Proportional Fairness in Federated Learning

Anonymous authors
Paper under double-blind review

Abstract

With the increasingly broad deployment of federated learning (FL) systems in the real world, it is critical but challenging to ensure fairness in FL, i.e. reasonably satisfactory performances for each of the numerous diverse clients. In this work, we introduce and study a new fairness notion in FL, called *proportional fairness* (PF), which is based on the relative change of each client's performance. From its connection with the bargaining games, we propose *PropFair*, a novel and easy-to-implement algorithm for finding proportionally fair solutions in FL, and study its convergence properties. Through extensive experiments on vision and language datasets, we demonstrate that PropFair can approximately find PF solutions, and it achieves a good balance between the average performances of all clients and of the worst 10% clients.

1 Introduction

Federated learning (FL, McMahan et al. 2017) has attracted an intensive amount of attention in recent years, due to its great potential in real world applications such as IoT devices (Imteaj et al., 2021), healthcare (Xu et al., 2021) and finance (Long et al., 2020). In FL, different clients collaboratively learn a global model that presumably benefits all, without sharing the local data.

However, clients differ. Due to the heterogeneity of client objectives and resources, the benefit each client receives may vary. How can we make sure each client is treated *fairly* in FL?

To answer this question, we first need to define what we mean by fairness. Similar to fairness in other fields (Jain et al., 1984; Sen, 1986; Rawls, 1999; Barocas et al., 2017), in FL, there is no unified definition of fairness. In social choice theory, two of the most popular definitions are *utilitarianism* and *egalitarianism*. The goal of utilitarian fairness is to maximize the utility of the total society; while egalitarian fairness requires the worst-off people to receive enough benefits. Coincidentally, they correspond to two of the fair FL algorithms: Federated Averaging (FedAvg, McMahan et al. 2017) and Agnostic Federated Learning (AFL, Mohri et al. 2019). In FedAvg (AFL), we minimize the averaged (worst-case) loss function, respectively. Utilitarian and egalitarian might be in conflict with each other: one could improve the worst-case clients, but better-off clients would be degraded to a large extent.

To achieve some balance between utilitarian and egalitarian fairness, other notions of fairness have been studied. Inspired by *α-fairness* from telecommunication (Mo & Walrand, 2000), Li et al. (2020c) proposed q-Fair Federated Learning (q-FFL). By replacing the client weights with the softmax function of the client losses, Li et al. (2021a) proposed Tilted Empirical Risk Minimization (TERM). However, it remains vague what type of balance these algorithms are trying to yield.

In this work, we bring another fairness notion into the zoo of fair FL, called *proportional fairness* (PF, Kelly 1997). It also balances between utilitarian and egalitarian fairness, but is more intuitive. As a illustrative example, suppose we only have two clients and if we can improve the performance of one client *relatively* by 2% while decreasing another one by 1%, then the solution is more proportionally fair. In practice, this view of relative change is quotidian. In stock market, people care more about how much they gain/lose compared to the cost; in telecommunication, people worry about the data transmission speed compared to the bandwidth. In a word, PF studies the *relative change* of each client, rather than the *absolute change*.

Under convexity, PF is equivalent to the *Nash bargaining solution* (NBS, Nash 1950), a well-known concept from cooperative game theory. Based on the notion of PF and its related NBS, we propose a new FL algorithm called *PropFair*. Our contributions are the following:

- With the *utility* perspective and Nash bargaining solutions, we propose a surrogate loss for achieving proportionally fair FL. This provides new insights to fair FL and is distinct from existing literature which uses the *loss* perspective for fairness (see Section 2).
- *Theoretical guarantee:* we prove the convergence of PropFair to a stationary point of our objective, under mild assumptions. This proof can generalize to any other FL algorithm in the unified framework we propose.
- *Empirical viability:* we test our algorithm on several popular vision and language datasets, and modern neural architectures. Our results show that PropFair not only approximately obtains proportionally fair FL solutions, but also attains more favorable balance between the averaged and worst-case performances.
- Compared to previous works (Mohri et al., 2019; Li et al., 2020c; 2021b), we provide a comprehensive benchmark for popular fair FL algorithms with systematic hyperparameter tuning. This could facilitate future fairness research in FL.

2 A Unified Framework of Fair FL Algorithms

In order to properly locate proportional fairness in the fairness literature, we first review existing fairness definitions that have been applied to FL. Although it is possible to adapt common approaches of fair ML (Appendix E.3.3) into FL, by e.g., treating each sensitive attribute as a client, we note that the adaptation may not always be straightforward due to the unique challenge of privacy in FL. Such adaptation can be interesting future work and we do not consider it here.

Suppose we have n clients, and a model parameterized by $\boldsymbol{\theta}$. Because of data heterogeneity, for each client i the data distribution $\mathcal{D}_i$ is different. The corresponding loss function becomes:

$$f_i(\boldsymbol{\theta}) := \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i}[\ell(\boldsymbol{\theta},(\mathbf{x},y))], \tag{2.1}$$

where ℓ is the prediction loss (such as cross entropy) of model $\boldsymbol{\theta}$ for each sample. The goal of FL is essentially to learn a model $\boldsymbol{\theta}$ that every element in the vector $\boldsymbol{f} = (f_1,\ldots,f_n)$ is small, a.k.a. multi-objective optimization (MOO, Jahn et al. 2009). Hu et al. (2022) took this approach and used Multiple Gradient Descent Algorithm (MGDA) to find Pareto stationary points.

Another popular approach to for MOO is through scalarization of $\boldsymbol{f}$ (Chapter 5, Jahn et al., 2009), by changing the vector optimization to some scalar optimization: $\min_{\boldsymbol{\theta}}(\Phi\circ\boldsymbol{f})(\boldsymbol{\theta})$ with $\Phi:\mathbb{R}^n\to\mathbb{R}$. In this work, we mainly focus on Φ being a (additively) separable function:

$$\min_{\boldsymbol{\theta}}\sum_i p_i(\varphi\circ f_i)(\boldsymbol{\theta}), \quad \varphi:\mathbb{R}\to\mathbb{R}. \tag{2.2}$$

The linear weights p_i's are usually pre-defined and satisfy $p_i\geq 0, \sum_i p_i = 1$. In FL, a usual choice of p_i is $p_i = n_i/N$ with n_i the number of samples for client i and N the total number of samples. Here φ is a monotonically increasing function, since if any f_i increases, the total loss should also increase. In the following subsections, we show that different choices of scalar function φ lead to different fair FL algorithms with their respective fairness principles.

2.1 Utilitarianism

The simplest choice of φ would be the identity function, $\varphi(f_i) = f_i$:

$$\min_{\boldsymbol{\theta}} F(\boldsymbol{\theta}) := \sum_i p_i f_i(\boldsymbol{\theta}), \text{ with } p_i\geq 0 \text{ pre-defined, and } \sum_i p_i = 1. \tag{2.3}$$

This corresponds to the first FL algorithm, Federated Averaging (FedAvg, McMahan et al. 2017). Combined with eq. 2.1, the objective eq. 2.3 is equivalent to centralized training with all the client samples in one place.

From a fairness perspective, eq. 2.3 can be called *utilitarianism*, which can be traced back to at least Bentham (1780). From a utilitarian perspective, a solution $\boldsymbol{\theta}$ is fair if it maximizes an average of the client utilities. (Here we treat client i's utility u_i as $-f_i$. In general, $u_i \in \mathbb{R}$ is some value client i wishes to maximize.)

2.2 Egalitarianism (Maximin Criterion)

In constract to FedAvg, Agnostic Federated Learning (AFL, Mohri et al. 2019) does not assume a pre-defined weight for each client, but aims to minimize the worst convex combination:

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{p}} \sum_i p_i f_i(\boldsymbol{\theta}), \text{ with } \mathbf{1}^\top \boldsymbol{p} = 1, \text{ and } \boldsymbol{p} \geq \mathbf{0}. \tag{2.4}$$

Note that $\boldsymbol{p} \in \mathbb{R}^n$ is a vector on the probability simplex. An equivalent formulation is:

$$\min_{\boldsymbol{\theta}} \max_i f_i(\boldsymbol{\theta}). \tag{2.5}$$

In other words, we minimize the worst-case client loss. In social choice, this corresponds to the egalitarian rule (or more specifically, the maximin criterion, see Rawls 1974). In MOO, this corresponds to $\Phi(\boldsymbol{f}) = \max_i f_i$ (above equation 2.2). There is one important caveat of AFL worth mentioning: the generalization. In practice, each client loss f_i is in fact the expected loss on the empirical distribution $\widehat{\mathcal{D}}_i$, i.e.,

$$\widehat{f}_i(\boldsymbol{\theta}) = \mathbb{E}_{(\mathbf{x},y)\sim\widehat{\mathcal{D}}_i}[\ell(\boldsymbol{\theta}, (\mathbf{x}, y))]. \tag{2.6}$$

In FL, some clients may have few samples and the empirical estimate $\widehat{f}_i$ may not faithfully reflect the underlying distribution. If such a client happens to be the worst-case client, then AFL would suffer from defective generalization. We provide a concrete example in Appendix B, and this phenomenon has also been observed in our experimental section.

2.3 α-Fairness

Last but not least, we may slightly modify the function φ in FedAvg to be $\varphi(f_i) = f_i^{q+1}/(q+1)$:

$$\min_{\boldsymbol{\theta}} \frac{1}{q+1} \sum_i p_i f_i^{q+1}(\boldsymbol{\theta}), \text{ with } p_i \geq 0 \text{ pre-defined, and } \sum_i p_i = 1. \tag{2.7}$$

This is called q-Fair Federated Learning (q-FFL, Li et al. 2020c), and $q \geq 0$ is required. If $q = 0$, then we retrieve FedAvg; if $q \to \infty$, then the client who has the largest loss f_i will be emphasized more, which corresponds to AFL. In general, q-FFL interpolates between the two. From a fairness perspective, q-FFL can relate to α-fairness (Mo & Walrand, 2000), a popular concept from the field of communication. Suppose each client has utility $u_i \in \mathbb{R}$ and $\mathbf{u} = (u_1, \ldots, u_n) \in \mathcal{U} \subseteq \mathbb{R}^n$, with $\mathcal{U}$ the feasible set of client utilities, then α-fairness associates with the following problem:

$$\max_{\mathbf{u}\in\mathcal{U}} \sum_i p_i \phi_\alpha(u_i), \text{ with pre-defined } p_i \geq 0 \text{ and } \phi_\alpha(u_i) = \begin{cases} \log u_i & \text{if } \alpha = 1, \\ u_i^{1-\alpha}/(1-\alpha) & \text{if } \alpha > 0 \text{ and } \alpha \neq 1. \end{cases} \tag{2.8}$$

q-FFL modifies the α-fairness with two changes: (1) take $\alpha = -q$, and allow $\alpha \leq 0$; (2) replace u_i with the loss f_i. Therefore, q-FFL is an analogy of α-fairness. However, the objective eq. 2.7 misses the important case with $\alpha = 1$, also known as *proportional fairness* (PF, Kelly et al. 1998), which we will study in § 3. Note that the formulation equation 2.7 is not fit for studying PF, since if we take $q \to -1$ (corresponding to $\alpha = 1$), then we obtain $\sum_i p_i \log f_i$, which need not be convex even when each f_i is (see also § 3.1.1).

Table 1: Different fairness concepts and their corresponding FL algorithms. f_i is the loss function for the ith client. The requirement of $\boldsymbol{\lambda}$ can be found in § 2.4 and § 3.2. We defer the description and the dual view of PropFair to § 3.

FL algorithm	**Principle**	**Objective**	**Constraints of $\boldsymbol{\lambda}$**
FedAvg	Utilitarian	$\sum_i p_i f_i$	$\lambda_i = p_i$
AFL	Egalitarian	$\max_i f_i$	$\boldsymbol{\lambda} \geq \mathbf{0}$, $\mathbf{1}^\top \boldsymbol{\lambda} \leq 1$
q-FFL	α-fairness	$\sum_i p_i f_i^{q+1}$	$\lambda_i \propto p_i f_i^q$, $\boldsymbol{\lambda} \geq \mathbf{0}$
TERM	n/a	$\sum_i p_i e^{\alpha f_i}$	$\lambda_i \propto p_i e^{\alpha f_i}$, $\boldsymbol{\lambda} \geq \mathbf{0}$, $\mathbf{1}^\top \boldsymbol{\lambda} = 1$
PropFair	**Proportional**	$-\sum_i p_i \log(M - f_i)$	$\lambda_i \propto \frac{p_i}{M - f_i}$, $\prod_i (\lambda_i / p_i)^{p_i} = 1$

2.4 Dual View of Fair FL Algorithms

In this subsection, we show that many existing fair FL algorithms can be treated in a surprisingly unified way. In fact, eq. 2.2 is equivalent to minimizing the Kolmogorov's generalized mean (Kolmogorov, 1930):

$$\mathsf{A}_\varphi(\boldsymbol{f}(\boldsymbol{\theta})) := \varphi^{-1}\left(\sum_{i=1}^n p_i \varphi(f_i(\boldsymbol{\theta}))\right). \tag{2.9}$$

Examples include $\varphi(f_i) = f_i$ (FedAvg), $\varphi(f_i) = f_i^{q+1}$ (q-FFL, $q \geq 0$) and $\varphi(f_i) = \exp(\alpha f_i)$ ($\alpha \geq 0$). The last choice is known as Tilted Empirical Risk Minimization (TERM, Li et al. 2020a).

We can now supply a dual view of the aforementioned FL algorithms that is perhaps more revealing. Concretely, let s be (strictly) increasing, convex and thrice differentiable. Then, the generalized mean function A_φ is convex iff $-s'/s''$ is convex (Theorem 1, Ben-Tal & Teboulle, 1986). Applying the convex conjugate of A_φ we obtain the equivalent problem:

$$\min_{\boldsymbol{\theta}} \mathsf{A}_\varphi(\boldsymbol{f}(\boldsymbol{\theta})) \equiv \min_{\boldsymbol{\theta}} \max_{\boldsymbol{\lambda} \geq \mathbf{0}} \sum_i \lambda_i f_i(\boldsymbol{\theta}) - \mathsf{A}_\varphi^*(\boldsymbol{\lambda}),\ \mathsf{A}_\varphi^*(\boldsymbol{\lambda}) := \sup_{\boldsymbol{f}} \boldsymbol{\lambda}^\top \boldsymbol{f} - \mathsf{A}_\varphi(\boldsymbol{f}), \tag{2.10}$$

where $\mathsf{A}_\varphi^*(\boldsymbol{\lambda})$ is the *convex conjugate* of A_φ. Note that $\boldsymbol{f} \geq \mathbf{0}$ and thus we require $\boldsymbol{\lambda} \geq \mathbf{0}$. Under strong duality, we may find the optimal dual variable $\boldsymbol{\lambda}^*$, with which our fair FL algorithms are essentially FedAvg with the fine-tuned weighting vector $\boldsymbol{\lambda}^*$.

Constraints of $\boldsymbol{\lambda}$. Solving the convex conjugate A_φ^* often gives additional constraints on $\boldsymbol{\lambda}$. For example, for FedAvg we can find that $\mathsf{A}_\varphi^*(\boldsymbol{\lambda}) = 0$ if $\lambda_i = p_i$ for all $i \in [n]$ and $\mathsf{A}_\varphi^*(\boldsymbol{\lambda}) = \infty$ otherwise. For $\varphi(f_i) = f_i^{q+1}$, we obtain the conjugate function corresponding to q-FFL:

$$\mathsf{A}_\varphi^*(\boldsymbol{\lambda}) = 0,\ \text{if } \boldsymbol{\lambda} \geq \mathbf{0} \text{ and } \sum_i p_i^{-1/q} \lambda_i^{(q+1)/q} \leq 1,\ \text{and } \infty \text{ otherwise.} \tag{2.11}$$

Bringing equation 2.11 into equation 2.10 and using Hölder's inequality we obtain the maximizer $\lambda_i \propto p_i f_i^q$. Similarly, we can derive the convex conjugate of TERM (Li et al., 2020a) as:

$$\mathsf{A}_\varphi^*(\boldsymbol{\lambda}) = \sum_i \frac{\lambda_i}{\alpha} \log \frac{\lambda_i}{p_i} \text{ if } \boldsymbol{\lambda} \geq \mathbf{0}, \boldsymbol{\lambda}^\top \mathbf{1} = 1,\ \text{and } \infty \text{ otherwise.} \tag{2.12}$$

The maximizer is achieved at $\lambda_i \propto p_i e^{\alpha f_i}$. In other words, TERM gives a higher weight to clients with worse losses. Detailed derivations of the convex conjugates can be found in Appendix D.

In Table 1, we summarize all the algorithms we have discussed, including their motivating principles, objectives as well as the constraints of $\boldsymbol{\lambda}$ induced by A_φ^*. Although the fair FL algorithms are motivated from different principles, most of them achieve a balance between utilitarianism and egalitarianism, thus allowing us to compare them on the same ground (§ 5).

3 Adapting Proportional Fairness to FL

Now we study how to add the missing piece mentioned in Section 2.3 to FL: proportional fairness. From a utility perspective, equation 2.8 with $\alpha = 1$ reduces to:

$$\max_{\mathbf{u}\in\mathcal{U}} \sum_i p_i \log u_i, \text{ with } p_i \geq 0 \text{ pre-defined, and } \sum_i p_i = 1. \tag{3.1}$$

Note that we now specify the domain of $\mathbf{u}$ to be $\mathcal{U} \subseteq \mathbb{R}^n_{++}$. The objective in equation 3.1 is sometimes known as the *Nash product* (up to logarithmic transformation), and the maximizer $\mathbf{u}^*$ is also called the *Nash bargaining solution* (NBS, Nash 1950). Axiomatic characterizations of the Nash bargaining solution are well-known, for instance by the following four axioms: Pareto optimality, symmetry, scale equivariance and monotonicity (e.g., Maschler et al., 2020, Theorem 16.35).

The first-order optimality condition (Bertsekas, 1997) of equation 3.1 can be written as:

$$\langle \mathbf{u} - \mathbf{u}^*, \nabla \sum_{i=1}^n p_i \log u_i^* \rangle \leq 0, \text{ for any } \mathbf{u} \in \mathcal{U}, \tag{3.2}$$

resulting in the following definition of proportional fairness (Kelly et al., 1998):

$$\mathbf{u}^* \in \mathcal{U} \text{ is proportionally fair if } \sum_i p_i \frac{u_i - u_i^*}{u_i^*} \leq 0, \text{ for any } \mathbf{u} \in \mathcal{U}. \tag{3.3}$$

Intuitively, $(u_i - u_i^*)/u_i^*$ is the relative utility gain for player i given its utility switched from u_i^* to u_i. PF simply states that at the solution $\mathbf{u}^*$, the average relative utility cannot be improved. For instance, for two players with $p_1 = p_2 = 1/2$ we have:

$$\frac{u_1 - u_1^*}{u_1^*} \leq -\frac{u_2 - u_2^*}{u_2^*}, \tag{3.4}$$

which says that if by deviating from the optimal solution (u_1^*, u_2^*), player 2 could gain p percentage more in terms of utility, then player 1 will have to lose a percentage at least as large as p.

The Nash bargaining solution is equivalent to the PF solution according to the following proposition:

Proposition 3.1 (**equivalence**, e.g. Kelly 1997; Boche & Schubert 2009)**.** *For any convex set $\mathcal{U} \in \mathbb{R}^n_{++}$, a point $u \in \mathcal{U}$ is the Nash bargaining solution iff it is proportionally fair. If $\mathcal{U}$ is non-convex, then a PF solution, when exists, is a Nash bargaining solution.*

A PF solution, whenever exists, is a Nash bargaining solution over $\mathcal{U}$. While the converse also holds if $\mathcal{U}$ is convex, for nonconvex $\mathcal{U}$, PF solutions may not exist. In contrast, NBS always exists if $\mathcal{U}$ is compact, and thus we solve equation 3.1 as a necessary condition of PF. From Jensen's inequality, we can show that:

$$\sum_i p_i \log u_i \leq \log \sum_i p_i u_i. \tag{3.5}$$

In other words, solving the NBS yields a lower bound of the averaged utility. On the other hand, if any of the utilities is close to 0, then the left hand side of equation 3.5 would decrease to $-\infty$. Therefore, the NBS does not yield extremely undesirable performance for any client. In a nutshell, the NBS achieves a balance between maximizing the averaged and the worst-case utilities.

3.1 The PropFair algorithm for federated learning

In order to realize proportional fairness in FL, we need to solve equation 3.1. With parametrization of u_i, our goal is to find a global model $\boldsymbol{\theta}$ to solve equation 3.1:

$$\max_{\boldsymbol{\theta}} \sum_i p_i \log u_i(\boldsymbol{\theta}). \tag{3.6}$$

3.1.1 What is the right choice of utilities?

One immediate question is: *how do we define these utilities in FL?* Ideally, the utility should be the test accuracy, which is unfortunately not amenable to optimize. Instead, we could use the training loss f_i. There are a few alternatives:

- Replace u_i with f_i as done in q-FFL, and minimize $\sum_i p_i \log f_i$;
- Replace u_i with f_i as done in q-FFL, and maximize $\sum_i p_i \log f_i$;
- Choose $u_i = M - f_i$, and maximize $\sum_i p_i \log(M - f_i)$, with M some hyperparameter to be determined.

The first approach will encourage the client losses to be even more disparate. For instance, suppose $p_1 = p_2 = \frac{1}{2}$, and then $(f_1, f_2) = (\frac{1}{3}, \frac{2}{3})$ has smaller product than $(f_1, f_2) = (\frac{1}{2}, \frac{1}{2})$. The second approach is not a choice either as it is at odds with minimizing client losses. Therefore, we are left with the third option. By contrast, for any $M \geq 1$ and $p_1 = p_2 = 1/2$, one can show that $(f_1, f_2) = (\frac{1}{2}, \frac{1}{2})$ always gives a better solution than $(f_1, f_2) = (\frac{1}{3}, \frac{2}{3})$. The resulting objective becomes:

$$\min_{\boldsymbol{\theta}} \pi(\boldsymbol{\theta}) := -\sum_i p_i \log(M - f_i(\boldsymbol{\theta})). \tag{3.7}$$

3.1.2 Huberization

However, the objective equation 3.7 also raises issues: what if $M - f_i$ is small and blows up the gradient, or even worse, what if $M - f_i$ is negative and the logarithm does not make sense at all? Inspired by Huber's approach of robust estimation (Huber, 1964), we propose a "huberized" version of equation 3.7:

$$\min_{\boldsymbol{\theta}} -\sum_i p_i \log_{[\epsilon]}(M - f_i(\boldsymbol{\theta})), \text{ with } \log_{[\epsilon]}(M - t) := \begin{cases} \log(M - t), & \text{if } t \leq M - \epsilon, \\ \log \epsilon - \frac{1}{\epsilon}(t - M + \epsilon), & \text{if } t > M - \epsilon. \end{cases} \tag{3.8}$$

Essentially, $\log_{[\epsilon]}(M - t)$ is a robust $\mathcal{C}^1$ extension of $\log(M - t)$ from $[0, M - \epsilon]$ to $\mathbb{R}_+$: its linear part ensures that at $t = M - \epsilon$, both the value and the derivative are continuous. If any f_i is close or greater than M, then equation 3.8 switches from logarithm to its linear version. Based on equation 3.8 we propose Algorithm 1 called *PropFair*. It modifies FedAvg (McMahan et al., 2017) with a simple drop-in replacement, by replacing the loss of each batch ℓ_S^i with $\log_{[\epsilon]}(M - \ell_S^i(\boldsymbol{\theta}))$. This allows easy adaptation of PropFair with minimal change into any of the current FL platforms, such as Flower (Beutel et al., 2020) and Tensorflow Federated.[1]

Remark. When $M \to \infty$ and $f_i(\boldsymbol{\theta})$ is small compared to M, the loss function for client i becomes:

$$f_i^{\log}(\boldsymbol{\theta}) = -\log(M - f_i(\boldsymbol{\theta})) \approx -\log M + \frac{f_i(\boldsymbol{\theta})}{M}.$$

Thus, FedAvg can be regarded as a first-order approximation of PropFair. We utilize this approximation in our implementation. Another way to obtain FedAvg is to take $\epsilon = M$ and thus $\log_{[\epsilon]}(M - t)$ always uses the linear branch. In contrast, if $\epsilon \to 0$, then equation 3.8 becomes more similar to equation 3.7.

3.2 Dual view of PropFair

With the dual view from Section 2.4, we can also treat PropFair as minimizing a weighted combination of loss functions (plus constants), similar to other fair FL algorithms. Note that if $\varphi(f_i) = -\log(M - f_i)$ in equation 2.9, then we have PropFair (see Table 1):

Proposition 3.2 (dual view of PropFair). *The generalized mean eq. 2.9 for PropFair can be written as:*

$$\mathsf{A}_\varphi(\boldsymbol{f}) = \max_{\boldsymbol{\lambda} \geq \mathbf{0}, \prod_i (\lambda_i / p_i)^{p_i} \geq 1} \boldsymbol{\lambda}^\top \boldsymbol{f} - M(\boldsymbol{\lambda}^\top \mathbf{1} - 1), \tag{3.9}$$

Solving the inner maximization of equation 2.10 gives $\prod_{i=1}^n \left(\frac{\lambda_i}{p_i}\right)^{p_i} = 1$ *and* $\lambda_i \propto \frac{p_i}{M - f_i}$.

Similar to TERM/q-FFL, PropFair puts a larger weight on worse-off clients with a larger loss.

[1] https://www.tensorflow.org/federated

Algorithm 1: PropFair

Input: global epoch T, client number n, loss function f_i for client i, number of samples n_i for client i, initial global model $\boldsymbol{\theta}_0$, local step number K_i, baseline M, threshold ϵ, $p_i = n_i/N$
for t ***in*** $0, 1 \ldots T-1$ **do**
  randomly select $\mathcal{S}_t \subseteq [n]$
  $\boldsymbol{\theta}_{t,0}^{(i)} = \boldsymbol{\theta}_t$ for $i \in \mathcal{S}_t$, $N = \sum_{i \in \mathcal{S}_t} n_i$
  for i ***in*** $\mathcal{S}_t$ **do** `// in parallel`
    $j = 1$, draw K_i batches from client i
    for S ***in*** *the* K_i *batches* **do**
      $\ell_S^i(\boldsymbol{\theta}) = \frac{1}{|S|} \sum_{(\mathbf{x},y) \in S} \ell(\boldsymbol{\theta}, (\mathbf{x}, y))$
      $f_i^{\log}(\boldsymbol{\theta}) = -\log_{[\epsilon]}(M - \ell_S^i(\boldsymbol{\theta}))$
      $\boldsymbol{\theta}_{t,j}^{(i)} \leftarrow \boldsymbol{\theta}_{t,j-1}^{(i)} - \eta \nabla f_i^{\log}(\boldsymbol{\theta}_{t,j-1}^{(i)})$, $j \leftarrow j+1$
  $\boldsymbol{\theta}_{t+1} = \sum_{i \in S_t} p_i \boldsymbol{\theta}_{t,K_i}^{(i)}$
Output: global model $\boldsymbol{\theta}_T$

4 The optimization side of PropFair

In this section, we discuss the convexity of our PropFair objective and show the convergence guarantee of Algorithm 1. This gives solid foundation for our algorithm, and potentially for many others in the scalarization class, equation 2.2. For simplicity we only study the case when $f_i \leq M - \epsilon$ for all i.

4.1 Convexity of the PropFair objective

Convexity is an important and desirable property in optimization (Boyd et al., 2004). With convexity, every stationary point is optimal (Bertsekas, 1997). From the composition rule, if f_i is convex for each client i, then $M - f_i$ is concave, and thus $\sum_i p_i \log(M - f_i(\boldsymbol{\theta}))$ is concave as well (e.g., Boyd et al., 2004). In other words, for convex losses, our optimization problem equation 3.7 is still convex as we are maximizing over concave functions. Moreover, our PropFair objective is convex *even* when f_i's are not. For example, this could happen if $f_i(\boldsymbol{\theta}) = M - \exp(\boldsymbol{\theta}^\top \mathbf{A}_i \boldsymbol{\theta})$ and each $\mathbf{A}_i$ is a positive definite matrix. In fact, it suffices to require each $M - f_i$ to be log-concave (Boyd et al., 2004).

4.2 Adaptive learning rate and curvature

Denote $\varphi(t) = -\log(M - t)$. We can compute the 1st- and 2nd-order derivatives of $\varphi \circ f_i$:

$$\nabla(\varphi \circ f_i) = \frac{\nabla f_i}{M - f_i}, \ \nabla^2(\varphi \circ f_i) = \frac{(M - f_i)\nabla^2 f_i + (\nabla f_i)(\nabla f_i)^\top}{(M - f_i)^2}. \tag{4.1}$$

This equation tells us that at each local gradient step, the gradient $\nabla(\varphi \circ f_i)$ has the same direction as ∇f_i, and the only difference is the step size. Compared to FedAvg, PropFair automatically has an *adaptive learning rate* for each client. When the local client loss function f_i is small, the learning rate is smaller; when f_i is large, the learning rate is larger. This agrees with our intuition that to achieve fairness, a worse-off client should be allowed to take a more aggressive step, while a better-off client moves more slowly to "wait for" other clients.

In the Hessian $\nabla^2(\varphi \circ f_i)$, an additional positive semi-definite (p.s.d.) term $(\nabla f_i)(\nabla f_i)^\top$ is added. Thus, $\nabla^2(\varphi \circ f_i)$ can be p.s.d. even if the original Hessian $\nabla^2 f_i$ is not. Moreover, the denominator $(M - f_i)^2$ has a similar effect of coordinating the curvatures of various clients as in the gradients.

4.3 Convergence results

Let us now formally prove the convergence of PropFair by bounding its progress, using standard assumptions (Li et al., 2019; Reddi et al., 2020) such as Lipschitz smoothness and bounded variance. Every norm discussed in this subsection is Euclidean (including the proofs in Appendix A).

In fact, PropFair can be treated as an easy variant of FedAvg, with the local objective f_i replaced with $f_i^{\log}$. Therefore, we just need to prove the convergence of FedAvg and the convergence of PropFair would be a simple corollary. In general, similar results also hold for objectives in the form of equation 2.2.

Let us state the assumptions first. Since in practice we use stochastic gradient descent (SGD) for training, we consider the effect of mini-batches. We also assume that the (local) variance of mini-batches and the (global) variance among clients are bounded.

Assumption 4.1 (**Lipschitz smoothness and bounded variances**). *Each function f_i is L-Lipschitz smooth, i.e., for any $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \mathbb{R}^d$ and any $i \in [n]$, we have $\|\nabla f_i(\boldsymbol{\theta}) - \nabla f_i(\boldsymbol{\theta}')\| \leq L\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|$ for any $\boldsymbol{\theta} \in \mathbb{R}^d$. For any $i, j \in [n]$, $f_i - f_j$ is σ-Lipschitz continuous and $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i} \|\nabla \ell(\boldsymbol{\theta}, (\mathbf{x}, y)) - \nabla f_i(\boldsymbol{\theta})\|^2 \leq \sigma_i^2 \ \forall \boldsymbol{\theta} \in \mathbb{R}^d$.*

Algorithm 2: FedAvg

1 **Input:** global epoch T, client number n, loss function f_i, number of samples n_i for client i, initial global model $\boldsymbol{\theta}_0$, local step number K_i for client i, batch size m, learning rate η, $p_i = n_i/N$
2 **for** t *in* $0, 1 \ldots T-1$ **do**
3 randomly select $\mathcal{S}_t \subseteq [n]$
4 $\boldsymbol{\theta}_t^{(i)} = \boldsymbol{\theta}_t$ for $i \in \mathcal{S}_t$, $N = \sum_{i \in \mathcal{S}_t} n_i$
5 **for** i *in* $\mathcal{S}_t$ **do** `// in parallel`
6 starting from $\boldsymbol{\theta}_t^{(i)}$, take K_i local SGD steps on f_i to find $\boldsymbol{\theta}_{t+1}^{(i)}$
7 $\boldsymbol{\theta}_{t+1} = \sum_{i \in \mathcal{S}_t} p_i \boldsymbol{\theta}_{t+1}^{(i)}$
8 **Output:** global model $\boldsymbol{\theta}_T$

Following the notations of Reddi et al. (2020), we use σ^2 and σ_i^2 to denote the global and local variances for client i. This assumption allows us to obtain the convergence result for FedAvg (see Algorithm 2). For easy reference, we include FedAvg (McMahan et al., 2017) in Algorithm 2, whose goal is to optimize the overall performance. At each round, each client takes local SGD steps to minimize the loss function based on the client data. Afterwards, the server computes a weighted average of the parameters of these participating clients, and shares this average among them. Note that for client i, the number of local steps is K_i with learning rate η. In line 3 of Algorithm 2, if $\mathcal{S}_t = [n]$ then we call it *full participation*, otherwise it is called *partial participation*. We prove the following convergence result of FedAvg. Note that we defined F in equation 2.3, and m is the batch size.

Theorem 4.2 (**FedAvg**). *Given Assumption 4.1, assume that the local learning rate satisfies $\eta K_i \leq \frac{1}{6L}$ for any $i \in [n]$ and*

$$\eta \leq \frac{1}{L}\sqrt{\frac{1}{24(e-2)(\sum_i p_i^2)(\sum_i K_i^4)}}. \tag{4.2}$$

Running Algorithm 2 for T global epochs we have:

$$\min_{0 \leq t \leq T-1} \mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 \leq \frac{12}{(11\mu - 9)\eta}\left(\frac{F_0 - F^*}{T} + \Psi_\sigma\right),$$

with $\mu = \sum_i p_i K_i$ for full participation and $\mu = \min_i K_i$ for partial participation, $F_0 = F(\boldsymbol{\theta}_0)$, $F^* = \min_{\boldsymbol{\theta}} F(\boldsymbol{\theta})$ the optimal value, and

$$\Psi_\sigma = \frac{\eta}{2}\Big(\sum_{i=1}^n p_i^2\Big)\left[\sum_{i=1}^n K_i^2\left(\frac{\sigma_i^2}{m} + 2\sigma^2\right) + 2(e-2)\eta^2 L^2 \sum_{i=1}^n K_i^3\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2\right)\right].$$

Compared to our result, Reddi et al. (2020) only considers a homogeneous number of steps for each client, and uniform aggregation for Adam-type methods. Li et al. (2019) also considers a homogeneous number of steps and adds strong convexity for their proof. These assumptions may not reflect the practical use of

FedAvg. For example, usually each client has a different number of samples and they may take different numbers of local updates. Moreover, for neural networks, (global) strong convexity is usually not present.

Based on Theorem 4.2, we can prove the convergence for any other FL algorithms which minimize eq. 2.2, as long as $-\log(M - f_i)$ satisfies a similar assumption as Assumption 4.1. For the PropFair algorithm, as an example, we need to additionally assume the Lipschitzness and bounded variances for the client losses:

Assumption 4.3 (**boundedness, Lipschitz continuity and bounded variances for client losses**). *For any $i \in [n]$, $\boldsymbol{\theta} \in \mathbb{R}^d$ and any batch $S \sim \mathcal{D}_i^m$ of m i.i.d. samples, we have:*

$$0 \leq \ell_S(\boldsymbol{\theta}) := \frac{1}{|S|} \sum\nolimits_{(\mathbf{x},y)\in S} \ell(\boldsymbol{\theta}, (\mathbf{x}, y)) \leq \frac{M}{2},$$

and $\|f_i(\boldsymbol{\theta}) - f_i(\boldsymbol{\theta}')\| \leq L_0\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|$. We also assume that for any $i, j \in [n]$ and $\boldsymbol{\theta} \in \mathbb{R}^d$, $\|f_i(\boldsymbol{\theta}) - f_j(\boldsymbol{\theta})\|^2 \leq \sigma_0^2$ and $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i} \|\ell(\boldsymbol{\theta}, (\mathbf{x}, y)) - f_i(\boldsymbol{\theta})\|^2 \leq \sigma_{0,i}^2$ hold.

we can obtain the convergence guarantee of PropFair to a neighborhood of some stationary point:

Theorem 4.4 (**PropFair**). *Denote $\widetilde{L} = \frac{4}{M^2}(\frac{3}{2}ML + L_0^2)$ and $p_i = \frac{n_i}{N}$. Given Assumptions 4.1 and 4.3, assume that the local learning rate satisfies:*

$$\eta \leq \min\left\{\min_{i\in[n]} \frac{1}{6\widetilde{L}K_i}, \frac{1}{L}\sqrt{\frac{1}{24(e-2)(\sum_i p_i^2)(\sum_i K_i^4)}}\right\}. \tag{4.3}$$

By running Algorithm 1 for T global epochs we have:

$$\min_{0\leq t\leq T-1} \mathbb{E}\|\nabla\pi(\boldsymbol{\theta}_t)\|^2 \leq \frac{12}{(11\mu - 9)\eta}\left(\frac{\pi_0 - \pi^*}{T} + \widetilde{\Psi}_\sigma\right),$$

with $\mu = \sum_i p_i K_i$ for full participation and $\mu = \min_i K_i$ for partial participation, $\pi_0 = \pi(\boldsymbol{\theta}_0)$, $\pi^* = \min_{\boldsymbol{\theta}} \pi(\boldsymbol{\theta})$ the optimal value, and

$$\widetilde{\Psi}_\sigma = \frac{\eta}{2}\Big(\sum_{i=1}^n p_i^2\Big)\left[\sum_{i=1}^n K_i^2\left(\frac{\widetilde{\sigma}_i^2}{m} + 2\widetilde{\sigma}^2\right) + 2(e-2)\eta^2 L^2 \sum_{i=1}^n K_i^3\left(\frac{\widetilde{\sigma}_i^2}{m} + 6K_i\widetilde{\sigma}^2\right)\right],$$

where $\widetilde{\sigma}_i^2 = \frac{8}{M^4}(9M^2\sigma_i^2 + 4L_0^2\sigma_{0,i}^2)$ and $\widetilde{\sigma} = \frac{4}{M}\left(\frac{3}{2}\sigma + \frac{L_0}{M}\sigma_0\right)$.

Our result is quite general: we allow for both full and partial participation; multiple and heterogeneous local steps; and non-uniform aggregation with weight p_i. The variance term $\widetilde{\Psi}_\sigma$ decreases with smaller local steps K_i, which agrees with our intuition that each K_i should be as small as possible given the communication constraint. Moreover, to minimize $\sum_i p_i^2$ we should take $p_i = 1/n$ for each client i, which means if the samples are more evenly distributed across clients, the error is smaller. In presence of convexity, we can see that PropFair converges to a neighborhood of the optimal solution, and the size of the neighborhood is controlled by the heterogeneity of clients and the variance of mini-batches. When we have the global variance term $\widetilde{\sigma} = 0$, a large batch size m, and the local step number $K_i = 1$ for all $i \in [n]$, our result reduces to the standard result of gradient descent, with $\min_{0\leq t\leq T-1} \mathbb{E}\|\nabla\pi(\boldsymbol{\theta}_t)\|^2 \leq O(\frac{1}{T})$.

5 Experiments

In this section, we verify properties of PropFair by answering the following questions: (1) can PropFair achieve proportional fairness as in equation 3.3? (2) what balance does PropFair achieve between the average and worst-case performances? We report them separately in Section 5.2 and Section 5.3.

5.1 Experimental setup

We first give details on our datasets, models and hyperparameters, which are in accordance with existing works. See Appendix C for additional experimental setup. A comprehensive survey of benchmarking FL algorithms can be found in e.g. Caldas et al. (2018); He et al. (2020).

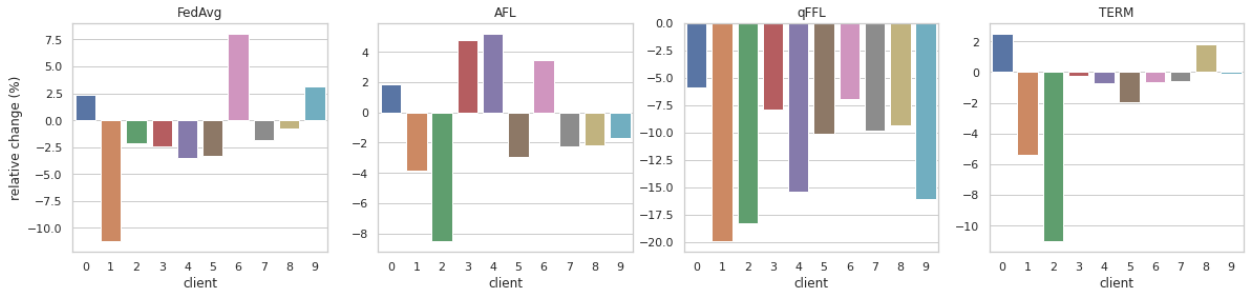

Figure 1: The relative improvement/deterioration $(u_i - u_i^*)/u_i^*$ of the test accuracy u_i of each client i of other baseline algorithms compared to PropFair. The dataset is CIFAR-10. The (weighted) average of the relative changes for FedAvg, AFL, q-FFL and TERM are respectively: -2.21%, -1.32%, -12.95%, -1.79%. We choose the hyperparameters based on Table 3 in Appendix C.

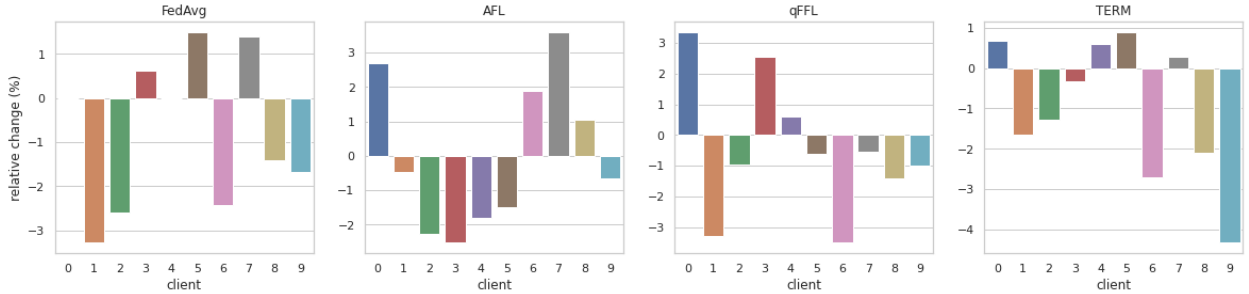

Figure 2: The relative improvement/deterioration $(u_i - u_i^*)/u_i^*$ of the test accuracy of each client i, pretrained with PropFair and fine-tuned with another baseline. The dataset is CIFAR-100. The average of the relative changes for FedAvg, AFL, q-FFL and TERM are respectively: -0.86%, $+0.05\%$, -0.67%, -1.01%.

Datasets. We follow standard benchmark datasets as in the existing literature, including CIFAR-10/100 (Krizhevsky et al., 2009), TinyImageNet (Le & Yang, 2015) and Shakespeare (McMahan et al., 2017). For vision datasets (CIFAR-10/100/TinyImageNet), the task is image classification, and following Wang et al. (2019b) we use Dirichlet allocation to split the dataset into different clients. For the language dataset (Shakespeare), the task is next-character prediction. We use the default realistic partition based on different users. We first partition the dataset into different clients, and further split each client dataset into its own training and test sets. This reflects the real scenario, where each client evaluates the performance by itself.

Models, optimizer and loss function. For vision datasets we use ResNet-18 (He et al., 2016) with Group Normalization (Wu & He, 2018). As discussed by Hsieh et al. (2020), Group Normalization (with `num_groups=2`) works better than batch normalization, especially in the federated settings. For the Shakespeare dataset, we use LSTM (Hochreiter & Schmidhuber, 1997). We find the best learning rates through grid search (see Appendix C).

Other hyperparameters. We implement full participation and one local epoch throughout (with many local steps for each client). Due to data heterogeneity, the number of local steps K_i for each client i varies. For CIFAR-10/100 we partition the data into 10 clients; for TinyImageNet/Shakespeare we choose 20 clients.

Evaluation metrics. We validate proportional fairness equation 3.3 of our PropFair algorithm, where we treat each u_i as the test accuracy of client i. To show that PropFair achieves a proper balance between utilitarian and egalitarian fairness, we use the average and the worst 10% test accuracies. These are standard fairness metrics used in the literature (e.g. Li et al., 2020a;c). In Appendix C we also present other standard metrics such as standard deviation and worst 20%.

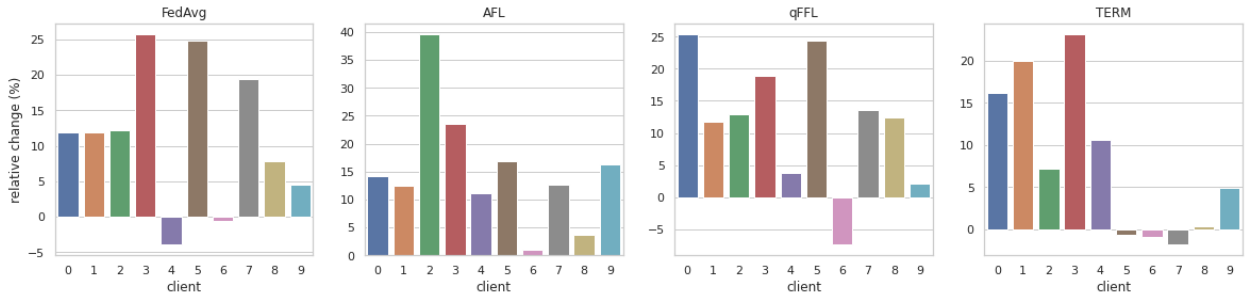

Figure 3: The relative improvement/deterioration $(u_i - u_i^*)/u_i^*$ of test accuracy of each client i, pretrained with another baseline and fine-tuned with PropFair. The dataset is CIFAR-100. The average of the relative changes over FedAvg, AFL, q-FFL and TERM are respectively: 10.88%, 13.93%, 11.58%, 8.67%.

5.2 Verification of proportional fairness

In this subsection, we show that PropFair can, to some extent, achieve proportional fairness as defined in equation 3.3. We treat u_i as the test accuracy of client i, and compute

$$\sum_i p_i \frac{u_i - u_i^*}{u_i^*}, \tag{5.1}$$

with $p_i = n_i/N$ and $\mathbf{u}^* := (u_1^*, \ldots, u_n^*)$ the test accuracies obtained by the PropFair model. Although we cannot verify equation 5.1 for every $\mathbf{u}$, we can at least validate the negativity for some competitive $\mathbf{u}$'s, e.g., models learned by other fair FL algorithms.

5.2.1 CIFAR-10

We first compute equation 5.1 where $\mathbf{u}^*$ is the test accuracies obtained by PropFair and $\mathbf{u}$ is the test accuracies found by one of the other fair FL algorithms, including FedAvg, AFL, q-FFL and TERM. Figure 1 shows the relative changes of each client, $(u_i - u_i^*)/u_i^*$, from which we can see that compared to the solution found by PropFair, for another fair FL solution, most clients are degraded by a large relative amount, and only a few clients are improved by a small amount.

5.2.2 CIFAR-100

In fact, we may compute equation 3.3 with a stronger $\mathbf{u}$. For CIFAR-100, we still treat $\mathbf{u}^*$ as the test accuracies obtained by PropFair. The difference is that we use $\mathbf{u}^*$ as the initialization, and fine-tune with other fair FL algorithms, to find $\mathbf{u}$. If other fair FL algorithms cannot improve the proportional fairness of $\mathbf{u}^*$, then equation 5.1 should be negative. As we see in Figure 2, this result indeed holds approximately (except the slight improvement for AFL).

By contrast, none of the baseline fair FL algorithms can achieve the same level of proportional fairness as our PropFair. In Figure 3, we see that if we start from a model pretrained with a baseline fair FL algorithm, and fine-tune with our Propfair, most client performances are improved, sometimes by a large margin.

5.3 Comparison between PropFair and existing fair FL algorithms on other metrics

In Figure 4, we compare PropFair with existing fair FL algorithms using the average and the worst 10% test accuracies across clients, including FedAvg (McMahan et al., 2017), q-FFL (Li et al., 2020c), AFL (Mohri et al., 2019) and TERM (Li et al., 2020a).

Average performance. From Figure 4 we can see that PropFair does not always yield the best average performance, e.g., compared to q-FFL on TinyImageNet. This is expected, since maximizing the Nash product does not necessarily give the best average performance. Nevertheless, PropFair remains competitive.

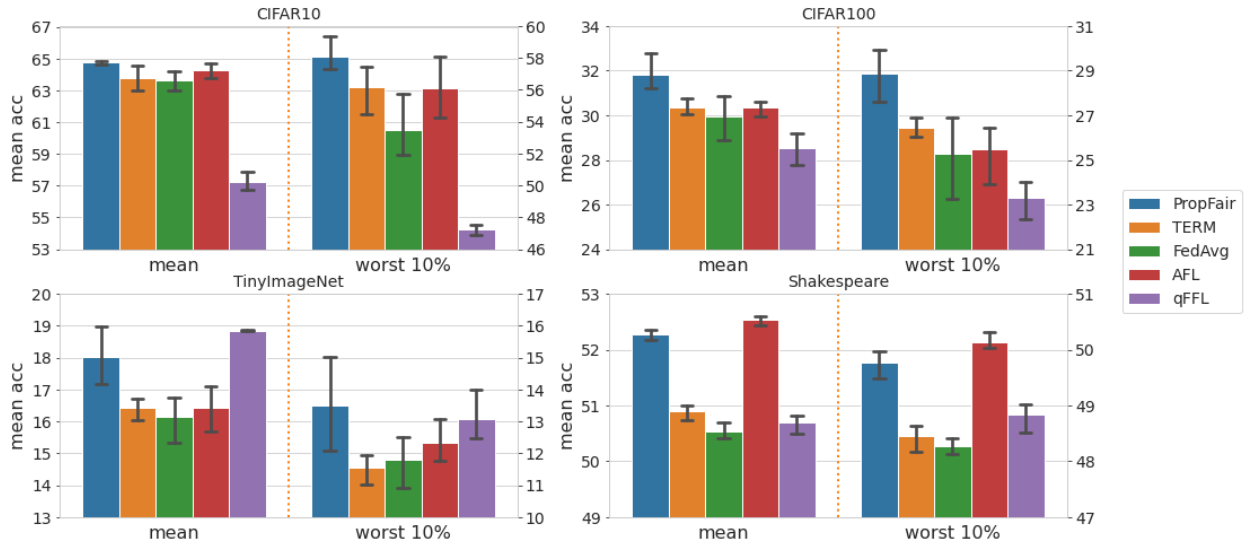

Figure 4: Mean and worst 10% test accuracies for different algorithms. The accuracies are in percentage. (**top left**): CIFAR-10; (**top right**): CIFAR-100; (**bottom left**): Shakespeare; (**bottom right**): StackOverflow. All subfigures share the same legends and axis labels.

Somewhat surprisingly, FedAvg does not always achieve the best average performance, which might be due to optimization issues (Pathak & Wainwright, 2020).

Worst 10% performance. We also compare the worst 10% performance of various fair FL algorithms. We observe that PropFair achieves the state-of-the-art in terms of the worst 10% performance, across various vision and language datasets. This is within our expectation, since from equation 3.5 and equation 3.7 we can see that low utility in *any* of the clients would result in a small Nash product.

Specifically, although AFL directly maximizes the worst-case loss function, it does not always achieve the best worst-case performances (see Table 5 in Appendix C), especially for vision datasets. This might be due to the generalization issue of AFL (see Appendix B).

6 Related Work in Fair FL

We review recent related work for fair federated learning. In additional to AFL (Mohri et al., 2019), q-FFL (Li et al., 2020c) and TERM (Li et al., 2020a), there have been other approaches for fairness in FL. For example, FedMGDA+ (Hu et al., 2022) defines fairness as achieving the Pareto frontier and they proposed to use the MGDA algorithm. As another example, GIFAIR-FL (Yue et al., 2021) encourages the similarity of the client losses by adding a regularization term of the pairwise ℓ_1 distances. Last but not least, Ditto (Li et al., 2021b) proposed a personalization approach to obtain fairness and robustness. A comprehensive recent survey of fairness in FL can be found in Shi et al. (2021), and we have included additional papers of fairness (in FL and in general) in Appendix E.

7 Conclusions

In this work, we introduce the concept of Proportional Fairness (PF) into the field of federated learning (FL), which is deeply rooted in cooperative game theory. By showing the connection between PF and the Nash bargaining solution, we propose PropFair that maximizes the product of client utilities. This guarantees PropFair to have good worst-case performance without sacrificing the total utility much. We verify proportional fairness and the balance between utilitarian and egalitarian fairness in our extensive experiments. As we have shown, many fair FL algorithms, including PropFair, can be unified using Kolmogorov's generalized mean, the deeper understanding of which may lead to future design of fair FL algorithms.

Broader Impact Statement

With the wide deployment of federated learning, how to ensure fairness in FL algorithms has become a major concern. In this work, we study proportional fairness in FL to make FL systems fairer and thus more trustworthy. This could have important positive social impacts as well. We are not aware of potential negative societal impacts yet but we welcome discussions on them.

References

Charles Audet, Gilles Savard, and Walid Zghal. Multiobjective optimization through a series of single-objective formulations. *SIAM Journal on Optimization*, 19(1):188–210, 2008.

Pranjal Awasthi, Corinna Cortes, Yishay Mansour, and Mehryar Mohri. Beyond individual and group fairness. *arXiv preprint arXiv:2008.09490*, 2020.

Sina Baharlouei, Maher Nouiehed, Ahmad Beirami, and Meisam Razaviyayn. Rényi fair inference. In *International Conference on Learning Representations*, 2019.

Solon Barocas, Moritz Hardt, and Arvind Narayanan. Fairness in machine learning. *NIPS tutorial*, 2017.

Aharon Ben-Tal and Marc Teboulle. Expected utility, penalty functions, and duality in stochastic nonlinear programming. *Management Science*, 32(11):1445–1466, 1986. URL `https://doi.org/10.1287/mnsc.32.11.1445`.

Jeremy Bentham. *An Introduction to the Principles of Morals and Legislation*. For T. Payne and Son, at the Mews Gate, 1780.

Dimitri Bertsekas and Robert Gallager. *Data networks*. Athena Scientific, 1987.

Dimitri P Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.

Dimitris Bertsimas, Vivek F Farias, and Nikolaos Trichakis. The price of fairness. *Operations research*, 59 (1):17–31, 2011.

Daniel J Beutel, Taner Topal, Akhil Mathur, Xinchi Qiu, Titouan Parcollet, Pedro PB de Gusmão, and Nicholas D Lane. Flower: A friendly federated learning research framework. *arXiv preprint arXiv:2007.14390*, 2020.

Holger Boche and Martin Schubert. Nash Bargaining and proportional fairness for wireless systems. *IEEE/ACM Transactions on Networking*, 17(5):1453–1466, October 2009.

Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečnỳ, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

Mina Dashti, Paeiz Azmi, and Keivan Navaie. Harmonic mean rate fairness for cognitive radio networks with heterogeneous traffic. *Transactions on Emerging Telecommunications Technologies*, 24(2), 2013. ISSN 2161-3915. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.2541`.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pp. 214–226, New York, NY, USA, January 2012. Association for Computing Machinery. ISBN 978-1-4503-1115-1. doi: 10.1145/2090236.2090255. URL `https://doi.org/10.1145/2090236.2090255`.

Martin Gebel. *Multivariate calibration of classifier scores into the probability space*. PhD thesis, Citeseer, 2009.

Arthur M Geoffrion. Proper efficiency and the theory of vector maximization. *Journal of mathematical analysis and applications*, 22(3):618–630, 1968.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. October 2016. URL `https://arxiv.org/abs/1610.02413v1`.

Chaoyang He, Songze Li, Jinhyun So, Xiao Zeng, Mi Zhang, Hongyi Wang, Xiaoyang Wang, Praneeth Vepakomma, Abhishek Singh, Hang Qiu, et al. FedML: A research library and benchmark for federated machine learning. *arXiv preprint arXiv:2007.13518*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Kevin Hsieh, Amar Phanishayee, Onur Mutlu, and Phillip Gibbons. The non-iid data quagmire of decentralized machine learning. In *International Conference on Machine Learning*, pp. 4387–4398. PMLR, 2020.

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. doi: 10.1109/TNSE.2022.3169117.

Peter J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1): 73 – 101, 1964. doi: 10.1214/aoms/1177703732. URL `https://doi.org/10.1214/aoms/1177703732`.

Ahmed Imteaj, Urmish Thakker, Shiqiang Wang, Jian Li, and M Hadi Amini. A survey on federated learning for resource-constrained IoT devices. *IEEE Internet of Things Journal*, 9(1):1–24, 2021.

Johannes Jahn et al. *Vector optimization*. Springer, 2009.

Rajendra K Jain, Dah-Ming W Chiu, William R Hawe, et al. A quantitative measure of fairness and discrimination. *Eastern Research Laboratory, Digital Equipment Corporation, Hudson, MA*, 1984.

Jiawen Kang, Zehui Xiong, Dusit Niyato, Yuze Zou, Yang Zhang, and Mohsen Guizani. Reliable federated learning for mobile networks. *IEEE Wireless Communications*, 27(2):72–80, April 2020.

FP Kelly, AK Maulloo, and DKH Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49(3):237–252, March 1998. ISSN 1476-9360. URL `https://doi.org/10.1057/palgrave.jors.2600523`.

Frank Kelly. Charging and rate control for elastic traffic. *European transactions on Telecommunications*, 8 (1):33–37, 1997.

Andrey Kolmogorov. On the notion of mean. *Atti della Academia Nazionale dei Lincei*, 12(9):388–391, 1930. URL `https://link.springer.com/book/9789027727961`. reprinted in "Selected Works I of Andrey Kolmogorov: Mathematics and Mechanics", pp. 144–146, 1991.

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

Matt Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 4069–4079, 2017.

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231n*, 2015.

Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. Tilted empirical risk minimization. In *International Conference on Learning Representations*, 2020a.

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, 2020b.

Tian Li, Maziar Sanjabi, Ahmad Beirami, and Virginia Smith. Fair resource allocation in federated learning. In *International Conference on Learning Representations*, 2020c.

Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. On tilted losses in machine learning: Theory and applications. *arXiv: 2109.06141*, 2021a.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6357–6368. PMLR, 2021b.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of FedAvg on non-iid data. In *International Conference on Learning Representations*, 2019.

Guodong Long, Yue Tan, Jing Jiang, and Chengqi Zhang. Federated learning for open banking. In *Federated learning*, pp. 240–254. Springer, 2020.

FA Lootsma, TW Athan, and PY Papalambros. Controlling the search for a compromise solution in multi-objective optimization. *Engineering Optimization+ A35*, 25(1):65–81, 1995.

Lingjuan Lyu, Jiangshan Yu, Karthik Nandakumar, Yitong Li, Xingjun Ma, Jiong Jin, Han Yu, and Kee Siong Ng. Towards fair and privacy-preserving federated deep models. *IEEE Transactions on Parallel and Distributed Systems*, 31(11):2524–2541, November 2020.

Michael Maschler, Eilon Solan, and Shmuel Zamir. Game theory. *Cambridge University Press, Cambridge, second edition*, 2020.

Eric Maskin. A theorem on utilitarianism. *The Review of Economic Studies*, 45(1):93–96, 1978.

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

Jeonghoon Mo and Jean Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking*, 8(5):556–567, October 2000. ISSN 1063-6692. doi: 10.1109/90.879343. URL `https://doi.org/10.1109/90.879343`.

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

John Forbes Nash. The bargaining problem. *Econometrica*, 18(2):155–162, 1950.

Reese Pathak and Martin J Wainwright. FedSplit: an algorithmic framework for fast federated optimization. *Advances in Neural Information Processing Systems*, 33:7057–7066, 2020.

Judea Pearl. Models, reasoning and inference. *Cambridge University Press*, 19, 2000.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon M Kleinberg, and Kilian Q Weinberger. On fairness and calibration. In *NIPS*, 2017.

John Rawls. Some reasons for the maximin criterion. *The American Economic Review*, 64(2):141–146, 1974.

John Rawls. *A theory of justice: Revised edition*. Harvard university press, 1999.

Sashank J Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*, 2020.

Amartya Sen. Chapter 22 Social choice theory. volume 3 of *Handbook of Mathematical Economics*, pp. 1073–1181. Elsevier, 1986. URL https://www.sciencedirect.com/science/article/pii/S1573438286030047. ISSN: 1573-4382.

Hanbyul Seo and Byeong Gi Lee. Proportional-fair power allocation with CDF-based scheduling for fair and efficient multiuser OFDM systems. *IEEE Transactions on Wireless Communications*, 5(5):978–983, May 2006.

William Shakespeare. The complete works of William Shakespeare. 1614. URL https://www.gutenberg.org/ebooks/100.

Yuxin Shi, Han Yu, and Cyril Leung. A survey of fairness-aware federated learning. *arXiv preprint arXiv:2111.01872*, 2021.

Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. In *International Conference on Learning Representations*, 2019a.

Kangkang Wang, Rajiv Mathews, Chloé Kiddon, Hubert Eichner, Françoise Beaufays, and Daniel Ramage. Federated evaluation of on-device personalization. October 2019b. URL https://arxiv.org/abs/1910.10252v1. arXiv: 1910.10252.

Tianhao Wang, Johannes Rausch, Ce Zhang, Ruoxi Jia, and Dawn Song. A principled approach to data valuation for federated learning. In Qiang Yang, Lixin Fan, and Han Yu (eds.), *Federated Learning: Privacy and Incentive*, Lecture Notes in Computer Science, pp. 153–167. Springer International Publishing, 2020.

Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

Jie Xu, Benjamin S Glicksberg, Chang Su, Peter Walker, Jiang Bian, and Fei Wang. Federated learning for healthcare informatics. *Journal of Healthcare Informatics Research*, 5(1):1–19, 2021.

Han Yu, Zelei Liu, Yang Liu, Tianjian Chen, Mingshu Cong, Xi Weng, Dusit Niyato, and Qiang Yang. A fairness-aware incentive scheme for federated learning. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '20, pp. 393–399, New York, NY, USA, February 2020. Association for Computing Machinery.

Po-Lung Yu and Milan Zeleny. The set of all nondominated solutions in linear cases and a multicriteria simplex method. *Journal of Mathematical Analysis and Applications*, 49(2):430–468, 1975.

Xubo Yue, Maher Nouiehed, and Raed Al Kontar. GIFAIR-FL: An approach for group and individual fairness in federated learning. *arXiv: 2108.02741*, 2021.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P. Gummadi. Fairness beyond disparate treatment & disparate impact: learning classification without disparate mistreatment. *Proceedings of the 26th International Conference on World Wide Web*, pp. 1171–1180, April 2017. doi: 10.1145/3038912.3052660. URL http://arxiv.org/abs/1610.08452. arXiv: 1610.08452.

Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *Proceedings of the 30th International Conference on Machine Learning*, pp. 325–333. PMLR, May 2013. URL https://proceedings.mlr.press/v28/zemel13.html. ISSN: 1938-7228.

Han Zhao and Geoff Gordon. Inherent tradeoffs in learning fair representations. *Advances in neural information processing systems*, 32:15675–15685, 2019.

A Proofs

Proposition 3.1 (**equivalence**, e.g. Kelly 1997; Boche & Schubert 2009)**.** *For any convex set $\mathcal{U} \in \mathbb{R}^n_{++}$, a point $u \in \mathcal{U}$ is the Nash bargaining solution iff it is proportionally fair. If $\mathcal{U}$ is non-convex, then a PF solution, when exists, is a Nash bargaining solution.*

Proof. The Nash bargaining solution $\mathbf{u}^*$ is equivalent to the maximum of the following:

$$\max_{\mathbf{u}\in\mathcal{U}} \sum_{i=1}^{n} \lambda_i \log u_i. \tag{A.1}$$

Since $\mathcal{U}$ is convex and $\sum_{i=1}^n \lambda_i \log u_i$ is concave in $\mathbf{u}$, the necessary and sufficient optimality condition (e.g,. Bertsekas, 1997) is:

$$\langle \mathbf{u} - \mathbf{u}^*, \nabla \sum_{i=1}^{n} \lambda_i \log u_i^* \rangle \leq 0, \text{ for any } \mathbf{u} \in \mathcal{U}, \tag{A.2}$$

or equivalently, equation 3.3. If $\mathcal{U}$ is non-convex, then the optimality condition equation A.2 also holds for the convex hull of $\mathcal{U}$. Therefore, $\mathbf{u}^*$ is a maximizer of $\sum_{i=1}^n \lambda_i \log u_i$ in the convex hull of $\mathcal{U}$ and thus $\mathcal{U}$. □

Theorem 4.2 (**FedAvg**)**.** *Given Assumption 4.1, assume that the local learning rate satisfies $\eta K_i \leq \frac{1}{6L}$ for any $i \in [n]$ and*

$$\eta \leq \frac{1}{L}\sqrt{\frac{1}{24(e-2)(\sum_i p_i^2)(\sum_i K_i^4)}}. \tag{4.2}$$

Running Algorithm 2 for T global epochs we have:

$$\min_{0\leq t\leq T-1} \mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 \leq \frac{12}{(11\mu-9)\eta}\left(\frac{F_0 - F^*}{T} + \Psi_\sigma\right),$$

with $\mu = \sum_i p_i K_i$ for full participation and $\mu = \min_i K_i$ for partial participation, $F_0 = F(\boldsymbol{\theta}_0)$, $F^* = \min_{\boldsymbol{\theta}} F(\boldsymbol{\theta})$ the optimal value, and

$$\Psi_\sigma = \frac{\eta}{2}\Big(\sum_{i=1}^{n} p_i^2\Big)\left[\sum_{i=1}^{n} K_i^2\left(\frac{\sigma_i^2}{m} + 2\sigma^2\right) + 2(e-2)\eta^2 L^2 \sum_{i=1}^{n} K_i^3\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2\right)\right].$$

Proof. We first assume full participation in the following theorem. The partial participation version is an easy extension and we discuss it in the end. We use $\boldsymbol{\theta}_{t,j}^{(i)}$ to denote the model parameters of client i at global epoch t and local step j. Due to the synchronization step, we have $\boldsymbol{\theta}_{t,0}^{(i)} = \boldsymbol{\theta}_t$, the global model at step t, and

$$\boldsymbol{\theta}_{t+1} = \sum_{i=1}^{n} p_i \boldsymbol{\theta}_{t,K_i}^{(i)},\ p_i = \frac{n_i}{N}, \tag{A.3}$$

where K_i is the local number of steps of client i. We also have:

$$\boldsymbol{\theta}_{t,j}^{(i)} = \boldsymbol{\theta}_{t,j-1}^{(i)} - \eta \boldsymbol{g}_{t,j}^{(i)}, \text{ for all } j \in [K_i]. \tag{A.4}$$

where $\boldsymbol{g}_{t,j}^{(i)}$ is an unbiased estimator of $\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})$ for $j \in [K_i]$. Combining equation A.3 and equation A.4 we have:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} \boldsymbol{g}_{t,j}^{(i)}. \tag{A.5}$$

Part I Since each f_i is L-Lipschitz smooth, so is their average $F = \sum_i \lambda_i f_i$, from which we obtain that:

$$F(\boldsymbol{\theta}_{t+1}) \leq F(\boldsymbol{\theta}_t) + \langle \nabla F(\boldsymbol{\theta}_t), \boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t \rangle + \frac{L}{2}\|\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t\|^2. \tag{A.6}$$

Plugging in equation A.5 yields:

$$F(\boldsymbol{\theta}_{t+1}) \leq F(\boldsymbol{\theta}_t) - \eta \left\langle \nabla F(\boldsymbol{\theta}_t), \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} \boldsymbol{g}_{t,j}^{(i)} \right\rangle + \frac{L\eta^2}{2} \left\| \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} \boldsymbol{g}_{t,j}^{(i)} \right\|^2. \tag{A.7}$$

From the following identity:

$$\boldsymbol{g}_{t,j}^{(i)} = \boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t) + \nabla F(\boldsymbol{\theta}_t), \tag{A.8}$$

we write equation A.7 as:

$$\begin{aligned} F(\boldsymbol{\theta}_{t+1}) \leq F(\boldsymbol{\theta}_t) - \eta \sum_{i=1}^{n} p_i K_i \|\nabla F(\boldsymbol{\theta}_t)\|^2 - \eta \left\langle \nabla F(\boldsymbol{\theta}_t), \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t)) \right\rangle + \\ + \frac{L\eta^2}{2} \left\| \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t)) + \sum_{i=1}^{n} p_i K_i \nabla F(\boldsymbol{\theta}_t) \right\|^2. \end{aligned} \tag{A.9}$$

By further expanding the last term we have:

$$\begin{aligned} F(\boldsymbol{\theta}_{t+1}) \leq F(\boldsymbol{\theta}_t) - \eta \sum_{i=1}^{n} p_i K_i \, \|\nabla F(\boldsymbol{\theta}_t)\|^2 - \eta \left\langle \nabla F(\boldsymbol{\theta}_t), \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t)) \right\rangle + \\ + \frac{L\eta^2}{2} \left\| \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t)) \right\|^2 + \frac{L\eta^2}{2} \left(\sum_{i=1}^{n} p_i K_i \right)^2 \|\nabla F(\boldsymbol{\theta}_t)\|^2 + \\ + L\eta^2 \left\langle \sum_{i=1}^{n} p_i K_i \nabla F(\boldsymbol{\theta}_t), \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t)) \right\rangle. \end{aligned} \tag{A.10}$$

For simplicity we use μ as a shorthand for $\sum_{i=1}^{n} p_i K_i$. Grouping similar terms together gives:

$$\begin{aligned} F(\boldsymbol{\theta}_{t+1}) \leq F(\boldsymbol{\theta}_t) - \eta\mu \left(1 - \frac{L\eta}{2}\mu\right) \|\nabla F(\boldsymbol{\theta}_t)\|^2 \\ - \eta(1 - L\eta\mu) \left\langle \nabla F(\boldsymbol{\theta}_t), \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t)) \right\rangle + \frac{L\eta^2}{2} \left\| \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t)) \right\|^2. \end{aligned} \tag{A.11}$$

Taking the expectation on both sides and with Cauchy–Schwarz inequality, we have:

$$\begin{aligned} \mathbb{E}F(\boldsymbol{\theta}_{t+1}) &\leq \mathbb{E}F(\boldsymbol{\theta}_t) - \eta\mu \left(1 - \frac{L\eta}{2}\mu\right) \mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 + \\ &\quad + \eta(1 - L\eta\mu)\mathbb{E} \left[\|\nabla F(\boldsymbol{\theta}_t)\| \cdot \left\| \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla F(\boldsymbol{\theta}_t)) \right\| \right] \\ &\quad + \frac{L\eta^2}{2} \mathbb{E} \left\| \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t)) \right\|^2 \\ &\leq \mathbb{E}F(\boldsymbol{\theta}_t) + \left(-\eta\mu \left(1 - \frac{L\eta}{2}\mu\right) + \frac{1}{2}\eta(1 - L\eta\mu) \right) \mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 + \\ &\quad + \left(\frac{L\eta^2}{2} + \frac{1}{2}\eta(1 - L\eta\mu) \right) \mathbb{E} \left\| \sum_{i=1}^{n} p_i \sum_{j=1}^{K_i} (\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t)) \right\|^2, \end{aligned} \tag{A.12}$$

where we used the inequality $ab \leq \frac{1}{2}(a^2 + b^2)$ in the second line. Let us now study the two coefficients separately. From the assumption, $6\eta L K_i \leq 1$ for any $i \in [n]$, and thus $6L\mu\eta \leq 1$. Hence we have:

$$\begin{aligned}
-\eta\mu\left(1 - \frac{L\eta}{2}\mu\right) + \frac{1}{2}\eta(1 - L\eta\mu) &\leq -\eta\left(\mu - \frac{1}{2} - \frac{L\eta}{2}\mu^2 + \frac{1}{2}L\eta\mu\right) \\
&\leq -\eta\left(\frac{11\mu - 6}{12} + \frac{L\eta\mu}{2}\right) \\
&\leq -\eta\frac{11\mu - 6}{12}. \qquad \text{(A.13)}
\end{aligned}$$

Moreover, the coefficient of the third term can be upper bounded as:

$$\frac{L\eta^2}{2} + \frac{1}{2}\eta(1 - L\eta\mu) = \frac{\eta}{2} + \frac{L}{2}\eta^2(1 - \mu) \leq \frac{\eta}{2}. \qquad \text{(A.14)}$$

Therefore, equation A.12 becomes:

$$\mathbb{E}F(\boldsymbol{\theta}_{t+1}) \leq \mathbb{E}F(\boldsymbol{\theta}_t) - \eta\frac{11\mu - 6}{12}\mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 + \frac{\eta}{2}\mathbb{E}\left\|\sum_{i=1}^{n} p_i \sum_{j=1}^{K_i}(\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t))\right\|^2. \qquad \text{(A.15)}$$

Part II With the following identity:

$$\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t) = \boldsymbol{g}_{t,j}^{(i)} - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) + \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla F(\boldsymbol{\theta}_t), \qquad \text{(A.16)}$$

the last term of equation A.12 can be simplified as:

$$\mathbb{E}\left\|\sum_{i=1}^{n} p_i \sum_{j=1}^{K_i}(\boldsymbol{g}_{t,j}^{(i)} - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}))\right\|^2 + \mathbb{E}\left\|\sum_{i=1}^{n} p_i \sum_{j=1}^{K_i}(\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla F(\boldsymbol{\theta}_t))\right\|^2, \qquad \text{(A.17)}$$

where we note that $\boldsymbol{g}_{t,j}^{(i)}$ is an unbiased estimator of $\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})$. We first bound the first term of equation A.17:

$$\begin{aligned}
\mathbb{E}\left\|\sum_{i=1}^{n} p_i \sum_{j=1}^{K_i}(\boldsymbol{g}_{t,j}^{(i)} - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}))\right\|^2 &\leq \mathbb{E}\left(\sum_{i=1}^{n} p_i \sum_{j=1}^{K_i}\left\|\boldsymbol{g}_{t,j}^{(i)} - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\right\|\right)^2 \\
&\leq \mathbb{E}\left(\sum_{i=1}^{n} p_i^2\right)\sum_{i=1}^{n}\left(\sum_{j=1}^{K_i}\left\|\boldsymbol{g}_{t,j}^{(i)} - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\right\|\right)^2 \\
&\leq \mathbb{E}\left(\sum_{i=1}^{n} p_i^2\right)\sum_{i=1}^{n} K_i \sum_{j=1}^{K_i}\left\|\boldsymbol{g}_{t,j}^{(i)} - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\right\|^2 \\
&= \left(\sum_{i=1}^{n} p_i^2\right)\sum_{i=1}^{n} K_i \sum_{j=1}^{K_i}\mathbb{E}\left\|\boldsymbol{g}_{t,j}^{(i)} - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\right\|^2 \\
&\leq \left(\sum_{i=1}^{n} p_i^2\right)\sum_{i=1}^{n} K_i^2 \frac{\sigma_i^2}{m}, \qquad \text{(A.18)}
\end{aligned}$$

where in the first line, we used triangle inequality; in the second and third lines, we used the Cauchy–Schwarz inequality; in the fourth line we used the linearity of expectation; in the final line we used Assumption 4.1

and the batch size $|S| = m$, so that:

$$\begin{aligned}\mathbb{E}\left\|\boldsymbol{g}_{t,j}^{(i)} - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\right\|^2 &= \mathbb{E}_{S\sim\mathcal{D}_i^m}\left\|\frac{1}{|S|}\sum_{(\mathbf{x},y)\in S}\nabla\ell(\boldsymbol{\theta}_{t,j-1}^{(i)},(\mathbf{x},y)) - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\right\|^2 \\ &= \frac{1}{|S|^2}\sum_{(\mathbf{x},y)\in S}\mathbb{E}_{(\mathbf{x},y))\sim\mathcal{D}_i}\left\|\nabla\ell(\boldsymbol{\theta}_{t,j-1}^{(i)},(\mathbf{x},y)) - \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\right\|^2 \\ &\leq \frac{\sigma_i^2}{|S|}. \end{aligned} \tag{A.19}$$

Note that in the second line we used the property that each $(\mathbf{x}, y)$ is an i.i.d. sample from $\mathcal{D}_i$. Similarly we bound the second term of equation A.17:

$$\mathbb{E}\left\|\sum_{i=1}^n p_i\sum_{j=1}^{K_i}(\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla F(\boldsymbol{\theta}_t))\right\|^2 \leq \left(\sum_{i=1}^n p_i^2\right)\sum_{i=1}^n K_i\sum_{j=1}^{K_i}\mathbb{E}\left\|\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla F(\boldsymbol{\theta}_t)\right\|^2. \tag{A.20}$$

With the following identity:

$$\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla F(\boldsymbol{\theta}_t) = \nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla f_i(\boldsymbol{\theta}_t) + \nabla f_i(\boldsymbol{\theta}_t) - \nabla F(\boldsymbol{\theta}_t), \tag{A.21}$$

and taking the squared norm on both sides, we have:

$$\begin{aligned}\|\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla F(\boldsymbol{\theta}_t)\|^2 &\leq 2\|\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla f_i(\boldsymbol{\theta}_t)\|^2 + 2\|\nabla f_i(\boldsymbol{\theta}_t) - \nabla F(\boldsymbol{\theta}_t)\|^2 \\ &\leq 2L^2\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t\|^2 + 2\sigma^2, \end{aligned} \tag{A.22}$$

where we note that:

$$\begin{aligned}\|\nabla f_i(\boldsymbol{\theta}_t) - \nabla F(\boldsymbol{\theta}_t)\| &= \left\|\nabla f_i(\boldsymbol{\theta}_t) - \sum_{j=1}^n p_j\nabla f_j(\boldsymbol{\theta}_t)\right\| \\ &= \left\|\sum_{j=1}^n p_j(\nabla f_i(\boldsymbol{\theta}_t) - \nabla f_j(\boldsymbol{\theta}_t))\right\| \\ &\leq \sum_{j=1}^n p_j\|\nabla f_i(\boldsymbol{\theta}_t) - \nabla f_j(\boldsymbol{\theta}_t)\| \\ &\leq \sigma, \end{aligned} \tag{A.23}$$

where in the third line we used the triangle inequality and in the last line we used Assumption 4.1. Plugging equation A.22 into equation A.20 yields:

$$\begin{aligned}\mathbb{E}\left\|\sum_{i=1}^n p_i\sum_{j=1}^{K_i}(\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla F(\boldsymbol{\theta}_t))\right\|^2 &\leq 2\left(\sum_{i=1}^n p_i^2\right)\sum_{i=1}^n K_i^2\sigma^2 + \\ &\quad + 2L^2\left(\sum_{i=1}^n p_i^2\right)\sum_{i=1}^n K_i\sum_{j=1}^{K_i}\mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t\|^2. \end{aligned} \tag{A.24}$$

Bringing equation A.18 and equation A.24 into equation A.17 we write:

$$\begin{aligned}\mathbb{E}\left\|\sum_{i=1}^{n} p_i \sum_{j=1}^{K_i}(\boldsymbol{g}_{t,j}^{(i)} - \nabla F(\boldsymbol{\theta}_t))\right\|^2 &\leq \left(\sum_{i=1}^{n} p_i^2\right)\sum_{i=1}^{n} K_i^2\left(\frac{\sigma_i^2}{m} + 2\sigma^2\right) + \\ &+ 2L^2\left(\sum_{i=1}^{n} p_i^2\right)\sum_{i=1}^{n} K_i \sum_{j=1}^{K_i}\mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t\|^2 \\ &\leq \left(\sum_{i=1}^{n} p_i^2\right)\sum_{i=1}^{n} K_i^2\left(\frac{\sigma_i^2}{m} + 2\sigma^2\right) + \\ &+ 2L^2\left(\sum_{i=1}^{n} p_i^2\right)\sum_{i=1}^{n} K_i \sum_{j=0}^{K_i-1}\mathbb{E}\|\boldsymbol{\theta}_{t,j}^{(i)} - \boldsymbol{\theta}_t\|^2. \end{aligned} \tag{A.25}$$

Part III Now let us give an upper bound for $\mathbb{E}\|\boldsymbol{\theta}_{t,j}^{(i)} - \boldsymbol{\theta}_t\|^2$. From equation A.25, we only need to focus on $K_i \geq 2$ and $j = 1, \ldots, K_i - 1$ since $\boldsymbol{\theta}_{t,0}^{(i)} = \boldsymbol{\theta}_t$. For $j \in [K_i - 1]$, we have from equation A.4:

$$\begin{aligned}\mathbb{E}\|\boldsymbol{\theta}_{t,j}^{(i)} - \boldsymbol{\theta}_t\|^2 &= \mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t - \eta\boldsymbol{g}_{t,j}^{(i)}\|^2 \\ &= \mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t - \eta\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) + \eta\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \eta\boldsymbol{g}_{t,j}^{(i)}\|^2 \\ &= \mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t - \eta\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\|^2 + \mathbb{E}\eta^2\|\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \boldsymbol{g}_{t,j}^{(i)}\|^2 \\ &= \mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t - \eta\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\|^2 + \eta^2\frac{\sigma_i^2}{m}, \end{aligned} \tag{A.26}$$

where in the third line we note that $\boldsymbol{g}_{t,j}^{(i)}$ is an unbiased estimator of $\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})$ and in the last line we used equation A.19. The first term in the last line above can be bounded as:

$$\mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t - \eta\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\|^2 \leq \left(1 + \frac{1}{2K_i - 1}\right)\mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t\|^2 + 2K_i\eta^2\|\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\|^2, \tag{A.27}$$

where we used $\|a+b\|^2 \leq (1+\frac{1}{\alpha})\|a\|^2 + (1+\alpha)\|b\|^2$ for any vectors a, b with the same dimension and $\alpha > 0$. Since

$$\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) = (\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla f_i(\boldsymbol{\theta}_t)) + (\nabla f_i(\boldsymbol{\theta}_t) - \nabla F(\boldsymbol{\theta}_t)) + \nabla F(\boldsymbol{\theta}_t), \tag{A.28}$$

taking the squared norm on both sides we have (note that $(a+b+c)^2 \leq 3(a^2+b^2+c^2)$):

$$\begin{aligned}\|\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\|^2 &\leq 3\|\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)}) - \nabla f_i(\boldsymbol{\theta}_t)\|^2 + 3\|\nabla f_i(\boldsymbol{\theta}_t) - \nabla F(\boldsymbol{\theta}_t)\|^2 + 3\|\nabla F(\boldsymbol{\theta}_t)\|^2 \\ &\leq 3L^2\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t\|^2 + 3\sigma^2 + 3\|\nabla F(\boldsymbol{\theta}_t)\|^2, \end{aligned} \tag{A.29}$$

where in the second line we used equation A.23 and Assumption 4.1. Plugging equation A.29 into equation A.27 we find:

$$\mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t - \eta\nabla f_i(\boldsymbol{\theta}_{t,j-1}^{(i)})\|^2 \leq \left(1 + \frac{1}{2K_i - 1} + 6K_i\eta^2L^2\right)\mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t\|^2 + 6K_i\eta^2\sigma^2 + 6K_i\eta^2\|\nabla F(\boldsymbol{\theta}_t)\|^2. \tag{A.30}$$

Combined with equation A.26, we obtain:

$$\mathbb{E}\|\boldsymbol{\theta}_{t,j}^{(i)} - \boldsymbol{\theta}_t\|^2 \leq \left(1 + \frac{1}{2K_i - 1} + 6K_i\eta^2L^2\right)\mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t\|^2 + \eta^2\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2\right) + 6K_i\eta^2\|\nabla F(\boldsymbol{\theta}_t)\|^2. \tag{A.31}$$

Recall that we assumed $\eta \leq \min_i\{\frac{1}{6K_iL}\}$. For $K_i \geq 2$ (note the assumption at the beginning of Part III), we have:

$$1 + \frac{1}{2K_i - 1} + 6K_i\eta^2L^2 \leq 1 + \frac{1}{2K_i - 1} + \frac{1}{6K_i} \leq 1 + \frac{1}{K_i}. \tag{A.32}$$

Therefore, equation A.31 becomes:

$$\mathbb{E}\|\boldsymbol{\theta}_{t,j}^{(i)} - \boldsymbol{\theta}_t\|^2 \leq \left(1 + \frac{1}{K_i}\right)\mathbb{E}\|\boldsymbol{\theta}_{t,j-1}^{(i)} - \boldsymbol{\theta}_t\|^2 + \eta^2\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2\right) + 6K_i\eta^2\|\nabla F(\boldsymbol{\theta}_t)\|^2. \tag{A.33}$$

We can treat $\{a_j = \mathbb{E}\|\boldsymbol{\theta}_{t,j}^{(i)} - \boldsymbol{\theta}_t\|^2\|\}_{j=1}^{K_i-1}$ as a sequence. Unrolling this sequence and with $a_0 = 0$, we have:

$$\begin{aligned}\mathbb{E}\|\boldsymbol{\theta}_{t,j}^{(i)} - \boldsymbol{\theta}_t\|^2 &\leq \frac{\left(1 + \frac{1}{K_i}\right)^j - 1}{1 + \frac{1}{K_i} - 1}\left(\eta^2\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2\right) + 6K_i\eta^2\|\nabla F(\boldsymbol{\theta}_t)\|^2\right) \\ &= K_i\left(\left(1 + \frac{1}{K_i}\right)^j - 1\right)\left(\eta^2\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2\right) + 6K_i\eta^2\|\nabla F(\boldsymbol{\theta}_t)\|^2\right).\end{aligned} \tag{A.34}$$

Summing over $j = 0, 1, \ldots, K_i - 1$ gives:

$$\begin{aligned}\sum_{j=0}^{K_i-1}\mathbb{E}\|\boldsymbol{\theta}_{t,j}^{(i)} - \boldsymbol{\theta}_t\|^2 &\leq K_i^2\left(\left(1 + \frac{1}{K_i}\right)^{K_i} - 2\right)\left(\eta^2\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2\right) + 6K_i\eta^2\|\nabla F(\boldsymbol{\theta}_t)\|^2\right) \\ &\leq (e-2)K_i^2\left(\eta^2\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2\right) + 6K_i\eta^2\|\nabla F(\boldsymbol{\theta}_t)\|^2\right) \\ &= (e-2)K_i^2\eta^2\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2 + 6K_i\|\nabla F(\boldsymbol{\theta}_t)\|^2\right)\end{aligned} \tag{A.35}$$

where in the first line we used the geometric series formula $1 + q + \cdots + q^{n-1} = \frac{q^n-1}{q-1}$; in the second line we used the fact that $\left(1 + \frac{1}{K_i}\right)^{K_i} \leq e$ for $K_i \geq 1$, with e the natural logarithm.

Part IV We finally put things together and finish our proof. Combining equation A.15, equation A.25 and equation A.35 we have:

$$\mathbb{E}F(\boldsymbol{\theta}_{t+1}) \leq \mathbb{E}F(\boldsymbol{\theta}_t) - \eta\frac{11\mu - 6}{12}\mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 + 6(e-2)\eta^3L^2\left(\sum_{i=1}^n p_i^2\right)\sum_{i=1}^n K_i^4\|\nabla F(\boldsymbol{\theta}_t)\|^2 + \Psi_\sigma, \tag{A.36}$$

where we denote

$$\Psi_\sigma = \frac{\eta}{2}\left(\sum_{i=1}^n p_i^2\right)\left[\sum_{i=1}^n K_i^2\left(\frac{\sigma_i^2}{m} + 2\sigma^2\right) + 2(e-2)\eta^2L^2\sum_{i=1}^n K_i^3\left(\frac{\sigma_i^2}{m} + 6K_i\sigma^2\right)\right]. \tag{A.37}$$

Since we assumed:

$$\eta \leq \frac{1}{L}\sqrt{\frac{1}{24(e-2)(\sum_{i=1}^n p_i^2)(\sum_{i=1}^n K_i^4)}}, \tag{A.38}$$

we have:

$$\frac{11\mu - 6}{12} - 6(e-2)L^2\left(\sum_{i=1}^n p_i^2\right)\sum_{i=1}^n K_i^4\eta^2 \geq \frac{11\mu - 6}{12} - \frac{1}{4} \geq \frac{11\mu - 9}{12}. \tag{A.39}$$

Therefore, equation A.36 becomes:

$$\mathbb{E}F(\boldsymbol{\theta}_{t+1}) \leq \mathbb{E}F(\boldsymbol{\theta}_t) - \eta\frac{11\mu - 9}{12}\mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 + \Psi_\sigma. \tag{A.40}$$

With some algebra we obtain:

$$\eta \frac{11\mu - 9}{12} \mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 \leq \mathbb{E}[F(\boldsymbol{\theta}_t) - F(\boldsymbol{\theta}_{t+1})] + \Psi_\sigma. \tag{A.41}$$

Summing both sides over $t = 0, \ldots, T-1$ and dividing by T, we have:

$$\eta \frac{11\mu - 9}{12} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 \leq \frac{F(\boldsymbol{\theta}_0) - F^*}{T} + \Psi_\sigma, \tag{A.42}$$

which gives:

$$\min_{0 \leq t \leq T-1} \mathbb{E}\|\nabla F(\boldsymbol{\theta}_t)\|^2 \leq \frac{12}{(11\mu - 9)\eta} \left(\frac{F(\boldsymbol{\theta}_0) - F^*}{T} + \Psi_\sigma \right), \tag{A.43}$$

with $F^* = \min_{\boldsymbol{\theta}} F(\boldsymbol{\theta})$ the optimal value.

Finally, for the partial participation, it suffices to replace the client set $\{1, \ldots, n\}$ with its subset. Note that after this substitution, the new variance term satisfies $\Psi'_\sigma \leq \Psi_\sigma$ since this term increases with more participants, and equation A.39 still holds since we subtract a smaller term with partial participation. We also need to modify equation A.39 so we further lower bound equation A.39 with $\mu \geq \min_i K_i$. □

Theorem 4.4 (**PropFair**). *Denote $\widetilde{L} = \frac{4}{M^2}(\frac{3}{2}ML + L_0^2)$ and $p_i = \frac{n_i}{N}$. Given Assumptions 4.1 and 4.3, assume that the local learning rate satisfies:*

$$\eta \leq \min \left\{ \min_{i \in [n]} \frac{1}{6\widetilde{L}K_i}, \frac{1}{L} \sqrt{\frac{1}{24(e-2)(\sum_i p_i^2)(\sum_i K_i^4)}} \right\}. \tag{4.3}$$

By running Algorithm 1 for T global epochs we have:

$$\min_{0 \leq t \leq T-1} \mathbb{E}\|\nabla \pi(\boldsymbol{\theta}_t)\|^2 \leq \frac{12}{(11\mu - 9)\eta} \left(\frac{\pi_0 - \pi^*}{T} + \widetilde{\Psi}_\sigma \right),$$

with $\mu = \sum_i p_i K_i$ for full participation and $\mu = \min_i K_i$ for partial participation, $\pi_0 = \pi(\boldsymbol{\theta}_0)$, $\pi^* = \min_{\boldsymbol{\theta}} \pi(\boldsymbol{\theta})$ the optimal value, and

$$\widetilde{\Psi}_\sigma = \frac{\eta}{2} \Big(\sum_{i=1}^n p_i^2 \Big) \left[\sum_{i=1}^n K_i^2 \left(\frac{\widetilde{\sigma}_i^2}{m} + 2\widetilde{\sigma}^2 \right) + 2(e-2)\eta^2 L^2 \sum_{i=1}^n K_i^3 \left(\frac{\widetilde{\sigma}_i^2}{m} + 6K_i \widetilde{\sigma}^2 \right) \right],$$

where $\widetilde{\sigma}_i^2 = \frac{8}{M^4}(9M^2\sigma_i^2 + 4L_0^2\sigma_{0,i}^2)$ and $\widetilde{\sigma} = \frac{4}{M}\left(\frac{3}{2}\sigma + \frac{L_0}{M}\sigma_0\right)$.

Proof. From the Lipschitzness we can obtain an upper bound for the gradient: $\|\nabla f_i(\boldsymbol{\theta})\| \leq L_0$ for any $\theta \in \mathbb{R}^d$. It suffices to determine the constants in Assumptions 4.1 after adding the composition $\varphi \circ f_i$, with $\varphi(t) = -\log(M - t)$. For Assumption 4.1, we write:

$$\begin{aligned}
\|\nabla(\varphi \circ f_i)(\boldsymbol{\theta}) - \nabla(\varphi \circ f_i)(\boldsymbol{\theta}')\| &= \left\| \frac{\nabla f_i(\boldsymbol{\theta})}{M - f_i(\boldsymbol{\theta})} - \frac{\nabla f_i(\boldsymbol{\theta}')}{M - f_i(\boldsymbol{\theta}')} \right\| \\
&= \left\| \frac{M(\nabla f_i(\boldsymbol{\theta}) - \nabla f_i(\boldsymbol{\theta}')) - \nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}') + \nabla f_i(\boldsymbol{\theta}') f_i(\boldsymbol{\theta})}{(M - f_i(\boldsymbol{\theta}))(M - f_i(\boldsymbol{\theta}'))} \right\| \\
&\leq \frac{4}{M^2} \left(M\|\nabla f_i(\boldsymbol{\theta}) - \nabla f_i(\boldsymbol{\theta}')\| + \|\nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}') - \nabla f_i(\boldsymbol{\theta}') f_i(\boldsymbol{\theta}))\| \right) \\
&\leq \frac{4}{M^2} (ML\|\boldsymbol{\theta} - \boldsymbol{\theta}'\| + \|\nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}') - \nabla f_i(\boldsymbol{\theta}') f_i(\boldsymbol{\theta}))\|).
\end{aligned} \tag{A.44}$$

The second term in the parenthesis above can be computed as:

$$\begin{aligned}
\|\nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}') - \nabla f_i(\boldsymbol{\theta}') f_i(\boldsymbol{\theta}))\| &= \|\nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}') - \nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}) + \nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}) - \nabla f_i(\boldsymbol{\theta}') f_i(\boldsymbol{\theta}))\| \\
&\leq \|\nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}') - \nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta})\| + \|\nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}) - \nabla f_i(\boldsymbol{\theta}') f_i(\boldsymbol{\theta}))\| \\
&= \|\nabla f_i(\boldsymbol{\theta})\| \cdot \|f_i(\boldsymbol{\theta}') - f_i(\boldsymbol{\theta})\| + \|\nabla f_i(\boldsymbol{\theta}) - \nabla f_i(\boldsymbol{\theta}')\| \cdot \|f_i(\boldsymbol{\theta})\| \\
&\leq L_0^2 \|\boldsymbol{\theta}' - \boldsymbol{\theta}\| + L\frac{M}{2}\|\boldsymbol{\theta}' - \boldsymbol{\theta}\|,
\end{aligned} \tag{A.45}$$

where in the second line we used triangle inequality; in the fourth line we used Assumptions 4.1 and 4.3. Plugging in back to equation A.44 we have:

$$\|\nabla(\varphi \circ f_i)(\boldsymbol{\theta}) - \nabla(\varphi \circ f_i)(\boldsymbol{\theta}')\| \leq \frac{4}{M^2}\left(\frac{3}{2}ML + L_0^2\right)\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|. \tag{A.46}$$

Let us now figure out the variance terms. For the global variance term, we similarly write:

$$\begin{aligned}
\|\nabla(\varphi \circ f_i)(\boldsymbol{\theta}) - \nabla(\varphi \circ f_j)(\boldsymbol{\theta})\| &= \left\|\frac{\nabla f_i(\boldsymbol{\theta})}{M - f_i(\boldsymbol{\theta})} - \frac{\nabla f_j(\boldsymbol{\theta})}{M - f_j(\boldsymbol{\theta})}\right\| \\
&= \left\|\frac{M(\nabla f_i(\boldsymbol{\theta}) - \nabla f_j(\boldsymbol{\theta})) - \nabla f_i(\boldsymbol{\theta}) f_j(\boldsymbol{\theta}) + \nabla f_j(\boldsymbol{\theta}) f_i(\boldsymbol{\theta})}{(M - f_i(\boldsymbol{\theta}))(M - f_j(\boldsymbol{\theta}))}\right\| \\
&\leq \frac{4}{M^2}\left(M\|\nabla f_i(\boldsymbol{\theta}) - \nabla f_j(\boldsymbol{\theta})\| + \|\nabla f_i(\boldsymbol{\theta}) f_j(\boldsymbol{\theta}) - \nabla f_j(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}))\|\right) \\
&\leq \frac{4}{M^2}(M\sigma + \|\nabla f_i(\boldsymbol{\theta}) f_j(\boldsymbol{\theta}) - \nabla f_j(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}))\|).
\end{aligned} \tag{A.47}$$

The second term in the parenthesis above can be computed as:

$$\begin{aligned}
\|\nabla f_i(\boldsymbol{\theta}) f_j(\boldsymbol{\theta}) - \nabla f_j(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}))\| &= \|\nabla f_i(\boldsymbol{\theta}) f_j(\boldsymbol{\theta}) - \nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}) + \nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}) - \nabla f_j(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}))\| \\
&\leq \|\nabla f_i(\boldsymbol{\theta}) f_j(\boldsymbol{\theta}) - \nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta})\| + \|\nabla f_i(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}) - \nabla f_j(\boldsymbol{\theta}) f_i(\boldsymbol{\theta}))\| \\
&= \|\nabla f_i(\boldsymbol{\theta})\| \cdot \|f_j(\boldsymbol{\theta}) - f_i(\boldsymbol{\theta})\| + \|\nabla f_i(\boldsymbol{\theta}) - \nabla f_j(\boldsymbol{\theta})\| \cdot \|f_i(\boldsymbol{\theta})\| \\
&\leq L_0\sigma_0 + \frac{M}{2}\sigma,
\end{aligned} \tag{A.48}$$

where in the second and fourth lines we used triangle inequality; in the last line we used Assumptions 4.1 and 4.3. Plugging equation A.48 into equation A.47 we find:

$$\|\nabla(\varphi \circ f_i)(\boldsymbol{\theta}) - \nabla(\varphi \circ f_j)(\boldsymbol{\theta})\|^2 \leq \left(\frac{4}{M}\left(\frac{3}{2}\sigma + \frac{L_0}{M}\sigma_0\right)\right)^2. \tag{A.49}$$

Let us finally compute the new local variance term. For simplicity we denote $\ell_{(\mathbf{x},y)}(\theta) := \ell(\boldsymbol{\theta}, (\mathbf{x}, y))$.

$$\begin{aligned}
\|\nabla(\varphi \circ f_i)(\boldsymbol{\theta}) - \nabla(\varphi \circ \ell_{(\mathbf{x},y)})(\boldsymbol{\theta})\| &= \left\|\frac{\nabla f_i(\boldsymbol{\theta})}{M - f_i(\boldsymbol{\theta})} - \frac{\nabla \ell_{(\mathbf{x},y)}(\boldsymbol{\theta})}{M - \ell_{(\mathbf{x},y)}(\boldsymbol{\theta})}\right\| \\
&\leq \frac{4}{M^2}\left(\frac{3M}{2}\|\nabla f_i(\boldsymbol{\theta}) - \nabla \ell_{(\mathbf{x},y)}(\boldsymbol{\theta})\| + L_0\|f_i(\boldsymbol{\theta}) - \ell_{(\mathbf{x},y)}(\boldsymbol{\theta})\|\right),
\end{aligned} \tag{A.50}$$

and the derivation follows similarly as equation A.47. Taking the square on both sides and taking the expectation over $(\mathbf{x}, y) \sim \mathcal{D}_i$, we obtain:

$$\begin{aligned}
\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i}\|\nabla(\varphi \circ f_i)(\boldsymbol{\theta}) - \nabla(\varphi \circ \ell_{(\mathbf{x},y)})(\boldsymbol{\theta})\|^2 &\leq \frac{16}{M^4}\Bigg(2\frac{9M^2}{4}\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i}\|\nabla f_i(\boldsymbol{\theta}) - \nabla \ell_{(\mathbf{x},y)}(\boldsymbol{\theta})\|^2 + \\
&\quad + 2L_0^2\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i}\|f_i(\boldsymbol{\theta}) - \ell_{(\mathbf{x},y)}(\boldsymbol{\theta})\|^2\Bigg) \\
&\leq \frac{8}{M^4}(9M^2\sigma_i^2 + 4L_0^2\sigma_{0,i}^2),
\end{aligned} \tag{A.51}$$

where in the first line we used $(a + b)^2 \leq 2(a^2 + b^2)$ and in the second line we used Assumptions 4.1 and 4.3. □

B A Failure Case of Agnostic Federated Learning

In this section we show that AFL might suffer from the generalization issue, in the case when some of the clients have very few samples that are outliers. Suppose the input space is $\mathbb{R}^2$ and the classification task

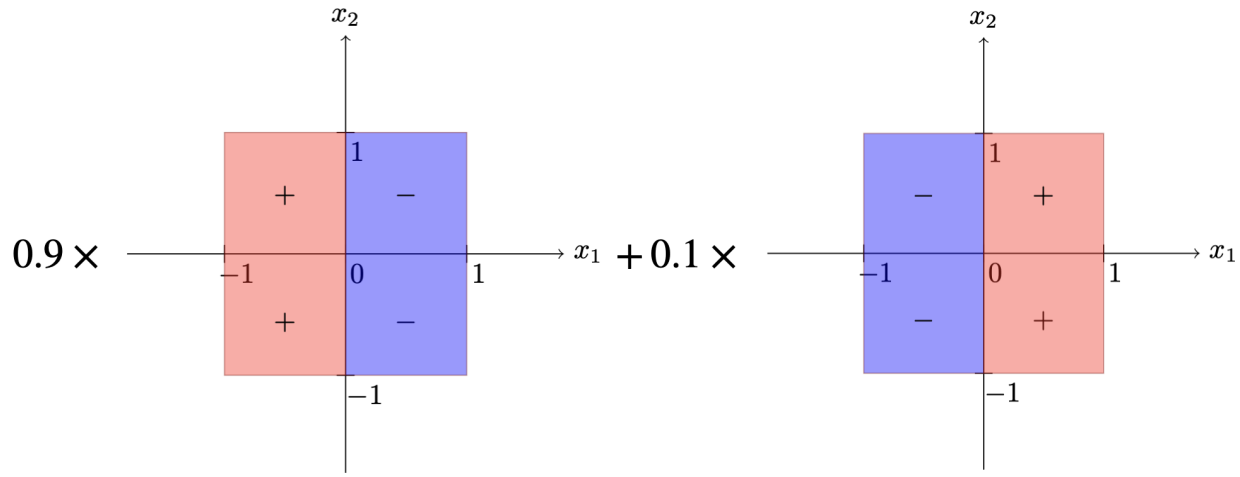

Figure 5: The visualization of the distribution shown in equation B.1. The plus sign means the positive label $y = 1$ and the negative sign means the negative label $y = -1$.

is binary with a linear classifier. We assume the simple case where every client has the same underlying distribution:

$$p(x|y) = \begin{cases} 0.9U([-1,0] \times [-1,1]) + 0.1U([0,1] \times [-1,1]) & \text{if } y = 1, \\ 0.9U([0,1] \times [-1,1]) + 0.1U([-1,0] \times [-1,1]) & \text{if } y = -1. \end{cases} \tag{B.1}$$

Note that $U(I)$ represents the density of the uniform distribution on interval I. A visualization of equation B.1 can be found in Figure 5.

In practice, we draw samples from each of the client. However, if one of the clients do not have enough samples, AFL might have an issue. For instance, two clients could have to opposite sample sets:

$$S_1 = \{((0.5, \pm 0.5), 1), ((-0.5, \pm 0.5), -1)\},\ S_2 = \{((0.5, \pm 0.5), -1), ((-0.5, \pm 0.5), 1)\}.$$

In this case, AFL could give unfavorable generalization error, since the algorithm gives minimal training error to be 50%. For example, this optimal AFL solution can be reached if one chooses the linear classifier to be perpendicular to the x_1-axis, resulting in the test error to be 50%. However, there exists an optimal classifier $\boldsymbol{w} = (-1, 0)$ such that the test error is 10%.

We can also verify this claim from the proof of Theorem 1 in Appendix C.2, Mohri et al. (2019). If one of the client has too few samples (i.e., some m_k is small), then the generalization bound on the right can be very large or even vacuous.

C Additional Experiments

In this section, we provide more details about our experimental results. Results for all experiments are provided based on an average over three runs with different seeds.

C.1 Datasets and models

We describe the benchmark datasets in this subsection. For all datasets we fix the batch size to be 64.

CIFAR-10/100 (Krizhevsky et al., 2009) are standard image classification datasets. There are 50000 samples with 10/100 balanced classes for CIFAR-10/100. By doing Dirichlet allocation (Wang et al., 2019a) we achieve the heterogeneity of label distributions. For all samples in each class k, denoted as the set $\mathcal{S}_k$,

we split $\mathcal{S}_k = \mathcal{S}_{k,1} \cup \mathcal{S}_{k,2} \ldots \mathcal{S}_{k,n}$ into n clients according a symmetric Dirichlet distribution $\mathrm{Dir}(\beta)$. Then we gather the samples for client j as**d** $\mathcal{S}_{1,j} \cup \mathcal{S}_{2,j} \ldots \mathcal{S}_{C,j}$ if we have C classes in total. We note that some of the clients might have too few samples (a few hundred). In this case the FL algorithm might overfit for such client and we regenerate the data split. We choose the number of clients to be 10 for both CIFAR-10/100. For each of the client dataset, we split it further into 80% training data and 20% test data.

TinyImageNet is from the course project of Stanford CS231N.[2] It contains 200 classes and each class has 500 images. Our FL split setting is the same as CIFAR-10/100, except that we choose 20 clients and the Dirichlet parameter $\beta = 0.05$.

Shakespeare (Shakespeare, 1614; McMahan et al., 2017) is a text dataset of Shakespeare dialogues, and we use it for the task of next character prediction. We treat each speaking role as a client resulting in a natural heterogeneous partition. We first filter out the clients with less than 10,000 samples and sample 20 clients from the remaining. Also, each client's dataset is split into 50% for training and 50% for test.

In Table 2, we summarize these datasets, our partition methods, as well as the models we implement.

Table 2: Details of the experiments and the used datasets. ResNet-18 is the residual neural network defined in He et al. (2016). GN: Group Normalization (Wu & He, 2018); FC: fully connected layer; CNN: Convolutional Neural Network; Conv: convolution layer; RNN: Recurrent Neural Network; LSTM: Long Short-Term Memory layer. The plus sign means composition.

Datasets	**Training set size**	**Test set size**	**Partition method**	**# of clients**	**Model**
CIFAR-10	39963	10037	Dirichlet partition ($\beta = 0.5$)	10	ResNet-18 + GN
CIFAR-100	39764	10236	Dirichlet partition ($\beta = 0.1$)	10	ResNet-18 + GN
TinyImageNet	78044	20135	Dirichlet partition ($\beta = 0.05$)	20	ResNet-18 + GN
Shakespeare	178796	177231	realistic partition	20	RNN (1 LSTM + 1 FC)

C.2 Algorithms to compare and tuning hyperparameters

We compare our PropFair algorithm with common FL baselines, including FedAvg (McMahan et al., 2017), q-FFL (Li et al., 2020c) and AFL (Mohri et al., 2019). For each dataset and each algorithm (algorithms with different hyperparameters are counted as different), we find the best learning rate from a grid. Here are the grids we used for each dataset:

- CIFAR-10: `{5e-3, 1e-2, 2e-2, 5e-2}`;
- CIFAR-100: `{5e-3, 1e-2, 2e-2, 5e-2}`;
- TinyImageNet: `{5e-3, 1e-2, 2e-2, 5e-2}`;
- Shakespeare: `{1e-1, 5e-1, 1, 2}`;

Table 3: The best values of hyperparameters used for different datasets, chosen based on grid search.

Algorithm	**Hyperparameter**	**CIFAR-10**	**CIFAR-100**	**TinyImageNet**	**Shakespeare**
q-FFL	q	0.1	0.1	0.1	0.1
TERM	α	0.5	0.5	0.5	0.5
GIFAIR-FL	$\lambda/\lambda_{\max}$	0.9	0.1	0.1	0.5
FedMGDA+	ϵ	0.5	0.05	0.05	0.5
PropFair	M	5.0	2.0	2.0	2.0

[2] `http://cs231n.stanford.edu/`

Table 4: The best learning rates used for different datasets and algorithms, based on grid search.

Datasets	FedAvg	q-FFL	AFL	PropFair	TERM	GIFAIR-FL	FedMGDA+
CIFAR-10	5e-3	5e-2	1e-2	5e-2	1e-2	1e-2	1e-2
CIFAR-100	5e-3	2e-2	1e-2	1e-2	5e-3	1e-2	1e-2
TinyImageNet	2e-2	2e-2	2e-2	5e-2	2e-2	2e-2	1e-2
Shakespeare	2	2	2	2	2	2	2

We adapt hierarchical TERM from Li et al. (2021a), with client-level fairness ($\alpha > 0$) and no sample-level fairness ($\tau = 0$). For each dataset, we tune α (user-level parameter) from $\{0.01, 0.1, 0.5\}$. table 3 shows the optimal value of α used for different datasets is 0.5. For AFL we tune the learning rate γ_w from the corresponding grid and choose the default hyperparameter $\gamma_\lambda = 0.1$. For q-FFL, we run the q-FedAvg algorithm from Li et al. (2020c) with the default Lipschitz constant $L = 1/\eta$ from where η is the learning rate.[3] For each dataset we tune q from $\{0.1, 1.0, 5.0\}$. For all datasets we find $q = 0.1$ has the best performance. We also find that $q = 5$ often leads to divergence during training.

For PropFair we fix $\epsilon = 0.2$ and tune M (Algorithm 1) from $M = 2, 3, 4, 5$. Table 3 shows the optimal values of M used for different datasets. A rule of thumb is to first take a large M (say $M = 10$) and then gradually reduce this value so as to obtain better performance. Given a learning rate η, we use the learning rate $\eta\frac{\epsilon}{M}$ when the loss is greater than $M - \epsilon$, and η otherwise.

In addition to the fair FL algorithms in the main text, we compare with two additional baselines in our appendices: GIFAIR-FL (Yue et al., 2021) and FedMGDA+ (Hu et al., 2022). For GIFAIR-FL we first compute $\lambda_{\max}$ and choose λ from $\{0.1\lambda_{\max}, 0.5\lambda_{\max}, 0.9\lambda_{\max}\}$. For FedMGDA+, we choose ϵ from $\{0.05, 0.1, 0.5\}$ as implemented in Hu et al. (2022). One minor difference is that we fix the global learning rate to be 1.0.

After finding the best hyperparameters for each algorithm, we record the best learning rates in Table 4. For CIFAR-10/CIFAR-100/TinyImageNet/Shakespeare, we take 100/400/400/100 communication rounds respectively, in which cases we find most fair FL algorithms converge.

C.3 Detailed results

In Table 5, we report different statistics across clients, for all the algorithms and datasets we study in this work. These statistical quantities include:

- The mean of test accuracies of all clients;
- The standard deviation of client accuracies;
- The worst test accuracy among the clients;
- The mean of test accuracies across the worst 10% clients;
- The best test accuracy among the clients.
- The mean of test accuracies across the best 10% clients.

For each algorithm we take three different runs and report the mean and standard deviation of different statistical indices. In all the experiments we have used 64 as the default batch size. Table 5 shows that PropFair is consistently advantageous across various datasets over the state-of-the-art algorithms.

C.4 Additional evaluation metrics

In this subsection, we perform comparison with baseline algorithms on CIFAR-100 using additional evaluating metrics, including worst 20% and 30% test accuracies. One can see that our algorithm remains the state-of-the-art among a large variety of algorithms.

[3] https://github.com/litian96/fair_flearn/tree/master/flearn/trainers

Table 5: Comparison among federated learning algorithms on CIFAR-10, CIFAR-100, Shakespeare and StackOverflow datasets with test accuracies (%) from clients. All algorithms are fine-tuned. **Mean**: the average of performances across all clients; **Std**: standard deviation of client test accuracies; **Worst/Best**: the worst/best test accuracy from clients; **Worst (10%)/Best(10%)**: the average of performance across the worst/best 10% clients. Note that for CIFAR-10/100 the worst (best) case accuracy is the same as the worst (best) 10% accuracy since we have 10 clients.

Dataset	Algorithm	Mean	Std	Worst	Worst (10%)	Best	Best (10%)
CIFAR-10	FedAvg	$63.63_{\pm 0.48}$	$5.38_{\pm 0.43}$	$53.49_{\pm 1.67}$	$53.49_{\pm 1.67}$	$72.37_{\pm 0.53}$	$72.37_{\pm 0.53}$
	q-FFL	$57.27_{\pm 0.47}$	$5.68_{\pm 0.16}$	$47.28_{\pm 0.26}$	$47.28_{\pm 0.26}$	$66.71_{\pm 1.24}$	$66.71_{\pm 1.24}$
	AFL	$64.29_{\pm 0.40}$	$4.48_{\pm 0.70}$	$56.16_{\pm 1.56}$	$56.16_{\pm 1.56}$	$71.55_{\pm 0.84}$	$71.55_{\pm 0.84}$
	TERM	$63.81_{\pm 0.62}$	$4.96_{\pm 0.42}$	$56.22_{\pm 1.24}$	$56.22_{\pm 1.24}$	$71.51_{\pm 0.42}$	$71.51_{\pm 0.42}$
	GIFAIR-FL	$63.81_{\pm 0.23}$	$5.05_{\pm 0.04}$	$54.24_{\pm 1.14}$	$54.24_{\pm 1.14}$	$72.41_{\pm 0.88}$	$72.41_{\pm 0.88}$
	FedMGDA+	$61.92_{\pm 0.93}$	$4.93_{\pm 0.44}$	$52.84_{\pm 1.12}$	$52.84_{\pm 1.12}$	$70.42_{\pm 1.72}$	$70.42_{\pm 1.72}$
	PropFair	$\mathbf{64.75}_{\pm 0.10}$	$\mathbf{4.46}_{\pm 0.63}$	$\mathbf{58.14}_{\pm 0.89}$	$\mathbf{58.14}_{\pm 0.89}$	$\mathbf{72.72}_{\pm 2.35}$	$\mathbf{72.72}_{\pm 2.35}$
CIFAR-100	FedAvg	$29.94_{\pm 0.81}$	$4.06_{\pm 0.37}$	$25.26_{\pm 1.50}$	$25.26_{\pm 1.50}$	$40.29_{\pm 0.85}$	$40.29_{\pm 0.85}$
	q-FFL	$28.53_{\pm 0.58}$	$4.53_{\pm 0.11}$	$23.33_{\pm 0.72}$	$23.33_{\pm 0.72}$	$39.82_{\pm 1.02}$	$39.82_{\pm 1.02}$
	AFL	$30.33_{\pm 0.27}$	$3.68_{\pm 0.40}$	$25.49_{\pm 1.12}$	$25.49_{\pm 1.12}$	$39.21_{\pm 0.98}$	$39.21_{\pm 0.98}$
	TERM	$30.35_{\pm 0.28}$	$3.50_{\pm 0.37}$	$26.46_{\pm 0.36}$	$26.46_{\pm 0.36}$	$39.39_{\pm 0.90}$	$39.39_{\pm 0.90}$
	GIFAIR-FL	$30.63_{\pm 0.37}$	$3.58_{\pm 0.17}$	$26.99_{\pm 0.38}$	$26.99_{\pm 0.38}$	$40.03_{\pm 0.62}$	$40.03_{\pm 0.62}$
	FedMGDA+	$23.69_{\pm 0.98}$	$3.52_{\pm 0.33}$	$19.01_{\pm 0.87}$	$19.01_{\pm 0.87}$	$32.51_{\pm 1.86}$	$32.51_{\pm 1.86}$
	PropFair	$\mathbf{31.84}_{\pm 0.67}$	$\mathbf{3.10}_{\pm 0.47}$	$\mathbf{28.85}_{\pm 0.94}$	$\mathbf{28.85}_{\pm 0.94}$	$\mathbf{40.12}_{\pm 1.80}$	$\mathbf{40.12}_{\pm 1.80}$
TinyImageNet	FedAvg	$16.14_{\pm 0.59}$	$\mathbf{2.33}_{\pm 0.07}$	$11.07_{\pm 0.78}$	$11.81_{\pm 0.67}$	$20.23_{\pm 1.11}$	$19.91_{\pm 0.90}$
	q-FFL	$\mathbf{18.84}_{\pm 0.02}$	$3.23_{\pm 0.25}$	$12.12_{\pm 0.58}$	$13.06_{\pm 0.66}$	$\mathbf{24.19}_{\pm 0.25}$	$\mathbf{23.69}_{\pm 0.19}$
	AFL	$16.43_{\pm 0.58}$	$2.34_{\pm 0.04}$	$11.34_{\pm 1.24}$	$12.32_{\pm 0.66}$	$20.70_{\pm 0.64}$	$20.21_{\pm 0.49}$
	TERM	$16.41_{\pm 0.29}$	$2.75_{\pm 0.27}$	$10.67_{\pm 0.47}$	$11.55_{\pm 0.40}$	$21.75_{\pm 1.19}$	$20.97_{\pm 0.71}$
	GIFAIR-FL	$16.54_{\pm 0.41}$	$2.70_{\pm 0.17}$	$11.34_{\pm 0.47}$	$11.92_{\pm 0.15}$	$22.28_{\pm 0.46}$	$21.47_{\pm 0.50}$
	FedMGDA+	$13.94_{\pm 0.20}$	$2.70_{\pm 0.30}$	$9.45_{\pm 0.03}$	$9.73_{\pm 0.12}$	$19.15_{\pm 0.06}$	$18.62_{\pm 0.53}$
	PropFair	$18.04_{\pm 0.74}$	$2.69_{\pm 0.08}$	$\mathbf{12.63}_{\pm 1.57}$	$\mathbf{13.51}_{\pm 1.19}$	$23.68_{\pm 0.49}$	$23.02_{\pm 0.30}$
Shakespeare	FedAvg	$50.54_{\pm 0.12}$	$1.22_{\pm 0.07}$	$48.18_{\pm 0.17}$	$48.26_{\pm 0.17}$	$52.33_{\pm 0.29}$	$52.15_{\pm 0.12}$
	qFFL	$50.69_{\pm 0.14}$	$1.05_{\pm 0.02}$	$48.74_{\pm 0.21}$	$48.83_{\pm 0.22}$	$52.35_{\pm 0.08}$	$52.25_{\pm 0.13}$
	AFL	$\mathbf{52.54}_{\pm 0.08}$	$1.25_{\pm 0.05}$	$\mathbf{49.86}_{\pm 0.29}$	$\mathbf{50.13}_{\pm 0.12}$	$\mathbf{54.47}_{\pm 0.29}$	$\mathbf{54.22}_{\pm 0.18}$
	TERM	$50.90_{\pm 0.11}$	$1.27_{\pm 0.03}$	$48.10_{\pm 0.15}$	$48.45_{\pm 0.20}$	$52.65_{\pm 0.39}$	$52.47_{\pm 0.26}$
	GIFAIR-FL	$50.67_{\pm 0.28}$	$1.25_{\pm 0.04}$	$48.22_{\pm 0.26}$	$48.32_{\pm 0.30}$	$52.50_{\pm 0.24}$	$52.45_{\pm 0.21}$
	FedMGDA+	$44.17_{\pm 0.18}$	$\mathbf{0.99}_{\pm 0.02}$	$42.42_{\pm 0.10}$	$42.67_{\pm 0.05}$	$46.30_{\pm 0.20}$	$46.10_{\pm 0.20}$
	PropFair	$52.28_{\pm 0.08}$	$1.20_{\pm 0.04}$	$49.50_{\pm 0.41}$	$49.76_{\pm 0.20}$	$54.10_{\pm 0.11}$	$53.88_{\pm 0.12}$

Table 6: Comparison using worst 20% and 30% test accuracies on the CIFAR-100 dataset. The hyperparameters and learning rates are the same as in Table 3 and Table 4.

Metric	PropFair	AFL	FedAvg	TERM	q-FFL	GIFAIR-FL	FedMGDA+
worst 20%	$\mathbf{29.08}_{\pm 0.77}$	$26.15_{\pm 0.69}$	$25.68_{\pm 1.66}$	$26.90_{\pm 0.33}$	$23.94_{\pm 0.51}$	$27.33_{\pm 0.35}$	$19.77_{\pm 0.87}$
worst 30%	$\mathbf{29.29}_{\pm 0.63}$	$26.63_{\pm 0.24}$	$26.26_{\pm 1.48}$	$27.25_{\pm 0.25}$	$24.53_{\pm 0.53}$	$27.66_{\pm 0.23}$	$20.33_{\pm 1.10}$

D Dual View of Fair FL Algorithms

In this section we derive the convex conjugates of the generalized means for each algorithm. We sometimes extend the domain of $\boldsymbol{f}$ to obtain a clear form of A_φ^*, while ensuring the equality of equation 2.10.

D.1 Dual View of FedAvg

For FedAvg, we have $\varphi(t) = t$ and the generalized mean can be written as:

$$\mathsf{A}_\varphi(\boldsymbol{f}) = \sum_i p_i f_i, \tag{D.1}$$

where we extend the domain of $\boldsymbol{f}$ to be $\mathbb{R}^n$. The convex conjugate can be written as:

$$\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \sup_{\boldsymbol{f}\in\mathbb{R}^n} (\boldsymbol{\lambda} - \boldsymbol{p})^\top \boldsymbol{f} \tag{D.2}$$

Solving it yields:

$$\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \begin{cases} 0 & \text{if } \boldsymbol{\lambda} = \boldsymbol{p}, \\ \infty & \text{otherwise.} \end{cases} \tag{D.3}$$

Bringing the equation above to equation 2.10 we obtain the original form of FedAvg.

D.2 Dual View of q-FFL and AFL

Let us now derive the conjugate function for q-FFL. With $\varphi(t) = t^{q+1}$ $(q > 0)$ we have:

$$\mathsf{A}_\varphi(\boldsymbol{f}) = \left(\sum_i p_i f_i^{q+1}\right)^{\frac{1}{q+1}}, \tag{D.4}$$

where we assume $\operatorname{dom} \boldsymbol{f} = \mathbb{R}^n_+$. The convex conjugate can thus be written as:

$$\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \sup_{\boldsymbol{f}\geq\boldsymbol{0}} \boldsymbol{\lambda}^\top \boldsymbol{f} - \left(\sum_i p_i f_i^{q+1}\right)^{\frac{1}{q+1}}. \tag{D.5}$$

If $\sum_i p_i^{-1/q} \lambda_i^{(q+1)/q} > 1$, we can take $f_i = \lambda_i^{1/q} p_i^{-1/q} t$ and the maximand of equation D.5 becomes:

$$\begin{aligned} \boldsymbol{\lambda}^\top \boldsymbol{f} - \left(\sum_i p_i f_i^{q+1}\right)^{\frac{1}{q+1}} &= \sum_i \lambda_i^{(q+1)/q} p_i^{-1/q} t - \left(\sum_i \lambda_i^{(q+1)/q} p_i^{-1/q}\right)^{\frac{1}{q+1}} t \\ &= \left(\sum_i \lambda_i^{(q+1)/q} p_i^{-1/q} - \left(\sum_i \lambda_i^{(q+1)/q} p_i^{-1/q}\right)^{\frac{1}{q+1}}\right) t. \end{aligned} \tag{D.6}$$

By taking $t \to \infty$ we have $\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \infty$. Therefore we must constrain $\sum_i p_i^{-1/q} \lambda_i^{(q+1)/q} \leq 1$. In this case, we can utilize Hölder's inequality to obtain $\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = 0$. In summary, the convex conjugate for $\boldsymbol{\lambda} \geq \boldsymbol{0}$ is:

$$\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \begin{cases} 0, & \text{if } \sum_i p_i^{-1/q} \lambda_i^{(q+1)/q} \leq 1, \\ \infty & \text{otherwise.} \end{cases} \tag{D.7}$$

Taking $q \to \infty$ the function above becomes the one for AFL:

$$\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \begin{cases} 0, & \text{if } \|\boldsymbol{\lambda}\|_1 \leq 1, \\ \infty & \text{otherwise.} \end{cases} \tag{D.8}$$

D.3 Dual View of TERM

We continue to derive the convex conjugate of the generalized mean of TERM. Recall that $\varphi(t) = e^{\alpha t}$ with $\alpha > 0$. The generalized mean can be written as:

$$\mathsf{A}_\varphi(\boldsymbol{f}) = \frac{1}{\alpha} \log \left(\sum_i p_i e^{\alpha f_i} \right), \tag{D.9}$$

where we extend the domain of f to be $\mathbb{R}^n$. The convex conjugate is:

$$\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \sup_{f \in \mathbb{R}^n} \boldsymbol{\lambda}^\top \boldsymbol{f} - \frac{1}{\alpha} \log \left(\sum_i p_i e^{\alpha f_i} \right). \tag{D.10}$$

If any $\lambda_i < 0$, we can take the corresponding $f_i \to -\infty$ and thus $\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \infty$. If $\boldsymbol{\lambda}^\top \mathbf{1} \neq 1$, we can impose $f = t\mathbf{1}$ and obtain that:

$$\boldsymbol{\lambda}^\top \boldsymbol{f} - \frac{1}{\alpha} \log \left(\sum_i p_i e^{\alpha f_i} \right) = (\boldsymbol{\lambda}^\top \mathbf{1} - 1)t. \tag{D.11}$$

By taking $t \to \infty$ or $t \to -\infty$ we get $\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \infty$. Now let us assume $\boldsymbol{\lambda} \geq \mathbf{0}$ and $\boldsymbol{\lambda}^\top \mathbf{1} = 1$. By requiring stationarity in equation D.10 we find the necessary and sufficient optimality condition:

$$\lambda_i = \frac{p_i e^{\alpha f_i}}{\sum_i p_i e^{\alpha f_i}}, \tag{D.12}$$

which can always to satisfied with our assumption. Denote $c = \sum_i p_i e^{\alpha f_i}$ we can solve equation D.12 to obtain $f_i = \frac{1}{\alpha} \log \left(\frac{c\lambda_i}{p_i} \right)$. Bringing it back to equation D.10 the convex conjugate becomes:

$$\begin{aligned} \mathsf{A}^*_\varphi(\boldsymbol{\lambda}) &= \sum_i \frac{\lambda_i}{\alpha} \log \left(\frac{c\lambda_i}{p_i} \right) - \frac{1}{\alpha} \log c \\ &= \sum_i \frac{\lambda_i}{\alpha} \log \frac{\lambda_i}{p_i}, \end{aligned} \tag{D.13}$$

where we used the condition $\boldsymbol{\lambda}^\top \mathbf{1} = 1$. Since we have the constraint that $\boldsymbol{\lambda} \geq \mathbf{0}$, equation 2.10 still holds. Therefore, we get:

$$\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \begin{cases} \sum_i \frac{\lambda_i}{\alpha} \log \frac{\lambda_i}{p_i} & \text{if } \boldsymbol{\lambda} \geq \mathbf{0}, \boldsymbol{\lambda}^\top \mathbf{1} = 1, \\ \infty & \text{otherwise.} \end{cases} \tag{D.14}$$

D.4 Dual View of PropFair

Let us derive the dual of the generalized mean for PropFair in the same framework as in Section 2.4. Note that

$$\varphi(t) = -\log(M - t), \tag{D.15}$$

and therefore the generalized mean is:

$$\mathsf{A}_\varphi(\boldsymbol{f}) = \varphi^{-1} \left(\sum_i p_i \varphi(f_i) \right) = M - \prod_{i=1}^n (M - f_i)^{p_i}, \tag{D.16}$$

where we require $\boldsymbol{f} \leq M\mathbf{1}$. We observe that A_φ is a convex function, since it is composition of the generalized geometric mean (which is concave) and affine transformation. Now we compute the dual function

$$\begin{aligned} \mathsf{A}^*_\varphi(\boldsymbol{\lambda}) &= \sup_{\boldsymbol{f} \leq M\mathbf{1}} \boldsymbol{\lambda}^\top \boldsymbol{f} - \mathsf{A}_\varphi(\boldsymbol{f}) \\ &= \sup_{\boldsymbol{f} \leq M\mathbf{1}} \boldsymbol{\lambda}^\top \boldsymbol{f} + \prod_{i=1}^n (M - f_i)^{p_i} - M \end{aligned} \tag{D.17}$$

If any entry λ_i is non-positive, clearly we can let $f_i \to -\infty$ so that $\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) \to \infty$. For positive $\boldsymbol{\lambda}$, and $\prod_{i=1}^n \left(\frac{\lambda_i}{p_i}\right)^{p_i} < 1$, we can take $f_i = M - c\frac{p_i}{\lambda_i}$ and get:

$$\begin{aligned}\boldsymbol{\lambda}^\top \boldsymbol{f} + \prod_{i=1}^n (M - f_i)^{p_i} - M &= \sum_{i=1}^n (M\lambda_i - cp_i) + \prod_{i=1}^n \left(\frac{cp_i}{\lambda_i}\right)^{p_i} - M \\ &= M(\boldsymbol{\lambda}^\top \mathbf{1} - 1) + \left(\prod_i \left(\frac{p_i}{\lambda_i}\right)^{p_i} - 1\right) c \end{aligned} \tag{D.18}$$

Since $c \geq 0$ is arbitrary, we can take $c \to \infty$ and thus $\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \infty$. Otherwise, if $\prod_{i=1}^n \left(\frac{\lambda_i}{p_i}\right)^{p_i} \geq 1$, then we have:

$$\begin{aligned}\boldsymbol{\lambda}^\top \boldsymbol{f} + \prod_{i=1}^n (M - f_i)^{p_i} - M &= \prod_{i=1}^n (M - f_i)^{p_i} - \boldsymbol{\lambda}^\top (M\mathbf{1} - \boldsymbol{f}) + M(\boldsymbol{\lambda}^\top \mathbf{1} - 1) \\ &\leq \prod_{i=1}^n (M - f_i)^{p_i} - \prod_{i=1}^n \left(\frac{\lambda_i}{p_i}\right)^{p_i} \prod_{i=1}^n (M - f_i)^{p_i} + M(\boldsymbol{\lambda}^\top \mathbf{1} - 1) \\ &\leq M(\boldsymbol{\lambda}^\top \mathbf{1} - 1), \end{aligned} \tag{D.19}$$

where in the second line we used the AM-GM inequality and in the last line we used $\prod_{i=1}^n \left(\frac{\lambda_i}{p_i}\right)^{p_i} \geq 1$. This equality can always be achieved by taking $\boldsymbol{f} = M\mathbf{1}$. In summary, we have:

$$\mathsf{A}^*_\varphi(\boldsymbol{\lambda}) = \begin{cases} M(\boldsymbol{\lambda}^\top \mathbf{1} - 1), & \text{if } \boldsymbol{\lambda} \geq \mathbf{0} \text{ and } \prod_{i=1}^n \left(\frac{\lambda_i}{p_i}\right)^{p_i} \geq 1, \\ \infty, & \text{otherwise.} \end{cases} \tag{D.20}$$

We remark that A^*_φ is closed (since its domain is closed). If we want to enforce $\boldsymbol{f} \geq \mathbf{0}$ when computing the dual function, we simply apply the convolution formula:

$$\overline{\mathsf{A}}^*_\varphi(\boldsymbol{\lambda}) = \inf_{\boldsymbol{\lambda} \leq \boldsymbol{\gamma}} \mathsf{A}^*_s(\boldsymbol{\gamma}). \tag{D.21}$$

However, the formula for A^*_φ suffices for our purpose so we need not compute the above explicitly.

Applying the above conjugation result we can rewrite PropFair as:

$$\min_{\boldsymbol{\theta}} \mathsf{A}_\varphi(\boldsymbol{f}(\boldsymbol{\theta})) = \min_{\boldsymbol{\theta}} \max_{\boldsymbol{\lambda} \geq \mathbf{0}} \boldsymbol{\lambda}^\top \boldsymbol{f}(\boldsymbol{\theta}) - \mathsf{A}^*_\varphi(\boldsymbol{\lambda}). \tag{D.22}$$

We focus on the inner maximization so that we know the weights we put on each client:

$$\begin{aligned}\max_{\boldsymbol{\lambda} \geq \mathbf{0}} \boldsymbol{\lambda}^\top \boldsymbol{f}(\boldsymbol{\theta}) - \mathsf{A}^*_\varphi(\boldsymbol{\lambda}) &= \max_{\boldsymbol{\lambda} \geq \mathbf{0}, \prod_{i=1}^n (\lambda_i/p_i)^{p_i} \geq 1} \boldsymbol{\lambda}^\top \boldsymbol{f} - (\boldsymbol{\lambda}^\top \mathbf{1} - 1)M \\ &= \max_{\boldsymbol{\lambda} \geq \mathbf{0}, \prod_{i=1}^n (\lambda_i/p_i)^{p_i} \geq 1} M - \boldsymbol{\lambda}^\top (M\mathbf{1} - \boldsymbol{f}). \end{aligned} \tag{D.23}$$

Using the AM-GM inequality we have:

$$\boldsymbol{\lambda}^\top (M\mathbf{1} - \boldsymbol{f}) \geq \prod_{i=1}^n \left(\frac{\lambda_i}{p_i}\right)^{p_i} \prod_{i=1}^n (M - f_i)^{p_i} \geq \prod_{i=1}^n (M - f_i)^{p_i}, \tag{D.24}$$

where the equality is attained iff $\prod_{i=1}^n \left(\frac{\lambda_i}{p_i}\right)^{p_i} = 1$ and

$$\lambda_i \propto \frac{p_i}{M - f_i}. \tag{D.25}$$

Thus, we verify again that the optimal value of equation D.23 is:

$$M - \prod_{i=1}^{n}(M - f_i)^{p_i} = \mathsf{A}_{\varphi}(\boldsymbol{f}), \tag{D.26}$$

and we retrieve our original objective. equation D.25 tells us that we are essentially solving a linearly weighted combination of $f_1, \ldots, f_n$, but with more weights on the worse-off clients, since $\frac{p_i}{M-f_i}$ is larger for larger f_i.

E More Related Work

In this appendix we introduce more related work, including multi-objective optimization, fairness in FL, as well as various definitions of fairness from multiple fields.

E.1 Multi-objective optimization

Multi-Objective Optimization (MOO) has been intensively studied in the field of operation research (Geoffrion, 1968; Yu & Zeleny, 1975; Jahn et al., 2009). The goal of MOO is to minimize a series of objectives $f_1, f_2, \ldots, f_n$ based on their best trade-offs. This is directly related to federated learning (Hu et al., 2022) because one can treat the loss function of each client as an objective.

In MOO, Pareto optimality is often desired. To find a Pareto optimum, one way is to use an *aggregating objective* (a.k.a. scalarizing function Lootsma et al. (1995)). We list some common choices of this aggregating objective:

- *Linear weighting method (Geoffrion, 1968):* this method converts MOO into the problem of minimizing the convex combination of client objectives:

$$\min_{\boldsymbol{x}\in\mathcal{X}} \sum_{i=1}^{n} \lambda_i f_i(\boldsymbol{x}), \tag{E.1}$$

  with $\boldsymbol{\lambda} \in \Delta_{n-1}$ in the $(n-1)$-simplex, and $\mathcal{X}$ the domain of $\boldsymbol{x}$. Such solution is always Pareto optimal and the method has been used in FedAvg (McMahan et al., 2017). A well-known difficulty is that it cannot generate point in the nonconvex part of the Pareto front (Audet et al., 2008).

- *Reference point* (Audet et al., 2008): This method requires proximity to the *ideal point*: $\boldsymbol{r} = (\min_{\boldsymbol{x}\in\mathcal{X}} f_1(\boldsymbol{x}), \ldots, \min_{\boldsymbol{x}\in\mathcal{X}} f_n(\boldsymbol{x}))$, measured by ℓ_q-norm:

$$\min_{\boldsymbol{x}\in\mathcal{X}} \|\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{r}\|_q^q := \sum_{i=1}^{n}(f_i(\boldsymbol{x}) - r_i)^q, \tag{E.2}$$

  with $\boldsymbol{f}(\boldsymbol{x}) := (f_1(\boldsymbol{x}), \ldots, f_n(\boldsymbol{x}))$ and $\|\cdot\|_q$ the ℓ_q-norm ($q \geq 1$). This method has been applied to federated learning as q-FFL (Li et al., 2020c) (by assuming $\boldsymbol{r} = \boldsymbol{0}$).

- *Weighted geometric mean* (Lootsma et al., 1995): this method converts MOO to a single-objective formulation by maximizing the weighted geometric mean between elements of the *nadir point* and the client objectives:

$$\max_{x\in\mathcal{X}} \prod_{i=1}^{n}(p_i - f_i(\boldsymbol{x}))^{\lambda_i}, \text{ such that } f_i(\boldsymbol{x}) \leq p_i \text{ for any } i \text{ and } \boldsymbol{x} \in \mathcal{X}, \tag{E.3}$$

  where $\boldsymbol{p}$ is called a *nadir point*, defined as (Lootsma et al., 1995):

$$p_i = \max_{j=1,2,\ldots,n} f_i(\boldsymbol{x}_j^*), \tag{E.4}$$

  with $\boldsymbol{x}_j^* = \arg\min_{\boldsymbol{x}\in\mathcal{X}} f_j(\boldsymbol{x})$ the optimizer of function f_j. The λ_i's are the weights for each client and they are positive. If we take $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n) = \boldsymbol{1}$, then it resembles our objective in equation 3.7.

E.2 Fairness in Federated Learning

As FL has been deployed to more and more real-world applications, it has become a major challenge to guarantee that FL models has no discrimination against certain clients and/or sensitive attributes. Since different participants may contribute differently to the final model's quality, it is necessary to provide a fair mechanism to encourage user participation.

Besides the related work we mentioned in the main paper (McMahan et al., 2017; Mohri et al., 2019; Li et al., 2020b), another direction of research tries to directly encourage the involvement of user participation, by providing some rewards to fairly recognize the *contributions* of clients. For example, Lyu et al. (2020) designed a local credibility mutual evaluation mechanism to enforce good contributors get more credits. Concretely, each client computes the contribution of every other client by investigating the label similarities of the synthetic samples generated by the clients' differential private GANs (Goodfellow et al., 2014). Kang et al. (2020) proposed a pairwise measurement of contribution. Reputation scores are kept at each client for all other clients, and are updated by a multi-weight subjective logic model. Yu et al. (2020) proposed a Federated Learning Incentivizer (FLI) payoff-sharing scheme, which dynamically divides a given budget among clients by optimizing their joint utility while minimizing their discrepancy. The objective function takes into account the amount of payoff and the waiting time to receive the payoff. Wang et al. (2020) analyzed the contribution from the data side, and proposed the federated Shapley Value (SV) for data valuation. While preserving the desirable properties of the canonical SV, this federated SV can be calculated with no extra communication overhead, making it suitable for the FL scenarios.

The above methods already applied some objective functions that reflect the concept of proportional fairness, e.g., payoff proportional to the contribution. However, they mostly apply fixed contribution-reward assignment rules, without explicit definitions of proportional fairness or theoretical guarantee.

E.3 Definitions of fairness

Fairness has been a perennial topic in social choice (Sen, 1986), communication (Jain et al., 1984), law (Rawls, 1999) and machine learning (Barocas et al., 2017). Whenever we have multiple agents and limited resources, we need fairness to allocate the resources. There have been many definitions of fairness, such as individual fairness (Dwork et al., 2012), demographic fairness, counterfactual fairness and proportional fairness.

In this section, we introduce definitions of fairness from various perspectives including social choice, communication and machine learning, and study the implications in the setting of FL.

E.3.1 Social Choice and Law

We review some principles for fairness and justice in social choice (Sen, 1986) and law (Rawls, 1999), which resembles FL: we can treat the shared global model as a public policy and clients as social agents.

- *Utilitarian rule (Maskin, 1978):* suppose we have n clients and their loss functions are f_i, the utilitarian rule aims to minimize the sum of the loss functions, e.g.,

$$\min_{\boldsymbol{\theta}} \sum_i f_i(\boldsymbol{\theta}), \tag{E.5}$$

  with $\boldsymbol{\theta}$ the global model parameter. This utilitarian rule represents the utilitarian philosophy: as long as the overall performance of the whole society is optimal, we call the society to be fair. A utilitarian policy is Pareto-optimal but not vice versa. With model homogeneity, equation equation E.5 is nothing but the objective for FedAvg (McMahan et al., 2017), although the FedAvg algorithm may not always converge to the global optimum even in linear regression (Pathak & Wainwright, 2020).

- *Egalitarian rule (Rawls, 1974; 1999):* The egalitarian rule, also known as the maximin criterion represents egalitarianism in political philosophy. Instead of maximizing the overall performance as

in equation E.5, an egalitarian wants to maximizing the performance of the worst-case client, i.e., we solve the following optimization problem:

$$\min_{\boldsymbol{\theta}} \max_i f_i(\boldsymbol{\theta}). \tag{E.6}$$

This accords with Agnostic FL (Mohri et al., 2019). The egalitarian problem equation E.6 may not always be Pareto optimal, e.g., $(f_1, f_2, f_3) = (1, 1, 1)$ and $(f_1, f_2, f_3) = (1, 0.9, 0.8)$ can both be the optimal solution of equation E.6, but the former is not Pareto optimal.

E.3.2 Fairness in wireless communications

Since resource allocation is common in communication, different notions of fairness have also been proposed and studied. We review some common fairness definitions in communication:

- *Max-min fairness / Pareto optimal* (Bertsekas & Gallager, 1987): this definition says at the fair solution, one cannot simultaneously improve the performance of all clients, which is equivalent to the definition of Pareto optimal. The corresponding algorithm in FL for finding a Pareto optimum is FedMGDA+ (Hu et al., 2022).
- *Proportional-fair rule (Kelly, 1997; Bertsimas et al., 2011):* proportional fairness aims to find a solution $\boldsymbol{\theta}^*$ such that for all $\boldsymbol{\theta}$ in the domain:

$$\sum_i \frac{u_i(\boldsymbol{\theta}) - u_i(\boldsymbol{\theta}^*)}{u_i(\boldsymbol{\theta}^*)} \leq 0, \tag{E.7}$$

  with u_i the utility function of client i, e.g., the test accuracy. This problem aims to find a policy such that the total relative utility cannot be improved. Proportional-fairness has been studied in communication (e.g. Seo & Lee, 2006) for scheduling but the application in FL has not been seen.
- *Harmonic mean* (Dashti et al., 2013): the method maximizes the harmonic mean of the utility functions of each client, that is, we solve the following optimization problem:

$$\max_{\boldsymbol{\theta}} \frac{n}{\sum_i u_i(\boldsymbol{\theta})^{-1}} \tag{E.8}$$

  In a similar vein we can find its optimality condition, assuming the utility set $\mathcal{U}$ is convex:

$$\sum_{i=1}^{n} \frac{u_i - u_i^*}{(u_i^*)^2} \leq 0, \text{ for all } \boldsymbol{u} \in \mathcal{U}. \tag{E.9}$$

  Compared to proportional fairness, it simply amounts to squaring the denominator.

E.3.3 Fairness in machine learning

Fairness has been studied in machine learning for almost a decade (Barocas et al., 2017). A large body of work focuses on proposing machine learning algorithms for achieving different definitions of fairness. These definitions are often incompatible with each other, i.e., one cannot achieve two definitions of fairness simultaneously. Let us review some common definitions, using classification as an illustrating example:

- *Group fairness / statistical parity / demographic parity* (DP, Dwork et al., 2012; Zemel et al., 2013): this definition requires that the prediction is independent of the subgroup (e.g., race, gender). Denote $\mathtt{Y}$ as the prediction and $\mathtt{S}$ as the sensitive attribute, this definition requires $\mathtt{Y} \perp \mathtt{S}$, where the symbol $\perp$ denotes statistical independence. This is the simplest definition of fairness, and probably what people think of at a first thought. However, this definition can be problematic. For instance, suppose in an FL system a subgroup of clients have poor performance (e.g. due to communication, memory), and then to achieve better group fairness one can deliberately lower the performance of high-performing clients, and thus the overall performance is lower. Moreover, DP would forbid us to achieve the optimal performance if the true labels are not independent of the sensitive attribute (Hardt et al., 2016; Zhao & Gordon, 2019).

- *Equalized odds (EO)* (Hardt et al., 2016): this defintion requires demographic parity given each true label class. Define $\mathtt{T}$ as the random variable for the true label. Equalized odds requires that $\mathtt{Y} \perp \mathtt{S} \,|\, \mathtt{T}$ for *any* $\mathtt{T}$ and equal opportunity requires that $\mathtt{Y} \perp \mathtt{S} \,|\, \mathtt{T}$ for *some* $\mathtt{T}$. Different from DP, this conditioning allows the prediction to align with the true label. In the binary setting, EO and DP cannot be simultaneously achieved (Barocas et al., 2017).
- *Calibration / Predictive Rate Parity* (Gebel, 2009): this definition requires that among the samples having a prediction score $\mathtt{Y}$, the expectation of the true label $\mathtt{T}$ should match the prediction score, i.e., $\mathbb{E}[\mathtt{T}|\mathtt{Y}] = \mathtt{Y}$. In the context of fairness, calibration says that $\mathtt{T} \perp \mathtt{S} \,|\, \mathtt{Y}$. Under mild assumptions, calibration and EO cannot be simultaneously achieved (Pleiss et al., 2017). Similarly, calibration and DP cannot be simultaneously achieved.
- *Individual fairness* (Dwork et al., 2012): this concept requires that similar samples, as measured by some metric, should have similar predictions.
- *Counterfactual fairness* (Kusner et al., 2017): this definition requires that from any sample, the prediction should be the same had the sensitive attribute taken different values. It follows the notion of counterfactual from casual inference (Pearl, 2000).
- *Accuracy parity* (Zafar et al., 2017): the accuracy for each group remains the same.

Since many concepts conflict with each other (Barocas et al., 2017), there is no unified definition of fairness. In light of this, a dynamical definition of fairness has been proposed (Awasthi et al., 2020). Algorithms for achieving different definitions of fairness include mutual information (Zemel et al., 2013), representation learning (Zemel et al., 2013; Zhao & Gordon, 2019) and Rényi correlation (Baharlouei et al., 2019).