FedLF: Layer-Wise Fair Federated Learning

Zibin Pan[1,2], **Chi Li**[1,4], **Fangchen Yu**[1], **Shuyi Wang**[1,2], **Haijin Wang**[1],
Xiaoying Tang[*1,2,3], **Junhua Zhao**[*1,2]

[1] The School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, China
[2] The Shenzhen Institute of Artificial Intelligence and Robotics for Society
[3] The Guangdong Provincial Key Laboratory of Future Networks of Intelligence
[4] Shenzhen Research Institute of Big Data
zibinpan@link.cuhk.edu.cn, chili@link.cuhk.edu.cn, fangchenyu@link.cuhk.edu.cn, shuyiwang@link.cuhk.edu.cn,
haijinwang@link.cuhk.edu.cn, tangxiaoying@cuhk.edu.cn, zhaojunhua@cuhk.edu.cn

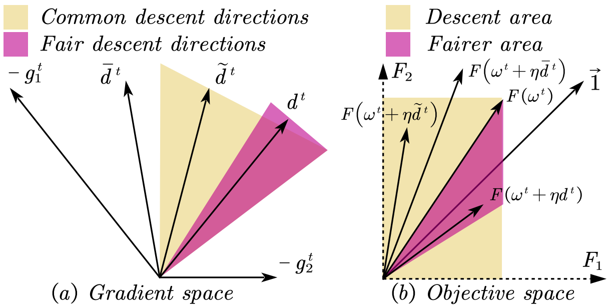
Abstract

Fairness has become an important concern in Federated Learning (FL). An unfair model that performs well for some clients while performing poorly for others can reduce the willingness of clients to participate. In this work, we identify a direct cause of unfairness in FL - the use of an unfair direction to update the global model, which favors some clients while conflicting with other clients' gradients at the model and layer levels. To address these issues, we propose a layer-wise fair Federated Learning algorithm (FedLF). Firstly, we formulate a multi-objective optimization problem with an effective fair-driven objective for FL. A layer-wise fair direction is then calculated to mitigate the model and layer-level gradient conflicts and reduce the improvement bias. We further provide the theoretical analysis on how FedLF can improve fairness and guarantee convergence. Extensive experiments on different learning tasks and models demonstrate that FedLF outperforms the SOTA FL algorithms in terms of accuracy and fairness. The source code is available at https://github.com/zibinpan/FedLF.

1 Introduction

Federated Learning (FL) is a popular machine learning paradigm that allows clients to collaboratively train a global model without sharing data. It is a promising methodology to address data island and privacy issues (Yu et al. 2022), where clients are able to obtain a more generalized model than local learning. However, due to many factors such as data heterogeneity, intermittent client participation, client dropout, etc., the global FL model is prone to be unfair since it favors part of clients and may perform poorly on some others (Li et al. 2020b; Kairouz et al. 2021), which would reduce their willingness to participate in FL.

Improving fairness in FL has drawn increased attention recently (Shi, Yu, and Leung 2021; Zhou et al. 2021; Pan et al. 2023). (Mohri, Sivek, and Suresh 2019) proposed AFL, aiming to prevent the overfitting of certain clients at the expense of others. Recent studies have explored proactive ways to enhance fairness by mitigating model-level gradient conflicts among clients (Wang et al. 2021; Hu et al. 2022). (Pan et al. 2023) developed this approach and proposed FedMDFG to calculate a fair-descent direction for the model update that doesn't conflict with clients' gradient in the model level, which made progress in enhancing FL fairness.



(a) *Gradient space*

(b) *Objective space*

Figure 1: A demo of two clients. (a) shows gradients and directions. Obtained by previous algorithms, direction $\bar{d}^t$ conflicts with client 2. d^t and $\tilde{d}^t$ are directions obtained by FedLF with/without the fair-driven objective. (b) describes several objective vectors of the models updated by different directions. $F(\omega^t)$ depicts clients' objectives at round t.

In this work, we extend this proactive way to improve fairness in FL. We observe that in addition to the model-level gradient conflict, there are also layer-level gradient conflicts that should be mitigated when determining a fair direction for the model update. In conclusion, we summarize that there exist three primary challenges when computing a fair direction: model-level gradient conflicts, improvement bias, and layer-level gradient conflicts.

Challenge 1: Model-level gradient conflict. Due to the data heterogeneity of clients, clients' gradients easily conflict with each other (Wang et al. 2021), i.e., at round t, there exist clients i, j whose gradients g_i^t and g_j^t satisfy $g_i^t \cdot g_j^t < 0$. When updating the model, a simple-aggregated direction $\bar{d}^t$ will easily conflict with some clients' gradients, i.e., $\bar{d}^t \cdot g_i^t > 0$, thus it will reduce the model's performance on those clients and harm fairness (see Fig. 1).

Challenge 2: Improvement bias. It is not enough to enhance fairness only by addressing the above challenge. Because despite a direction that doesn't conflict with each client's gradient can improve the model performance on

[*] Xiaoying Tang and Junhua Zhao are corresponding authors.

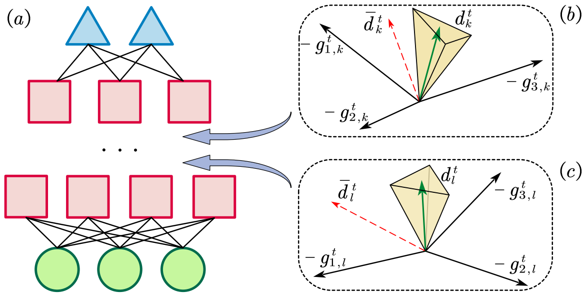


Figure 2: A demo of three clients. (a) shows the model. (b) and (c) describe clients' gradients that conflict at layer l and k. $g_{i,l}^t, g_{i,k}^t \in \mathbb{R}^3$ denote client i's gradient fragments at layer l and k. $\bar{d}^t$ is a conflicting direction obtained by previous methods. $\bar{d}_l^t$ and $\bar{d}_k^t$ in (b) and (c) are two fragments of $\bar{d}^t$ at layer l and k, which conflict with client 2 and 3 at these layers, i.e., $\bar{d}_l^t \cdot g_{2,l}^t > 0$, $\bar{d}_l^t \cdot g_{3,l}^t > 0$, $\bar{d}_k^t \cdot g_{2,k}^t > 0$, and $\bar{d}_k^t \cdot g_{3,k}^t > 0$. d^t is the direction obtained by FedLF, with fragments d_l^t and d_k^t lying in the yellow area, which depicts all feasible directions that do not conflict with clients' gradients at these layers.

each client, this improvement may be substantial for some clients but just marginal for others. Fig. 1 illustrates this issue, where $\tilde{d}^t$ is a common descent direction that reduces each client's local objective. However, the reduction in client 1's objective is much larger than that of client 2, resulting in a model that favors client 1, thereby reducing fairness.

To handle the above two challenges, FedMDFG applied the multiple gradient descent algorithm to a designed multi-objective optimization problem with a dynamic objective (see Equ.5 on its paper). But it would drive the model out of the fairer area in some cases and thus requires a necessary hyperparameter to determine the activation of the dynamic objective, which makes it difficult to prove the convergence (while FedMDFG only gave the convergence proof without considering the hyperparameter). We discuss this in more detail in Section 3.1. Differently, we propose a simpler but more effective fair-driven objective and give theoretical analysis on the convergence and convergence rate.

Challenge 3: Layer-level gradient conflict. In deep learning, different layers of a neural network can have different utilities (Yu et al. 2018; Ma et al. 2022; Lee, Zhang, and Avestimehr 2023). For instance, shallow layers primarily hone in on local feature extraction, while deeper layers focus on extracting global features. Therefore, simply aggregating or only mitigating gradient conflicts at the model level cannot prevent the obtained direction from favoring parts of clients at some layers, while drifting away from others, which would reduce the layer utility for those clients. Fig. 2 shows a demo of three clients, where their gradients conflict with each other at layers l and k. Then, the direction $\bar{d}^t$ obtained by previous FL methods favors client 1 while conflicting with clients 2 and 3 at layers k and l. This will cause the new model's parameters at these two layers to favor client 1 while drifting away from the others. In the experimental results of Table 2 and Table 3, we verify the existence of layer-level gradient conflicts and show that mitigating layer-level gradient conflicts can help improve FL fairness.

In this paper, we propose a layer-wise Fair Federated Learning (FedLF) algorithm that can compute a layer-wise fair direction to enhance fairness. First, we formulate a multi-objective optimization problem for FL, incorporating an effective fair-driven objective. Then, we design a layer-wise multiple gradient descent algorithm (LMGDA) to obtain such a direction that mitigates both the model and layer-level gradient conflicts. To the best of our knowledge, FedLF is the first one capable of identifying a layer-wise fair direction that does not conflict with clients' gradients at layers.

Our contributions are summarized as follows.

- We identify the layer-level gradient conflict that would impair the FL fairness. A layer-level conflicting direction would drift away from some clients and diminish the model layer utility on these clients.
- We formulate a multi-objective optimization problem with an effective fair-driven objective to improve the model performance on each client and enhance fairness.
- We propose FedLF that establishes a layer-wise fair direction to drive the FL model fairer. We theoretically analyze that it can mitigate the gradient conflicts among clients at both the model and the layer levels, and reduce the improvement bias.
- We conduct extensive experiments on multiple FL scenarios, validating that FedLF outperforms the SOTA approaches in terms of accuracy and fairness.

2 Background & Related Work

2.1 Background of Federated Learning

The traditional FL (McMahan et al. 2017) allows clients to collaboratively train a global model $\omega \in \mathbb{R}^n$ with the goal of minimizing the weighted average objective of clients:

$$\min_{\omega} \sum_{i=1}^{m} p_i F_i(\omega), \tag{1}$$

where $p_i \geq 0$, $\sum_{i=1}^{m} p_i = 1$, and m is the number of clients. $F_i(\omega)$ represents the local objective of client i, which is usually defined by a specific loss function such as cross-entropy loss and calculated across N_i samples by $F_i(\omega) = \sum_{j=1}^{N_i} \frac{1}{N_i} F_{i_j}(\omega)$. $F_{i_j}(\omega)$ is the loss on the j^{th} sample.

However, traditional FL easily suffers from performance degradation in heterogeneous settings since a simple aggregated direction for model updating may conflict with the gradients of some clients (Wang et al. 2021) while favoring other clients, and thus the model would be more unfair. So there is another way to consider FL as a multi-objective optimization problem (MOP) (Hu et al. 2022):

$$\min_{\omega} \ (F_1(\omega), F_2(\omega), \cdots, F_m(\omega)). \tag{2}$$

One typical gradient-based method to solve Problem (2) is Multiple Gradient Descent Algorithm (MGDA) (Désidéri 2012; Gebken, Peitz, and Dellnitz 2019). It updates the model at each round t by $\omega^{t+1} = \omega^t + \eta^t d^t$ with a step size η^t and a common descent direction d^t, which is computed by solving Problem (3) and satisfies $d^t \cdot g_i^t < 0, \forall i$, so that it can drive to reduce the local objectives. g_i^t is the local gradient of the model ω on data samples of client i, which is calculated by $g_i^t{=}\nabla F_i(\omega^t){=}\sum_{j=1}^{N_i}\frac{1}{N_i}\nabla F_{i_j}(\omega^t)$. MGDA stops when ω^t reaches the Pareto stationarity (Désidéri 2012).

$$\begin{aligned}(d^t,\alpha^t) = &\mathop{\arg\min}_{d^t\in\mathbb{R}^n,\alpha^t\in\mathbb{R}} \ \alpha^t+\tfrac{1}{2}\|d^t\|^2,\\ s.t.\quad & g_i^t\cdot d^t\le\alpha^t, i=1,\cdots,m\end{aligned}\tag{3}$$

Definition 1 (Pareto Stationarity): ω^* is called Pareto stationary iff $\exists\xi_i\geq 0, \sum_i\xi_i=1$, such that $\sum_i\xi_i g_i=0$.

Pan et al. (2023) successfully applied MGDA in FL, proposing FedMDFG by adding a dynamic objective to Problem (3), which firstly ensured obtaining a common descent direction that doesn't conflict with each client's gradient in the model level. But the layer-level gradient conflict still remains a significant challenge, which brings the layer-wise bias that reduces the layer's capability on some clients.

2.2 Fair Federated Learning

Our work follows the definition of the FL fairness summarized by (Li et al. 2021), where a model ω_1 is fairer than ω_2 if the standard deviation of ω_1's performance (e.g., test accuracy or local objective) across clients is smaller than that of ω_2. A general goal of fair FL is to train a model with better average and more uniform performance across clients (Li et al. 2020b; Wang et al. 2021; Pan et al. 2023).

Increasing fairness has attracted great interest in recent years. (1) Some previous works tried reweighting aggregation methods (Li et al. 2020b; Huang et al. 2020; Zhao and Joshi 2022a; Li et al. 2023). (2) Li et al. (2020a) reduced the update bias across clients to alleviate the negative impact of client drift in heterogeneous settings, and thus it may enhance fairness in a way. (4) The works of (Huang et al. 2022; Salazar et al. 2022) use momentum approaches. (5) Recently, the works of FedFV (Wang et al. 2021), FedMGDA+ (Hu et al. 2022) and FedMDFG (Pan et al. 2023) explore a proactive way of fair FL by trying to mitigate clients' model-level gradient conflicts. Our method also aims to mitigate this kind of conflict. But differently, we further mitigate the layer-level gradient conflicts among clients.

2.3 Layer-wise Federated Learning

Several previous works designed layer-wise approaches for FL. For example, (Lee, Zhang, and Avestimehr 2023) used a layer-wise model aggregation method to reduce the communication cost in FL; (Son, Kim, and Chung 2022) regularized parts of layers during FL training; Some works (Mei et al. 2021; Ma et al. 2022) employed the layer-wise aggregation to build personalized models in personalized FL. Contrary to these methods, we design a novel method to calculate a layer-wise fair direction to enhance fairness in FL.

3 The Proposed Approach

Our proposed FedLF aims to handle the challenges mentioned in Section 1 to obtain layer-wise fair directions to drive the FL model fairer. Algorithm 1 demonstrates the steps of FedLF. In each communication round t, after receiving the local gradient g_i^t and the training loss $F_i(\omega^t)$ of each online client i (Alg. 1, Line 4), the server computes the gradient g_P^t of the fair-driven objective. Direction fragments d_l^t for each layer l are then calculated in parallel to produce the layer-wise fair direction d^t (Alg. 1, Lines 6-13). Finally, after sending d^t to clients, the model is updated by $\omega^{t+1}{\leftarrow}\omega^t{+}\eta d^t$ both on the server and online clients.

Algorithm 1: **L**ayer-wise **F**air **Fed**erated Learning (**FedLF**)

```
Input: Initialize model parameters $\omega^0$, learning rate $\eta$.
 1: for $t = 0, 1, \cdots, T-1$ do
 2:     $S^t \leftarrow$ The set of online clients.
 3:     Broadcast $\omega^t$ to all client $i$, $i \in S^t, i \notin S^{t-1}$.
 4:     Server receives the gradient $g_i^t$ and the loss $F_i(\omega^t)$
        from each client $i \in S^t$, where $g_i^t = (\omega^t - \omega_i^t)/\eta$ and
        $\omega_i^t$ is updated by client $i$.
 5:     Calculate $g_P^t$ by Equ. (5).
 6:     for each layer $l \in L$ in parallel do
 7:         $d_l^t \leftarrow$ Calculate the direction fragment by Equ.(7).
 8:     end for
 9:     while $\exists\, l$ that $d_l^t = \vec{0}$ do
10:         $l \leftarrow$ Combine layer $l$ with its later/previous layer.
11:         Recompute $d_l^t$.
12:     end while
13:     $d^t \leftarrow$ concatenate all layer-wise directions $d_l^t$.
14:     Stop if $\|d^t\| = 0$.
15:     Rescale $d^t$ by $d^t \leftarrow d^t/\|d^t\| \cdot \|\frac{1}{|S_t|}\sum_i^{|S_t|} g_i^t\|$, and
        then broadcast it to all online clients.
16:     Both of the server and online clients update model
        parameters by $\omega^{t+1} \leftarrow \omega^t + \eta d^t$ .
17: end for
Output: Model parameters $\omega^t$.
```

3.1 Problem Formulation

We follow the idea of considering FL as a multi-objective optimization problem (Hu et al. 2022; Pan et al. 2023), which is shown in Problem (2). However, it cannot avoid the improvement bias among objectives. For example, there are two clients whose local objectives are $F_1(\omega^t)=4$ and $F_2(\omega^t)=5$. After updating the model, $F_1(\omega^{t+1})=1$ but $F_2(\omega^{t+1})=4$, thus there is a greater disparity between the objectives, which decreases fairness.

Toward this end, FedMDFG (Pan et al. 2023) added a dynamic objective $\min_\omega F(\omega)^T h^t$ to Problem (2) (see Equ.5 in their paper), where h^t denotes an opposite normalized vector of the projection of $\vec{1}$ on the normal plane of $L(\omega^t)$. They called it a fair-driven objective. However, this objective doesn't directly drive the model fairer, it would drive the model out of the fairer area if the local objectives are quite close. To handle it, they ran a step size line search to search for a smaller step size, which would bring extra communication costs. Besides, a hyperparameter was added to control whether to use the added objective, making it hard to guarantee the convergence, while they only proved the convergence without considering the effect of the hyperparameter.

Differently, we design a more effective fair-driven objective: $\min_\omega P(\omega) = -cos(\vec{1}, F(\omega))$, which aims to increase the cosine similarity between the vector $\vec{1}$ and the objective vector $F(\omega)=(F_1(\omega),\cdots,F_m(\omega))$. Hence, it can reduce the difference among the objectives. Therefore, the goal of FedLF is to optimize the following MOP instead of Problem (2) to render ω Pareto stationary.

$$\min_{\omega} \ (F_1(\omega), F_2(\omega), \cdots, F_m(\omega), P(\omega)). \tag{4}$$

Compared with the dynamic objective proposed in FedMDFG, our fair-driven objective retains the formulation as a static optimization problem, and thus we can easily analyze the convergence and the convergence rate (see Section 3.4). Note that for a non-Pareto stationary solution ω^t of Problem (2), there always exists a direction d^t, such that $g_i^t \cdot d^t < 0, \forall i \in \{1, \cdots, m\}$ and $g_P^t \cdot d^t < 0$, where g_P^t is the gradient of the fair-driven objective obtained by Equ.(5), $g^t = \text{concat}(g_1^t, \cdots, g_m^t)$. It indicates that the added fair-driven objective doesn't affect the convergence of FL. We provide the proof in detail in Appendix A.1.

$$g_P^t = \frac{g^t}{\|F(\omega^t)\|^2} \cdot \left(\frac{F(\omega^t)^T \vec{1} F(\omega^t)}{\|\vec{1}\|\|F(\omega^t)\|} - \frac{\vec{1}\|F(\omega^t)\|}{\|\vec{1}\|}\right). \tag{5}$$

In Fig. 1, we show that without the fair-driven objective, the obtained direction $\tilde{d}^t$ cannot prevent the objective vector of the updated model from being far away from $\vec{1}$ and thus decrease fairness, even though $\tilde{d}^t$ is common descent.

3.2 Layer-wise Fair Direction

In FedLF, we solve Problem (4) by iterating $\omega^{t+1} = \omega^t + \eta^t d^t$, where d^t is a layer-wise fair direction at t^{th} round. In this section, we discuss how to compute such a direction. We start by defining model-level gradient conflicts following (Wang et al. 2021) and layer-level gradient conflicts.

Definition 2 (Model-level Gradient Conflict): The gradients of client i and j conflict with each other iff $g_i \cdot g_j < 0$.

Besides, given that a deep learning model with multiple layers often exhibits distinct utility at each layer (Ma et al. 2022), such as data feature extraction, classification, etc., the heterogeneity of data across clients often leads to substantial variance in the gradient fragments at different layers. Consequently, they may easily come into conflict with one another.

Definition 3 (Layer-level Gradient Conflict): Let n_l be the dimension of the model parameters at layer l. $\sum_{l\in L} n_l = n$. L is a set of all l. Client i's gradient $g_i \in \mathbb{R}^n$ conflicts with client j's gradient $g_j \in \mathbb{R}^n$ at layer l iff $g_{i,l} \cdot g_{j,l} < 0$, where $g_{i,l} \in \mathbb{R}^{n_l}$ is the fragment of client i's gradient at layer l.

It is worth noting that the model/layer-level gradient conflicts also exist in the case of partial client participation or client dropout. We discuss more about it in Section 3.3.

Existing fair FL algorithms focus on model-level aggregation. They cannot prevent the direction fragment d_l^t, which is used to update the model parameters at layer l, from favoring some clients while conflicting with the others at some layers l, i.e., $d_l^t \cdot g_{i,l}^t > 0$, where $g_{i,l}^t$ denotes a fragment of client i's gradient (g_i^t) at the layer l in round t.

To mitigate the layer-level gradient conflicts, we need to ensure that the obtained direction d^t satisfies $d_l^t \cdot g_{i,l}^t < 0, \forall$ client i and layer l. To achieve this, we can solve the following problem (6) to obtain d_l^t for each layer l, where $g_{P,l}^t$ is the fragment of g_P^t at layer l, and then concatenate all d_l^t to obtain d^t. Different from the MGDA formula (3), it is done by layers, and it contains a constraint $g_{P,l}^t \cdot d_l^t < 0$ that is used to reduce the fair-driven objective.

$$\begin{aligned} (d_l^t, \alpha_l^t) = &\underset{d_l^t \in \mathbb{R}^{n_l}, \alpha_l^t \in \mathbb{R}}{\arg\min} \ \alpha_l^t + \tfrac{1}{2}\|d_l^t\|^2, \\ s.t. \quad & g_{i,l}^t \cdot d_l^t \leq \alpha_l^t, \ \forall \, i = 1, \cdots, m, \\ & g_{P,l}^t \cdot d_l^t \leq \alpha_l^t. \end{aligned} \tag{6}$$

Scalable Method to obtain d_l^t. Problem (6) itself does not scale well for the high dimensional decision space, because some layer l often contains millions of parameters. Thus, solving Problem (6) in this scale would be extremely slow. Inspired by (Fliege and Svaiter 2000; Pan et al. 2023) we achieve d_l^t in another way. Based on the KKT conditions:

$$d_l^t = -\left(\sum\nolimits_{i=1}^{m} \lambda_i g_{i,l}^t + \mu g_{P,l}^t\right), \ \sum\nolimits_{i=1}^{m} \lambda_i + \mu = 1, \tag{7}$$

where $\lambda_1, \cdots, \lambda_m, \mu \geq 0$ is the optimal solution of the dual problem of Problem (6):

$$\begin{aligned} \max_{\lambda_i, \mu} & -\tfrac{1}{2}\| \textstyle\sum_{i=1}^{m} \lambda_i g_{i,l}^t + \mu g_{P,l}^t\|^2 \\ s.t. \ & \textstyle\sum_{i=1}^{m} \lambda_i + \mu = 1, \\ & \lambda_i, \mu \geq 0, \forall i = 1, 2, \cdots, m. \end{aligned} \tag{8}$$

Therefore, we only need to solve a simple quadratic optimization problem (8) in $(m+1)$-dimension, which is not time-consuming. We report the actual computation time of FedLF in Appendix B.3.

Combine Layers. Inspired by (Gebken, Peitz, and Dellnitz 2019), the obtained d_l^t satisfies:

1. If $\exists \, \xi \in \mathbb{R}^{m+1} \geq \vec{0}$, $\sum_i \xi_i = 1$, such that $\sum_{i=1}^{m} g_{i,l}^t \xi_i + g_{P,l}^t \xi_{m+1} = \vec{0}$, then $d_l^t = \vec{0}$.
2. Otherwise, $g_{i,l}^t \cdot d_l^t < 0, i = 1, \cdots, m$, and $g_{P,l}^t \cdot d_l^t < 0$.

If $d_l^t = \vec{0}$ is satisfied for all layer $l \in L$, then $d^t = \vec{0}$ and thus the model ω^t reaches to Pareto stationarity. However, if $d_l^t = \vec{0}$ is satisfied only for part of $l \in L$, the parameters in these layers will stay in stagnation and stop updating. This case would happen if rank$(g_{1,l}^t, \cdots, g_{m,l}^t, g_{P,l}^t) < m$. When $n_l \geq m$, from a probabilistic point of view, it would have almost probability 1 that rank$(g_{1,l}^t, \cdots, g_{m,l}^t, g_{P,l}^t) = m$. But when some layers have only few parameters, $n_l < m$ may be satisfied and thus rank$(g_{1,l}^t, \cdots, g_{m,l}^t, g_{P,l}^t) < m$, making $d_l^t = \vec{0}$ and leading to the stagnation of parameters at layer l. To handle this, when $d_l^t = \vec{0}$, we combine layer l with its next layer (if there is no next layer, combine it with the previous layer) and then recompute d_l^t. If d_l^t is still $\vec{0}$, repeat the layer-combination until $d_l^t \neq \vec{0}$ or all layers are combined.

After calculating $d_l^t, \forall l \in L$, we combine them to build the direction d^t for the model update: d^t=concat$(\{d_l^t, \forall l \in L\})$. Ultimately, since optimizing Problem (8) can make the norm of d^t smaller, to prevent the step size from being affected by $\|d^t\|$, we scale d^t by $d^t = d^t/\|d^t\| \cdot \|\bar{d}\|$, where $\bar{d} = -\frac{1}{m}\sum_{i=1}^{m} g_i^t$ is a simple aggregated direction.

The obtained direction d^t satisfies:

1. If ω^t is Pareto stationary, then $d^t = \vec{0}$.

2. If ω^t is not Pareto stationary, then $g_i^t \cdot d^t < 0, i = 1, \cdots, m$, and $g_P^t \cdot d^t < 0$,

where $g_P^t \cdot d^t < 0$ indicates that d^t can drive to reduce the fair-driven objective. $g_i^t \cdot d^t < 0$ implies that d^t is a common descent direction that doesn't conflict with clients' gradients in model-level. Since $d_l^t \cdot g_{i,l}^t < 0$, in conclusion, d^t is a fair direction that can mitigate the model and layer-level gradient conflicts and reduce the improvement bias.

3.3 Improving Absent Client Fairness

Due to various factors, including partial client participation and intermittent client availability, etc., sometimes only parts of clients can participate in FL at each communication round (Abay et al. 2020; Cho et al. 2020). Given that the model update direction may conflict with the gradients of those absent clients if they were online. This could cause the model to perform significantly worse on these clients when they return (Wang et al. 2021) and thus harm fairness.

To mitigate the gradient conflicts in this regard, FedMDFG takes into account those who were online at the last communication round when calculating the update direction. In other words, it operates as if these clients are still online, using their historical gradients in the direction calculation. But this would still ignore those clients who have been absent for not too long, but more than one round.

Differently, we take into account those absent clients who were online from $t - \tau$ to $t - 1$ communication rounds when calculating the direction. $\tau = M/|S^t|$ is the expected length of time for each of the recorded clients to participate in FL one more time, where M is the number of recorded clients that have already joined in FL, and S^t is a set of online clients. These absent clients can be considered as temporarily absent. We follow (Wang et al. 2021) to estimate these absent clients' gradients according to their last gradients during round $t-\tau$ to $t-1$, and treat them as if they were still online to compute the update direction. This allows us to obtain a layer-wise fair direction d^t that does not conflict with the online clients' gradients and the absent clients' estimated gradients. In the ablation experiments (Section 4.3), we show that this strategy outperforms that of FedMDFG, since FedMDFG only considers those clients who have been absent for only one round (see M7). We also observe that we cannot consider all absent clients, since it is easily influenced by clients with long absences (see M8).

3.4 Convergence Analysis

We analyze the convergence of FedLF. Assume that all clients are online, according to Theorem 1, which is inspired by (Bertsekas 1999; Fliege and Svaiter 2000), the global model ω can converge to the Pareto stationarity.

Theorem 1. Suppose client i's objective $F_i(\omega)$ is Lipschitz smooth. $\forall i$, L_i denotes the corresponding Lipschitz constant. Let $\mathbb{L}$ be a set containing all $L_i, \forall i$, which is the Lipschitz constant of the fair-driven objective. If the model is updated by $\omega^{t+1} = \omega^t + \eta^t d^t$ with a non-zero direction d^t calculated by FedLF and the step size η^t in the bound:

$$0 < \eta^t \leq 2 \cdot \min_{L_i \in \mathbb{L}} \frac{|g_i^t \cdot d^t|}{L_i \|d^t\|^2}, \tag{9}$$

then FedLF converges to a Pareto stationary point sublinearly. If the local objective $F_i(\omega)$ is strongly convex, then FedLF converges linearly to a Pareto stationary point. The detailed proof can be seen in Appendix A.2.

4 Experiments

To evaluate fairness, we follow (Pan et al. 2023) to utilize a scale-invariant metric as the fairness indicator: $\arccos(\frac{A(\omega)\cdot\vec{1}}{\|A(\omega)\|\|\vec{1}\|})$. It is the inverse cosine value of the cosine similarity between $A(\omega)$ and $\vec{1}$, i.e., the angle between $A(\omega)$ and $\vec{1}$, where $A(\omega)$ denotes a vector that contains the test accuracy of the global model on each client. A lower value of the fairness indicator means that $A(\omega)$ is closer to $\vec{1}$, meaning that the model has a higher fairness capability.

Baselines. We first consider well-known FL approaches, such as FedAvg, FedProx (Li et al. 2020a), and some fair FL methods, including qFedAvg (Li et al. 2020b), AFL (Mohri, Sivek, and Suresh 2019), Ditto (Li et al. 2021), FedFV (Wang et al. 2021), DRFL (Zhao and Joshi 2022b), FedFa (Huang et al. 2022), FedGini (Li et al. 2023), and FedMGDA+ (Hu et al. 2022). Moreover, we consider FedCKA (Son, Kim, and Chung 2022) and FedMDFG (Pan et al. 2023), which are relevant to ours. We directly rewrite the code of baselines based on the authors' open-source code.

Hyper-parameters. We follow the settings of (Wang et al. 2021; Pan et al. 2023) that all clients use Stochastic Gradient Descent (SGD) on local datasets with local epoch $E = 1$. We set the learning rate $\eta \in \{0.01, 0.05, 0.1\}$ decay of 0.999 per round and choose the best performance of each method in comparison. We take the average of results in 5 runs with different random seeds.

Datasets and Models. We evaluate the performance of algorithms on the public datasets Fashion MNIST (FMNIST) (Xiao, Rasul, and Vollgraf 2017) and CIFAR-10/100 (Krizhevsky and Hinton 2009), where the training and testing data have already been split. To simulate heterogeneous clients in FL, we consider three scenarios: (1) Dir(α): We follow (Hsu, Qi, and Brown 2019) to simulate m clients in Dirichlet heterogeneous partition. When $\alpha < 1$, most of the training/testing data of one specific class are probably assigned to a small portion of clients, and the sizes of clients' data are different. (2) Pat-2: We follow (McMahan et al. 2017) to build pathological non-IID data that each client has the data of two classes. (3) Pat-1: It constructs a difficult data-island scenario where each client only has the data of one class. We adopt Multilayer perceptron (MLP) (Popescu et al. 2009) for FMNIST, CNN (Wang et al. 2021) with two convolutional layers for CIFAR-10, and NFResNet-18 (Brock, De, and Smith 2021) for CIFAR-100.

4.1 Performance and Fairness

We first evaluate the average test accuracy and the fairness indicator on FMNIST, CIFAR-10, and CIFAR-100. Table 1 lists the comparison results, revealing that FedLF outperforms the previous FL methods in terms of the mean test accuracy and fairness. The results of FedMDFG verify that mitigating model-level gradient conflicts can boost both the

	FMNIST			CIFAR-10			CIFAR-100		
Algorithm	Dir(0.1)	Pat-1	Pat-2	Dir(0.1)	Pat-1	Pat-2	Dir(0.1)	Pat-1	Pat-2
FedAvg	.861(.116)	.828(.170)	.838(.135)	.690(.214)	.575(.341)	.681(.276)	.343(.201)	.199(.667)	.222(.604)
qFedAvg	.847(.133)	.831(.161)	.813(.118)	.681(.204)	.565(.301)	.661(.267)	.344(.196)	.183(.690)	.238(.529)
FedProx	.825(.121)	.834(.142)	.836(.105)	.544(.242)	.572(.205)	.566(.212)	.197(.291)	.207(.617)	.201(.538)
AFL	.865(.109)	.829(.204)	.854(.137)	.679(.201)	.561(.251)	.685(.202)	.382(.181)	.177(.753)	.261(.509)
Ditto	.820(.129)	.749(.278)	.815(.124)	.598(.216)	.463(.240)	.553(.251)	.301(.241)	.070(1.08)	.114(.784)
FedFV	.850(.132)	.836(.165)	.853(.135)	.682(.208)	.568(.376)	.681(.204)	.339(.198)	.191(.664)	.229(.558)
DRFL	.861(.109)	.855(.136)	.847(.157)	.692(.190)	.578(.307)	.684(.270)	.341(.201)	.193(.644)	.228(.540)
FedFa	.844(.174)	.815(.205)	.836(.116)	.653(.244)	.482(.297)	.695(.232)	.387(.189)	.114(1.09)	.222(.776)
FedGini	.867(.115)	.839(.160)	.837(.134)	.698(.195)	.587(.315)	.672(.246)	.349(.191)	.203(.621)	.198(.666)
FedCKA	.861(.117)	.816(.227)	.840(.129)	.690(.205)	.575(.341)	.674(.211)	.344(.201)	.190(.691)	.222(.575)
FedMGDA+	.809(.161)	.750(.305)	.815(.221)	.531(.264)	.440(.314)	.569(.282)	.173(.358)	.035(1.15)	.080(.839)
FedMDFG	.873(.089)	.863(.101)	.874(.084)	.729(.176)	.744(.142)	.714(.153)	.387(.181)	.278(.485)	.332(.387)
FedLF	**.892(.084)**	**.894(.089)**	**.898(.074)**	**.766(.140)**	**.765(.126)**	**.761(.127)**	**.420(.158)**	**.409(.347)**	**.413(.305)**

Table 1: The average test accuracy of all clients (and the fairness indicator) in Dir(0.1), Pat-1, and Pat-2 on FMNIST, CIFAR-10, and CIFAR-100 with batch size 50 over 3000 communication rounds. 10% of 100 clients are online per round.

	Acc(Fair)	MC	LC_1	LC_2	LC_3	LC_4	LC_5
FedAvg	.440(.282)	2.4	2.7	2.6	2.5	2.3	3.4
qFedAvg	.488(.248)	1.4	2.6	2.7	2.4	1.6	2.9
FedProx	.479(.230)	1.7	2.4	2.2	1.7	1.3	3.1
AFL	.398(.203)	4.7	4.5	4.5	4.4	4.5	4.7
Ditto	.465(.340)	1.5	2.6	2.9	2.6	1.8	3.2
FedFV	.418(.468)	1.9	2.7	2.7	1.9	2.6	4.3
DRFL	.383(.393)	2.9	3.1	3.1	3.0	2.7	3.6
FedFa	.357(.581)	4.3	4.4	4.8	4.7	4.5	4.4
FedGini	.408(.490)	2.2	2.6	2.4	2.3	2.5	3.5
FedCKA	.385(.393)	2.2	2.5	2.5	2.3	3.5	3.5
FedMGDA+	.429(.456)	2.2	2.5	2.4	2.2	2.3	3.4
FedMDFG	.723(.158)	0.0	3.0	2.5	1.7	2.8	3.8
FedLF	**.752(.086)**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**

Table 2: Mean test acc. (and fairness indicator); the average number of online clients whose gradients conflict with the model update direction on the model level (MC), layer 1 (LC_1), ..., layer 5 (LC_5) at each round on CIFAR-10 Pat-1 with batch size 200. 10% of 100 clients are online per round.

model's average performance and fairness. Compared with FedMDFG, FedLF showcases marked improvement, especially in CIFAR-100, Pat-1, where the average test accuracy of FedLF increases by 47.1%. In Appendix B.1, we present the complete experimental results, including the worst 5% and the best 5% of the model test accuracy across clients.

To elucidate the adverse impact of the conflicting direction on performance and fairness, we report the mean test acc., fairness indicator, and the average number of clients whose gradients conflict with the update direction in the model level and at each layer. Table 2 lists the experimental results, verifying that mitigating model-level gradient conflicts can achieve notable progress in the model performance and fairness (see FedMDFG's results). In comparison, the update directions of FedLF don't conflict with clients in the model and layer levels, revealing that mitigating the layer-level gradient conflict can further improve model fairness.

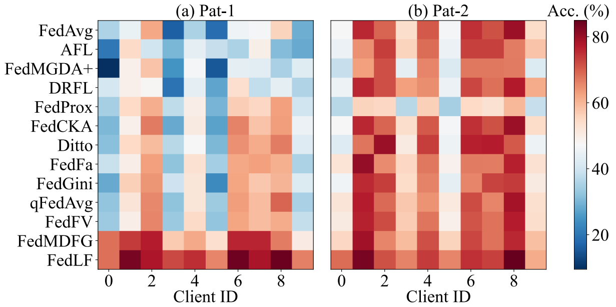
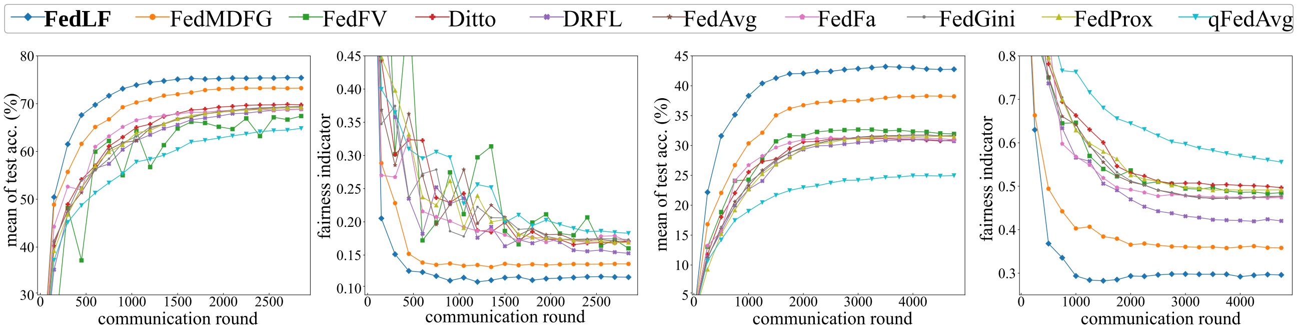


Figure 3: The test accuracy of 10 clients in (a) Pat-1 and (b) Pat-2 on CIFAR-10 with batch size 200 over 3000 communication rounds. 100% of ten clients are online per round.

We further visualize the model's test accuracy of each client in Fig. 3, depicting that some previous methods suffer significant performance reduction on some clients while favoring others. In comparison, FedLF obtains a fairer model with more uniform and higher accuracy across clients.

4.2 Accuracy and Efficiency

We compare the convergent efficiency of algorithms. Fig. 4 partially visualizes the curves for the mean test accuracy and the fairness indicator over communication rounds. All methods are tuned to their best performance, while the methods with poor performance are excluded. Full results are available in Appendix B.1. We unveil that FedLF converges substantially faster, reaches dramatically higher mean test accuracy, and keeps the model fairer compared to previous methods. Furthermore, the fairness indicator of FedLF decreases more stably than most of the previous methods. Besides, some previous methods such as FedFa and FedFV suffer from performance vibration during the training. If we tune a lower learning rate to stabilize them, it will result in a slower convergence and a lower test accuracy (see Appendix B.1).



(a) CIFAR-10 **(b) CIFAR-100**

Figure 4: The mean test accuracy of 100 clients (left) and the fairness indicator (right) in Pat-1 on (a) CIFAR-10 and (b) CIFAR-100 with batch size 200 and $E = 1$. 100% clients are online per round.

	FMNIST	CIFAR-10	CIFAR-100
FedLF	**.896(.052)**	**.766(.106)**	**.418(.321)**
M1	.849(.104)	.679(.201)	.259(.563)
M2	.880(.062)	.736(.117)	.378(.398)
M3	.882(.061)	.738(.115)	.384(.414)
M4	.878(.061)	.741(.136)	.377(.462)
M5	.890(.055)	.757(.123)	.389(.388)
M6	.874(.091)	.672(.186)	.229(.533)
M7	.884(.058)	.735(.132)	.321(.440)
M8	.887(.064)	.750(.126)	.391(.361)

Table 3: The average test accuracy (and the fairness indicator) in the ablation experiments on FMNIST, CIFAR-10, and CIFAR-100 in the partition of Pat-2 with 100 clients. The batch size is 200, and 10% clients are online per round.

4.3 Ablation Experiments

In Table 3, we evaluate several variants of FedLF (M1 to M8) to study the effect of each part.

M1: Replace the layer-wise fair direction d^t with the simple-aggregated direction $\bar{d}{=}-\frac{1}{|S_t|}\sum_{i=1}^{|S_t|} g_i^t$, which easily conflicts with clients' gradients. The results demonstrate that such a direction would make the model much more unfair and have much poorer mean test accuracy.

M2: Combine all layers together to compute a model-level fair direction, i.e., do not mitigate layer-level gradient conflicts. As a result, the average test accuracy and fairness of M2 are inferior to those of FedLF, because it cannot mitigate the layer-level gradient conflicts in FL.

M3: When calculating the update direction, do not separate the gradient by layers, but separate the gradient by parameters into several fragments (if the obtained direction fragment is $\vec{0}$, combine the fragments with its neighbor, just like what we do in combining layers), so that the obtained direction does not conflict with these gradient fragments. The results imply that it is preferable to separate the gradient by layers to calculate a layer-wise fair direction. Because each layer of a model often has its utility (Ma et al. 2022), it's meaningful to calculate a layer-wise fair direction.

M4: Remove the added fair-driven objective in Problem (4). Compared to FedLF, both the mean test accuracy and fairness are much worse in M4, verifying that it's not enough to enhance fairness by mitigating the gradient conflicts since it cannot prevent the improvement bias among clients.

M5: Replace the fair-driven objective $\min_\omega P(\omega) = -cos(\vec{1}, F(\omega))$ of FedLF to the dynamic objective $\min_\omega F(\omega)^T h^t$ utilized in FedMDFG (Pan et al. 2023). Both the average model performance and fairness deteriorated. We discussed its limitation in Section 3.1.

M6: Do not handle the absent client fairness mentioned in Section 3.3. Its results are much worse than FedLF, revealing that when only part of the clients join in FL in each communication round, the model is prone to being unfair since the directions easily conflict with absent clients.

M7: When handling absent client fairness, follow FedMDFG to consider only those who were online at the last communication round when calculating the update direction. M7 outperforms M6, because M6 ignores all absent clients. However, M7 significantly lags behind FedLF, since it ignores those who have not been absent for too long.

M8: Consider the historical gradients of all absent clients when handling the absent client fairness, regardless of how long they have been absent. The results get worse, mainly because it is easily influenced by clients with long absences, where their historical gradients can significantly deviate from what they would be if they were online.

5 Conclusion and Future Work

In this work, we identify three significant challenges that exist in computing a fair direction to drive the FL model fairer: model-level gradient conflicts, improvement bias, and layer-level gradient conflicts. To address these challenges, we propose FedLF, which achieves a layer-wise fair direction. Extensive experiments verify that FedLF outperforms SOTA methods in terms of performance and fairness. A number of interesting topics warrant future exploration, such as making more precise predictions about the gradients of absent clients to better enhance absent client fairness and designing privacy-protection techniques for FL.

Acknowledgments

This work is supported in part by National Natural Science Foundation of China (72331009, 72171206, 62001412), in part by the Shenzhen Institute of Artificial Intelligence and Robotics for Society (AIRS), in part by Shenzhen Key Lab of Crowd Intelligence Empowered Low-Carbon Energy Network (No. ZDSYS20220606100601002), and in part by the Guangdong Provincial Key Laboratory of Future Networks of Intelligence (Grant No. 2022B1212010001).

References

Abay, A.; Zhou, Y.; Baracaldo, N.; Rajamoni, S.; Chuba, E.; and Ludwig, H. 2020. Mitigating bias in federated learning. *arXiv preprint arXiv:2012.02447*.

Bertsekas, D. P. 1999. Nonlinear programming Second Edition. *Journal of the Operational Research Society*, 48(3): 46–46.

Brock, A.; De, S.; and Smith, S. L. 2021. Characterizing signal propagation to close the performance gap in unnormalized ResNets. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Cho, Y. J.; Gupta, S.; Joshi, G.; and Yağan, O. 2020. Bandit-based communication-efficient client selection strategies for federated learning. In *2020 54th Asilomar Conference on Signals, Systems, and Computers*, 1066–1069. IEEE.

Désidéri, J.-A. 2012. Multiple-gradient descent algorithm (MGDA) for multiobjective optimization. *Comptes Rendus Mathematique*, 350(5-6): 313–318.

Fliege, J.; and Svaiter, B. F. 2000. Steepest descent methods for multicriteria optimization. *Mathematical Methods of Operations Research*, 51(3): 479–494.

Gebken, B.; Peitz, S.; and Dellnitz, M. 2019. A descent method for equality and inequality constrained multiobjective optimization problems. In *Numerical and Evolutionary Optimization–NEO 2017*, 29–61. Springer.

Hsu, T.-M. H.; Qi, H.; and Brown, M. 2019. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*.

Hu, Z.; Shaloudegi, K.; Zhang, G.; and Yu, Y. 2022. Federated Learning Meets Multi-Objective Optimization. *IEEE Transactions on Network Science and Engineering*, 9(4): 2039–2051.

Huang, W.; Li, T.; Wang, D.; Du, S.; and Zhang, J. 2020. Fairness and accuracy in federated learning. *arXiv preprint arXiv:2012.10069*.

Huang, W.; Li, T.; Wang, D.; Du, S.; Zhang, J.; and Huang, T. 2022. Fairness and accuracy in horizontal federated learning. *Information Sciences*, 589: 170–185.

Kairouz, P.; McMahan, H. B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A. N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. 2021. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2): 1–210.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. *Handbook of Systemic Autoimmune Diseases*, 1(4).

Lee, S.; Zhang, T.; and Avestimehr, A. S. 2023. Layer-wise adaptive model aggregation for scalable federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 8491–8499.

Li, T.; Hu, S.; Beirami, A.; and Smith, V. 2021. Ditto: Fair and Robust Federated Learning Through Personalization. In *ICML*, 6357–6368.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020a. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2: 429–450.

Li, T.; Sanjabi, M.; Beirami, A.; and Smith, V. 2020b. Fair Resource Allocation in Federated Learning. In *8th International Conference on Learning Representations, ICLR-2020*. OpenReview.net.

Li, X.; Zhao, S.; Chen, C.; and Zheng, Z. 2023. Heterogeneity-aware fair federated learning. *Information Sciences*, 619: 968–986.

Ma, X.; Zhang, J.; Guo, S.; and Xu, W. 2022. Layer-wised Model Aggregation for Personalized Federated Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10092–10101.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, 1273–1282. PMLR.

Mei, Y.; Guo, B.; Xiao, D.; and Wu, W. 2021. FedVF: Personalized Federated Learning Based on Layer-wise Parameter Updates with Variable Frequency. In *2021 IEEE International Performance, Computing, and Communications Conference (IPCCC)*, 1–9. IEEE.

Mohri, M.; Sivek, G.; and Suresh, A. T. 2019. Agnostic federated learning. In *International Conference on Machine Learning*, 4615–4625. PMLR.

Pan, Z.; Wang, S.; Li, C.; Wang, H.; Tang, X.; and Zhao, J. 2023. FedMDFG: Federated Learning with Multi-Gradient Descent and Fair Guidance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 9364–9371.

Popescu, M.-C.; Balas, V. E.; Perescu-Popescu, L.; and Mastorakis, N. 2009. Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems*, 8(7): 579–588.

Salazar, T.; Fernandes, M.; Araujo, H.; and Abreu, P. H. 2022. FAIR-FATE: Fair Federated Learning with Momentum. *arXiv preprint arXiv:2209.13678*.

Shi, Y.; Yu, H.; and Leung, C. 2021. A survey of fairness-aware federated learning. *arXiv preprint arXiv:2111.01872*.

Son, H. M.; Kim, M. H.; and Chung, T.-M. 2022. Comparisons Where It Matters: Using Layer-Wise Regularization to Improve Federated Learning on Heterogeneous Data. *Applied Sciences*, 12(19): 9943.

Wang, Z.; Fan, X.; Qi, J.; Wen, C.; Wang, C.; and Yu, R. 2021. Federated Learning with Fair Averaging. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, 1615–1623. IJCAI Organization.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

Yu, B.; Mao, W.; Lv, Y.; Zhang, C.; and Xie, Y. 2022. A survey on federated learning in data mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 12(1): e1443.

Yu, F.; Wang, D.; Shelhamer, E.; and Darrell, T. 2018. Deep layer aggregation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2403–2412.

Zhao, Z.; and Joshi, G. 2022a. A Dynamic Reweighting Strategy For Fair Federated Learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8772–8776. IEEE.

Zhao, Z.; and Joshi, G. 2022b. A Dynamic Reweighting Strategy For Fair Federated Learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8772–8776. IEEE.

Zhou, Z.; Chu, L.; Liu, C.; Wang, L.; Pei, J.; and Zhang, Y. 2021. Towards fair federated learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 4100–4101.