FedFDP: Fairness-Aware Federated Learning with Differential Privacy

Xinpeng Ling[⋆,1][0009−0006−1605−7054], Jie Fu[⋆,2][0009−0009−9337−1452], Kuncan Wang[3][0009−0004−4756−5938], Huifa Li[1][0009−0009−5806−385X], Tong Cheng[1][0009−0001−3120−6103], and Zhili Chen[†,1][0000−0002−2231−3652]

[1] East China Normal University, Shanghai, China
{xpling,huifali,tcheng}@stu.ecnu.edu.cn,zhlchen@sei.ecnu.edu.cn
[2] Stevens Institute of Technology, Hoboken NJ 07030, USA
jfu13@stevens.edu
[3] Nanyang Technological University, Singapore
kuncan001@e.ntu.edu.sg

Abstract. Federated learning (FL) is an emerging machine learning paradigm designed to address the challenge of data silos, attracting considerable attention. However, FL encounters persistent issues related to fairness and data privacy. To tackle these challenges simultaneously, we propose a fairness-aware federated learning algorithm called FedFair. Building on FedFair, we introduce differential privacy to create the FedFDP algorithm, which addresses trade-offs among fairness, privacy protection, and model performance. In FedFDP, we developed a fairness-aware gradient clipping technique to explore the relationship between fairness and differential privacy. Through convergence analysis, we identified the optimal fairness adjustment parameters to achieve both maximum model performance and fairness. Additionally, we present an adaptive clipping method for uploaded loss values to reduce privacy budget consumption. Extensive experimental results show that FedFDP significantly surpasses state-of-the-art solutions in both model performance and fairness.

Keywords: Federated Learning · Differential Privacy · Fairness.

1 Introduction

Federated Learning (FL) [30] is a distributed machine learning framework that allows clients to collaboratively train a shared model without exposing their respective datasets. After training their local models, each client only transmits model parameters to the central server, rather than the original training data. The server aggregates these parameters to update the global model, which is then sent back to the clients for further training. Due to its ability to improve model performance through collaboration without the need to upload original data, overcoming data silos, FL has received significant attention in recent years [6, 20, 44]. Particularly, many studies have achieved significant breakthroughs in terms of model performance [23, 24, 38] and communication costs [28, 39] in FL.

⋆: Those authors contribute equally.
†: Corresponding author.

To encourage more clients to participate in federated learning, establishing a fairness-aware federated learning algorithm is necessary [22]. Current fairness-aware machine learning algorithms, such as [8, 10], have not fully addressed the issue of client performance disparities in federated learning. Due to non-independent and identically distributed (Non-IID) data, inconsistent client objectives (orange squares) lead to significant performance disparities of the collaboratively trained global model (red square) across different datasets held by various clients. An increasing number of scholars are beginning to focus on this phenomenon of unfair performance disparity [23, 25, 42].

On the other hand, recent studies have pointed out that training parameters may leak data privacy, such as recovering original data [43] or member inference attacks [35]. Even if only models or gradients are uploaded instead of original data, data leakage can still occur [31]. To enhance data privacy in FL, differential privacy (DP) [11] has become the preferred method for protecting privacy in federated learning due to its moderate computational overhead and solid mathematical foundation. Incorporating appropriate DP during the training phase can effectively prevent the accidental leakage of sensitive training information [12].

Although some works have explored balanced performance fairness and differential privacy in federated learning separately, no prior research has considered them in a unified framework. For the first time, we propose a fairness-aware federated learning with differential privacy. However, balancing differential privacy protection, fairness, and model performance presents a significant challenge. Specifically, there are two main challenges: First, the clipping and noise addition processes in differential privacy have the potential to affect both fairness and model utility in federated learning. Moreover, achieving fairness in federated learning requires uploading loss information to the server, which entails additional privacy budget consumption.

In this paper, the relationship between the fairness loss function and differential privacy in federated learning is analyzed to address the first challenge. We design an fairness-aware gradient clipping strategy to balance fairness and differential privacy. This gradient clipping strategy not only meets the requirements for fairness but also satisfies the sensitivity control needed for differential privacy. Furthermore, through convergence analysis, we demonstrate that an optimal adjustment parameter can be found to achieve the best model performance and fairness. To address the second challenge, we design an adaptive clipping method for loss uploading, significantly reducing the privacy budget consumption in this part.

Our main contributions are as follows:

1. First, a fairness-aware federated algorithm, FedFair, is proposed. It integrates a novel loss function specifically designed to simultaneously optimize fairness and model performance.
2. Based on FedFair, we propose an algorithm called FedFDP to further equip the system with differential privacy. In particular, we designed a fairness-aware gradient clipping strategy that ensures differential privacy and allows for adjustment of fairness in federated learning. In addition, we propose an adaptive clipping method for the additional loss values uploaded by each client to achieve optimal utility.
3. Furthermore, we conducted a convergence analysis of FedFDP and identified the fairness parameter λ^* that results in the fastest convergence. We analyzed the pri-

vacy loss of the FedFDP algorithm using the concept of Rényi Differential Privacy (RDP) to verify its compliance with differential privacy requirements.
4. Finally, through comprehensive empirical evaluations on three public datasets, we confirmed that FedFair and FedFDP are superior to existing solutions in terms of model performance and fairness. What's more, we investigated the impact of clipping bound and noise multiplier on fairness.

In Section 2, we formalize our problem and introduce the three objectives of our method. Section 3 presents a fair federated learning approach—FedFair. In Section 4, we enhance FedFair by incorporating the DP guarantee, thereby arriving at the proposed FedFDP. A *fairness parameter* λ is embedded within FedFDP, and in Section 5, we derive the analytical solution for the optimal λ^* through convergence analysis and differential methods. The privacy analysis of FedFDP are examined in detail in Section 6. In Section 7, we conduct extensive experiments on three datasets, comparing six baseline algorithms to rigorously validate the effectiveness of both FedFair and FedFDP. Following the review of related work in Section B, Section 9 provides a summary of our contributions. Due to space limitations, the detailed procedures of the convergence analysis are included in Appendix A.

2 Problem Formulation

In this work, we propose FedFDP which aims to serve threefold goals as follows. Then, we will elaborate on and formalize how fairness and DP are defined in FL.

- **Goal 1 (Fairness)**: Strive to achieve federated learning with the highest possible fairness, where lower values of the federated fairness metric (Equation 2) indicate better fairness.
- **Goal 2 (Privacy)**: Implement federated learning under formal differential privacy guarantees (Equation 3).
- **Goal 3 (Utility)**: Ensure that the federated learning model retains strong predictive performance, particularly in terms of classification accuracy.

2.1 Fairness in FL

We use $\mathbf{w}_t^i$ signifies the model acquired from client i at iteration t, while $\mathbf{w}_t$ denotes the server-aggregated model. The objective of FL is to minimize $F(\cdot)$, in other words, to seek:

$$\mathbf{w}^* = \arg\min F(\mathbf{w}), \text{where } F(\mathbf{w}) = \sum_{i=1}^{N} p_i F_i(\mathbf{w}). \tag{1}$$

Here, $F_i(\cdot)$ represents the local loss function of client i. The weights are defined as $p_i = \frac{|D_i|}{\sum_j |D_j|}$, D_i represents the dataset of client i.

We have defined balanced performance fairness in FL as follows.

Definition 1. *(**Balanced performance fairness [25]**) For trained models $\mathbf{w}_1$ and $\mathbf{w}_2$, we informally recognize that model $\mathbf{w}_1$ provides a more fair solution to the federated learning objective in Equation (1) than model $\mathbf{w}_2$ if the performance of model $\mathbf{w}_1$ on the N devices is more uniform than the **performance** of model $\mathbf{w}_2$ on the N devices.*

Differing from [28] that uses the variance of accuracy for "performance", we employ the variance of loss. This enables the metric to be incorporated directly into the objective function, facilitating more straightforward iterative optimization. Equation (2) represents the specific form of fairness in federated learning, which is the **weighted variance** of the loss values of global trained model $\mathbf{w}$ across all clients. Essentially, the smaller the value of Ψ, the fairer the $\mathbf{w}$ for all clients:

$$\Psi(\mathbf{w}) = \sum_{i=1}^{N} p_i \left(F_i(\mathbf{w}) - F(\mathbf{w})\right)^2. \tag{2}$$

2.2 DP guarantee in FL

DP is a formal mathematical model that quantifies privacy leakage in data analysis algorithms. It stipulates that alterations to a single record within the training data should not induce significant statistical variations in the algorithm's results.

Definition 2. *(**DP [11]**). (ϵ, δ)-DP is achieved by a randomized mechanism $\mathcal{M} : \mathcal{X}^n \to \mathbb{R}^d$, for any two neighboring databases $D_i, D_i' \in \mathcal{X}^n$ that **differ in only a single data sample**, and $\forall\, \mathcal{S} \subseteq$ Range$(\mathcal{R})$:*

$$\Pr\left[\mathcal{M}\left(D_i\right) \in \mathcal{S}\right] \leq e^{\epsilon} \Pr\left[\mathcal{M}\left(D_i'\right) \in \mathcal{S}\right] + \delta. \tag{3}$$

By adding random noise, we can achieve differential privacy for a function $f : \mathcal{X}^n \to \mathbb{R}^d$ according to Definition 2. The l_2-sensitivity determines how much noise is needed and is defined as follow.

DPSGD. In gradient-based methods, DPSGD [1] is widely used in privacy preservation. It performs per-sample gradient clipping and noise addition:

- Firstly, computing the gradient for data sample ξ_j, donated by $\mathbf{g}_t^{i,j} = \nabla F_i(\mathbf{w}_t^i, \xi_j)$.
- Secondly, clipping the gradient of per-sample to get the sensitivity:

$$\hat{\mathbf{g}}_t^{i,j} = \mathbf{g}_t^{i,j} / \max\left(1, \frac{\|\mathbf{g}_t^{i,j}\|}{C}\right), \tag{4}$$

  where C represents the clipping bound.
- Thirdly, adding Gaussian noise with mean 0 and standard deviation $C \cdot \sigma$ to the sum of clipped gradients, where σ is the noise multiplier, $\mathbf{I}$ is the identity matrix:

$$\tilde{\mathbf{g}}_t^i = \frac{1}{|\mathcal{B}_i|}\left(\sum_{j=1}^{|\mathcal{B}_i|} \hat{\mathbf{g}}_t^{i,j} + C\sigma \cdot \mathcal{N}(0, \mathbf{I})\right). \tag{5}$$

The final step is to proceed with gradient descent as usual. When each client in FL utilizes DPSGD as the optimization algorithm to safeguard their private data, the framework transitions from FL to DPFL.

Algorithm 1: FedFair

Input: loss function $F(\mathbf{w})$. Parameters: learning rate η, fairness hyperparameter λ, batch sample ratio q, local dataset D_i.
Output: the final trained model $\mathbf{w}_T$

1 Initialize $\mathbf{w}_0, F(\mathbf{w}_0) =$ Initial();
2 ***SERVER***
3 **for** $t = 0, 1, \cdots, T-1$ **do**
4 **for** $i = 1, 2, \cdots, N$ *parallel* **do**
5 $\mathbf{w}_{t+1}^i, F_i(\mathbf{w}_{t+1}^i) =$ LOCAL $(\mathbf{w}_t, F(\mathbf{w}_t))$;
6 $\mathbf{w}_{t+1} = \sum_{i=1}^{N} p_i \mathbf{w}_{t+1}^i$;
7 $F(\mathbf{w}_{t+1}) = \sum_{i=1}^{N} p_i F_i(\mathbf{w}_{t+1}^i)$;
8 **return** $\mathbf{w}_T$;
9 ***LOCAL***
10 Download $\mathbf{w}_t, F(\mathbf{w}_t)$ and $\mathbf{w}_t^i \leftarrow \mathbf{w}_t$;
11 $\{\mathcal{B}_i\} \leftarrow$ Split D_i to batches;
12 **for** *batch* $b \in \{\mathcal{B}_i\}$ **do**
13 $\Delta_i = F_i(\mathbf{w}_t^i, b) - F(\mathbf{w}_t)$;
14 $\eta_i = \eta \cdot (1 + \lambda \cdot \Delta_i)$;
15 $\mathbf{w}_t^i = \mathbf{w}_t^i - \eta_i \cdot \nabla F_i(\mathbf{w}_t^i, b)$;
16 $\mathbf{w}_{t+1}^i \leftarrow \mathbf{w}_t^i$, compute $F_i(\mathbf{w}_{t+1}^i)$;
17 **return** $\mathbf{w}_{t+1}^i, F_i(\mathbf{w}_{t+1}^i)$;

3 FedFair: Equipping FL with Fairness

We previously defined the performance objective in Equation (1) and balanced performance fairness in Definition 1. Here, we use these to develop a new objective function that informs our proposed fairness-aware federated learning algorithm.

Upon incorporating Equation (2) into Equation (1), we introduce a comprehensive objective function that integrates considerations of fairness:

$$\min_{\mathbf{w}} H(\mathbf{w}) = F(\mathbf{w}) + \frac{\lambda}{2} \sum_{i=1}^{N} p_i \left(F_i(\mathbf{w}) - F(\mathbf{w})\right)^2. \tag{6}$$

For client i during round t, the refined objective is articulated as follows:

$$\min_{\mathbf{w}_t^i} H_i(\mathbf{w}_t^i) = F_i(\mathbf{w}_t^i) + \frac{\lambda}{2} \left(F_i(\mathbf{w}_t^i) - F(\mathbf{w}_t^i)\right)^2, \tag{7}$$

where $F(\mathbf{w}_t^i) = \sum_{i=1}^{N} p_i F_i(\mathbf{w}_t^i)$ and $F_i(\mathbf{w}_t^i)$ is determined using the training dataset[4] from round $t-1$, as delineated in line 16 of Algorithm 1, and the λ ($\lambda \geq 0$) is a hyperparameter used to adjust the degree of fairness [5]. Consequently, within the context of

[4] We use $F(\mathbf{w}_t) = \sum_{i=1}^{N} p_i F_i(\mathbf{w}_t^i)$, the reason for $F(\mathbf{w}_t) \neq F(\mathbf{w}_t, \xi_{test})$ is that the test data ξ_{test} from the server cannot be used for training.

[5] When $\lambda = 0$, the algorithm becomes traditional federated learning. When $\lambda \to \infty$, the algorithm only focuses on fairness. However, if λ is too large, it will cause very steep gradients

Equation (7), $F(\mathbf{w}_t^i)$ is considered a constant when the gradient is being computed. So, the gradient of Equation (7) as follows:

$$\begin{aligned}\nabla H_i(\mathbf{w}_t^i) &= \nabla F_i(\mathbf{w}_t^i) + \lambda \Delta_i \cdot \nabla F_i(\mathbf{w}_t^i) \\ &= (1 + \lambda \Delta_i) \nabla F_i(\mathbf{w}_t^i),\end{aligned} \tag{8}$$

where $\Delta_i = F_i(\mathbf{w}_t^i) - F(\mathbf{w}_t^i)$.

The stochastic gradient descent by using Equation (8) as show:

$$\mathbf{w}_{t+1}^i = \mathbf{w}_t^i - \eta(1 + \lambda \Delta_i) \nabla F_i(\mathbf{w}_t^i) = \mathbf{w}_t^i - \eta_i \nabla F_i(\mathbf{w}_t^i), \tag{9}$$

where $\eta_i = \eta(1 + \lambda \Delta_i)$.

Therefore, our approach can be recognized as establishing an **dynamic learning rate** that allows the model update to achieve both performance and fairness objectives simultaneously. Based on these principles, we introduce the FedFair algorithm, as describe in Algorithm 1:

1. **Server (lines 3-7):** The server executes T rounds of communication, each round broadcasting the aggregated model $\mathbf{w}_t$ and the loss $F(\mathbf{w}_t)$ to clients.
2. **Local (lines 12-15):** For each batch b, we compute the dynamic learning rate η_i in **lines 13-14**. For **line 15**, we execute gradient descent by using η_i.

4 FedFDP: Further Equipping FedFair with Differential Privacy

In this section, we present our algorithm, FedFDP, which further integrates FedFair with differential privacy. The flow of FedFDP algorithm is shown in Figure 1. In addition to adding Gaussian noise to the gradients, we introduce two additional processes to balance fairness and DP: 1) A fair-clipping strategy is added to FedFair. This strategy not only achieves sensitivity but also allows adjustments to the level of fairness. 2) An adaptive clipping method is employed when protecting the loss values that clients share with differential privacy, aiming to maximize utility. Next, we discuss the details of these two strategies.

4.1 Fair-clipping Strategy for Gradient

In the general DPSGD algorithm [1, 5, 13, 41], per-sample clipping is an essential process for obtaining sensitivity. In order to be compatible with per-sample clipping, initially, we expand our objective function of client i in FedFair, Equation (7), into Equation (10):

$$\min_{\mathbf{w}_t^i} H_i(\mathbf{w}_t^i, \xi_j) = F_i(\mathbf{w}_t^i, \xi_j) + \frac{\lambda}{2} \left(F_i(\mathbf{w}_t^i, \xi_j) - F(\mathbf{w}_t^i) \right)^2, \tag{10}$$

which might prevent the model from converging. Fortunately, we theoretically found an optimal value for λ in Section 5.1. In addition, as shown in Figure 2, we experimentally confirmed the existence of the optimal λ.

where $\xi_j \in \mathcal{B}_t^i$, $j = \{1, 2, \cdots, |\mathcal{B}_i|\}$, and $|\mathcal{B}_i|$ is the batch size of client i.
Next, we compute the gradient of Equation (10):

$$\nabla H_i(\mathbf{w}_t^i, \xi_j) = \left(1 + \lambda \cdot \Delta_i^j\right) \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j), \tag{11}$$

where $\Delta_i^j = F_i(\mathbf{w}_t^i, \xi_j) - F(\mathbf{w}_t^i)$.

To ensure the algorithm is safeguarded by differential privacy, we replace the gradient of the loss function with Equation (11) and reformulate the gradient descent computation for client i as follows:

$$\begin{aligned}
\mathbf{w}_{t+1}^i &= \mathbf{w}_t^i - \eta \tilde{\mathbf{g}}_t^i \\
&= \mathbf{w}_t^i - \frac{\eta}{|\mathcal{B}_i|} \left[\sum_{j=1}^{|\mathcal{B}_i|} \frac{\left(1 + \lambda \cdot \Delta_i^j\right) \nabla F_i(\mathbf{w}_t^i, \xi_j)}{\max\left(1, \frac{\left(1+\lambda \cdot \Delta_i^j\right) \|\nabla F_i(\mathbf{w}_t^i, \xi_j)\|}{C}\right)} + \sigma C \cdot \mathcal{N}(0, \mathbf{I}) \right] \\
&= \mathbf{w}_t^i - \frac{\eta}{|\mathcal{B}_i|} \left[\sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j) + \sigma C \cdot \mathcal{N}(0, \mathbf{I}) \right],
\end{aligned} \tag{12}$$

where:

$$C_t^{i,j} = \min\left(1 + \lambda \cdot \Delta_i^j, \frac{C}{\|\nabla F_i(\mathbf{w}_t^i, \xi_j)\|}\right). \tag{13}$$

Hence, our strategy can be considered as a **fair-clipping** technique, tailored to meet the demands of differential privacy while also nurturing fairness within the model. When $\lambda = 0$, Equation (13) corresponds to the traditional DP clipping method without fairness. For large C, Equation (13) simplifies to the first term, prompting FedFDP to adjust towards enhanced fairness. Conversely, for small C, Equation (13) simplifies to the second term, resulting in the gradient being clipped to the bound C. The impact of C on the performance of FedFDP will be further discussed in the experiments.

4.2 Adaptive Clipping Method for Loss

Since calculating Δ_i^j requires clients to upload $F_i(\mathbf{w}_t^i)$, differential privacy must be incorporated to ensure the algorithm preserves privacy. As shown in lines 17-20 of Algorithm 2, $F_i(\mathbf{w}_t^i)$ must be clipped to the interval $[0, C_l^{i,t}]$, followed by the addition of Gaussian noise with a mean of 0 and a standard deviation of $\sigma_l \cdot C_l^{i,t}$ to ensure differential privacy. However, determining an appropriate clipping bound $C_l^{i,t}$ for $F_i(\mathbf{w}_t^i)$ is challenging. In federated learning, due to the heterogeneity of data held by different clients, $F_i(\mathbf{w}_t^i)$ can vary significantly. A clipping norm that is too large introduces excessive noise, while one that is too small leads to aggressive clipping of gradient directions, impairing model performance. Therefore, an adaptive clipping strategy is necessary.

Algorithm 2: FedFDP

Input: loss function $F(\mathbf{w})$, learning rate η, noise multiplier for gradient σ, noise multiplier for loss σ_l, fairness hyperparameter λ, batch sample ratio q, original clipping bound C.

Output: the final trained model $\mathbf{w}_T$

SERVER

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

  for $i = 1, 2, \cdots, N$ *parallel* **do**

    $\mathbf{w}_{t+1}^i, \tilde{F}_i(\mathbf{w}_{t+1}^i) = \text{LOCAL}\,(\mathbf{w}_t, F(\mathbf{w}_t))$;

  $\mathbf{w}_{t+1} = \sum_{i=1}^N p_i \mathbf{w}_{t+1}^i$;

  $F(\mathbf{w}_{t+1}) = \sum_{i=1}^N p_i \tilde{F}_i(\mathbf{w}_{t+1}^i)$;

return $\mathbf{w}_T$;

LOCAL

Download $\mathbf{w}_t, F(\mathbf{w}_t)$ and $\mathbf{w}_t^i \leftarrow \mathbf{w}_t$;

Sample randomly a batch $\mathcal{B}_i$ with probability q;

for $j = 1, 2, \cdots, |\mathcal{B}_i|$ **do**

  $\Delta_i^j = F_i(\mathbf{w}_t^i, \xi_j) - F(\mathbf{w}_t)$;

  Compute $C_t^{i,j}$ by Equation (13);

  $\tilde{\mathbf{g}}_t^{i,j} = C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j)$;

$\mathbf{w}_{t+1}^i = \mathbf{w}_t^i - \frac{\eta}{|\mathcal{B}_i|}\left(\sum_{j=1}^{|\mathcal{B}_i|} \tilde{\mathbf{g}}_t^{i,j} + \sigma C \cdot \mathcal{N}(0, \mathbf{I})\right)$;

Compute clipping bound of loss $C_l^{i,t}$ by Equation (14) and (15) ;

for $j = 1, 2, \cdots, |\mathcal{B}_i|$ **do**

  Compute $F_i(\mathbf{w}_{t+1}^i, \xi_j)$;

  $f_{i,j} = \min(C_l^{i,t}, \max(0, F_i(\mathbf{w}_{t+1}^i, \xi_j)))$;

$\tilde{F}_i(\mathbf{w}_{t+1}^i) = \frac{1}{|\mathcal{B}_i|}\left(\sum_{j=1}^{|\mathcal{B}_i|} f_{i,j} + \sigma_l C_l^{i,t} \mathcal{N}(0, \mathbf{I})\right)$;

return $\mathbf{w}_{t+1}^i, \tilde{F}_i(\mathbf{w}_{t+1}^i)$;

Based on the data heterogeneity of federated learning, we propose an adaptive clipping method for $F_i(\mathbf{w}_t^i)$. For the i-th client, the adaptive clipping method uses the *differentially private mean of the previous round* as the clipping bound for the current round. The clipping bound $C_l^{i,t}$ for round t and client i is defined as follows, when $F_i(\mathbf{w}_{t-1}^i)$ is the individual loss of the previous round and σ_l^{t-1} is the individual noise multiplier of the previous round.

$$C_l^{i,t} = \frac{\sum_{j=1}^{|\mathcal{B}_i|} \text{clip}\left(F_i(\mathbf{w}_{t-1}^i, \xi_j)\right) + \mathcal{N}\left(0, (C_l^{i,t-1} \cdot \sigma_l)^2\right)}{|\mathcal{B}_i|}, \tag{14}$$

where

$$\text{clip}\left(F_i(\mathbf{w}_{t-1}^i, \xi_j)\right) = \min(C_l^{i,t-1}, \max(0, F_i(\mathbf{w}_{t-1}^i, \xi_j))). \tag{15}$$

To ensure differential privacy during the clipping process, we use the noise-added loss $C_l^{i,t}$ as the clipping bound. For each communication round, the clipping bound follows the post-processing property of differential privacy.

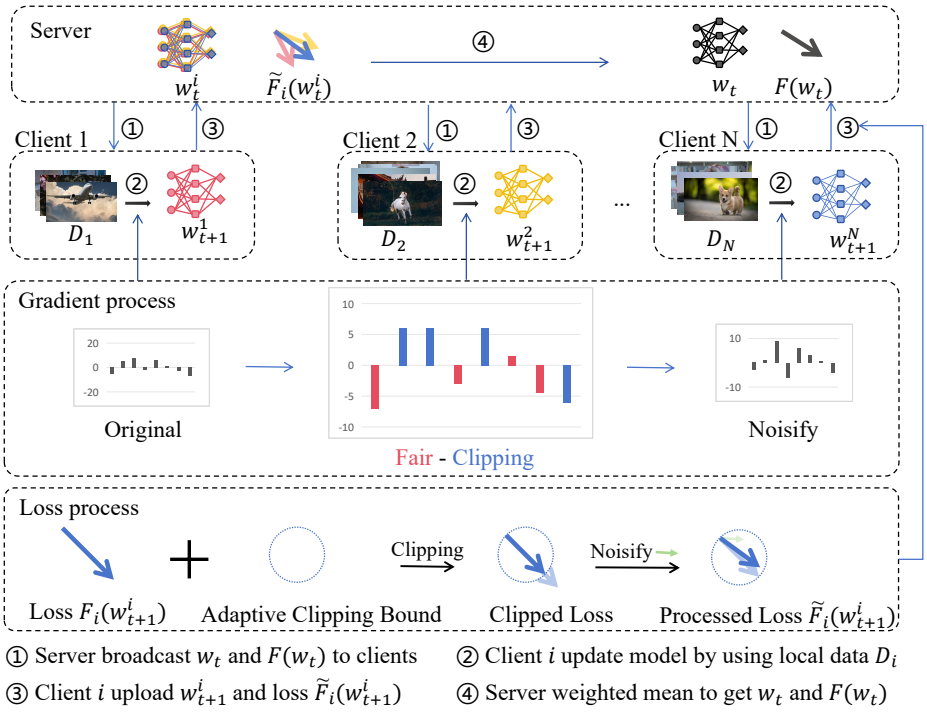

Fig. 1. FedFDP framework. Besides the privately computed local model $\mathbf{w}_{t+1}^i$ from the "Fair-Clipping", client i is also required to upload the private loss $\widetilde{F}_i(\mathbf{w}_{t+1}^i)$.

4.3 Overall Algorithm

We utilize the equation form Equation (12) to enhance Algorithm 2. Compared with the Algorithm 1, FedFDP solely necessitates the application of differential privacy to gradient and loss.

1. **DPSGD (lines 11-15):** In **lines 12-14**, we compute the fair-clipping bound $C_t^{i,j}$ and the per-sample gradient $\nabla F_i(\mathbf{w}_t^i, \xi_j)$ to get the processed gradient $\tilde{\mathbf{g}}_t^{i,j}$. Then, we add mean 0 noise to the sum of per-sample processed gradient $\tilde{\mathbf{g}}_t^{i,j}$ in **line 15** and execute the gradient decent.
2. **DP for local loss (lines 16-20):** As described in Section 4.2, add noise to the clipped local loss by using adaptive clipping bound $C_l^{i,t}$.

5 Finding Optimal Fairness Parameter

In this section, we discuss how to identify an optimal fairness parameter λ through the convergence analysis of the FedFDP algorithm.

All assumptions and detailed procedures for the convergence analysis are provided in Appendix A.2, leading to the following result:

Theorem 1 (Convergence rate of FedFDP (Simplified version of Theorem 5)).

$$\mathbb{E}[F(\mathbf{w}_T)] - F^* \leq \mathcal{O}\left(\frac{LG^2\lambda^2}{\mu^2 T}\right) + \mathcal{O}\left(\frac{L^2\Gamma\lambda}{\mu^2 T}\right) + \mathcal{O}\left(\frac{L\sigma^2 C^2 d}{\mu^2 T\lambda}\right) + \mathcal{O}\left(\frac{L\mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2}{T}\right), \tag{16}$$

where L, μ, and G are parameters defined in Assumptions 1–3, and Γ is a measure of heterogeneity; see Appendix A.1 for details.

As demonstrated in (16), the fairness parameter λ ($\lambda > 0$) exhibits convex influence on the convergence rate. The analytical solution of this relationship will be obtained via extremum analysis in Section 5.1.

5.1 Optimal Fairness Parameter λ^*

Assuming Δ_i^j is bounded as $Q_0 \leq \Delta_i^j \leq Q_1$, then $1 + Q_0\lambda \leq C_t^{i,j} \leq 1 + Q_1\lambda$. We rearrange the convergence upper bound in Equation (16) to obtain a function in terms of λ:

$$P(\lambda) = \frac{L}{2\mu T}\frac{a_1\lambda^3 + a_2\lambda^2 + a_3\lambda + a_4}{a_5\lambda + 1}, \tag{17}$$

where $a_1 = G^2Q_1^3$, $a_2 = 6G^2Q_1^2$, $a_3 = 9Q_1G^2 + 2L\Gamma Q_1 + 2Q_0\mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2$, $a_4 = 4G^2 + 2L\Gamma + \frac{2\sigma^2C^2d}{\hat{B}^2} + \mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2$, $a_5 = 2Q_0$.

Let $\mathcal{F}(\lambda) = \frac{a_1\lambda^3 + a_2\lambda^2 + a_3\lambda + a_4}{a_5\lambda + 1}$, then $P(\lambda) = \frac{L}{2\mu T}\mathcal{F}(\lambda)$, where $\frac{L}{2\mu T}$ can be considered a constant term. Obviously, when $\mathcal{F}(\lambda)$ is minimized, $P(\lambda)$ is also minimized. The derivative of $\mathcal{F}(\lambda)$ is as follows:

$$\mathcal{F}'(\lambda) = \frac{2a_1a_5\lambda^3 + (a_2a_5 + 3a_1)\lambda^2 + 2a_2\lambda + a_3 - a_4a_5}{(a_5\lambda + 1)^2}. \tag{18}$$

Let the denominator of $\mathcal{F}'(\lambda)$ be a new function:

$$\mathcal{G}(\lambda) = 2a_1a_5\lambda^3 + (3a_1 + a_2a_5)\lambda^2 + 2a_2\lambda + a_3 - a_4a_5. \tag{19}$$

Since $(a_5\lambda + 1)^2 > 0$, the sign of $\mathcal{F}'(\lambda)$ is consistent with that of $\mathcal{G}(\lambda)$.

Using the discriminant of a cubic function $\Delta = b^2 - 3ac$ (a general cubic function is expressed as $a\lambda^3 + b\lambda^2 + c\lambda + d$),

$$\begin{aligned}\Delta &= (3a_1 + a_2a_5)^2 - 3(2a_1a_5 \cdot 2a_2) \\ &= 9a_1^2 + a_2^2a_5^2 + 6a_1a_2a_5 - 12a_1a_2a_5 \\ &= (3a_1 - a_2a_5)^2 \geq 0.\end{aligned} \tag{20}$$

The derivative of $\mathcal{G}(\lambda)$ is given by

$$\mathcal{G}'(\lambda) = 6a_1a_5\lambda^2 + 2(a_2a_5 + 3a_1)\lambda + 2a_2. \tag{21}$$

It is evident that when $\lambda > 0$, $\mathcal{G}'(\lambda) > 0$.

If $\mathcal{F}(\lambda)$ has an extremum, then $\mathcal{F}'(\lambda)$ has a root, meaning $\mathcal{G}(\lambda)$ has a root. Substituting $\lambda = 0$, we have $\mathcal{G}(0) = a_3 - a_4 a_5$, evidently when $a_3 < a_4 a_5$, $\mathcal{G}(\lambda)$ has a root.

Let $\lambda = x - \frac{a_2 a_5 + 3a_1}{6a_1 a_5}$, replacing λ with x in $\mathcal{G}(\lambda)$ yields

$$\mathcal{G}(x) = x^3 + a_6 x + a_7, \tag{22}$$

where

- $a_6 = \frac{-(3a_1 - a_2 a_5)^2}{12 a_1^2 a_5^2}$,
- $a_7 = \frac{108 a_1^2 a_5^2 (a_3 - a_4 a_5) - 36(a_1 a_2 a_5)(3a_1 + a_2 a_5) + 2(3a_1 + a_2 a_5)^3}{216 a_1^3 a_5^3}$.

Using the root-finding formula to calculate $\mathcal{G}(x) = 0$ gives the solutions x_1, x_2, x_3.

$$x_1 = \sqrt[3]{-\frac{a_7}{2} + \sqrt{\frac{a_7^2}{4} + \frac{a_6^3}{27}}} + \sqrt[3]{-\frac{a_7}{2} - \sqrt{\frac{a_7^2}{4} + \frac{a_6^3}{27}}},$$

$$x_2 = \omega \sqrt[3]{-\frac{a_7}{2} + \sqrt{\frac{a_7^2}{4} + \frac{a_6^3}{27}}} + \omega^2 \sqrt[3]{-\frac{a_7}{2} - \sqrt{\frac{a_7^2}{4} + \frac{a_6^3}{27}}},$$

$$x_3 = \omega^2 \sqrt[3]{-\frac{a_7}{2} + \sqrt{\frac{a_7^2}{4} + \frac{a_6^3}{27}}} + \omega \sqrt[3]{-\frac{a_7}{2} - \sqrt{\frac{a_7^2}{4} + \frac{a_6^3}{27}}}.$$

where imaginary number $\omega = \frac{-1 + \sqrt{3}\mathbf{i}}{2}$, $\lambda_l = x_l - \frac{a_2 a_5 + 3a_1}{6a_1 a_5}$, $l \in \{1, 2, 3\}$.

Furthermore, since $\lambda_1 \lambda_2 \lambda_3 = -\frac{a_3 - a_4 a_5}{2a_1 a_5} > 0$, λ has only one positive real root. If the equation has m real roots, the optimal solution for λ is $\lambda^* = \max_{\mathrm{m}}(\lambda_{\mathrm{m}})$.

In conclusion, $\mathcal{F}'(\lambda) \geq 0 \iff G(\lambda) \geq 0 \iff \lambda \geq \lambda^*$, therefore $\mathcal{F}(\lambda)$ has a minimum value $\mathcal{F}(\lambda^*)$, and $P(\lambda)$ has a minimum value $P(\lambda^*)$.

6 Privacy Analysis

At the beginning of privacy analysis, we clarify that: (1) FedFDP does not assume Secure Aggregation (SecAgg) [4], allowing the server to observe individual client uploads; (2) we guarantee *sample-level DP* via per-sample clipping to protect individual data records; and (3) SecAgg is orthogonal to our method—while it hides per-client updates, it cannot replace DP in preventing the aggregated model from memorizing sensitive data.

As Algorithm 2 shown, the i-th client will return model parameters $\mathbf{w}_{t+1}^i$ and loss value $\tilde{F}_i(\mathbf{w}_{t+1}^i)$ before the server aggregates the model parameters in round $t+1$. $\mathbf{w}_{t+1}^i$ and $\tilde{F}_i(\mathbf{w}_{t+1}^i)$ both access the private training data of client i, so they are both to perform differential privacy preservation, as in lines 11-14 and lines 17-20 of the Algorithm 2, respectively. We will analyze their privacy with RDP separately and finally combine their privacy loss to get the overall privacy loss of FedFDP algorithm.

6.1 Privacy Loss of $\mathbf{w}_{t+1}^{i}$

Theorem 2. *After T rounds local updates, the RDP of the $\mathbf{w}_T^i$ in i-th client satisfies:*

$$R_{model}^{i}(\alpha) = \frac{T}{\alpha - 1} \sum_{k=0}^{\alpha} \binom{\alpha}{k} (1-q)^{\alpha-k} q^k \exp\left(\frac{k^2 - k}{2\sigma^2}\right), \tag{23}$$

where σ is noise multiplier of the $\mathbf{w}_{t+1}^i$, and $\alpha > 1$ is the order.

Proof. We will prove Theorem 2 in the following two steps: (i) use the RDP of the sampling Gaussian mechanism to calculate the privacy cost of each model update, and (ii) use the composition of RDP mechanisms to compute the privacy cost of multiple model updates.

Definition 3. *(RDP privacy budget of SGM [33]). Let $SG_{q,\sigma}$, be the Sampled Gaussian Mechanism for some function f. If f has sensitivity 1, $SG_{q,\sigma}$ satisfies (α, R)-RDP whenever*

$$R \leq \frac{1}{\alpha - 1} \log \max(A_\alpha(q,\sigma), B_\alpha(q,\sigma)), \tag{24}$$

where

$$\begin{cases} A_\alpha(q,\sigma) = \mathbb{E}_{z\sim\vartheta_0}\left[(\vartheta(z)/\vartheta_0(z))^\alpha\right], \\ B_\alpha(q,\sigma) = \mathbb{E}_{z\sim\vartheta}\left[(\vartheta_0(z)/\vartheta(z))^\alpha\right]. \end{cases} \tag{25}$$

with $\vartheta_0 = \mathcal{N}\left(0, \sigma^2\right), \vartheta_1 = \mathcal{N}\left(1, \sigma^2\right)$ and $\vartheta = (1-q)\vartheta_0 + q\vartheta_1$.

Further, it holds for $\forall(q,\sigma) \in (0,1], \mathbb{R}^{+*}, A_\alpha(q,\sigma) \geq B_\alpha(q,\sigma)$. Thus, $SG_{q,\sigma}$ satisfies $\left(\alpha, \frac{1}{\alpha-1}\log\left(A_\alpha(q,\sigma)\right)\right)$-RDP .

Finally, the existing work [33] describes a procedure to compute $A_\alpha(q,\sigma)$ depending on integer α:

$$A_\alpha = \sum_{k=0}^{\alpha} \binom{\alpha}{k} (1-q)^{\alpha-k} q^k \exp\left(\frac{k^2 - k}{2\sigma^2}\right). \tag{26}$$

Definition 4. *(Composition of RDP [32]). For two randomized mechanisms f, g such that f is (α, R_1)-RDP and g is (α, R_2)-RDP the composition of f and g which is defined as (X, Y)(a sequence of results), where $X \sim f$ and $Y \sim g$, satisfies $(\alpha, R_1 + R_2) - RDP$*

From Definition 3 and Definition 4, the Theorem 2 is obtained.

6.2 Privacy Loss of $\tilde{F}_i(\mathbf{w}_{t+1}^i)$

Theorem 3. *After T rounds local updates, the RDP of the $\tilde{F}_i(\mathbf{w}_{t+1}^i)$ in i-th client satisfies:*

$$R_{loss}^{i}(\alpha) = \frac{T}{\alpha - 1} \sum_{k=0}^{\alpha} \binom{\alpha}{k} (1-q)^{\alpha-k} q^k \exp\left(\frac{k^2 - k}{2\sigma_l^2}\right), \tag{27}$$

where σ_l is noise multiplier of the $\tilde{F}_i(\mathbf{w}_{t+1}^i)$, and $\alpha > 1$ is the order.

The proof is similar to that of the $\mathbf{w}_{t+1}^i$, so we omit it.

6.3 Privacy Loss of FedFDP

Since both $\mathbf{w}_T^i$ and $\tilde{F}_i(\mathbf{w}_{t+1}^i)$ access the training set, we need to combine their RDP sequentially using Definition 4, and then use Lemma 1 to convert it to (ϵ, δ)-DP. Lastly, we can get the Privacy loss of FedFDP as follows.

Theorem 4. *(**Privacy loss of FedFDP**). The privacy loss in i-th client of FedFDP satisfies:*

$$\begin{aligned}(\epsilon^i, \delta^i) =& (R_{model}^i(\alpha) + R_{loss}^i(\alpha) + \ln((\alpha-1)/\alpha) \\ & - (\ln\delta + \ln\alpha)/(\alpha-1), \delta),\end{aligned} \tag{28}$$

where $0 < \delta < 1$, $R_{model}^i(\alpha)$ is the RDP of w_{t+1}^i is computed by Theorem 2, and $R_{loss}^i(\alpha)$ is the RDP of $\tilde{F}_i(\mathbf{w}_{t+1}^i)$ which is computed by Theorem 3.

Lemma 1. *(**Conversion from RDP to DP [3]**). if a randomized mechanism $f : \mathcal{X}^n \to \mathbb{R}^d$ satisfies (α, R)-RDP , then it satisfies$(R+\ln((\alpha-1)/\alpha)-(\ln\delta+\ln\alpha)/(\alpha-1), \delta)$-DP for any $0 < \delta < 1$.*

7 Experiments

In this section, we demonstrate the effectiveness of FedFair/FedFDP through answering the following three research questions:

- **RQ1** How effective is FedFair / FedFDP under classification tasks with real-world datasets?
- **RQ2** How robust is FedFair / FedFDP across different experimental settings?
- **RQ3** What is the performance of FedFDP under differential hyper-parameters?

To address the aforementioned three RQs, Section 7.1 introduces the default experimental settings used in this study, including the description of baselines and the configuration of the experimental code. Section 7.2 presents comparative experiments conducted on three datasets under default parameters involving FedFair, FedFDP, and six baseline methods to answer RQ1. Section 7.3 addresses RQ2 through experiments on heterogeneity and scalability. Finally, Section 7.4 performs ablation studies on λ, C, σ, ϵ, and the target accuracy to answer RQ3.

7.1 Default Settings

Prior to addressing the three research questions, an outline of the default experimental configuration is provided.

Baselines. We compared our algorithms against six baseline methods. For FedFair, the baseline algorithms were left unmodified; for FedFDP, DP was applied to each baseline algorithm to enable a fair comparison, which include FedAvg [30], SCAFFOLD [17], FedProx [24], FedDyn [2], ALI-DPFL [28] and q-FFL [25].

Communication Overhead Analysis. Table 1 presents a detailed comparison of the communication complexity per round for FedFDP and other state-of-the-art baselines.

Table 1. Comparison of Communication Overhead and Transmission Requirements. d represent the dimension of model.

Algorithm	Comm. Cost (Per Round)	Extra Transmitted Variables	Description of Overhead
FedAvg	$\mathcal{O}(d)$	None	Transmits only model parameters/gradients.
FedProx	$\mathcal{O}(d)$	None	Adds proximal term locally.
q-FFL	$\mathcal{O}(d)$	Scalar	Sends scalar loss for re-weighting.
SCAFFOLD	$\mathcal{O}(2d)$	Control Variates	Transmits both model and control variates.
FedDyn	$\mathcal{O}(d)$	None	Improve SCAFFOLD without control variables.
ALI-DPFL	$\mathcal{O}(d)$	Scalar	Server broadcast scalar local steps to clients.
FedFDP	$\mathcal{O}(d)$	Scalar	Transmits scalar loss for fairness/privacy.

While methods like SCAFFOLD introduce control variates to mitigate client drift, they double the communication payload to $\mathcal{O}(2d)$, which can be prohibitive in bandwidth-constrained wireless networks. In contrast, FedFDP maintains a communication complexity of $\mathcal{O}(d)$, aligning with efficient baselines such as FedAvg and FedDyn . Although FedFDP incorporates a fairness-aware mechanism and differential privacy, the additional transmission overhead is restricted to a negligible scalar value (representing the fairness loss), similar to the approach in q-FFL and ALI-DPFL. This demonstrates that FedFDP achieves robust privacy and fairness without compromising communication efficiency.

Tasks setting. By presenting the comparative experimental results of FedFair and FedFDP with multiple baseline methods on three datasets: MNIST [19], FashionMNIST [40], and CIFAR10 [18], we adopted a widely used heterogeneous settings [21, 27] to 10 clients which controlled by a Dirichlet distribution denoted as $\mathrm{Dir}(\beta)$, and the default value of $\beta = 0.1$ [27, 38]. We use a 4-layer CNN architecture [30] which consists of two convolutional layers and two fully connected layers as model architecture.

Implementation environment. We implement our experiment using PyTorch-1.8 and run all experiments on a server with one Intel i9 13900ks CPU (24 cores), 64GB of memory, and one NVIDIA 4090 GPU, running on Windows 10.

Hyperparameters. For FedFair and its baseline algorithms, we set learning rate $\eta = 0.1$. For FedFDP and its baseline algorithms, we set the $\eta = 1.0$, the batch sample ratio $q = 0.05$, clipping bound for gradient $C = 0.1$, the noise multipliers for gradient $\sigma = 2.0$, the privacy budget $\epsilon = 3.52$ and the $\delta = 1.0 \times 10^{-5}$. In particular, for the loss values in the FedFDP algorithm that additionally require differential privacy processing, we set the clipping bound for loss $C_l = 2.5$ and the noise multipliers for loss $\sigma_l = 5.0$.

7.2 Effective in Real-world Datasets

To address **RQ1**, we conducted experiments on three datasets for both FedFair and FedFDP under the default parameter settings described in Section 7.1. The results, presented in Table 2 and Table 3, report the test accuracy ($\%$) and fairness measure (Ψ).

Table 2 presents the experimental results of the FedFair algorithm compared to baseline methods. FedFair significantly enhances fairness while maintaining substantial

Table 2. FedFair and baselines: results of test accuracy (%) and fairness (Ψ).

Datasets	MNIST		FashionMNIST		CIFAR10	
	Acc.(↑)	Ψ(↓)	Acc.(↑)	Ψ(↓)	Acc.(↑)	Ψ(↓)
FedAvg	98.67±0.09	7.3e-3±5.4e-3	87.05±1.02	3.6e-1±2.2e-1	62.03±1.00	1.1e0±8.9e-1
SCAFFOLD	98.45±0.03	3.0e-2±2.6e-2	86.77±0.25	6.1e0±4.0e0	62.21±1.37	9.9e0±1.1e1
FedProx	98.70±0.13	7.5e-3±5.1e-3	87.30±0.51	3.5e-1±2.1e-1	62.13±0.20	1.2e0±3.1e-1
FedDyn	98.40±0.23	8.0e-3±2.1e-3	87.48±0.31	6.2e-1±3.5e-1	62.33±1.15	3.2e0±8.8e-1
q-FFL	98.72±0.03	4.7e-3±1.0e-3	87.35±0.13	5.1e-1±5.2e-2	**63.33±0.21**	9.4e-1±3.3e-2
FedFair	**98.75±0.09**	**3.9e-3±1.3e-3**	**87.70±0.76**	**2.6e-1±2.3e-1**	62.38±0.88	**8.5e-1±6.6e-1**

Table 3. FedFDP and baselines: results of test accuracy (%) and fairness (Ψ), while the privacy budget $\epsilon = 3.52$.

Datasets	MNIST		FashionMNIST		CIFAR10	
	Acc.(↑)	Ψ(↓)	Acc.(↑)	Ψ(↓)	Acc.(↑)	Ψ(↓)
FedAvg	93.40±0.47	1.1e11±0.5e10	84.15±1.97	3.0e8±1.4e8	52.61±0.17	6.1e9±5.2e9
SCAFFOLD	93.95±1.39	7.0e10±4.9e9	83.41±4.87	3.3e9±4.7e9	53.85±1.16	6.2e9±1.3e9
FedProx	90.50±3.95	5.3e10±7.3e9	83.85±0.98	7.9e8±4.5e8	51.34±0.91	4.4e9±2.1e9
FedDyn	91.42±2.25	5.6e10±9.8e9	84.04±1.21	8.8e8±3.2e8	52.14±1.39	5.5e9±1.2e9
ALI-DPFL	90.89±1.65	3.3e10±8.3e9	83.65±1.68	6.6e8±3.5e8	52.05±1.26	3.6e9±1.6e9
q-FFL	93.74±1.41	7.8e10±1.1e11	83.13±2.74	4.2e9±2.6e9	48.46±1.00	4.7e9±1.9e9
FedFDP	**95.13±0.83**	**2.3e10±1.0e10**	**85.99±0.76**	**2.8e8±0.8e8**	**54.21±0.98**	**2.6e9±1.6e9**

model performance across the MNIST, FashionMNIST, and CIFAR10 datasets, with respective increases of 17.0%, 25.7%, and 9.6%. A plausible explanation is that, as the training process progresses, FedFair increasingly prioritizes fairness, resulting in a reduction in η_i. Compared to training methods with a fixed learning rate, learning rate decay accelerate the convergence of gradient-based methods [45].

Table 3 presents the experimental results of the FedFDP algorithm compared to baseline methods. In comparison with Table 2, the introduction of DP leads to a slight decrease in accuracy, approximately in the range of 3% - 10%. However, the fairness issue becomes more pronounced, with the corresponding metric increasing significantly in magnitude, indicating that DP exerts a substantial influence on fairness in FL. Under a fixed privacy budget of $\epsilon = 3.52$, FedFDP significantly enhances fairness while matching the accuracy of the best-performing baseline, achieving improvements of 30.3%, 6.7%, and 27.8% across three distinct datasets, respectively.

7.3 Robust in Different Settings

To answer **RQ2**, as shown in Table 4 and Table 5, we extended the heterogeneity setting from the default $\mathrm{Dir}(0.1)$ to $\mathrm{Dir}(0.5)$ and $\mathrm{Dir}(1)$, and scaled the number of clients from the default 10 to 20 and 50 to examine scalability.

Heterogeneity. Table 4 shows that FedFair outperforms baseline methods in accuracy while maintaining significant fairness when faced with diverse data distributions

Table 4. FedFair and baselines: results of test accuracy (%) and fairness (Ψ) in different heterogeneity and scalability.

	Heterogeneity						Scalability					
Datasets	MNIST				FashionMNIST		CIFAR10					
Method	$\mathrm{Dir}(0.1)$		$\mathrm{Dir}(0.5)$		$\mathrm{Dir}(1)$		10 clients		20 clients		50 clients	
	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$
FedAvg	98.67	7.3e-3	98.37	1.2e-3	89.41	3.6e-2	62.03	1.1e0	60.89	1.5e0	59.78	1.5e0
SCAFFOLD	98.45	5.0e-3	98.51	2.1e-3	86.39	3.4e-2	62.21	9.9e0	59.48	8.7e0	58.64	2.3e0
FedProx	98.70	7.5e-3	98.35	1.2e-3	88.33	3.5e-2	62.13	1.2e0	60.33	2.2e0	61.31	1.6e0
FedDyn	98.40	8.0e-3	98.36	1.8e-3	88.26	3.5e-2	62.33	9.4e-1	61.35	2.6e0	62.52	1.5e0
q-FFL	98.72	4.7e-3	98.61	**1.1e-3**	89.03	3.6e-2	**63.33**	9.4e-1	60.63	1.3e0	59.65	1.5e0
FedFair	**98.75**	**3.9e-3**	**98.72**	1.2e-3	**89.60**	**1.7e-2**	62.38	**8.5e-1**	**61.58**	**1.1e0**	**62.90**	**1.5e0**

Table 5. FedFDP and baselines: results of test accuracy (%) and fairness (Ψ) in different heterogeneity and scalability, while the privacy budget $\epsilon = 3.52$.

	Heterogeneity						Scalability					
Datasets	MNIST				FashionMNIST		CIFAR10					
Method	$\mathrm{Dir}(0.1)$		$\mathrm{Dir}(0.5)$		$\mathrm{Dir}(1)$		10 clients		20 clients		50 clients	
	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$
FedAvg	93.40	1.1e11	94.17	4.1e9	85.60	3.7e9	52.61	6.1e9	51.58	2.0e9	55.90	1.5e9
SCAFFOLD	93.95	7.0e10	94.06	1.9e9	84.56	3.6e9	53.85	6.2e9	54.34	2.9e8	53.98	2.2e9
FedProx	90.50	5.3e10	94.23	3.1e9	84.38	5.6e9	51.34	4.4e9	53.67	4.5e8	54.14	2.2e9
FedDyn	91.42	5.6e10	93.08	3.8e9	85.21	9.8e10	52.14	5.5e9	50.21	1.2e10	51.86	2.0e10
ALI-DPFL	90.89	3.3e10	93.98	2.6e9	83.66	2.5e10	52.05	3.6e9	38.35	2.4e10	33.46	1.6e10
q-FFL	93.74	7.8e10	93.86	**2.5e8**	85.20	7.4e8	48.46	4.7e9	51.12	2.7e8	49.83	**1.1e7**
FedFDP	**94.13**	**2.3e10**	**94.56**	3.4e9	**86.77**	**6.0e8**	**54.21**	**2.6e9**	**54.52**	**2.2e8**	**56.49**	5.3e8

exhibiting varying levels of heterogeneity. Except for the $\mathrm{Dir}(0.5)$ scenario, where it falls slightly short of q-FFL, FedFair demonstrated fairness improvements of 22.0% and 50% over the best baseline in $\mathrm{Dir}(0.1)$ and $\mathrm{Dir}(1.0)$, respectively. Table 5 shows that FedFDP outperforms baseline methods in accuracy while maintaining significant fairness when confronted with diverse data distributions exhibiting varying levels of heterogeneity. Except for the $\mathrm{Dir}(0.5)$ scenario, where it slightly underperforms q-FFL, FedFDP shows fairness improvements of 30.3% and 18.9% over the best baseline in $\mathrm{Dir}(0.1)$ and $\mathrm{Dir}(1.0)$, respectively.

Scalability. Table 4 indicates that FedFair generally surpasses baselines in accuracy and shows fairness improvements of 9.6%, 15.4%, and 0.01% across three client counts. Table 5 demonstrates that as the number of clients increases, FedFDP consistently achieves accuracy comparable to the best baseline. Moreover, except in the 50 clients scenario where it slightly lags behind q-FFL in fairness, FedFDP exhibits fairness improvements of 27.8% and 18.5% with 10 and 20 clients, respectively.

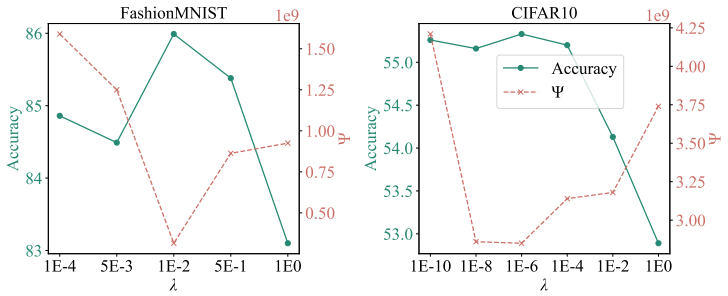

Fig. 2. Impact of different λ in FedFDP algorithm.

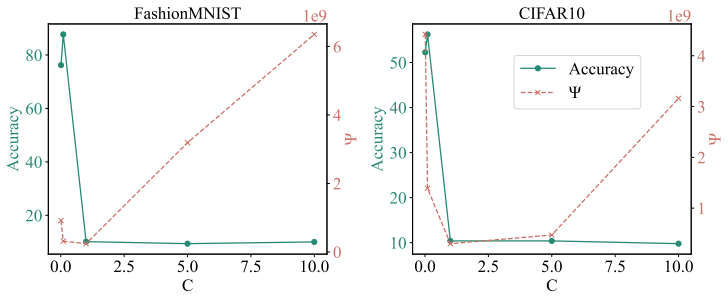

Fig. 3. Impact of different C in FedFDP algorithm.

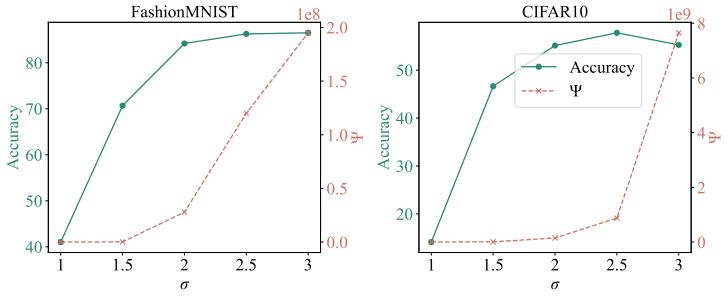

Fig. 4. Impact of different σ in FedFDP algorithm.

7.4 Performance under Different Hyper-parameters

In response to **RQ3**, we performed extensive hyper-parameter experiments. In Table 6, we observed results under fixed privacy budgets $\epsilon = \{1.0, 2.0, 3.0, 4.0\}$. Table 7 presents experiments aiming at fixed target accuracies $\{80\%, 85\%, 90\%, 95\%\}$. Fig. 2 explores the existence of an optimal λ, with results consistent with those in Section 5. The influence of the clipping norm C is investigated in Fig. 3, and the effect of the noise multiplier σ is examined in Fig. 4.

Table 6. The average accuracy (%) and fairness (Ψ) on different privacy budget (ϵ) setting at FashionMNIST dataset.

ϵ	1.0		2.0		3.0		4.0	
	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$	Acc.(↑)	$\Psi(\downarrow)$
FedAvg	61.68	1.1e6	82.69	2.3e7	84.07	3.8e8	86.83	8.0e8
SCAFFOLD	62.12	1.2e6	82.36	2.3e7	83.39	2.6e9	85.41	5.6e8
FedProx	61.25	1.1e6	81.62	2.3e7	85.32	3.8e8	85.68	2.4e9
FedDyn	62.38	1.5e6	82.43	3.1e7	83.29	2.9e8	84.51	5.2e9
ALI-DPFL	61.78	1.1e6	83.01	2.3e7	84.12	3.9e8	85.58	4.5e9
q-FFL	58.99	3.6e6	80.24	9.6e7	82.92	5.5e8	85.01	3.8e9
FedFDP	**63.36**	**1.0e6**	**83.15**	**2.1e7**	**85.59**	**9.5e7**	**86.97**	**3.8e8**

Impact of Different λ. In Section 5.1, we derive a closed-form solution for λ^*, which, however, relies on constants or upper bounds (e.g., L, μ, G, Γ) whose specific values change with different models, datasets, and loss functions. Therefore, we conduct the experiment in Fig. 2 to investigate the practical range of λ^*. We investigated the correlation between λ and Ψ in the FedFDP algorithm. As depicted in Fig. 2, our findings align with the theoretical analysis: an initial increase in λ leads to a decrease in Ψ, followed by an increase, indicating the existence of an optimal λ.

Impact of Different C. We emphasized that excessively large or small values of C cannot ensure both performance and fairness simultaneously. As illustrated in Fig. 3, we examined the correlation between C, performance, and fairness across two datasets, with C taking values in the range $\{0.01, 0.1, 1, 5, 10\}$. Although Ψ reaches its minimum when $C = 1$, the model does not converge due to excessive noise addition. As C continues to increase, the model's performance further deteriorates, resulting in an increase in Ψ. When $C = 0.1$, both accuracy and Ψ achieve relatively good performance.

Impact of Different σ. We tested the variations in accuracy and fairness on two datasets when σ was set to $\{1, 1.5, 2, 2.5, 3\}$, under the condition of $\epsilon = 2$ and $q = 0.05$, which supports $T \in \{6, 115, 268, 463, 708\}$ rounds of communication. As shown in Fig. 4, an increase in σ allows more training epochs to advance the model's convergence process, but when $\sigma = 3$, excessive noise hinders the convergence. When σ is smaller, the model performance is relatively similar, with a lower Ψ value. As σ gradually increases, the performance differences between models become apparent, and the Ψ value increases.

Impact of Different ϵ. We conducted experiments with $\epsilon \in \{1, 2, 3, 4\}$ on the FashionMNIST dataset, as detailed in Table 6. For $\epsilon \in \{1, 2, 3\}$, we allowed larger values of ϵ to support more iterations. At $\epsilon = 4$, we supported a smaller noise multiplier $\sigma = 1.65$ with a similar number of iterations as at $\epsilon = 3$. The experimental results demonstrate that FedFDP consistently achieves performance on par with the best baseline in terms of accuracy across various privacy budgets. In terms of fairness, the improvements are 9.1%, 8.7%, 67.2%, and 32.1%, respectively.

ϵ at Different Target Accuracy. TABLE 7 presents experiments on MNIST under the $\mathbf{Dir}(0.05)$ partition, reporting the privacy budget ϵ and fairness measure Ψ at

Table 7. The privacy budget (ϵ) and fairness (Ψ) on different target accuracy setting at MNIST Dir(0.05).

Acc.	80%		85%		90%		95%	
	$\epsilon(\downarrow)$	$\Psi(\downarrow)$	$\epsilon(\downarrow)$	$\Psi(\downarrow)$	$\epsilon(\downarrow)$	$\Psi(\downarrow)$	$\epsilon(\downarrow)$	$\Psi(\downarrow)$
FedAvg	2.43	2.10e9	3.02	8.90e9	3.45	1.30e10	4.89	1.30e11
SCAFFOLD	2.55	3.30e9	3.15	1.10e10	3.67	3.70e10	5.32	1.10e11
FedProx	3.02	1.30e9	2.98	1.30e10	4.31	5.80e10	4.89	2.50e11
FedDyn	2.68	2.20e9	3.55	7.80e9	3.98	1.10e10	5.15	3.30e11
ALI-DPFL	2.85	3.10e9	2.98	1.30e10	3.67	2.30e10	5.32	9.80e10
q-FFL	2.68	9.80e8	3.15	9.70e9	4.12	8.80e10	4.04	1.02e11
FedFDP	**2.25**	**6.60e8**	**2.70**	**1.20e9**	**3.05**	**5.80e9**	**3.77**	**2.80e10**

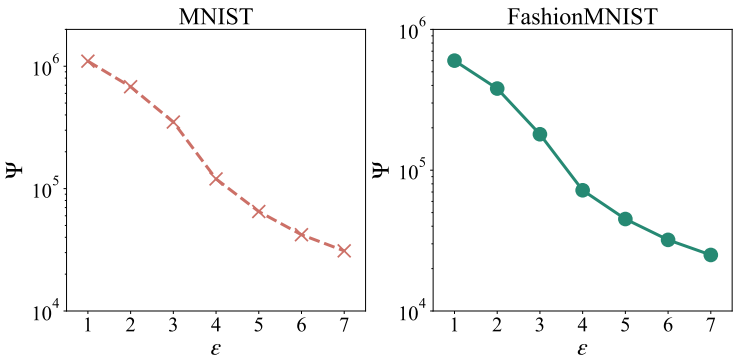

Fig. 5. Tradeoff between privacy and fairness on 60% accuracy.

the point of achieving accuracy targets $\{80\%, 85\%, 90\%, 95\%\}$. As the target accuracy increases (from 80% to 95%), the privacy budget (ϵ) for most methods rises slightly (e.g., FedAvg increases from 2.43 to 3.02), while fairness (Ψ) deteriorates significantly (e.g., FedAvg rises from 2.10×10^9 to 1.3×10^{11}), highlighting the intensified trade-off between privacy and fairness under higher accuracy demands. FedFDP demonstrates exceptional performance in balancing these objectives: At the high 95% accuracy target, its ϵ (2.70) is lower than all other methods, and its Ψ (1.20×10^9) is significantly lower than competitors by an order of magnitude. At the 80% accuracy target, FedFDP achieves the lowest Ψ (6.60×10^8) and the lowest ϵ (2.25) in the entire table, validating its comprehensive superiority.

Tradeoff between Privacy and Fairness. With the target accuracy fixed at 60% on both MNIST and FashionMNIST, we investigate the trade-off between fairness and privacy. As shown in Fig. 5, as the privacy budget ϵ increases, the fairness metric Ψ decreases significantly on both datasets, indicating a clear trade-off between privacy and fairness under the given accuracy constraint. The noise introduced by the privacy mechanism, while protecting data, amplifies performance disparities across groups, whereas

relaxing privacy constraints creates room for fairness optimization. This monotonically decreasing relationship reveals the tripartite trade-off characteristics among privacy, fairness, and accuracy.

8 Limitations and Future Work

Despite the promising results achieved by FedFDP in balancing fairness, privacy, and utility, there are several limitations in the current study that open avenues for future research.

First, regarding heterogeneity, our current evaluation primarily focuses on statistical heterogeneity (Non-IID data) simulated via Dirichlet partitioning. We have not yet deeply explored *model heterogeneity*, where clients may possess different network architectures due to varying storage or memory capacities. Furthermore, the challenges associated with *cross-domain* data—where feature spaces or distributions differ fundamentally across clients—remain unaddressed. Future work could investigate adapting the fairness-aware gradient clipping strategy to support heterogeneous model architectures and cross-domain federated learning scenarios.

Second, the current framework assumes that all selected clients participate successfully in the training process (full participation). However, in practical real-world deployments, *device heterogeneity* (e.g., varying computational power) and unstable network conditions (e.g., bandwidth limits and high latency) often lead to the problem of *stragglers*. These stragglers can significantly delay the synchronization process in the global aggregation phase. In this work, we have not specifically optimized FedFDP for such system-level challenges. Future iterations of FedFDP could explore asynchronous update mechanisms or robust client selection protocols to mitigate the impact of stragglers and communication dropouts.

9 Conclusion

In this paper, a FedFair algorithm is initially proposed to effectively address fairness concerns by optimizing a novel local loss function. The FedFDP, building upon FedFair, is introduced to incorporate a fairness-aware gradient clipping strategy and an adaptive clipping method for additional loss values, thereby achieving both fairness and differential privacy protection. Then, we find an optimal fairness parameter λ^* through convergence analysis and numerical analysis methods, striking a balance between model performance and fairness. Subsequently, a comprehensive privacy analysis of the approach is conducted using RDP. Through extensive experiments, the results indicate that FedFair and FedFDP significantly outperform state-of-the-art solutions in terms of model performance and fairness. It is believed that our work contributes a valuable FL framework for addressing fairness and privacy challenges.

Acknowledge

This work is supported by National Natural Science Foundation of China Key Program (Grant No. 62132005), Natural Science Foundation of Shanghai (Grant No. 22ZR1419100), and CAAI-Huawei MindSpore Open Fund (Grant No. CAAIXSJLJJ-2022-005A).

References

1. Abadi, M., Chu, A., Goodfellow, I.J., McMahan, H.B., Mironov, I., Talwar, K., Zhang, L.: Deep learning with differential privacy. CoRR **abs/1607.00133** (2016), http://arxiv.org/abs/1607.00133
2. Acar, D.A.E., Zhao, Y., Navarro, R.M., Mattina, M., Whatmough, P.N., Saligrama, V.: Federated learning based on dynamic regularization. arXiv preprint arXiv:2111.04263 (2021)
3. Balle, B., Barthe, G., Gaboardi, M., Hsu, J., Sato, T.: Hypothesis testing interpretations and renyi differential privacy. In: International Conference on Artificial Intelligence and Statistics. pp. 2496–2506. PMLR (2020)
4. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H.B., Patel, S., Ramage, D., Segal, A., Seth, K.: Practical secure aggregation for privacy-preserving machine learning. In: proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. pp. 1175–1191 (2017)
5. Bu, Z., Wang, Y., Zha, S., Karypis, G.: Automatic clipping: Differentially private deep learning made easier and stronger. CoRR **abs/2206.07136** (2022). https://doi.org/10.48550/arXiv.2206.07136, https://doi.org/10.48550/arXiv.2206.07136
6. Chen, X., Xu, G., Xu, X., Jiang, H., Tian, Z., Ma, T.: Multicenter hierarchical federated learning with fault-tolerance mechanisms for resilient edge computing networks. IEEE Transactions on Neural Networks and Learning Systems (2024)
7. Cong, M., Yu, H., Weng, X., Yiu, S.M.: A game-theoretic framework for incentive mechanism design in federated learning. Federated Learning: Privacy and Incentive pp. 205–222 (2020)
8. Cotter, A., Jiang, H., Gupta, M., Wang, S., Narayan, T., You, S., Sridharan, K.: Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. Journal of Machine Learning Research **20**(172), 1–59 (2019)
9. Ding, J., Zhang, X., Li, X., Wang, J., Yu, R., Pan, M.: Differentially private and fair classification via calibrated functional mechanism. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 622–629 (2020)
10. Dwork, C., Hardt, M., Pitassi, T., Reingold, O., Zemel, R.: Fairness through awareness. In: Proceedings of the 3rd innovations in theoretical computer science conference. pp. 214–226 (2012)
11. Dwork, C., Roth, A., et al.: The algorithmic foundations of differential privacy. Foundations and Trends® in Theoretical Computer Science **9**(3–4), 211–407 (2014)
12. Fu, J., Hong, Y., Ling, X., Wang, L., Ran, X., Sun, Z., Wang, W.H., Chen, Z., Cao, Y.: Differentially private federated learning: A systematic review. arXiv preprint arXiv:2405.08299 (2024)
13. Fu, J., Ye, Q., Hu, H., Chen, Z., Wang, L., Wang, K., Xun, R.: Dpsur: Accelerating differentially private stochastic gradient descent using selective update and release. arXiv preprint arXiv:2311.14056 (2023)
14. Gálvez, B.R., Granqvist, F., van Dalen, R., Seigel, M.: Enforcing fairness in private federated learning via the modified method of differential multipliers. In: NeurIPS 2021 Workshop Privacy in Machine Learning (2021)
15. Gu, X., Tianqing, Z., Li, J., Zhang, T., Ren, W., Choo, K.K.R.: Privacy, accuracy, and model fairness trade-offs in federated learning. Computers & Security **122**, 102907 (2022)
16. Jagielski, M., Kearns, M., Mao, J., Oprea, A., Roth, A., Sharifi-Malvajerdi, S., Ullman, J.: Differentially private fair learning. In: International Conference on Machine Learning. pp. 3000–3008. PMLR (2019)
17. Karimireddy, S.P., Kale, S., Mohri, M., Reddi, S., Stich, S., Suresh, A.T.: Scaffold: Stochastic controlled averaging for federated learning. In: International conference on machine learning. pp. 5132–5143. PMLR (2020)

18. Krizhevsky, A.: Learning multiple layers of features from tiny images (2009)
19. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
20. Lee, G., Jeong, M., Shin, Y., Bae, S., Yun, S.Y.: Preservation of the global knowledge by not-true distillation in federated learning. Advances in Neural Information Processing Systems **35**, 38461–38474 (2022)
21. Li, Q., He, B., Song, D.: Model-contrastive federated learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10713–10722 (2021)
22. Li, Q., Wen, Z., Wu, Z., Hu, S., Wang, N., Li, Y., Liu, X., He, B.: A survey on federated learning systems: Vision, hype and reality for data privacy and protection. IEEE Transactions on Knowledge and Data Engineering **35**(4), 3347–3366 (2021)
23. Li, T., Hu, S., Beirami, A., Smith, V.: Ditto: Fair and robust federated learning through personalization. In: International Conference on Machine Learning. pp. 6357–6368. PMLR (2021)
24. Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V.: Federated optimization in heterogeneous networks. Proceedings of Machine learning and systems **2**, 429–450 (2020)
25. Li, T., Sanjabi, M., Beirami, A., Smith, V.: Fair resource allocation in federated learning (2020), https://openreview.net/forum?id=ByexElSYDr
26. Li, X., Huang, K., Yang, W., Wang, S., Zhang, Z.: On the convergence of fedavg on non-iid data. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net (2020), https://openreview.net/forum?id=HJxNAnVtDS
27. Lin, T., Kong, L., Stich, S.U., Jaggi, M.: Ensemble distillation for robust model fusion in federated learning. Advances in Neural Information Processing Systems **33**, 2351–2363 (2020)
28. Ling, X., Fu, J., Wang, K., Liu, H., Chen, Z.: Ali-dpfl: Differentially private federated learning with adaptive local iterations. In: 2024 IEEE 25th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM). pp. 349–358. IEEE (2024)
29. Lyu, L., Xu, X., Wang, Q., Yu, H.: Collaborative fairness in federated learning. Federated Learning: Privacy and Incentive pp. 189–204 (2020)
30. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data pp. 1273–1282 (2017)
31. Melis, L., Song, C., Cristofaro, E.D., Shmatikov, V.: Exploiting unintended feature leakage in collaborative learning. ieee symposium on security and privacy (2022)
32. Mironov, I.: Rényi differential privacy. ieee computer security foundations symposium (2017)
33. Mironov, I., Talwar, K., Zhang, L.: Rényi differential privacy of the sampled gaussian mechanism. arXiv: Learning (2019)
34. Padala, M., Damle, S., Gujar, S.: Federated learning meets fairness and differential privacy. In: Neural Information Processing: 28th International Conference, ICONIP 2021, Sanur, Bali, Indonesia, December 8–12, 2021, Proceedings, Part VI 28. pp. 692–699. Springer (2021)
35. Song, C., Ristenpart, T., Shmatikov, V.: Machine learning models that remember too much pp. 587–601 (2017)
36. Stich, S.U.: Local sgd converges fast and communicates little. arXiv preprint arXiv:1805.09767 (2018)
37. Tran, C., Fioretto, F., Van Hentenryck, P.: Differentially private and fair deep learning: A lagrangian dual approach. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 9932–9939 (2021)
38. Wang, J., Liu, Q., Liang, H., Joshi, G., Poor, H.V.: Tackling the objective inconsistency problem in heterogeneous federated optimization. Advances in neural information processing systems **33**, 7611–7623 (2020)

39. Wang, S., Tuor, T., Salonidis, T., Leung, K.K., Makaya, C., He, T., Chan, K.: Adaptive federated learning in resource constrained edge computing systems. IEEE journal on selected areas in communications **37**(6), 1205–1221 (2019)
40. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017)
41. Yang, X., Zhang, H., Chen, W., Liu, T.: Normalized/clipped SGD with perturbation for differentially private non-convex optimization. CoRR **abs/2206.13033** (2022). https://doi.org/10.48550/arXiv.2206.13033, https://doi.org/10.48550/arXiv.2206.13033
42. Zhao, Z., Joshi, G.: A dynamic reweighting strategy for fair federated learning. In: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 8772–8776. IEEE (2022)
43. Zhu, L., Liu, Z., Han, S.: Deep leakage from gradients. Advances in neural information processing systems **32** (2019)
44. Zhu, Z., Hong, J., Zhou, J.: Data-free knowledge distillation for heterogeneous federated learning. In: International conference on machine learning. pp. 12878–12889. PMLR (2021)
45. Zinkevich, M.: Online convex programming and generalized infinitesimal gradient ascent pp. 928–936 (2003), http://www.aaai.org/Library/ICML/2003/icml03-120.php

A Appendix

In the beginning, Table 8 presents some notations utilized in this paper. Symbols that have not been previously introduced will be defined in subsequent sections.

Table 8. Summary of main notations

$F(\mathbf{w})$	Global loss function
$F_i(\mathbf{w})$	Local loss function for client i
T	Communication rounds
t	Communication round index
D_i	Dataset of client i
D	The union of D_i
$\mathbf{w}_t^i$	Local model at client i at round t
$\mathbf{w}^*$	Optimal model that minimizes $F(\mathbf{w})$
η	Gradient descent step size
λ	Fairness parameter
N	Total number of clients
q	batch sample ratio
$\|\mathcal{B}_i\|$	Batch size of client i, with the mathematical expectation equals to $\|D_i\| \cdot q$
p_i	Weight of client i, equals to $\|D_i\|/\|D\|$
ξ_j	Single data sample in D_i, indexed by j
σ	Noise multiplier for gradient
σ_l	Noise multiplier for loss
C	Original clipping bound for gradient
C_l	Original clipping bound for loss
$\|\|\cdot\|\|$	L_2-norm

A.1 Convergence Analysis

We analyze the convergence of Algorithm 2 and derive insights on selecting the hyper-parameter λ based on the convergence upper bound. Prior to proving this, we need to establish several assumptions:

Assumption 1 *$F_1, \cdots, F_N$ are all L-smooth: for all $\mathbf{v}$ and $\mathbf{w}$, $F_i(\mathbf{v}) \leq F_i(\mathbf{w}) + (\mathbf{v} - \mathbf{w})^\top \nabla F_i(\mathbf{w}) + \frac{L}{2}\|\mathbf{v} - \mathbf{w}\|^2$.*

Assumption 2 *$F_1, \cdots, F_N$ are all μ-strongly convex: for all $\mathbf{v}$ and $\mathbf{w}$, $F_i(\mathbf{v}) \geq F_i(\mathbf{w}) + (\mathbf{v} - \mathbf{w})^\top \nabla F_i(\mathbf{w}) + \frac{\mu}{2}\|\mathbf{v} - \mathbf{w}\|^2$.*

Assumption 3 *Let ξ_j be sampled from the i-th device's local data uniformly at random. The expected squared norm of stochastic gradients is uniformly bounded, i.e., $\left\|\nabla F_i\left(\mathbf{w}_t^i, \xi_j\right)\right\| \leq G$ for all $i \in [1, \cdots, N]$, $t \in [0, \cdots, T-1]$ and $j \in [1, \cdots, |\mathcal{B}_i|]$*

Theorem 5. *Under the assumptions mentioned above, we obtain the convergence upper bound for Algorithm 2 as follows:*

$$\mathbb{E}[F(\mathbf{w}_t)] - F^* \leq \frac{L}{2t}\left(\frac{A}{\mu^2(2C_t-1)} + \mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2\right), \tag{29}$$

where:

- $A = G^2C_t^3 + 3G^2C_t^2 + 2L\Gamma C_t + \frac{2\sigma^2C^2d}{\hat{B}^2}$,
- $\hat{B} = \min_i |\mathcal{B}_i|$,
- $\Gamma = F^* - \sum_{i=1}^N p_i F_i^*$,
- $C_t = \sum_{i=1}^N p_i C_t^i, C_t^i = \frac{1}{|\mathcal{B}_i|}\sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j}$.

The meaning of Γ is consistent with [26], where a larger value of Γ indicates that the data among different clients is more Non-IID.

proof sketch: We investigate the relationship between $\mathbb{E}\|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2$ and $\mathbb{E}\|\mathbf{w}_t - \mathbf{w}^*\|^2$, and then use mathematical induction to obtain an upper bound for $\mathbb{E}\|\mathbf{w}_t - \mathbf{w}^*\|^2$. Finally, by utilizing Assumption 1, we derive Equation (29). For the detailed proof, please refer to Appendix A.2.

A.2 Proof of Theorem 5

For the purpose of validation, we introduce an additional variable $\mathbf{v}_t^i$ to represent the immediate result of a single-step DPSGD update from $\mathbf{w}_t^i$. We interpret $\mathbf{w}_{t+1}^i$ as the parameter obtained after a single communication step. Consequently, the fair-clipping DPSGD in client i at iteration t transitions from Equation (12) to:

$$\mathbf{v}_{t+1}^i = \mathbf{w}_t^i - \frac{\eta}{|\mathcal{B}_i|}[\sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j) + \sigma C \cdot \mathcal{N}(0, \mathbf{I})], \tag{30}$$

where:

$$C_t^{i,j} = \min\left(1 + \lambda \cdot \Delta_i^j, \frac{C}{\|\nabla F_i(\mathbf{w}_t^i, \xi_j)\|}\right).$$

In our analysis, we define two virtual sequences $\mathbf{v}_t = \sum_{i=1}^N p_i \mathbf{v}_t^i$ and $\mathbf{w}_t = \sum_{i=1}^N p_i \mathbf{w}_t^i$, which is motivated by [36]. Therefore,

$$\mathbf{v}_{t+1} = \mathbf{w}_t - \sum_{i=1}^N p_i \frac{\eta}{|\mathcal{B}_i|}[\sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j) + \sigma C \cdot \mathcal{N}(0, \mathbf{I})] \tag{31}$$

Key Lemma

Lemma 2. *(Results of one iteration.)Assume Assumption 1-3 hold, we have:*

$$\mathbb{E}\|\mathbf{v}_{t+1} - \mathbf{w}^*\|^2 \leq (1 - \mu\eta C_t)\,\mathbb{E}\|\mathbf{w}_t - \mathbf{w}^*\|^2 + \eta^2 A,$$

where:

- $A = G^2C_t^3 + 3G^2C_t^2 + 2L\Gamma C_t + \frac{2\sigma^2C^2d}{\hat{B}^2}$,

- $\hat{B} = \min_i |\mathcal{B}_i|$,
- $\Gamma = F^* - \sum_{i=1}^N p_i F_i^*$,
- $C_t = \sum_{i=1}^N p_i C_t^i, C_t^i = \frac{1}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j}$.

Let $\Delta_t = \mathbb{E}\|\mathbf{w}_t - \mathbf{w}^*\|^2$. It is evident that we always have $\mathbf{w}_{t+1} = \mathbf{v}_{t+1}$. According to Lemma 2, this implies:

$$\Delta_{t+1} \leq (1 - \mu\eta C_t)\, \Delta_t + \eta^2 A$$

We use mathematical induction to obtain $\Delta_t \leq \frac{v}{t}$ where $v = \max\{\frac{\beta^2 A}{\mu\beta C_t - 1}, \Delta_1\}$, $\eta = \frac{\beta}{t}$ for some $\beta > \frac{1}{\mu}$.

STEP 1. When $t = 1$, the equation $\Delta_1 \leq v$ holds obviously.

STEP 2. We assume $\Delta_t \leq \frac{v}{t}$ holds.

STEP 3.

$$\Delta_{t+1} \leq \left(1 - \mu\frac{\beta}{t}C_t\right)\frac{v}{t} + \frac{\beta^2 A}{t^2} = \frac{t-1}{t^2}v + \left(\frac{\beta^2 A}{t^2} - \frac{\mu\beta C_t - 1}{t^2}v\right) \leq \frac{t-1}{t^2}v \leq \frac{v}{t+1}$$

Therefore, $\Delta_{t+1} \leq \frac{v}{t+1}$ holds, completing the proof by mathematical induction. Hence, $\Delta_t \leq \frac{v}{t}$ holds.

Then by the L-smoothness of $F(\cdot)$, let $\beta = \frac{2}{\mu}$, we get

$$\mathbb{E}[F(\mathbf{w}_t)] - F^* \leq \frac{L}{2}\Delta_t \leq \frac{L}{2t}v \leq \frac{L}{2t}\left(\frac{A}{\mu^2(2C_t - 1)} + \Delta_1\right)$$

Proof of Lemma 2 By the Equation (31), we get

$$\begin{aligned}
&\|\mathbf{v}_{t+1} - \mathbf{w}^*\|^2 \\
&= \|\mathbf{w}_t - \mathbf{w}^* - \sum_{i=1}^N p_i \frac{\eta}{|\mathcal{B}_i|}[\sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j) + \sigma C \cdot \mathcal{N}(0, \mathbf{I})]\|^2 \\
&= \|\mathbf{w}_t - \mathbf{w}^*\|^2 \\
&\underbrace{-2\langle \mathbf{w}_t - \mathbf{w}^*, \sum_{i=1}^N p_i \frac{\eta}{|\mathcal{B}_i|}[\sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \nabla F_i(\mathbf{w}_t^i, \xi_j) + \sigma C \mathcal{N}(0, \mathbf{I})]\rangle}_{\mathcal{A}_1} \\
&\underbrace{+ \left\| \sum_{i=1}^N p_i \frac{\eta}{|\mathcal{B}_i|}[\sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j) + \sigma C \cdot \mathcal{N}(0, \mathbf{I})] \right\|^2}_{\mathcal{A}_2}
\end{aligned}$$

Firstly, we process $\mathcal{A}_2$:

$$\mathcal{A}_2 \leq \underbrace{\left\| \sum_{i=1}^{N} p_i \frac{\eta}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j) \right\|^2}_{\mathcal{B}_1} + \underbrace{\sum_{i=1}^{N} p_i \frac{\eta}{|\mathcal{B}_i|} \left\langle \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j), \sigma C \cdot \mathcal{N}(0, \mathbf{I}) \right\rangle}_{\mathcal{B}_0} + \underbrace{\| \sum_{i=1}^{N} p_i \frac{\eta}{|\mathcal{B}_i|} \sigma C \cdot \mathcal{N}(0, \mathbf{I}) \|^2}_{\mathcal{B}_2}$$

Since $\mathbb{E}[\mathcal{B}_0] = 0$, we focus on $\mathcal{B}_1$ and $\mathcal{B}_2$:

$$\mathbb{E}[\mathcal{B}_2] \leq \frac{\eta^2}{\hat{B}^2} \sum_{i=1}^{N} \mathbb{E} \| \sigma C \mathcal{N}(0, \mathbf{I}) \|^2 \leq \frac{\eta^2 \sigma^2 C^2 d}{\hat{B}^2},$$

where $\frac{1}{\hat{B}^2} = \max_i \frac{1}{|\mathcal{B}_i|}$.

By the convexity of $\| \cdot \|^2$,

$$\mathcal{B}_1 \leq \eta^2 \sum_{i=1}^{N} p_i \left\| \frac{1}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j) \right\|^2,$$

taking exception and according to assumption 3:

$$\mathbb{E}[\mathcal{B}_1] \leq \eta^2 \mathbb{E} \left[\sum_{i=1}^{N} p_i \left\| C_t^i \nabla F_i(\mathbf{w}_t^i) \right\|^2 \right] \leq \eta^2 C_t^2 G^2,$$

where $C_t = \sum_{i=1}^{N} p_i C_t^i$, $C_t^i = \frac{1}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j}$.

Now, we obtain the bound for the expectation of $\mathcal{A}_2$:

$$\mathbb{E}[\mathcal{A}_2] \leq \eta^2 \left(\frac{\sigma^2 C^2 d}{\hat{B}^2} + G^2 C_t^2 \right)$$

The process of $\mathcal{A}_1$ show as below:

$$\mathcal{A}_1 = -2\langle \mathbf{w}_t - \mathbf{w}^*, \sum_{i=1}^{N} p_i \frac{\eta}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \nabla F_i(\mathbf{w}_t^i, \xi_j) \rangle \underbrace{-2\langle \mathbf{w}_t - \mathbf{w}^*, \sum_{i=1}^{N} p_i \frac{\eta}{|\mathcal{B}_i|} \sigma C \mathcal{N}(0, \mathbf{I}) \rangle}_{\mathcal{C}_0}$$

$$= \mathcal{C}_0 \underbrace{-2\sum_{i=1}^{N} p_i \langle \mathbf{w}_t - \mathbf{w}_t^i, \frac{\eta}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \nabla F_i(\mathbf{w}_t^i, \xi_j) \rangle}_{\mathcal{C}_1} \underbrace{-2\sum_{i=1}^{N} p_i \langle \mathbf{w}_t^i - \mathbf{w}^*, \frac{\eta}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \nabla F_i(\mathbf{w}_t^i, \xi_j) \rangle}_{\mathcal{C}_2}$$

It's obvious that $\mathbb{E}[\mathcal{C}_0] = 0$.
By Cauchy-Schwarz inequality and AM-GM inequality, we have

$$\begin{aligned}
\mathcal{C}_1 &= -2\sum_{i=1}^{N} p_i \langle \mathbf{w}_t - \mathbf{w}_t^i, \frac{\eta}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \nabla F_i(\mathbf{w}_t^i, \xi_j) \rangle \\
&\leq \sum_{i=1}^{N} p_i \|\mathbf{w}_t - \mathbf{w}_t^i\|^2 + \| \sum_{i=1}^{N} p_i \frac{\eta}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \nabla F_i(\mathbf{w}_t^i, \xi_j) \|^2 \\
&= \sum_{i=1}^{N} p_i \|\mathbf{w}_t - \mathbf{w}_t^i\|^2 + \mathcal{B}_1
\end{aligned}$$

So we get

$$\mathbb{E}[\mathcal{C}_1] \leq \sum_{i=1}^{N} p_i \mathbb{E}\|\mathbf{w}_t - \mathbf{w}_t^i\|^2 + \eta^2 G^2 C_t^2$$

According to Assumption 2, we know that

$$-\langle \mathbf{w}_t^i - \mathbf{w}^*, \nabla F_i(\mathbf{w}_t^i) \rangle \leq -\big(F_i(\mathbf{w}_t^i) - F_i(\mathbf{w}^*)\big) - \frac{\mu}{2}\|\mathbf{w}_t^i - \mathbf{w}^*\|^2$$

So we get

$$\begin{aligned}
\mathcal{C}_2 &\leq 2\sum_{i=1}^{N} p_i \frac{\eta}{|\mathcal{B}_i|} \sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \Big[-\big(F_i(\mathbf{w}_t^i, \xi_j) - F_i(\mathbf{w}^*)\big) - \frac{\mu}{2}\|\mathbf{w}_t^i - \mathbf{w}^*\|^2 \Big] \\
&\leq -2\eta C_t \sum_{i=1}^{N} p_i \big(F_i(\mathbf{w}_t^i) - F_i(\mathbf{w}^*)\big) - \mu\eta C_t \sum_{i=1}^{N} p_i \|\mathbf{w}_t^i - \mathbf{w}^*\|^2 \\
&= -2\eta C_t \sum_{i=1}^{N} p_i \big(F_i(\mathbf{w}_t^i) - F^* + F^* - F_i(\mathbf{w}^*)\big) - \mu\eta C_t \sum_{i=1}^{N} p_i \|\mathbf{w}_t^i - \mathbf{w}^*\|^2 \\
&= -2\eta C_t \underbrace{\sum_{i=1}^{N} p_i \big(F_i(\mathbf{w}_t^i) - F^*\big)}_{\mathcal{D}_1} - 2\eta C_t \Gamma - \mu\eta C_t \|\mathbf{w}_t - \mathbf{w}^*\|^2,
\end{aligned}$$

where $\Gamma = \sum_{i=1}^{N} p_i(F^* - F_i^*) = F^* - \sum_{i=1}^{N} p_i F_i^*$.

Next, we proceed to handle $\mathcal{D}_1$.

$$\begin{aligned}
\mathcal{D}_1 &= \sum_{i=1}^{N} p_i \left(F_i(\mathbf{w}_t^i) - F_i(\mathbf{w}_t)\right) + \sum_{i=1}^{N} p_i \left(F_i(\mathbf{w}_t) - F^*\right) \\
&\geq \sum_{i=1}^{N} p_i \langle \nabla F_i(\mathbf{w}_t), \mathbf{w}_t^i - \mathbf{w}_t \rangle + (F(\mathbf{w}_t) - F^*) \\
&\text{(from the Assumption 2)} \\
&\geq -\frac{1}{2} \sum_{i=1}^{N} p_i \left[\eta \|\nabla F_i(\mathbf{w}_t)\|^2 + \frac{1}{\eta} \|\mathbf{w}_t^i - \mathbf{w}_t\|^2\right] + (F(\mathbf{w}_t) - F^*) \\
&\text{(from the AM-GM inequality)} \\
&\geq -\sum_{i=1}^{N} p_i \left[\eta L (F_i(\mathbf{w}_t) - F_i^*) + \frac{1}{2\eta} \|\mathbf{w}_t^i - \mathbf{w}_t\|^2\right] + (F(\mathbf{w}_t) - F^*) \\
&\text{(from the L-smooth inference)} \\
&\geq -(\eta L + 1)\Gamma - \frac{1}{2\eta} \sum_{i=1}^{N} p_i \|\mathbf{w}_t^i - \mathbf{w}_t\|^2,
\end{aligned}$$

where L-smooth inference as show:

$$\|\nabla F_i(\mathbf{w}_t^i)\|^2 \leq 2L \left(F_i(\mathbf{w}_t^i) - F_i^*\right). \tag{32}$$

Thus, we get

$$\mathcal{C}_2 \leq 2\eta^2 C_t L \Gamma + C_t \sum_{i=1}^{N} p_i \|\mathbf{w}_t^i - \mathbf{w}_t\|^2 - \mu\eta C_t \|\mathbf{w}_t - \mathbf{w}^*\|^2$$

To sum up,

$$\begin{aligned}
\mathbb{E}[\mathcal{A}_1] &= \mathbb{E} \sum_{i=1}^{N} p_i \|\mathbf{w}_t - \mathbf{w}_t^i\|^2 + \eta^2 G^2 C_t^2 + 2\eta^2 C_t L \Gamma \\
&\quad + C_t \mathbb{E} \left[\sum_{i=1}^{N} p_i \|\mathbf{w}_t^i - \mathbf{w}_t\|^2\right] - \mu\eta C_t \mathbb{E} \|\mathbf{w}_t - \mathbf{w}^*\|^2 \\
&= (1 + C_t) \mathbb{E} \left[\sum_{i=1}^{N} p_i \|\mathbf{w}_t - \mathbf{w}_t^i\|^2\right] - \mu\eta C_t \mathbb{E} \|\mathbf{w}_t - \mathbf{w}^*\|^2 \\
&\quad + \eta^2 \left(G^2 C_t^2 + 2L\Gamma C_t\right),
\end{aligned}$$

and

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i=1}^{N} p_i \|\mathbf{w}_t - \mathbf{w}_t^i\|^2\right] \\
&= \mathbb{E}\left[\sum_{i=1}^{N} p_i \|(\mathbf{w}_t - \mathbf{v}_{t+1}^i) - (\mathbf{w}_t^i - \mathbf{v}_{t+1}^i)\|^2\right] \\
&\leq \mathbb{E}\left[\sum_{i=1}^{N} p_i \|\mathbf{w}_t^i - \mathbf{v}_{t+1}^i\|^2\right] \\
&\leq \mathbb{E}\left\|\sum_{i=1}^{N} p_i \frac{\eta}{|\mathcal{B}_i|}[\sum_{j=1}^{|\mathcal{B}_i|} C_t^{i,j} \cdot \nabla F_i(\mathbf{w}_t^i, \xi_j) + \sigma C \cdot \mathcal{N}(0, \mathbf{I})]\right\|^2 \\
&= \mathbb{E}\left[\mathcal{A}_2\right] = \eta^2 C_t^2 G^2 + \frac{\eta^2 \sigma^2 C^2 d}{\hat{B}^2}.
\end{aligned}$$

So we get,

$$\mathbb{E}\left[\mathcal{A}_1\right] = (1 - \mu\eta C_t)\mathbb{E}\|\mathbf{w}_t - \mathbf{w}^*\|^2 + (1 + C_t)\eta^2 C_t^2 G^2 + \eta^2\left(2G^2 C_t^2 + 2L\Gamma C_t + 2\frac{\sigma^2 C^2 d}{\hat{B}^2}\right)$$

All in all, we get

$$\mathbb{E}\|\mathbf{v}_{t+1} - \mathbf{w}^*\|^2 \leq (1 - \mu\eta C_t)\mathbb{E}\|\mathbf{w}_t - \mathbf{w}^*\|^2 + \eta^2 A,$$

where

$$A = G^2 C_t^3 + 3G^2 C_t^2 + 2L\Gamma C_t + \frac{2\sigma^2 C^2 d}{\hat{B}^2}.$$

B Related Work

While various fairness concepts exist in machine learning [7,29], recent studies in DPFL have predominantly focused on *group fairness* to mitigate biases related to sensitive attributes like gender and race [14, 15, 34]. Notable works have integrated DP with fairness constraints (e.g., equal opportunity or decision boundary fairness) to protect attribute privacy while reducing discrimination [9, 16, 37].

In contrast, our research shifts focus to *balanced performance fairness*, a relatively unexplored area within DPFL. Unlike group fairness which targets demographic parity, balanced performance fairness—introduced by Li et al. [25] and further explored in [23, 42]—prioritizes uniform model performance across clients, making it particularly suitable for federated settings involving diverse institutions. While parameter tuning (e.g., clipping thresholds) can improve group fairness [15], achieving balanced performance fairness necessitates fundamentally adjusting the direction of model updates. Balancing these directional adjustments with the noise and clipping inherent in DP presents a unique challenge. To the best of our knowledge, this is the first study to address balanced performance fairness within the context of DPFL.