# FEDFFT: TAMING CLIENT DRIFT IN FEDERATED SAM VIA SPECTRAL PERTURBATION FILTERING

**Anonymous authors** 

Paper under double-blind review

## **ABSTRACT**

Federated Learning (FL) enables decentralized training without data sharing, but suffers from statistical heterogeneity across clients, leading to client drift, poor generalization, and sharp minima compared to centralized training. Sharpness-Aware Minimization (SAM) has emerged as a promising approach to improve generalization, yet its application in federated learning still suffers from divergence problems, since perturbations are computed locally and reflect client-specific loss geometries. To better understand this issue, we provide analysis from a new perspective, the frequency domain, for SAM perturbations in federated settings, revealing that inter-client perturbation inconsistencies are predominantly concentrated in the low-frequency spectrum. Motivated by this insight, we propose **Fed**erated learning with **F**requency-domain **F**iltering of SAM per**t**urbations (**FedFFT**). It is a lightweight and plug-and-play method that filters out lowfrequency components of SAM perturbations without requiring additional communication, thereby suppressing inconsistent components in client updates while preserving consistent learning signals. Extensive experiments across multiple benchmarks and diverse backbones demonstrate that FedFFT consistently outperforms SAM-based FL methods, particularly under severe non-IID distributions. These results highlight the effectiveness, scalability, and general applicability of our frequency-domain perspective for sharpness-aware federated optimization.

# 1 Introduction

Federated Learning (FL) (McMahan et al., 2017) is a distributed learning paradigm where multiple clients collaboratively train a global model under the coordination of a central server, while keeping their raw data local to preserve privacy. In each communication round, clients perform local training and only transmit model parameters or updates, which are aggregated to update the global model. This framework has been widely applied in privacy-sensitive domains such as healthcare, finance, mobile applications, and autonomous systems (Antunes et al., 2022; Rauniyar et al., 2024; Fantauzzo et al., 2022). However, the practical effectiveness of FL is severely hampered by the statistical heterogeneity inherent in real-world data, where the local data distributions across clients are typically non-independent and identically distributed (Non-IID). This causes the optimization objectives of individual clients to become misaligned with one another and with the global goal, resulting in local updates that pull the shared model in conflicting directions. This phenomenon are known as client drift (Karimireddy et al., 2021; Woodworth et al., 2020; Li et al., 2020a; Fan et al., 2022), which not only slows convergence but can also limits the model's ability to generalize to the overall underlying distribution.

To address the generalization challenges posed by client drift, a promising research direction has shifted from traditional client-side regularization (Acar et al., 2021; Karimireddy et al., 2021; Li et al., 2020b; 2021; Xu et al., 2021) or aggregation methods (Ye et al., 2023; Li et al., 2023; Shi et al., 2025) to exploring the geometry of the loss landscape. These methods build upon the insight that convergence to sharp minima correlates with poor generalization (Li et al., 2020a), where flatter minima often yield better performance. Sharpness-Aware Minimization (SAM) (Foret et al., 2020) is a representative technique designed for this purpose, seeking flatter regions by optimizing a perturbed loss. Building on this, FedSAM (Qu et al., 2022; Caldarola et al., 2022) pioneered the application of SAM to local training in Federated Learning. While FedSAM (Sun et al., 2023a; Fan et al., 2024) have shown strong performance across different settings, they primarily focus on

local flatness, implicitly assuming that minimizing sharpness locally leads to a globally flat minimum. In practice, however, under substantial data heterogeneity, the local and global loss landscapes may diverge considerably, and improvements in local flatness do not necessarily guarantee global flatness. Subsequent works (Sun et al., 2023a; Fan et al., 2024; Caldarola et al., 2025; Dai et al., 2024) have attempted to bridge this local-global gap through various strategies, such as enhancing client-side updates (Sun et al., 2023a) or shifting the sharpness optimization to the server (Caldarola et al., 2025), sometimes employing complex frameworks like the Alternating Direction Method of Multipliers (ADMM) (Boyd et al., 2011). Despite these advances, existing solutions face a difficult trade-off: purely client-side methods struggle with the deviation of local sharpness estimates, while server-centric approaches often come at the cost of significant communication or computational overhead. Crucially, none of these approaches explicitly investigate the intrinsic nature of the perturbations themselves, leaving open the question of whether they can be refined to better align clients with the global objective.

In this work, we address this question by introducing a novel frequency-domain perspective. To our knowledge, we are the first to systematically analyze the spectral properties of client-side SAM perturbations in FL. Our key finding is that the inter-client disagreements are not random noise; they are predominantly concentrated in the low-frequency spectrum, which we hypothesize is strongly tied to client-specific data biases. Motivated by this insight, we propose Federated learning with Frequency-domain Filtering of SAM perturbations (FedFFT), a lightweight and plug-and-play method. FedFFT applies a high-pass filter to the locally computed perturbations, systematically removing the discordant low-frequency components while preserving consistent learning signals in the higher frequencies. This alignment in the spectral domain is achieved without requiring any additional communication overhead. Notably, FedFFT can be seamlessly integrated as a plug-andplay module with federated learning framework that utilizes client-side SAM optimizers, further broadening its applicability. Our contributions are summarized as below: (1) Frequency-domain analysis of perturbations. We provide the first study of SAM perturbations in FL across clients based on spectral decomposition, and reveal that heterogeneity is primarily concentrated in the lowfrequency bands. (2) Algorithm design. Based on this insight, we introduce FedFFT, a simple yet effective approach that filters out low-frequency perturbation components to suppress inconsistent client updates while retaining retaining consistent learning signals. (3) Extensive empirical validation. We conduct comprehensive experiments across multiple benchmarks and backbones, under varying degrees of data heterogeneity. The results show that FedFFT outperforms related baselines, particularly in highly non-IID settings, demonstrating both effectiveness and scalability.

# 2 RELATED WORK

Sharpness-Aware Minimization (SAM). The connection between generalization and flat minima was first recognized in early studies (Hochreiter & Schmidhuber, 1994), and later work confirmed that smoother loss landscapes generally lead to better generalization (Keskar et al., 2017; Neyshabur et al., 2017). Building on this insight, Sharpness-Aware Minimization (SAM) was introduced as a PAC-Bayesian inspired method that explicitly minimizes loss sharpness and achieves strong generalization across image classification benchmarks (Foret et al., 2020). Since then, numerous extensions have been developed. A scale-invariant version improves training stability (Kwon et al., 2021), while another reformulates sharpness from both theoretical and intuitive perspectives (Zhuang et al., 2022). Further studies focus on perturbation strategies, including adaptive or random amplitudes (Liu et al., 2022; Ahn et al., 2024), dynamic adjustment through DSAM (Chen et al., 2024), and variance reduction across domains with DISAM (Zhang et al., 2024).

Federated Learning. Federated Learning (FL), introduced with FedAvg (McMahan et al., 2017), enables collaborative model training without raw data sharing. While preserving privacy, this decentralized design exacerbates the *client-drift problem*—the divergence between local and global updates—mainly due to non-IID data and multi-step local errors (Acar et al., 2021; Woodworth et al., 2020; Li et al., 2020a). Limited client participation further aggravates drift and degrades performance. To mitigate client drift, existing methods can be broadly grouped into two categories: (i) *local objective regularization*, which modifies local training to align client updates with the global objective, such as SCAFFOLD (Karimireddy et al., 2021), FedProx (Li et al., 2020b) and FedDyn (Acar et al., 2021); and (ii) *modified aggregation strategies*, which design more robust global update rules beyond simple averaging, such as FedAWA (Shi et al., 2025), FedLAW (Li et al., 2023), and

FedDisco (Ye et al., 2023). While these approaches improve optimization stability, they are primarily rooted in empirical risk minimization and often overlook the relationship between the global loss landscape and generalization ability. This motivates a new research line that leverages SAM in FL.

SAM in Federated Learning. FedSAM (Qu et al., 2022; Caldarola et al., 2022) first brought SAM into FL by applying local perturbations to improve generalization. Subsequent variants extended this idea. For example, FedSpeed (Sun et al., 2023b) used an Alternating Direction Method of Multipliers (ADMM) framework to enhance communication efficiency. PLGU (Qu et al., 2023) and FedSOL (Lee et al., 2024), explored layer-wise perturbation and proximal-based corrections. FedGAMMA (Dai et al., 2024) introduced variance-reduction techniques to align client updates from a global perspective. FedSMOO (Sun et al., 2023a) reduced inconsistency by correcting both updates and perturbations through ADMM, while FedLESAM (Fan et al., 2024) introduced global perturbations to better guide local training. FedFSA (Xing et al., 2025) focused on parameter sensitivity, applying stronger perturbations only to the most sensitive layers to balance convergence and generalization. FedGloSS (Caldarola et al., 2025) shifted attention from local sharpness to global flatness by applying SAM on the server, highlighting the importance of global geometry in federated optimization. Unlike these approaches, our method is motivated by a novel spectral analysis, revealing inter-client disagreement in low-frequency components, which we filter to produce consistent perturbations and flatter minima.

# 3 BACKGROUND

## 3.1 SHARPNESS-AWARE MINIMIZATION

To improve model generalization and robustness, modern optimization methods have shifted focus from merely finding solutions with low training loss to finding solutions that reside in flat minima of the loss landscape. SAM (Foret et al., 2020) is a leading technique for this purpose. It jointly minimizes the loss value and the sharpness by solving the following minimax objective:

$$\min_{w} \max_{\|\delta\| \le \rho} \mathcal{L}(w + \delta),\tag{1}$$

where  $\mathcal{L}(\cdot)$  is the empirical loss on the training data, and  $\rho$  is the neighborhood size. In practice, the inner maximization is approximated with a single step of gradient ascent. The full optimization process for parameter w involves two steps: (1) Compute the perturbation that approximately maximizes the loss:  $\delta^*(w) = \rho \frac{\nabla \mathcal{L}(w)}{\|\nabla \mathcal{L}(w)\|_2}$ ; (2) Update the model parameters using the gradient at the perturbed point:  $w \leftarrow w - \eta \nabla \mathcal{L}(w + \delta^*(w))$ . This procedure encourages the optimizer to converge to flat minima, which are empirically linked to better generalization performance.

#### 3.2 Federated Learning

FL is a distributed learning paradigm that enables training a global model on data from K clients, coordinated by a central server, without centralizing the private client datasets  $\mathcal{D}_k$ . The core objective in FL is to minimize the global empirical risk F(w), defined as the weighted average of the local empirical losses  $f_k(w)$ :

$$\min_{w} F(w) := \frac{1}{K} \sum_{k=1}^{K} f_k(w), \quad \text{where} \quad f_k(w) = \frac{1}{|\mathcal{D}_k|} \sum_{(x,y) \in \mathcal{D}_k} \mathcal{L}_k(w; x, y). \tag{2}$$

The widely-adopted FedAvg algorithm (McMahan et al., 2017) solves this objective via iterative communication rounds. In each round, the server (1) **Broadcasts** the global model  $w_t$  to clients. Clients then perform (2) **Local Updates** on their data to produce  $w_k^{t+1}$ . These are (3) **Uploaded** to the server for (4) **Aggregation** into the new global model  $w_{t+1} = \sum_k p_k w_k^{t+1}$ .

A key challenge in FL is **data heterogeneity**, where client data distributions are Non-Independent and Identically Distributed (Non-IID). This causes the local objectives  $f_k(w)$  to be inconsistent with one another, leading to misaligned loss landscapes and the "client drift" phenomenon, which poses a significant challenge to training a robust global model.

#### 3.3 SAM IN FEDERATED LEARNING

Given the challenge of training on misaligned local landscapes, a natural strategy is to seek solutions in flat minima. This motivates applying SAM to FL, known as FedSAM (Qu et al., 2022), which incorporates SAM into each client's local training. The local and global objectives are:

$$\min_{w} F^{\text{SAM}}(w) := \frac{1}{K} \sum_{k=1}^{K} f_k^{\text{SAM}}(w), \quad \text{where} \quad f_k^{\text{SAM}}(w) = \max_{\|\delta_k\| \le \rho} f_k(w + \delta_k). \tag{3}$$

Compared with FedAvg, FedSAM differs in that it optimizes the sharpness-aware local objectives rather than the original local losses. While FedSAM encourages convergence to locally flat regions, it does not guarantee a flat global landscape. Under data heterogeneity, the locally computed perturbation vectors  $\delta_k$ , which capture the directions of sharpness, can themselves diverge. This raises a critical question that existing works have not fully explored: **what is the underlying structure of these inter-client perturbation disagreements?** Understanding this is key to mitigating their negative impact, which directly motivates our work.

#### 4 Our Proposed Method

In this section, we introduce our proposed method, Federated learning with Frequency domain Filtering of SAM perturbations (FedFFT). We begin by elaborating on the motivation stemming from our observations of client perturbations in federated environments. Subsequently, we provide a detailed description of the FedFFT algorithm.

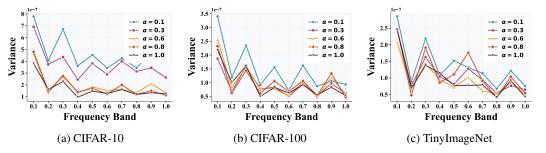


Figure 1: Frequency Domain Analysis of Client Perturbations. (a-c) Variance of SAM perturbations across clients under varying data heterogeneity ( $\alpha$ ) on three datasets. Across all settings, inter-client variance is consistently concentrated in the low-frequency components, indicating that client disagreement is predominantly a low-frequency phenomenon.

#### 4.1 MOTIVATION: ANALYZING CLIENT PERTURBATIONS IN THE FREQUENCY DOMAIN

In FL, non-IID data distributions among clients are the primary cause of the client drift phenomenon. While Sharpness-Aware Minimization (SAM) is a powerful technique for improving generalization, its application in FL may amplify this issue. Since SAM perturbations are computed locally, they intrinsically reflect the geometry of client-specific loss landscapes, causing the perturbation vectors themselves to diverge. However, the underlying structure of these perturbation divergences remains poorly understood. To bridge this gap, we introduce a novel diagnostic approach.

To the best of our knowledge, this work is the first to employ frequency-domain analysis to systematically investigate the nature of SAM perturbation disagreements in a federated context. We treat each client's perturbation vector as a signal and use the Real-valued Fast Fourier Transform (RFFT) to observe its characteristics. Our analysis is conducted on a per-layer basis to respect the model's architectural integrity, with the full implementation details provided in below Section 4.2. For each layer, we compute the variance across clients within different frequency bands. As illustrated in Figure 1, which averages these variances across all layers, we discover a clear pattern:

The primary disagreements are concentrated in the low-frequency components, while the disagreements in high-frequency components are much smaller. This finding suggests that the low-frequency

parts are highly correlated with client-specific biases, whereas the high-frequency parts may represent more consistent features of the shared learning task.

Based on this key insight, we formulate our central hypothesis: by filtering out the discordant low-frequency components while preserving the consistent high-frequency information, we can mitigate client drift and improve the global model's performance.

#### 4.2 FEDFFT: THE PROPOSED METHOD

Considering that inter-client disagreements are predominantly a low-frequency phenomenon, we propose **Fed**erated learning with **F**requency-domain **F**iltering of SAM perturbations (**FedFFT**). Our method refines the client-side SAM update by integrating a frequency filtering module. This module is designed as a lightweight, plug-and-play replacement for the standard perturbation calculation within any federated learning framework that employs SAM-based optimizers on client devices.

Let's consider a single local update step for a client k at communication round  $t \in [1, T]$  and local iteration  $e \in [1, E]$ . For simplicity, we omit the round and local iteration indices. Now, we first need to compute the standard SAM perturbation  $\delta^k$  for client k like FedSAM, denoted as

$$\delta^k = \rho \cdot \frac{\nabla \mathcal{L}_k(w^k)}{\|\nabla \mathcal{L}_k(w^k)\|_2},\tag{4}$$

where  $\delta^k$  has the same size with model parameter  $w^k$  and includes all layer-wise perturbations  $\delta^k_{1:L}$  and  $\delta^k_l$  is the perturbation of parameter  $w^k_l$  at layer l. The core idea of FedFFT is to **selectively discard the discordant low-frequency components** of the SAM perturbation while preserving the high-frequency components that capture more consistent aspects of the sharpness landscape. This filtering is performed on a per-layer basis to respect the model's architectural integrity. For any given weight layer l with parameters  $w^k_l$ , the FedFFT procedure is as follows:

1. **Transform to Frequency Domain**: Instead of directly applying this perturbation, we transform it into the frequency domain. We flatten the perturbation tensor  $\delta^k_l$  into a vector  $\mathbf{v}^k_l$  and apply the Real-valued Fast Fourier Transform (rFFT):

$$\hat{\mathbf{v}}_l^k = \text{rFFT}(\mathbf{v}_l^k),\tag{5}$$

where  $\hat{\mathbf{v}}_l^k$  is the frequency-domain representation of the perturbation.

 2. **Apply High-Pass Filter**: Next, we apply a high-pass filter operator,  $\mathcal{H}_r(\cdot)$ , which zeroes out the lowest r fraction of frequency coefficients. Given a truncation ratio  $r \in [0, 1)$ , the filtering operation is defined as:

$$[\mathcal{H}_r(\hat{\mathbf{v}}_l^k)]_m = \begin{cases} 0, & \text{if } m < \lfloor r \cdot \text{len}(\hat{\mathbf{v}}_l^k) \rfloor, \\ [\hat{\mathbf{v}}_l^k]_m, & \text{otherwise,} \end{cases}$$
 (6)

where m indexes the frequency coefficients in ascending order. This step is the crux of our method, as it explicitly removes the client-specific biases encoded in the low-frequency domain.

3. **Reconstruct Filtered Perturbation**: We then transform the filtered vector back into the parameter domain using the inverse rFFT (iRFFT) and reshape it to its original tensor shape to obtain the refined perturbation  $\tilde{\delta}_l$ :

$$\tilde{\delta}_l^k = \text{reshape}(iRFFT(\mathcal{H}_r(\hat{\mathbf{v}}_l^k))). \tag{7}$$

For simplicity, for any given layer's perturbation tensor  $\delta_l^k$  at client k, we summarize the three steps above as a filtering operation:

$$\tilde{\delta}_l^k = \text{Filter}(\delta_l^k, r). \tag{8}$$

Finally, the local model in client k is updated using this filtered perturbation, following the standard SAM procedure:

$$w^k \leftarrow w^k - \eta \cdot \nabla \mathcal{L}_k(w^k + \tilde{\delta}^k), \tag{9}$$

where  $\tilde{\delta}^k$  is the collection of all layer-wise filtered perturbations  $\tilde{\delta}^k_l$ .

By replacing the standard SAM perturbation with our filtered version, FedFFT forces the local optimizer to ignore the most heterogeneous directions of sharpness, thereby promoting greater consistency among client updates and facilitating convergence to a more robust global minimum. As shown in, we provide the workflow of FedAvg, FedSAM and our FedFFT in Algorithm 1. Benefit from the effective and general filtering method, ours can also be combined with some classifical or advanced FL-based methods; please see more details in Appendix A.

# 5 EXPERIMENTS

270

271

272

273

274

275276277

278279

280 281

282

283

284

285

287

288

289

290

291

292

293

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315 316

317318

319

320

321

322

323

# 5.1 EXPERIMENTAL SETUPS

**Datasets and Baselines.** We evaluate on CIFAR-10 (Krizhevsky & Hinton, 2009), CIFAR-100 (Krizhevsky & Hinton, 2009), and Tiny-ImageNet (CS231N, 2015). To simulate data heterogeneity, we partition datasets using a Dirichlet distribution with  $\alpha \in 0.1, 0.6$ , where smaller  $\alpha$  yields more non-IID distributions and larger  $\alpha$  yields more IID ones.

We benchmark FedFFT against a comprehensive suite of baselines in three main categories: (i) foundational FL algorithms, including FedAvg (McMahan et al., 2017), SCAFFOLD (Karimireddy et al., 2021), and FedDyn (Acar et al., 2021); (ii) the direct application of SAM in FL, namely FedSAM and its momentum-enhanced variant MoFedSAM (Caldarola et al., 2022; Qu et al., 2022); and (iii) advanced FL-SAM variants that aim to improve consistency, such as FedGAMMA (Dai et al., 2024), FedSMOO (Sun et al., 2023a), FedLESAM (Fan et al., 2024), and Fed-GloSS (Caldarola et al., 2025). To ensure fair comparisons, we integrate our approach with these foundational algorithms following FedLESAM, yielding FedFFT

# Algorithm 1 FedAvg, FedSAM and Our FedFFT

**Require:** Communication rounds T, local epochs E, perturbation radius  $\rho$ , local learning rate  $\eta$ , frequency truncation ratio r.

```
Ensure: Global model w_q^T
 1: Initialize global model w_a^0
 2: for t = 0 to T - 1 do
 3:
            Randomly select active client set S_t.
            for all clients k \in S_t in parallel do w^{k,t,0} \leftarrow w_g^t
 4:
 5:
                  for e = 0 to E - 1 do
 6:
 7:

    ▶ perturbation stage

                         FedAvg: \delta^{k,t,e} = 0
 8:
                        9:
                       \begin{aligned} & & \textbf{FedFFT:} \ \delta_{1:L}^{k,t,e} \!=\! \text{Filter}(\delta_{1:L}^{k,t,e},r) \\ & w^{k,t,e+1} \!=\! w^{k,t,e} \!\!-\! \eta \mathcal{L}_k(w^{k,t,e} \!\!+\! \delta^{k,t,e}) \end{aligned}
10:
11:
12:
                  Send local model w^{k,t,E} to server.
13:
14:
           w_g^{t+1} \leftarrow \frac{1}{|S_t|} \sum_{k \in S_t} w^{k,t,E}
15:
16: end for
17: return w_a^T
```

(FedAvg-based), FedFFT-S (SCAFFOLD-based), and FedFFT-D (FedDyn-based). We summarize the characteristics of SAM in FL methods; please refer to the Appendix B for details.

**Implementation Details.** Following prior works (Sun et al., 2023a; Fan et al., 2024), we adopt ResNet-18 (He et al., 2016) from the PyTorch model zoo (Paszke et al., 2019) as the backbone. We use the following settings: 100 clients with about 10% sampled per round, local/global learning rates of 0.1/1.0, 5 local epochs, and up to 800/800/300 communication rounds for CIFAR-10, CIFAR-100, and TinyImageNet. For SAM-based methods, we use a perturbation radius of 0.1 with SGD, weight decay 1e-3, and exponential LR decay (0.998 per round). For FedFFT, the frequency truncation ratio is set to 0.01. Further details are in Appendix C.

#### 5.2 Main Results

Comparison with State-of-the-Art Baselines. Table 1 presents the results on CIFAR-10, CIFAR-100, and Tiny-ImageNet with ResNet-18. Overall, FedFFT delivers consistent improvements under both SCAFFOLD and FedDyn frameworks, indicating that frequency-domain perturbation modeling complements existing federated optimization paradigms. On CIFAR-10, FedFFT-D achieves the best performance under both Dirichlet partitions, demonstrating its effectiveness in standard federated settings. For CIFAR-100, while FedFFT-D is slightly outperformed by FedLESAM-S under  $\alpha=0.6$ , it regains superiority in the more heterogeneous  $\alpha=0.1$  case. This suggests that suppress-

Table 1: Test accuracy comparison (%) of different methods on CIFAR-10, CIFAR-100 and Tiny-ImageNet, with ResNet-18. "-S" and "-D" means using SCAFFOLD and FedDyn as the base algorithms. All results on Tiny-ImageNet are reproduced by us. Results marked with † are reproduced by us for CIFAR-10 and CIFAR-100. Others are reported from (Sun et al., 2023a) and (Fan et al., 2024).

Method	CIFAR-10		CIFAR-100		Tiny-ImageNet	
	$\alpha = 0.6$	$\alpha = 0.1$	$\alpha = 0.6$	$\alpha = 0.1$	$\alpha = 0.6$	$\alpha = 0.1$
FedAvg	79.52	76.00	46.35	42.64	28.31	27.48
FedSAM†	81.91	74.92	48.08	45.53	33.16	29.46
MoFedSAM	84.13	78.71	54.38	44.85	33.50	29.77
FedLESAM	81.04	76.93	47.92	44.48	27.91	26.91
Our FedFFT	83.02	77.53	48.59	46.83	33.58	30.43
SCAFFOLD	81.81	78.57	51.98	44.41	35.34	32.11
FedSAM-S†	83.88	76.68	50.19	49.14	35.84	31.73
FedGamma-S	82.64	78.95	53.41	46.39	36.85	30.09
FedLESAM-S	84.94	79.52	54.61	48.07	28.47	27.70
Our FedFFT-S	84.69	79.24	52.75	<u>49.85</u>	36.15	33.08
FedDyn	83.22	78.08	50.82	42.50	28.01	24.19
FedSAM-D†	82.29	79.11	53.70	46.28	38.18	31.39
FedSMOO-D	84.55	80.82	53.92	46.48	38.71	32.45
FedLESAM-D	84.27	80.08	53.27	46.42	27.36	25.32
FedGloSS-D†	82.58	79.23	50.92	47.36	31.72	28.04
Our FedFFT-D	87.19	83.05	54.46	50.90	40.85	34.46

ing low-frequency perturbations is particularly beneficial when client data distributions are highly skewed. Moreover, on Tiny-ImageNet, FedFFT-D clearly surpasses all baselines, highlighting its robustness and scalability to larger and more challenging tasks. Furthermore, the convergence curves in Figure 6 of Appendix D.1 show that FedFFT not only achieves higher final accuracy but also converges significantly faster. This suggests that by harmonizing client updates at a spectral level, FedFFT facilitates a more direct and stable optimization path toward a high-quality global minimum, reducing wasted communication rounds spent reconciling conflicting updates. Additionally, we have plotted the loss landscapes of different algorithms, which reveal that our method yields a flatter loss landscape. For details, please refer to the Appendix D.2.

Generalization Across Diverse Model Architectures. To verify that the efficacy of FedFFT is not confined to a specific model class, we evaluate its performance across a diverse range of architectures, from lightweight ResNet-18, ResNet-20 to deeper DenseNet-121 and Vision Transformers (ViT). We perform on CIFAR-10 and CIFAR-100 with a moderately heterogeneous setting  $\alpha=0.6$ . As shown in Tab 2, FedFFT usually outperforms baselines across different architectures. Critically,

Table 2: Test accuracy (%) across different backbones on CIFAR-10 (C10) and CIFAR-100 (C100) with Dirichlet ( $\alpha = 0.6$ ). All methods use FedDyn as the base algorithm.

Data	Method	ResNet18	ResNet20	DenseNet121	ViT
	FedSAM	82.29	88.82	89.47	49.04
C10	FedSMOO	<u>84.55</u>	<u>89.86</u>	88.72	50.23
CIU	FedGloSS	82.58	84.17	86.84	50.10
	FedFFT	87.19	90.56	90.57	53.31
	FedSAM	53.70	58.92	64.19	28.01
C100	FedSMOO	53.92	58.17	63.74	29.48
C100	FedGloSS	50.92	46.68	57.14	27.54
	FedFFT	54.46	61.60	61.66	30.36

the performance gains are substantial even on powerful models like ViT. This suggests that the problem FedFFT addresses—the inconsistency in the low-frequency spectrum of client perturbations—is a fundamental artifact of the federated optimization process itself, independent of a model's representation capacity. Simply using a larger model does not automatically resolve the geometric misalignment between clients. Our spectral filtering acts as a complementary and orthogonal improvement, harmonizing the local updates to allow these powerful architectures to converge more effectively. These results therefore underscore the broad applicability and scalability of FedFFT, establishing it as a model-agnostic enhancement for sharpness-aware federated learning.

#### 5.3 FURTHER ANALYSIS

**Ablation on Filtering Strategy.** To provide direct empirical validation for our central hypothesis—that inter-client perturbation disagreement is concentrated in the low-frequency spectrum—we compare our proposed low-frequency filtering against two alternative strategies: high-frequency fil-

tering and random filtering, where we adopt the same filtering ratio r=0.01. As shown in Table 3, removing high-frequency or random components yields negligible gains over the FedSAM baseline. In contrast, filtering the low-frequency components, as FedFFT does, consistently provides substantial accuracy improvements, particularly under the high-heterogeneity  $(\alpha=0.1)$  setting. This result demonstrates that the low-frequency spectrum is indeed the primary source

Table 3: Results of different filtering methods on CIFAR-10 using ResNet-18 backbone.

Filtering Strategy	ring Strategy Accuracy (	
	$\alpha = 0.6$	$\alpha = 0.1$
None (FedSAM)	81.91	74.92
High-frequency	81.66	74.81
Random	81.97	75.09
Low-frequency (ours)	83.02	77.53

of client-specific, discordant information. By selectively removing it, FedFFT effectively isolates and preserves the task-relevant, high-frequency signals, leading to more consistent client updates and a superior global model. Furthermore, we have also explored the impact of different filtering strategies when applied to various backbones. For further details, please refer to the Appendix E.1.

Impact of Filtering Ratio. We explore the impact of varying the filtering ratio r from 0.1% to 8% to highlight the flexibility and adaptability of hyperparameter tuning in our proposed FedFFT. As shown in Fig. 2, we report the performance of different r values in FedFFT, where we view FedSAM as baseline and use FedAvg, SCAFFOLD and FedDyn as the base algorithms, respectively. We can find that even with a small r such as 0.1%, ours can consistently outperform its baseline. It proves that removing the discordant low-frequency components is beneficial for sharpness-aware federated optimization. Besides, high filtering ratios (>7%) lead to gradual performance degradation, approaching or slightly falling below the baseline, which may remove useful information for optimization. Overall, these re-

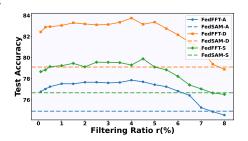


Figure 2: Test accuracy with different filtering ratios on CIFAR-10 ( $\alpha=0.1$ ). "-A", "-S" and "-D" mean using FedAvg, SCAFFOLD and FedDyn as the base algorithms.

sults confirm that our FedFFT can stably improve accuracy across all models when maintaining a reasonable filtering ratio. Based on our experiments, we recommend a safe range for r between 0.5% and 4%.

Robustness to FL Settings. We further assess the robustness of FedFFT across three critical federated learning hyperparameters: client activation rate, number of local epochs, and total number of clients. As summarized in Fig. 3, FedFFT consistently and significantly outperforms FedSAM across all tested configurations. Regarding client participation in Fig. 3 (a), FedFFT maintains a stable performance advantage even with a low activation rate of 5%, a challenging scenario that often exacerbates client drift. Similarly, when varying the number of local epochs in Fig. 3 (b), FedFFT demonstrates larger gains with fewer epochs, suggesting faster convergence, while still maintaining a clear edge with more local training. Finally, Fig. 3 (c) shows that our method's superiority holds as the number of clients scales, confirming its applicability to large-scale federated networks. These results collectively demonstrate that the benefits of spectral filtering are not confined to a specific setting but are robust to the practical constraints of real-world FL systems.

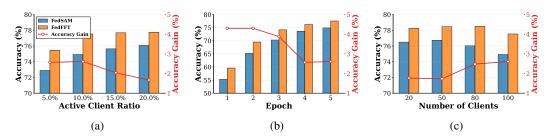


Figure 3: Comprehensive performance comparison between FedFFT and FedSAM across diverse experimental configurations on CIFAR-10 (Non-IID  $\alpha=0.1$ ) using ResNet-18 backbone.

Visualizing the Effect of Spectral Filtering. To qualitatively understand how FedFFT achieves greater consistency, we visualize the distributions of client perturbations, model features, and model

parameters. The visualizations in Fig. 4 reveal a clear causal chain. (1) More Cohesive Perturbations (Fig. 4a): The process begins with the perturbations themselves. The original SAM perturbations from different clients are widely scattered in the PCA space. After applying FedFFT's low-frequency filter, these perturbations become significantly more compact, confirming that our method successfully reduces inter-client discrepancy at its source. (2) Aligned Feature Representations (Fig. 4b): This improved consistency in perturbations directly translates to more aligned model behavior. We observe that the average features extracted by the FedFFT-trained model are much more tightly clustered for each class compared to the scattered features from the FedSAM model. (3) Consolidated Client Models (Fig. 4c): Ultimately, this leads to better convergence of the models themselves. The parameters of client models trained with FedFFT exhibit a much smaller variance and are clustered closer to a central point, indicating that spectral filtering effectively mitigates client drift and guides all clients toward a more unified and robust global solution.

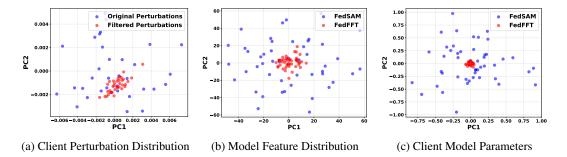


Figure 4: Effect of low-frequency perturbation filtering on ResNet-18/CIFAR-10 ( $\alpha=0.1$ ). The comparison between FedSAM (blue) and FedFFT (red) shows: (a) increased compactness of client perturbations (conv1.weight), leading to (b) more consistent model features (layer4.1.gn2) and (c) more aligned client model parameters (conv1.weight).

Communication cost. Communication cost is a critical bottleneck in federated learning (FL), making its optimization an important challenge. In this study, we take FedAvg as the baseline, and define B as the total number of bits exchanged by FedAvg during T training rounds. For each method, we measure its communication cost in terms of (i) the number of rounds required to reach FedAvg's performance, and (ii) the total number of bits exchanged in these rounds. As shown in Table 4, our method significantly reduces both the number of communication rounds and the total transmitted bits compared with state-of-the-art baselines, while achieving comparable performance to FedAvg.

Table 4: Communication cost comparison on different datasets with ResNet-18. *Note:* Each cell reports **Rounds / Relative Communication Cost (B)**.

Method	CIFAR-10		CIFAR-100		Tiny-ImageNet	
111041104	$\alpha = 0.6$	$\alpha = 0.1$	$\alpha = 0.6$	$\alpha = 0.1$	$\alpha = 0.6$	$\alpha = 0.1$
FedAvg	800 / 1.00	800 / 1.00	800 / 1.00	800 / 1.00	300 / 1.00	300 / 1.00
FedSAM	465 / 0.58	718 / 0.90	483 / 0.60	491 / 0.61	158 / 0.53	250 / 0.83
FedSMOO	205 / 0.51	312 / 0.78	201 / 0.50	225 / 0.56	143 / 0.95	204 / 1.36
FedGLOSS	386 / 0.48	487 / 0.61	261 / 0.33	285 / 0.36	206 / 0.69	264 / 0.88
FedFFT-D (Ours)	190 / 0.24	302 / 0.38	158 / 0.20	211 / 0.26	131 / 0.44	179 / 0.60

#### 6 CONCLUSION

In this work, we introduce a novel frequency-domain perspective to address divergent perturbations in sharpness-aware federated learning. We identify that inter-client disagreements are predominantly a low-frequency phenomenon and accordingly propose FedFFT, a lightweight filtering method to suppress these discordant components. Extensive experiments validate that FedFFT consistently outperforms state-of-the-art methods, particularly in highly non-IID settings, by converging to visibly flatter and wider global minima. This result not only explains the superior generalization and communication efficiency of our method but also establishes spectral analysis as a powerful new tool for designing robust federated optimization algorithms.

# 7 ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics. Our research does not involve human subjects, personally identifiable information, or sensitive data. All datasets used in this work are publicly available and widely used in the research community. We have carefully considered potential risks of misuse, fairness, and bias, and we provide detailed analysis in the experiments to ensure that our methods do not amplify harmful stereotypes or unfair treatment. The results and methodologies are intended solely for academic research and are not designed for deployment in safety-critical or harmful applications.

# 8 REPRODUCIBILITY STATEMENT

We have taken multiple steps to ensure the reproducibility of our results. All details regarding datasets, preprocessing, model architectures, training procedures, and hyperparameters are described in the Section 5.1 and Appendix B. Complete proofs of theoretical claims are provided in the supplementary materials. To further support reproducibility, we will release the source code and instructions for reproducing all experiments in the supplementary materials.

#### 9 LLM USAGE STATEMENT

Large language models (LLMs) were used solely as an assistive tool for language polishing, grammar correction, and improving readability. No part of the research ideation, methodology design, experimental implementation, or result analysis was conducted by LLMs.

#### REFERENCES

- Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=B7v4QMR6Z9w.
- K. Ahn, A. Jadbabaie, and S. Sra. How to escape sharp minima with random perturbations. In *In Forty-first International Conference on Machine Learning.*, 2024.
- Rodolfo Stoffel Antunes, Cristiano André da Costa, Arne Küderle, Imrana Abdullahi Yari, and Björn Eskofier. Federated learning for healthcare: Systematic review and architecture proposal. *ACM Trans. Intell. Syst. Technol.*, 13(4), May 2022. ISSN 2157-6904. doi: 10.1145/3501813. URL https://doi.org/10.1145/3501813.
- Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. 2011. doi: 10.1561/2200000016.
- Debora Caldarola, Barbara Caputo, and Marco Ciccone. Improving generalization in federated learning by seeking flat minima. In *European Conference on Computer Vision*, pp. 654–672. Springer, 2022.
- Debora Caldarola, Pietro Cagnasso, Barbara Caputo, and Marco Ciccone. Beyond local sharpness: Communication-efficient global sharpness-aware minimization for federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.
- Junhong Chen, Hong Li, and C.L. Philip Chen. Boosting sharpness-aware training with dynamic neighborhood. *Pattern Recogn.*, 153(C), September 2024. ISSN 0031-3203. doi: 10.1016/j.patcog.2024.110496. URL https://doi.org/10.1016/j.patcog.2024.110496.
- Stanford University CS231N. Tiny imagenet visual recognition challenge. http://cs231n.stanford.edu/, 2015.
- Rong Dai, Xun Yang, Yan Sun, Li Shen, Xinmei Tian, Meng Wang, and Yongdong Zhang. Fedgamma: Federated learning with global sharpness-aware minimization. *IEEE Transactions on Neural Networks and Learning Systems*, 35(12):17479–17492, 2024. doi: 10.1109/TNNLS. 2023.3304453.

- Ziqing Fan, Yanfeng Wang, Jiangchao Yao, Lingjuan Lyu, Ya Zhang, and Qi Tian. Fedskip: Combatting statistical heterogeneity with federated skip aggregation. In 2022 IEEE International Conference on Data Mining (ICDM), pp. 131–140, 2022. doi: 10.1109/ICDM54844.2022.00023.
  - 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 *International Conference on Machine Learning*, 2024.
  - Lidia Fantauzzo, Eros Fanì, Debora Caldarola, Antonio Tavera, Fabio Cermelli, Marco Ciccone, and Barbara Caputo. Feddrive: Generalizing federated learning to semantic segmentation in autonomous driving. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 11504–11511, 2022. doi: 10.1109/IROS47612.2022.9981098.
  - Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-Aware Minimization for Efficiently Improving Generalization. art. arXiv:2010.01412, 2020.
  - Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.
  - S. Hochreiter and J. Schmidhuber. Simplifying neural nets by discovering flat minima. In *Advances in neural information processing systems*, 1994.
  - Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J. Reddi, Sebastian U. Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning, 2021. URL https://arxiv.org/abs/1910.06378.
  - Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017.
  - Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.
  - Jungmin Kwon, Jeongseop Kim, Hyunseo Park, and In Kwon Choi. Asam: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5905–5914. PMLR, 18–24 Jul 2021.
  - 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 (CVPR)*, 2024.
  - Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10708–10717, 2021. doi: 10.1109/CVPR46437.2021.01057.
  - Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020a. doi: 10.1109/MSP.2020.2975749.
  - Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks, 2020b. URL https://arxiv.org/abs/1812.06127.
- Zexi Li, Tao Lin, Xinyi Shang, and Chao Wu. Revisiting weighted aggregation in federated learning
   with neural networks. In *Proceedings of the 40th International Conference on Machine Learning*,
   ICML'23. JMLR.org, 2023.
  - Y. Liu, S. Mai, M. Cheng, X. Chen, C.-J. Hsieh, and Y. You. Random sharpness-aware minimization. In *Advances in Neural Information Processing Systems*, 2022.

- Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.
  - B. Neyshabur, S. Bhojanapalli, D. McAllester, and Srebro. Exploring generalization in deep learning. In *Advances in neural information processing systems*, 2017.
  - Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019. URL https://arxiv.org/abs/1912.01703.
  - Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization, 2022. URL https://arxiv.org/abs/2206.02618.
  - Zhe Qu, Xingyu Li, Xiao Han, Rui Duan, Chengchao Shen, and Lixing Chen. How to prevent the poor performance clients for personalized federated learning? In 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12167–12176, 2023. doi: 10.1109/CVPR52729.2023.01171.
  - Ashish Rauniyar, Desta Haileselassie Hagos, Debesh Jha, Jan Erik Håkegård, Ulas Bagci, Danda B. Rawat, and Vladimir Vlassov. Federated learning for medical applications: A taxonomy, current trends, challenges, and future research directions. *IEEE Internet of Things Journal*, 11(5):7374–7398, 2024. doi: 10.1109/JIOT.2023.3329061.
  - Changlong Shi, He Zhao, Bingjie Zhang, Mingyuan Zhou, Dandan Guo, and Yi Chang. Fedawa: Adaptive optimization of aggregation weights in federated learning using client vectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
  - 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 *International Conference on Machine Learning*, pp. 32991–33013. PMLR, 2023a.
  - Yan Sun, Li Shen, Tiansheng Huang, Liang Ding, and Dacheng Tao. Fedspeed: Larger local interval, less communication round, and higher generalization accuracy, 2023b. URL https://arxiv.org/abs/2302.10429.
  - Blake Woodworth, Kumar Kshitij Patel, Sebastian U. Stich, Zhen Dai, Brian Bullins, H. Brendan McMahan, Ohad Shamir, and Nathan Srebro. Is local sgd better than minibatch sgd? In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.
  - Xinda Xing, Qiugang Zhan, Xiurui Xie, Yuning Yang, Qiang Wang, and Guisong Liu. Flexible sharpness-aware personalized federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 21707–21715, Philadelphia, Pennsylvania, USA, 2025.
  - Jing Xu, Sen Wang, Liwei Wang, and Andrew Chi-Chih Yao. Fedem: Federated learning with client-level momentum. *ArXiv*, abs/2106.10874, 2021. URL https://api.semanticscholar.org/CorpusID:235490679.
  - Rui Ye, Mingkai Xu, Jianyu Wang, Chenxin Xu, Siheng Chen, and Yanfeng Wang. Feddisco: federated learning with discrepancy-aware collaboration. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.
- Ruipeng Zhang, Ziqing Fan, Jiangchao Yao, Ya Zhang, and Yanfeng Wang. Domain-inspired sharpness-aware minimization under domain shifts. In *International Conference on Learning Representations*, 2024.
- Juntang Zhuang, Boqing Gong, Liangzhe Yuan, Yin Cui, Hartwig Adam, Nicha Dvornek, Sekhar Tatikonda, James Duncan, and Ting Liu. Surrogate gap minimization improves sharpness-aware training. In *International Conference on Learning Representations*, 2022.

# ALGORITHMS

648

649

650 651

652

653

654 655

656

657

658

659

660

661 662

663

665

666

667

668

669

670

671

672 673

674

675 676 677

678 679

680

681

682

684

685

686

687

688

689

690

691

692 693

696

697

699

701

A.1 FEDFFT-D ALGORITHM.

#### **Algorithm 2** The FedFFT-D Algorithm.

**Require:** Communication rounds T, local epochs E, perturbation radius  $\rho$ , local learning rate  $\eta$ , frequency truncation ratio r, penalty parameter  $\beta$ , local multipliers  $\{\lambda_k\}_{k=1}^K$ , global multiplier

**Ensure:** Global model  $w_q^T$ 

Initialize global model  $w_a^0$ , local multipliers  $\forall k, \lambda_k = 0$ , and global multiplier  $\lambda = 0$ .

- 2: **for** t = 0 to T 1 **do** 
  - Randomly select active client set  $S_t$ .
- for all clients  $k \in S_t$  in parallel do 4:
- $w^{k,t,0} \leftarrow w_q^t$

> Server sends global model to client 

 $\begin{aligned} & \overset{w}{\text{for }} e = 0 \text{ to } E - 1 \text{ do} \\ & \delta^{k,t,e} \leftarrow \rho \cdot \frac{\nabla \mathcal{L}_k(w^{k,t,e})}{\|\nabla \mathcal{L}_k(w^{k,t,e})\|_2} \end{aligned}$ 

▶ Apply filtering process

$$\begin{split} \tilde{\delta}_l^{k,t,e} &= \operatorname{Filter}(\tilde{\delta}_l^{k,t,e},r), \ \forall l \in [1,L] \\ w^{k,t,e+1} &\leftarrow w^{k,t,e} - \eta_l \Big( \nabla \mathcal{L}_k(w^{k,t,e} + \tilde{\delta}^{k,t,e}) + \lambda_k + \frac{1}{\beta}(w^{k,t,e} - w_g^t) \Big) \end{split}$$

Dyn

update step

- 10: end for
- Send local model  $w^{k,t,E}$  to the server.
- Update local multiplier:  $\lambda_k \leftarrow \lambda_k \frac{1}{\beta}(w^{k,t,E} w_a^t)$ . 12:

- Aggregate models:  $w_g^{t+1} \leftarrow w_g^t \eta_g \sum_{k \in S_t} (w_g^t w^{k,t,E})$ .  $ightharpoonup \text{Aggregate models on server } \text{Update global multiplier: } \lambda \leftarrow \lambda \frac{1}{\beta |S_t|} \sum_{k \in S_t} (w^{k,t,E} w_g^t).$ 14:
- 16: **end for** return  $w_a^T$

#### A.2 FEDFFT-S ALGORITHM.

#### **Algorithm 3** The FedFFT-S Algorithm.

**Require:** Communication rounds T, local epochs E, perturbation radius  $\rho$ , local learning rate  $\eta$ , frequency truncation ratio r, penalty parameter  $\beta$ , local control variates  $\{C_k\}_{k=1}^K$ , global control

**Ensure:** Global model  $w_q^T$ 

Initialize global model  $w_q^0$ , local control variates  $\forall k, C_k = 0$ , and global control variate C = 0.

- 2: **for** t = 0 to T 1 **do** 
  - Randomly select active client set  $S_t$ .
- for all clients  $k \in S_t$  in parallel do 4:

6:

 $\begin{aligned} w^{k,t,0} &\leftarrow w_g^t \\ \textbf{for } e &= 0 \text{ to } E - 1 \textbf{ do} \\ \delta^{k,t,e} &\leftarrow \rho \cdot \frac{\nabla \mathcal{L}_k(w^{k,t,e})}{\|\nabla \mathcal{L}_k(w^{k,t,e})\|_2} \end{aligned}$ 

▷ Server sends global model to client

$$\begin{split} &\tilde{\delta}_{l}^{k,t,e} = \text{Filter}(\delta_{l}^{k,t,e},r), \ \forall l \in [1,L] \\ &w^{k,t,e+1} \leftarrow w^{k,t,e} - \eta_{l} \Big( \nabla \mathcal{L}_{k}(w^{k,t,e} + \tilde{\delta}^{k,t,e}) + C_{k} - C \Big) \quad \triangleright \text{Scaffold update step} \end{split}$$
8:

10:  $C_k \leftarrow C_k - C + \frac{1}{\eta E} (w_g^t - w^{k,t,E})$ 

Send local model  $w^{k,t,E}$  and  $C_k$  to the server. 12:

- Aggregate models:  $w_g^{t+1} \leftarrow w_g^t \eta_g \sum_{k \in S_t} (w_g^t w^{k,t,E})$ .  $\triangleright$  Aggregate models on server 14: Update global control variate:  $C \leftarrow C + \frac{1}{\kappa}C_i$
- 16: **end for** return  $w_a^T$

# B SAM IN FL

Research Work	Base Algorithm	Minimizing Target	Perturbation
FedSAM (ECCV22, ICML22)	FedAvg	Local Sharpness	$\rho \cdot \frac{\nabla \mathcal{L}_k(w^{k,t,e})}{\ \nabla \mathcal{L}_k(w^{k,t,e})\ }$
MoFedSAM (ICML22)	FedAvg with Momentum	Local Sharpness	$\rho \cdot \frac{\nabla \mathcal{L}_k(w^k, t, e)}{\ \nabla \mathcal{L}_k(w^k, t, e)\ }$
FedGAMMA (TNNLS23)	Scaffold	Local Sharpness	$\rho \cdot \frac{\nabla \mathcal{L}_{k}(w^{k,t,e})}{\ \nabla \mathcal{L}_{k}(w^{k,t,e})\ }$
FedSMOO (ICML23)	FedDyn	Local Sharpness with Correction	$\rho \cdot \frac{\nabla \mathcal{L}_k(w^{k,t,e}) - \mu_i - s}{\ \nabla \mathcal{L}_k(w^{k,t,e}) - \mu_i - s\ }$
FedLESAM (ICML24)	FedAvg, Scaffold, FedDyn	Global Sharpness	$\rho \cdot \frac{\frac{w_{old}^k - w_g^t}{\ w_{old}^k - w_g^t\ }}{\ \omega_{old}^k - w_g^t\ }$ $\rho \cdot \frac{\Delta_w^{t-1}}{\ \Delta_w^{t-1}\ }$
FEDGLOSS (CVPR25)	FedDyn-like	Global Sharpness via Pseudo-gradient	$ ho \cdot rac{\Delta_w^{\overline{t}-1}}{\ \Delta_w^{\overline{t}-1}\ }$
FedFFT (ours)	FedAvg, Scaffold, FedDyn	Local Sharpness with Filtering	$\rho \cdot Filter(\frac{\nabla \mathcal{L}_k(w^{k,t,e})}{\ \nabla \mathcal{L}_k(w^{k,t,e})\ })$

Table 5: Summary of federated SAM-based algorithms for solving data heterogeneity.

# C IMPLEMENTATION OF THE EXPERIMENTS

# C.1 HYPERPARAMETERS

For experiments on CIFAR-10 and CIFAR-100, we adopt the training configurations consistent with FedSMOO (Sun et al., 2023a) and FedLESAM (Fan et al., 2024) for fair comparison. The backbone network is ResNet-18 equipped with Group Normalization and optimized using SGD. The total number of communication rounds is set to 800 for CIFAR-10 and CIFAR-100, and 300 for Tiny ImageNet. The initial local learning rate is  $\eta=0.1$ . Unless otherwise specified, the learning rate decays exponentially by a factor of  $0.998\times$  per round; however, FedDyn, FedSMOO, FedLESAM-D and FedFFT-D use a slower decay rate of  $0.9995\times$  for the proxy term. For CIFAR-10, we use a batch size of 50 and set the number of local epochs to 5. For CIFAR-100, the batch size is 20 with 2 local epochs. For Tiny ImageNet, we follow the same configuration as that of CIFAR-10.

#### C.2 Models

In our experiments, we adopt different backbone architectures for evaluation. For the experiments reported in Table 1, we use the standard ResNet18 model from the torchvision library, where all Batch Normalization (BN) layers are replaced by Group Normalization (GN) layers to improve training stability in federated settings.

For the experiments in Table 2, we further evaluate three representative architectures: (i) ResNet20, implemented following the CIFAR variant with GN layers instead of BN; (ii) DenseNet121, where we use DenseNet\_fedlaw (Li et al., 2023) implementation with GN applied after the final dense block; and (iii) a Vision Transformer, specifically the vit\_tiny\_patch16\_224 model from the timm library. These choices allow us to validate the generality of our approach across both convolutional and transformer-based models.

#### C.3 DATASETS

CIFAR-10 and CIFAR-100 are widely used benchmark datasets in computer vision and federated learning research. Both consist of small-scale natural images of size 32×32 with three color channels. CIFAR-10 contains 10 object categories while CIFAR-100 extends this to 100 finer-grained classes, making it a more challenging variant. Despite their limited resolution, these datasets remain popular due to their balanced composition and ease of use in distributed training scenarios.

To further evaluate scalability, Tiny ImageNet is employed, which provides 200 categories of images with higher resolution (64×64). Compared with CIFAR datasets, Tiny ImageNet introduces more diverse classes and larger input dimensions, enabling more comprehensive testing of algorithms under settings with higher model capacity and increased class heterogeneity. Such datasets are especially valuable in federated learning studies, where both efficiency and robustness to distributional challenges are critical.

Table 6: Dataset introductions.

Dataset	Training Data	Test Data	Class Size / Image
CIFAR-10	50000	10000	10 / 3×32×32
CIFAR-100	50000	10000	100 / 3×32×32
Tiny ImageNet	100000	10000	200 / 3×64×64

#### C.4 CLIENT DATA DISTRIBUTION VISUALIZATIONS

In this appendix, we present the client data distributions under different Dirichlet parameters  $\alpha$  for CIFAR-10, CIFAR-100, and Tiny ImageNet. Each heatmap shows the number of samples per class for each client.

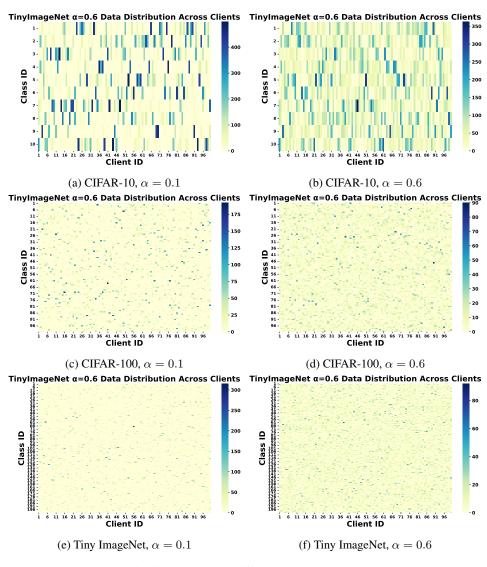


Figure 5: Client data distributions across different datasets and Dirichlet parameters  $\alpha$ .

# D VISUALIZATION

#### D.1 LEARNING CURVE

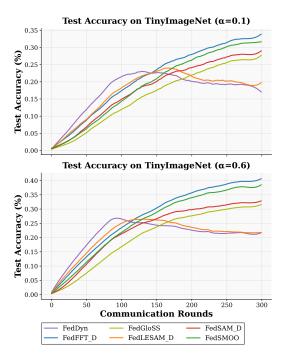


Figure 6: Test Accuracy on TinyImageNet ( $\alpha = 0.1$  and  $\alpha = 0.6$ ).

# D.2 3D LANDSCAPE VISUALIZATION

Visualization of the Global Loss Landscape. To visualize the 3D loss landscape, we perturb the parameters of the best-performing checkpoint along its top-two Hessian eigen-directions—computed via power iteration on 500 CIFAR-10 test samples—and plot the corresponding loss values as a smooth surface. As shown in Figure 7, Compared to FedSAM, FedSMOO, and FedGloSS, the loss surface corresponding to the FedFFT\_D solution is visibly flatter and wider. This provides a clear geometric explanation for our superior generalization performance. By filtering out discordant, client-specific sharpness directions, our method successfully guides the global model to converge not just to a point of low loss, but to a broad, flat minimum that is inherently more robust to the data distribution shifts present across clients. The 2D landscape can be found in the D.3.

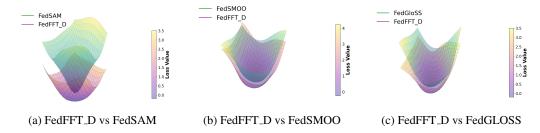
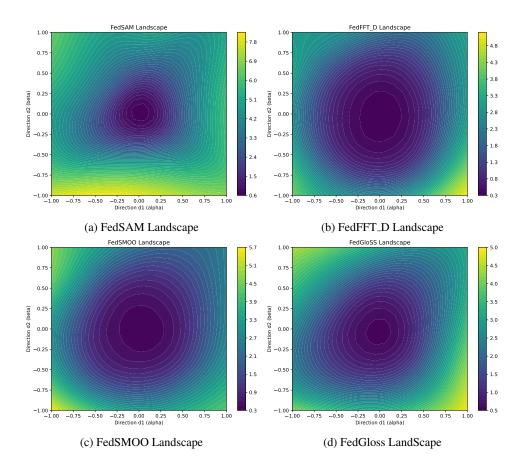


Figure 7: Comparison of 3D loss landscapes on CIFAR-10/ResNet-18 ( $\alpha=0.1$ ). Visualization of loss landscapes for different federated learning algorithms: (a) FedFFT\_D compared with FedSAM, (b) FedFFT\_D compared with FedSMOO, (c) FedFFT\_D compared with FedGLOSS.

#### D.3 2D LANDSCAPE VISUALIZATION



#### E OTHERS

#### E.1 DIFFFERENT FILTERING METHODS ON RESNET20.

We further investigate the effect of different spectral filtering strategies on CIFAR-10 using the ResNet-20 backbone, as summarized in Table 7. For a fair comparison, all methods adopt the same filtering ratio of 0.01, i.e., the lowest 1% or highest 1% frequency components are removed, or 1% of frequencies are randomly removed. All methods are implemented based on the FedDyn framework.

Table 7: Performance Comparison of Different Filtering Methods on CIFAR-10 using ResNet-20 backbone.

Filtering Approach	Accuracy (%)		
Thering Approach	$\alpha = 0.6$	$\alpha = 0.1$	
FedDyn + SAM	88.82	77.10	
High-frequency Filtering	89.01	77.20	
Random Filtering	88.48	76.38	
Low-frequency Filtering (FedFFT-D)	91.23	81.24	

#### E.2 Combine with other methods.

To further validate the universality of our method, we selected to combine it with approaches that optimize weights. (Shi et al., 2025) We implemented the SAM optimizer on all client sides and used the FedAvg framework for verification on the ResNet-20 model. As shown in 8.

Table 8: Test accuracy (%) on CIFAR-10 and CIFAR-100 with Dirichlet distribution parameter  $\alpha=0.1$ . The federated learning setup is fixed as **400 rounds**, **100 clients**, active ratio = **0.1**, and SAM is used as the client optimizer. We compare FedAWA and FedAWA+FedFFT.

Method	CIFAR-10	CIFAR-100
FedAWA	47.20	33.40
FedAWA+FFT	50.16	34.04