FedDUAL: A Dual-Strategy with Adaptive Loss and Dynamic Aggregation for Mitigating Data Heterogeneity in Federated Learning

Anonymous authors
Paper under double-blind review

Abstract

Federated Learning (FL) marks a transformative approach to distributed model training by combining locally optimized models from various clients into a unified global model. While FL preserves data privacy by eliminating centralized storage, it encounters significant challenges such as performance degradation, slower convergence, and reduced robustness of the global model due to the heterogeneity in client data distributions. Among the various forms of data heterogeneity, label skew emerges as a particularly formidable and prevalent issue, especially in domains such as image classification. To address these challenges, we begin with comprehensive experiments to pinpoint the underlying issues in the FL training process. Based on our findings, we then introduce an innovative dual-strategy approach designed to effectively resolve these issues. First, we introduce an adaptive loss function for client-side training, meticulously crafted to preserve previously acquired knowledge while maintaining an optimal equilibrium between local optimization and global model coherence. Secondly, we develop a dynamic aggregation strategy for aggregating client models at the server. This approach adapts to each client's unique learning patterns, effectively addressing the challenges of diverse data across the network. Our comprehensive evaluation, conducted across three diverse real-world datasets, coupled with theoretical convergence guarantees, demonstrates the superior efficacy of our method compared to several established state-of-the-art approaches. The code can be found at https://anonymous.4open.science/r/FedDUAL-B04F/README.md.

1 Introduction

Federated learning (FL) has revolutionized collaborative model training by enabling multiple clients to contribute to a global model without compromising the privacy of their local data (McMahan et al., 2017). This decentralized strategy avoids the need for sending data to a central server, thus maintaining data privacy. As the digital landscape evolves, with an increasing number of distributed data sources emerging from mobile devices, healthcare institutions, and Internet of Things (IoT) networks, FL has emerged as a pivotal solution for training sophisticated deep networks across geographically dispersed and heterogeneous environments (Bonawitz et al., 2016), Sahoo et al. (2024b), (Hu et al., 2024). However, a significant practical obstacle encountered during federated training is data heterogeneity in the form of skewness in labels and quantity of the data across various clients (Kairouz et al., 2021), (Li et al., 2020). Diverse user behaviors can lead to significant heterogeneity in the local data of different clients, leading to non-independent and identically distributed (non-IID) data. This variability can introduce biases in model training, leading to unstable convergence and potentially degrading the model's performance or making it counterproductive (Li et al., 2022), (Zhao et al., 2018). While FedAvg (McMahan et al., 2017) is effective and widely used, it often falls short in accuracy and convergence with static aggregation methods. These methods combine model updates from different clients in a fixed manner, failing to adapt to heterogeneous data distributions and client drift, as discussed in (Karimireddy et al., 2020).

Previous studies have addressed the issue of client drift by implementing penalties for deviations between client and server models (Li et al., 2020), (Li et al., 2021a), employing variance reduction techniques during client updates (Karimireddy et al., 2020), (Acar et al., 2021), or utilizing novel aggregation methods on the server side (Chen et al., 2023), (Chowdhury & Halder, 2024).

1.1 Motivation

Prior studies by Yashwanth et al. (2024), Hu et al. (2024) have demonstrated that in non-IID scenarios, federated models tend to converge to 'sharp minima', resulting in significant performance degradation and compromised generalizability. In this study, we investigate the root causes of this phenomenon and propose a novel solution to mitigate its effects. Our study begins with a detailed analysis of loss landscapes for FedAvg-trained models across IID and non-IID data distributions. Figure 1 visually depicts the loss landscapes of two models on the FMNIST dataset with systematic parameter perturbations. The model trained on IID data exhibits a notably smoother and wider valley in its loss landscape, suggesting greater robustness and better generalization. In contrast, the model trained on non-IID manifests sharper peaks and narrower valleys, indicating higher sensitivity to parameter variations and potential overfitting. These visualizations offer strong evidence that in the presence of non-IID data, the FedAvg algorithm achieves suboptimal generalization. Motivated by this observation, we investigate the underlying mechanisms by analyzing gradient norms to identify which parts of the neural network are most affected by data heterogeneity. Our findings, presented in Fig. 2, reveal a notable pattern: in non-IID scenarios, the gradient norms of the final layers, including the classification layer, exhibit significant amplification compared to their IID counterparts. Such amplification leads to model instability, impedes convergence, and ultimately compromises the generalizability of the federated model. Our investigation suggests that effective federated training in non-IID environments necessitates targeted adjustments during server-side aggregation, particularly for these highly affected layers, to achieve performance comparable to IID settings.

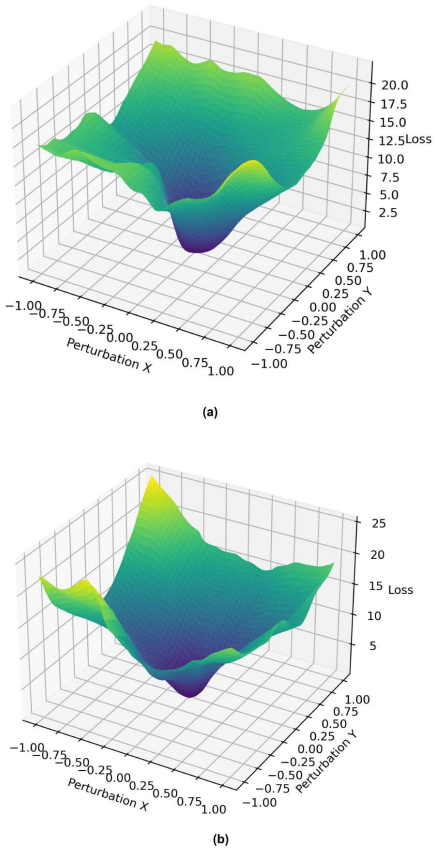

Figure 1: Visualization of the loss surface for the global model trained on the FMNIST dataset using the FedAvg algorithm: (a) depicts the loss landscape when trained on IID data, while (b) illustrates the landscape for non-IID data distribution.

This prompts one critical question: *Can static aggregation methods effectively address severe non-IID data distributions across clients while maintaining higher convergence, performance, and generalizability in federated models?* The answer is decidedly negative. Static aggregation methods inherently struggle with the dynamic heterogeneity present in federated networks, where adjusting parameters based on client distributions and performance in each communication round is crucial. Although incorporating predetermined parameters into the aggregation process may provide some partial mitigation, these methods fail to address the complex challenges posed by non-IID data distributions. A more dynamic and nuanced approach is necessary to effectively manage these multifaceted issues. To address this challenge, we apply dynamic aggregation to the model's final layers, where gradient norms fluctuate significantly in non-IID scenarios, while using traditional aggregation (FedAvg) for the lower layers. For dynamic aggregation, we leverage the concept of Wasserstein Barycenter (Agueh & Carlier, 2011), derived from optimal transport theory, to integrate client-specific learning behaviors in these affected layers. By minimizing discrepancies from non-IID data, the Wasserstein Barycenter helps to align gradients from diverse clients, offering precise

model updates. This approach ensures fair aggregation, adapts to data heterogeneity, reduces bias, and enhances robustness, ultimately leading to more stable model convergence and improved generalization.

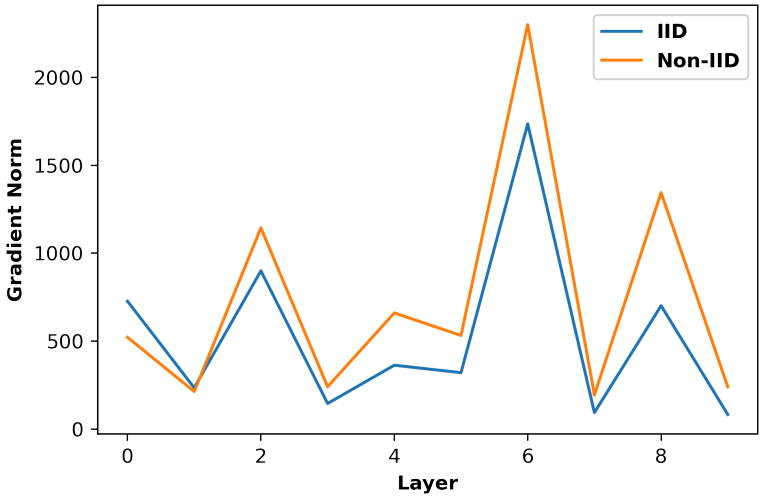

Figure 2: Comparison of gradient norms between models trained on IID and non-IID datasets using the FedAvg algorithm.

In addition to the server-side dynamic aggregation, we introduce an adaptive loss function for local training on the client side. This function allows clients to effectively explore the minima on their local datasets while preventing overfitting, thereby enhancing local optimization. Simultaneously, it preserves the global knowledge of the federated model, ensuring that the benefits from all participating clients are integrated. By incorporating a regularization parameter, β, the local loss function dynamically balances the trade-offs between local and global objectives. The contributions of this paper are as follows:

- We introduce FedDUAL, an innovative dual-strategy approach designed to effectively develop a robust and generalized federated model in highly heterogeneous data environments.
- We introduce an adaptive loss function for client-side training to balance the trade-offs between local and global objectives.
- Instead of straightforward server-side averaging, we propose a dynamic aggregation technique that uses Wasserstein Barycenter to reduce the effects of non-IID data by integrating the learning behaviors of participating clients.
- We conducted extensive experiments on three real-world datasets, demonstrating significant performance improvements over state-of-the-art methods and offering theoretical convergence guarantees for both convex and non-convex scenarios.

2 Related Work

The landscape of FL research has been significantly shaped by efforts to address data heterogeneity challenges, yielding a diverse array of innovative solutions. These approaches can be divided into three primary categories: (1) client drift mitigation strategies, which refine local client objectives to foster better alignment with the global model (Li et al., 2021a), (Karimireddy et al., 2020), (Acar et al., 2021), (Luo et al., 2021), (Li et al., 2023) (2) aggregation scheme optimization, aimed at enhancing server-side fusion of model updates (Hsu et al., 2019), (Lin et al., 2020), (Wang et al., 2020b), (Wang et al., 2020a) and (3) personalized FL, which tailors models to individual clients (Fallah et al., 2020), (Sattler et al., 2020), (Bui et al., 2019).

Our research primarily focuses on two interconnected aspects of FL: mitigating client drift and optimizing server-side aggregation, and we will discuss the same in the literature review.

McMahan et al. (2017) introduced FL as an extension of local Stochastic Gradient Descent (SGD) (Stich, 2019), enabling increased local gradient updates on client devices before server synchronization and significantly reducing communication costs in identically distributed data settings. However, the method faces considerable obstacles when dealing with non-IID scenarios. Since then, various methods have emerged to address the challenge of data heterogeneity in FL (Li et al., 2019), (Yang et al., 2021), (Lin et al., 2018), (Hsu et al., 2019). FedProx (Li et al., 2020) incorporates a proximal regularization term to the optimization function to reduce model drift and addresses client stragglers. However, this term can also lead to local updates being biased towards the previous global model, which may result in misalignment between local and global optima. Building on previous work, Acar et al. (2021) introduced a dynamic regularization term to align local updates more closely with global model parameters, effectively reducing client drift caused by local model overfitting. Sun et al. (2023) further advanced the field with a momentum-based algorithm that accelerates convergence by combining global gradient descent with a locally adaptive optimizer. Similarly, several studies use variance reduction techniques, such as SCAFFOLD (Karimireddy et al., 2020). However, this approach often results in higher communication costs due to the transmission of additional control variates (Halgamuge et al., 2009). FedPVR (Li et al., 2023) addresses these limitations by reassessing FedAvg's performance on deep neural networks, uncovering substantial diversity in the final classification layers. By proposing a targeted variance reduction strategy focused solely on these final layers, FedPVR outperforms several benchmarks. MOON (Li et al., 2021a) introduces an innovative model-contrastive framework leveraging a contrastive loss to align local client representations with the global model, effectively mitigating client drift, and enhancing convergence, particularly in challenging non-IID environments. Luo et al. (2021) introduced CCVR (Classifier Calibration and Variance Reduction), which employs a classifier regularization and calibration method to enhance federated learning performance. CCVR's approach involves fine-tuning the classifier using virtual representations sampled from an approximated Gaussian mixture model. Shi et al. (2023) introduced a novel differentially private federated learning (DPFL) algorithm that integrates the Sharpness-Aware Minimization (SAM) optimizer to enhance stability and robustness against weight perturbations. By generating flatter loss landscapes and reducing the impact of differential privacy (DP) noise, it mitigates performance degradation and achieves state-of-the-art results, supported by theoretical analysis and rigorous privacy guarantees. Fanì et al. (2024) proposed FED3R, leveraging Ridge Regression on pre-trained features to tackle non-IID data challenges, effectively mitigating client drift, enhancing convergence, and optimizing efficiency in cross-device settings.

Another line of research targets optimizing server-side aggregation in FL. For instance, Hsu et al. (2019) investigated the impact of non-IID data on visual classification by creating datasets with diverse distributions and found that increased data heterogeneity negatively affected performance, leading them to propose server momentum as a potential solution. FedNova (Wang et al., 2020b) addressed the problem of objective inconsistency due to client heterogeneity in federated optimization by introducing a normalized averaging technique, which resolves this inconsistency and ensures rapid error convergence. Addressing the limitations of traditional parameter averaging methods, Lin et al. (2020) introduced ensemble distillation for model fusion. This approach allows for the flexible aggregation of heterogeneous client models by training a central classifier on unlabeled data, using the outputs from the client models as guidance. FedMRL (Sahoo et al., 2024a) introduced a novel framework by using a loss function that promotes fairness among clients and employed a multi-agent reinforcement learning for personalized proximal terms , and a self-organizing map to dynamically adjust server-side weights during aggregation.

3 Methods and Materials

We consider a practical FL scenario with non-IID data distribution among K independent clients, each with local training data $D_k(x, y)$, where (x, y) denoting the data points. We initialize the global model weights θ_r^g and share it to the participating clients. The clients download the weighs from the server and train it using their local dataset $D_k(x, y)$. The updated model parameters θ_k^r from each client k for r^{th} communication round are uploaded to the server to aggregate into a global model θ_r^g. Our objective is to

develop a robust global model by collaboratively training local models across clients, even under varying heterogeneous conditions. To formalize, we define the optimal global model θ^* as follows:

$$\theta^* = \min_{\theta} F(\theta) = \frac{1}{K} \sum_{k=1}^{K} f_k(\theta) \tag{1}$$

where $f_k(\theta)$ is defined in Eq. 2.

$$f_k(\theta) = E_{(x,y)\sim D_k}[\ell(f_\theta(x), y)] \tag{2}$$

where θ represents the global model parameters, $f_\theta(x)$ is the model's prediction, and ℓ is the loss function.

3.0.1 Client Side Update.

At the beginning of each round t, the server randomly selects a subset $S_t \subset K$ of clients to participate in the federated training process and subsequently shares the current global model θ_r^g to these participating clients. Each client updates its local model by initializing with the global model parameters ($\theta_k^r = \theta_g^r$) and then updates its local model by minimizing the local objective function. For local training, we have developed an adaptive objective function that balances local loss with the divergence between local and global models. The extent of this divergence is quantified using the Kullback-Leibler (KL) divergence (Csiszár, 1975), which effectively compares the probability distributions of the local model weights $p^k(w)$ with the global model weights $q(w)$. The KL divergence is mathematically defined in Eq. 4. To obtain the probability distributions of the local and global model weights, we first flatten the weights and then apply the softmax function. This process yields the desired probability distributions (p), as specified in Eq. 3.

$$\text{p} = \frac{\exp(\text{flatten weights})}{\sum \exp(\text{flatten weights})} \tag{3}$$

$$D_{\text{KL}}(p^k \| q) = \sum_i p_i^k(w) \log \left(\frac{p_i^k(w)}{q_i(w)} \right) \tag{4}$$

where p_i^k and q_i are the probabilities associated with the ith component of the weight vectors. The local model must excel on local data while maintaining alignment with the global model to enhance overall generalization. This balance between minimizing local loss and aligning with the global model is defined as local adaptive function $\mathcal{L}_{adaptive}^k$ in Eq. 5.

$$\mathcal{L}_{adaptive}^k = (1-\beta) * \mathcal{L}_{\text{local}}^k + \beta * D_{\text{KL}}(p^k \| q) \tag{5}$$

where $\mathcal{L}_{\text{local}}^k$ is cross-entropy loss for k^{th} client and β is a regularization parameter and should be adaptive to account for the performance discrepancy between the local and global models. When the local model substantially outperforms the global model, β should increase to enforce greater alignment. Conversely, if the models perform similarly, β should decrease, allowing the local model to focus more on local optimization. The definition of β is given in Eq. 6.

$$\beta = \sigma(\mathcal{A}_{\text{local}}^k - \mathcal{A}_{\text{global}}^k) \tag{6}$$

where σ is the sigmoid function, $\mathcal{A}_{\text{local}}^k$ represents the local model accuracy, and $\mathcal{A}_{\text{global}}^k$ is the global model accuracy for client k. We calculated the global model's accuracy $\mathcal{A}_{\text{global}}^k$ for client k by evaluating it on the training data of client k prior to performing local updates in the current round. Incorporating the adaptive parameter β in Eq. 5, the adaptive loss function for client k is represented in Eq. 7.

$$\begin{aligned}\mathcal{L}^k_{\text{adaptive}} = (1 - (\sigma(\mathcal{A}^k_{\text{local}} - \mathcal{A}^k_{\text{global}})) * \mathcal{L}^k_{\text{local}} \\ + \sigma(\mathcal{A}^k_{\text{local}} - \mathcal{A}_{\text{global}}) * D_{\text{KL}}(p^k \| q)\end{aligned} \tag{7}$$

After defining the adaptive loss function for each client, we optimize the local model parameters using stochastic gradient descent (SGD). The gradient update for the local model weights w_k based on the adaptive loss function is given in Eq. 8.

$$w^{t+1}_k = w^t_k - \eta \nabla_w \mathcal{L}^k_{\text{adaptive}}(w^t_k) \tag{8}$$

where η is the local learning rate. Expanding the gradient term $\nabla_w \mathcal{L}^k_{\text{adaptive}}(w^t_k)$, we obtain Eq. 9.

$$\begin{aligned}\nabla_w \mathcal{L}^k_{\text{adaptive}}(w^t_k) = (1 - (\sigma(\mathcal{A}^k_{\text{local}} - \mathcal{A}^k_{\text{global}}))\nabla_w \mathcal{L}^k_{\text{local}}(w^t_k) + \\ \sigma(\mathcal{A}^k_{\text{local}} - \mathcal{A}_{\text{global}})\nabla_w D_{\text{KL}}(p^k \| q)\end{aligned} \tag{9}$$

The KL divergence term, $\sigma(\mathcal{A}^k_{\text{local}} - \mathcal{A}^k_{\text{global}})\nabla_w D_{\text{KL}}(p^k \| q)$ in Eq. 9, acts as a regularizer to keep the local model gradients aligned with the global model gradients, thereby preserving model coherence despite non-IID data.

By dynamically adjusting the regularization parameter β based on the performance difference between local and global models, the adaptive loss function enhances the alignment of local models with the global model, thereby improving generalization and performance in non-IID scenarios. When local performance is less compared to global, the regularization term β amplifies the focus on local optimization (first term in Eq. 5), enabling better-performing clients to explore local optima more effectively. Conversely, if the global model performs better, this term shifts the emphasis towards aligning with the global model (second term in Eq. 5), thereby supporting clients that are struggling by incorporating global knowledge.

3.0.2 Server Side Update.

After obtaining the weights from the participating clients at round t, the server calculates the Wasserstein Barycenter to effectively aggregate the weights of the last layers of the client models. Computing exact Wasserstein Barycenter can be computationally expensive, so we have approximated it using the Sinkhorn-Knopp (Knight, 2008) algorithm for efficient computation. We consider the local model weights as distributions and assign equal importance to each client in the computation of the Wasserstein Barycenter ($\bar{\mu}$). This barycenter represents the distribution that minimizes the sum of Wasserstein distances to the individual client gradient distributions, as formally defined in Eq. 10.

$$\hat{\mu} = \arg\min_{\nu} \sum_{k=1}^{K} \lambda_k W(\mu_k, \nu) \tag{10}$$

where λ_k are weights corresponding to the importance or reliability of the client k. The Wasserstein distance $W(\mu_k, \mu_j)$ between two gradient distributions μ_k and μ_j of clients j and k is defined in Eq. 11.

$$W(\mu_k, \mu_j) = \left(\inf_{\gamma \in \Gamma(\mu_k, \mu_j)} \int_{\mathcal{X} \times \mathcal{X}} d(x, y)^p \, d\gamma(x, y) \right)^{1/p} \tag{11}$$

where $\Gamma(\mu_k, \mu_j)$ denotes the set of all couplings (or joint distributions) γ on $\mathcal{X} \times \mathcal{X}$ with marginals μ_k and μ_j respectively, and $d(x, y)$ is the distance between points x and y in the metric space $\mathcal{X}$. After that, we use Sinkhorn-Knopp algorithm to calculate the Wasserstein Barycenter.

This barycenter is computed iteratively, starting by calculating a scaling factor γ using Eq. 12, followed by Eq. 13.

$$\gamma = \exp\left(-\frac{\mathrm{W}(\bar{p}, p_i)}{\epsilon} \right) \tag{12}$$

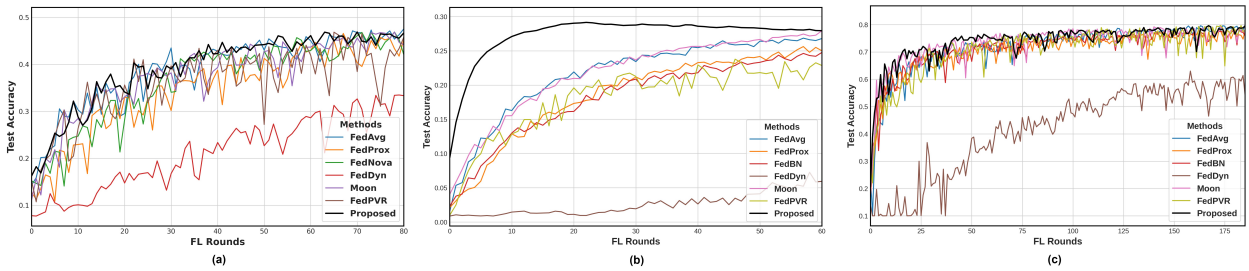

Figure 3: Learning curves comparing the proposed method with baselines across various datasets: (a) CIFAR-10, (b) CIFAR-100, and (c) FMNIST.

$$\bar{p}_{\text{new}} = \frac{\sum_i \lambda_i \gamma p_i}{\sum_i \lambda_i \gamma_i} \tag{13}$$

where $\bar{p}$ is the current estimate of the barycenter, p_i refers to the ith client's gradient distribution, ϵ is a small positive constant, and the iterations continue until convergence. After few iterations, we get the Wasserstein barycenter that is used to update the global model weights. We update the the global model weights for the last layers by substracting them from the calculated Wasserstein barycenter for effectively aggregating the updates from the last layers. The algorithm of proposed method FedDUAL is given in the Algorithm 1. The proof of the convergence for both convex and non-convex settings for the proposed method can be found in the Section A of the Appendix.

Algorithm 1 FedDUAL

Input: Number of clients K, Number of communication rounds T, and Global model $\mathcal{G}$.
Output: Trained global model $\mathcal{G}^*$.
Define a mask $e \in \{0,1\}^d$, where $e_j = 1$ for the last few layers and 0 for the rest layers.
Let $S_{naive} = \{j : e_j = 0\}$ and $S_{dynamic} = \{j : e_j = 1\}$.
Initialize global model weights θ^g
for $t = 1$ to T **do**
  Sample a subset of clients $\mathcal{S}_t \subseteq \{1, \ldots, K\}$
  Initialize lists: local model weights $\mathcal{W} \leftarrow []$, gradients $\Delta \leftarrow []$
  for each client $k \in \mathcal{S}_t$ **do**
    Initialize local model $\mathcal{M}_k$ with global weights θ^g.
    Train $\mathcal{M}_k$ on local dataset $\mathcal{D}_k$ using adaptive loss function defined in Eq. 7.
    $\mathcal{W} \leftarrow \mathcal{W} \cup \{\theta_k\}$ ▷ Store local model weights θ_k
    Compute gradients ∇_k for $\mathcal{M}_k$
    $\Delta \leftarrow \Delta \cup \{\nabla_k\}$ ▷ Store gradients ∇_k
  end for
  for $j \in \{1, \ldots, d\}$ **do**
    if $e_j = 1$ **then** ▷ Layer belongs to $S_{dynamic}$
      Extract last layers' gradients $\{\nabla_k[j]\}$ from Δ
      Compute Wasserstein Barycenter of last layer j gradients $\bar{\nabla}_j$
      Update global model's last layer j weights $\theta^g[j] \leftarrow \theta^g[j] - \bar{\nabla}_j$
    else ▷ Layer belongs to S_{naive}
      Perform Federated Averaging for layer j:
      $\theta^g[j] \leftarrow \frac{1}{|\mathcal{S}_t|} \sum_{k \in \mathcal{S}_t} \theta_k[j]$
    end if
  end for
end for
$\mathcal{G}^* \leftarrow \theta^g$ ▷ Final trained global model

Table 1: Top-1 accuracy (%) on CIFAR10, CIFAR100, and FMNIST datasets. The values in bold represent the highest accuracy achieved. '*' denotes algorithms that failed to achieve convergence.

	CIFAR10	CIFAR100	FMNIST
FedAvg	46.68 ± 0.25	26.88 ± 0.18	81.70 ± 0.20
FedProx	47.58 ± 0.30	26.89 ± 0.22	80.54 ± 0.28
FedNova	48.44 ± 0.35	*	*
FedBN	*	26.88 ± 0.19	81.36 ± 0.23
FedDyn	43.97 ± 0.40	18.27 ± 0.32	71.86 ± 0.45
MOON	46.57 ± 0.28	28.50 ± 0.25	80.09 ± 0.27
SCAFFOLD	*	*	*
FedPVR	42.26 ± 0.42	23.78 ± 0.31	80.32 ± 0.33
Proposed	**48.70 ± 0.20**	**29.15 ± 0.24**	**81.99 ± 0.21**

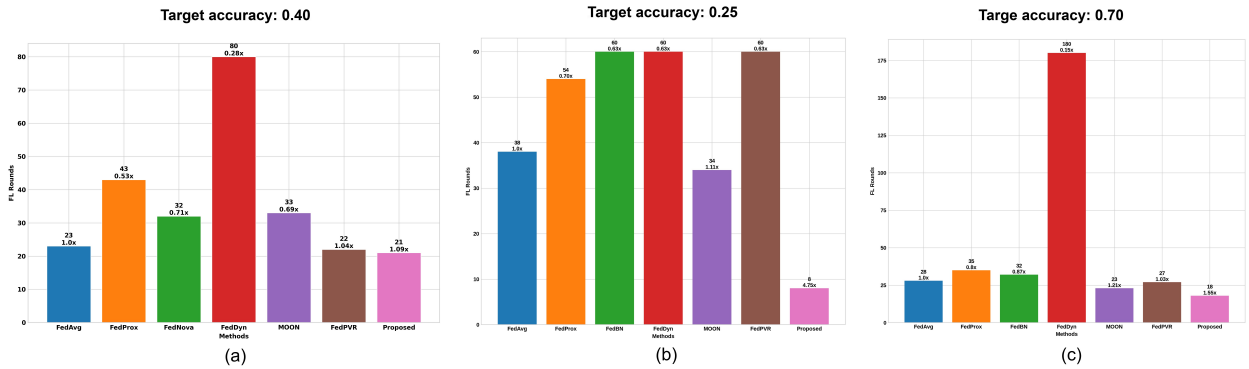

Figure 4: Number of FL rounds required to reach the target accuracy for the proposed method and other baselines on different datasets: (a) CIFAR-10, (b) CIFAR-100, and (c) FMNIST.

4 Convergence Proof

Notation

Let $\mathcal{W}$ denote the parameter space, $f_i : \mathcal{W} \to \mathbb{R}$ be the loss function for the ith client, and $F : \mathcal{W} \to \mathbb{R}$ be the global objective function.

Assumptions

1. Lipschitz Smoothness: For all i, f_i is L-smooth such that it follows inequality 14:

$$\|\nabla f_i(w) - \nabla f_i(w')\| \leq L\|w - w'\|, \forall w, w' \in \mathcal{W}. \tag{14}$$

2. Bounded Variance: The variance of stochastic gradients is bounded by inequality 15:

$$\mathbb{E}\|\nabla f_i(w, \xi) - \nabla f_i(w)\|^2 \leq \sigma^2, \forall w \in \mathcal{W}. \tag{15}$$

where ξ represents the random sampling of data.

Theorem 1: If the above assumptions hold, then our proposed FedDUAL achieves below rates in convex and non-convex settings.
Convex setting:

$$F(\bar{w}_T) - F(w^*) \leq \frac{\|w_0 - w^*\|^2}{2\eta T\sqrt{T}} + \frac{F(w_0) - F(w^*)}{2T} + \frac{L\eta G^2\sqrt{T}}{2T}.$$

Non-Convex setting:

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(w_t)\|^2 \leq \sqrt{\frac{2LG^2(F(w_0)-F(w^*))}{T}}$$

where G is the bound on the true gradient norm, L is the Lipschitz constant, and T is the number of communication rounds. The complete proof can be found in Section A of the Appendix and the discussion of obtained rates and comparison with existing methods can be found in Section B of the Appendix.

5 Experimental Results

5.1 Experimental Setup

To assess the effectiveness of the proposed FedDUAL approach, we conducted extensive experiments using three widely recognized classification benchmarks: CIFAR10 (Krizhevsky et al., 2009), CIFAR100 (Krizhevsky, 2009), and FMNIST (Xiao et al., 2017). To simulate real-world non-IID data distributions, we employed a client-wise partitioning strategy based on the Dirichlet distribution (Hsu et al., 2019). This distribution is governed by a concentration parameter α, which controls the degree of data heterogeneity among clients. Lower α values result in more skewed data distributions, closely mimicking uneven data partitions. In all experiments, we set $\alpha = 0.01$ to simulate severe data heterogeneity, closely approximating real-world conditions. Throughout the communication rounds, each client retains a fixed local data partition. To evaluate the global model's classification performance, we use a separate test dataset maintained at the server, which remains unseen during training. For our experiments, we used LeNet (LeCun et al., 1998) for FMNIST dataset and a pre-trained VGG16 (Simonyan & Zisserman, 2015) for CIFAR-10 and CIFAR-100 dataset, following the methodology outlined in (Hu et al., 2024). We applied the proposed dynamic aggregation mechanism only to the last two layers of these models. Our setup involved 100 clients, with 10% randomly sampled per communication round, and a batch size of 32. Each client performed three local epochs of model updates. We have computed each result three times with different seed values and reported the mean value with standard deviation. To determine the optimal client learning rate for each experiment, we conducted a grid search over $0.05, 0.01, 0.2, 0.3$. For the baseline FedProx, we tested proximal values of 0.001, 0.1, 0.4, 0.7 to find the optimal setting, and for FedNova, we evaluated proximal SGD values from 0.001, 0.003, 0.05, 0.1, following the recommendations in Li et al. (2024). Across all experiments, we used the Adam optimizer for consistency. We have run each algorithm three times and reported the average outcome. The experimental setup utilized an NVIDIA Quadro RTX 4000 GPU boasting 40GB of memory. The implementation was crafted using Python [1], leveraging the TensorFlow framework [2] utilizing Windows 11.

5.2 Comparison with the State-of-the-art Methods

5.2.1 Baseline.

We evaluate the proposed FedDUAL method against eight notable state-of-the-art (SOTA) FL baselines, including FedAvg (McMahan et al., 2017), FedProx (Li et al., 2020), FedNova (Wang et al., 2020b), SCAFFOLD (Karimireddy et al., 2020), FedBN (Li et al., 2021b), FedDyn (Acar et al., 2021), MOON (Li et al., 2021a) and FedPVR (Li et al., 2023).

5.2.2 Comparison of Accuracy.

The results, detailed in Table 1, reveal that many recent FL methods often fall short compared to the standard FedAvg baseline. In contrast, our proposed method consistently achieves SOTA performance, surpassing FedAvg along with other baselines across all evaluated scenarios. Furthermore, our approach exhibits remarkable adaptability across diverse datasets. Unlike some algorithms that excel on specific

[1] https://www.python.org/
[2] https://www.tensorflow.org/

datasets but falter on others, the proposed FedDUAL consistently outperforms baselines across a wide range of data environments. This improvement suggests that our method addresses fundamental challenges in FL, potentially offering a more generalizable solution to the issues posed by data heterogeneity in federated settings. We also observed that FedNova, FedBN, and Scaffold did not perform effectively in our experimental setup.

5.2.3 Comparison of Convergence.

We also provide a comparative review of learning process, as shown in Fig. 3, where we present the learning curves of our method with the baselines. The results demonstrate a consistent pattern: our approach exhibits accelerated learning and achieves superior accuracy across all tested scenarios. While the communication rounds vary by dataset, we observe that model performance reaches a saturation point by the conclusion of these rounds. Notably, our method not only attains a more robust final model but also displays markedly faster convergence across all datasets examined. This effectiveness is further highlighted in Fig. 4, where it consistently reaches target accuracy with far fewer communication rounds compared to baseline approaches.

5.3 Validation of the Motivation

To substantiate our claim that the proposed method yields models in flatter loss landscapes compared to FedAvg, we conducted a comparative analysis. Using VGG-16 models trained on the FMNIST dataset under non-IID conditions ($\alpha = 0.01$), we visualized their respective loss landscapes following the approach outlined in Li et al. (2018). Figure 6 in the Appendix depicts these landscapes, with each model centrally located within its respective terrain. The visualization reveals that our proposed method situates the model in a notably flatter region compared to FedAvg. This finding supports our assertion that our approach guides federated training towards more stable and generalizable solutions, characterized by flatter loss landscapes. The performance improvement of the proposed models stems from two key innovations: a Wasserstein Barycenter-based aggregation for final layer gradients, mitigating client drift in heterogeneous data environments, and an adaptive loss function balancing local optimization with global consistency during client training. This synergistic approach preserves global knowledge while promoting client-specific optimization, addressing fundamental FL challenges.

Table 2: Ablation study results for the proposed FedDUAL method.

Adaptive Loss	Dynamic Aggregation	Accuracy (%)
✓	✗	80.70 ± 0.22
✗	✓	80.91 ± 0.25
✓	✓	**81.99 ± 0.18**

6 Ablation Study

In our ablation study, we performed all experiments on the FMNIST dataset with $\alpha = 0.01$. The study comprised four types of experiments: (1) performance analysis of the individual modules, (2) assessment of the impact of dynamic aggregation across different neural network layers, (3) hyperparameter analysis, and (4) evaluation of various levels of data heterogeneity. The results of the first and second experiments are detailed in the following subsections, while the third and fourth experiments are presented in Sections C of the Appendix.

6.0.1 Performance Analysis of Individual Modules.

To demonstrate the significance of our proposed adaptive loss function and dynamic aggregation technique, we conducted three distinct experiments, with results summarized in Table 2. The first experiment utilized only the adaptive loss function, paired with simple server-side aggregation. The second implemented our dynamic aggregation technique at the server, while retaining the conventional cross-entropy loss function locally. Finally, the third experiment combined both proposed methods: the adaptive loss function and

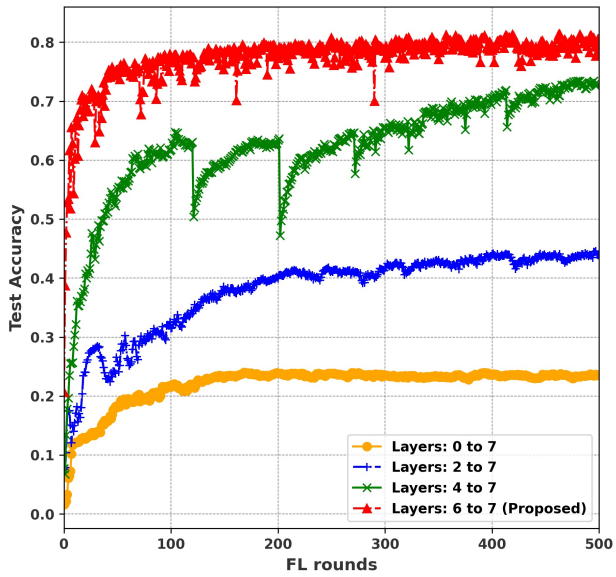

Figure 5: Illustration of the Dynamic aggregation method applied across various layers of the neural network.

dynamic aggregation technique. As evidenced by Table 2, the integration of both proposed approaches in the third experiment yielded the highest accuracy, highlighting the impact of our dual strategy on model performance. The learning curves for these experiments are illustrated in Fig. 7 of the Appendix.

6.0.2 Impact of Dynamic Aggregation on Different Network Layers.

To substantiate our decision to apply dynamic aggregation technique selectively to last layers, we examined its impact across various layers of the neural network. Our earlier findings highlighted that data heterogeneity primarily affects last layers of the network. Figure 5 illustrates that random utilization of the dynamic aggregation to all layers diminishes performance. Conversely, targeted implementation on layers proximal to the classifier yielded optimal accuracy and convergence. These outcomes validate our hypothesis and demonstrate the method's efficacy in mitigating heterogeneity-induced issues. By focusing our dynamic aggregation technique on the most susceptible layers, we directly address the core challenge of data heterogeneity in federated training, resulting in enhanced model performance and faster convergence.

7 Limitation

The proposed FedDUAL framework tackles heterogeneous data challenges in federated training, outperforming baselines through an adaptive client-side loss function and dynamic server-side aggregation via the Wasserstein Barycenter. While the proposed method significantly enhance performance and accelerate convergence, they introduce additional computational overhead at the server due to the iterative calculation of the Wasserstein Barycenter. Conversely, the adaptive loss function on the client side is computationally efficient, incurring minimal cost. Future research will focus on reducing the computational burden of the Wasserstein Barycenter calculation by developing more efficient algorithms, aiming to maintain or even improve the performance of the proposed framework.

8 Conclusion

This research presents a novel approach to address the challenges posed by data heterogeneity among clients in the federated approach. We systematically analyze the factors contributing to federated model performance degradation under severe data heterogeneity and propose an architecture incorporating dual-strategy innovations. First, we implement an adaptive loss function for client-side training. Second, we create a

dynamic aggregation strategy for server side aggregation, tailored to client-specific learning behaviors. The proposed FedDUAL effectively overcomes the challenges of heterogeneous data, outperforming eight SOTA baselines. It demonstrates faster convergence and consistently improved performance, making it an excellent solution for large-scale FL applications in real-world scenarios. Our approach's flexibility paves the way for research into hybrid federated learning models that adapt to changing client environments and data. Future studies will focus on integrating personalized learning paths to enhance model adaptability and efficiency across various datasets.

References

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas Navarro, Matthew Mattina, Paul N. Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization, 2021.

Martial Agueh and Guillaume Carlier. Barycenters in the wasserstein space. *SIAM Journal on Mathematical Analysis*, 43(2):904–924, 2011.

Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for federated learning on user-held data. *arXiv preprint arXiv:1611.04482*, 2016.

Duc Bui, Kshitiz Malik, Jack Goetz, Honglei Liu, Seungwhan Moon, Anuj Kumar, and Kang G Shin. Federated user representation learning. *arXiv preprint arXiv:1909.12535*, 2019.

Dengsheng Chen, Jie Hu, Vince Junkai Tan, Xiaoming Wei, and Enhua Wu. Elastic aggregation for federated optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12187–12197, June 2023.

Sujit Chowdhury and Raju Halder. Fedsat: A statistical aggregation approach for class imbalaced clients in federated learning, 2024. URL https://arxiv.org/abs/2407.03862.

Imre Csiszár. I-divergence geometry of probability distributions and minimization problems. *The annals of probability*, pp. 146–158, 1975.

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*, 2020.

Eros Fanì, Raffaello Camoriano, Barbara Caputo, and Marco Ciccone. Accelerating heterogeneous federated learning with closed-form classifiers. *Proceedings of the International Conference on Machine Learning*, 2024.

Malka Nisha Halgamuge, Moshe Zukerman, Kotagiri Ramamohanarao, and Hai L Vu. An estimation of sensor energy consumption. *Progress In Electromagnetics Research B*, (12):259–295, 2009.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

Ming Hu, Yue Cao, Anran Li, Zhiming Li, Chengwei Liu, Tianlin Li, Mingsong Chen, and Yang Liu. Fedmut: Generalized federated learning via stochastic mutation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 12528–12537, 2024.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pp. 5132–5143. PMLR, 2020.

Philip A Knight. The sinkhorn–knopp algorithm: convergence and applications. *SIAM Journal on Matrix Analysis and Applications*, 30(1):261–275, 2008.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

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

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Bo Li, Mikkel N Schmidt, Tommy S Alstrøm, and Sebastian U Stich. On the effectiveness of partial variance reduction in federated learning with heterogeneous data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3964–3973, 2023.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31, 2018.

Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10713–10722, 2021a.

Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-iid data silos: An experimental study. In *2022 IEEE 38th international conference on data engineering (ICDE)*, pp. 965–978. IEEE, 2022.

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

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019.

Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fedbn: Federated learning on non-iid features via local batch normalization. *arXiv preprint arXiv:2102.07623*, 2021b.

Xingyu Li, Zhe Qu, Bo Tang, and Zhuo Lu. Fedlga: Toward system-heterogeneity of federated learning via local gradient approximation. *IEEE Transactions on Cybernetics*, 54(1):401–414, January 2024. ISSN 2168-2275. doi: 10.1109/tcyb.2023.3247365. URL http://dx.doi.org/10.1109/TCYB.2023.3247365.

Tao Lin, Sebastian U Stich, Kumar Kshitij Patel, and Martin Jaggi. Don't use large mini-batches, use local sgd. *arXiv preprint arXiv:1808.07217*, 2018.

Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, 33:2351–2363, 2020.

Mi Luo, Fei Chen, Dapeng Hu, Yifan Zhang, Jian Liang, and Jiashi Feng. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. *Advances in Neural Information Processing Systems*, 34:5972–5984, 2021.

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

Pranab Sahoo, Ashutosh Tripathi, Sriparna Saha, and Samrat Mondal. Fedmrl: Data heterogeneity aware federated multi-agent deep reinforcement learning for medical imaging, 2024a. URL https://arxiv.org/abs/2407.05800.

Pranab Sahoo, Ashutosh Tripathi, Sriparna Saha, Samrat Mondal, Jyoti Prakash Singh, and Bhisham Sharma. Adafedprox: A heterogeneity-aware federated deep reinforcement learning for medical image classification. *IEEE Transactions on Consumer Electronics*, 2024b.

Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE transactions on neural networks and learning systems*, 32(8):3710–3722, 2020.

Yifan Shi, Yingqi Liu, Kang Wei, Li Shen, Xueqian Wang, and Dacheng Tao. Make landscape flatter in differentially private federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24552–24562, 2023.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. URL https://arxiv.org/abs/1409.1556.

Sebastian U. Stich. Local sgd converges fast and communicates little, 2019. URL https://arxiv.org/abs/1805.09767.

Yan Sun, Li Shen, Hao Sun, Liang Ding, and Dacheng Tao. Efficient federated learning via local adaptive amended optimizer with linear speedup. *arXiv preprint arXiv:2308.00522*, 2023.

Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. In *International Conference on Learning Representations*, 2020a. URL https://openreview.net/forum?id=BkluqlSFDS.

Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33: 7611–7623, 2020b.

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

Haibo Yang, Minghong Fang, and Jia Liu. Achieving linear speedup with partial worker participation in non-iid federated learning. *arXiv preprint arXiv:2101.11203*, 2021.

M. Yashwanth, Gaurav Kumar Nayak, Harsh Rangwani, Arya Singh, R. Venkatesh Babu, and Anirban Chakraborty. Minimizing layerwise activation norm improves generalization in federated learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2287–2296, January 2024.

Hao Yu, Sen Yang, and Shenghuo Zhu. Parallel restarted sgd with faster convergence and less communication: demystifying why model averaging works for deep learning. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press, 2019. ISBN 978-1-57735-809-1. doi: 10.1609/aaai.v33i01.33015693. URL https://doi.org/10.1609/aaai.v33i01.33015693.

Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.

Appendix

A Convergence Proof

A.1 Convergence Analysis for Convex Setting

For the convex setting, we first start with the local training at client side using gradient descent. The update rule for gradient descent is presented in Eq. 16.

$$w_{t+1} = w_t - \eta_t \nabla F(w_t), \tag{16}$$

where η_t is the learning rate. Using the smoothness property of $F(w)$, we expand $F(w_{t+1})$ as follows:

$$F(w_{t+1}) \leq F(w_t) + \langle \nabla F(w_t), w_{t+1} - w_t \rangle + \frac{L}{2}\|w_{t+1} - w_t\|^2. \tag{17}$$

Substituting $w_{t+1} - w_t = -\eta_t \nabla F(w_t)$ in inequality 17, we get inequality 18:

$$F(w_{t+1}) \leq F(w_t) - \eta_t \|\nabla F(w_t)\|^2 + \frac{L\eta_t^2}{2} \|\nabla F(w_t)\|^2. \tag{18}$$

Taking expectations of inequality 18 we get inequality 19:

$$\mathbb{E}[F(w_{t+1})] \leq \mathbb{E}[F(w_t)] - \eta_t \mathbb{E}[\|\nabla F(w_t)\|^2] + \frac{L\eta_t^2}{2} \mathbb{E}[\|\nabla F(w_t)\|^2]. \tag{19}$$

Rearranging inequality 19, we get inequality 20.

$$\eta_t \mathbb{E}[\|\nabla F(w_t)\|^2] \leq \mathbb{E}[F(w_t)] - \mathbb{E}[F(w_{t+1})] + \frac{L\eta_t^2}{2} \mathbb{E}[\|\nabla F(w_t)\|^2]. \tag{20}$$

Summing over $t = 0$ to $T - 1$ in the inequality 20, we get inequality 21:

$$\sum_{t=0}^{T-1} \eta_t \mathbb{E}[\|\nabla F(w_t)\|^2] \leq F(w_0) - F(w_T) + \sum_{t=0}^{T-1} \frac{L\eta_t^2}{2} \mathbb{E}[\|\nabla F(w_t)\|^2. \tag{21}$$

Using the learning rate $\eta_t = \frac{\eta}{\sqrt{t+1}}$, the second term on the RHS of inequality 21 scales as $O(1/\sqrt{t+1})$. Dividing both sides by $\sum_{t=0}^{T-1} \eta_t$, we get inequality 22.

$$\frac{1}{\sum_{t=0}^{T-1} \eta_t} \sum_{t=0}^{T-1} \eta_t \mathbb{E}[\|\nabla F(w_t)\|^2] \leq \frac{F(w_0) - F(w_T)}{\sum_{t=0}^{T-1} \eta_t} + \frac{\sum_{t=0}^{T-1} \frac{L\eta_t^2}{2}}{\sum_{t=0}^{T-1} \eta_t}. \tag{22}$$

Simplifying further using the convexity of $F(w)$ and properties of η_t, we arrive at inequality 23.

$$\mathbb{E}[F(w_T) - F(w^*)] \leq \frac{C}{\sqrt{T}}, \tag{23}$$

where C is a constant that depends on L, σ, and the initial distance $\|w_0 - w^*\|^2$. The global objective function is defined as:

$$F(w) = \frac{1}{K} \sum_{i=1}^{K} f_i(w),$$

where $f_i(w)$ is the loss function for the client i. Using the smoothness property, we get inequality 24.

$$F(w_{t+1}) \leq F(w_t) + \langle \nabla F(w_t), w_{t+1} - w_t \rangle + \frac{L}{2} \|w_{t+1} - w_t\|^2. \tag{24}$$

Substitute $w_{t+1} - w_t = -\eta_t \nabla F(w_t)$ in the above inequality 24, we have inequality 25:

$$F(w_{t+1}) \leq F(w_t) - \eta_t \|\nabla F(w_t)\|^2 + \frac{L\eta_t^2}{2} \|\nabla F(w_t)\|^2. \tag{25}$$

Taking expectations of inequality 25 we get inequality 26:

$$\mathbb{E}[F(w_{t+1})] \leq \mathbb{E}[F(w_t)] - \eta_t \mathbb{E}[\|\nabla F(w_t)\|^2] + \frac{L\eta_t^2}{2} \mathbb{E}[\|\nabla F(w_t)\|^2]. \tag{26}$$

Rearranging the above inequality 26, we have inequality 27 as:

$$\eta_t \mathbb{E}[\|\nabla F(w_t)\|^2] \leq \mathbb{E}[F(w_t)] - \mathbb{E}[F(w_{t+1})] + \frac{L\eta_t^2}{2} \mathbb{E}[\|\nabla F(w_t)\|^2]. \tag{27}$$

Simplifying the above inequality, we get:

$$\mathbb{E}[F(w_t) - F(w_{t+1})] \geq \eta_t \mathbb{E}[\|\nabla F(w_t)\|^2] - \frac{L\eta_t^2}{2}\mathbb{E}[\|\nabla F(w_t)\|^2].$$

Let Δ_t denote the gradient noise. Using the bounded variance assumption:

$$\mathbb{E}[\|\Delta_t\|^2] \leq \sigma^2,$$

The overall gradient norm is bounded as:

$$\mathbb{E}[\|\nabla F(w_t)\|^2] \leq G^2 + \frac{\sigma^2}{K},$$

where G^2 is a bound on the true gradient norm. Incorporating the gradient variance bounds, we get the below inequality 28.

$$\mathbb{E}[F(w_t) - F(w_{t+1})] \geq \eta_t \mathbb{E}[\|\nabla F(w_t)\|^2] - \frac{L\eta_t^2}{2}\|\Delta_t\|^2. \tag{28}$$

Substitute $\|\Delta_t\|^2 \leq G^2$ in the above inequality, we have:

$$\mathbb{E}[F(w_t) - F(w_{t+1})] \geq \eta_t \mathbb{E}[\|\nabla F(w_t)\|^2] - \frac{L\eta_t^2 G^2}{2}.$$

Expressing this equivalently, we get inequality 29 and 30:

$$\mathbb{E}[F(w_{t+1}) - F(w_t)] \leq -\eta_t \|\nabla F(w_t)\|^2 + \frac{L\eta_t^2 G^2}{2} \tag{29}$$

$$\mathbb{E}[F(w_{t+1}) - F(w_t)] \leq -\eta_t \|\nabla F(w_t)\|^2 + \frac{L\eta_t^2}{2}\mathbb{E}[\|\Delta_t\|^2. \tag{30}$$

So we have bounded $\mathbb{E}\|\Delta_t\|^2 \leq 2(\lambda_{\max} L_{KL})^2 + 2\sigma^2/K$. Let's denote $G^2 = 2(\lambda_{\max} L_{KL})^2 + 2\sigma^2/K$ for simplicity. Now, by convexity of F, we have the inequality 31.

$$F(w_t) - F(w^*) \leq \langle \nabla F(w_t), w_t - w^* \rangle \tag{31}$$

Rearranging inequality 29, 30 and using this convexity property defined in inequality 31, we achieve the following:

$$\begin{aligned}
\eta_t \|\nabla F(w_t)\|^2 &\leq F(w_t) - \mathbb{E}[F(w_{t+1})] + \frac{L\eta_t^2 G^2}{2} \\
\langle \nabla F(w_t), w_t - w^* \rangle &\leq \frac{\|w_t - w^*\|^2 - \mathbb{E}[\|w_{t+1} - w^*\|^2]}{2\eta_t} + \frac{\eta_t}{2}\|\nabla F(w_t)\|^2 \\
F(w_t) - F(w^*) &\leq \frac{\|w_t - w^*\|^2 - \mathbb{E}[\|w_{t+1} - w^*\|^2]}{2\eta_t} + \frac{F(w_t) - \mathbb{E}[F(w_{t+1})]}{2} + \frac{L\eta_t G^2}{4}
\end{aligned}$$

Summing over $t = 0$ to $T - 1$ and using telescoping sums:

$$\begin{aligned}
\sum_{t=0}^{T-1}(F(w_t) - F(w^*)) \leq \frac{\|w_0 - w^*\|^2}{2\eta_0} + \sum_{t=1}^{T-1}\|w_t - w^*\|^2\left(\frac{1}{2\eta_t} - \frac{1}{2\eta_{t-1}}\right) \\
+ \frac{F(w_0) - \mathbb{E}[F(w_T)]}{2} + \frac{LG^2}{4}\sum_{t=0}^{T-1}\eta_t
\end{aligned}$$

Choosing $\eta_t = \frac{\eta}{\sqrt{t+1}}$ where $\eta > 0$ is a constant, we have:

$$\sum_{t=0}^{T-1} \eta_t \leq 2\eta\sqrt{T}$$

Using Jensen's inequality and the convexity of F:

$$F(\bar{w}_T) - F(w^*) \leq \frac{1}{T}\sum_{t=0}^{T-1}(F(w_t) - F(w^*)).$$

Combining these results:

$$F(\bar{w}_T) - F(w^*) \leq \frac{\|w_0 - w^*\|^2}{2\eta T\sqrt{T}} + \frac{F(w_0) - F(w^*)}{2T} + \frac{L\eta G^2\sqrt{T}}{2T}.$$

This is the convergence rate for the convex setting of the proposed FedDUAL method.

A.2 Convergence Analysis for Non-Convex Setting

From the smoothness assumption, we have below inequality 32:

$$F(w_{t+1}) \leq F(w_t) - \eta_t\|\nabla F(w_t)\|^2 + \frac{L\eta_t^2}{2}\|\nabla F(w_t)\|^2. \tag{32}$$

Taking expectations of inequality 32, we get 33:

$$\mathbb{E}[F(w_{t+1})] \leq \mathbb{E}[F(w_t)] - \eta_t\mathbb{E}[\|\nabla F(w_t)\|^2] + \frac{L\eta_t^2}{2}\mathbb{E}[\|\nabla F(w_t)\|^2]. \tag{33}$$

Rearranging the above inequality 33, we get inequality 34:

$$\eta_t\mathbb{E}[\|\nabla F(w_t)\|^2] \leq \mathbb{E}[F(w_t)] - \mathbb{E}[F(w_{t+1})] + \frac{L\eta_t^2}{2}\mathbb{E}[\|\nabla F(w_t)\|^2]. \tag{34}$$

Summing over $t = 0$ to $T - 1$ in the inequality 34, we get 35:

$$\sum_{t=0}^{T-1} \eta_t\mathbb{E}[\|\nabla F(w_t)\|^2] \leq F(w_0) - F(w_T) + \sum_{t=0}^{T-1}\frac{L\eta_t^2}{2}\mathbb{E}[\|\nabla F(w_t)\|^2]. \tag{35}$$

Assuming a constant learning rate $\eta_t = \eta$ and substituting it into inequality 35, we obtain 35:

$$\sum_{t=0}^{T-1} \eta\mathbb{E}[\|\nabla F(w_t)\|^2] \leq F(w_0) - F(w_T) + \frac{L\eta^2 T}{2}. \tag{36}$$

Rearranging inequality 36 yields 37.

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|\nabla F(w_t)\|^2] \leq \frac{F(w_0) - F(w_T)}{\eta T} + \frac{L\eta}{2}. \tag{37}$$

With η chosen as $\sqrt{\frac{2(F(w_0)-F(w^*))}{LG^2T}}$ to balance the terms, substituting it into inequality 37 results the following result:

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|\nabla F(w_t)\|^2] \leq C\sqrt{\frac{1}{T}}, \tag{38}$$

where C is a constant depending on $F(w_0)$, $F(w^*)$, L, and G^2. We proceed with the same inequality as established in the convex setting:

$$\mathbb{E}[F(w_{t+1}) - F(w_t)] \leq -\eta_t\|\nabla F(w_t)\|^2 + \frac{L\eta_t^2 G^2}{2} \tag{39}$$

Summing over $t = 0$ to $T-1$ in inequality 39, we obtain the following inequality:

$$\sum_{t=0}^{T-1}\mathbb{E}[F(w_{t+1}) - F(w_t)] \leq -\sum_{t=0}^{T-1}\eta_t\|\nabla F(w_t)\|^2 + \frac{LG^2}{2}\sum_{t=0}^{T-1}\eta_t^2$$
$$F(w^*) - F(w_0) \leq -\sum_{t=0}^{T-1}\eta_t\|\nabla F(w_t)\|^2 + \frac{LG^2}{2}\sum_{t=0}^{T-1}\eta_t^2.$$

Using the learning rate $\eta_t = \frac{\eta}{\sqrt{T}}$ (constant for all t), we have:

$$F(w^*) - F(w_0) \leq -\frac{\eta}{\sqrt{T}}\sum_{t=0}^{T-1}\|\nabla F(w_t)\|^2 + \frac{LG^2\eta^2}{2}$$
$$\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla F(w_t)\|^2 \leq \frac{\sqrt{T}}{\eta T}(F(w_0) - F(w^*)) + \frac{LG^2\eta}{2\sqrt{T}}.$$

By choosing $\eta = \sqrt{\frac{2(F(w_0)-F(w^*))}{LG^2}}$ to balance the terms, we obtain:

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(w_t)\|^2 \leq \sqrt{\frac{2LG^2(F(w_0) - F(w^*))}{T}}$$

This yields the required convergence rate for the non-convex setting of the proposed FedDUAL method.

B Discussion on convergence rates

B.1 Convex Setting

In the convex setting, The rate is optimal for first-order stochastic methods in convex optimization. The key points are:

1. The result is for the average iterate $\bar{w}_T$, which is common in convex stochastic optimization.
2. The rate depends on the initial distance to the optimum $\|w_0 - w^*\|$, the initial suboptimality $F(w_0) - F(w^*)$, and the bound on the stochastic gradients G^2.

B.2 Non-Convex Setting

For the non-convex setting, The result is significantly different from the convex case:

1. We measure convergence in terms of the gradient norm, not the function value, as finding global minima is generally intractable in non-convex settings.
2. The result is for the average of the squared gradient norms, not the minimum.
3. This rate is also optimal for first-order stochastic methods in non-convex smooth optimization.

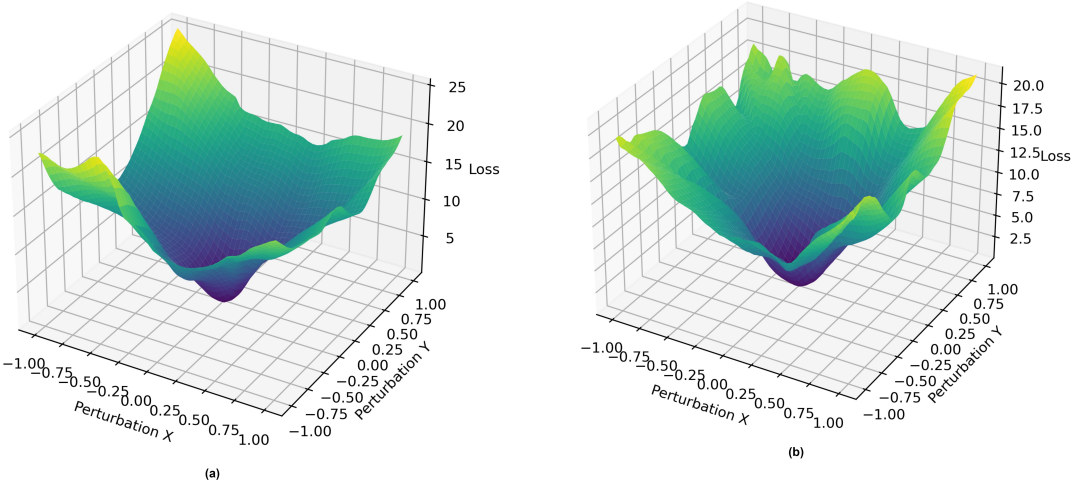

Figure 6: Visualization of the loss surface for the global model trained on the FMNIST dataset with non-IID data ($\alpha = 0.01$): (a) shows the loss surface for the global model trained using FedAvg, while (b) depicts the loss surface for the global model trained with the proposed method FedDUAL.

B.3 Comparison and Implications

1. Different Metrics: The convex and non-convex results use different metrics for convergence, which is standard in optimization theory. In the convex case, we bound the suboptimality of the function value, while in the non-convex case, we bound the average squared gradient norm.

2. Similarity in Rates: Both settings achieve an $O(1/\sqrt{T})$ rate, but the meaning is different. In the convex case, this rate applies to the optimality gap, while in the non-convex case, it applies to the average squared gradient norm.

3. Adaptive Loss and Wasserstein Barycenter: These novel components of our algorithm are captured in the constant factors and the bound G^2 on the stochastic gradients. They do not affect the asymptotic rate but improves the practical performance by reducing client heterogeneity.

4. Communication Efficiency: The $O(1/\sqrt{T})$ rate in both cases is in terms of the number of communication rounds, suggesting that our method is communication-efficient in both convex and non-convex settings.

5. Practical Implications: While the asymptotic rates are the same, the non-convex case may require more iterations to achieve a desired level of accuracy, as we're working with squared gradient norms rather than function values.

In conclusion, our analysis differentiates between the convex and non-convex settings, providing appropriate convergence guarantees for each case. The proposed federated learning algorithm achieves optimal rates in both settings, demonstrating its theoretical efficiency and robustness to the convexity of the objective function. We achieve a convergence rate of $O\left(\frac{1}{\sqrt{T}}\right)$ towards a stationary point under smooth, non-convex conditions. This rate is on par with the traditional FedAvg method in non-convex settings (Yu et al., 2019).

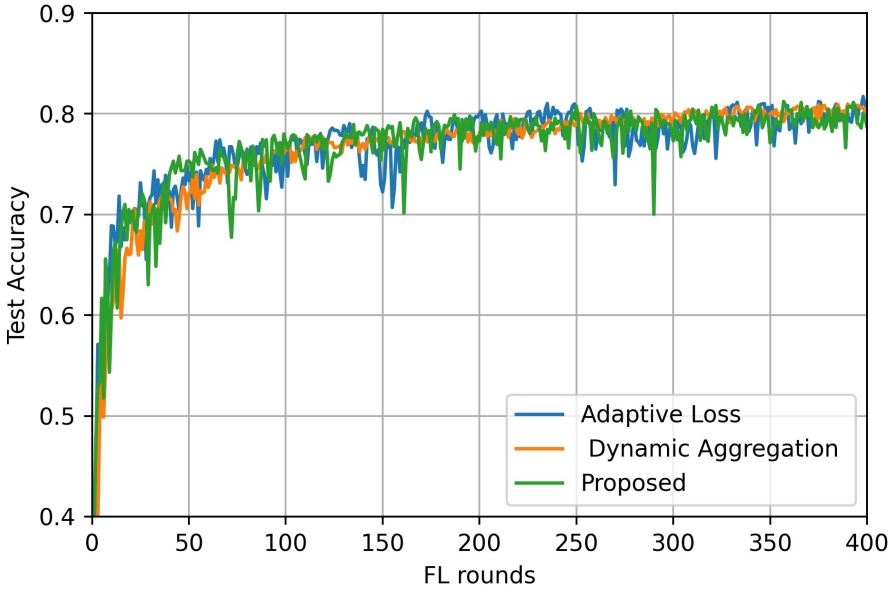

Figure 7: Learning curves of the individual modules and the proposed method.

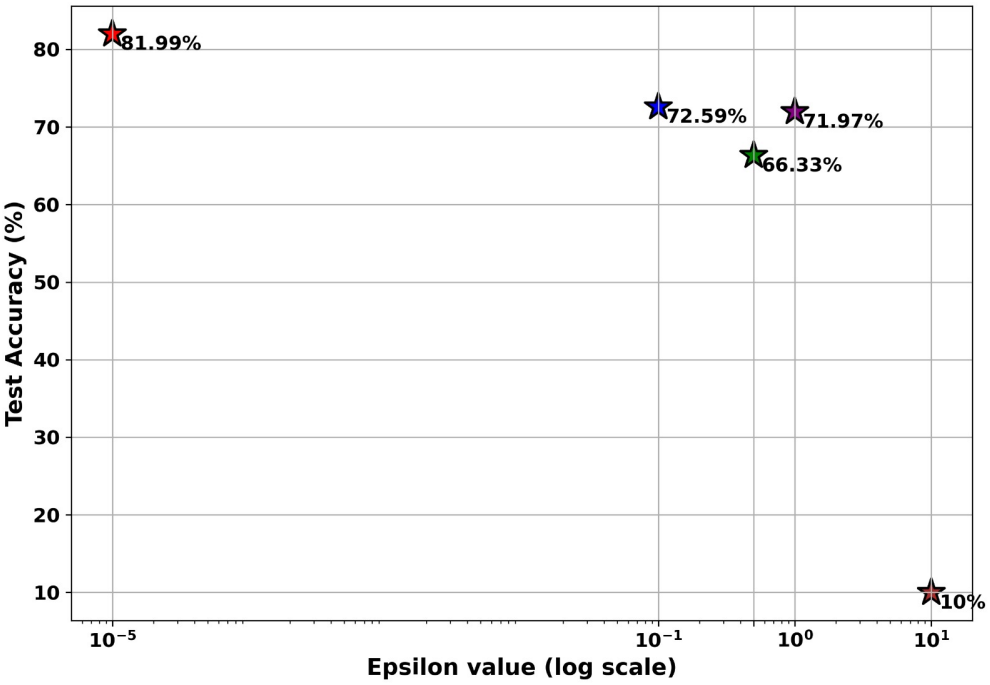

Figure 8: Performance of the proposed method with different epsilon values.

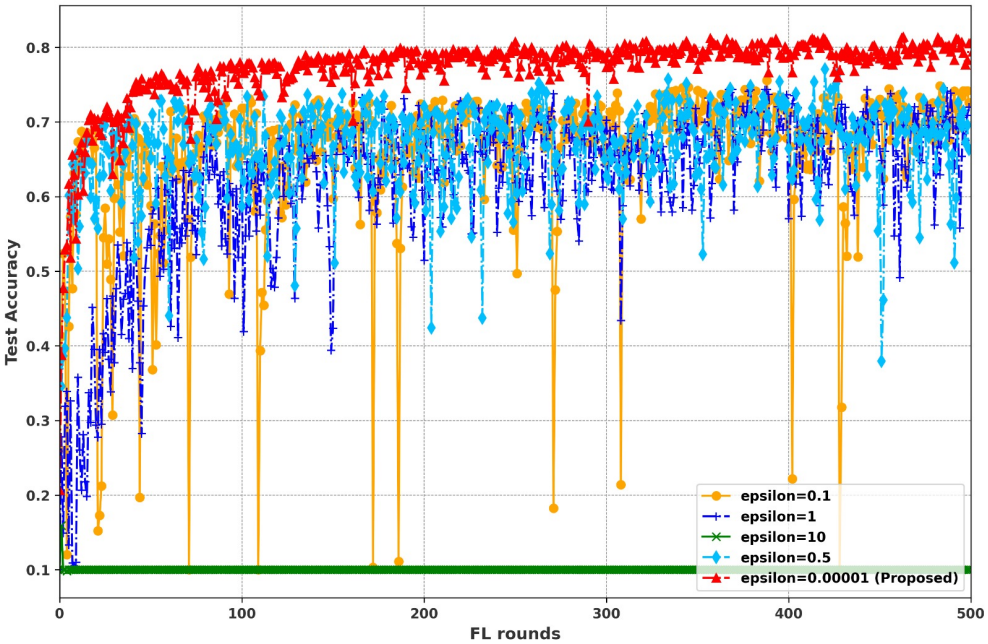

Figure 9: Learning curve of the proposed method with different epsilon values.

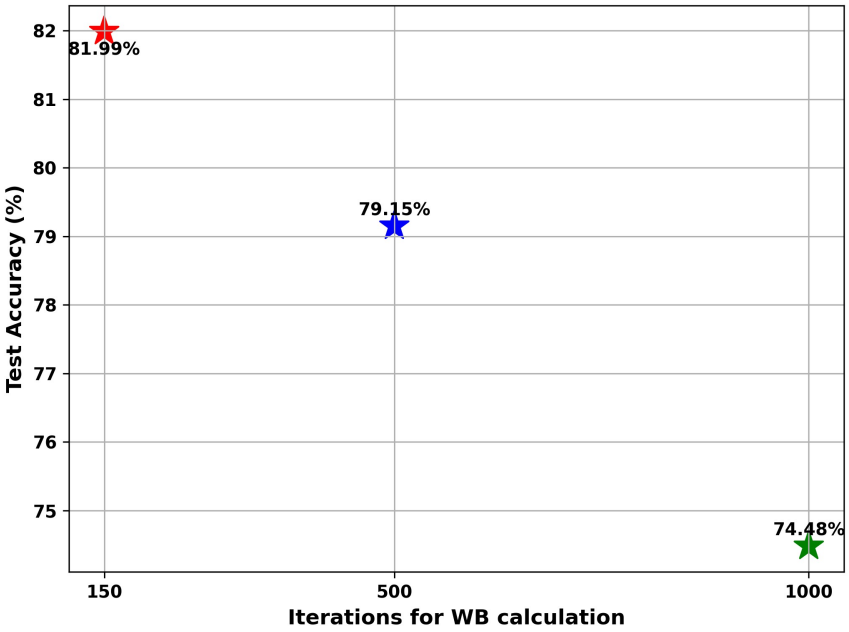

Figure 10: Performance across different number of Iterations for WB calculation.

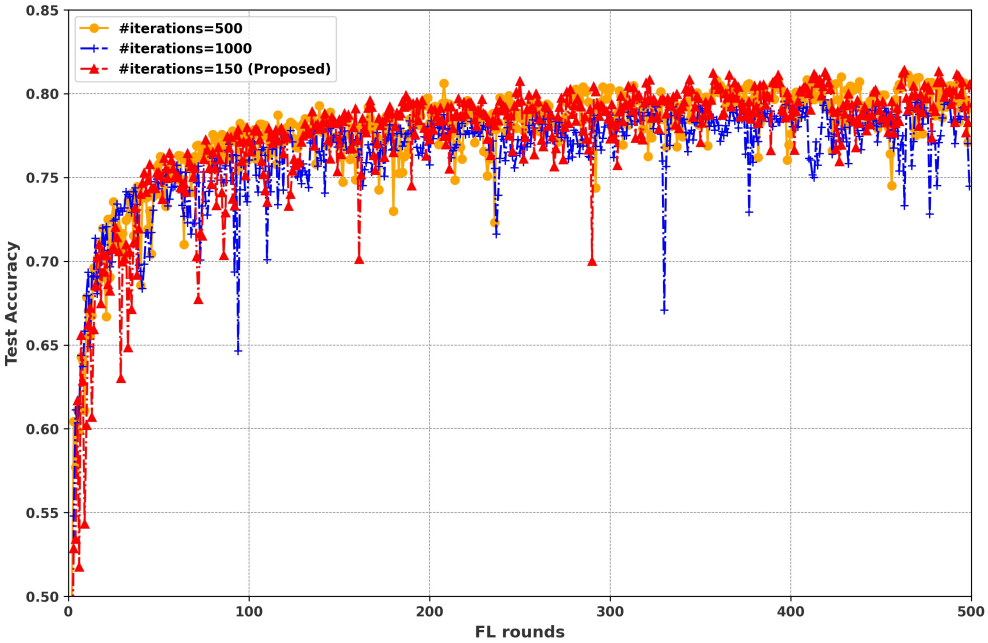

Figure 11: Learning curves for different number of Iterations for WB calculation.

C Ablation Study

C.1 Hyperparameter Analysis

In the proposed architecture, there are two key hyperparameters to consider: the scaling factor (γ) and the number of iterations used to compute the Wasserstein Barycenter. The detailed analysis of these hyperparameters is provided below.

The proposed FedDUAL approach utilizes dynamic server-side aggregation by applying the Wasserstein Barycenter concept to combine the weights of the final layers from local models. This iterative process involves a small positive constant (ϵ) to determine the scaling factor (γ). To optimize performance, we conducted two sets of experiments on the FMNIST dataset with $\alpha = 0.01$, each exploring a range of values for these crucial hyperparameters. The hyperparameter ϵ influences the sensitivity of the barycenter calculation to variations in Wasserstein distance. A smaller ϵ makes the barycenter more responsive to differences in Wasserstein distance, while a larger ϵ diminishes this sensitivity. This impacts how the barycenter integrates each distribution according to its distance from the current estimate. During the iterative update of the barycenter, ϵ affects the scaling factor γ applied to each distribution. An excessively small ϵ can result in slow or potentially non-existent convergence due to minimal scaling factor, whereas a too-large ϵ may cause oversmoothing, reducing the barycenter's effectiveness in accurately representing the distributions. For this setting we have fixed the number of iterations to compute Wasserstein Barycenter as 150. Figure 8 illustrates the test accuracy for different values of ϵ, suggesting that larger ϵ values may degrade performance or even failed to achieve convergence. Figure 9 presents the corresponding learning curves for these settings.

The number of iterations in the Wasserstein Barycenter function is another critical hyperparameter that affects both the accuracy and efficiency of the barycenter computation. Generally, more iterations enhance convergence and accuracy, ensuring that the barycenter more closely approximates the optimal value. However, increasing the number of iterations also prolongs computation time, necessitating a balance between accuracy and efficiency. Finding the optimal number of iterations involves a trade-off: too few iterations may result in suboptimal outcomes, while too many can yield diminishing returns in accuracy. To achieve the best performance, begin with a reasonable default value, monitor convergence by observing changes in the barycenter, and adjust iteratively based on empirical results and available computational resources. For

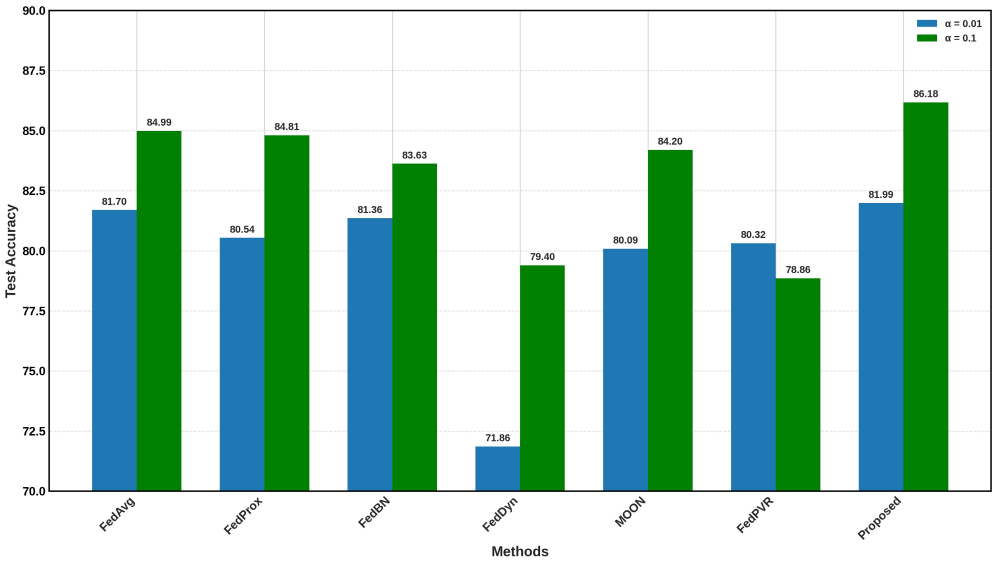

Figure 12: Illustrates the accuracy of the proposed method and baselines across different levels of data heterogeneity on the FMNIST dataset.

this setting, we fixed the *epsilon* value as 0.0001, which yields the highest results in previous experiment. Figure 10 illustrates the test accuracy for different values of iterations to calculate Wasserstein Barycenter, suggesting that larger iterations may adversely affect performance. Figure 11 presents the corresponding learning curves for these settings.

From both experiments, we observe that the highest performance is achieved with $\epsilon = 0.00001$ and 150 iterations. Therefore, to optimize performance, it is advisable to set ϵ to a smaller value while keeping the number of iterations between 100 and 150.

C.2 Experiment on Different Level of Data Heterogeneity

Figure 12 illustrates the accuracy of the proposed method and various baselines across different levels of data heterogeneity on the FMNIST dataset. In this context, heterogeneity is quantified by α, with lower values indicating greater data heterogeneity. The results show that as α decreases, the test accuracy for all models increases, because data heterogeneity among clients is decreased. Remarkably, the proposed method consistently achieves the highest test accuracy and exhibits the slowest performance decline compared to other algorithms, demonstrating superior performance of the proposed method on varying degrees of non-IID data partitioning. The learning curve is presented in the Fig. 13.

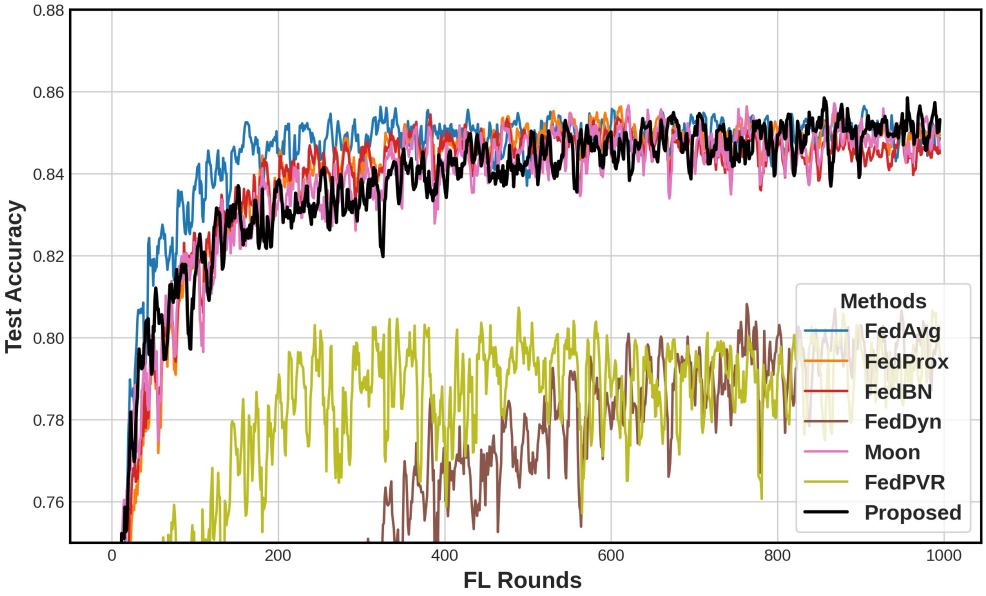

Figure 13: Learning curve of the proposed method and other baselines on FMNIST dataset with data heterogeneity level α=0.1.