓	✓	$\mathbf{86.85 \pm 0.49}$

Table 3: **Additional per-round computational overhead (ΔFLOPs) relative to FedAvg.** Baseline estimates follow FedPhoenix; the FedEcho estimate is derived from its additional server-side operations.

Method	Main overhead	ΔFLOPs
FedEcho (Ours)	Server: trajectory projection, coordinates, and model-state dispatch	$\approx 6KM + O(M)$
FedPhoenix	Server: statistics and parameter sampling	$M(6 + 10\theta K)$
FedProx	Client: proximal term in each local step	$\approx 3PKM$
FedMut	Server: gradient aggregation and mutation	$M(1 + 2K)$
ClusteredSampling	Server: client similarity computation	$\approx 3KNM$
FedGen	Server/client: generator training and distillation	High*

M: model parameters; K/N: participating/total clients; P: local steps; θ: reset ratio. Scalar operations are omitted. *Configuration-dependent.

ward or backward passes. Under rank-one control, the additional computation is approximately $6KM + O(M)$, where K is the number of participating clients per round and M is the number of model parameters, and therefore scales linearly with both K and M.

4.5 Limitations

Our experiments are limited to image classification tasks and convolutional neural networks, and the structured control space is instantiated only for convolutional networks. Its applicability to other tasks and Transformer architectures remains to be validated. FedEcho's predictive mechanism depends on repeated client participation and cross-participation persistence of relative responses; sparse participation may leave historical information insufficient or stale. Under the tested setting with $N = 500$ and $K = 10$, FedEcho trails most baselines during the early and middle stages of training.

5 Conclusion

Motivated by the observation that client relative responses exhibit short-term cross-participation persistence, we presented FedEcho, which models server-to-client local training starting-point dispatch as a low-dimensional optimization control driven by longitudinal relative-response prediction and jointly coordinated among clients participating in the same round. FedEcho jointly assigns client-specific starting points while keeping their weighted average equal to the current global model, leaves the original local training and aggregation rules unchanged, and continues to learn a single shared global model. The theoretical analysis provides an upper bound on the aggregated gradient mismatch in terms of local-path divergence, providing an optimization interpretation of the dispatch proxy. Across 27 settings spanning three image classification datasets, three network architectures, and three data partitions, FedEcho achieves higher mean test accuracy than all compared methods, improving over the strongest baseline in each setting by 2.80 percentage points on average.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, organization, and presentation of the manuscript. All AI-assisted content was reviewed and verified by the authors, who take full responsibility for the technical correctness, experimental results, and final content of this work.

REPRODUCIBILITY STATEMENT

The formulation of FedEcho is given in Section 3, and its complete per-round procedure is summarized in Algorithm 1. Appendix B describes the estimation of the temporal persistence coefficient and the numerical validity criterion, Appendix C derives the consensus-preserving dispatch solution, including the trust-region-constrained case, and Appendix D provides the proof of the gradient-mismatch bound and the construction of the structured control space. Datasets, data partitioning, models, training hyperparameters, and evaluation metrics are specified in Section 4.1; all main results are averaged over three random seeds and reported as mean $\pm$ sample standard deviation.

REFERENCES

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas Navarro, Matthew Mattina, Paul N. Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=B7v4QMR6Z9w.

Yajie Bao, Michael Crawshaw, and Mingrui Liu. Provable benefits of local steps in heterogeneous federated learning for neural networks: A feature learning perspective. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 2857–2902. PMLR, 2024. URL https://proceedings.mlr.press/v235/bao24a.html.

Yae Jee Cho, Pranay Sharma, Gauri Joshi, Zheng Xu, Satyen Kale, and Tong Zhang. On the convergence of federated averaging with cyclic client participation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 5677–5721. PMLR, 2023. URL https://proceedings.mlr.press/v202/cho23b.html.

Ziqing Fan, Shengchao Hu, Jiangchao Yao, Gang Niu, Ya Zhang, Masashi Sugiyama, and Yanfeng Wang. Locally estimated global perturbations are better than local perturbations for federated sharpness-aware minimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12858–12881. PMLR, 2024. URL https://proceedings.mlr.press/v235/fan24c.html.

Yann Fraboni, Richard Vidal, Laetitia Kameni, and Marco Lorenzi. Clustered sampling: Low-variance and improved representativity for clients selection in federated learning. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 3407–3416. PMLR, 2021. URL https://proceedings.mlr.press/v139/fraboni21a.html.

Yingxun Fu, Shulan Yin, Li Ma, and Jie Liu. Learning from history for personalized federated learning. *Neural Networks*, 193:108071, 2026. doi: 10.1016/j.neunet.2025.108071. URL https://www.sciencedirect.com/science/article/abs/pii/S0893608025009517.

Liang Gao, Huazhu Fu, Li Li, Yingwen Chen, Ming Xu, and Cheng-Zhong Xu. FedDC: Federated learning with non-iid data via local drift decoupling and correction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10112–10121, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Gao_FedDC_Federated_Learning_With_Non-IID_Data_via_Local_Drift_Decoupling_CVPR_2022_paper.html.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016. URL https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html.

Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. MobileNets: Efficient convolutional neural networks for mobile vision applications, 2017. URL https://arxiv.org/abs/1704.04861.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification, 2019. URL https://arxiv.org/abs/1909.06335.

Ming Hu, Yue Cao, Anran Li, Zhiming Li, Chengwei Liu, Tianlin Li, Mingsong Chen, and Yang Liu. FedMut: Generalized federated learning via stochastic mutation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 12528–12537, 2024. doi: 10.1609/aaai.v38i11.29146. URL https://ojs.aaai.org/index.php/AAAI/article/view/29146.

Qing Hu, Tianchi Liao, Shuyi Wu, Lei Yang, and Chuan Chen. FedLAGC: Towards high performance system-heterogeneous federated learning via layer-adaptive submodel extraction and gradient correction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 21858–21866, 2026. doi: 10.1609/aaai.v40i26.39338. URL https://ojs.aaai.org/index.php/AAAI/article/view/39338.

Wei Huang, Ye Shi, Zhongyi Cai, and Taiji Suzuki. Understanding convergence and generalization in federated learning through feature learning theory. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/6d892051f91812700f427250e95e04f5-Abstract-Conference.html.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5132–5143. PMLR, 2020. URL https://proceedings.mlr.press/v119/karimireddy20a.html.

Sai Praneeth Karimireddy, Martin Jaggi, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian U Stich, and Ananda Theertha Suresh. Breaking the centralized barrier for cross-device federated learning. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/f0e6be4ce76ccfa73c5a540d992d0756-Abstract.html.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Ya Le and Xuan Yang. Tiny ImageNet visual recognition challenge. Stanford University, CS231n course project report, 2015. URL https://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf.

Gihun Lee, Minchan Jeong, Sangmook Kim, Jaehoon Oh, and Se-Young Yun. FedSOL: Stabilized orthogonal learning with proximal restrictions in federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12512–12522, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Lee_FedSOL_Stabilized_Orthogonal_Learning_with_Proximal_Restrictions_in_Federated_Learning_CVPR_2024_paper.html.

Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10713–10722, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Li_Model-Contrastive_Federated_Learning_CVPR_2021_paper.html.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, volume 2, 2020. URL https://proceedings.mlsys.org/paper/2020/hash/1f5fe83998a09396ebe6477d9475ba0c-Abstract.html.

Yuhang Li, Tong Liu, Yangguang Cui, Ming Hu, and Xiaoqiang Li. One arrow, two hawks: Sharpness-aware minimization for federated learning via global model trajectory. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 35317–35341. PMLR, 2025. URL https://proceedings.mlr.press/v267/li25bd.html.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282. PMLR, 2017. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Kumar Kshitij Patel, Ali Zindari, Sebastian Stich, and Lingxiao Wang. Revisiting consensus error: A fine-grained analysis of local SGD under second-order data heterogeneity. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1959. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/54cee855c7eb9429b4b3668d06bf670f-Abstract-Conference.html.

Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18250–18280. PMLR, 2022. URL https://proceedings.mlr.press/v162/qu22a.html.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. URL https://www.robots.ox.ac.uk/~vgg/publications/2015/Simonyan15/.

Yan Sun, Li Shen, Shixiang Chen, Liang Ding, and Dacheng Tao. Dynamic regularized sharpness aware minimization in federated learning: Approaching global consistency and smooth landscape. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32991–33013. PMLR, 2023a. URL https://proceedings.mlr.press/v202/sun23h.html.

Yan Sun, Li Shen, Tiansheng Huang, Liang Ding, and Dacheng Tao. FedSpeed: Larger local interval, less communication round, and higher generalization accuracy. In *International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=bZjxxYURKT.

Yan Sun, Li Shen, and Dacheng Tao. Understanding how consistency works in federated learning via stage-wise relaxed initialization. In *Advances in Neural Information Processing Systems*, volume 36, 2023c. doi: 10.52202/075280-3530. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/fef126561bbf9d4467dbb8d27334b8fe-Abstract-Conference.html.

Zhenyu Sun, Xiaochun Niu, and Ermin Wei. Understanding generalization of federated learning via stability: Heterogeneity matters. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 676–684. PMLR, 2024. URL https://proceedings.mlr.press/v238/sun24a.html.

Zhenyu Sun, Ziyang Zhang, Zheng Xu, Gauri Joshi, Pranay Sharma, and Ermin Wei. Debiasing federated learning with correlated client participation. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/464917b6103e074e1f9df7a2bf3bf6ba-Abstract-Conference.html.

Zhenheng Tang, Yonggang Zhang, Shaohuai Shi, Xinmei Tian, Tongliang Liu, Bo Han, and Xiaowen Chu. FedImpro: Measuring and improving client update in federated learning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/5b2fa23e4ef0f7ac6c4f01d7998e6237-Abstract-Conference.html.

Shiqiang Wang and Mingyue Ji. A unified analysis of federated learning with arbitrary client participation. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-1390. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/79ba1b827d3fc58e129d1cbfc8ff69f2-Abstract-Conference.html.

Jiahao Wu, Ming Hu, Yanxin Yang, Xiaofei Xie, ZeKai Chen, Chenyu Song, and Mingsong Chen. Rising from ashes: Generalized federated learning via dynamic parameter reset. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0524. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/16e71d1a24b98a02c17b1be1f634f979-Abstract-Conference.html.

Haoran Xu, Jiaze Li, Wanyi Wu, and Hao Ren. Federated learning with sample-level client drift mitigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 21752–21760, 2025. doi: 10.1609/aaai.v39i20.35480. URL https://ojs.aaai.org/index.php/AAAI/article/view/35480.

Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. FedALA: Adaptive local aggregation for personalized federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11237–11244, 2023. doi: 10.1609/aaai.v37i9.26330. URL https://ojs.aaai.org/index.php/AAAI/article/view/26330.

Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12878–12889. PMLR, 2021. URL https://proceedings.mlr.press/v139/zhu21b.html.

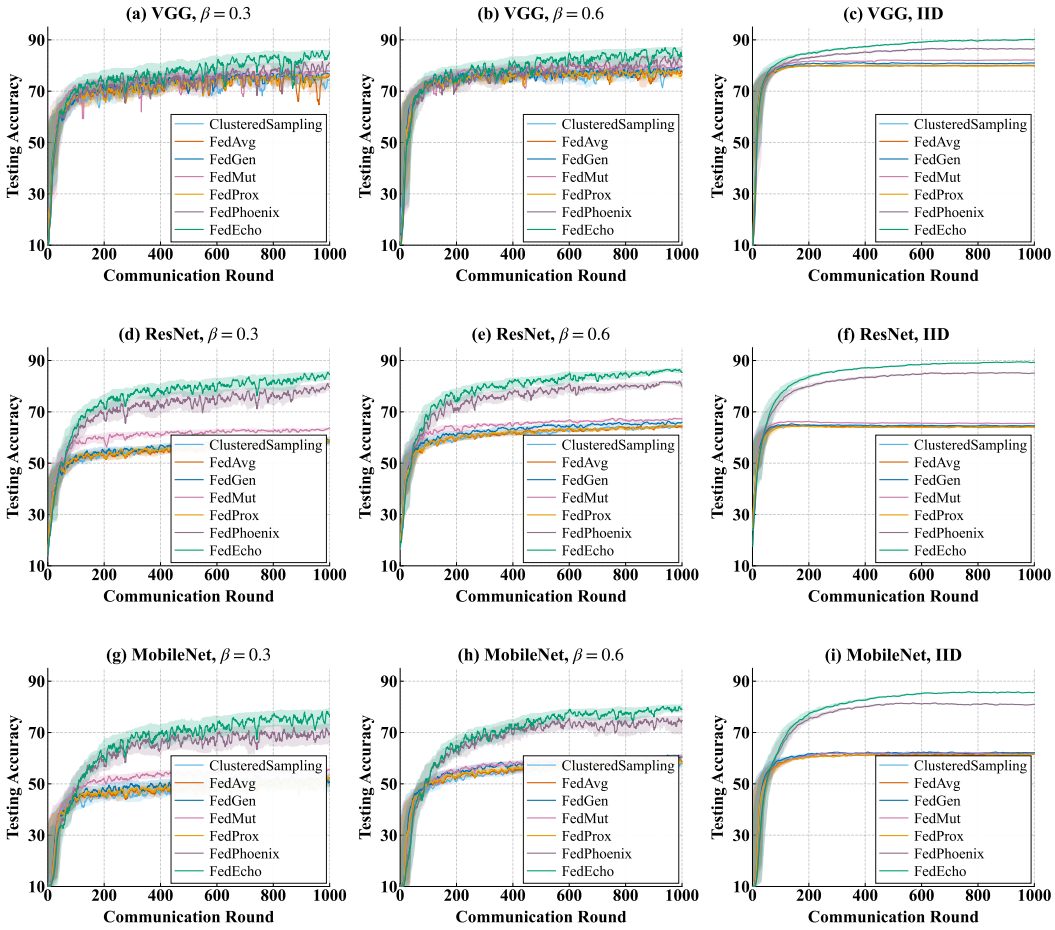

Figure 6: Learning curve comparison on CIFAR-10 across three network architectures and data distributions. Rows correspond to VGG-16, ResNet-18, and MobileNet-V1, from top to bottom. Columns correspond to Dirichlet distributions with concentration parameters $\beta = 0.3$ and $\beta = 0.6$, and an IID setting, from left to right.

A Learning Curves of Different FL Methods

To complement the quantitative results in the main text, Figure 6 reports the learning curves of all compared methods on CIFAR-10 across three network architectures and three data distributions. FedEcho achieves the highest test accuracy in all nine settings, with consistent gains under both non-IID and IID data partitions.

B Implementation and Response Estimation

B.1 Pooled temporal persistence estimation

Let $\mathcal{P}_t$ contain all pairs of consecutive valid response observations available before round t. For two valid observations of client i at rounds $s < q$, with responses δ_i^s and δ_i^q and participation gap $g = q - s$, the prediction model in equation 3 gives $\delta_i^q \approx \rho^g \delta_i^s$. We estimate the shared persistence coefficient as

$$\hat{\rho}_t = \arg\min_{\rho \in [0,1]} \sum_{(\delta, \delta', g) \in \mathcal{P}_t} \left(\delta' - \rho^g \delta\right)^2 . \tag{8}$$

Algorithm 1 FedEcho at communication round t

Require: Global model $\mathbf{w}_t$; previous global model $\mathbf{w}_{t-1}$ when $t > 0$; participants $\mathcal{S}_t$ with weights $\{p_i^t\}$; client histories $\{(\delta_i, \tau_i)\}$; response-pair pool $\mathcal{P}_t$; tolerance $\varepsilon_{\text{traj}}$

$\mathbf{m}_t \leftarrow \mathbf{0}$ if $t = 0$; otherwise $\mathbf{m}_t \leftarrow \mathbf{w}_t - \mathbf{w}_{t-1}$
if $\|\mathbf{m}_t\| < \varepsilon_{\text{traj}}$ **then**
  $\mathbf{\Delta}_i^t \leftarrow \mathbf{0}$ for all $i \in \mathcal{S}_t$
else
  Estimate $\hat{\rho}_t$ from $\mathcal{P}_t$ using equation 8
  for $i \in \mathcal{S}_t$ **do**
    $\hat{\delta}_i^t \leftarrow \hat{\rho}_t^{\,t-\tau_i}\delta_i$ if valid history exists; otherwise $\hat{\delta}_i^t \leftarrow 0$
    $\hat{\mathbf{r}}_i^t \leftarrow \hat{\delta}_i^t \mathbf{m}_t$
  end for
  Solve equation 4 for $\{\mathbf{\Delta}_i^t\}_{i \in \mathcal{S}_t}$
end if
for $i \in \mathcal{S}_t$ **in parallel do**
  Server sends $\mathbf{x}_i^t \leftarrow \mathbf{w}_t + \mathbf{\Delta}_i^t$
  Client returns $\mathbf{y}_i^t \leftarrow \mathcal{U}_i(\mathbf{x}_i^t; \mathcal{D}_i)$
end for
$\mathbf{u}_i^t \leftarrow \mathbf{y}_i^t - \mathbf{x}_i^t$ for all $i \in \mathcal{S}_t$
$\bar{\mathbf{u}}^t \leftarrow \sum_{i \in \mathcal{S}_t} p_i^t \mathbf{u}_i^t$
$\mathbf{w}_{t+1} \leftarrow \sum_{i \in \mathcal{S}_t} p_i^t \mathbf{y}_i^t$
$\mathcal{P}_{t+1} \leftarrow \mathcal{P}_t$
if $\|\mathbf{m}_t\| \geq \varepsilon_{\text{traj}}$ **then**
  for $i \in \mathcal{S}_t$ **do**
    Compute δ_i^t using equation 2
    Add the consecutive response pair to $\mathcal{P}_{t+1}$ if a previous valid observation exists
    $(\delta_i, \tau_i) \leftarrow (\delta_i^t, t)$
  end for
end if
return $\mathbf{w}_{t+1}$

This is a bounded one-dimensional problem and is solved once per communication round. If $\mathcal{P}_t$ is empty, we set $\hat{\rho}_t = 0$. For a returning client with valid history (δ_i, τ_i), the prediction is $\hat{\delta}_i^t = \hat{\rho}_t^{\,t-\tau_i}\delta_i$; a client without valid history uses $\hat{\delta}_i^t = 0$. When a new valid response is observed, the pair formed with the client's previous valid observation is added to $\mathcal{P}_{t+1}$ before the stored history is updated.

B.2 Numerical validity

Let $\varepsilon_{\text{traj}} > 0$ denote the fixed numerical tolerance used to determine whether the current global trajectory is valid. When $\|\mathbf{m}_t\| < \varepsilon_{\text{traj}}$, the response coordinate in equation 2 is not recorded, existing client histories remain unchanged, and the control space is treated as $\{\mathbf{0}\}$. Consequently, $\mathbf{\Delta}_i^t = \mathbf{0}$ for all participating clients in that round. This criterion prevents the normalization by $\|\mathbf{m}_t\|^2$ from amplifying numerical noise when the latest global displacement is negligible.

B.3 FedEcho procedure

Algorithm 1 summarizes the core rank-one procedure.

C Consensus-preserving Dispatch Optimization

For compactness, we omit the round superscript in this section and write $p_i = p_i^t$, $\mathbf{\Delta}_i = \mathbf{\Delta}_i^t$, and $\hat{\mathbf{r}}_i = \hat{\mathbf{r}}_i^t$.

C.1 UNCONSTRAINED SOLUTION

For each client,

$$\int_0^1 \|\boldsymbol{\Delta}_i + \kappa \hat{\mathbf{r}}_i\|^2 \, d\kappa = \|\boldsymbol{\Delta}_i\|^2 + \langle \boldsymbol{\Delta}_i, \hat{\mathbf{r}}_i \rangle + \frac{1}{3}\|\hat{\mathbf{r}}_i\|^2. \tag{9}$$

The last term is independent of the dispatch variables. When the trust-region constraint is inactive, equation 4 therefore reduces to

$$\min_{\{\boldsymbol{\Delta}_i \in \mathcal{V}_t\}} \sum_{i \in \mathcal{S}_t} p_i \left(\|\boldsymbol{\Delta}_i\|^2 + \langle \boldsymbol{\Delta}_i, \hat{\mathbf{r}}_i \rangle \right) \quad \text{s.t.} \quad \sum_{i \in \mathcal{S}_t} p_i \boldsymbol{\Delta}_i = \mathbf{0}. \tag{10}$$

Since $\hat{\mathbf{r}}_i \in \mathcal{V}_t$, introducing a Lagrange multiplier $\boldsymbol{\lambda} \in \mathcal{V}_t$ gives

$$2\boldsymbol{\Delta}_i + \hat{\mathbf{r}}_i + \boldsymbol{\lambda} = \mathbf{0}. \tag{11}$$

Using $\sum_i p_i = 1$ and the zero-sum constraint yields $\boldsymbol{\lambda} = -\bar{\mathbf{r}}$, where $\bar{\mathbf{r}} = \sum_{i \in \mathcal{S}_t} p_i \hat{\mathbf{r}}_i$. Hence

$$\boldsymbol{\Delta}_i^* = -\frac{1}{2} \left(\hat{\mathbf{r}}_i - \bar{\mathbf{r}} \right), \tag{12}$$

which recovers equation 5. The cohort centering is induced by the zero-sum constraint, while the factor $1/2$ follows from the linear path surrogate and the uniform integration over $\kappa \in [0, 1]$.

C.2 TRUST-REGION-CONSTRAINED RANK-ONE SOLUTION

For the core rank-one method, write $\boldsymbol{\Delta}_i = \alpha_i \mathbf{m}_t$ and $\hat{\mathbf{r}}_i = \beta_i \mathbf{m}_t$, where $\beta_i = \hat{\delta}_i^t$. For $\mathbf{m}_t \neq \mathbf{0}$, the trust region becomes $|\alpha_i| \leq 1$, and the consensus constraint becomes $\sum_i p_i \alpha_i = 0$. After dropping terms independent of $\{\alpha_i\}$ and the common factor $\|\mathbf{m}_t\|^2$, the problem is

$$\min_{\{\alpha_i\}} \sum_{i \in \mathcal{S}_t} p_i \left(\alpha_i + \frac{\beta_i}{2} \right)^2 \quad \text{s.t.} \quad \sum_{i \in \mathcal{S}_t} p_i \alpha_i = 0, \qquad -1 \leq \alpha_i \leq 1. \tag{13}$$

For a scalar multiplier λ associated with the equality constraint, the KKT solution is

$$\alpha_i(\lambda) = \operatorname{clip}\left(-\frac{\beta_i + \lambda}{2}, -1, 1 \right). \tag{14}$$

The optimal multiplier satisfies

$$\sum_{i \in \mathcal{S}_t} p_i \alpha_i(\lambda) = 0. \tag{15}$$

The left-hand side is continuous and non-increasing in λ, so a root can be obtained by scalar bisection. The final dispatch is $\boldsymbol{\Delta}_i^* = \alpha_i^* \mathbf{m}_t$. If no box constraint is active, $\lambda = -\sum_i p_i \beta_i$, and equation 14 reduces to equation 12.

D ADDITIONAL ANALYSIS AND STRUCTURED CONTROL

D.1 PATH DISAGREEMENT AND GRADIENT MISMATCH

Let $\mathbf{y}_{i,0}^t = \mathbf{x}_i^t$ and let $\mathbf{y}_{i,k}^t$ be the model of client i after local step k, and define the weighted consensus path $\bar{\mathbf{y}}_k^t = \sum_{i \in \mathcal{S}_t} p_i^t \mathbf{y}_{i,k}^t$. If every F_i is L-smooth, then

$$\begin{aligned} \left\| \sum_{i \in \mathcal{S}_t} p_i^t \left[\nabla F_i(\mathbf{y}_{i,k}^t) - \nabla F_i(\bar{\mathbf{y}}_k^t) \right] \right\|^2 &\leq \sum_{i \in \mathcal{S}_t} p_i^t \left\| \nabla F_i(\mathbf{y}_{i,k}^t) - \nabla F_i(\bar{\mathbf{y}}_k^t) \right\|^2 \\ &\leq L^2 \sum_{i \in \mathcal{S}_t} p_i^t \left\| \mathbf{y}_{i,k}^t - \bar{\mathbf{y}}_k^t \right\|^2. \end{aligned} \tag{16}$$

The first inequality follows from convexity of the squared norm and the second from L-smoothness, recovering equation 7.

For a common local horizon of K steps, define

$$D_k^t = \sum_{i \in \mathcal{S}_t} p_i^t \left\| \mathbf{y}_{i,k}^t - \bar{\mathbf{y}}_k^t \right\|^2 . \tag{17}$$

Accumulating equation 16 over local steps motivates the normalized quantity $K^{-1} \sum_{k=0}^{K-1} D_k^t$. With $\kappa = k/K$, this discrete quantity admits a continuous approximation over $\kappa \in [0, 1]$. Since the future local paths are unavailable at dispatch time, FedEcho models only their predicted controllable component as $\boldsymbol{\Delta}_i^t + \kappa \hat{\mathbf{r}}_i^t$.

Under the zero-sum constraint, the weighted mean of this predicted component is $\kappa \bar{\mathbf{r}}^t$. Therefore,

$$\sum_{i \in \mathcal{S}_t} p_i^t \|\boldsymbol{\Delta}_i^t + \kappa \hat{\mathbf{r}}_i^t\|^2 = \sum_{i \in \mathcal{S}_t} p_i^t \|\boldsymbol{\Delta}_i^t + \kappa(\hat{\mathbf{r}}_i^t - \bar{\mathbf{r}}^t)\|^2 + \kappa^2 \|\bar{\mathbf{r}}^t\|^2. \tag{18}$$

Hence the objective in equation 4 is equivalent, up to a term independent of the dispatch variables, to the predicted disagreement around the consensus path. It therefore targets a low-dimensional proxy for the path-disagreement quantity on the right-hand side of equation 16. Reducing this surrogate does not, by itself, imply monotonic improvement of the global objective.

D.2 Structured rank-r control space

Let a fixed construction decompose the current global trajectory into pairwise orthogonal components,

$$\mathbf{m}_t = \sum_{a=1}^{r} \mathbf{d}_{t,a}, \qquad \langle \mathbf{d}_{t,a}, \mathbf{d}_{t,b} \rangle = 0 \quad (a \neq b), \tag{19}$$

and define $\mathcal{V}_t = \operatorname{span}\{\mathbf{d}_{t,a}\}_{a=1}^{r}$. In each round, only components with $\|\mathbf{d}_{t,a}\| \geq \varepsilon_{\text{traj}}$ are used for response estimation and control; invalid components are excluded from $\mathcal{V}_t$, and their histories remain unchanged. For every valid component,

$$\delta_{i,a}^t = \frac{\langle \mathbf{u}_i^t - \bar{\mathbf{u}}^t, \mathbf{d}_{t,a} \rangle}{\|\mathbf{d}_{t,a}\|^2}. \tag{20}$$

The server stores the latest valid response and observation round $(\delta_{i,a}, \tau_{i,a})$ for each component and predicts

$$\hat{\delta}_{i,a}^t = \hat{\rho}_t^{\,t-\tau_{i,a}} \delta_{i,a}, \qquad \hat{\mathbf{r}}_i^t = \sum_{a=1}^{r} \hat{\delta}_{i,a}^t \mathbf{d}_{t,a}. \tag{21}$$

The persistence coefficient $\hat{\rho}_t$ is shared across components and is estimated from valid consecutive coordinate pairs pooled across clients and components.

Writing

$$\boldsymbol{\Delta}_i^t = \sum_{a=1}^{r} \alpha_{i,a}^t \mathbf{d}_{t,a}, \tag{22}$$

orthogonality gives the coordinate-wise consensus constraints

$$\sum_{i \in \mathcal{S}_t} p_i^t \alpha_{i,a}^t = 0, \qquad a = 1, \ldots, r, \tag{23}$$

while the trust region becomes

$$\sum_{a=1}^{r} \left(\alpha_{i,a}^t\right)^2 \|\mathbf{d}_{t,a}\|^2 \leq \|\mathbf{m}_t\|^2. \tag{24}$$

When the trust region is inactive, the solution remains $\boldsymbol{\Delta}_i^{t,*} = -\frac{1}{2}(\hat{\mathbf{r}}_i^t - \bar{\mathbf{r}}^t)$. When it is active, equation 4 reduces to a low-dimensional convex problem in $\{\alpha_{i,a}^t\}$; because equation 24 couples the coordinates of each client, the coefficients are solved jointly rather than by independent coordinate-wise clipping.

D.3 CONVOLUTION-AWARE RANK-TWO INSTANCE

For a convolutional parameter block $\mathbf{Z}^{(\ell)} \in \mathbb{R}^{C_{\mathrm{out}} \times C_{\mathrm{in}} \times H_\ell \times W_\ell}$, define the spatial mean of each input–output channel pair as

$$\mu_{o,c}^{(\ell)}(\mathbf{Z}) = \frac{1}{H_\ell W_\ell} \sum_{h=1}^{H_\ell} \sum_{w=1}^{W_\ell} Z_{o,c,h,w}^{(\ell)}. \tag{25}$$

The operator P_{sp} removes this spatial mean,

$$\left(P_{\mathrm{sp}} \mathbf{Z}^{(\ell)}\right)_{o,c,h,w} = Z_{o,c,h,w}^{(\ell)} - \mu_{o,c}^{(\ell)}(\mathbf{Z}), \tag{26}$$

and is zero on non-convolutional parameters. Thus P_{sp} is the orthogonal projection onto the subspace of convolutional parameters with zero spatial mean.

Applying this projection to the current global trajectory gives

$$\mathbf{d}_{t,\mathrm{sp}} = P_{\mathrm{sp}} \mathbf{m}_t, \qquad \mathbf{d}_{t,\mathrm{base}} = (I - P_{\mathrm{sp}}) \mathbf{m}_t. \tag{27}$$

By construction,

$$\mathbf{d}_{t,\mathrm{sp}} + \mathbf{d}_{t,\mathrm{base}} = \mathbf{m}_t, \qquad \langle \mathbf{d}_{t,\mathrm{sp}}, \mathbf{d}_{t,\mathrm{base}} \rangle = 0. \tag{28}$$

The resulting convolution-aware control space is

$$\mathcal{V}_t^{\mathrm{CNN}} = \mathrm{span}\{\mathbf{d}_{t,\mathrm{base}}, \mathbf{d}_{t,\mathrm{sp}}\}. \tag{29}$$

Since P_{sp} is zero on non-convolutional parameters, all non-convolutional coordinates belong to $\mathbf{d}_{t,\mathrm{base}}$. Each numerically valid component maintains its own response coordinate and observation round according to equation 20–equation 21. The rank-two construction changes only the structured control basis; response prediction, consensus-preserving dispatch, local optimization, and global aggregation remain unchanged.