Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.085467

ARTICLE

SAM-ADPFL: A Geometry-Aware Adaptive Framework for Privacy-Preserving Federated Learning Systems

Fangfang Shan*, Yuhang Liu*, Lulu Fan, Zhuo Chen, Yifan Mao and Peixue Wang

College of Computer, Zhongyuan University of Technology, Zhengzhou, China

*Corresponding Authors: Fangfang Shan. Email: 6129@zut.edu.cn; Yuhang Liu. Email: 2024007015@zut.edu.cn

Received: 11 May 2026; Accepted: 08 July 2026; Published: 15 September 2026

ABSTRACT: The engineering of Federated Learning (FL) systems faces significant challenges in balancing two critical non-functional requirements: ensuring robust system utility and maintaining high privacy protection standards under non-independent and identically distributed (Non-IID) data environments. Existing software architectures often struggle to achieve an optimal trade-off between these competing demands. This paper proposes SAM-ADPFL, a novel architectural framework designed to improve the engineering and management of privacy-preserving distributed machine learning systems. First, we design a geometry-aware adaptive aggregation component that dynamically reallocates aggregation weights based on local landscape properties, guiding the global model to effectively suppress model drift. Second, we propose a sharpness-guided dynamic differential privacy mechanism to handle security requirements, adaptively managing the privacy budget through refined gradient clipping and noise injection. Through empirical studies on standard benchmark datasets, we demonstrate that SAM-ADPFL exhibits superior robustness in extreme heterogeneous scenarios. By achieving a superior trade-off between system utility and privacy protection, this research contributes reusable architectural patterns and engineering practices for the design, optimization, and evaluation of privacy-preserving distributed software systems.

KEYWORDS: Federated learning; distributed software systems; loss landscape geometry; privacy-preserving engineering; adaptive aggregation

1 Introduction

Federated Learning (FL) has emerged as a critical architectural pattern for distributed software systems, enabling collaborative model training across decentralized clients while keeping data localized. By exchanging only model parameters or gradients without sharing raw data, FL effectively addresses the "data silos" problem [1,2] and provides a privacy-preserving engineering solution for distributed machine learning applications. However, in actual large-scale deployments, managing FL systems presents two fundamental software engineering challenges that urgently need to be addressed: maintaining high system utility (reliability) under statistical data heterogeneity, and satisfying strict non-functional requirements for privacy and security [3].

First, since the local data distributions of clients often exhibit a high degree of heterogeneity, this distribution shift leads to inconsistencies between local optimization objectives and the global objective. This triggers severe "model drift", causing the global model to converge slowly or fail to converge, thereby significantly compromising generalization performance. Second, although FL avoids the direct transmission of raw data, attackers can still analyze uploaded gradients or model parameters to reconstruct raw training

Copyright © 2026 The Authors. Published by Tech Science Press. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

data via gradient inversion attacks, or infer sample presence via membership inference attacks [4–6]. To resist such attacks, multiple studies [6–8] point out that injecting appropriate differential privacy (DP) noise into gradients before iterative updates can significantly reduce the risk of training data leakage. Consequently, gradient-level noise injection is regarded as a preferred scheme for privacy protection. However, injecting noise inevitably destroys the precision of gradients, resulting in degraded model utility.

Addressing the model drift problem caused by Non-IID data, algorithms such as FedProx [9] and SCAFFOLD [10] attempt to correct local updates through local regularization terms or control variates. To improve theoretical convergence performance, some studies [11] have introduced server-side extrapolation, while FedDA [12] proposed resource-adaptive split models and aggregation strategies that align parameter feature spaces and output spaces. Additionally, FedNova [13] eliminates the impact of heterogeneous update steps through normalized aggregation weights, and MOON [14] improves local training objectives by introducing model contrastive loss. Although these methods mitigate the negative impact of Non-IID to a certain extent, their effectiveness remains limited in extreme heterogeneous scenarios. Progress was made in [15] by learning aggregation weights on the server side, which significantly enhanced model utility; however, its heavy reliance on the assumption of proxy data raises risks regarding privacy leakage and distribution shifts.

In terms of privacy protection, due to the severe impact of injecting fixed noise into model parameters or gradients on model utility in DP-FedAvg [16,17], recent research has increasingly explored adaptive differential privacy. Adaptive differential privacy balances privacy protection and model performance by dynamically adjusting noise magnitude. Some works [18–22] reduce noise interference via adaptive strategies by calculating the importance of local model parameters or injecting noise adaptively based on data attributes, thereby safeguarding model performance under privacy security conditions. However, these methods ignore the intrinsic differences in client data. Recent works have attempted to introduce Sharpness-Aware Minimization (SAM) [23–25] to enhance model robustness against noise [26], they have failed to establish a connection between sharpness and privacy budget allocation, and thus have not broken through the limitations of static noise.

Facing the aforementioned limitations, this paper proposes a new perspective by deeply analyzing the geometric characteristics of the loss landscape in Sharpness-Aware Minimization: the "sharpness" of the loss landscape can serve as a bridge connecting "model generalization" and "data privacy".

Our core insight for this system design is based on the following two theoretical findings:

- **Sharpness is negatively correlated with generalization:** Models that converge to flat minima possess wide and gentle loss function landscapes and are insensitive to minute perturbations in parameters.
- **Sharpness is positively correlated with sensitivity:** Higher sharpness (steeper landscape) implies that variations in a single data point will cause drastic changes in gradients, indicating higher data sensitivity.

Based on this, our contributions are as follows:

(1) We design a collaborative optimization framework, SAM-ADPFL, based on the dual efficacy of the Sharpness-Aware Minimization algorithm.
(2) We propose a novel sharpness-aware adaptive aggregation framework. It dynamically reallocates weights based on local loss landscape geometry, assigning higher weights to flat regions and lower weights to sharp ones. Combined with global weight shrinkage, it guides the global model to flat minima, overcoming data heterogeneity and improving accuracy and robustness.
(3) We propose, for the first time, a method utilizing sharpness information to dynamically quantify the privacy sensitivity of each client, thereby achieving adaptive noise allocation and effectively balancing privacy protection with model utility.

(4) Experimental results demonstrate that our method exhibits superior robustness in non-independent and identically distributed (Non-IID) environments. Furthermore, it significantly outperforms existing differential privacy federated learning schemes in balancing model utility and privacy protection.

To intuitively demonstrate the validity of the core insights mentioned above and to provide empirical support for the methodology proposed in this paper, we conducted two sets of exploratory experiments.

(1) Visualization of Loss Landscapes: We compared the global loss landscapes after convergence of DP-FedAvg and our proposed method, SAM-ADPFL, using the CIFAR-10 dataset and a VGG-like convolutional network adapted for CIFAR-10 [27].

As shown in Fig. 1, the loss landscape of DP-FedAvg exhibits high curvature, where minute shifts in model parameters cause a sharp increase in loss function values. In contrast, the loss surface of our proposed method is relatively flat with smaller curvature. Even if parameters fluctuate within a certain range, the loss function value remains at a lower level.

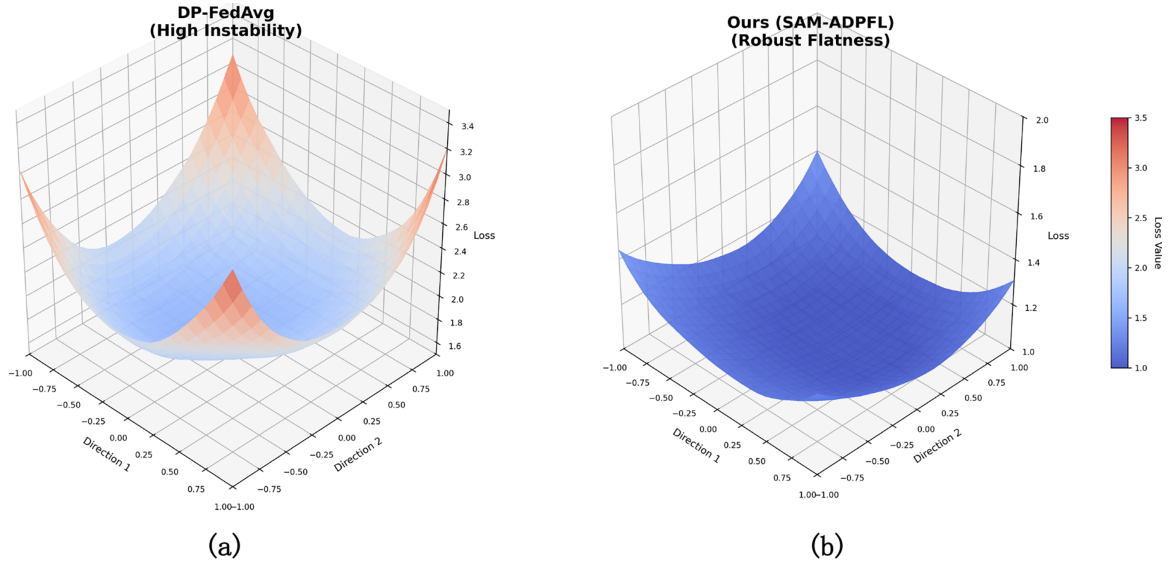

Figure 1: Visualization of loss landscapes. (**a**) DP-FedAvg exhibits a sharp, high-instability loss surface; (**b**) Our proposed SAM-ADPFL yields a robust, flat loss landscape.

(2) As shown in Fig. 2, the strong linear correlation between exact sharpness and local gradient norm (Spearman ρ_s: MNIST 0.97, FMNIST 0.90, CIFAR-10 0.72) validates our first-order Taylor approximation ($S_k \approx \rho \|\nabla F_k\|_2$). Since DP L_2-sensitivity is bounded by gradient norms, sharpness serves as an empirically supported and theoretically motivated proxy, rather than a fully rigorous equivalent.

The remainder of this paper is organized as follows: Section 2 presents related work. Section 3 introduces preliminaries and theoretical foundations. In Section 4, we detail our method. Section 5 provides experimental results and relevant analysis. Section 6 concludes the paper.

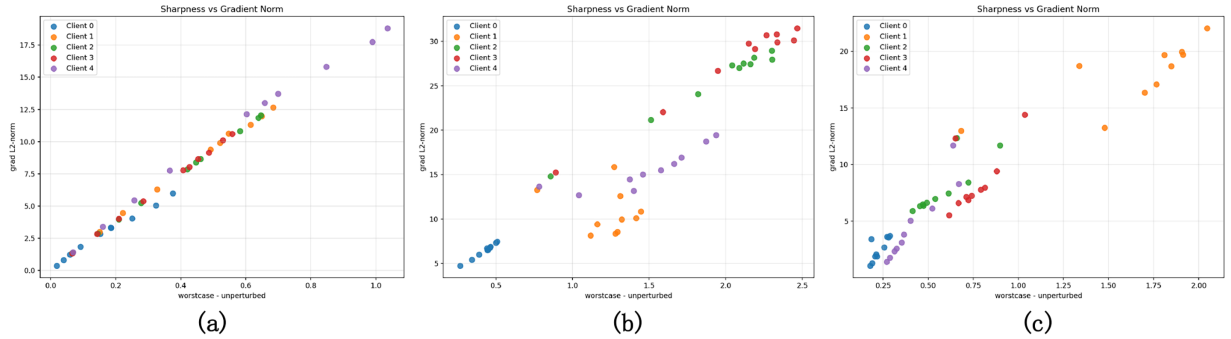

Figure 2: Scatter plots of sharpness vs. gradient norm on three benchmark datasets. (**a**) MNIST; (**b**) FMNIST; (**c**) CIFAR-10.

2 Related Work

This section reviews research progress in related fields from three dimensions: weighted aggregation strategies in Federated Learning, loss landscape theory in deep learning, and dynamic allocation mechanisms in differential privacy.

2.1 Utility-Based Federated Aggregation Strategies

Traditional FedAvg [28] adopts a weighting strategy based on sample size, which often leads the global model to be biased towards clients with more data but singular data distributions in Non-IID scenarios. To alleviate this issue, researchers have explored various utility-based weighted aggregation schemes [29]. One class of methods is contribution-based weighting, which allocates weights by assessing the marginal contribution of local models to global model performance. Study [30] proposed an adaptive client weighting mechanism based on Shapley values; however, calculating Shapley values typically requires exponential computational overhead, making it difficult to deploy in large-scale federated networks. Another class of methods is based on attention mechanism aggregation [31,32]. By calculating the layer-wise correlation or distance between the server model and client models, these methods automatically assign higher weights to clients that are more consistent with the global direction. However, they are easily susceptible to interference from parameter magnitude differences and directional noise. Some works attempt to use loss values or accuracy on a validation set as the basis for weighting, but this inevitably necessitates reliance on auxiliary data at the server side [15]. In contrast, our method utilizes geometric attributes during the local training process as the basis for aggregation.

2.2 Flat Minima and Generalization Theory in Deep Learning

The generalization capability of deep neural networks depends not only on finding a feasible solution with low loss but, more critically, on the geometric morphology of that solution within the loss landscape. If parameters are located in a wide, flat basin with small curvature, they exhibit stronger tolerance to small perturbations, random noise, or data distribution shifts, thereby performing stably on actual test data. Conversely, if located in sharp, narrow minima, minute parameter fluctuations can cause a sharp rise in loss, significantly degrading generalization performance.

The close relationship between the generalization capability of deep neural networks and the geometric structure of their convergence points can be traced back to the research of. Hochreiter and Schmidhuber [33]. Extensive theoretical and empirical studies [34–36] indicate that models converging to “flat minima” usually possess better generalization capabilities than those converging to “sharp minima.” This is because, in flat

regions, slight shifts in test data distribution do not lead to drastic fluctuations in loss values. To seek flat minima, the SAM algorithm proposed by Foret et al. explicitly minimizes the neighborhood maximum of the loss value by solving a Min-Max optimization problem in each iteration [23]. Subsequently, Adaptive SAM [37] further introduced the concept of adaptively adjusting the perturbation radius to accommodate the anisotropy of different parameter scales. Although flatness theory has matured in centralized training, existing Federated Learning works mostly employ SAM merely as a local optimizer for clients [26,38,39], aiming to enhance the robustness of single clients.

2.3 Refined Budget Management in Differential Privacy

To achieve a better balance between privacy protection and model utility, researchers are gradually shifting from static noise to more refined privacy budget management mechanisms. Based on dynamic factors such as local data volume, sample importance, user privacy preferences, and training stages, differentiated privacy budgets are allocated to different clients, layers, rounds, or even parameters, injecting noise on demand. Considering the varying contributions of different neural network layers (e.g., convolutional layers vs. fully connected layers) to feature extraction, some studies propose applying stronger noise to feature extraction layers and weaker noise to classification heads, or performing layer-wise noise injection based on parameter sparsity [40,41]. Another class of methods focuses on differences in user privacy protection needs, allowing different clients to set different clipping thresholds [42] or noise multipliers based on their own privacy preferences. In [19], Fisher information is proposed to measure parameter sensitivity, applying personalized differential privacy constraints to local parameters with high Fisher values. Currently, mainstream adaptive methods mainly dynamically adjust clipping thresholds based on parameter importance or data attributes [18–22]. Although these methods improve the privacy-utility trade-off to a certain extent, they are mainly adjusted based on statistics, ignoring the geometric sensitivity of data distribution. A large gradient does not completely equate to the model being in a vulnerable state; only by combining the curvature information of the loss landscape can one truly judge the model's tolerance to noise.

3 Preliminaries

This section formally defines the core theories and mathematical models involved in this paper, including the Federated Learning framework, Differential Privacy, and Sharpness-Aware Minimization.

3.1 Federated Learning Framework

FL aims to collaboratively train a global model across multiple clients while keeping data stored locally. Assume there are K clients in the system, indexed by $k \in \{1, 2, \ldots, K\}$. Each client k possesses a private dataset D_k, where $|D_k| = n_k$ denotes the number of samples. Let $N = \sum_{k=1}^{K} n_k$ be the total number of samples. The global optimization objective of Federated Learning is to minimize the weighted empirical risk across all clients:

$$\min_{\mathbf{w}} F(\mathbf{w}) = \sum_{k=1}^{K} p_k F_k(\mathbf{w}). \tag{1}$$

where $\mathbf{w}$ represents the global model parameters, $p_k = \frac{n_k}{N}$ is the aggregation weight for client k (typically based on sample size), and $F_k(\mathbf{w})$ is the local loss function (e.g., cross-entropy loss). The traditional FedAvg [28] algorithm optimizes this through multiple rounds of communication. The framework of FedAvg is shown in Fig. 3. In the t-th communication round:

- **Broadcast:** The server sends the current global model $\mathbf{w}_t$ to a selected subset of clients.

- **Local Update:** Client k initializes with $\mathbf{w}_t$ and runs E epochs of Stochastic Gradient Descent (SGD) on its local dataset to obtain the updated local model:

$$\mathbf{w}_k^{t+1} \leftarrow \mathbf{w}_t - \eta \nabla F_k(\mathbf{w}_t). \tag{2}$$

- **Aggregation:** The server collects updates from clients and computes the weighted average to generate the global model for the next round:

$$\mathbf{w}_{t+1} \leftarrow \sum_{k=1}^{K} p_k \mathbf{w}_k^{t+1}. \tag{3}$$

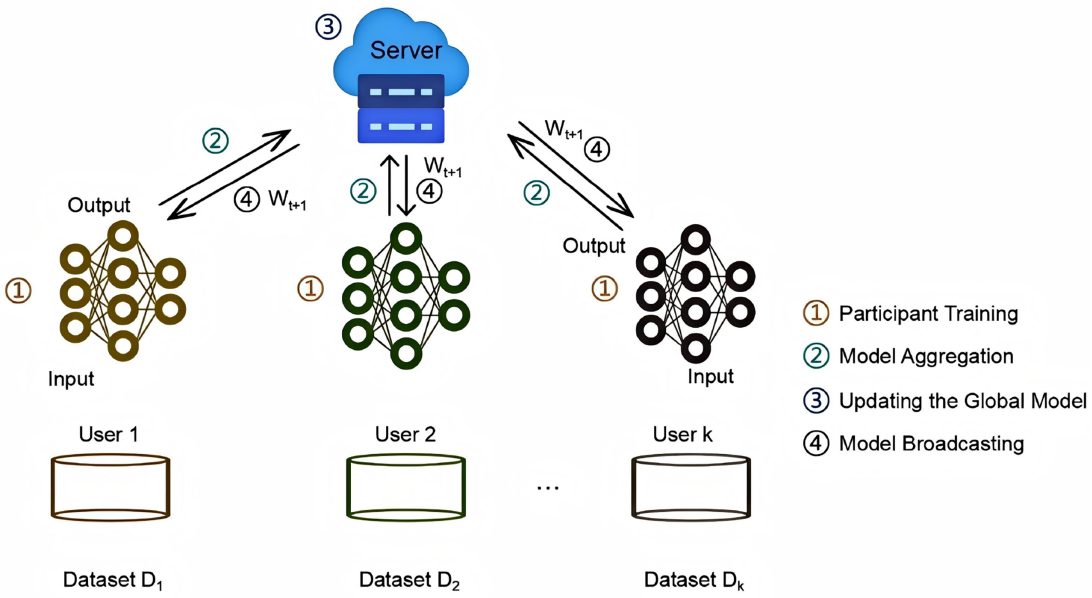

Figure 3: Federated learning framework diagram.

In real-world scenarios, data often exhibits Non-Independent and Identically Distributed (Non-IID) characteristics, meaning the data distribution varies across clients: $P_i(x, y) \neq P_j(x, y)$. This distribution shift causes the local optimal solution of each client to deviate from the global optimal solution. During local training, client models drift towards their respective local minima. This phenomenon is known as *Client Drift* [43]. It makes simple weighted averaging aggregation strategies difficult to converge to the global optimum, severely affecting the generalization performance of the model.

3.2 Differential Privacy Mechanism

To prevent attackers from reconstructing original data by analyzing uploaded model parameters or gradients, Differential Privacy (DP) is widely applied in Federated Learning. DP introduces randomness into the output, making it impossible for attackers to distinguish whether the calculation result comes from a dataset containing a target sample or one without it.

Definition 1 ((ϵ, δ)-DP): A randomized algorithm $\mathcal{M}$ satisfies (ϵ, δ)-DP if for any two adjacent datasets D and D', and for any output S within the range of the algorithm, the following condition holds:

$$\Pr[\mathcal{M}(D) \in S] \leq e^{\epsilon} \Pr[\mathcal{M}(D') \in S] + \delta. \tag{4}$$

where $\epsilon > 0$ is the privacy budget, limiting the upper bound of the difference in output probability distributions (a smaller ϵ implies stronger privacy protection), and δ is a very small slack term representing the minimal probability of privacy leakage.

The core of implementing differential privacy lies in adding noise based on the sensitivity of the function.

Definition 2 (L_2-Sensitivity): For a function f, the L_2-sensitivity Δf is defined as the maximum Euclidean distance of the outputs on any adjacent datasets D and D':

$$\Delta f = \max_{D,D'} \|f(D) - f(D')\|_2. \tag{5}$$

In deep learning, the *Gaussian Mechanism* is commonly used. Privacy protection is achieved by adding noise following a Gaussian distribution to the function output, where the standard deviation of the noise σ must satisfy:

$$\sigma \geq \frac{\Delta f}{\epsilon}\sqrt{2\ln(1.25/\delta)}. \tag{6}$$

It can be seen that sensitivity directly determines the magnitude of the noise required. In Federated Learning, since the gradient norm may be unbounded, gradients are usually clipped first to limit the upper bound of sensitivity, followed by noise injection. However, fixed noise levels often fail to adapt to the data characteristics of different clients, causing unnecessary loss of model utility.

3.3 Sharpness-Aware Minimization

Traditional Empirical Risk Minimization (ERM) focuses only on reducing the training loss value, easily leading to convergence to sharp minima [44]. Studies indicate [33,35] that models at sharp minima possess poor generalization capabilities and are extremely sensitive to parameter perturbations (such as DP noise).

The Loss Landscape characterizes the geometric morphology of the loss function in high-dimensional parameter space. If the optimal point is located in a wide, flat region (flat minimum), small perturbations in parameters cause almost no increase in loss, making the model more robust and generalizable. Conversely, if it is in a steep region (sharp minimum), minute changes can cause a sharp increase in loss, leading to degraded generalization performance. Fig. 4 illustrates the geometric robustness comparison between flat and sharp minima.

The Sharpness-Aware Minimization (SAM) algorithm aims to minimize both the training loss value and the sharpness of the loss landscape simultaneously. Its core idea is to find a neighborhood such that the maximum loss value within that neighborhood is minimized. The optimization objective of SAM can be formalized as the following Min-Max problem:

$$\min_{\mathbf{w}} \max_{\|\mathbf{e}\|_2 \leq \rho} L(\mathbf{w} + \mathbf{e}). \tag{7}$$

where ρ is the preset neighborhood radius (perturbation magnitude), and $\mathbf{e}$ is the perturbation vector in the parameter space. This formula intends to find parameters $\mathbf{w}$ such that the worst-case loss within its ρ-neighborhood remains low.

To solve the inner maximization problem, SAM uses a first-order Taylor expansion to approximate the optimal perturbation vector:

$$\hat{\mathbf{e}} = \rho \frac{\nabla_{\mathbf{w}} L(\mathbf{w})}{\|\nabla_{\mathbf{w}} L(\mathbf{w})\|_2}. \tag{8}$$

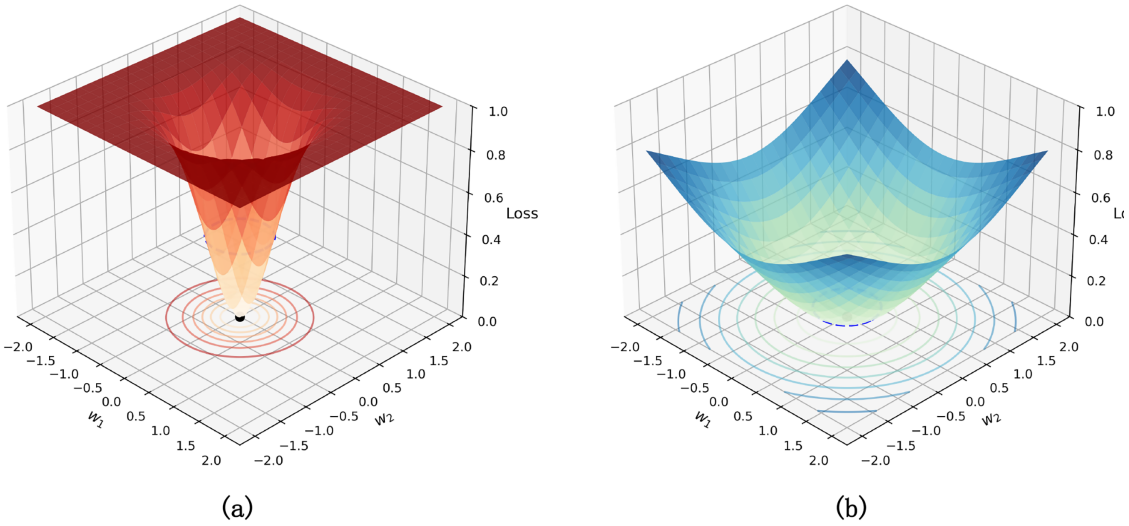

Figure 4: Schematic comparison of geometric robustness between sharp and flat minima. (**a**) A sharp minimum exhibits high sensitivity to parameter perturbations; (**b**) A flat minimum shows greater robustness against parameter perturbations.

The final model update is performed based on the gradient at this perturbed point:

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta \nabla_{\mathbf{w}} L(\mathbf{w}_t + \hat{\mathbf{e}}). \tag{9}$$

Fig. 5 shows the comparison of optimization trajectories between traditional SGD [45,46] and the SAM algorithm. Through this mechanism, SAM can guide the model to avoid steep valleys and converge to wide flat regions, whereas SGD tends to converge directly to the nearest local optimum along the direction of steepest descent, easily falling into sharp minima regions with poor generalization and high sensitivity to parameter perturbations.

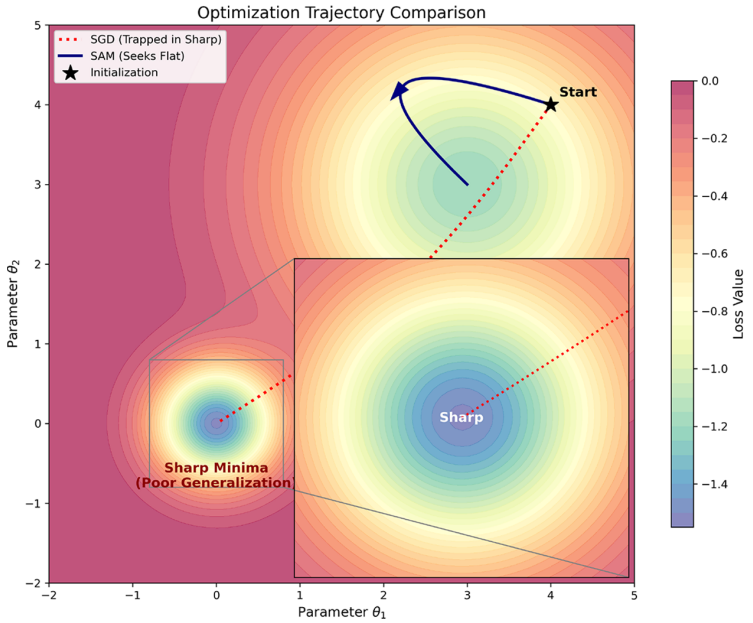

Figure 5: Comparison of optimization trajectories between traditional SGD and SAM algorithm.

4 Methodology

4.1 Framework Overview

To address the issues of model drift caused by data heterogeneity in Federated Learning and the utility loss resulting from traditional static noise strategies in Differential Privacy, this section proposes a novel framework named SAM-ADPFL. This framework aims to better balance the relationship between privacy and utility. The core idea is to utilize the geometric "sharpness" of the loss landscape as a bridge connecting model generalization capability with data privacy sensitivity. As shown in Fig. 6, SAM-ADPFL consists of three collaborative modules:

- **Local Sharpness-Aware Update:** Clients utilize the SAM algorithm instead of SGD for local updates. While seeking flat minima, they compute a sharpness score that characterizes the local data distribution properties.
- **Geometry-Aware Adaptive Aggregation:** Weights are dynamically reallocated based on the geometric characteristics of the local loss landscape. Higher aggregation weights are assigned to models in flat regions, while weights for models in sharp regions are reduced. Combined with a global weight shrinkage strategy, this effectively guides the global model to converge towards flat minima.
- **Sharpness-Guided Dynamic Privacy Protection:** The sensitivity of local data is dynamically quantified using the sharpness score. Based on this, the clipping threshold and noise injection intensity are adaptively adjusted to generate protected gradients that satisfy differential privacy requirements.

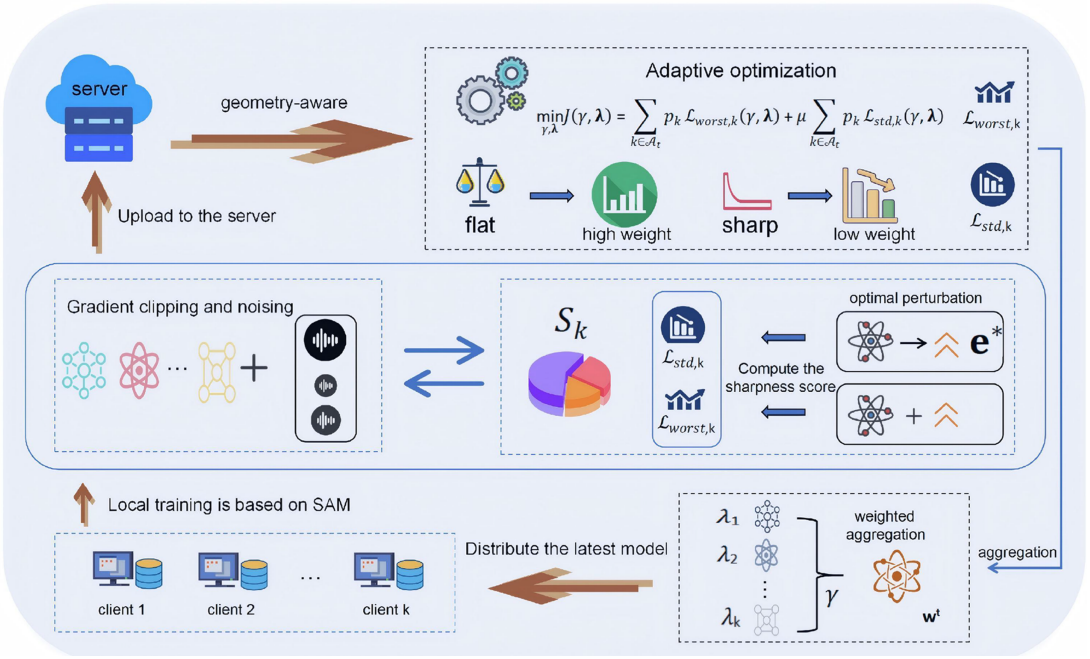

Figure 6: SAM-ADPFL framework diagram.

4.2 Local Sharpness Calculation and Measurement

To accurately quantify the data heterogeneity of each client while preserving privacy, we need an indicator that reflects the geometric characteristics of the local data distribution. We propose defining "sharpness" using the geometry of the loss landscape during training. Unlike the traditional perspective that focuses solely on the absolute value of the loss function, sharpness focuses on the rate of change of the loss function within a parameter neighborhood.

For the k-th client in the t-th communication round, let the local model parameter be $\mathbf{w}_k^{(t)}$. Inspired by the SAM algorithm, we posit that if a model is at a flat minimum, applying a small perturbation to its parameters should not cause drastic fluctuations in its loss value; conversely, if it is at a sharp minimum, a small perturbation will lead to a significant increase in the loss value.

Definition 3 (Sharpness Score): The difference between the maximum loss value achievable within a preset neighborhood radius ρ and the current original loss value. The formal definition is as follows:

$$S_k = \max_{\|\mathbf{e}\|_2 \leq \rho} \left(F_k(\mathbf{w}_k^{(t)} + \mathbf{e}) - F_k(\mathbf{w}_k^{(t)}) \right). \tag{10}$$

where $\mathbf{e}$ represents any perturbation vector in the parameter space, and $F_k(\cdot)$ is the local empirical loss function of client k. To solve the maximum loss problem mentioned above, the steepest perturbation vector is required. We use a first-order Taylor expansion to perform a local linear approximation of the objective function and derive the following proposition:

Proposition 1. *Assuming the local loss function $F_k(w)$ is L-smooth, for a sufficiently small radius ρ, the local sharpness can be approximated as $S_k \approx \rho||\nabla F_k(w_k^{(t)})||_2$. By Taylor's theorem, the residual approximation error is strictly bounded by $\frac{L}{2}\rho^2$.*

$$S_k \approx \rho \|\nabla F_k(\mathbf{w}_k^{(t)})\|_2. \tag{11}$$

Since clients need to compute gradients to solve for the perturbation direction when executing SAM optimization updates, the calculation of S_k incurs almost no additional computational overhead.

4.3 Geometry-Aware Adaptive Aggregation Mechanism

To prevent global model drift caused by local models overfitting to sharp minima in heterogeneous settings, we propose sharpness-guided weight redistribution. Instead of passive averaging, the server actively determines the optimal relative weight vector through a deterministic, closed-form calculation using client losses. In the t-th communication round, the server receives two key metrics uploaded by m selected clients. The first is the undisturbed loss $\mathcal{L}_{\text{std},k}$, i.e., the empirical risk $F_k(\mathbf{w}_k^{(t)})$ of client k on its local data, which characterizes the model's fitting degree to the current data. The second is the worst-case loss $\mathcal{L}_{\text{worst},k}$, i.e., the maximum loss $F_k(\mathbf{w}_k^{(t)} + \hat{\mathbf{e}})$ of client k after SAM perturbation, which reflects the model's generalization potential and local robustness.

To guide the global model towards flat minima without server-side validation or iterative optimization, we employ a deterministic, closed-form weighting mechanism. The adaptive weight λ_k exponentially penalizes clients with high sharpness (the gap between worst-case and standard loss):

$$\lambda_k = \frac{p_k \cdot \exp\left(-\mu \cdot (\tilde{\mathcal{L}}_{worst,k} - \tilde{\mathcal{L}}_{std,k})\right)}{\sum_{j \in \mathcal{A}_t} p_j \cdot \exp\left(-\mu \cdot (\tilde{\mathcal{L}}_{worst,j} - \tilde{\mathcal{L}}_{std,j})\right)} \tag{12}$$

Remark 1 (Server-Side Complexity): Computing (Eq. (12)) incurs a strictly linear complexity of $O(|\mathcal{A}_t|)$. This closed-form operation is highly scalable and completely eliminates any iterative backpropagation overhead on the server.

Where p_k is the prior base weight preserving the sample scale, and $\mu > 0$ is a temperature scalar controlling the sharpness penalty. A larger μ more aggressively penalizes models in sharp minima (i.e., a large gap between $\tilde{\mathcal{L}}_{worst,k}$ and $\tilde{\mathcal{L}}_{std,k}$). Geometrically, this closed-form strategy directly translates local landscape attributes into normalized weights, explicitly steering the global model away from steep valleys towards robust, flat minima without requiring iterative optimization.

4.4 Sharpness-Guided Dynamic Differential Privacy Mechanism

According to our analysis, high sharpness is usually accompanied by a high gradient norm. Therefore, for high-sharpness clients, we should appropriately loosen the clipping threshold to preserve more valuable gradient information; conversely, the threshold should be tightened to reduce the base of noise injection.

To prevent invalid negative bounds under extreme landscape deviations, the adaptive clipping threshold C_k for client k incorporates a strictly positive lower bound $C_{min} > 0$:

$$C_k = \max\left(C_{min}, C_{base} \cdot \left(1 + \alpha \frac{S_k - \overline{S}}{\overline{S}}\right)\right) \tag{13}$$

where C_{base} is the baseline threshold, $\overline{S}$ is the average sharpness, and α is the adjustment strength. This allows larger gradients when $S_k > \overline{S}$, while safely truncating the threshold at C_{min} when $S_k < \overline{S}$.

To satisfy differential privacy, larger clipping bounds necessitate proportionally larger noise. We link the Gaussian noise standard deviation σ_k directly to sharpness. To eliminate division-by-zero risks when models converge to exceptionally flat minima ($\min_j(S_j) \to 0$), we introduce a smoothing constant ξ (e.g., 10^{-4}):

$$\sigma_k = \sigma_{base} \cdot \sqrt{\frac{S_k}{\min_j(S_j) + \xi}} \tag{14}$$

where σ_{base} is the baseline noise multiplier. $\min_j(S_j)$ is the minimum sharpness score in the current round. Finally, the privacy-protected gradient uploaded by client k is calculated as:

$$\tilde{g}_k = Clip(g_k, C_k) + \mathcal{N}(0, (C_k\sigma_k)^2 I) \tag{15}$$

Prevention of Scalar Leakage: To strictly close the side-channel privacy leakage associated with the uploaded variables, clients first clip the sharpness and loss values to predefined maximum bounds B_S and B_L to fix their L_1-sensitivities ($\Delta S = B_S$ and $\Delta L = B_L$). Then, clients inject lightweight Laplacian noise before uploading: $\tilde{S}_k = S_k + \text{Lap}(\Delta S/\ \epsilon_S)$, and identically for the loss values $\tilde{\mathcal{L}}_{std,k}$ and $\tilde{\mathcal{L}}_{worst,k}$ using ϵ_L. We allocate a small, fixed budget $\epsilon_{scalars}$ evenly across these three scalars ($\epsilon_S = \epsilon_L = \epsilon_{scalars}\ /3$). By the basic composition theorem, this $\epsilon_{scalars}$ is directly added to the overall privacy budget. This ensures the geometric weights in the server are computed over a strictly differentially private manifold.

Theorem 1. For any target privacy parameter $\delta \in (0,1)$ and a total of T communication rounds, the complete SAM-ADPFL training procedure rigorously satisfies (ϵ, δ)-DP under the Rényi Differential Privacy (RDP) framework. The total privacy budget ϵ is strictly bounded by:

$$\epsilon = \min_{q>1}\left(\frac{Tq}{2\sigma_{base}^2} + \frac{\ln(1/\delta)}{q-1}\right) + \epsilon_{scalars} \tag{16}$$

where q is the RDP order, σ_{base} is the baseline noise multiplier, and $\epsilon_{scalars}$ is the constant privacy budget consumed by the scalar perturbation. The detailed proof is provided in Appendix A.

To intuitively explain the internal logic of sharpness and privacy budget allocation, we compare two typical local geometric scenarios in Fig. 7. As shown in Fig. 7a, when the client model is located in a high-sharpness region (red terrain), the loss function is extremely steep, leading to a significant increase in the gradient L_2-norm. To mask this feature and prevent severe distortion during gradient clipping, the clipping threshold is adaptively adjusted, and a Gaussian noise distribution with larger variance is matched. In contrast, as shown in Fig. 7b, when the model is located in a flat region (blue terrain), the gradient is small

and insensitive to parameter perturbation. The clipping threshold is adaptively tightened, and the noise level is significantly reduced. Algorithm 1 outlines the pseudocode of our framework.

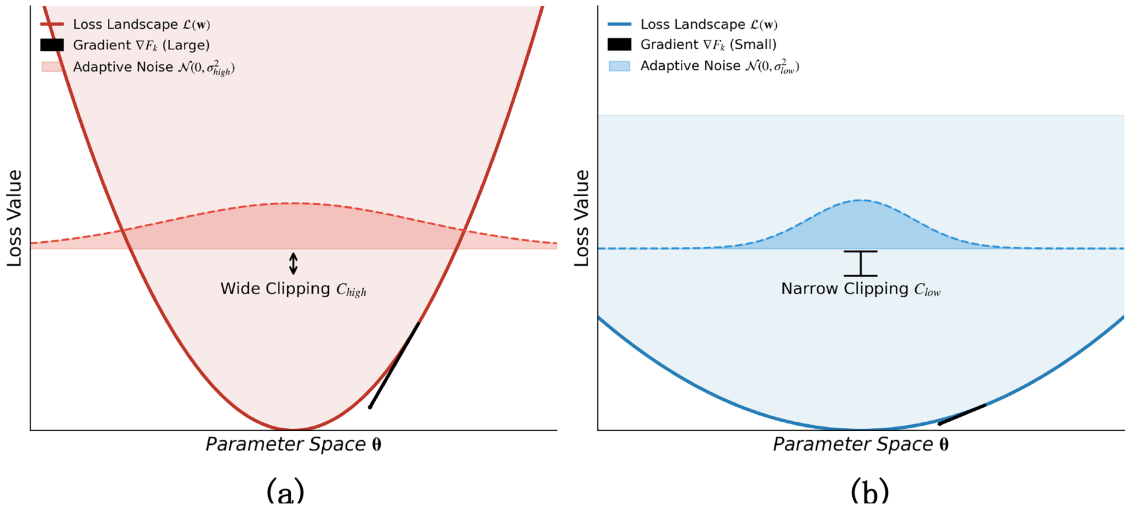

Figure 7: Schematic diagram of sharpness-guided adaptive differential privacy mechanism. (**a**) High-sharpness scenario with large gradient, wide clipping bound, and strong noise injection; (**b**) Low-sharpness scenario with small gradient, narrow clipping bound, and weak noise injection.

Algorithm 1: SAM-ADPFL training process

Input: Initial model $\mathbf{w}_0$, Client set $\mathcal{A}_t$, Rounds T, LR η, SAM radius ρ, DP params $C_{base}, \sigma_{base}, \alpha, C_{min}, \xi$, Prior weights $\{p_k\}$.

Output: Trained global model $\mathbf{w}_T$.

Server: Initialize $\mathbf{w}_0$;

for $t = 0, \ldots, T-1$ **do**

for *client* $k \in \mathcal{A}_t$ *in parallel* **do**

Step 1: Sharpness Evaluation & SAM Update;

Compute local grad $\mathbf{g}_k \leftarrow \nabla F_k(\mathbf{w}_t)$ and perturbation $\mathbf{e} \leftarrow \rho \frac{\mathbf{g}_k}{\|\mathbf{g}_k\|_2}$;

Estimate sharpness $S_k \approx \rho \|\mathbf{g}_k\|_2$;

Compute losses: $\mathcal{L}_{std,k} \leftarrow F_k(\mathbf{w}_t)$, $\mathcal{L}_{worst,k} \leftarrow F_k(\mathbf{w}_t + \mathbf{e})$;

Step 2: Adaptive Privacy Protection;

if $t > 0$ **then**

$C_k \leftarrow \max\left(C_{min}, C_{base} \cdot \left(1 + \alpha \frac{S_k - \bar{S}}{\bar{S}}\right)\right)$;

$\sigma_k \leftarrow \sigma_{base} \cdot \sqrt{\frac{S_k}{S_{min} + \xi}}$;

else

$C_k \leftarrow C_{base}$; $\sigma_k \leftarrow \sigma_{base}$;

end

Generate DP gradient: $\tilde{\mathbf{g}}_k \leftarrow \text{Clip}(\mathbf{g}_k, C_k) + \mathcal{N}(0, (C_k \sigma_k)^2 \mathbf{I})$;

Generate DP scalars: $\tilde{S}_k \leftarrow S_k + \text{Lap}(\Delta S / \epsilon_s)$;

$\tilde{\mathcal{L}}_{*,k} \leftarrow \mathcal{L}_{*,k} + \text{Lap}(\Delta L / \epsilon_L)$;

Upload $\tilde{\mathbf{g}}_k, \tilde{S}_k, \tilde{\mathcal{L}}_{std,k}, \tilde{\mathcal{L}}_{worst,k}$ to Server;

(Continued)

Algorithm 1 (continued)

end
Server:;
Compute closed-form adaptive weights λ_k^* (Eq. (12)) using uploaded DP scalars;
Update global model: $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta \sum_{k \in \mathcal{A}_t} \lambda_k^* \tilde{\mathbf{g}}_k$;
Compute global geometric bounds $\bar{S} = \frac{1}{|\mathcal{A}_t|} \sum \tilde{S}_k$ and $S_{min} = \min_k \tilde{S}_k$ for the next round;
end
return $\mathbf{w}_T$

5 Experiments

In this section, we conducted extensive experiments on three datasets to evaluate the high utility of the proposed SAM-ADPFL and its superior balance between privacy protection and model utility. We divide the experimental evaluation into two independent but progressive stages: Phase 1: Validation of the effectiveness of the geometry-aware aggregation strategy (non-differential privacy scenario), and Phase 2: Evaluation of sharpness-guided adaptive privacy protection performance (differential privacy scenario).

5.1 Experiment Setup

Datasets: MNIST [47], Fashion-MNIST, and CIFAR-10 [48].

Data Partitioning: To simulate Non-IID data heterogeneity, we partition client datasets using a Dirichlet distribution. The concentration parameter α controls heterogeneity: $\alpha = 0.1$ for extreme skewness and $\alpha = 0.5$ for moderate imbalance.

For MNIST and Fashion-MNIST, we adopt an improved LeNet-5 comprising two convolutional layers (5×5 kernels, 16 and 32 channels) with ReLU and 2×2 max pooling, followed by a fully connected (FC) layer. For CIFAR-10, we employ a VGG-like network with three stacked convolutional modules (each containing two 3×3 conv layers and one max pooling layer, with channels scaling from 32 to 128) and two FC layers (hidden size 128).

Hyperparameters and Reproducibility: We use $E = 5$ local epochs, batch size 64, and $T = 200$ global rounds. For scale-up evaluations (e.g., $K = 50$), we uniformly sample $m = 10$ clients without replacement per round. The Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 10^{-4}) is applied with learning rates of 0.005 for MNIST/Fashion-MNIST and 0.01 for CIFAR-10. Privacy budgets are tracked via the Rényi Differential Privacy (RDP) accountant. All tabular results report the mean and standard deviation across 5 independent runs (seeds $\{42, 43, 44, 45, 46\}$). PyTorch source code is available from the corresponding author upon reasonable request.

We compare SAM-ADPFL with the following mainstream Federated Learning algorithms:

- **FedAvg [28]:** As the standard baseline method for Federated Learning, used to measure basic performance.
- **FedProx [9]:** Introduces a proximal term in the local objective function to restrict local updates from deviating, representing a classic regularization method for solving Non-IID problems.
- **SCAFFOLD [10]:** Uses control variates to correct local gradient directions, representing one of the advanced methods for solving model drift.
- **FedLAW [15]:** Optimizes aggregation weights via a server-side validation set. In particular, we focus on comparing the performance of our method with FedLAW to demonstrate that SAM-ADPFL can achieve or even surpass aggregation effects dependent on auxiliary data using only sharpness information, without relying on any server-side proxy data.

- **DP-FedAvg [16]:** A classic differential privacy Federated Learning algorithm employing fixed clipping thresholds and fixed noise standard deviations, serving as a benchmark for static noise strategies.
- **DP-FedSAM [26]:** Applies static differential privacy noise on top of FedSAM.
- **DP-SCAFFOLD [10]:** A differentially private variant of the SCAFFOLD algorithm, serving as a robust structural baseline designed to handle Non-IID data distributions under strict privacy constraints.

All experiments in this paper were run on a workstation equipped with a 12th Gen Intel(R) Core(TM) i9-12900K CPU and an NVIDIA RTX A5000 GPU.

5.2 Performance Evaluation of Geometry-Aware Adaptive Aggregation (Noise-Free Scenario)

To isolate the impact of differential privacy and purely evaluate our "Geometry-Aware Adaptive Aggregation" against data heterogeneity, we introduce its non-DP variant, SAM-AFL. Before comparing baselines, we first optimize the perturbation radius ρ. Since ρ dictates the search range for flat minima, an optimal value is crucial: an excessively small ρ fails to capture landscape sharpness, whereas a large ρ induces excessive gradient noise and training instability.

We conducted sensitivity tests for $\rho \in \{0.01, 0.03, 0.05, 0.10, 0.20\}$ under extreme heterogeneous scenarios (Dirichlet $\alpha = 0.1$) for all three datasets. The experimental results, as shown in Fig. 8, indicate that the model accuracy reached its maximum value at $\rho = 0.05$ for all three datasets. In subsequent comparative experiments, we selected the best-performing $\rho = 0.05$ as the primary configuration and retained $\rho = 0.03$ to demonstrate the algorithm's stability under different regularization strengths. We compare it with FedAvg, FedProx, SCAFFOLD, and FedLAW. To fully exploit the upper performance limit of FedLAW, we configured it with a server-side proxy dataset with a very ideal distribution and balanced classes in the experiment.

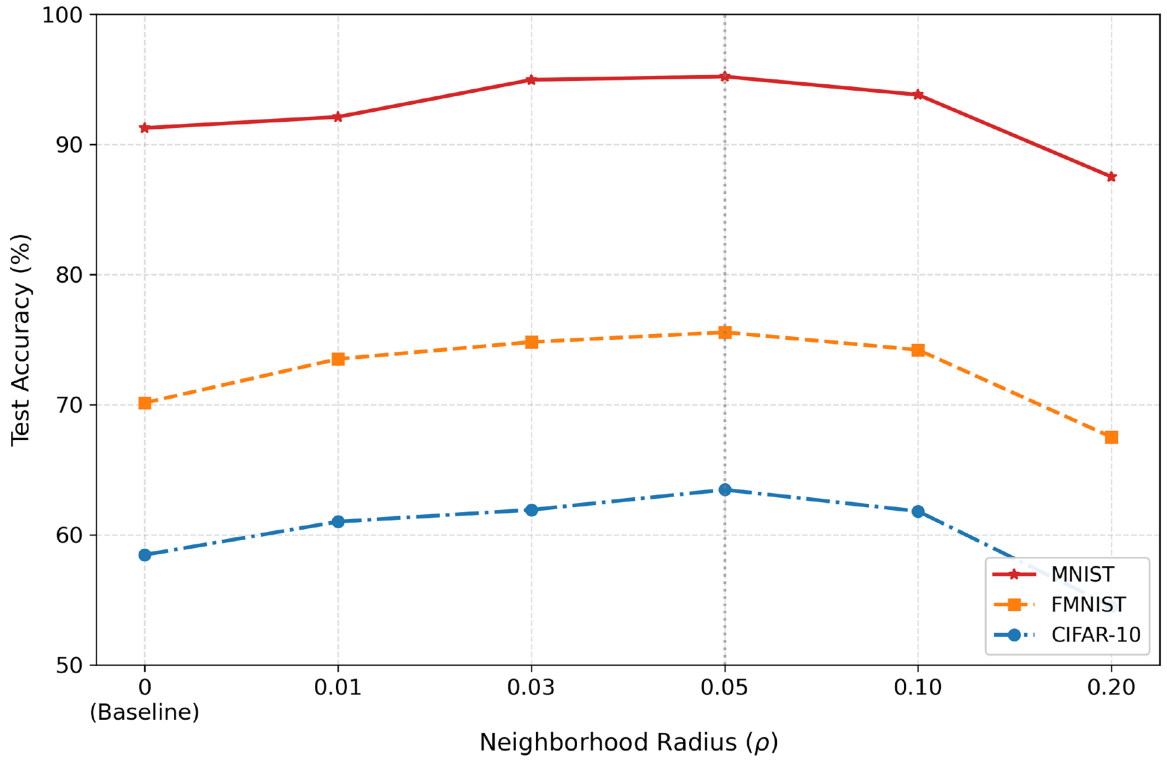

Figure 8: Sensitivity analysis of neighborhood radius.

Table 1 summarizes test accuracies across the MNIST, FMNIST, and CIFAR-10 datasets under varying heterogeneity (α = 0.1, α = 0.5, and IID). Under extreme heterogeneity (α = 0.1), traditional methods struggle with sharp local minima and global drift; for example, on CIFAR-10, FedAvg achieves only 58.45%. While FedProx (60.20%) and SCAFFOLD (61.65%) provide limited improvements, SAM-AFL (ρ = 0.05) reaches 63.40%. Crucially, SAM-AFL outperforms FedLAW on both CIFAR-10 (63.40% vs. 62.88%) and FMNIST (75.85% vs. 75.20%) without relying on any auxiliary proxy data. This confirms that directly leveraging loss landscape geometry for aggregation generalizes better than fitting external proxy data.

Table 1: Performance evaluation of algorithms under different data heterogeneity settings (Dirichlet $\alpha = 0.1, 0.5$ and IID).

Dataset	Algorithm	Dirichlet α = 0.1 Accuracy (%)	Dirichlet α = 0.5 Accuracy (%)	IID Accuracy (%)
MNIST	FedAvg	91.25 ± 0.63	94.41 ± 0.28	96.77 ± 0.54
	FedProx	92.85 ± 0.41	94.95 ± 0.33	96.90 ± 0.17
	SCAFFOLD	93.10 ± 0.45	95.60 ± 0.23	97.15 ± 0.08
	FedLAW	93.50 ± 0.37	96.25 ± 0.51	97.60 ± 0.45
	SAM-AFL (ρ = 0.03)	92.95 ± 0.29	95.55 ± 0.17	97.82 ± 0.65
	SAM-AFL (ρ = 0.05)	95.20 ± 0.24	96.88 ± 0.44	97.65 ± 0.89
FMNIST	FedAvg	70.13 ± 0.32	79.45 ± 0.46	82.34 ± 0.84
	FedProx	72.50 ± 0.81	80.90 ± 0.35	83.05 ± 0.67
	SCAFFOLD	73.85 ± 0.34	81.25 ± 0.56	83.70 ± 0.36
	FedLAW	75.20 ± 0.94	84.10 ± 0.47	84.35 ± 0.83
	SAM-AFL (ρ = 0.03)	75.10 ± 0.02	83.65 ± 0.60	84.60 ± 0.11
	SAM-AFL (ρ = 0.05)	75.85 ± 0.51	85.20 ± 0.65	84.88 ± 0.35
CIFAR10	FedAvg	58.45 ± 0.47	62.38 ± 0.24	64.99 ± 0.42
	FedProx	60.20 ± 0.43	63.85 ± 0.06	65.70 ± 0.76
	SCAFFOLD	61.65 ± 0.74	64.50 ± 0.83	67.45 ± 0.05
	FedLAW	62.88 ± 0.90	65.25 ± 1.82	69.20 ± 0.51
	SAM-AFL (ρ = 0.03)	62.91 ± 0.24	68.60 ± 0.47	69.65 ± 0.86
	SAM-AFL (ρ = 0.05)	63.40 ± 0.66	69.35 ± 0.45	69.30 ± 0.31

Fig. 9 shows the impact of client number $K \in \{5, 10, 20, 30, 50\}$ on algorithm performance under extreme heterogeneous scenarios (Dirichlet α = 0.1). As K increases, the dilution of data volume per client exacerbates the risk of local overfitting. In all three datasets, SAM-AFL (ρ = 0.05) can still reach accuracies of 93%, 75.76%, and 66.23% respectively when the number of clients reaches 50, demonstrating excellent resistance to sparsity. Its accuracy curve is the flattest, indicating that its performance is not significantly affected by the number of clients. Although FedLAW maintained good performance with the help of ideal server-side proxy data, SAM-AFL achieved comprehensive superiority without relying on auxiliary data.

Fig. 10 evaluates the ability of each algorithm to combat model drift by setting local training epochs $E \in \{1, 5, 10, 20\}$. Experimental results show that FedAvg suffers from severe catastrophic forgetting due to overfitting local heterogeneous distributions at E = 20, causing a precipitous drop in accuracy. Although SCAFFOLD corrects the drift direction through control variates and achieves continuous performance

improvement with E, its convergence upper bound is still lower than that of the method in this paper. SAM-AFL (ρ = 0.05) achieved optimal performance in all settings, maintaining extremely high accuracy even under the large epoch setting of E = 20.

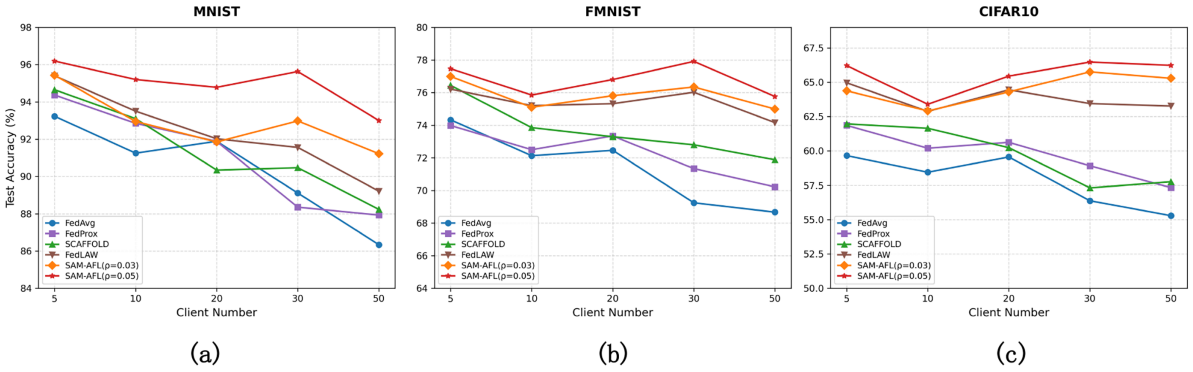

Figure 9: Accuracy comparison under different numbers of clients. (**a**) Results on the MNIST dataset; (**b**) Results on the FMNIST dataset; (**c**) Results on the CIFAR-10 dataset.

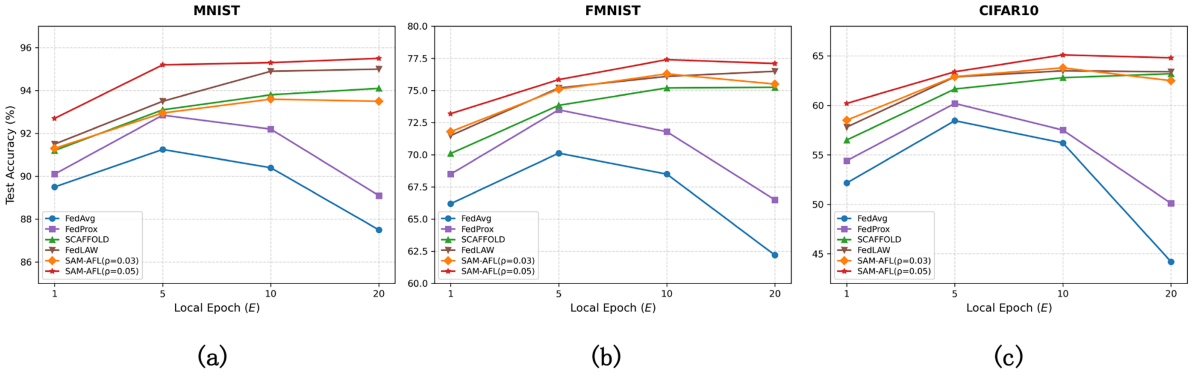

Figure 10: Accuracy comparison under different local training epochs. (**a**) Results on the MNIST dataset; (**b**) Results on the FMNIST dataset; (**c**) Results on the CIFAR-10 dataset.

5.3 Performance Evaluation of Sharpness-Guided Adaptive Privacy Protection (Noise Scenario)

To comprehensively evaluate SAM-ADPFL under strict privacy constraints and extreme data heterogeneity (Dirichlet α = 0.1), we conducted experiments on MNIST, FMNIST, and CIFAR-10. We compare our framework against DP-FedAvg, DP-FedSAM, DP-SCAFFOLD, and DP-FedLAW (extended from [15] for consistent privacy). We first verify the adaptive clipping mechanism by comparing SAM-ADPFL (ρ = 0.05) against baselines using various fixed thresholds ($C \in \{0.5, 1.0, 2.0, 4.0, 8.0\}$) under an identical per-round privacy budget. Subsequently, we explore the overall privacy-utility trade-off.

As shown in Fig. 11, all fixed-threshold baselines exhibit a distinct inverted-U trend across the three datasets, confirming the severe utility degradation caused by either excessive gradient truncation (small C) or noise explosion (large C). Taking CIFAR-10 as an example, baseline accuracies peak at C = 4.0 (e.g., DP-FedLAW reaches 58.79%, DP-FedSAM 57.27%, and DP-SCAFFOLD 54.60%). In stark contrast, our SAM-ADPFL (ρ = 0.05) circumvents this sensitivity trade-off entirely through adaptive clipping. It maintains a dominant accuracy of 61.95% regardless of the baseline thresholds, yielding substantial improvements of over 3.16% against the strongest baselines.

Based on the results in Fig. 11, to ensure the rigor of subsequent comparisons, in the next phase of privacy budget experiments, we selected the best-performing fixed clipping threshold from Fig. 12 for all baseline methods (C = 2.0 for MNIST and FMNIST, C = 4.0 for CIFAR-10), while SAM-ADPFL continued to use adaptive clipping. We further explored the noise robustness of each algorithm under different privacy budgets $\epsilon \in \{0.5, 1.0, 2.0, 5.0, 10.0\}$.

Fig. 12 illustrates the privacy-utility trade-off across varying privacy budgets (ϵ). While the newly added DP-SCAFFOLD outperforms DP-FedAvg by utilizing control variates, it remains consistently inferior to DP-FedSAM. This empirically confirms that navigating towards flat minima provides fundamentally stronger resilience to DP noise than post-hoc drift correction. Building upon this geometric robustness, our SAM-ADPFL achieves state-of-the-art performance across all settings. Under strict privacy constraints (e.g., ϵ = 2.0 on CIFAR-10), SAM-ADPFL (ρ = 0.05) attains a dominant accuracy of 51.15%, substantially outperforming DP-FedSAM and surpassing even the proxy-data-assisted DP-FedLAW. These results conclusively demonstrate that dynamically tailoring noise allocation via local sharpness optimally preserves gradient utility without compromising strict privacy.

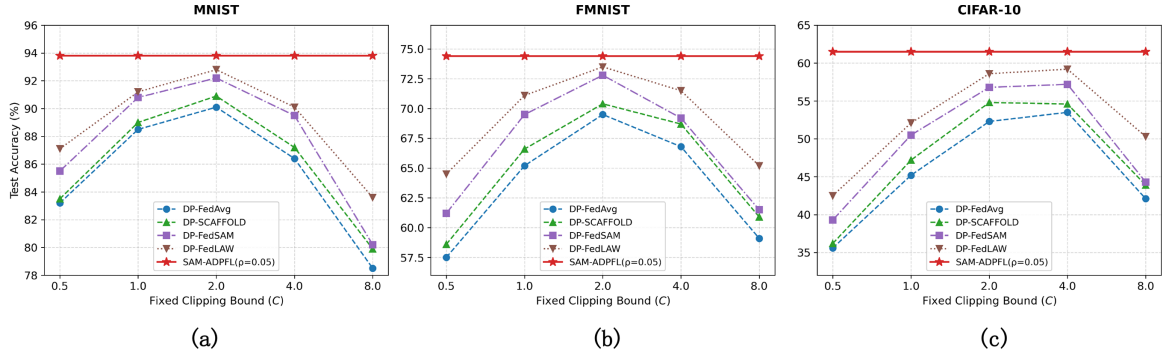

Figure 11: Accuracy comparison under different clipping thresholds. (**a**) Results on the MNIST dataset; (**b**) Results on the FMNIST dataset; (**c**) Results on the CIFAR-10 dataset.

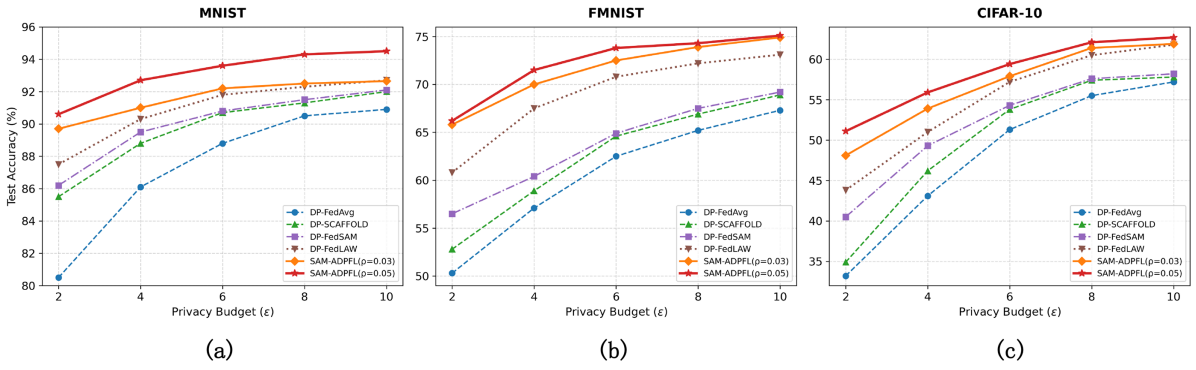

Figure 12: Accuracy comparison under different privacy budgets. (**a**) Results on the MNIST dataset; (**b**) Results on the FMNIST dataset; (**c**) Results on the CIFAR-10 dataset.

To evaluate communication efficiency, Table 2 records the rounds required to reach target accuracies under a fixed privacy budget (ϵ = 10). While DP-SCAFFOLD converges faster than DP-FedAvg by mitigating client drift, SAM-ADPFL achieves the fastest convergence across all datasets. On CIFAR-10, SAM-ADPFL

requires only 56 rounds, reducing communication overhead by 60.6% against DP-FedAvg and 37.1% against the strongest baseline, DP-FedLAW.

Table 2: Communication rounds required to achieve target accuracy.

Dataset	Target Accuracy	DP-FedAvg	DP-SCAFFOLD	DP-FedSAM	DP-FedLAW	SAM-ADPFL (ρ = 0.05)
MNIST	85%	98	72	68	55	39
FMNIST	65%	127	85	69	63	55
CIFAR10	55%	142	121	104	89	56

Table 3 reports the total running time required for convergence. Although the SAM optimizer doubles local gradient computations per step, the massive reduction in communication rounds entirely offsets this overhead. On CIFAR-10, SAM-ADPFL converges in just 1996.17s, reducing total time by 47.4% against DP-FedAvg, 42.7% against DP-SCAFFOLD, and 28.0% against DP-FedSAM. This confirms that our geometry-aware adaptive strategy not only enhances noise resilience but also significantly accelerates the end-to-end efficiency of the federated system.

Table 3: Running time required to achieve target accuracy.

Dataset	Target Accuracy	DP-FedAvg	DP-SCAFFOLD	DP-FedSAM	DP-FedLAW	SAM-ADPFL (ρ = 0.05)
MNIST	85%	1725.7789 s	1521.5620 s	1350.1105 s	1498.3231 s	1195.0349 s
FMNIST	65%	2238.5063 s	1785.2241 s	1579.4556 s	1804.9478 s	1439.0021 s
CIFAR10	55%	3792.5651 s	3486.7415 s	2771.2750 s	3035.4095 s	1996.1735 s

5.4 Ablation Study

To evaluate the individual contributions of local SAM training, geometry-aware aggregation, and adaptive DP, we designed four configurations: Config A replaces our aggregation with standard FedAvg; Config B replaces local SAM with SGD; Config C replaces adaptive DP with fixed noise; and Config D is the full SAM-ADPFL. Table 4 presents the ablation results on CIFAR-10 under extreme heterogeneity (Dirichlet α = 0.1) and a strict privacy budget (ϵ = 2.0).

Table 4: Ablation results on CIFAR-10.

Configuration	Local Opt.	Aggregation	DP Mechanism	Accuracy	Rounds
A	SAM	FedAvg	Adaptive DP	45.12 ± 0.45	98
B	SGD	Geometry	Adaptive DP	41.38 ± 0.61	115
C	SAM	Geometry	Fixed DP	47.85 ± 0.53	82
D (ours)	SAM	Geometry	Adaptive DP	51.45 ± 0.38	65

The results confirm that all three components are highly synergistic and indispensable. Replacing SAM with SGD (Config B) yields the lowest accuracy, proving that searching for flat minima is foundational for geometric noise tolerance. Removing the geometry-aware aggregation (Config A) fails to suppress severe client drift in Non-IID settings. Furthermore, reverting to fixed DP noise (Config C) noticeably degrades utility, highlighting that our sharpness-guided dynamic DP effectively minimizes unnecessary noise in flat

regions. Ultimately, the full SAM-ADPFL (Config D) achieves the highest accuracy and fastest convergence, validating our integrated architectural design.

6 Conclusion

This paper introduces SAM-ADPFL, an adaptive software framework for federated learning systems operating under statistical heterogeneity and strict privacy constraints. The proposed geometry-aware aggregation mechanism and sharpness-guided dynamic privacy module collectively address model drift and privacy-utility trade-offs through principled engineering of loss landscape geometry. Comprehensive experiments confirm superior performance in accuracy, noise robustness, and convergence speed. These findings provide empirical evidence and design patterns for developing quality-assured, privacy-preserving distributed software systems.

In future work, we plan to extend this algorithmic foundation into more complex real-world distributed software systems, explicitly addressing practical system-level challenges such as asynchronous updates, large-scale partial participation, and dynamic client dropouts.

Acknowledgement: The authors would like to express their sincere gratitude to all individuals who provided valuable assistance and support to this research.

Funding Statement: This work is jointly supported by the National Natural Science Foundation of China (Grant No. 62302540, author Fangfang Shan; https://www.nsfc.gov.cn), the Key Research and Development Program of Henan Province (Grant No. 251111212000, author Fangfang Shan; http://xt.hnkjt.gov.cn/data/), and the Industry-University-Research Innovation Fund of Chinese Universities (Special Project on AI+ Cybersecurity Governance Technology) (Grant No. 2025SE052, author Fangfang Shan; https://www.cutech.edu.cn).

Author Contributions: The authors confirm contribution to the paper as follows: Fangfang Shan conducted conceptualization, formal analysis and schematic design, and was responsible for manuscript review, editing and project supervision; Yuhang Liu performed the experiments and completed relevant investigation, as well as drafting the original manuscript; Lulu Fan participated in the investigation, wrote the original draft, and revised and edited the manuscript; Yifan Mao took charge of software development, visualization production and manuscript revision; Zhuo Chen finished the experimental writing and participated in manuscript review and editing; Peixue Wang engaged in manuscript review and editing and co-supervised this research. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available from the public repositories: MNIST (http://yann.lecun.com/exdb/mnist/), Fashion-MNIST (https://github.com/zalandoresearch/fashion-mnist), and CIFAR-10 (https://www.cs.toronto.edu/~kriz/cifar.html).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A Proof of Theorem 1

To explicitly address the coupling of the dynamically changing clipping threshold $C_k^{(t)}$ and noise multiplier $\sigma_k^{(t)}$, we analyze the localized privacy cost exclusively using the Rényi Differential Privacy (RDP) framework.

In any communication round t, the explicit gradient clipping structurally bounds the L_2-sensitivity of the local update for client k to exactly $\Delta f \leq C_k^{(t)}$. According to Eq. (15), the injected Gaussian noise has a variance of $(C_k^{(t)}\sigma_k^{(t)})^2$. By the properties of the Gaussian mechanism in RDP, applying noise with variance ν^2

to a query with sensitivity Δf satisfies $(q, \frac{q\Delta f^2}{2v^2})$-RDP for any order $q > 1$. Substituting our coupled adaptive parameters, the RDP cost $\epsilon_k^{(t)}(q)$ is:

$$\epsilon_k^{(t)}(q) = \frac{q(\Delta f)^2}{2\left(C_k^{(t)}\sigma_k^{(t)}\right)^2} \leq \frac{q\left(C_k^{(t)}\right)^2}{2\left(C_k^{(t)})^2(\sigma_k^{(t)}\right)^2} = \frac{q}{2\left(\sigma_k^{(t)}\right)^2} \tag{A1}$$

This calculation elegantly cancels out the dynamic clipping bound $C_k^{(t)}$, mathematically isolating the noise multiplier $\sigma_k^{(t)}$ as the sole variable determining the privacy cost. Based on the adaptive allocation strategy defined in Eq. (14), we have $\sigma_k^{(t)} = \sigma_{base} \cdot \sqrt{S_k/(\min_j(S_j) + \xi)}$. Since the condition $S_k \geq \min_j(S_j)$ strictly holds for all participating clients, it is guaranteed that $\sigma_k^{(t)} \geq \sigma_{base}$. Consequently, the per-round privacy cost is universally bounded independent of individual client geometry:

$$\epsilon_k^{(t)}(q) \leq \frac{q}{2\sigma_{base}^2} \tag{A2}$$

To account for the complete training process, we utilize the RDP Sequential Composition Theorem. Because the training spans T independent communication rounds, the cumulative RDP cost is linearly bounded by $\frac{Tq}{2\sigma_{base}^2}$.

Finally, applying the standard RDP-to-(ϵ, δ)-DP conversion lemma, the model update process guarantees $(\epsilon_{model}, \delta)$-DP, where $\epsilon_{model} = \min_{q>1}\left(\frac{Tq}{2\sigma_{base}^2} + \frac{\ln(1/\delta)}{q-1}\right)$. Because the complete algorithm also uploads geometry scalars protected by Laplacian noise (consuming a strict $\epsilon_{scalars}$ budget), the total privacy cost is the sum of these two components. This directly yields the formal bound stated in Theorem 1 and concludes the proof. □

References

1. Bharati S, Mondal MRH, Podder P, Prasath VS. Federated learning: applications, challenges and future directions. Int J Hybrid Intell Syst. 2022;18(1–2):19–35.
2. Yin X, Zhu Y, Hu J. A comprehensive survey of privacy-preserving federated learning: a taxonomy, review, and future directions. ACM Comput Surv. 2021;54(6):1–36. doi:10.1145/3460427.
3. Zhang J, Li M, Zeng S, Xie B, Zhao D. A survey on security and privacy threats to federated learning. In: Proceedings of the 2021 International Conference on Networking and Network Applications (NaNA); 2021 Oct 29–Nov 1; Lijiang, China. p. 319–26.
4. Al-Rubaie M, Chang JM. Reconstruction attacks against mobile-based continuous authentication systems in the cloud. IEEE Trans Inf Forensics Secur. 2016;11(12):2648–63. doi:10.1109/tifs.2016.2594132.
5. Shokri R, Stronati M, Song C, Shmatikov V. Membership inference attacks against machine learning models. In: Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP); 2017 May 22–26; San Jose, CA, USA. p. 3–18.
6. Shokri R, Shmatikov V. Privacy-preserving deep learning. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security; 2015 Oct 12–16; Denver, CO, USA. p. 1310–21.
7. Jayaraman B, Evans D. Evaluating differentially private machine learning in practice. In: Proceedings of the 28th USENIX Security Symposium (USENIX Security 19); 2019 Aug 14–16; Santa Clara, CA, USA. p. 1895–912.
8. Xiang L, Yang J, Li B. Differentially-private deep learning from an optimization perspective. In: Proceedings of the IEEE INFOCOM 2019-IEEE Conference on Computer Communications; 2019 Apr 29–May 2; Paris, France. p. 559–67.

9. Li T, Sahu AK, Zaheer M, Sanjabi M, Talwalkar A, Smith V. Federated optimization in heterogeneous networks. Proc Mach Learn Syst. 2020;2:429–50. doi:10.48550/arxiv.1812.06127.
10. Karimireddy SP, Kale S, Mohri M, Reddi S, Stich S, Suresh AT. SCAFFOLD: stochastic controlled averaging for federated learning. In: Proceedings of the 37th International Conference on Machine Learning; 2020 Jul 13–18; Virtual. p. 5132–43.
11. Jhunjhunwala D, Wang S, Joshi G. Fedexp: speeding up federated averaging via extrapolation. arXiv:2301.09604. 2023.
12. Cao S, Wu H, Wu X, Ma R, Wang D, Han Z, et al. FedDA: resource-adaptive federated learning with dual-alignment aggregation optimization for heterogeneous edge devices. Future Gener Comput Syst. 2025;163:107551.
13. Wang J, Liu Q, Liang H, Joshi G, Poor HV. Tackling the objective inconsistency problem in heterogeneous federated optimization. Adv Neural Inf Process Syst. 2020;33:7611–23. doi:10.48550/arxiv.2007.07481.
14. Li Q, He B, Song D. Model-contrastive federated learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021 Jun 20–25; Nashville, TN, USA. p. 10713–22.
15. Li Z, Lin T, Shang X, Wu C. Revisiting weighted aggregation in federated learning with neural networks. In: Proceedings of the 40th International Conference on Machine Learning; 2023 Jul 23–29; Honolulu, HI, USA. p. 19767–88.
16. Geyer RC, Klein T, Nabi M. Differentially private federated learning: a client level perspective. In: Proceedings of the 7th International Conference on Learning Representations; 2017 May 6–9; New Orleans, LA, USA.
17. McMahan HB, Ramage D, Talwar K, Zhang L. Learning differentially private recurrent language models. In: Proceedings of the 6th International Conference on Learning Representations; 2018 Apr 30–May 3; Vancouver, BC, Canada.
18. Talaei M, Izadi I. Adaptive differential privacy in federated learning: a priority-based approach. arXiv:2401.02453. 2024.
19. Yang X, Huang W, Ye M. Dynamic personalized federated learning with adaptive differential privacy. Adv Neural Inf Process Syst. 2023;36:72181–92. doi:10.52202/075280-3160.
20. Yuan H, Wang H. Tailoring noise to fit: an adaptive noise optimization mechanism against gradient leakage. In: Blockchain and Web3 Technology Innovation and Application Exchange Conference. Singapore: Springer Nature Singapore; 2024. p. 25–36.
21. Errounda FZ, Liu Y. Adaptive differential privacy in vertical federated learning for mobility forecasting. Future Gener Comput Syst. 2023;149(2):531–46. doi:10.1016/j.future.2023.07.033.
22. Jiang S, Wang X, Que Y, Fed-MPS L H. Federated learning with local differential privacy using model parameter selection for resource-constrained CPS. J Syst Archit. 2024;150(1):103108. doi:10.1016/j.sysarc.2024.103108.
23. Foret P, Kleiner A, Mobahi H, Neyshabur B. Sharpness-aware minimization for efficiently improving generalization. arXiv:2010.01412. 2020.
24. Andriushchenko M, Flammarion N. Towards understanding sharpness-aware minimization. In: Proceedings of the 39th International Conference on Machine Learning; 2022 Jul 17–23; Baltimore, MD, USA. p. 639–68.
25. Wen K, Ma T, Li Z. How does sharpness-aware minimization minimize sharpness? arXiv:2211.05729. 2022.
26. Shi Y, Liu Y, Wei K, Shen L, Wang X, Tao D. Make landscape flatter in differentially private federated learning. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2023 Jun 17–24; Vancouver, BC, Canada. p. 24552–62.
27. Li H, Xu Z, Taylor G, Studer C, Goldstein T. Visualizing the loss landscape of neural nets. Adv Neural Inf Process Syst. 2018;31:6391–401. doi:10.48550/arxiv.1712.09913.
28. McMahan B, Moore E, Ramage D, Hampson S, Arcas BA. Communication-efficient learning of deep networks from decentralized data. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS) 2017; 2017 Apr 20–22; Fort Lauderdale, FL, USA. p. 1273–82.
29. Qi P, Chiaro D, Guzzo A, Ianni M, Fortino G, Piccialli F. Model aggregation techniques in federated learning: A comprehensive survey. Future Gener Comput Syst. 2024;150(6245):272–93. doi:10.1016/j.future.2023.09.008.

30. Sun Q, Li X, Zhang J, Xiong L, Liu W, Liu J, et al. ShapleyFL: robust federated learning based on shapley value. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining; 2023 Aug 6–10; Long Beach, CA, USA. p. 2096–108.
31. Ji S, Pan S, Long G, Li X, Jiang J, Huang Z. Learning private neural language modeling with attentive aggregation. In: Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN); 2019 Jul 14–19; Budapest, Hungary. p. 1–8.
32. Huang Y, Chu L, Zhou Z, Wang L, Liu J, Pei J, et al. Personalized cross-silo federated learning on Non-IID data. Proceedings of The AAAI Conference on Artificial Intelligence. 2021;35(9):7865–73. doi:10.1609/aaai.v35i9.16960.
33. Hochreiter S, Schmidhuber J. Flat minima. Neural Comput. 1997;9(1):1–42. doi:10.1162/neco.1997.9.1.1.
34. Keskar NS, Mudigere D, Nocedal J, Smelyanskiy M, Tang PTP. On large-batch training for deep learning: generalization gap and sharp minima. arXiv:1609.04836. 2016.
35. Chaudhari P, Choromanska A, Soatto S, LeCun Y, Baldassi C, Borgs C, et al. Entropy-SGD: biasing gradient descent into wide valleys. J Stat Mech Theory Exp. 2019;2019(12):124018.
36. Caldarola D, Caputo B, Ciccone M. Improving generalization in federated learning by seeking flat minima. In: European Conference on Computer Vision. Cham, Switzerland: Springer Nature; 2022. p. 654–72.
37. Kwon J, Kim J, Park H, Choi IK. ASAM: adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In: Proceedings of the 38th International Conference on Machine Learning; 2021 Jul 18–24; Virtual. p. 5905–14.
38. Qu Z, Li X, Duan R, Liu Y, Tang B, Lu Z. Generalized federated learning via sharpness aware minimization. In: Proceedings of the 39th International Conference on Machine Learning; 2022 Jul 17–23; Baltimore, MD, USA. p. 18250–80.
39. Dai R, Yang X, Sun Y, Shen L, Tian X, Wang M, et al. FedGAMMA: federated learning with global sharpness-aware minimization. IEEE Trans Neural Netw Learn Syst. 2024;35(12):17479–92.
40. Tan Q, Yang S, Ren X, Zhang Y. Rethinking layer-wise gaussian noise injection: bridging implicit objectives and privacy budget allocation. arXiv:2509.04232. 2025.
41. Fan K, Wang Z, FedANC YG. Adaptive sparse noise scheduling for federated differential privacy. In: The Fourteenth International Conference on Learning Representations; 2026 Apr 23–27. Rio de Janeiro, Brazil.
42. Shan F, Lu Y, Li S, Mao S, Li Y, Wang X. Efficient adaptive defense scheme for differential privacy in federated learning. J Inf Secur Appl. 2025;89(12):103992. doi:10.1016/j.jisa.2025.103992.
43. Shi Y, Zhang Y, Xiao Y, Niu L. Optimization strategies for client drift in federated learning: A review. Procedia Comput Sci. 2022;214(1):1168–73. doi:10.1016/j.procs.2022.11.292.
44. Montanari A, Saeed BN. Universality of empirical risk minimization. In: Proceedings of the Thirty Fifth Conference on Learning Theory; 2022 Jul 2–5; London, UK. p. 4310–2.
45. Bottou L, Curtis FE, Nocedal J. Optimization methods for large-scale machine learning. SIAM Rev. 2018;60(2):223–311. doi:10.1137/16m1080173.
46. Tian Y, Zhang Y, Zhang H. Recent advances in stochastic gradient descent in deep learning. IMathematics. 2023;11(3):682. doi:10.3390/math11030682.
47. LeCun Y. The MNIST database of handwritten digits [Internet]. 1998 [cited 2026 Jun 16]. Available from: http://yann.lecun.com/exdb/mnist/.
48. Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images [master's thesis]. Toronto, ON, Canada: University of Toronto; 2009.