FedLF: Layer-wise Fair Federated Learning Appendix

Contents

A Theoretical Analysis and Proof

In Section A.1, we first analyze how previous works calculate directions for the model update. We then analyze how the layer-wise fair direction obtained by FedLF can handle the three challenges and drive the model fairer. In Section A.2, we discuss the convergence of FedLF. Finally, in Section A.4, we discuss the expectation of privacy protection in FedLF.

A.1 Analysis of Layer-wise Fair Direction

There are three challenges in computing an update direction that can drive the model fairer: (1) Model-level gradient conflicts; (2) Improvement bias; (3) Layer-level gradient conflicts.

First, if a direction conflicts with clients' gradients in the model level, it cannot be a common descent direction and will lead to a decrease in the model performance on some clients and thus harm fairness.

Besides, only ensuring the direction being common descent cannot prevent the improvement bias, which also harms fairness. This observation is supported by the analysis presented in Section 1 and Section 3.1 of the main paper.

Furthermore, although a common descent direction doesn't conflict with clients' gradients, it cannot prevent from conflicting with clients' gradient fragments in the layer level, which would favor parts of clients and thus decrease the model's capability, such as data feature extraction on some clients. Consequently, it can reduce the fairness of the model. For example, there are two clients whose gradients are $g_1^t = (0.1, 0.1, 0.2, -0.1)^T$, $g_2^t = (-0.1, 0.2, -0.1, 0.4)^T$, respectively. Suppose the model has two layers, and there are two parameters in each layer. Then not only g_1^t and g_2^t are conflicting, but also their gradient fragments at the second layers, i.e., $g_{1,2}^t$ and $g_{2,2}^t$, are conflicting, where $g_{1,2}^t = (0.2, -0.1)^T$ and $g_{2,2}^t = (-0.1, 0.4)^T$ denote the gradient fragment of the second layer of two clients, respectively. If the direction used for the model update is simply obtained by $d^t = -(g_1^t + g_2^t)/2 = (0.0, -0.15, -0.05, -0.15)^T$, then d^t is a common descent direction, but conflicts with $g_{1,2}^t$ and thus will make the new model drift away from client 1 at layer 2.

Remarks. It's worth noting that the statement "gradient conflicts" describes the conflict relation between two clients' gradients, which easily appears in non-IID settings. And the statement "a direction conflicts with some clients' gradients" describes the conflicting relation between an updated direction and the gradients, which can happen when improper methods are employed to calculate the direction.

[Previous tries]. Calculating a direction by simply aggregating the local updates would conflict with clients' gradients in the model level and layer level in the non-IID settings (see $d_{aggregated}^t$ in Fig. 1). Some previous works tried to calculate a direction that does not conflict with clients' gradients. Wang et al. [23] proposed FedFV by using a gradient projection method. But it cannot ensure obtaining a non-gradient-conflicting direction when there are more than two clients in FL; Hu et al. [8] proposed FedMGDA+ by designing a modified Multiple Gradient Descent Algorithm. However, it contains a hyper-parameter that is hard to tune to ensure that the obtained direction is a common descent direction. Here we report a commonly observed counterexample in Fig. 1. Recently, FedMDFG [19] successfully adopted the multiple gradient descent algorithm to FL to compute a common descent direction that does not conflict with each client's gradient in the model level. But it cannot prevent from being conflicting with clients' gradient in the layer level that it may reduce the layer utility on some clients and thus harm fairness.

[Analysis]. Now, we analyze how the proposed FedLF can achieve a fair direction capable of handling the above three challenges to enhance fairness in FL.

To mitigate the model-level gradient conflict, we follow [8] and [19] to consider FL as a multi-objective optimization problem:

$$\min_{\omega} \ (F_1(\omega), F_2(\omega), \cdots, F_m(\omega)). \tag{1}$$

To handling the challenge of the improvement bias, we design a fair-driven objective $\min_\omega P(\omega) = -cos(\vec{1}, F(\omega))$ and add it to Problem (1), so that the formulation for FL is:

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

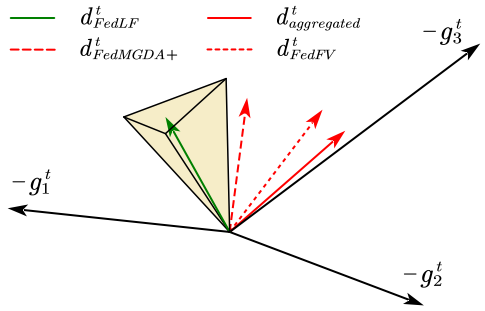


Figure 1: A demo of three clients with three parameters. $g^t_1 = (-1.0, 0.5, 0.0)^T$, $g^t_2 = (0.8, -1.0, 0.0)^T$, $g^t_3 = (1.0, 1.0, -1.0)^T$ denote clients' gradients at round t. d^t_{FedLF}, $d^t_{aggregated}$, $d^t_{FedMGDA+}$, and d^t_{FedFV} are the directions obtained by FedLF, simply aggregation, FedMGDA+, and FedFV, respectively. The area in yellow depicts all possible common descent directions.

where ω denotes the global model parameters, m represents the number of clients. The fair-driven objective can help drive to increase the cosine similarity between the vector $F(\omega)$ and the vector $\vec{1}$, so that it can reduce the improvement bias and enhance the model fairness.

We then propose a layer-wise multiple gradient algorithm that can calculate direction that satisfies $d^t_l \cdot g^t_{i,l} < 0, \forall i \in \{1, \cdots, m\}, \forall l \in L$. Thus, d^t also satisfies $d^t \cdot g^t_i < 0, \forall i \in \{1, \cdots, m\}$, so that d^t can mitigate both the model-level and layer-level gradient conflicts. Moreover, the obtained d^t satisfies $g^t_P \cdot d^t < 0$, where $g^t_P = -\nabla \cos(\vec{1}, F(\omega^t))$, so that it can lead the model to increase the cosine similarity between $F(\omega^t)$ and $\vec{1}$, and thus drive the model fairer. Therefore, the obtained direction is called a layer-wise fair direction.

To achieve the goal, at first, we solve the following quadratic problem to get $\lambda_1, \cdots, \lambda_m$ and μ, and then calculate d^t_l by $d^t_l = -\left(\sum_{i=1}^m \lambda_i g^t_{i,l} + \mu g^t_{P,l}\right)$.

$$\begin{aligned} &\max_{\lambda_i, \mu} -\tfrac{1}{2} \| \textstyle\sum_{i=1}^m \lambda_i g^t_{i,l} + \mu g^t_{P,l} \|^2 \\ &s.t. \ \ \textstyle\sum_{i=1}^m \lambda_i + \mu = 1, \\ &\quad \lambda_i, \mu \geq 0, \forall i = 1, 2, \cdots, m. \end{aligned} \tag{3}$$

Since $\| \sum_{i=1}^m \lambda_i g^t_{i,l} + \mu g^t_{P,l} \|^2$ is positive-semidefinite, we can always get the optimal solution to the above problem. In practice, we utilize cvxopt [1] to obtain a numerical solution by setting an optimality tolerance 1e-6. It's not time-consuming to solve Problem (3) since it's a simple quadratic problem. We report the actual computation time of the server in Section B.3.

Inspired by [5], based on the KKT condition, the above problem is the dual form of the following problem, where α^t_l is a variable in the problem.

$$\begin{aligned} &(d^t_l, \alpha^t_l) = \arg \min_{d^t_l \in \mathbb{R}^{n_l}, \alpha^t_l \in \mathbb{R}} \alpha^t_l + \tfrac{1}{2} \|d^t_l\|^2, \\ &s.t. \quad g^t_{i,l} \cdot d^t_l \leq \alpha^t_l, i = 1, \cdots, m, \\ &\quad\ \ g^t_{P,l} \cdot d^t_l \leq \alpha^t_l. \end{aligned} \tag{4}$$

Inspired by [5], the solution of Problem (4) satisfies:

1. If $\exists \xi \in \mathbb{R}^m, \xi \geq \vec{0}$, such that $\sum_{i=1}^m g^t_{i,l} \xi_i + g^t_{P,l} \xi_{m+1} = \vec{0}$, then $d^t_l = \vec{0}$.
2. Otherwise, d^t_l satisfies the following two inequalities and thus does not conflict with $g^t_{i,l}$.

$$\begin{aligned} &g^t_{i,l} \cdot d^t_l < 0, i = 1, \cdots, m, \\ &g^t_{P,l} \cdot d^t_l < 0. \end{aligned} \tag{5}$$

[Combine Layers]. For some models, some layers may only contain a few parameters, which would make the obtained layer-wise direction be $\vec{0}$. In order to prevent $d^t_l = \vec{0}$ for part of layer l but $d^t \neq \vec{0}$,

we repeatly combine the layer l with its next layer (or the previous layer if l has no next layer) $k \in L$ if $d_l^t = \vec{0}$ and recalculate d_l^t until $d_l^t \neq \vec{0}$ or all layers are combined.

After that, we build d^t by combining all d_l^t. The obtained d^t satisfies:

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

$$\begin{aligned} &g_i^t \cdot d^t < 0, i = 1, \cdots, m, \\ &g_P^t \cdot d^t < 0. \end{aligned} \tag{6}$$

Ultimately, before using the direction d^t to update the model, we scale its length to that of the simple-aggregated direction $\bar{d} = -\frac{1}{m}\sum_{i=1}^m g_i^t$. This step can also be seen in [23, 19], which can prevent the step size from being affected by the norm of d^t, so that in the experiments, we can use the same setting of the learning rate as utilized by the previous algorithms.

It's worth noting that, for a non-Pareto stationary solution ω^t of Problem (1), there always exists a direction d^t, such that $g_i^t \cdot d^t < 0, \forall i \in \{1, \cdots, m\}$ and $g_P^t \cdot d^t < 0$. Meaning that the fair-driven objective doesn't affect the convergence of FL.

Proof:

Denote g_P^t as the gradient of the fair-driven objective, g^t =concat($g_1^t, \cdots, g_m^t$), thus we have:

$$\begin{aligned} g_P^t &= -\nabla cos(\vec{1}, F(\omega^t)) \\ &= \frac{g^t}{\|F(\omega^t)\|^2} \cdot \left(\frac{F(\omega^t)^T \vec{1} F(\omega^t)}{\|\vec{1}\|\|F(\omega^t)\|} - \frac{\vec{1}\|F(\omega^t)\|}{\|\vec{1}\|}\right) \\ &= \frac{g^t}{\|F(\omega^t)\|} \cdot \left(\frac{F(\omega^t)^T \vec{1} F(\omega^t)}{\|\vec{1}\|\|F(\omega^t)\|^2} - \frac{\vec{1}}{\|\vec{1}\|}\right). \end{aligned} \tag{7}$$

Denote $q = \frac{1}{\|F(\omega^t)\|} \cdot \left(\frac{F(\omega^t)^T \vec{1} F(\omega^t)}{\|\vec{1}\|\|F(\omega^t)\|^2} - \frac{\vec{1}}{\|\vec{1}\|}\right)$, then we have $q \perp F(\omega^t)$.

Since ω^t is not Pareto stationary in Problem (1),

so that $F(\omega^t) > \vec{0}$ and thus there is at least one element of q that is not smaller than 0.

Thus, when ω^t is non-Pareto stationary in Problem (1), $\exists\, d^t$ such that $g^t q \cdot d^t < 0$, i.e., $g_P^t \cdot d^t < 0$. Q.E.D.

[Improving Absent Client Fairness]. The model-level and layer-level conflicts also exist in the case of partial client participation or client dropout. Since the server cannot get any information about the absent clients, the obtained direction would conflict with their gradients if they were online.

In FedLF, we take into account those absent clients who were online from $t-\tau$ to $t-1$ communication rounds, and utilize their last historical gradients as the estimation when calculating the layer-wise fair 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. Concretely, denote $H_1, \cdots, H_K$ as the estimation gradients of these absent clients, similar to Problem (3) we calculate the direction fragment d_l^t in this case by: $d_l^t = -(\sum_{i=1}^m \lambda_i g_{i,l}^t + \mu g_{P,l}^t + \sum_{k=1}^K \delta_i H_{k,l})$, where λ, μ, δ are obtained by:

$$\begin{aligned} \max_{\lambda_i, \mu} \; & -\frac{1}{2}\|\sum_{i=1}^m \lambda_i g_{i,l}^t + \mu g_{P,l}^t + \sum_{k=1}^K \delta_i H_{k,l}\|^2 \\ s.t. \; & \sum_{i=1}^m \lambda_i + \mu + \sum_{k=1}^K \delta_i = 1, \\ & \lambda_i, \mu, \delta_k \geq 0, \forall i = 1, 2, \cdots, m, \forall k = 1, 2, \cdots, K. \end{aligned} \tag{8}$$

A.2 Convergence Analysis

We first prove the convergence of FedLF and then analyze the convergence rate. We assume that all clients are online at each communication round.

A.2.1 Convergence of FedLF

In part 1, we prove the convergence of FedLF as follows, which is also proof of Theorem 1 of the main paper. In part 2, and part 3, we discuss the convergence under some special cases.

[Part 1]. Denote x as an accumulation point of the sequence $\{\omega^t\}_t$. Assume that each client i's objective is differentiable and Lipschitz-smooth (L-smooth) [3] with the Lipschitz constant L_i. Set $\mathbb{L}$ as a set of all Lipschitz constants. Since d^t satisfies $g_i^t \cdot d^t < 0, \forall i \in \{1, \cdots, m\}$, by L-smooth and the descent lemma [3] we have

$$\begin{aligned} F_i(\omega^t) - F_i(\omega^t + \eta^t d^t) &\geq -\eta^t g_i^t \cdot d^t - \frac{1}{2}(\eta^t)^2 L_i \|d^t\|^2, \\ &= \eta^t(|g_i^t \cdot d^t| - \frac{1}{2}\eta^t L_i \|d^t\|^2), \end{aligned} \tag{9}$$

The right-hand side yields

$$|g_i^t \cdot d^t| - \frac{1}{2}\eta^t L_i \|d^t\|^2 \geq \frac{1}{2}\epsilon |g_i^t \cdot d^t|, \ \forall i = 1, \cdots, m. \tag{10}$$

where $\epsilon \in (0, 1]$ is a fixed scalar.

Given ω^t and the common descent direction d^t, the cost difference of $F_i(\omega^t) - F_i(\omega^t + \eta^t d^t)$ is majorized by

$$\eta g_i^t \cdot d^t + \frac{1}{2}\eta L_i \|d^t\|^2. \tag{11}$$

By the minimum of 11 over η, we can obtain the upper bound of the step size that can reduce the local objective F_i:

$$2 \cdot \frac{|g_i^t \cdot d^t|}{L_i \|d^t\|^2}, \tag{12}$$

Thus, the upper bound of the step size that can reduce all local objectives is

$$2 \cdot \min_{L_i \in \mathbb{L}} \frac{|g_i^t \cdot d^t|}{L_i \|d^t\|^2}. \tag{13}$$

On the other hand, η^t is bounded by $\eta^t > 0$.

Using the step size boundary and the inequality (9), we can get the following bound on the cost improvement obtained at round t:

$$F(\omega^t) - F(\omega^t + \eta^t d^t) \geq -\frac{1}{2}\epsilon^2 g_i^t \cdot d^t \geq \vec{0}. \tag{14}$$

Since $\forall i, F_i(\omega^t)$ is decreasing and bounded by 0, we have

$$\lim_{t \to \infty} ||F(\omega^t) - F(\omega^{t+1})|| = \vec{0}. \tag{15}$$

Hence, by (14), we obtain

$$\lim_{l \to \infty} (g^t)^T \cdot d^t = \vec{0}, \tag{16}$$

which implies it converges to a Pareto stationary point.

[Part 2]. Assume that the learning rate η is set too high that is bigger than the upper bound (13), but satisfies:

$$\frac{1}{m}\sum\nolimits_{i=1}^{m} F_i(\omega^t + \eta^t d^t) < \frac{1}{m}\sum\nolimits_{i=1}^{m} F_i(\omega^t). \tag{17}$$

Since d^t is the common descent direction, $\frac{1}{m}\sum_{i=1}^{m} F_i(\omega^t)$ is decreasing and bounded by 0, according to Monotone Convergence Theorem [21], FedLF can converge to a local optimum of the following Federated Learning problem

$$\min_{\omega} \frac{1}{m}\sum\nolimits_{i=1}^{m} L_i(\omega). \tag{18}$$

[Part 3]. In the case of partial client participation and client dropout, not all clients are online at each communication round in FL. In Section 3.3 of the main paper, we introduce an approach to handle the issue of absent clients. Assuming that the obtained direction d^t is a common descent direction for those absent clients, then the model can still converge. The proof is the same to the above two parts.

A.2.2 Convergence Rate

The convergence of FedLF requires $\|d^t\| \leq \epsilon$, where ϵ is the tolerance. In part 1, we will prove that FedLF converges sublinearly. We further prove that if the local objective $L_i(\omega)$ is strongly convex, FedLF converges linearly (see part 2).

[Part 1]. Assume that the client's objective F_i is differential and L-smooth with the Lipschitz constant L_i. We rewrite Problem (2) to the following dynamic single objective problem, which is the dynamic weight-sum of the local objectives with the weights vector μ:

$$\min_{\omega^t} \mathcal{F}(\omega^t) = \sum_{i=1}^{m} \mu_i F_i(\omega^t), \tag{19}$$

where $\mu_i = \lambda_i + q_i\lambda_{m+1}$, $q = \frac{1}{\|F(\omega^t)\|} \cdot (\frac{F(\omega^t)^T \vec{1} F(\omega^t)}{\|\vec{1}\|\|F(\omega^t)\|^2} - \frac{\vec{1}}{\|\vec{1}\|})$, and λ is the solution of Problem (3).

By the definition of $\mathcal{F}(\omega^t)$ we have $\nabla\mathcal{F}(\omega^t) = -d^t$, and $\mathcal{F}$ is L-smooth.

Define $L_{\mathcal{F}} := \max_{\zeta} \sum_{i=1}^{m} \zeta_i L_i$, where $\zeta \geq \vec{0}$ and $\zeta_1 + \cdots + \zeta_m = 1$, we have

$$\begin{aligned}
\mathcal{F}(\omega^{t+1}) &\leq \mathcal{F}(\omega^t) + \nabla\mathcal{F}(\omega^t)(\omega^{t+1} - \omega^t) + \frac{L_{\mathcal{F}}}{2}\|\omega^{t+1} - \omega^t\|^2 \\
&= \mathcal{F}(\omega^t) - \eta^t\|d^t\|^2 + \frac{L_{\mathcal{F}}(\eta^t)^2}{2}\|d^t\|^2 \\
&= \mathcal{F}(\omega^t) - [\eta^t - \frac{L_{\mathcal{F}}(\eta^t)^2}{2}]\|d^t\|^2.
\end{aligned} \tag{20}$$

Consider a fixed step size $\eta^t = \frac{1}{L_{\mathcal{F}}}$ that can make $\mathcal{F}(\omega^t) - \mathcal{F}(\omega^{t+1}) \geq 0$ hold. Thus, we have:

$$\mathcal{F}(\omega^{t+1}) \leq \mathcal{F}(\omega^t) - \frac{1}{2L_{\mathcal{F}}}\|d^t\|^2. \tag{21}$$

Take t from 0 to T, and take the summation of them, we have

$$\mathcal{F}(\omega^T) \leq \mathcal{F}(\omega^0) - \frac{1}{2L_{\mathcal{F}}}\sum_{t=0}^{T-1}\|d^t\|^2. \tag{22}$$

Denote ω^* as the accumulation point, then we can get

$$\frac{1}{2L_{\mathcal{F}}}\sum_{t=0}^{T-1}\|d^t\|^2 \leq \mathcal{F}(\omega^0) - \mathcal{F}(\omega^T) \leq \mathcal{F}(\omega^0) - \mathcal{F}(\omega^*). \tag{23}$$

Further,

$$\min_t \|d^t\|^2 \leq \frac{1}{T}\sum_{t=0}^{T-1}\|d^t\|^2 \leq \frac{2L_{\mathcal{F}}[\mathcal{F}(\omega^0) - \mathcal{F}(\omega^*)]}{T}. \tag{24}$$

Thus,

$$\mathbb{E}(\min_t \|d^t\| \leq \epsilon) = O(\frac{1}{T}), \tag{25}$$

meaning that it needs at least $T = O(\frac{1}{\epsilon})$ rounds to converge, which indicates the sublinear convergence rate of FedLF.

[Part 2]. Assume that the local objective $F_i(\omega)$ are strongly convex with parameter $u > 0$. Denote ω^* is a Pareto stationary solution. According to the strongly convex property, we have

$$F(\omega^*) - F(\omega^t) \geq \nabla F(\omega^t)^T(\omega^* - \omega^t) + \frac{u}{2}\|\omega^* - \omega^t\|^2. \tag{26}$$

Hence,

$$\nabla F(\omega^t)^T(\omega^* - \omega^t) \leq F(\omega^*) - F(\omega^t) - \frac{u}{2}\|\omega^* - \omega^t\|^2. \tag{27}$$

Denote λ as the optimal solution of Problem (3), which satisfies $\lambda_i \geq 0$ and $\sum_i \lambda_i = 1$. Since $\nabla F(\omega^t)^T(\omega^* - \omega^t) < \vec{0}$, we can get

$$\lambda^T \nabla F(\omega^t)^T(\omega^* - \omega^t) \leq F(\omega^*) - F(\omega^t) - \frac{u}{2}\|\omega^* - \omega^t\|^2. \tag{28}$$

Since $d^t = -\nabla F(\omega^t)\lambda$, we have

$$(\omega^t - \omega^*) \cdot d^t \leq F(\omega^*) - F(\omega^t) - \frac{u}{2}\|\omega^* - \omega^t\|^2. \tag{29}$$

Besides, since $F_i(\omega)$ is Lipschitz-smooth, by (14), we have

$$\begin{aligned} F(\omega^t) - F(\omega^{t+1}) &\geq -\frac{1}{2}\epsilon^2 \nabla F(\omega^t)^T d^t \\ &\vdots \\ F(\omega^{(\cdot)}) - F(\omega^*) &\geq -\frac{1}{2}\epsilon^2 \nabla F(\omega^{(\cdot)})^T d^t. \end{aligned} \tag{30}$$

Take the summation of the left-hand side, and remain only one item of the right-hand side, we have

$$F(\omega^t) - F(\omega^*) \geq \sum_i -\frac{1}{2}\epsilon^2 \nabla F(\omega^i)^T d^t \geq -\frac{1}{2}\epsilon^2 \nabla F(\omega^t)^T d^t. \tag{31}$$

Take $\eta^t \leq \epsilon^2$, we obtain

$$\begin{aligned} F(\omega^t) - F(\omega^*) &\geq -\frac{\eta^t}{2}\nabla F(\omega^t)^T d^t \\ F(\omega^*) - F(\omega^t) &\leq -\frac{\eta^t}{2}\nabla F(\omega^t)^T(-d^t) \\ F(\omega^*) - F(\omega^t) &\leq -\frac{\eta^t}{2}\|d^t\|^2. \end{aligned} \tag{32}$$

Hence, by (29) and (32), we obtain

$$(\omega^t - \omega^*) \cdot d^t \leq -\frac{\eta^t}{2}\|d^t\|^2 - \frac{u}{2}\|\omega^* - \omega^t\|^2. \tag{33}$$

It yields that

$$\begin{aligned} \|\omega^{t+1} - \omega^*\|^2 &= \|\omega^t + \eta^t d^t - \omega^*\|^2 \\ &= \|\omega^t - \omega^*\|^2 + 2\eta^t(\omega^t - \omega^*)^T d^t + (\eta^t)^2\|d^t\|^2 \\ &\leq \|\omega^t - \omega^*\|^2 - u\eta^t\|\omega^t - \omega^*\|^2 - (\eta^t)^2\|d^t\|^2 + (\eta^t)^2\|d^t\|^2 \\ &= (1 - u\eta^t)\|\omega^t - \omega^*\|^2. \end{aligned} \tag{34}$$

Since $(1 - u\eta^t) \in (0, 1)$, we complete the proof that FedLF converges linearly when the local objectives are strongly convex.

In conclusion, FedLF converges sublinearly, and if the local objectives are strongly convex, it converges linearly.

A.3 Related Work and Difference

In this section, we clarify the difference among our proposed FedLF and some previous FL algorithms.

For handling the challenge of model-level gradient conflicts, there are related works FedFV [23], FedMGDA+ [8], and FedMDFG [19].

[FedFV]. FedFV calculates the direction for the FL model update by repeatedly projecting the local gradient g_i^t to each of other clients' local gradient g_j^t, $j \neq i$ when there exists model-level gradient conflict between g_i^t and g_j^t, i.e., $g_i^t \cdot g_j^t < 0$. However, [19] have shown that this approach cannot guarantee that the obtained direction has no conflict with the local gradient g_i^t when there are more than two clients in FL. We also give a counterexample in Fig. 1.

[FedMGDA+]. FedMGDA+ designed a modified multiple gradient descent algorithm to solve Problem (1) to compute a direction for the FL model update. But it contains a hyperparameter that it's hard to tune to ensure the obtained direction is a common descent direction [19]. We also give a counterexample in Fig. 1.

[FedMDFG]. FedMDFG successfully applied the multiple gradient descent algorithm in FL, which can ensure to compute a common descent direction for the model update. First, they designed a multi-objective optimization with a dynamic objective $\min_\omega F(\omega)^T h^t$:

$$\min_{\omega} \ (F_1(\omega), F_2(\omega), \cdots, F_m(\omega), F(\omega)^T h^t), \tag{35}$$

and then apply the multiple gradient descent algorithm (MGDA) to solve it to obtain a common descent direction. h^t is a dynamic vector that is an opposite normalized vector of the projection of $\vec{1}$ on the normal plane of $F(\omega^t)$, which change FL to a dynamic multi-objective optimization problem. This objective can make the local objective vector closer to $\vec{1}$ generally, but when the local objectives are quite close, this objective will make the local objective vector far away from $\vec{1}$ and thus make the model more unfair. Therefore, [19] add a step size line search approach to search for a smaller step size (learning rate) to ease this negative impact, but this would bring extra communication cost (see Table 9). Besides, they design a hyperparameter to control whether to use the dynamic objective, i.e., when the angle between the local objective vector and $\vec{1}$ is smaller than a threshold θ, the added dynamic objective is not activated. But it brings difficulties to the convergence guarantee, while they only provided the convergence proof without considering this hyperparameter.

Differently, we design a more effective objective $\min_\omega P(\omega) = -cos(\vec{1}, F(\omega))$ and formulate Problem (2) for FL. This fair-driven objective directly aims to drive the local objective vector to be closer to $\vec{1}$, which doesn't have the above issue of making the model more unfair and doesn't change the FL formulation to a dynamic optimization problem, and we provide convergence proof and calculate the convergence rate.

As for the update direction, the direction obtained by FedMDFG can just guarantee to have no conflicts with clients' gradients in the model level, while our proposed FedLF can ensure the direction doesn't conflict with clients' gradients in both the model level and the layer level, which can further protect the layer utility on clients and enhance fairness.

As for handling the absent clients, FedMDFG takes into account those absent clients who were online at the last communication round, which would ignore those who were online for more than one round but not too long. Differently, we just ignore those who were absent for too long based on the expectation of the arrival time, so that we can take more absent clients into consideration and would not be affected by those who were absent for too long.

A.4 Privacy Discussion about FedLF.

We start the discussion of privacy with the calculation of clients' local gradients. In the experiment of the main paper, we follow [23, 19] to use Stochastic Gradient Descent (SGD) on clients' local dataset with local epoch $E = 1$, so that the client i's gradient information g_i^t is the same as its local update divided by the learning rate, i.e., $g_i^t = (\omega^t - \omega_i^{t+1})/\eta^t$, where η^t is the learning rate, ω^t is the current global model, and ω_i^{t+1} is the model obtained by the local training in client i. Therefore, at each communication round t, each client i upload its local training result ω_i^{t+1} to the server, then the server can compute g_i^t by $g_i^t = (\omega^t - \omega_i^{t+1})/\eta^t$.

[Remarks.] This practical approach can also be seen in many other well-known gradient-based FL algorithms such as qFFL [15], FedFV [23], FedMGDA+ [8], and FedMDFG [19]. It's worth noting

that, when the local epoch E is larger than 1, the value of $(\omega^t - \omega_i^{t+1})/\eta^t$ is just an estimation of the local gradient and would affect the performance. We do the additional experiments in "Exp.4" in Section B.2.1 to test the performance of algorithms under the setting of $E = 5$.

Gradient privacy protection is a distinct research direction in FL. According to the privacy analysis of [10], in practice, **it's privacy-safe enough** to directly upload the plaintext of g_i^t or the local training result ω_i^{t+1} to the server **when using a batch size larger than 32**.

Moreover, there are many approaches such as Homomorphic Encryption [2] that can be adopted in FedLF to improve privacy protection, where **clients no longer need to upload the plaintext of g_i^t or ω_i^{t+1} to the server**, and the value $\sum_{i=1}^m \lambda_i g_{i,l}^t + \mu g_{P,l}^t$, which is used for calculating the layer-wise fair direction can be obtained without knowing any plaintext of g_i^t or ω_i^{t+1}. Besides, the value $\sum_{i=1}^m \lambda_i g_{i,l}^t + \mu g_{P,l}^t$ itself is privacy-safe because it doesn't leak any gradient of clients.

Hence, in the future, we will work on designing privacy-protection techniques to enhance privacy preservation.

B Complete and Extra Experimental Results

Datasets and Models. In experiments, we utilize Multilayer perceptron (MLP) [20] on Fashion MNIST (FMNIST [24]), where there are three layers and each layer has 200 neurons. For CIFAR-10, we follow [17, 23] to adopt CNN [12] with two convolutional layers and three fully-connected layers. Both the two convolutional layers have 64 channels, respectively. The fully connected layers have 384, 192, and 10 neurons, respectively. For CIFAR-100 [11], we adopt NFResNet-18 [4], which is an advanced ResNet for distributed training tasks. In the supplementary experiments (Appendix Section B.3), we also utilize Compact Convolutional Transformer (CCT [6]) for CIFAR-100 to test the efficiency of the algorithms under a large network.

Data Partition. We consider three kinds of data partitions for simulating heterogeneous clients.

- Dir(0.1) [7]: Most of the training/test data of one specific class are probably assigned to a small portion of clients. In other words, each client may have the data in all classes, but most of these data belong to one class. Besides, the data amounts of clients are different.
- Dir(1.0): A close-to-IID case, where each client has all the classes of the dataset and the data amounts of all clients are almost the same.
- Pat-2 [17, 23]: Randomly assign the data from two classes to each client, such that each client only has data in two classes, and two clients may have data in the same class. The data amounts of clients are the same.
- Pat-1: A difficult data-island scenario where each client only has the data in one class. For example, for CIFAR-10, there are 10 classes at all. If we allocate the data to 100 clients, then the data in one class are separated into 10 clients randomly. The data amounts of clients are the same.

Baselines. We list the baselines we used in experiments.

- FedAvg [17]: A classical FL algorithm.
- qFedAvg [15]: A method that utilizes a resource allocation strategy to enhance fairness.
- FedProx [14]: Using a proximal term to limit local updates.
- AFL [18]: A method that tries to prevent the model from overfitting certain clients at the expense of others.
- Ditto [13]: Enhancing fairness by personalization.
- FedFV [23]: Mitigating gradient conflicts by projecting local gradients.
- DRFL [25]: A dynamic reweighting strategy to enhance fairness.
- FedFa [9]: Using momentum and reweighting approaches to enhance fairness in FL.
- FedGini [16]: Introducing a penalty term to enhance fairness.
- FedCKA [22]: A FL method by using layer-wise aggregation.
- FedMGDA+ [8]: Using a modified multiple gradient descent approach to enhance fairness.
- FedMDFG [19]: Using the multiple gradient descent algorithm on a multi-objective optimization for FL with a designed dynamic objective.

Implementation Details. We adopt the commonly employed hyper-parameters of the previous algorithms based on their papers. We follow the setting of [23] to use Stochastic Gradient Descent (SGD) on clients' local dataset with local epoch $E = 1$, and set the learning rate $\eta \in \{0.01, 0.05, 0.1\}$ decayed 0.999 per round and choose the best performance of each method in comparison. In "Exp.4" of Section B.2.1, we also do the test under a larger epoch $E = 5$. The batch size is set to $B \in \{50, 200\}$. All experiments are implemented on a server with Intel(R) Xeon(R) Silver 4216 CPU and NVidia(R) 3090 GPU.

B.1 Full Experimental Results

The full experimental results of 'Table 1' of the main paper are presented here. Since there are too many records, we separate the results into Table 1, Table 2, and Table 3. It can be seen that FedLF

outperforms previous algorithms in terms of average accuracy and fairness. Some previous methods have very poor 5% worst performance, especially on CIFAR-100, where some of them even have 0% test accuracy for the worst 5% clients. In comparison, FedLF does well in protecting the model's worst performance on clients.

Table 1: The average, the fairness indicator, and worst 5% and the best 5% of the test accuracy of all clients in Dir(0.1), Pat-1, and Pat-2 on FMNIST with batch size 50 over 3000 communication rounds. 10% of 100 clients are online per round. The results are averaged over 5 runs with different random seeds.

	FMNIST Dir(0.1)				FMNIST Pat-1				FMNIST Pat-2			
	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%
FedAvg	.861±.011	.116±.019	.601±.058	.999±.001	.828±.019	.170±.057	.401±.144	.988±.006	.838±.065	.135±.074	.325±.135	.999±.003
qFedAvg	.847±.007	.133±.012	.631±.045	1.00±.000	.831±.021	.161±.059	.424±.134	.983±.006	.813±.018	.118±.031	.508±.072	.993±.006
FedProx	.825±.003	.121±.010	.587±.042	.98±.005	.834±.002	.142±.011	.561±.044	.973±.003	.836±.002	.105±.016	.545±.048	.976±.004
AFL	.865±.001	.109±.005	.643±.000	1.00±.000	.829±.001	.204±.006	.351±.001	.984±.001	.854±.002	.137±.005	.533±.002	.990±.001
Ditto	.820±.001	.129±.005	.546±.020	.975±.002	.749±.017	.278±.054	.157±.124	.956±.004	.815±.005	.124±.030	.487±.078	.974±.003
FedFV	.850±.009	.132±.020	.649±.070	1.00±.000	.836±.031	.165±.069	.416±.165	.993±.006	.853±.026	.135±.035	.510±.091	.996±.006
DRFL	.861±.014	.109±.022	.619±.073	.999±.001	.855±.022	.136±.057	.457±.144	.983±.005	.847±.048	.157±.048	.533±.092	.995±.005
FedFa	.844±.021	.174±.030	.526±.089	.999±.001	.815±.046	.205±.100	.198±.186	.996±.003	.836±.055	.116±.066	.338±.122	.998±.003
FedGini	.867±.009	.115±.020	.597±.069	1.00±.000	.839±.024	.160±.063	.429±.147	.990±.004	.837±.025	.134±.042	.535±.084	.997±.006
FedCKA	.861±.011	.117±.019	.601±.059	.999±.002	.816±.021	.227±.055	.277±.132	.984±.008	.840±.065	.129±.096	.326±.136	.999±.003
FedMGDA+	.809±.001	.161±.005	.472±.021	.975±.002	.750±.017	.305±.031	.042±.059	.979±.002	.815±.010	.221±.022	.316±.048	1.00±.000
FedMDFG	.873±.001	.089±.003	.690±.012	1.00±.000	.863±.005	.101±.011	.628±.049	.986±.001	.874±.007	.084±.017	.681±.049	.983±.004
FedLF	**.892±.001**	**.084±.002**	**.719±.013**	**1.00±.000**	**.894±.001**	**.089±.002**	**.717±.008**	**.996±.001**	**.898±.001**	**.074±.002**	**.731±.009**	**1.00±.000**

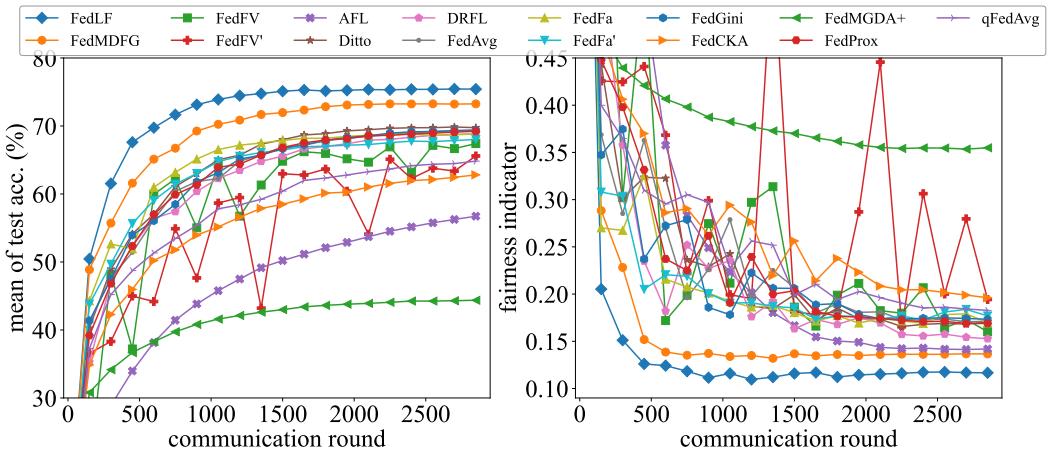
Table 2: The average, the fairness indicator, and worst 5% and the best 5% of the test accuracy of all clients in Dir(0.1), Pat-1, and Pat-2 on CIFAR-10 with batch size 50 over 3000 communication rounds. 10% of 100 clients are online per round. The results are averaged over 5 runs with different random seeds.

	CIFAR-10 Dir(0.1)				CIFAR-10 Pat-1				CIFAR-10 Pat-2			
	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%
FedAvg	.690±.014	.214±.026	.346±.065	.952±.021	.575±.022	.341±.047	.221±.069	.887±.03	.681±.047	.276±.107	.250±.130	.905±.040
qFedAvg	.681±.005	.204±.012	.381±.033	.982±.012	.565±.015	.301±.037	.232±.052	.859±.033	.661±.011	.267±.045	.357±.074	.876±.027
FedProx	.544±.003	.242±.011	.195±.034	.877±.004	.572±.001	.205±.004	.355±.010	.770±.009	.566±.001	.212±.002	.308±.008	.770±.003
AFL	.679±.001	.201±.006	.363±.001	.954±.006	.561±.002	.251±.001	.283±.003	.806±.001	.685±.001	.202±.002	.477±.002	.895±.002
Ditto	.598±.003	.216±.012	.319±.018	.890±.036	.463±.004	.240±.028	.192±.036	.689±.024	.553±.002	.251±.012	.321±.013	.788±.017
FedFV	.682±.006	.208±.010	.342±.024	.980±.015	.568±.030	.376±.067	.162±.089	.886±.029	.681±.010	.204±.032	.388±.054	.893±.016
DRFL	.692±.017	.190±.028	.361±.066	.970±.018	.578±.025	.307±.060	.206±.079	.870±.034	.684±.029	.270±.066	.356±.110	.897±.030
FedFa	.653±.014	.244±.029	.232±.070	.960±.039	.482±.005	.297±.025	.000±.000	**1.00±.000**	.695±.011	.232±.044	.419±.087	**.926±.017**
FedGini	.698±.006	.195±.013	.361±.037	.986±.014	.587±.020	.315±.050	.225±.066	.886±.032	.672±.012	.246±.036	.403±.067	.907±.024
FedCKA	.690±.014	.205±.026	.346±.065	.952±.021	.575±.022	.341±.047	.221±.069	.887±.030	.674±.047	.211±.011	.276±.140	.900±.044
FedMGDA+	.531±.001	.264±.005	.194±.025	.924±.009	.440±.001	.314±.009	.163±.006	.710±.014	.569±.002	.282±.015	.282±.009	.823±.005
FedMDFG	.729±.002	.176±.002	.412±.013	.987±.007	.744±.002	.142±.002	.544±.021	.904±.007	.714±.001	.153±.013	.535±.024	.859±.013
FedLF	**.766±.001**	**.140±.001**	**.482±.001**	**.994±.004**	**.765±.001**	**.126±.001**	**.538±.005**	.908±.001	**.761±.001**	**.127±.002**	**.558±.005**	.920±.004

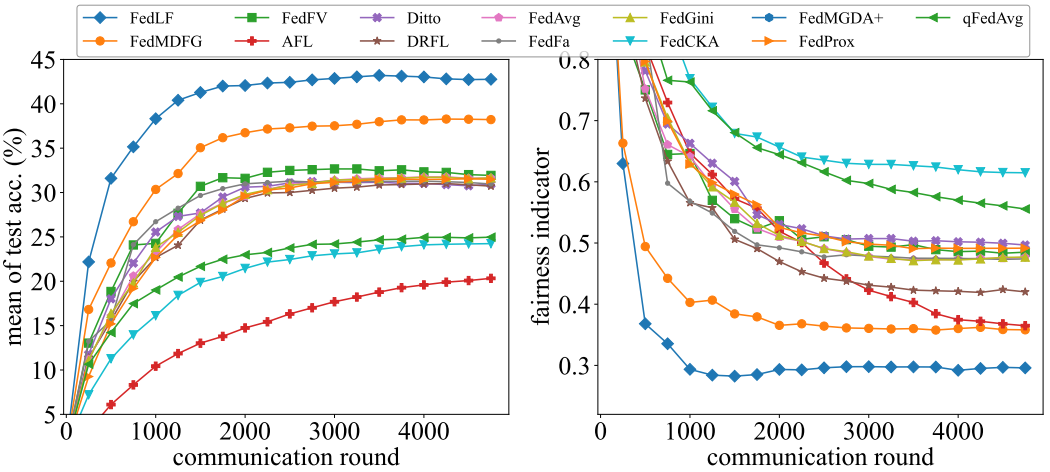
Table 3: The average, the fairness indicator, and worst 5% and the best 5% of the test accuracy of all clients in Dir(0.1), Pat-1, and Pat-2 on CIFAR-100 with batch size 50 over 3000 communication rounds. 10% of 100 clients are online per round. The results are averaged over 5 runs with different random seeds.

	CIFAR-100 Dir(0.1)				CIFAR-100 Pat-1				CIFAR-100 Pat-2			
	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%
FedAvg	.343±.002	.201±.005	.210±.005	.487±.006	.199±.005	.667±.019	.012±.005	.596±.035	.222±.014	.604±.063	.040±.016	.573±.050
qFedAvg	.344±.002	.196±.006	.208±.005	.477±.008	.183±.007	.690±.020	.005±.003	.582±.038	.238±.004	.529±.027	.045±.010	.544±.032
FedProx	.197±.002	.291±.006	.096±.004	.327±.005	.207±.002	.617±.010	.014±.005	.567±.012	.201±.003	.538±.013	.026±.005	.472±.017
AFL	.382±.001	.181±.003	.236±.000	.531±.001	.177±.002	.753±.012	.000±.000	.602±.000	.261±.002	.509±.002	.069±.001	.635±.000
Ditto	.301±.001	.241±.003	.150±.003	.455±.004	.070±.001	1.08±.012	.000±.000	.406±.020	.114±.002	.784±.024	.001±.001	.477±.024
FedFV	.339±.002	.198±.005	.218±.003	.494±.006	.191±.007	.664±.019	.010±.005	.599±.035	.229±.010	.558±.050	.035±.014	.569±.043
DRFL	.341±.002	.201±.005	.209±.007	.481±.008	.193±.007	.644±.029	.018±.008	.544±.025	.228±.010	.540±.040	.045±.013	.517±.037
FedFa	.387±.004	.189±.012	.235±.010	.523±.009	.114±.023	1.00±.071	.000±.000	**.739±.036**	.222±.024	.776±.064	.000±.000	**.733±.044**
FedGini	.349±.002	.191±.005	.204±.007	.493±.008	.203±.006	.621±.023	.006±.007	.578±.039	.198±.006	.666±.023	.006±.007	.578±.039
FedCKA	.344±.002	.201±.005	.210±.005	.487±.006	.190±.005	.691±.019	.012±.005	.596±.035	.222±.014	.575±.063	.040±.016	.573±.050
FedMGDA+	.173±.001	.358±.003	.063±.001	.319±.002	.035±.001	1.15±.012	.000±.000	.314±.008	.080±.001	.839±.012	.000±.000	.314±.008
FedMDFG	.387±.003	.181±.012	.220±.006	.535±.006	.278±.002	.485±.011	.058±.008	.620±.013	.332±.002	.387±.007	.131±.009	.674±.007
FedLF	**.420±.002**	**.158±.004**	**.290±.007**	**.563±.007**	**.409±.006**	**.347±.010**	**.146±.001**	.689±.001	**.413±.006**	**.305±.010**	**.166±.001**	.632±.001

Fig. 2 illustrates the full results of the experiment in Section 4.2 of the main paper. In Fig. 2. (a), we further set a lower learning rate for FedFa and FedFV and denote them as FedFa' and FedFV', respectively. The results depict that FedFa' converges more slowly but more stable than FedFa. But for FedFV', it converges more slowly and remains unstable, since FedFV cannot guarantee that the obtained direction will not substantially conflict with clients' gradients. The results verify that our proposed FedLF outperforms SOTA algorithms in terms of mean test accuracy and fairness.



(a) CIFAR-10



(b) CIFAR-100

Figure 2: 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. 100% clients are online per round.

B.2 Additional Experiments

In this section, we do many additional experiments and discussion about the algorithms.

B.2.1 Additional Experiments on Performance and Fairness

(Exp.1) Conflict Test. In Table 2 of the main paper, we listed the conflicting status of the update directions on CIFAR-10 Pat-1, verifying that using directions conflicting with clients' gradient in the model level and layer level will reduce the model's average performance and fairness. Here, we show the conflicting status in the case of CIFAR-10 Dir(1.0) as a comparison of the previous results. Table 4 reveals that, since CIFAR-10 Dir(1.0) is a close-to-IID case, it is easier for a simple aggregated direction to have no conflicts with clients' gradients in the model level and the layer level, thus most of the previous algorithms can achieve a good average test acc and fairness. But in practice, the assumption of IID cannot be satisfied, and the model-level and layer-level gradient conflicts will diminish the model's average performance and fairness. Compared with previous methods, there is no model-level or layer-level conflict between clients' gradient and the directions obtained by FedLF, allowing for a fairer model with better performance.

(Exp.2) Effect of the partition rate. We list the performance and fairness under different partition rates of clients on CIFAR-100, Pat-1. Most of the previous methods perform poorly when there

Table 4: 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 and Dir(1.0) with batch size 200. 10% of 100 clients are online per round.

	CIFAR-10 Pat-1							CIFAR-10 Dir(1.0)						
	Acc(Fair)	MC	LC_1	LC_2	LC_3	LC_4	LC_5	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	.732(.050)	0	0.3	0	0	0	0.5
qFedAvg	.488(.248)	1.4	2.6	2.7	2.4	1.6	2.9	.723(.042)	0	0	0	0	0	0
FedProx	.479(.230)	1.7	2.4	2.2	1.7	1.3	3.1	.720(.055)	0.2	1.1	0.6	0.2	0.1	0.8
AFL	.398(.203)	4.7	4.5	4.5	4.4	4.5	4.7	.730(.044)	0	0.5	0.1	0	0	0.6
Ditto	.465(.340)	1.5	2.6	2.9	2.6	1.8	3.2	.725(.052)	0	1.1	0.8	0	0	0.3
FedFV	.418(.468)	1.9	2.7	2.7	1.9	2.6	4.3	.729(.046)	0	0.2	0	0	0	0.3
DRFL	.383(.393)	2.9	3.1	3.1	3	2.7	3.6	.722(.044)	0.3	1.2	1.5	0.1	0	0.1
FedFa	.357(.581)	4.3	4.4	4.8	4.7	4.5	4.4	.721(.068)	1	2.4	2.6	0.1	0	1.9
FedGini	.408(.490)	2.2	2.6	2.4	2.3	2.5	3.5	.726(.046)	0	0.3	0	0	0	0.5
FedCKA	.385(.393)	2.2	2.5	2.5	2.3	3.5	3.5	.721(.046)	0	0	0	0	0	0
FedMGDA+	.429(.456)	2.2	2.5	2.4	2.2	2.3	3.4	.723(.044)	0	0.2	0	0	0	0
FedMDFG	.723(.158)	0	3	2.5	1.7	2.8	3.8	.731(.035)	0	0	0	0	0	0
FedLF	**.752(.086)**	**0**	**0**	**0**	**0**	**0**	**0**	**.732(.035)**	**0**	**0**	**0**	**0**	**0**	**0**

Table 5: The mean test accuracy (and the fairness indicator) of 100 clients with 20%, 50%, and 100% online in Pat-1 on CIFAR-100 with batch size 200.

Online rate	20%	50%	100%
FedAvg	.197(.769)	.294(.528)	.316(.477)
qFedAvg	.226(.651)	.232(.630)	.250(.556)
FedProx	.185(.774)	.292(.562)	.315(.492)
AFL	.184(.684)	.192(.644)	.203(.364)
Ditto	.145(.810)	.257(.607)	.309(.497)
FedFV	.280(.577)	.307(.517)	.319(.485)
DRFL	.266(.562)	.292(.482)	.307(.420)
FedFa	.183(.887)	.320(.415)	.310(.474)
FedGini	.262(.583)	.296(.507)	.317(.477)
FedCKA	.204(.648)	.227(.523)	.242(.615)
FedMGDA+	.034(1.00)	.042(.979)	.048(.922)
FedMDFG	.336(.413)	.347(.404)	.382(.358)
FedLF	**.406(.347)**	**.411(.334)**	**.428(.296)**

are only 20% of clients online per round. With the help of improving the absent client fairness, the performances of FedLF are more stable under different participation rates of clients.

(Exp.3) Performance in weak non-IID settings. We further compare the performance and fairness in Dir(1.0) with 100 clients, which is a close-to-IID case where each client has almost all the classes of the dataset, and the data amounts of all clients are almost the same. Table 6 list the compared results, depicting that most of the previous FL methods perform well and are comparatively fair. This is because the data distributions of clients in this scenario are similar to IID, and thus there are almost no gradient conflicts. Therefore, it's easy for a simple aggregated direction obtained by previous FL methods to be a common descent direction that would not lead to performance reduction across clients. But since there is a challenge of improvement bias, FedLF still outperforms the previous methods in terms of fairness and average performance because of using a fair-driven objective.

(Exp.4) Performance under a larger local epoch. We further evaluate the test accuracy and fairness of algorithms under a larger local epoch $E = 5$. Table 7 and Table 8 list the results on FMNIST and CIFAR-10, respectively. It can be seen that the performance and fairness of most of the algorithms are deteriorating. This is because a larger local epoch can easily enlarge the gradient conflicts among clients, and for the gradient-based FL algorithms, such as AFL, FedFV, FedMGDA+, FedMDFG, and FedLF, the gradient uploaded by each client is no longer the actual gradient, but rather an estimation of the local gradient, which would affect the performance. However, in terms of average test accuracy and fairness, FedLF still outperforms the previous algorithms.

Table 6: The average, the fairness indicator, and worst 5% and the best 5% of the test accuracy of all clients in Dir(1.0) on FMNIST, CIFAR-10, and CIFAR-100 with batch size 50 over 3000 communication rounds. 10% of 100 clients are online per round. The results are averaged over 5 runs with different random seeds.

	FMNIST Dir(1.0)				CIFAR-10 Dir(1.0)				CIFAR-100 Dir(1.0)			
	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%
FedAvg	.895±.002	.046±.003	.799±.014	.968±.004	.737±.001	.077±.001	.610±.004	.837±.002	.374±.001	.134±.001	.279±.003	.483±.003
qFedAvg	.894±.002	.045±.003	**.805±.010**	.964±.004	.732±.001	.079±.001	.610±.005	.846±.003	.391±.001	.132±.003	.283±.005	.508±.006
FedProx	.840±.002	.064±.003	.712±.010	.932±.003	.561±.001	.100±.001	.462±.004	.687±.004	.186±.001	.199±.005	.113±.005	.257±.004
AFL	.892±.001	.045±.001	.804±.001	.965±.001	.730±.001	.076±.001	.623±.001	**.854±.001**	.415±.001	.132±.001	.304±.001	.533±.001
Ditto	.862±.001	.057±.001	.748±.003	.942±.004	.694±.001	.080±.002	.587±.006	.803±.003	.366±.001	.138±.003	.270±.003	.471±.003
FedFV	.897±.002	.046±.003	.804±.013	.970±.003	.737±.001	.078±.001	.620±.002	.854±.003	.372±.001	.160±.001	.251±.002	.500±.002
DRFL	.894±.003	.048±.004	.793±.020	.967±.004	.737±.001	.075±.002	.623±.005	.852±.004	.370±.001	.141±.001	.271±.002	.479±.002
FedFa	.895±.002	.049±.002	.790±.009	**.972±.002**	.735±.001	.072±.002	.612±.005	.830±.004	.402±.001	.126±.001	.307±.001	.500±.002
FedGini	.896±.002	.046±.004	.799±.016	.970±.004	.731±.001	.079±.001	.616±.006	.849±.002	.378±.001	.141±.002	.273±.002	.484±.002
FedCKA	.895±.002	.046±.003	.799±.014	.968±.004	.737±.001	.077±.001	.610±.004	.837±.002	.374±.001	.134±.001	.279±.003	.483±.003
FedMGDA+	.887±.001	.048±.001	.788±.003	.960±.003	.688±.001	.076±.001	.591±.003	.801±.003	.261±.001	.172±.002	.167±.002	.360±.002
FedMDFG	.897±.002	.045±.002	.802±.005	.963±.004	.742±.001	.068±.001	.633±.004	.837±.003	.420±.001	.121±.003	.314±.004	.516±.004
FedLF	**.897±.001**	**.043±.001**	.802±.005	.965±.003	**.749±.001**	**.066±.001**	**.638±.002**	.840±.002	**.435±.001**	**.112±.003**	**.338±.004**	**.536±.005**

Table 7: The average, the fairness indicator, and worst 5% and the best 5% of the test accuracy of all clients in Dir(0.1), Pat-1, and Pat-2 on FMNIST with batch size 50 over 3000 communication rounds. 10% of 100 clients are online per round. The local epoch $E = 5$. The results are averaged over 5 runs with different random seeds.

	FMNIST Dir(0.1)				FMNIST Pat-1				FMNIST Pat-2			
	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%
FedAvg	.854±.011	.133±.023	.581±.071	.999±.002	.806±.032	.234±.077	.379±.168	.994±.004	.810±.072	.199±.098	.468±.166	1.00±.000
qFedAvg	.867±.006	.109±.013	.657±.054	1.00±.000	.798±.027	.232±.072	.365±.153	.984±.005	.835±.024	.149±.040	.534±.086	.995±.006
FedProx	.826±.004	.131±.012	.567±.048	.982±.004	.836±.002	.139±.011	.561±.044	.974±.002	.839±.002	.120±.010	.570±.029	.975±.004
AFL	.876±.001	.100±.001	.696±.001	1.00±.000	.828±.001	.211±.002	.336±.003	.986±.001	.856±.001	.133±.000	.529±.000	.998±.000
Ditto	.856±.003	.114±.006	.636±.030	1.00±.000	.751±.015	.290±.051	.221±.103	.958±.006	.813±.005	.159±.022	.484±.059	.982±.006
FedFV	.863±.008	.122±.019	.608±.053	1.00±.000	.807±.034	.233±.080	.380±.181	.996±.003	.849±.027	.149±.060	.542±.134	.998±.004
DRFL	.856±.010	.124±.018	.596±.054	.999±.001	.789±.058	.251±.108	.360±.187	.988±.008	.837±.027	.134±.036	.579±.071	.993±.006
FedFa	.803±.042	.203±.066	.416±.137	.997±.003	.100±.000	1.25±.000	.000±.000	**1.00±.000**	.676±.048	.363±.080	.185±.098	.994±.007
FedGini	.863±.008	.123±.018	.605±.053	1.00±.000	.806±.032	.234±.078	.379±.171	.993±.004	.842±.029	.149±.049	.551±.102	.997±.006
FedCKA	.854±.011	.133±.023	.581±.071	.999±.002	.806±.032	.234±.077	.379±.168	.994±.004	.810±.072	.199±.098	.468±.166	1.00±.000
FedMGDA+	.860±.002	.117±.004	.638±.019	1.00±.000	.725±.008	.359±.029	.083±.096	.993±.001	.804±.012	.232±.026	.365±.055	1.00±.000
FedMDFG	.871±.002	.091±.004	.682±.009	1.00±.000	.870±.011	.124±.009	.610±.035	.980±.001	.879±.005	.087±.011	.684±.031	.993±.002
FedLF	**.890±.001**	**.080±.002**	**.715±.009**	**1.00±.000**	**.890±.001**	**.091±.002**	**.714±.006**	.995±.001	**.895±.001**	**.078±.002**	**.725±.007**	**1.00±.000**

Table 8: The average, the fairness indicator, and worst 5% and the best 5% of the test accuracy of all clients in Dir(0.1), Pat-1, and Pat-2 on CIFAR-10 with batch size 50 over 3000 communication rounds. 10% of 100 clients are online per round. The local epoch $E = 5$. The results are averaged over 5 runs with different random seeds.

	CIFAR-10 Dir(0.1)				CIFAR-10 Pat-1				CIFAR-10 Pat-2			
	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%	Acc.	fair	worst 5%	best 5%
FedAvg	.620±.020	.259±.039	.336±.059	.897±.020	.307±.046	.770±.122	.005±.015	.881±.081	.608±.072	.321±.126	.248±.138	.904±.037
qFedAvg	.636±.011	.242±.016	.348±.037	.918±.020	.271±.037	.797±.124	.001±.003	.836±.101	.610±.020	.263±.060	.300±.084	.884±.038
FedProx	.548±.002	.276±.002	.264±.004	.825±.006	.573±.001	.211±.003	.360±.010	.781±.007	.557±.002	.226±.004	.295±.006	.779±.005
AFL	.657±.001	.200±.001	.417±.001	.888±.001	.396±.001	.456±.001	.077±.002	.699±.002	.652±.001	.226±.001	.331±.002	.861±.004
Ditto	.545±.004	.259±.011	.309±.016	.836±.017	.271±.017	.672±.097	.005±.012	.699±.091	.536±.009	.306±.041	.218±.066	.805±.025
FedFV	.646±.011	.239±.025	.353±.049	.917±.022	.334±.048	.731±.116	.003±.007	.873±.058	.629±.025	.261±.066	.310±.109	.884±.031
DRFL	.627±.020	.252±.036	.334±.061	.909±.021	.294±.051	.794±.128	.002±.007	.917±.078	.617±.034	.274±.068	.282±.106	.885±.038
FedFa	.712±.014	.203±.029	.438±.059	.944±.010	.100±.000	1.25±.000	.000±.000	**1.00±.000**	.705±.010	.199±.047	.408±.094	.904±.015
FedGini	.644±.013	.234±.023	.361±.042	.908±.018	.306±.045	.772±.117	.004±.012	.885±.080	.621±.027	.273±.073	.304±.107	.899±.035
FedCKA	.620±.020	.259±.039	.336±.059	.897±.020	.307±.046	.770±.122	.005±.015	.881±.081	.608±.072	.321±.126	.248±.138	.904±.037
FedMGDA+	.583±.003	.265±.008	.185±.032	.907±.014	.439±.002	.320±.017	.157±.009	.691±.030	.313±.005	.294±.020	.294±.011	.841±.009
FedMDFG	.735±.003	.167±.002	.423±.009	.990±.004	.732±.004	.155±.006	.504±.021	.890±.011	.720±.002	.149±.009	.530±.018	.870±.009
FedLF	**.755±.001**	**.146±.001**	**.458±.003**	**.996±.003**	**.748±.001**	**.145±.001**	**.511±.008**	.924±.002	**.750±.001**	**.135±.001**	**.537±.008**	**.948±.003**

B.2.2 Additional Experiments on Accuracy and Efficiency

As a comparison to Fig. 2, we test the convergence of algorithms in the case of Pat-1 on CIFAR-10 with 100 clients and batch size 50. The results can be seen in Fig. 3, verifying that our proposed FedLF still performs more stably and converges faster than previous methods when using a smaller batch size.

Besides, we evaluate the convergence of algorithms on CIFAR-10 in the case of Pat-1 with only 10 clients, which is a challenging data-island scenario than the case of Fig. 2, where each client has only one data class. Fig. 4 depicts the results, showing that FedLF can still achieve results with much better performance.

As a comparison to Fig. 2(b), where we use NFResNet-18 on CIFAR-100, we now utilize a much larger model, Compact Convolutional Transformer (CCT [6]), to test the convergent efficiency and accuracy of algorithms. The results are depicted in Fig. 5. It can be observed that most of the

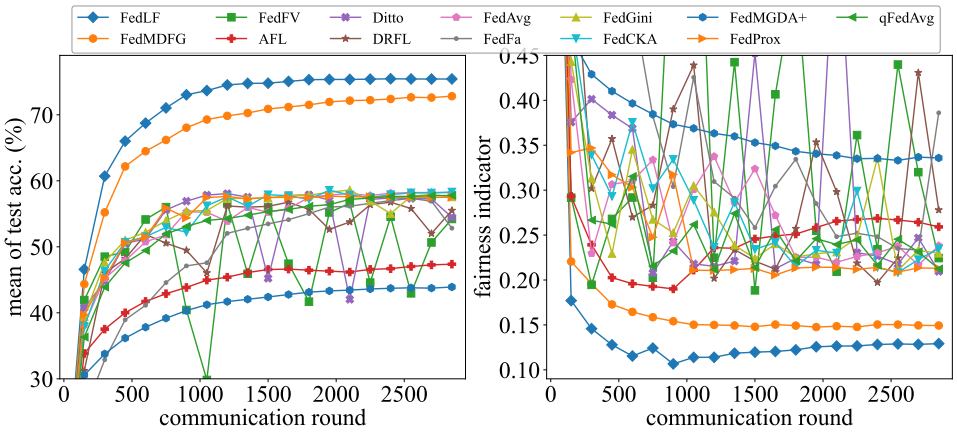


Figure 3: The mean test accuracy of 100 clients (left) and the fairness indicator (right) in Pat-1 on CIFAR-10 with batch size 50. 100% clients are online per round.

algorithms perform better compared to the case of using NFResNet-18 on CIFAR-100 (Fig. 2.(b)). And FedLF still performs significantly better in terms of average test accuracy and fairness.

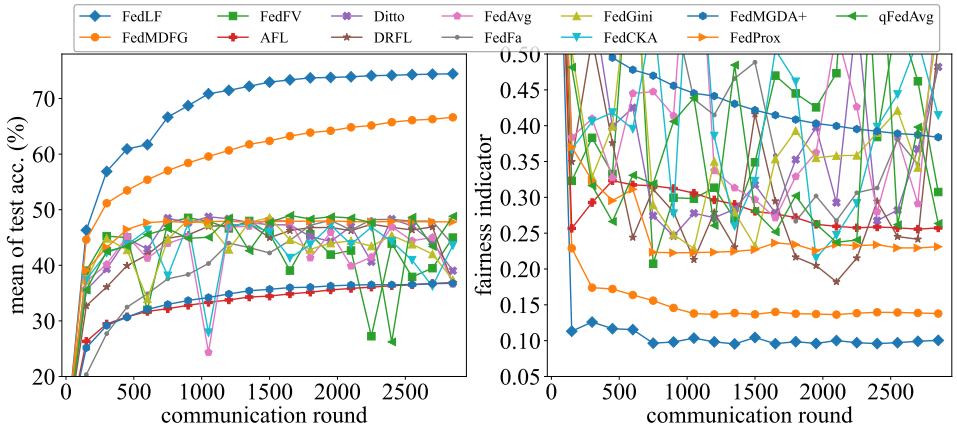


Figure 4: The mean test accuracy of 10 clients (left) and the fairness indicator (right) in Pat-1 on CIFAR-10 with batch size 200. 100% clients are online per round.

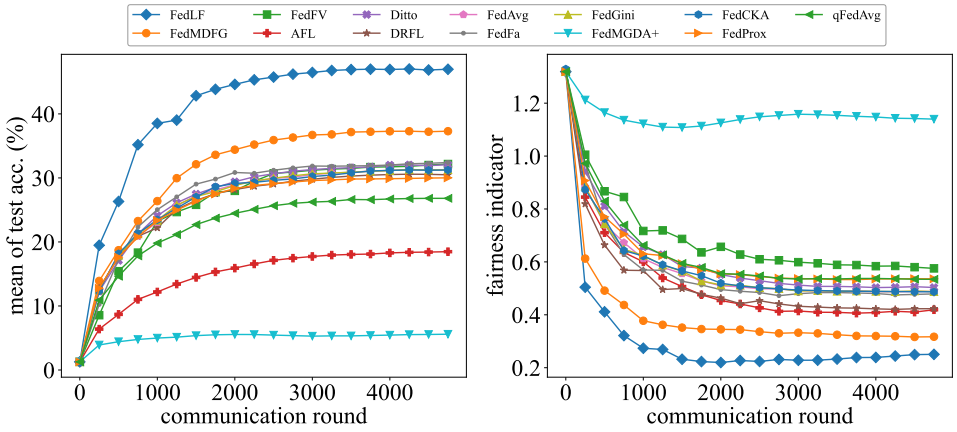


Figure 5: The mean test accuracy of 100 clients (left) and the fairness indicator (right) of a large model CCT in Pat-1 on CIFAR-100 with batch size 200. 100% clients are online per round.

Table 9: The computation time (s) of clients (and the server) on FMNIST and CIFAR-10 over 3000 rounds with 100 clients.

	FedAvg	AFL	Ditto	FedFV	FedCKA	FedMGDA+	FedMDFG	FedLF
FMNIST - 10% Online	1131(19)	1050(37)	1161(17)	1080(380)	1580(19)	1049(35)	1488(32)	1076(91)
CIFAR-10 - 10% Online	2485(40)	2468(76)	2867(42)	2457(997)	3158(41)	2448(111)	3326(132)	2461(153)
CIFAR-10 - 100% Online	24135(121)	24192(142)	26019(127)	24806(87105)	30370(124)	24373(808)	41345(136)	24401(207)

B.3 Runtime

In Table 9, we report the actual computation time of clients (and the server) on FMNIST and CIFAR-10 over 3000 rounds with 100 clients. It can be seen that it's not time-consuming to compute the layer-wise fair direction in FedLF, where the computation time of the server is much lower than that of FedFV. Because FedFV needs to repeatedly project each client's gradient to all the other clients' gradients and thus takes much computational time, especially when there are more clients online per round. For FedCKA, since it performs the layer-wise calculation on each client, its computation time of clients is much larger. For FedMDFG, since it requires a step size line search approach, which requires waiting for clients to do the model inference, so that its communication time is much larger.

C Limitation Analysis and Future Work

(1) When improving the absent client fairness (mentioned in Section 3.3 of the main paper), we follow [23] to estimate the gradients of the absent clients according to their gradients when they were online before, which is not so accurate.

It's worth noting that, even though some estimated gradients may deviate from what they would be if they were online, the direction obtained by the proposed FedLF is still a common descent direction to those online clients and thus would not cause the performance reduction on them. So it offers a promising way to enhance fairness when there are absent clients.

So in the future, we will work on making more precise predictions about the gradients of absent clients to better enhance fairness.

(2) Like many previous FL algorithms such as FedMGDA+ [8], q-FFL [15], FedFV [23], FedMDFG [19], etc., this paper focuses on improving fairness in FL, so that we haven't applied advanced privacy-preserving techniques to FedLF. More privacy discussion can be seen in Section A.4, and we will work on designing privacy techniques for FedLF in the future.

D Codes

The source codes are available at `https://github.com/Anonymous-Fed/Anonymous_FL`.

References

[1] Martin Andersen, Joachim Dahl, and Lieven Vandenberghe. Cvxopt: Convex optimization. *Astrophysics Source Code Library*, pages ascl–2008, 2020.

[2] Yoshinori Aono, Takuya Hayashi, Lihua Wang, Shiho Moriai, et al. Privacy-preserving deep learning via additively homomorphic encryption. *IEEE Transactions on Information Forensics and Security*, 13(5):1333–1345, 2017.

[3] Dimitri P Bertsekas. Nonlinear programming second edition. *Journal of the Operational Research Society*, 48(3):46–46, 1999.

[4] Andrew Brock, Soham De, and Samuel L. Smith. 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, 2021.

[5] Jörg Fliege and Benar Fux Svaiter. Steepest descent methods for multicriteria optimization. *Mathematical methods of operations research*, 51(3):479–494, 2000.

[6] Ali Hassani, Steven Walton, Nikhil Shah, Abulikemu Abuduweili, Jiachen Li, and Humphrey Shi. Escaping the big data paradigm with compact transformers. *arXiv preprint arXiv:2104.05704*, 2021.

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

[8] Zeou Hu, Kiarash Shaloudegi, Guojun Zhang, and Yaoliang Yu. Federated learning meets multi-objective optimization. *IEEE Transactions on Network Science and Engineering*, 9(4):2039–2051, 2022.

[9] Wei Huang, Tianrui Li, Dexian Wang, Shengdong Du, Junbo Zhang, and Tianqiang Huang. Fairness and accuracy in horizontal federated learning. *Information Sciences*, 589:170–185, 2022.

[10] Yangsibo Huang, Samyak Gupta, Zhao Song, Kai Li, and Sanjeev Arora. Evaluating gradient inversion attacks and defenses in federated learning. *Advances in Neural Information Processing Systems*, 34:7232–7241, 2021.

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

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

[13] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *ICML*, pages 6357–6368, 2021.

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

[15] Tian Li, Maziar Sanjabi, Ahmad Beirami, and Virginia Smith. Fair resource allocation in federated learning. In *8th International Conference on Learning Representations, ICLR-2020*. OpenReview.net, 2020.

[16] Xiaoli Li, Siran Zhao, Chuan Chen, and Zibin Zheng. Heterogeneity-aware fair federated learning. *Information Sciences*, 619:968–986, 2023.

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

[18] Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *International Conference on Machine Learning*, pages 4615–4625. PMLR, 2019.

[19] Zibin Pan, Shuyi Wang, Chi Li, Haijin Wang, Xiaoying Tang, and Junhua Zhao. Fedmdfg: Federated learning with multi-gradient descent and fair guidance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 9364–9371, 2023.

[20] Marius-Constantin Popescu, Valentina E Balas, Liliana Perescu-Popescu, and Nikos Mastorakis. Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems*, 8(7):579–588, 2009.

[21] Walter Rudin et al. *Principles of mathematical analysis*, volume 3. McGraw-hill New York, 1976.

[22] Ha Min Son, Moon Hyun Kim, and Tai-Myoung Chung. Comparisons where it matters: Using layer-wise regularization to improve federated learning on heterogeneous data. *Applied Sciences*, 12(19):9943, 2022.

[23] Zheng Wang, Xiaoliang Fan, Jianzhong Qi, Chenglu Wen, Cheng Wang, and Rongshan Yu. Federated learning with fair averaging. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 1615–1623. IJCAI Organization, 2021.

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

[25] Zhiyuan Zhao and Gauri Joshi. A dynamic reweighting strategy for fair federated learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8772–8776. IEEE, 2022.