Mitigating Unfairness in Differentially-Private Federated Learning

BINGQIAN DU, Huazhong University of Science and Technology, Wuhan, China
LIYAO XIANG, Shanghai Jiao Tong University, Shanghai, China
CHUAN WU, Department of Computer Science, The University of Hong Kong, Hong Kong, Hong Kong

Federated learning is a new learning paradigm which utilizes crowdsourced data stored at dispersed user devices (aka clients) to learn a global model. Studies have shown that even though data are kept on local devices, an adversary is still able to infer client information during the training process or from the learned model. Differential privacy has recently been introduced to deep learning model training, to protect data privacy of clients. Nonetheless, it exacerbates unfairness with the learned model among participating clients due to its uniform clipping and noise addition, even when the training loss function explicitly considers unfairness. To validate the impact of the differential privacy mechanism in federated learning, we carefully approximate the correlation between fairness performance across clients and the fundamental operations within the differential privacy mechanism and quantify the influence of differential privacy mechanisms on model performance across various clients. Subsequently, leveraging our theoretical findings regarding the effect of the differential privacy mechanism, we formulate the unfairness mitigation problem and propose an algorithm based on the modified method of differential multipliers. Extensive evaluation shows that our method outperforms state-of-the-art differentially private federated learning algorithm by about 30% for non-i.i.d. data distribution in terms of the variance of model performance across clients.

CCS Concepts: • **Computing methodologies** → **Cooperation and coordination**; • **Security and privacy** → *Distributed systems security;*

Additional Key Words and Phrases: Federated learning, fairness, differential privacy

ACM Reference Format:
Bingqian Du, Liyao Xiang, and Chuan Wu. 2025. Mitigating Unfairness in Differentially-Private Federated Learning. *ACM Trans. Model. Perform. Eval. Comput. Syst.* 10, 2, Article 12 (May 2025), 25 pages. https://doi.org/10.1145/3725847

1 Introduction

To learn a global machine learning model with dispersed data at a vast number of volatile clients, federated learning has been proposed [20] and used in various applications. For example, it has

This work was supported in part by grants from National Natural Science Foundation of China (Grant No. 62302187), Hubei Provincial Natural Science Foundation of China (Grant No. 2024AFB047), Hong Kong RGC under the contracts C7004-22G (CRF), C5032-23G (CRF), CRS_PolyU501/23 (CRS) and T43-513/23-N (TRS).
Authors' Contact Information: Bingqian Du, Huazhong University of Science and Technology, Wuhan, Hubei, China; e-mail: bqdu@hust.edu.cn; Liyao Xiang, Shanghai Jiao Tong University, Shanghai, China; e-mail: xiangliyao08@sjtu.edu.cn; Chuan Wu, Department of Computer Science, The University of Hong Kong, Hong Kong, Hong Kong; e-mail: cwu@cs.hku.hk.

ACM 2376-3639/2025/05-ART12
https://doi.org/10.1145/3725847

been used for next-word prediction of smartphone keyboard [16] and hospitalization prediction for cardiac events [6].

Federated learning is iterative: the global model is distributed to clients, which compute parameter updates using local data and send updates to a central server; the central server performs global model updates and redistributes the up-to-date model to the clients for further training. As compared to conventional distributed learning, federated learning features high volatility of clients, highly-skewed non-i.i.d. data distribution and privacy issue [22]. A large number of studies have shown that reconstruction attack and membership attack are possible in federated learning based on knowledge of gradients and the final learned model [13, 17], and additional privacy preserving measures should be taken.

Differential privacy mechanisms have recently been adopted to deep learning settings and shown effectiveness in preventing client identity leakage in both the training process and with the final learned model [2, 28]. The most common differential privacy mechanism used in **Stochastic Gradient Descent** (**SGD**)-based deep learning is to add additional noise to the gradients computed by clients [2]. The amount of noise added is proportional to the norm of the respective gradients, in order to hide the contribution of any single client to model learning. To bound the amount of noises and achieve high test accuracy, gradient clipping is usually applied [2], which scales down the gradients of all clients by a uniform clipping value so that all clients' gradients have a norm smaller than this value.

Recently, studies have shown that with non-i.i.d. sample class distribution across clients, federated learning itself leads to unfairness among different clients [23, 29], in terms of the accuracy (the percentage of correct predictions compared to the labels) or loss (the average errors made for samples) evaluated with the data samples of each client using the trained model, respectively. For example, the experiments in [23] show that naive federated averaging training over the Sentiment140 dataset [14] of 1101 X (formerly known as Twitter) accounts/clients would cause hundreds of clients to experience a model accuracy of less than 0.25 while other clients have an accuracy higher than 0.75. Even though clients all contribute their private data to the training process, they receive severely unfair treatment on the learned model. Furthermore, it has been shown [3] that adding differential privacy mechanisms in deep learning may further aggravate this unfairness issue. That is, the trained model may result in more significant skewness in accuracy/loss achieved by different groups/clients, as compared to not employing differential privacy. We have observed that *despite applying unfairness mitigation methods during the gradient calculation phase, the fact that the differential privacy mechanism is added after gradient calculation undermines the efficiency of the unfairness mitigation method utilized beforehand.* As a motivating instance, we investigate the model performance of the FedAvg [27] algorithm and the fairness-aware federated learning algorithm q-FedAvg [23], both with and without the application of differential privacy during the training process in Figure 1. The results confirm that unfairness worsens after the introduction of differential privacy. Moreover, they suggest that **designing fairness mitigation methods in differentially private federated learning without accounting for the operations within the differential privacy mechanism could be inadequate.**

Mitigating this unfairness in differentially private federated learning is imperative primarily because the learned model may achieve high accuracy on average, while individual clients have no performance guarantee. This unfairness could discriminate against certain clients, especially given the application of federated learning in privacy-sensitive domains such as healthcare and finance. Consequently, it could deter participation and data contribution from clients experiencing inferior performance. To address unfairness in differentially-private federated learning, we aim at answering the following two questions:

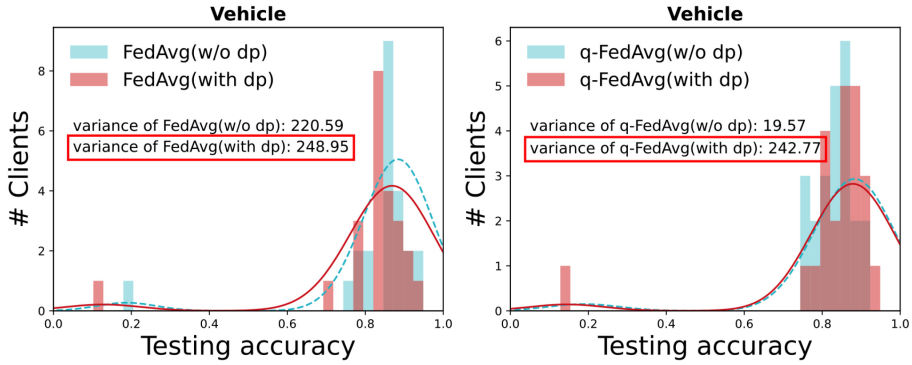


Fig. 1. The fairness level of the learned model with and without the differential privacy mechanism.

(1) What are the key factors contributing to the unfairness in differentially private federated learning? And how do these factors influence each individual client?
(2) How can we mitigate unfairness in differentially private federated learning without imposing additional computation and communication overhead on clients?

Targeting these questions, we first carefully analyze the effects of both clipping and noise addition to gradients, and then quantify how differential privacy influences model performance across different clients. We proceed by formulating a constrained optimization problem based on our theoretical analysis results. Subsequently, we propose an algorithm to derive adaptive clipping values aimed at achieving both loss minimization and unfairness mitigation. Our key technical contributions are summarized as follows:

- First, we examine the fluctuation in fairness level during the parameter update process in differentially private federated learning and identify the critical factors associated with it. Our findings indicate that statistical heterogeneity among clients and the parameters within the differential privacy mechanism directly determine the model's performance in terms of fairness. Next, We analyze the gradient deviation incurred by differential privacy for a single client, which consists of two parts: one bias term reflecting how much of the original gradients is kept after clipping, and one variance term showing the impact of the magnitude of the noise on gradients. This result exposes an inherent tradeoff between gradient clipping operations and noise addition operations within the differential privacy mechanism when considering the fairness of the model.
- Drawing from our theoretical analysis results, we delineate several properties that a fairness mitigation method in the differentially private federated learning process should adhere to and we formulate the optimization problem accordingly, aiming at optimizing the clipping values for all clients to strike a balance between loss minimization and unfairness mitigation. Due to the complexity of NN loss computation, accurately calculating a closed-form function for the optimal clipping values is technically challenging. Therefore, we employ the modified method of differential multipliers and Taylor approximation to derive the solution to the optimization problem, all while avoiding any additional communication and computation overhead for clients.
- To the best of our knowledge, this is the first work to emphasize that the design of the differential privacy mechanism should be integrated with unfairness mitigation, rather than merely adding it to the gradients calculated by a fairness-aware learning algorithm, as commonly practiced in existing unfairness mitigation solutions. Evaluation results indicate that

our method can enhance fairness by approximately 20% to 30% in terms of the variance of model performance among different clients while achieving accuracy that is almost comparable to state-of-the-art differentially privacy federated learning algorithm.

2 Preliminaries

2.1 Differential Privacy

Differential privacy is a privacy-preservation paradigm which ensures that an adversary with arbitrary side information (e.g., the identity of other clients except the one it is inferring) can not infer, with high probability, whether a particular client has contributed its data, by hiding the contribution of the client using some mechanisms (e.g., adding noise proportional to client contribution). The (ϵ, δ) differential privacy definition is given as follows:

Definition 1 (Differential Privacy [2]). A randomized mechanism $\mathcal{M}$: $\mathcal{D} \rightarrow \mathcal{R}$, which maps the domain $\mathcal{D}$ to range $\mathcal{R}$, satisfies (ϵ, δ)-differential privacy if for any pair of adjacent inputs (only differ in one entry) $d, d' \in \mathcal{D}$, it holds that

$$\Pr[\mathcal{M}(d) \in O] \leq e^{\epsilon} \Pr[\mathcal{M}(d') \in O] + \delta, \tag{1}$$

where $O \in \mathcal{R}$.

Here, ϵ and δ are privacy parameters. Smaller ϵ corresponds to better privacy. δ accounts for the probability that $\Pr[\mathcal{M}(d) \in O] > e^{\epsilon} \Pr[\mathcal{M}(d') \in O]$.

A commonly adopted differential privacy scheme is the **Gaussian mechanism** (**GM**), which preserves the (ϵ, δ)-differential privacy of function f by adding noises sampled from a normal distribution to the output of f. The noise distribution has a variance proportional to the sensitivity of f, so that it is capable to hide the contribution of any single client. The sensitivity S_f of function f is defined as the maximum l_2-norm difference between f for any adjacent input pair: $S_f = \max_{d, d' \in D} ||f(d) - f(d')||_2$. In order to achieve a bounded S_f, clipping operation is usually applied. For each $x \in d$, we clip the value of $f(x)$ by $\frac{f(x)}{\max(1, \frac{||f(x)||_2}{S})}$, where S is the uniform clipping value for all x (i.e., uniform clipping).

2.2 Federated Learning with Differential Privacy

The most popular federated learning scheme is FedAvg [27], which randomly samples a subset of clients in each training round and lets each of them perform several training epochs using local data before sending gradients to the central server, which aggregates all gradients from these clients for global model update. Given complete client set C, the goal of federated learning typically is

$$\min_{w} \mathcal{L}(w) = \sum_{k \in C} p_k L_k(w), \text{with } L_k(w) = \frac{1}{n_k} \sum_{x_d \in \Omega_k} L(w, x_d), \tag{2}$$

where $L_k(w)$ is the loss function of client k, which equals to the empirical loss on its local data sample set Ω_k. Each client k is associated with a particular weight p_k, which indicates the contribution of client k and normally is calculated by $\frac{n_k}{n}$. n_k denotes the training sample number of client k and $n = \sum_{k \in C} n_k$. By setting $p_k = \frac{n_k}{n}$, federated learning is minimizing a traditional empirical risk objective over the complete dataset across all clients. In FedAvg, at round t of training, central server uniformly at random samples a subset of clients C_t, then the gradient would be updated by averaging the local gradients/updates g_t^k from client $k \in C_t$ as follows:

$$w_{t+1} = w_t + \frac{|C|}{|C_t|} \sum_{k \in C_t} p_k g_t^k. \tag{3}$$

There have been recent studies applying differential privacy to federated learning, due to possible data reconstruction attack and client membership attack in the training process and with the trained model [12, 31]. There are two different notions of differential privacy in the federated learning setting:

Local differential privacy. Each client obfuscates gradients locally before gradient aggregation. This ensures that the adversary cannot infer the knowledge of single data sample used for client's gradient computation [18, 33]. But this method can be too stringent for practical usage, with increasing noise injection when the client number increases [5, 32].

Global differential privacy. It ensures differential privacy across all clients by applying differential privacy mechanisms to aggregated gradients so that contribution of a single client is hidden and noises are added at the central server side. McMahan et al. [28] design a global differentially-private federated learning scheme, employing the Gaussian mechanism with uniform clipping, and add noise to the average of clipped gradients from sampled clients. They find that realistic training of differentially private federated learning is possible with negligible loss in model utility.

We focus on global differential privacy instead of local differential privacy in this article with the goal to seek the possibility of mitigating unfairness through careful design of differential privacy mechanism. The autonomous characteristic of local differential privacy hinders the achievement of such a goal. That is clients may deviate from the goal of achieving a fair learned model and design local differential privacy mechanism to maximize its own performance.

2.3 Fairness in Federated Learning

We consider fair treatment of the learned model for different clients [23], i.e., similar performance evaluated at different clients with their own data samples using the model. We use the following fairness definition in this work and use the loss as the performance metric in theoretical analysis.

Definition 2 (Fairness of Model Performance [23]). For any two machine learning models w and $\widetilde{w}$, we say model w is more fair than $\widetilde{w}$ if the performance of w on total C clients is more uniform than performance of $\widetilde{w}$ on C clients.

This fairness definition is different from the traditional ones, such as demographic parity or equalized odds, which are evaluated on groups with different sensitive features, such as gender or race. For federated learning, the heterogeneity of data distribution is mainly among clients and a more equitable outcome of the learned model could offer increased performance assurance for all participating clients. These concerns motivate our fairness definition for federated learning framework.

3 Related Work

3.1 Fairness in Machine Learning

Most of existing machine learning works consider fairness as parity performance with respect to sensitive attributes, which is different from the goal we want to achieve in this work. In centralized machine learning, Dwork et al. [9] investigate fairness in classification problems to prevent discrimination against protected population subgroups, e.g., minority population groups should have similar outcome as the majority. Their fairness is defined as Lipschitz mapping, which requires the distance between outcomes of two individuals is upper bounded by the distance between individuals themselves. This fairness definition proposed can be seen as a generalization of differential privacy, by requiring the distance between the outcomes of two similar input to be bounded. Cummings et al. [7] show the existence of a **probably approximately correct** (**PAC**) learner with high probability, that is both differentially private and fair with Equality of False Positives and

Equality of False Negatives for sensitive attributes. The fairness of Demographic Parity for protected attributes and differential privacy of a logistic regression model are investigated by Xu et al. [36]. Jagielski et al. [19] study the fair learning of classifier with protected attributes, such as race, gender, and so on, under the constraint of differential privacy. The relationship between fairness of different groups and differential privacy has been empirically studied by [11].

In federated and decentralized learning, Du et al. [8] utilize a minimax game to assign individual reweighing values to training samples and consider reweighing in the loss function and the fairness constraint of demographic parity with respect to sensitive attributes. Zhang et al. [38] propose to use a multi-agent reinforcement learning framework and a secure information aggregation protocol to optimize accuracy and fairness towards some demographic attributes. The decentralized setting was considered by Lyu et al. [25], which builds a reputation system to ensure that clients with more contributions have higher credibility and better performing final model for fairness, which is the opposite of the fairness we investigate. Gu et al. FairFed [10] introduces a novel fairness-aware aggregation method aimed at addressing group fairness in federated learning, helping to mitigate potential biases against certain populations. [15] investigate the impact of differential privacy to fairness in FL through extensive empirical study, but the specific strategy for designing DP mechanism is missing.

There are few works that study similar fairness definition as ours. Mohri et al. [29] are among the first to study fairness in federated learning. They show that the learned model by existing federated learning methods can be biased towards some clients. They propose an agnostic federated learning scheme to mitigate this unfairness, optimizing the loss function with respect to the data distribution of the client experiencing the worst model performance. Li et al. [23] further study this unfairness and propose qFedAvg training, whose objective to optimize is $\sum_{k=1}^{m} \frac{p_k}{1+\alpha} L_k(w)^{1+\alpha}$ instead of standard $\sum_{k=1}^{C} p_k L_k(w)$, where $L_k(w)$ is the loss function of client k and p_k is the weight for client k. α is a hyper-parameter to control the level of fairness, and C is the client set. By improving the objective to optimize in federated learning, qFedAvg and agnostic federated learning give more weights to clients with poor performance to achieve better fairness. None of them consider privacy preserving in their works. However, Bagdasaryan et al. [3] show that differential privacy itself can cause unexpected unfairness among participants, worsening the situation of federated learning. Ling et al. [24] addresses unfairness by proposing a novel loss function that includes an unfairness penalty term. The weight of this penalty is optimized by balancing the tradeoff between fairness and the clipping operation in local differential privacy, a method that operates independently of the global differential privacy mechanisms examined in our work.

Our work investigates the unfairness in differentially-private federated learning, and seek the possibility to mitigate the unfairness by carefully designing the differential privacy mechanism, which is orthogonal to methods with reweighted loss function and can be combined with them.

3.2 Privacy Protection in Federated Learning

One of the basic motivations of federated learning is to protect the data privacy of participating clients. Zhang et al. [40] leverage this property of federated learning to resolve the data deficiency problem in data-driven machinery fault diagnosis and develop self-supervised learning and dynamic validation scheme for the learning process. Zhang et al. [39] further investigate the data privacy-preserving federated transfer learning problem in machinery fault diagnosis tasks and utilize adversarial learning and a prediction consistency scheme to solve the domain shift issue. The privacy threat and defense mechanism in federated learning have been discussed by Xiong et al. [35]. They analyse the privacy leakage in federated learning with non-i.i.d data distribution and propose a novel algorithm to protect the privacy for federated learning. Wei et al. [33] consider local differential privacy mechanism for federated learning and develop a theoretical convergence

bound for the trained model in local DP scenario. Ma et al. [26] analyze the privacy and security issues in federated learning and point out several challenges when deploying privacy and security preserving methods in federated learning. Even though a large number of works focus on the privacy protection issue in federated learning, few of them consider the fairness problem in differentially private federated learning scenario.

4 Problem Model

4.1 The Federated Learning System

We consider federated learning of a model over a large number of C clients. Each client stands for a user or a device, which has collected a local dataset. We use w to denote the set of parameters in the model. In each training round t, a central server randomly selects a subset of m_t clients, where each client is sampled independently with probability q to utilize the moment accountant [2], sends the global model w_t to them, and invites them to train on their local dataset for E epochs. Each epoch represents the training with the entire local dataset to compute a loss and derive the gradients of model parameters based on the loss [27].

In each training round, the sampled clients jointly optimize loss function (2). In the global update process, term $\frac{1}{q}$ is added to ensure the aggregated gradient is an unbiased estimator of the complete gradient $\sum_{k\in C} p_k g_t^k$.

$$w_{t+1} = w_t + \frac{1}{q}\sum_{k\in m_t} p_k g_t^k. \tag{4}$$

After E epochs of local training, each client sends its computed gradients to the central server. After aggregating all gradients from the selected clients, the server updates its global model parameters accordingly to obtain w_{t+1}. The training repeats until the global model converges. Let T denote the total number of training rounds that ensures model convergence. Relevant notations are listed in Table 1.

4.2 Differential Privacy Mechanism

We consider the Gaussian mechanism [2] to achieve **differential privacy** (**DP**) in federated learning. Gradient clipping and noise addition are both applied to bound the sensitivity of the sum

Table 1. Notation

C	total # of clients	k	client index
q	sampling probability	η	learning rate
$\mathcal{L}_k$	loss function of client k	E	# of epochs
B	batch size of clients	ϵ, δ	privacy parameter
n	# of samples from all clients	d	# of parameters
n_k	# of samples of client k	p_k	weight of client k
m_t	the set of clients sampled at round t		
g_t^k	gradient updates from client k at round t		
C_t^k	clipping value to client k's gradients at round t		
$\mathbb{C}_t$	upper bound of $C_t^k, \forall k \in [1, m_t]$ at round t		
ϕ_t	random vector of sampled noises at round t		
l_t^k	gradient norm of client k at round t		
σ	variance parameter for noise distribution		
w_t	parameters of neural network at round t		
$\mathcal{M}$	differential privacy mechanism		

operation and to achieve privacy. The summation operation is to aggregate gradient updates from sampled clients, and the sensitivity is therefore the maximum difference of l_2-norm of the aggregated gradient update for any two adjacent sampled client sets. We grant this framework maximum flexibility to investigate how operations in differential privacy impact the fairness of the model. Let C_t^k denote the clipping value for the gradient of client k at round t and let $\mathbb{C}_t$ denote the upper bound of the clipping values across all clients, which also represents the sensitivity. ϕ_t is a vector of d independently sampled random variables according to the normal distribution $\mathcal{N}(0, \sigma^2\mathbb{C}_t^2)$, where d is the number of parameters in w (i.e., $d = |w|$) and σ is the variance parameter for noise distribution. Larger σ corresponds to higher privacy. Let g_t^k be the vector of gradients that the server receives from client k, computed with k's local dataset. The global model is updated at the server as follows with the differential privacy mechanism in place [28]. Here we use w_t to represent the vector of model parameters obtained with different privacy mechanism added.

$$w_{t+1} = w_t - \eta \left(\sum_{k \in m_t} \frac{p_k}{q} g_t^k \cdot \min\left(1, \frac{C_t^k}{||g_t^k||_2}\right) + \phi_t \right). \tag{5}$$

With the Gaussian differential privacy mechanism, we multiply gradients computed by each client, g_t^k, by $\min(1, \frac{C_t^k}{||g_t^k||_2})$ for gradient clipping to ensure the l_2-norm of its gradient is no larger than clipping value C_t^k, before adding all of them from the m_t clients; we then add noise ϕ_t to the aggregated gradients. The central server is responsible for setting the clipping value C_t^k, $\mathbb{C}_t$ and noise variance σ to enforce differential privacy.

In the global differential privacy scenario, reliable transmission (i.e., no eavesdropping) and trusted server are assumed, as otherwise, DP mechanisms have to be added to the gradients locally before sending gradients to the server, which is the case of local differential privacy. The adversary has access to noisy gradients in each training round and the final learned model, and has arbitrary auxiliary information. The clients can be honest-but-curious, meaning that they would not deviate from defined protocols and use harmful data for training, but they will attempt to infer all possible information from received messages. The central server responsible for setting differential privacy mechanism and global model update is trusted by all clients.

4.3 Goal

Our goal is to train a fair model under the federated learning framework, with differential privacy mechanism (5) applied at each round of model update. We assume that the dataset of individual client only consists of unpolluted samples so that the fairness requirement of final learned model is reasonable. We exploit the maximum gap between the expected model performance and the model performance for a single client as our fairness metric to measure the uniformity of model performance and we utilize loss as the reference for the model performance. In particular, within each training round t, our aim is at minimizing the following term alongside the expected model loss:

$$\min \max_{j \in C} \left| \sum_{k \in C} \frac{1}{|C|} \mathcal{L}_k(w_{t+1}) - \mathcal{L}_j(w_{t+1}) \right| \text{ subject to: } w_{t+1} = w_t - \eta \left(\sum_{k \in m_t} \frac{p_k}{q} g_t^k \cdot \min\left(1, \frac{C_t^k}{||g_t^k||_2}\right) + \phi_t \right) \tag{6}$$

where $\mathcal{L}_k(w_{t+1})$ is the loss function of client k. To achieve this goal, we address the two questions posed in Sec 1 and proceed with two steps: (i) **Discover the reasons behind why the model may exhibit unfairness in differentially private federated learning:** we characterize the fairness dynamics during the training process and investigate the impact of the differential privacy on the

gradients used in model update, to provide a theoretical understanding of how we can effectively mitigate unfairness; (ii) **Develop a fairness mitigation method based on theoretical insights:** we define the complete optimization problem using the insights gathered from theoretical analysis and introduce a centralized algorithm to address it, ensuring that no additional burdens are imposed on clients. We will present the details in the next section.

5 Connecting Model Performance with Differential Privacy Mechanism

In this section, our objective is to address the first question, delving into the factors that contribute to the issue of unfairness within the context of differentially private federated learning. To accomplish this objective, our first step involves analyzing the fluctuations in fairness levels throughout a single round of the training process. From this examination, it becomes evident that fairness level for a single client in differentially private federated learning results from the complex combined effects of the model itself, the distribution of training data across all clients, and the two primary operations, i.e., gradient clipping and noise addition, within the framework of differential privacy mechanisms. Therefore, addressing unfairness in this situation would pose a significant challenge. Next, our focus shifts solely to the differential privacy mechanism employed during the global model update phase. We aim at isolating the impact of gradient clipping and noise addition on the contribution of an individual client to the updated model. Additionally, we seek to explore the inherent tradeoffs between these two operations when addressing unfairness.

5.1 Fairness in Differentially Private Federated Learning

We start with analyzing the fairness under differentially private federated learning, with an attempt to understand what factors cause this unfairness issue. We investigate the dynamics of the fairness under the parameter update rule (5). The following theorem elucidates the relationship between fairness performance and the influencing factors.

Theorem 1. *Assume the loss function ℓ is twice differentiable with respect to parameter w_t and the parameter update follows Equation (5). Given w_t, the gap between the expected model performance and the model performance of client j, $\forall j \in C$ at the end of round t is upper bounded by*

$$\mathbb{E}\left[\sum_{k=1}^{m}\frac{1}{m}\mathcal{L}_k(w_{t+1})\right] - \mathbb{E}[\mathcal{L}_j(w_{t+1})], \tag{7a}$$

$$\approx \underbrace{\sum_{k=1}^{m}\frac{1}{m}\mathcal{L}_k(w_t) - \mathcal{L}_j(w_t)}_{\text{initial gap}} \underbrace{-\eta\left\langle\sum_{k=1}^{m}\frac{1}{m}\nabla\mathcal{L}_k(w_t), \bar{g}_t\right\rangle}_{(a)} \underbrace{+\eta\langle\nabla\mathcal{L}_j(w_t), \bar{g}_t\rangle}_{(b)}, \tag{7b}$$

$$+ \underbrace{\frac{\eta^2}{2}\bar{g}_t^T\left(\sum_{k=1}^{m}\frac{1}{m}H_k(w_t) - H_j(w_t)\right)\bar{g}_t}_{(c)} + \underbrace{\frac{\eta^2}{2}\mathbb{C}_t^2\sigma^2\left(\sum_{k=1}^{m}\frac{1}{m}Tr(H_k(w_t)) - Tr(H_j(w_t))\right)}_{(d)}, \tag{7c}$$

where $\bar{g}_t$ represents the aggregation of the clipped gradients $\sum_{k\in m_t}\frac{p_k}{q}g_t^k\cdot\min(1, \frac{C_t^k}{||g_k^t||_2})$, $H_k(w_t)\ \forall k \in C$ denotes the Hessian matrix of client k, and $Tr(\cdot)$ calculates the trace of a matrix. The expectation is computed over the random noise distribution of the differential privacy mechanism.

The result of Theorem 1 highlights the critical factors that influence the fairness level of the model performance in differentially private federated learning: (1) the initial disparity between

the expected model performance across all clients and the model performance for a single client, i.e., the initial fairness level of the model; (2) the combined impact of the clipping values of all clients and the heterogeneity in data distribution among all clients, as reflected by terms (a), (b), and (c); (3) the interaction between the upper bound for the clipping value across all clients $\mathbb{C}_t$, the noise variance σ of the differential privacy mechanism, and the statistical heterogeneity among all clients, as illustrated by term (d). Next, we delve into terms from the result of Theorem 1 in detail to present our findings:

- Term (a) quantifies the similarity between the gradient that minimizes the expected loss of all clients and the clipped gradient actually used during model updating. The value of this term is determined by the *clipping values of all clients*. A smaller value of this term corresponds to a smaller expected model loss (i.e., better expected model performance) across all clients $\sum_{k=1}^{m} \frac{1}{m} \mathcal{L}_k(w_{t+1})$.
- Term (b) quantifies the similarity between the gradient minimizing the loss of client j and the clipped gradient actually used for model updating, whose value is also determined by the *clipping values used by all clients*. A larger value of this term corresponds to a smaller model loss (i.e., better model performance) for client j.
- Term (c) is associated with the *clipping values of all clients* and the *heterogeneity of data distribution among them*, as the value of $H_k(w_t)$ is influenced by the training data of client k. If the data distribution is the same for all clients, then the value of $\sum_{k=1}^{m} \frac{1}{m} H_k(w_t) - H_j(w_t)$ would be close to zero.
- Term (d) is the only term that reflects the influence of the noise distribution ($\mathbb{C}_t$ and σ) of the differential privacy mechanism. Additionally, it is also related to the heterogeneity of data distribution (trace of Hessian matrix $H_k(w_t)$) among all clients.

Takeaways: Upon analyzing the terms comprising the expression of the fairness level of the model performance in Theorem 1, we conclude that **(a) The fairness issue is closely tied to the statistical heterogeneity that arises in the federated learning setting. (b) The level of model fairness could be optimized by appropriately setting the clipping values for clients participating in the training process, as well as the upper bound of all clipping values $\mathbb{C}_t$ and the noise scale σ. (c) Suppose the fairness-aware algorithm has been utilized for calculating gradient g_t^k, the operations in differential privacy mechanism can still compromise its effectiveness according to term (a) and (b). (d) The mode of client participation in federated learning directly impacts the feasibility of achieving a desired level of fairness. Full client participation, in which the updating gradients from partial clients ($\bar{g}_t$) in Theorem 1 are replaced by gradients computed through full client participation ($\sum_{k=1}^{m} p_k \nabla \mathcal{L}_k(w_t) \cdot \min(1, \frac{C_t^k}{\|\nabla \mathcal{L}_k(w_t)\|_2})$), offers greater opportunities for fairness adjustment. This approach enables the clipping value to be strategically designed to control each client's contribution, thereby achieving more effective fairness adjustment, especially when compared to scenarios with highly uneven client participation.**

We provide the detailed proof of Theorem 1 below.

Proof. We establish the proof of Theorem 1 by utilizing the Taylor approximation of the empirical loss function. This allows us to derive the difference between the expected model performance across all clients $\sum_{k=1}^{m} \frac{1}{m} \mathcal{L}_k(w_{t+1})$ and the model performance $\mathcal{L}_j(w_{t+1})$ experienced by a single client j during the training process:

For the expected model performance across all clients: Based on the parameter update rule (5) at round t, we have the following approximation for the expected model performance, where

the expectation is taken over the randomness of the noise ϕ_t:

$$\mathbb{E}\left[\sum_{k=1}^{m}\frac{1}{m}\mathcal{L}_k(w_{t+1})\right] \approx \mathbb{E}\left[\sum_{k=1}^{m}\frac{1}{m}\left[\mathcal{L}_k(w_t) - \eta\langle\nabla\mathcal{L}_k(w_t), (\bar{g}_t+\phi_t)\rangle + \frac{\eta^2}{2}(\bar{g}_t+\phi_t)^T H_k(w_t)(\bar{g}_t+\phi_t)\right]\right], \tag{8a}$$

$$= \sum_{k=1}^{m}\frac{1}{m}\mathcal{L}_k(w_t) - \eta\left\langle\sum_{k=1}^{m}\frac{1}{m}\nabla\mathcal{L}_k(w_t), \bar{g}_t\right\rangle + \frac{\eta^2}{2}\sum_{k=1}^{m}\frac{1}{m}\bar{g}_t^T H_k(w_t)\bar{g}_t + \frac{\eta^2}{2}\mathbb{E}\left[\sum_{k=1}^{m}\frac{1}{m}\phi_t^T H_k(w_t)\phi_t\right], \tag{8b}$$

$$= \sum_{k=1}^{m}\frac{1}{m}\mathcal{L}_k(w_t) - \eta\left\langle\sum_{k=1}^{m}\frac{1}{m}\nabla\mathcal{L}_k(w_t), \bar{g}_t\right\rangle + \frac{\eta^2}{2}\sum_{k=1}^{m}\frac{1}{m}\bar{g}_t^T H_k(w_t)\bar{g}_t + \frac{\eta^2}{2}\sum_{k=1}^{m}\frac{1}{m}\mathrm{Tr}(H_k(w_t))\mathbb{C}_t^2\sigma^2, \tag{8c}$$

where (8b) follows from the fact that $\mathbb{E}[\phi_t] = 0$, and (8c) follows from the independence of each dimension of ϕ_t, which is sampled independently from the distribution $\mathcal{N}(0, \mathbb{C}_t^2\sigma^2)$. This is demonstrated as follows:

$$\mathbb{E}\left[\phi_t^T H_k(w_t)\phi_t\right] = \mathbb{E}\left[\sum_{i\in[d]}(\phi_t^i)^2(H_k(w_t))_{i,i}\right] = \mathrm{Tr}(H_k(w_t))\mathbb{C}_t^2\sigma^2. \tag{9}$$

where $(H_k(w_t))_{i,i}$ is the (i,i)th entry of Hessian matrix $H_k(w_t)$ and ϕ_t^i is the ith entry of noise vector ϕ_t.

For the model performance of an individual client: Similarly, we could derive the model performance for a single client $\mathcal{L}_j(w_{t+1}), \forall j \in C$ as follows:

$$\mathbb{E}[\mathcal{L}_j(w_{t+1})] \approx \mathcal{L}_j(w_t) - \mathbb{E}\left[\eta\langle\nabla\mathcal{L}_j(w_t), \bar{g}_t+\phi_t\rangle + \frac{\eta^2}{2}(\bar{g}_t+\phi_t)^T H_j(w_t)(\bar{g}_t+\phi_t)\right], \tag{10a}$$

$$= \mathcal{L}_j(w_t) - \eta\langle\nabla\mathcal{L}_j(w_t), \bar{g}_t\rangle + \frac{\eta^2}{2}\bar{g}_t^T H_j(w_t)\bar{g}_t + \mathbb{E}\left[\frac{\eta^2}{2}\phi_t^T H_j(w_t)\phi_t\right], \tag{10b}$$

$$= \mathcal{L}_j(w_t) - \eta\langle\nabla\mathcal{L}_j(w_t), \bar{g}_t\rangle + \frac{\eta^2}{2}\bar{g}_t^T H_j(w_t)\bar{g}_t + \frac{\eta^2}{2}\mathrm{Tr}(H_j(w_t))\mathbb{C}_t^2\sigma^2. \tag{10c}$$

To obtain the gap between the expected model performance across all clients and the model performance for client j, we take the differences between (8c) and (10c),

$$\mathbb{E}\left[\sum_{k=1}^{m}\frac{1}{m}\mathcal{L}_k(w_{t+1})\right] - \mathbb{E}[\mathcal{L}_j(w_{t+1})], \tag{11a}$$

$$\approx \sum_{k=1}^{m}\frac{1}{m}\mathcal{L}_k(w_t) - \eta\left\langle\sum_{k=1}^{m}\frac{1}{m}\nabla\mathcal{L}_k(w_t), \bar{g}_t\right\rangle + \frac{\eta^2}{2}\sum_{k=1}^{m}\frac{1}{m}\bar{g}_t^T H_k(w_t)\bar{g}_t + \frac{\eta^2}{2}\sum_{k=1}^{m}\frac{1}{m}\mathrm{Tr}(H_k(w_t))\mathbb{C}_t^2\sigma^2, \tag{11b}$$

$$- \mathcal{L}_j(w_t) + \eta\langle\nabla\mathcal{L}_j(w_t), \bar{g}_t\rangle - \frac{\eta^2}{2}\bar{g}_t^T H_j(w_t)\bar{g}_t - \frac{\eta^2}{2}\mathrm{Tr}(H_j(w_t))\mathbb{C}_t^2\sigma^2, \tag{11c}$$

$$= \underbrace{\sum_{k=1}^{m}\frac{1}{m}\mathcal{L}_k(w_t) - \mathcal{L}_j(w_t)}_{\text{initial gap}} \underbrace{-\eta\left\langle\sum_{k=1}^{m}\frac{1}{m}\nabla\mathcal{L}_k(w_t), \bar{g}_t\right\rangle}_{\text{(a)}} \underbrace{+\eta\langle\nabla\mathcal{L}_j(w_t), \bar{g}_t\rangle}_{\text{(b)}}, \tag{11d}$$

$$+ \underbrace{\frac{\eta^2}{2}\bar{g}_t^T\left(\sum_{k=1}^{m}\frac{1}{m}H_k(w_t) - H_j(w_t)\right)\bar{g}_t}_{\text{(c)}} + \underbrace{\frac{\eta^2}{2}\mathbb{C}_t^2\sigma^2\left(\sum_{k=1}^{m}\frac{1}{m}\mathrm{Tr}(H_k(w_t)) - \mathrm{Tr}(H_j(w_t))\right)}_{\text{(d)}}, \tag{11e}$$

which concludes the proof. □

5.2 Differential Privacy's Effect on Single Client's Contribution

The preceding analysis demonstrates that the fairness level in differentially private federated learning is a complex outcome resulting from the interplay between the operations of the differential privacy mechanism and the inherent statistical heterogeneity present in the federated learning scenario. To disentangle the effect of the differential privacy mechanism on fairness, we investigate the impact of the two key components in the differential privacy mechanism on the fairness issue. Let $\overline{g}_t^k$ be the gradients of client k after applying the clipping operation to its gradients g_t^k at round t, and $\hat{g}_t^k$ denote the gradients after both clipping and noise addition. Since ϕ_t is added to the aggregated gradients from the local updates of $|m_t|$ sampled clients, for a single client, we consider noise $\frac{\phi_t}{|m_t|}$. Then we have $\overline{g}_t^k = g_t^k \cdot \min(1, \frac{C_t^k}{||g_t^k||_2})$ and $\hat{g}_t^k = \overline{g}_t^k + \frac{\phi_t}{|m_t|}$. We quantify the impact of differential privacy mechanism on each client's contribution to model training, by the expected difference[1] between the original gradients and gradients after applying differential privacy mechanism. The expectation is computed with respect to the noise distribution $\mathcal{N}(0, \sigma^2\mathbb{C}_t^2)$.

Theorem 2. *The gradient deviation from the original gradient that minimizes the loss of a single client after the differential privacy mechanism can be upper bounded as follows:*

$$\mathbb{E}\left[\left\|\hat{g}_t^k - g_t^k\right\|_1\right] \leq \mathbb{E}\left[\left\|\hat{g}_t^k - \overline{g}_t^k\right\|_1\right] + \mathbb{E}\left[\left\|\overline{g}_t^k - g_t^k\right\|_1\right] = \mathbb{E}\left[\left\|\frac{\phi_t}{|m_t|}\right\|_1\right] + \left\|\overline{g}_t^k - g_t^k\right\|_1$$

$$= \frac{1}{|m_t|}\sum_{i=1}^{d} 2\int_{x=0}^{+\infty} x \frac{1}{\sqrt{2\pi}\sigma\mathbb{C}_t} e^{-\frac{x^2}{2\sigma^2\mathbb{C}_t^2}} + \left\|g_t^k - g_t^k \cdot \min\left(1, \frac{C_t^k}{||g_t^k||_2}\right)\right\|_1, \tag{12a}$$

$$= \frac{d\sqrt{1/\pi}\sigma\mathbb{C}_t}{|m_t|} + \left\|g_t^k\right\|_1 \cdot \max\left(0, 1 - \frac{C_t^k}{||g_t^k||_2}\right). \tag{12b}$$

The first inequality above is due to the triangle inequality. Since noise ϕ_t is d-dimensional noise sampled from Gaussian distribution $\mathcal{N}(0, \sigma^2\mathbb{C}_t^2)$, the value of $\mathbb{E}[||\phi_t||_1]$ can be calculated based on $\int_{x=0}^{+\infty} x\exp(-x^2) = -\frac{1}{2}\exp(-x^2)|_0^{+\infty} = \frac{1}{2}$. The change brought by differential privacy on a client's gradients is upper bounded by a *bias term* ($||g_t^k||_1 \cdot \max(0, 1 - \frac{C_t^k}{||g_t^k||_2})$) and a *variance term* ($d\sqrt{1/\pi}\sigma\mathbb{C}_t$). The bias term is due to the clipping operation, while the variance term is related to both the clipping value and the noise variance.

To mitigate unfairness in model performance across clients, we can give clients experiencing worse model performance larger clipping values C_t^k to keep more fraction of their original gradients (smaller value of $||g_t^k||_1 \cdot \max(0, 1 - \frac{C_t^k}{||g_t^k||_2})$), allowing their gradients to contribute more during model training. However, on the other hand, as the noise scale is decided by the upper bound $\mathbb{C}_t$ of all clipping values C_t^k across every client $k \in m_t$, a large C_t^k results in a larger value of $\mathbb{C}_t$, thus leading to a greater amount of noise being added to the respective gradients (larger $\frac{d\sqrt{1/\pi}\sigma\mathbb{C}_t}{|m_t|}$), increasing the deviation of gradients from their original values for all clients prior to applying the differential privacy mechanism, reducing the contribution of a client's original gradients. **We need to carefully strike the sweet spot within the tradeoff, and balance the contributions from different clients in model training, through strategically setting different clipping values C_t^k for different clients k.**

[1]We utilize l_1-norm here. Given that $||x||_q \leq ||x||_p$ for $x \in \mathbb{R}^n$ and $1 \leq p \leq q < \infty$, by using l_1-norm, we could obtain an upper bound of the impact of DP for all possible l-norm ($l \neq \infty$).

6 Differential Privacy Mechanism for Fair Federated Learning

From the analysis presented above, we aim at addressing the unfairness issue in differentially private federated learning. Our design is grounded in the following principles, derived from our analysis of the key factors influencing fairness levels in differentially private federated learning and the examination of the impact of the differential privacy mechanism on gradient deviation for individual clients:

(1) **When designing fairness mitigation methods, it's essential to balance both minimizing loss and mitigating unfairness.** Prioritizing fairness alone may compromise model performance. For instance, optimizing the clipping value in the differential privacy mechanism at each round only to consider the fair treatment among clients could result in parameter updates that reduce effectiveness of the model in addressing downstream tasks.

(2) **Taking into account the parameters involved in the operations of the differential privacy mechanism, such as the clipping values C_t^k for all clients, the upper bound of clipping values $\mathbb{C}_t$, and the noise variance σ, when designing fairness mitigation methods can enhance the fairness level of the model.** The commonly used approach to ensure model fairness involves incorporating a fairness metric as a penalty term into the loss function, with fairness achieved through parameter updates. However, the efficiency of this method may diminish when the differential privacy mechanism is applied post-gradient calculation. As indicated in Theorem 1, the differential privacy mechanism directly impacts the fairness level of the updated model. Improper parameter configuration can potentially compromise the calculated gradient intended to ensure fairness.

(3) **There is an inherent tradeoff between the gradient clipping operation and the noise addition operation within the differential privacy mechanism when addressing unfairness issues.** The result in Theorem 2 demonstrates that a larger clipping value for a single client can enhance fairness when that client experiences poorer model treatment. However, this also increases the value of $\mathbb{C}_t$, leading to heightened noise injection onto the gradient, potentially undermining performance improvements for this client. Therefore, when optimizing parameters in the differential privacy mechanism for unfairness mitigation, it's crucial to address this inherent contradiction specifically.

In line with these principles, the optimization problem that all clients collaborate to solve in fair differentially private federated learning is as follows:

$$\min_{w\in\mathbb{R}^d} \sum_{k=1}^{m} p_k \mathcal{L}_k(w) \qquad \text{subject to: } \max_{j\in C} \left| \sum_{k=1}^{m} \frac{1}{m} \mathcal{L}_k(w) - \mathcal{L}_j(w) \right| \leq \alpha. \tag{13}$$

In this formulation, we consider both loss minimization to enhance the overall model performance and ensuring fair treatment of the learned model for all participating clients. Note that fairness is ensured through a relaxed constraint. This is because achieving exact fairness may not be possible while maintaining both differential privacy and non-trivial model performance, particularly under certain loss functions, such as the one used in binary classification, which assesses the model's probability of predicting label zero when the true label of the data sample is one [7]. The parameter α in Equation (13) is a hyperparameter that controls the level of fairness. A smaller α corresponds to stricter and improved fairness.

To address this constrained optimization problem in differentially private federated learning, recalling the second and third principles drawn from Theorems 1 and 2 respectively, we propose a method that *integrates parameters in the operations of the differential privacy mechanism into the optimization of problem (13)*. This approach diverges from simply applying the differential privacy mechanism to the solution that optimizes problem (13), as it may potentially undermine

the effectiveness of the calculated solution, as demonstrated in Theorem 1. Additionally, we *consider the tradeoff between gradient clipping operation and noise addition operation*, as reflected by Theorem 2, explicitly taking into account the impact of the noise distribution used in the differential privacy mechanism when solving problem (13).

During each training round of differentially private federated learning, parameters are updated according to $w_{t+1} = w_t - \eta(\sum_{k\in m_t} \frac{p_k}{q} g_t^k \cdot \min(1, \frac{C_t^k}{||g_t^k||_2}) + \phi_t)$, which incorporates both gradient clipping and noise addition operations. Therefore, our method approaches problem (13) from the perspective of the updated parameters, ensuring they optimize model performance, fairness metrics, and the tradeoff between gradient clipping and noise addition, even after the application of the differential privacy mechanism. The optimization problem at each training round is as follows:

$$\min_{C_t^k} \sum_{k\in m_t} p_k \mathcal{L}_k(w_{t+1}) + \gamma \left[d\sqrt{1/\pi}\sigma\mathbb{C}_t + \sum_{k\in m_t} \left\| g_t^k \right\|_1 \cdot \max\left(0, 1 - \frac{C_t^k}{||g_t^k||_2}\right)\right], \tag{14a}$$

$$\text{subject to: } \max_{j\in m_t} \left| \sum_{k\in m_t} \frac{1}{|m_t|} \mathcal{L}_k(w_{t+1}) - \mathcal{L}_j(w_{t+1}) \right| \le \alpha, \tag{14b}$$

$$\mathbb{C}_t \ge C_t^k, \forall k \in m_t. \tag{14c}$$

Here, γ represents the weight for the tradeoff present in the differential privacy mechanism when addressing fairness issues.

Due to the non-convex nature of the loss function $\mathcal{L}(\cdot)$, obtaining a closed-form solution for the clipping value C_t^k is impossible. Therefore, we employ the **modified method of differential multipliers** (**MMDM**) algorithm to solve constrained differential optimization. This algorithm involves the following steps:

(1) We begin by deriving the Lagrangian of problem (14), which will be addressed via gradient descent and gradient ascent during each training round. To facilitate the application of the MMDM method, we transform the inequality constraint into an equality constraint. Specifically, constraint (14b) will be equal to zero if $\max_{j\in m_t} | \sum_{k\in m_t} \frac{1}{|m_t|} \mathcal{L}_k(w_{t+1}) - \mathcal{L}_j(w_{t+1})| \le \alpha$ is satisfied. However, if this condition is not met, the constraint will be equal to $\max_{j\in m_t} | \sum_{k\in m_t} \frac{1}{|m_t|} \mathcal{L}_k(w_{t+1}) - \mathcal{L}_j(w_{t+1})| - \alpha$. We use $f(w_{t+1})$ to denote this transformation, then we have

$$f(w_{t+1}) = \begin{cases} \max_{j\in m_t} | \sum_{k\in m_t} \frac{1}{|m_t|} \mathcal{L}_k(w_{t+1}) - \mathcal{L}_j(w_{t+1})| - \alpha, & \text{if } \max_{j\in m_t} | \sum_{k\in m_t} \frac{1}{|m_t|} \mathcal{L}_k(w_{t+1}) - \mathcal{L}_j(w_{t+1})| > \alpha \\ 0, & \text{otherwise} \end{cases}. \tag{15}$$

However, the same technique cannot be applied to constraint (14c), as it is a hard constraint that must be adhered to for each application of the differential privacy mechanism to ensure privacy. Instead, we directly set $\mathbb{C}_t = \max_{j\in m_t} C_t^k$ during the solving process. The complete Lagrangian function would be:

$$Lagrangian = \sum_{k\in m_t} p_k \mathcal{L}_k(w_{t+1}) + \gamma \left[d\sqrt{1/\pi}\sigma \max_{j\in m_t} C_t^j + \sum_{k\in m_t} \left\| g_t^k \right\|_1 \cdot \max\left(0, 1 - \frac{C_t^k}{||g_t^k||_2}\right)\right] + \lambda f_{w_{t+1}}, \tag{16}$$

where λ is the Lagrangian multiplier for constraint (19).

(2) Subsequently, during each training round of differentially private federated learning, the clipping values and Lagrangian multiplier are updated using gradient descent and gradient

ascent, respectively, according to:

$$C_t^j = C_t^j - \zeta \begin{cases} \sum_{k\in m_t} p_k \nabla_{C_t^j} \mathcal{L}_k(w_{t+1}) + \gamma d\sqrt{1/\pi}\sigma + \gamma ||g_t^j||_1 \cdot \nabla_{C_t^j} \max(0, 1 - \frac{C_t^j}{||g_t^j||_2}) \\ +\lambda \nabla_{C_t^j} f(w_{t+1}) + cf(w_{t+1}) \nabla_{C_t^j} f(w_{t+1}), \qquad \text{if } C_t^j = \max_{k\in m_t} C_t^k \\ \sum_{k\in m_t} p_k \nabla_{C_t^j} \mathcal{L}_k(w_{t+1}) + \gamma ||g_t^j||_1 \cdot \nabla_{C_t^j} \max(0, 1 - \frac{C_t^j}{||g_t^j||_2})] + \lambda \nabla_{C_t^j} f(w_{t+1}) \\ +cf(w_{t+1}) \nabla_{C_t^j} f(w_{t+1}), \qquad \text{otherwise} \end{cases} \tag{17}$$

$$\lambda = \lambda + \xi f(w_{t+1}). \tag{18}$$

Here, ζ and ξ represent the learning rates for the clipping values and Lagrangian multiplier, respectively. Intuitively, these updates aim at driving C_t^k and consequently the updated parameter after the differential privacy mechanism w_{t+1} to optimize (14). The term $\sum_{k\in m_t} p_k \nabla_{C_t^j} \mathcal{L}_k(w_{t+1})$ enforces loss minimization, while the term $\gamma d\sqrt{1/\pi}\sigma + \gamma ||g_t^j||_1 \cdot \nabla_{C_t^j} \max(0, 1 - \frac{C_t^j}{||g_t^k||_2})$ optimizes the inherent tradeoff in the differential privacy mechanism for fairness mitigation. λ and $cf(w_{t+1})$ control the extent of penalty when the fairness constraint is violated. Next, we discuss how to calculate each term in the above update method. Since w_{t+1} is a function of C_t^k, which changes whenever we update the value of C_t^k, it becomes impractical for the central server to request $\frac{\partial \mathcal{L}_k(w_{t+1})}{\partial w_{t+1}}$ to compute $\nabla_{C_t^j} \mathcal{L}_k(w_{t+1})$ every time we update the clipping values. Doing so would result in significant computation and communication overhead. To circumvent this issue, we approximate the value of $\mathcal{L}_k(w_{t+1})$ using a fixed parameter w_t, yielding $\mathcal{L}_k(w_{t+1}) \approx \mathcal{L}_k(w_t) + \nabla_{w_t} \mathcal{L}_k(w_t)^T (w_{t+1} - w_t) = \mathcal{L}_k(w_t) + \langle g_t^k, w_{t+1} - w_t \rangle$. Therefore, the calculation of $f(w_{t+1})$ and the derivatives in the update rule of the clipping values can be performed as follows:

(i) For $f(w_{t+1})$:

$$f(w_{t+1}) \approx \begin{cases} \max_{j\in m_t} & |\sum_{k\in m_t} \frac{1}{|m_t|}(\mathcal{L}_k(w_t) + \langle g_t^k, w_{t+1} - w_t \rangle) - (\mathcal{L}_j(w_t) + \langle g_t^j, w_{t+1} - w_t \rangle)| - \alpha, \\ & \text{if } \max_{j\in m_t} |\sum_{k\in m_t} \frac{1}{|m_t|}(\mathcal{L}_k(w_t) + \langle g_t^k, w_{t+1} - w_t \rangle) - (\mathcal{L}_j(w_t) \\ & +\langle g_t^j, w_{t+1} - w_t \rangle)| > \alpha \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

(ii) For $\sum_{k\in m_t} p_k \nabla_{C_t^j} \mathcal{L}_k(w_{t+1})$:

$$\sum_{k\in m_t} p_k \nabla_{C_t^j} \mathcal{L}_k(w_{t+1}) \approx \frac{\partial \sum_{k\in m_t} p_k (\mathcal{L}_k(w_t) + \langle g_t^k, w_{t+1} - w_t \rangle)}{\partial C_t^j}$$

$$= \frac{\partial \sum_{k\in m_t} p_k \langle g_t^k, -\eta \sum_{k\in m_t} \frac{p_k}{q} g_t^k \cdot \min(1, \frac{C_t^k}{||g_t^k||_2}) \rangle}{\partial C_t^j}, \tag{20a}$$

$$= \left\langle \sum_{k\in m_t} p_k g_t^k, -\eta \frac{p_j}{q} g_t^j \nabla_{C_t^j} \min\left(1, \frac{C_t^j}{||g_t^j||_2}\right) \right\rangle. \tag{20b}$$

(iii) For $\nabla_{C_t^i} f(w_{t+1})$, we use ω as a shorthand for $\text{sign}[\sum_{k\in m_t} \frac{1}{|m_t|}(\mathcal{L}_k(w_t) + \langle g_t^k, w_{t+1} - w_t \rangle) - (\mathcal{L}_j(w_t) + \langle g_t^j, w_{t+1} - w_t \rangle)]$, where j represents the client with the highest degree of

ALGORITHM 1: Fair and Differentially-Private Federated Learning

Input: client sample probability q, noise scale σ, number of training rounds T, C_t^k update rate ζ, λ update rate ξ
Output: w_T
Initialize: model parameter w_0

for $t \in [0, T-1]$ **do**
  sample users with probability q to form sampled client set m_t;
  for *each client* $k \in [1, m_t]$ *in parallel* **do**
    Calculate $\mathcal{L}_k(w_t)$;
    g_t^k = ClientUpdate(k, w_t);
    $l_t^k = ||g_t^k||_2$;
  end
  Initialize $C_t^k = l_t^k, \lambda = \lambda_0$;
  for *each client* $k \in m_t$ *in parallel* **do**
    for $\tau \in [\chi]$ **do**
      for *each client* $k \in [1, m_t]$ *in parallel* **do**
        Update C_t^k, λ according to (17) and (18) ;
      end
    end
  end
  $\mathbb{C}_t = \max_{j \in m_t} C_t^j$;
  $w_{t+1} = w_t - \eta(\sum_{k=1}^{m_t} \frac{p_k}{q} g_t^k \cdot \min(1, \frac{C_t^k}{||g_t^k||_2}) + \mathcal{N}(0, \mathbb{C}_t^2 \sigma^2))$;
end

unfairness:

$$\nabla_{C_t^i} f(w_{t+1}) \approx \begin{cases} \omega \cdot & (\langle \sum_{k \in m_t} \frac{1}{|m_t|} g_t^k - g_t^j, -\eta \frac{p_i}{q} g_t^i \nabla_{C_t^i} \min(1, \frac{C_t^i}{||g_t^i||_2}) \rangle, \\ & \text{if } \max_{j \in m_t} | \sum_{k \in m_t} \frac{1}{|m_t|} (\mathcal{L}_k(w_t) + \langle g_t^k, w_{t+1} - w_t \rangle) \\ & -(\mathcal{L}_j(w_t) + \langle g_t^j, w_{t+1} - w_t \rangle)| > \alpha \\ 0, & \text{otherwise} \end{cases} \tag{21a}$$

During each training round, we iteratively apply the updates (17) and (18) multiple times at the central server to optimize problem (14). Clients do not experience any additional computation or communication overhead, except for transmitting the negligible-sized loss value $\mathcal{L}_k(w_t)$. The MMDM algorithm ensures convergence to a constrained minimum provided that the parameter initialization lies within the vicinity of each constrained minimum and the parameters remain bounded.

Our complete algorithm to achieve fair, differentially-private federated learning is given in Algorithm 1. The *ClientUpdate* procedure is given in Algorithm 2, which is carried out on each sampled client for training local dataset for E epochs, with the latest model parameters received from the server. At the same time, the clipping value C_t^k of sampled clients is updated according to (17) after E epochs of local training. We want to point out that, the unclipped gradient and C_t^k of each client are only known to the central server and will not be distributed to clients, and extra differential privacy mechanism is not needed to protect their privacy.

Next, we discuss the privacy cost incurred by our method. The moment accountant [2] method requires selecting each user independently with probability q rather than always selecting a fixed

ALGORITHM 2: ClientUpdate

Input: client index k, global parameter w_t
Output: g_k

$w = w_t$;
for *each epoch* $\in [1, E]$ *(E is local training epoch number)* **do**
  for *batch* $b \in B$ *(B is local batch number)* **do**
    $w = w - \eta \nabla \mathcal{L}_k(w; b)$ // η: local learning rate
  end
end
$g_t^k = (w - w_t) * \frac{-1}{\eta}$;
return g_t^k;

number of clients [28]. Based on this client sampling strategy, we have the following result on privacy cost.

Theorem 3. *There exist constants c_1 and c_2 such that for $\epsilon < c_1 q^2 T$, Algorithm 1 satisfies (ϵ, δ)-differential privacy with $\sigma \geq c_2 \frac{q\sqrt{T \log(1/\delta)}}{\epsilon}$.*

To provide the state-of-the-art result of σ which ensures (ϵ, δ)-differential privacy given values of ϵ and δ, we utilize the result of the moment accountant method [2]: there exist constants c_1, c_2, sampling probability q and the number of steps T such that for function f with sensitivity smaller than 1 and any $\epsilon < c_1 q^2 T$, $\mathcal{M}(x) = f(x) + \mathcal{N}(0, \sigma^2 \mathbf{I})$ satisfies (ϵ, δ)-differential privacy for any $\delta > 0$ if we choose $\sigma \geq c_2 \frac{q\sqrt{T \log(1/\delta)}}{\epsilon}$.

Lemma 1. *Suppose that a differential privacy mechanism $\mathcal{M}(x) = f(x) + Z$ satisfies (ϵ, δ)-differential privacy, with the clipping value $S_f \leq 1$, $Z \sim \mathcal{N}(0, \sigma^2)$. For mechanism $\bar{\mathcal{M}}(x) = \bar{f}(x) + \bar{Z}$ with $S_{\bar{f}} \leq S$, it also satisfies (ϵ, δ)-differential privacy if $\bar{Z}$ is sampled according to distribution $\mathcal{N}(0, \bar{\sigma}^2)$ with $\bar{\sigma} \geq \sigma S$.*

Proof. We first provide the following relationship [4]: For any adjacent dataset d, d', a mechanism is (ϵ, δ)-differential privacy if and only if $\Pr(l_{\mathcal{M},d,d'} \geq \epsilon) - e^{\epsilon} \Pr(l_{\mathcal{M},d,d'} \leq -\epsilon) < \delta$, where $l_{\mathcal{M},d,d'} = \ln \frac{\Pr(\mathcal{M}(d)=o)}{\Pr(\mathcal{M}(d')=o)}$ is termed as privacy loss.

We show that $l_{\mathcal{M},d,d'}$ is a Gaussian random variable for any function f when noises are sampled from Gaussian distribution. Without loss of generality, we suppose the output of $f(x)$ is m-dimensional, and each dimension of Z is sampled independently from $\mathcal{N}(0, \sigma^2)$. Consider the worst case, d and d' differ in one entry with the largest sensitivity. Suppose $(Z_1, Z_2, \ldots, Z_m) = o - f(d)$ and we have $s_i = f(d)_i - f(d')_i, \forall i \in [1, m]$, $\sqrt{\sum_{i=1}^m s_i^2} = S_f$. So we have $(Z_1 + s_1, \ldots, Z_m + s_m) = o - f(d')$. Next, the privacy loss is

$$\ln \frac{\Pr(\mathcal{M}(d) = o)}{\Pr(\mathcal{M}(d') = o)} = \ln \frac{\prod_{i=1}^m (\frac{1}{\sqrt{2\pi}\sigma} \exp(\frac{-Z_i^2}{2\sigma^2}))}{\prod_{i=1}^m (\frac{1}{\sqrt{2\pi}\sigma} \exp(\frac{-(Z_i + s_i)^2}{2\sigma^2}))}$$
$$= \sum_{i=1}^m \left(\frac{s_j^2}{2\sigma^2} \right) + \sum_{i=1}^m \left(\frac{Z_i s_i}{\sigma^2} \right).$$

Since $Z_i \sim \mathcal{N}(0, \sigma^2)$, according to the fact that if $X \sim \mathcal{N}(0, \sigma^2)$, $aX + b \sim \mathcal{N}(b, a^2\sigma^2)$ and if $X \sim \mathcal{N}(\mu_x, \sigma_x^2), Y \sim \mathcal{N}(\mu_y, \sigma_y^2), X + Y \sim \mathcal{N}(\mu_x + \mu_y, \sigma_x^2 + \sigma_y^2)$, we can get that $l_{\mathcal{M},d,d'} \sim \mathcal{N}(\frac{S_f^2}{2\sigma^2}, \frac{S_f^2}{\sigma^2})$.

Let $X = \frac{S_f^2}{\sigma^2}$. We have $l_{\mathcal{M},d,d'} \sim \mathcal{N}(\frac{X}{2}, X)$. According to the conclusion of [37], we know that function $\Pr(l_{\mathcal{M},d,d'} \geq \epsilon) - e^{\epsilon}\Pr(l_{\mathcal{M},d,d'} \leq -\epsilon)$ is monotonically increasing in X.

Back to the functions in Lemma 1, function $f(x)$ with $S_f \leq 1$ satisfying (ϵ, δ)-differential privacy must have $\Pr(l_{\mathcal{M},d,d'} \geq \epsilon) - e^{\epsilon}\Pr(l_{\mathcal{M},d,d'} \leq -\epsilon) < \delta$, and the corresponding $X = \frac{1}{\sigma^2}$. In order to make the function $\bar{f}(x)$ with $S_{\bar{f}} \leq S$ also satisfy (ϵ, δ)-DP, we need the X in this scenario to be smaller than X value of $f(x)$, which equals to $\frac{S^2}{\bar{\sigma}^2} \leq \frac{1}{\sigma^2}$. □

Proof of Theorem 3. The differential privacy mechanism $\mathcal{M}$ in Algorithm 1 adds noises sampled from $\mathcal{N}(0, \sigma^2\mathbb{C}_t^2)$ to $\sum_{k=1}^{m_t} \frac{p_k}{q}\frac{g_k}{\cdot} \min(1, \frac{C_t^k}{||g_t^k||_2})$ (which is equivalent to $\bar{f}(x)$ in Lemma 1). Due to clipping operation and the constraint $C_t^k \leq \mathbb{C}_t$, the sensitivity of $\bar{f}(x)$ is upper bounded by $\mathbb{C}_t$. Given the result of Lemma 1 and moment accountant, since the noise variance of our mechanism is already $\sigma^2\mathbb{C}_t^2$, it is easy to know: in each training round, it suffices to have $\sigma = c_2\frac{q\sqrt{T\log(1/\delta)}}{\epsilon}$ given ϵ and δ to achieve (ϵ, δ)-differential privacy, which concludes the proof. □

7 Performance Evaluation

7.1 Fairness Evaluation

We implement our algorithm on TensorFlow [1], and run simulations on a machine with two NVIDIA RTX 4,090 GPUs and 48 GB RAM. We extensively study the empirical performance of our method in both convex and non-convex settings. The detailed setup is presented below.

7.1.1 Experimental Setup. **Models and Datasets.** We evaluate the performance of our differentially-private federated learning scheme using the same datasets and models as in prior work [23] on the fairness issue in federated learning. Specifically, we consider the following setups: (1) **Synthetic:** Linear regression model on a synthetic dataset [30] to introduce additional statistical heterogeneity. The synthetic dataset (x_k, y_k) for client k is generated by the following function: $y = argmax(\text{softmax}(W_k x + b_k))$, where $W_k \sim \mathcal{N}(u_k, 1)$, $b_k \sim \mathcal{N}(u_k, 1)$, and $u_k \sim \mathcal{N}(0, 1)$. The goal of federated learning is to learn a global model $y = argmax(\text{softmax}(Wx + b))$. There are a total of 100 clients, with the number of training samples on each client following a power-law distribution. (2) **Vehicle**: Vehicle sensor dataset collected from 23 distributed sensors, where each sensor corresponds to a client. A SVM model is utilized to perform a binary classification task on this dataset to predict the type of the vehicle. (3) **Sent140:** 1101 X accounts of tweets sourced from Sentiment140 [14], with each account treated as a client. The model architecture comprises two LSTM layers followed by one fully connected layer. The task involves analyzing text sentiment, which is formulated as a classification problem. For all three datasets, the data on each client is randomly partitioned into training, testing, and validation sets according to a ratio of 8:1:1. (4) **Fashion MNIST:** a dataset consisting of clothing images, with 60,000 samples in the training set and 10,000 samples in the test set [34]. Samples belonging to the same class label are treated as a single client, resulting in 10 distinct classes. The logistic regression model used is the same as in [23]. (5) **CIFAR-10:** the CIFAR-10 dataset consists of colored images categorized into 10 classes, with 6,000 images per class [21]. In this setup, images from the same class are assigned to individual clients. A 4-layer **convolutional neural network** (**CNN**) is employed for model training [27].

Baselines. We compare our algorithm against four baselines, **each of which integrates DP-SGD [2] into its gradient aggregation step, using the median norm of the gradients from sampled clients as the clipping value applied to all clients**: (1) FedAvg [27], a widely employed federated learning algorithm, samples a subset of clients according to p_k and aggregates gradients from them to update the global model. (2) q-FedAvg [23], which utilizes the objective

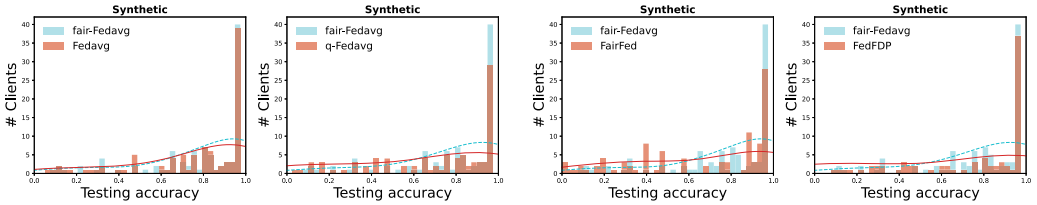


Fig. 2. Accuracy-client histogram on synthetic dataset.

$\sum_{k=1}^{m} \frac{p_k}{q+1} \mathcal{L}_k^{q+1}(w)$ in federated learning to ensure fairness. (3) FairFed [10], originally designed to promote demographic fairness, adjusts the aggregation weights for different groups based on both local and global fairness performance. We adapt this approach to the fairness definition used in our work by updating the aggregation weights to prioritize clients experiencing worse fairness treatment from the learned model. Specifically, the weight for client k at iteration $t+1$ is updated as $w_{t+1}^k = w_t^k - \beta(\ell - \ell_k)$, where ℓ represents the average loss across all clients and ℓ_k denotes the loss for client k. (4)FedFDP [24], addresses unfairness in federated learning by incorporating an unfairness penalty term, $\frac{\lambda}{2} \sum_{i=1}^{N}(F_i(w) - F(w))$, into the loss function. The weight λ is optimized to enhance model convergence under local differential privacy. In our work, we adapt this approach to a global differential privacy mechanism and perform a grid search to find the optimal λ. *fair-FedAvg* denotes the result of our algorithm. No differential privacy mechanism is applied on the client side to isolate the impact of global differential privacy, which is the focus of our work.

Default setting and Hyperparameters. The number of training rounds for Synthetic, Vehicle and Sent140 datasets are 1,000, 200, and 100, respectively, to ensure model convergence and small privacy cost. We adhere to well-tuned learning rates and batch sizes for the FedAvg and q-FedAvg algorithms. Specifically, for Synthetic, Vehicle, and Sent140, the learning rates are set to 0.1, 0.01, and 0.03, respectively. For the Fashion MNIST and CIFAR-10 datasets, the number of training rounds is set to 800, with a learning rate of 0.1. The batch sizes for these datasets are 10, 64, 32, 64, and 64 respectively. The local training epoch number E is set to 1, and the noise scale σ of the differential privacy mechanism is set to 1.0. For each round, the initial value of C_t^k is set to be the gradient norm of client k, and the learning rate ζ for it is set to be the minimum value between 0.0001 and the square of the minimum gradient norm across all clients, to prevent the value of $\frac{1}{||g_t^k||_2}$ in the gradient of $\sum_{k \in m_t} p_k \nabla_{C_t^j} \mathcal{L}_k(w_{t+1})$ and $\nabla_{C_t^j} f(w_{t+1})$ from dominating the update of clipping values. The initial λ value at each round is set to 30, and the learning rate ξ for it is 0.005. The optimal value of q in the objective of q-FedAvg follows the choice in [23], taking values of 1, 5, and 1 for the Synthetic, Vehicle, and Sent140 datasets, respectively. The number of local updates χ for C_t^k and λ is 5.

Metrics. We use the following metrics for fairness comparison: (i) Accuracy-client histogram: We partition the accuracy range [0,1] into 40 bins and tally the number of clients whose test accuracy falls into each bin. A more concentrated histogram indicates better fairness among clients. (ii) Mean of the test accuracy expectation across all clients, and variance of test accuracy[2] across all clients.

Figures 2 to 6 shows the test accuracy achieved among the clients when training the corresponding models for Synthetic, Vehicle, Sent140, Fashion MNIST and CIFAR-10 separately. The blue bars denote the test accuracy distribution among all clients with our method, while the red bars represent baselines. The brown colored areas indicate overlaps between our method and the baselines. In generally, in all empirical studies, our method effectively mitigates unfairness by reducing the

[2]We multiply this variance by 10,000 to facilitate comparison, as the accuracy variance is typically a very small number.

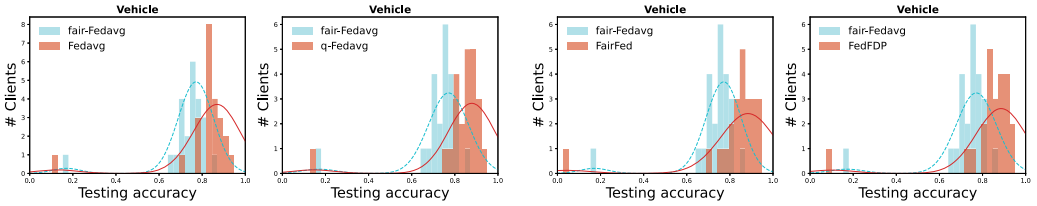


Fig. 3. Accuracy-client histogram on vehicle dataset.

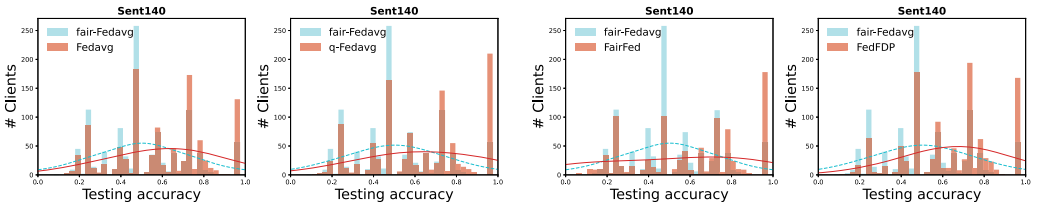


Fig. 4. Accuracy-client histogram on Sent140 dataset.

number of clients experiencing extreme test accuracy, both high and low, therefore at the expense of a slightly reduced overall test accuracy. While q-FedAvg utilizes a fairness-aware objective for gradient calculation, its efficacy in mitigating unfairness diminishes when the differential privacy mechanism is applied after gradient calculation, leading to a fairness performance similar to that of FedAvg. However, the exponential form of its aggregate loss prioritizes clients with extremely low accuracy, thereby improving their performance and resulting in higher overall accuracy compared to other methods. The comparison of fairness performance among the learned models across these methods underscores the importance of adjusting operations within the differential privacy mechanism when addressing fairness issues in differentially private stochastic gradient descent.

For the Synthetic dataset (Figure 2), all five methods achieve an average test accuracy between 0.7 and 0.8. Fair-FedAvg improves fairness by reducing the number of clients experiencing extremely low accuracy and maintaining more clients' accuracy within the 0.6 to 1.0 range, compared to FedAvg, q-FedAvg, FairFed and FedFDP. In Figure 3, for the Vehicle dataset, our method, fair-FedAvg, keeps all clients' accuracy within the range of 0.6 to 0.8, although with a slight decrease in overall model accuracy. In contrast, FedAvg, q-FedAvg, FairFed, and FedFDP exhibit a much broader accuracy distribution range. A similar trend is observed in the Sent140 dataset (Figure 4), where fair-FedAvg maintains a more concentrated accuracy distribution by aligning more clients' accuracy around the overall model's expected accuracy. Meanwhile, FedAvg, q-FedAvg, FairFed, and FedFDP display a more dispersed accuracy distribution, spanning a broader range of 0.2 to 1.0. For the Fashion MNIST dataset (Figure 5), our method delivers performance comparable to q-FedAvg while strictly outperforming all other baselines in terms of fairness by maintaining a narrower client accuracy distribution. On the CIFAR-10 dataset (Figure 6), our method achieves superior fairness by keeping the accuracy distribution of all clients between 0.4 and 0.6. The evaluation results across all five datasets support the conclusion from Theorem 1, which suggests that while fairness mitigation methods are applied during training, the differential privacy mechanism must be designed with fairness in mind.

Table 2 presents the mean of accuracy, the average accuracy of the worst 10% clients, the average accuracy of the best 10% clients and the variance of accuracy distribution across all clients. The results match our observation in Figures 2 to 6. Specifically, to promote fairness, we dynamically adjust the clipping value in DP during each training iteration to minimize the maximum

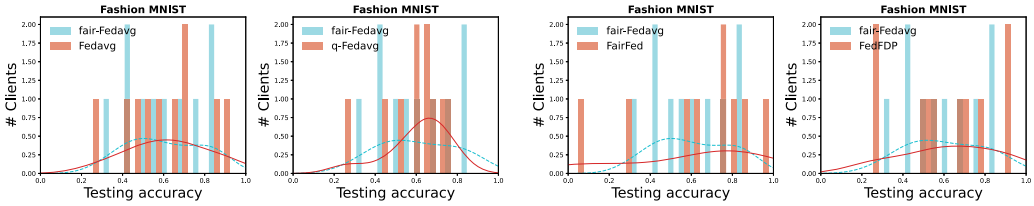


Fig. 5. Accuracy-client histogram on fashion MNIST dataset.

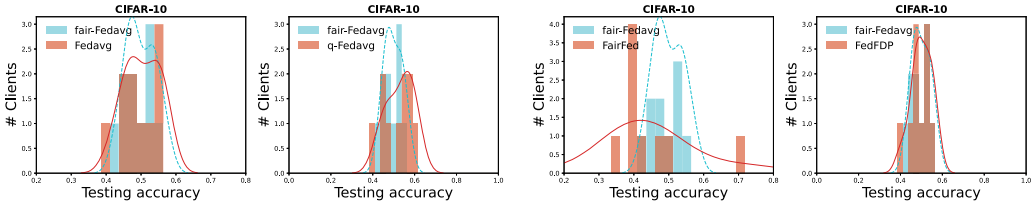


Fig. 6. Accuracy-client histogram on CIFAR-10 dataset.

Table 2. Statistics of the Test Accuracy

Dataset	Mechanism	Mean	Worst 10%	Best 10%	Variance
Synthetic	FedAvg	75.8%	10.8%	100%	844.77
	q-FedAvg	63.3%	0.0%	100%	1,290.33
	FairFed	62.4%	0.0%	100%	1,120.52
	FedFDP	68.3%	0.0%	100%	1,605.95
	fair-FedAvg	**78.5%**	**15.5%**	100%	**735.13**
Vehicle	FedAvg	83.3%	43.2%	94.3%	248.95
	q-FedAvg	84.5%	**46.7%**	94.8%	242.77
	FairFed	84.3%	38.3%	96.0%	323.88
	FedFDP	**85.0%**	43.2%	**96.5%**	279.67
	fair-FedAvg	74.5%	42.4%	85.4%	**170.14**
Sent140	FedAvg	59.9%	15.7%	100%	615.62
	q-FedAvg	62.3%	14.4%	100%	729.80
	FairFed	52.6%	0.0%	100%	1,149.0
	FedFDP	**64.9%**	**20.7%**	100%	568.0
	fair-FedAvg	50.0%	9.5%	90.7%	**537.19**
Fashion MNIST	FedAvg	62.1%	28.5%	92.4%	310.55
	q-FedAvg	60.9%	26.9%	87%	**201.28**
	FairFed	58.8%	0.0%	**100%**	1,070.34
	FedFDP	**62.6%**	27.4%	92.9%	497.45
	fair-FedAvg	60.6%	**32.2%**	85.6%	240.96
CIFAR-10	FedAvg	50.3%	42.0%	57.0%	23.01
	q-FedAvg	**52.9%**	42.0%	61.9%	37.76
	FairFed	41.1%	35.0%	**72.7%**	288.94
	FedFDP	50.1%	40.8%	57.4%	21.12
	fair-FedAvg	49.9%	**43.2%**	57.0%	**16.89**

Table 3. Average Duration of One Training Round

Mechanism	Synthetic	Vehicle	Sent140	Fashion MNIST	CIFAR-10
FedAvg	0.12s	0.85s	32.5s	1.0s	3.9s
q-FedAvg	0.14s	0.87s	34.0s	1.3s	4.5s
FairFed	0.13s	0.86s	33.6s	1.1s	4.2s
FedFDP	0.14s	0.86s	34.2s	1.2s	4.5s
fair-FedAvg	0.15s	0.96s	46.3s	1.5s	5.4s

performance gap between any individual client in the sampled client set and the expected performance across all sampled clients at that iteration, in pursuit of meeting constraint (14b). By doing so, our approach seeks to enhance fairness by minimizing the disparity between the most outlying client's performance and the expected model performance across all clients. This is demonstrated by the performance of the "worst 10%" of clients, as presented in Table 2 of the empirical study. Notably, our method outperforms all other baselines on 3 out of 5 datasets, while no other single method surpasses all baselines on more than one datasets. We also use accuracy variance across all clients as a fairness evaluation metric to capture the average performance disparity among clients. While (14b) does not directly minimize accuracy variance, its iterative application to *sampled clients* during training is expected to result in a more uniform accuracy distribution across all clients. To further substantiate this, we provide a comparative analysis of variance performance between our method and the baselines in Table 2, highlighting the superior fairness achieved by our approach.

7.2 Complementary Experiments

We next empirically study the computational efficiency, the effect of differential privacy to model fairness and the model convergence behavior of our method.

7.2.1 Computational Efficiency. We measure the average duration of one training round, which starts from client sampling and up-to-date global model distribution and ends with updates aggregating and global model updating. Differential privacy is applied before global model updating. The detailed results are given in Table 3. FedAvg is the fastest method due to its simple gradient calculation and the application of standard differential privacy. The increased time with our method is due to the computation and update of C_t^k and λ, which is derived based on the multiplication between terms involving g_t^k. The matrix multiplication can be efficiently computed when GPU is enabled. We observe that our method incurs only a moderate increase in training time compared to the baselines when training the five different models on different datasets.

7.2.2 Effect of DP. We further evaluate the impact of the differential privacy mechanism on other fairness mitigation methods to validate the conclusion of Theorem 1. The theorem suggests that, without careful design, differential privacy can degrade the performance of fairness mitigation solutions. Using the Vehicle dataset as an example, we evaluate the accuracy distribution and variance across all clients for the four baseline methods, with the results presented in Figure 7. These results align with our theoretical findings in Theorem 1. After applying the differential privacy mechanism, the fairness performance of all four baselines is affected to varying degrees, demonstrating that differential privacy can indeed impact the fairness of the learned model, regardless of whether fairness is considered during the training process. Therefore, the design of differential privacy should account for fairness considerations.

7.2.3 Convergence. Figure 8(a)–(c) show the convergence curves for FedAvg, q-Fedavg and fair-Fedavg on the Synthetic, Vehicle and Sent140 dataset respectively. Our approach achieves convergence at a comparable speed in terms of communication rounds to both FedAvg and q-FedAvg.

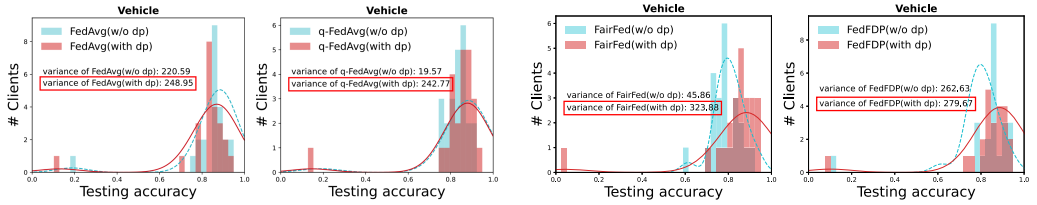
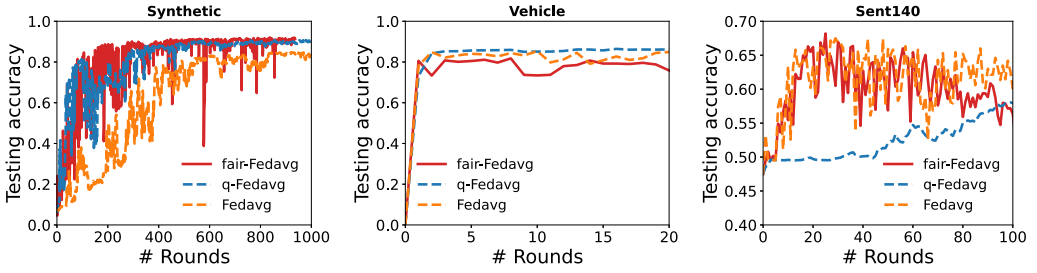


Fig. 7. The effect of DP on vehicle dataset.



(a) Convergence on Synthetic Dataset (b) Convergence on Vehicle Dataset (c) Convergence on Sent140 Dataset

Fig. 8. Convergence result.

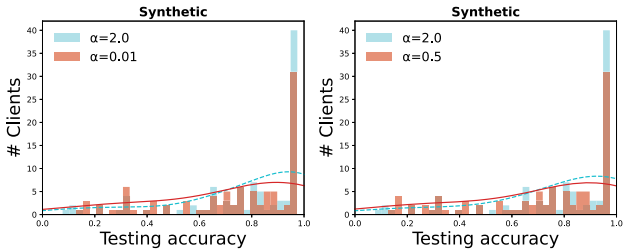


Fig. 9. The fairness level of the learned model when varying the value of α.

This finding confirms that our scheme does not adversely affect the proper convergence of the trained model. This is primarily attributed to our approach of designing fairness mitigation methods alongside model loss minimization, which is the first principle outlined in Section 6.

7.2.4 Effect of α. The hyperparameter α in the problem formulation (13) controls the desired level of fairness, directly influencing the performance of the final learned model. In this subsection, we vary its value to demonstrate the sensitivity of our method to this parameter. Specifically, using the Synthetic dataset as an example, we vary the value of α to approximately 0.01, 0.5, 2.0,[3] observing that the average loss decreases from roughly 4.0 to 1.0 over the course of the training process. The results are presented in Figure 9. A smaller value of α enforces a more stringent fairness requirement by reducing the number of clients with extremely low accuracy. However, this comes at the cost of degrading the overall accuracy of the final learned model. Therefore, α should be carefully selected based on prior knowledge of the trained model and the desired tradeoff between fairness and accuracy.

[3] 2.0 is the default α value for previous empirical results.

8 Conclusion

This article studies unfairness mitigation in differentially-private federated learning. By carefully analyzing the effect of clipping and noise addition on training dynamics and on the gradients of single client, we discover that this unfairness should be addressed with the consideration of the proper clipping values to be applied to different clients' gradients to control the contribution of clients during the training process. To achieve this, we formulate the optimization problem based on the theoretical analysis results to consider loss minimization and unfairness mitigation together, and propose an adaptive solution to adjust the clipping values for different clients. We provide theoretical guarantee for our differential privacy mechanism. Evaluation results show that our method can substantially improve fairness in federated learning and has favorable convergence result, compared to state-of-the-art differentially private federated learning algorithm.

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: A system for large-scale machine learning. In *Proceedings of the 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI}'16)*. 265–283.

[2] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 308–318.

[3] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. 2019. Differential privacy has disparate impact on model accuracy. In *Proceedings of the Advances in Neural Information Processing Systems*. 15479–15488.

[4] Borja Balle and Yu-Xiang Wang. 2018. Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *Proceedings of the International Conference on Machine Learning*. PMLR, 394–403.

[5] Abhishek Bhowmick, John Duchi, Julien Freudiger, Gaurav Kapoor, and Ryan Rogers. 2018. Protection against reconstruction and its applications in private federated learning. arXiv:1812.00984. Retrieved from https://arxiv.org/abs/1812.00984

[6] Theodora S. Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch Paschalidis, and Wei Shi. 2018. Federated learning of predictive models from federated electronic health records. *International Journal of Medical Informatics* 112, April 2018 (2018), 59–67.

[7] Rachel Cummings, Varun Gupta, Dhamma Kimpara, and Jamie Morgenstern. 2019. On the compatibility of privacy and fairness. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*. 309–315.

[8] Wei Du, Depeng Xu, Xintao Wu, and Hanghang Tong. 2021. Fairness-aware agnostic federated learning. In *Proceedings of the 2021 SIAM International Conference on Data Mining*. SIAM, 181–189.

[9] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. 214–226.

[10] Yahya H. Ezzeldin, Shen Yan, Chaoyang He, Emilio Ferrara, and A. Salman Avestimehr. 2023. Fairfed: Enabling group fairness in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 7494–7502.

[11] Tom Farrand, Fatemehsadat Mireshghallah, Sahib Singh, and Andrew Trask. 2020. Neither private nor fair: Impact of data imbalance on utility and fairness in differential privacy. In *Proceedings of the 2020 Workshop on Privacy-Preserving Machine Learning in Practice*. 15–19.

[12] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. 2015. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. 1322–1333.

[13] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. 2020. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in Neural Information Processing Systems* 33 (2020), 16937–16947.

[14] Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford* 1, 12 (2009), 2009.

[15] Xiuting Gu, Tianqing Zhu, Jie Li, Tao Zhang, and Wei Ren. 2020. The impact of differential privacy on model fairness in federated learning. In *Proceedings of the International Conference on Network and System Security*. Springer, 419–430.

[16] Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. 2018. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604* (2018).

[17] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. 2017. Deep models under the GAN: Information leakage from collaborative deep learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 603–618.

[18] Rui Hu, Yuanxiong Guo, Hongning Li, Qingqi Pei, and Yanmin Gong. 2020. Personalized federated learning with differential privacy. *IEEE Internet of Things Journal* 7, 10 (2020), 9530–9539.
[19] Matthew Jagielski, Michael Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan Ullman. 2019. Differentially private fair learning. In *Proceedings of the International Conference on Machine Learning*. 3000–3008.
[20] Jakub Konečnỳ, H. Brendan McMahan, Felix X. Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. 2016. Federated learning: Strategies for improving communication efficiency. arXiv:1610.05492. Retrieved from https://arxiv.org/abs/1610.05492
[21] Alex Krizhevsky, Geoffrey Hinton, and others. 2009. Learning multiple layers of features from tiny images. (2009). Retrieved from https://www.cs.utoronto.ca/~kriz/learning-features-2009-%0ATR.pdf
[22] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. 2020. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine* 37, 3 (2020), 50–60.
[23] Tian Li, Maziar Sanjabi, Ahmad Beirami, and Virginia Smith. 2019. Fair resource allocation in federated learning. In *Proceedings of the International Conference on Learning Representations*.
[24] Xinpeng Ling, Jie Fu, Zhili Chen, Kuncan Wang, Huifa Li, Tong Cheng, Guanying Xu, and Qin Li. 2024. FedFDP: Federated learning with fairness and differential privacy. arXiv:2402.16028. Retrieved from https://arxiv.org/abs/2402.16028
[25] Lingjuan Lyu, Yitong Li, Karthik Nandakumar, Jiangshan Yu, and Xingjun Ma. 2022. How to democratise and protect AI: Fair and differentially private decentralised deep learning. *IEEE Transactions on Dependable and Secure Computing* 19, Mar.-Apr. (2022), 1003–1017.
[26] Chuan Ma, Jun Li, Ming Ding, Howard H. Yang, Feng Shu, Tony Q. S. Quek, and H. Vincent Poor. 2020. On safeguarding privacy and security in the framework of federated learning. *IEEE Network* 34, 4 (2020), 242–248.
[27] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the Artificial Intelligence and Statistics*. 1273–1282.
[28] H. B. McMahan, D. Ramage, K. Talwar, et al. 2018. Learning differentially private recurrent language models. *International Conference on Learning Representations*. 2018.
[29] Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. 2019. Agnostic federated learning. In *Proceedings of the International Conference on Machine Learning*. PMLR, 4615–4625.
[30] Ohad Shamir, Nati Srebro, and Tong Zhang. 2014. Communication-efficient distributed optimization using an approximate newton-type method. In *Proceedings of the International Conference on Machine Learning*. PMLR, 1000–1008.
[31] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. 2016. Stealing machine learning models via prediction apis. In *Proceedings of the 25th {USENIX} Security Symposium ({USENIX} Security 16)*. 601–618.
[32] Stacey Truex, Nathalie Baracaldo, Ali Anwar, Thomas Steinke, Heiko Ludwig, Rui Zhang, and Yi Zhou. 2019. A hybrid approach to privacy-preserving federated learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*. 1–11.
[33] Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H. Yang, Farhad Farokhi, Shi Jin, Tony Q. S. Quek, and H. Vincent Poor. 2020. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security* 15 (2020), 3454–3469.
[34] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. arXiv:1708.07747. Retrieved from https://arxiv.org/abs/1708.07747
[35] Zuobin Xiong, Zhipeng Cai, Daniel Takabi, and Wei Li. 2021. Privacy threat and defense for federated learning with non-iid data in AIoT. *IEEE Transactions on Industrial Informatics* 18, 2 (2021), 1310–1321.
[36] Depeng Xu, Shuhan Yuan, and Xintao Wu. 2019. Achieving differential privacy and fairness in logistic regression. In *Companion Proceedings of The 2019 World Wide Web Conference*. 594–599.
[37] Zhiying Xu, Shuyu Shi, Alex X. Liu, Jun Zhao, and Lin Chen. 2020. An adaptive and fast convergent approach to differentially private deep learning. In *Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 1867–1876.
[38] Daniel Yue Zhang, Ziyi Kou, and Dong Wang. 2020. Fairfl: A fair federated learning approach to reducing demographic bias in privacy-sensitive classification models. In *Proceedings of the 2020 IEEE International Conference on Big Data*. IEEE, 1051–1060.
[39] Wei Zhang and Xiang Li. 2021. Federated transfer learning for intelligent fault diagnostics using deep adversarial networks with data privacy. *IEEE/ASME Transactions on Mechatronics* 27, 1 (2021), 430–439.
[40] Wei Zhang, Xiang Li, Hui Ma, Zhong Luo, and Xu Li. 2021. Federated learning for machinery fault diagnosis with dynamic validation and self-supervision. *Knowledge-Based Systems* 213, 15 February 2021 (2021), 106679.

Received 25 April 2024; revised 31 December 2024; accepted 7 March 2025