Paper Summary - FCFL: Fairness Compensation-based Federated Learning

Asif Ahmed Utsa

Mobaswirul Islam

0423052090@grad.cse.buet.ac.bd

0424052053@grad.cse.buet.ac.bd

Bangladesh University of Engineering & Technology

Dhaka, Bangladesh

INTRODUCTION

Federated Learning (FL) is a mechanism to train one shared global model keeping the raw data private. But it has some inherent problems. The model can be biased easily mainly for system heterogeneity, participation imbalance, non iid-data etc. This problem is called **unfairness**: some clients get high accuracy, others get poor, some are rarely selected, and a few do most of the work.

FCFL (Fairness Compensation–based Federated Learning) is an approach to reduce this unfairness. It keeps a small "fairness queue value" for each client. The value goes up when a client's accuracy is worse than the global model, and it goes down when the client is selected and contributes. Two lightweight rules then use this queue:

- **Client Selection:** most training slots go to clients with larger queue values with a small random fraction r kept for diversity.
- **Aggregation:** when any queue-values are non-zero client updates are weighted by their queue-value, if all queue-values are zero, the method falls back to standard FedAvg.

With only two knobs— α (how fast the queue value grows) and r (how many(%) random clients to include)—FCFL narrows accuracy gaps across clients while keeping the average accuracy almost unchanged, and it adds only a tiny overhead (each client sends one extra accuracy number per round).

RELATED WORKS

Different works tried to solve this **unfairness** problem in different ways.

Reweighting at aggregation. Some methods change **how the server averages updates**. They give more weight to clients with higher loss or lower accuracy, so their updates count more (e.g., *q-FedAvg*). This can shrink the accuracy gap when data are very different. However, most still **pick clients at random**, so slow or unlucky clients may wait many rounds before they join, and strong reweighting can hurt average accuracy.

New training objectives. Other works **change the goal** from "maximize average accuracy" to "improve the worst client" or "balance several goals at once" (e.g., *AFL*, multi-objective FL). These

can improve fairness but often need **extra tuning** and heavier optimization, which makes deployment harder.

Selection fairness. A third line **controls who gets selected** each round. Clients that have trained less often get picked more; some methods also consider **resource limits** (battery, bandwidth) to spread the workload. This improves participation fairness and reduces communication spikes, but if aggregation stays unchanged the final model can still favor majority data.

In short, most prior work fixes one side of the problem—aggregation, objective design, selection, stability or personalization. Approaches that combine selection and aggregation with one simple signal are less common but attractive due to low overhead and easy deployment.
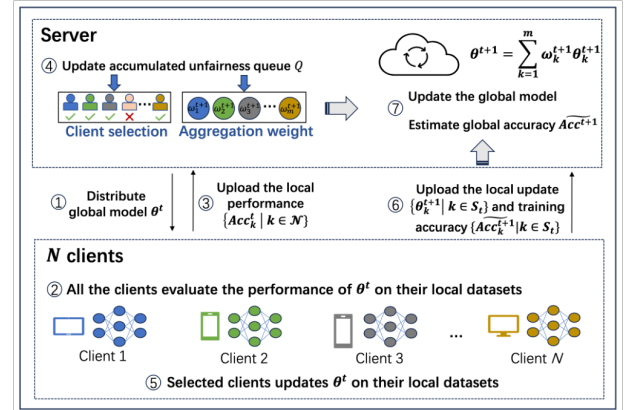
FCFL OVERVIEW

Figure 1: FCFL Framework

Figure 1 shows the FCFL workflow at a high level.

Intuition. Each client keeps a small *unfairness queue value*. The queue value goes up when the client's local accuracy is worse than the global average, and it goes down when the client is selected and its update is used. This single number drives both whether a client will be selected in the next round and how much its weight will be.

(1) Local evaluation & queue update. At the start of a round, clients test the current global model on their own data. If a client

is behind the global model, its gap is positive and the queue increases (scaled by α). If the client was selected last round, the queue decreases slightly.

(2) Client selection. Most clients are chosen with the highest queue values, and a small fraction r is selected at random to ensure diversity.

(3) Aggregation. After local training, the server combines updates. If any queue values are non-zero, updates from higher-queue-value clients get more weight. If all queue-values are zero, the rule falls back to standard FedAvg.

Parameters and overhead. There are only two knobs: α (how fast the queue value grows) and r (how much random selection to keep). Setting $\alpha=0$ or $r=1$ recovers vanilla FedAvg.

Accumulated Unfairness Queue and Client Selection

In Federated Learning, client unfairness is measured by the difference between the estimated global model accuracy and the evaluated local accuracy as in equation 1, since the actual global model is not available during training. To track the cumulative unfairness of each participant, providing a basis for subsequent client selection and weight allocation, a queue $Q_i(t)$ is used. The value of $Q_i(t)$ is calculated using the equation 2.

$$uf_i^t = \begin{cases} \widetilde{Acc}^t - Acc_i^t, & \text{if } \widetilde{Acc}^t > Acc_i^t \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$Q_i(t) = \max\left\{Q_i(t-1) + \alpha uf_i^t - \omega_i^t \cdot \mathbf{1}_{[x_i(t-1)=1]}, 0\right\} \quad (2)$$

$$\omega_i^{t+1} = \begin{cases} \frac{n_i}{\sum_{i=1}^m n_i}, & \text{if } Q_1(t) = \cdots = Q_m(t) = 0 \\ \frac{Q_i(t)}{\sum_{i=1}^m Q_i(t)}, & \text{otherwise} \end{cases} \quad (3)$$

The cumulative unfairness queue $Q_i(t)$ is designed to favor vulnerable clients with low accuracy, those not selected, or those with small weights, resulting in a higher $Q_i(t)$ value that the FCFL algorithm uses to compensate them through client selection and aggregation re-weighting using equation 3. After updating the unfairness queue for all clients, the server selects the top-m clients with the highest unfairness scores to participate in the next training round. This process ensures that vulnerable clients are prioritized and then assigned more weight during aggregation to increase their contribution. However, selecting clients based solely on unfairness can bias the global model towards clients with rare datasets which can negatively impact overall accuracy. To balance this, the FCFL algorithm uses a hyper-parameter r to combine random and unfairness-based selection, ensuring both fairness and global accuracy are maintained.

Communication and Computation Overhead

The primary bottleneck of Federated Learning (FL) is the communication cost between the server and edge devices, as well as the limited local computation power. The proposed FCFL scheme introduces minimal communication overhead, requiring clients to upload their local accuracy (Acc_i^{t+1}) which costs only 8 more bits per round. On the computation side, the FCFL scheme reduces the local computational cost by allowing clients to use mini-batch samples to estimate accuracy instead of evaluating it on their entire local dataset.

EXPERIMENTS

The performance of the FCFL algorithm is evaluated on three public datasets: MNIST, CIFAR-10, and Shakespeare, using the non-IID data distribution to simulate real-world conditions. The experiments utilize three different model types (MLP, CNN, and RNN) and compare FCFL against the classic FedAvg algorithm and other state-of-the-art fairness methods.

Fairness and Efficiency of FCFL

The FCFL method is compared with four other FL algorithms, including FedAvg, q-FedAvg, FedFA, and GIFAIR, to verify its fairness. The results show that FCFL significantly reduces variance, producing the lowest variance in MNIST, CIFAR-10, and Shakespeare, which is a key indicator of fairness. FCFL also improves the performance of the worst-performing 10% of clients. So, the FCFL method can achieve better fairness in FL while maintaining a competitive average accuracy in most cases. In terms of efficiency, the FCFL method learns just as quickly as other methods, with its loss decreasing and accuracy increasing over time. It is also very efficient, not adding much extra time per communication round. This means that FCFL can improve fairness without sacrificing speed or consuming too much time.

Effect of Hyper-parameter r

The effect of the hyperparameter r is evaluated on MNIST and CIFAR-10 datasets. Theoretically, increasing the randomness r improves the overall accuracy, but also makes the system less fair by increasing the variances. On the simple MNIST dataset, changing the parameter r has little effect, but on the more complex CIFAR-10 dataset, a larger r increases both accuracy and performance variance among clients. The best value for r was found to be 0.6, as it reaches a perfect trade-off between accuracy and fairness.

LIMITATIONS

According to the study, though the proposed FCFL improves the faireness by 30.4% with high accuracy, the system was not tested on largely scaled heterogenous datasets. Also the system only focused on accuracy-based fairness, not considerng other fairness metrics. Furthermore, there is a potential risk of overcompensating for certain vulnerable clients, which could negatively impact the global model's performance.

FUTURE WORK

In future, the study can be extended on how to estimate unfairness accurately by incorporating global accuracy and how to select hyper-parameter adaptively to improve overall performance. Furthermore, multi-fairness integration that means combining this approach with selection fairness would be an interesting idea for future work.