Towards Understanding Biased Client Selection in Federated Learning

Yae Jee Cho
Carnegie Mellon University
yaejeec@andrew.cmu.edu

Jianyu Wang
Carnegie Mellon University
jianyuw1@andrew.cmu.edu

Gauri Joshi
Carnegie Mellon University
gaurij@andrew.cmu.edu

Abstract

Federated learning is a distributed optimization paradigm that enables a large number of resource-limited client nodes to cooperatively train a model without data sharing. Previous works analyzed the convergence of federated learning by accounting of data heterogeneity, communication/computation limitations, and partial client participation. However, most assume unbiased client participation, where clients are selected such that the aggregated model update is unbiased. In our work, we present the convergence analysis of federated learning with biased client selection and quantify how the bias affects convergence speed. We show that biasing client selection towards clients with higher local loss yields faster error convergence. From this insight, we propose Power-of-Choice, a communication- and computation-efficient client selection framework that flexibly spans the trade-off between convergence speed and solution bias. Extensive experiments demonstrate that Power-of-Choice can converge up to 3× faster and give 10% higher test accuracy than the baseline random selection.

1 INTRODUCTION

Until recently, machine learning models were largely trained in data centers (Dean et al., 2012) using powerful computing nodes, fast inter-node communication links, and large centrally available training datasets. The future of machine learning lies in moving both data collection as well as model training to the edge. Federated learning (FL) (McMahan et al., 2017; Kairouz et al., 2019; Bonawitz et al., 2019) considers a large number of resource-constrained mobile devices that collect training data from their environment. Due to the devices' limited communication capabilities and privacy, the training data of the devices cannot be directly sent to the server. Instead, the devices locally perform a few iterations of training using local-update stochastic gradient descent (SGD) (Yu et al., 2019; Stich, 2019; Wang and Joshi, 2021, 2019), and send model updates periodically to the aggregating server.

Besides communication limitations, the key scalability challenge faced by FL is that the devices (clients) can have highly heterogeneous local datasets and computation speed. The effect of data heterogeneity on the convergence of local-update SGD is analyzed in recent works (Reddi et al., 2021; Haddadpour and Mahdavi, 2019; Khaled et al., 2020; Stich and Karimireddy, 2020; Woodworth et al., 2020; Koloskova et al., 2020; Huo et al., 2020; Zhang et al., 2020; Pathak and Wainwright, 2020; Malinovsky et al., 2020; Sahu et al., 2020) and methods to overcome the adverse effects of data and computational heterogeneity are proposed by Sahu et al. (2020); Wang et al. (2021); Karimireddy et al. (2020), among others.

Partial Client Participation. In practice, only a small fraction of client nodes participate in each training round of FL, which can exacerbate the adverse effects of data heterogeneity. While existing convergence guarantees for full client participation and methods to tackle heterogeneity generalize to partial client participation (Li et al., 2020b), these are limited to *unbiased* client participation, where each client's contribution to the expected global objective optimized in each round is proportional to its dataset size. Horváth and Richtárik (2021) consider a biased client sampling scheme. However, the updates sent by the selected clients are normalized such that the aggregated update is unbiased. To the best of our knowledge, this work is the first to analyze biased client selection in FL through the lens of *selection skew* towards clients with higher local losses that results in a biased aggregated update. Ruan et al. (2020) analyze the convergence

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

with flexible device participation where the aggregated update can be biased, but the effect of selecting clients with higher local losses to the convergence of models trained with FL is not investigated. Another line of work including Nishio and Yonetani (2019) propose to group clients based on hardware and wireless resources to save communication resources.

Client Selection Aware of Local Loss. Adaptive client selection that is cognizant of the training progress of clients is not yet well-understood. Such biased client selection strategies can accelerate error convergence in heterogeneous environments by preferentially selecting clients with higher local loss values, as we show in this paper. This idea has been explored in recent empirical studies (Goetz et al., 2019; Ribero and Vikalo, 2020; Kim et al., 2020; Cho et al., 2020). Goetz et al. (2019) proposed client selection with local loss (benchmarked in our experiments) and Ribero and Vikalo (2020) proposed utilizing the progression of clients' weights. Kim et al. (2020); Cho et al. (2020) also utilizes client loss information with multi-arm bandits for client selection in FL. But these schemes are limited to empirical demonstration without rigorous analysis of how selection skew affects convergence speed. Another relevant line of work (Jiang et al., 2019; Katharopoulos and Fleuret, 2018; Shah et al., 2020; Salehi et al., 2018) employs importance sampling of data to speed-up convergence of centralized SGD. They propose selecting samples with highest loss or gradient norm to perform the next SGD iteration. In contrast, Shah et al. (2020) proposes biased selection of lower loss samples to improve robustness to outliers. Generalizing such strategies to the FL setting is non-trivial due to the distributed and heterogeneous nature of the training data. Concurrent work Fraboni et al. (2021) has proposed a clustered client sampling scheme for FL to reduce the variance of the clients' aggregation weights in FL.

Our Contributions. In this paper, we present the first convergence analysis of FL with biased client selection that is cognizant of the training progress at each client. We prove theoretically that biasing client selection towards clients with higher local losses increases the rate of convergence compared to unbiased client selection. Using this insight, we propose the Power-of-Choice client selection strategy and show that Power-of-Choice yields up to 3× faster convergence with 10% higher test performance than the standard federated averaging with random selection. We also propose communication and computation efficient variants of Power-of-Choice that incur minimal additional resource overhead. In fact, we show that even with 3× less clients participating in each round as compared to random selection, Power-of-Choice gives 2× faster convergence and 5% higher test accuracy.

2 PROBLEM FORMULATION

Consider a cross-device FL setup with total K clients, where client k has local dataset $\mathcal{B}_k$ consisting $|\mathcal{B}_k| = D_k$ data samples. Clients are connected via a central aggregating server and seek to collectively find the model parameter $\mathbf{w}$ that minimizes the empirical risk:

$$F(\mathbf{w}) = \frac{1}{\sum_{k=1}^{K} D_k} \sum_{k=1}^{K} \sum_{\xi \in \mathcal{B}_k} f(\mathbf{w}, \xi) = \sum_{k=1}^{K} p_k F_k(\mathbf{w}) \tag{1}$$

where $f(\mathbf{w}, \xi)$ is the composite loss function for sample ξ and parameter vector $\mathbf{w}$. The term $p_k = D_k / \sum_{k=1}^{K} D_k$ is the fraction of data at the k-th client, and $F_k(\mathbf{w}) = \frac{1}{|\mathcal{B}_k|} \sum_{\xi \in \mathcal{B}_k} f(\mathbf{w}, \xi)$ is the local objective function of client k. In FL, $\mathbf{w}^*$, and $\mathbf{w}_k^*$ for $k = 1, \ldots, K$ that minimize $F(\mathbf{w})$ and $F_k(\mathbf{w})$ respectively can be different from each other. We define $F^* = \min_{\mathbf{w}} F(\mathbf{w}) = F(\mathbf{w}^*)$ and $F_k^* = \min_{\mathbf{w}} F_k(\mathbf{w}) = F_k(\mathbf{w}_k^*)$.

FL with Partial Client Participation. The most common algorithm to solve (1) is *federated averaging* (FedAvg) proposed by McMahan et al. (2017). The algorithm divides the training into communication rounds. At each round, the global server only selects a fraction C of $m = CK$ clients to participate in the training. Each selected/active client performs τ iterations of local SGD (Stich, 2019; Wang and Joshi, 2021; Yu et al., 2019) and sends its locally updated model back to the server. Then, the server updates the global model using the local models and broadcasts the global model to a new set of active clients.

Formally, we index the local SGD iterations with $t \geq 0$. The set of active clients at iteration t is denoted by $\mathcal{S}^{(t)}$. Since active clients performs τ steps of local update, the active set $\mathcal{S}^{(t)}$ also remains constant for every τ iterations. That is, if $(t+1) \bmod \tau = 0$, then $\mathcal{S}^{(t+1)} = \mathcal{S}^{(t+2)} = \cdots = \mathcal{S}^{(t+\tau)}$. Accordingly, the update rule of FedAvg is written as follows:

$$\mathbf{w}_k^{(t+1)} = \begin{cases} \mathbf{w}_k^{(t)} - \eta_t g_k(\mathbf{w}_k^{(t)}, \xi_k^{(t)}) & \text{for } (t+1) \bmod \tau \neq 0 \\ \frac{1}{m} \sum_{j \in \mathcal{S}^{(t)}} \left(\mathbf{w}_j^{(t)} - \eta_t g_j(\mathbf{w}_j^{(t)}, \xi_j^{(t)}) \right) \triangleq \overline{\mathbf{w}}^{(t+1)} & \text{o.w.} \end{cases} \tag{2}$$

where $\mathbf{w}_k^{(t+1)}$ denotes the local model parameters of client k at iteration t, η_t is the learning rate, and $g_k(\mathbf{w}_k^{(t)}, \xi_k^{(t)}) = \frac{1}{b} \sum_{\xi \in \xi_k^{(t)}} \nabla f(\mathbf{w}_k^{(t)}, \xi)$ is the stochastic gradient over mini-batch $\xi_k^{(t)}$ of size b that is randomly sampled from client k's local dataset $\mathcal{B}_k$. Moreover, $\overline{\mathbf{w}}^{(t+1)}$ denotes the global model at server. Although $\overline{\mathbf{w}}^{(t)}$ is only updated every τ iterations, for the purpose

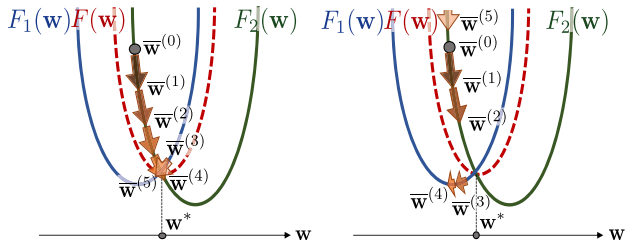


Figure 1: A toy example performing global model updates with quadratic functions $F_1(\mathbf{w})$ and $F_2(\mathbf{w})$ as the local objective, and $F(\mathbf{w}) = (F_1(\mathbf{w}) + F_2(\mathbf{w}))/2$ as the global objective function with global minimum $\mathbf{w}^*$ for different sampling strategies (a) and (b). At each round, a single client is selected to perform local updates; (a): sampling client with larger local loss; (b): sampling client uniformly at random (in the order of 2,2,1,1,2).

of convergence analysis we consider a virtual sequence of $\overline{\mathbf{w}}^{(t)}$ that is updated at each iteration as follows:

$$\overline{\mathbf{w}}^{(t+1)} = \overline{\mathbf{w}}^{(t)} - \eta_t \overline{\mathbf{g}}^{(t)} = \overline{\mathbf{w}}^{(t)} - \frac{\eta_t}{m} \sum_{k \in \mathcal{S}^{(t)}} g_k(\mathbf{w}_k^{(t)}, \xi_k^{(t)}) \tag{3}$$

with $\overline{\mathbf{g}}^{(t)} = \frac{1}{m} \sum_{k \in \mathcal{S}^{(t)}} g_k(\mathbf{w}_k^{(t)}, \xi_k^{(t)})$. Note that in (2) and (3) we do not weight the client models by their dataset fractions p_k because p_k is considered in the client selection scheme used to decide the set $\mathcal{S}^{(t)}$. Our convergence analysis can be generalized to when the global model is a weighted average instead of a simple average of client models, which we show in Appendix E that our convergence analysis also covers the sampling uniformly at random without replacement scheme proposed by Li et al. (2020b). The set $\mathcal{S}^{(t)}$ can be sampled either with or without replacement. For sampling with replacement, we assume that multiple copies of the same client in the set $\mathcal{S}^{(t)}$ behave as different clients, that is, they perform local updates independently.

Client Selection Strategy. To guarantee FedAvg to converge to the stationary points of the objective function (1), most analysis frameworks (Li et al., 2020b; Karimireddy et al., 2020; Wang et al., 2021) consider a strategy that selects the set $\mathcal{S}^{(t)}$ by sampling m clients at random (with replacement) such that client k is selected with probability p_k, the fraction of data at that client. This sampling scheme is *unbiased* since it ensures that in expectation, the update rule (3) is the same as full client participation. Hence, it enjoys the same convergence properties as local-update SGD methods (Stich, 2019; Wang and Joshi, 2021). We denote this unbiased random client selection strategy as π_{rand}. In our work, we consider a class of biased client selection strategies that is cognizant of the global training progress. In the toy example for quadratic functions in Fig. 1(a), we set $\mathcal{S}^{(t+1)} = \arg\max_{k \in [K]} F_k(\overline{\mathbf{w}}^{(t)})$, a single client with the highest local loss at the current global model. In this example, the selection strategy cannot guarantee that (3) equals to the full client participation case in expectation. Nevertheless, the biased selection strategy gives faster convergence to the global minimum than the unbiased selection strategy (random selection) in Fig. 1(b). With this observation, we define a client selection strategy π as a function that maps the current global model $\mathbf{w}$ to a specific selected set of clients $\mathcal{S}(\pi, \mathbf{w})$. Note that we do not restrict π to only be a biased client selection strategy.

3 CONVERGENCE ANALYSIS

In this section we analyze the convergence of federated averaging with partial device participation for *any* client selection strategy π as defined above. This analysis reveals that biased client selection can give faster convergence, albeit at the risk of having a non-vanishing gap between the true optimum $\mathbf{w}^* = \arg\min F(\mathbf{w})$ and $\lim_{t \to \infty} \overline{\mathbf{w}}^{(t)}$. We use this insight in Section 4 to propose an efficient client selection strategy that balances convergence speed and bias as well as an adaptive strategy that modulates the selection bias to gradually decrease the non-vanishing gap.

3.1 Assumptions and Definitions

First we introduce the assumptions and definitions utilized for our convergence analysis.

Assumption 3.1. *F_1, ..., F_k are all $L-$smooth, i.e., for all $\mathbf{v}$ and $\mathbf{w}$, we have $F_k(\mathbf{v}) \leq F_k(\mathbf{w}) + (\mathbf{v} - \mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{L}{2} \|\mathbf{v} - \mathbf{w}\|_2^2$.*

Assumption 3.2. *F_1, ..., F_k are all $\mu-$strongly convex, i.e., for all $\mathbf{v}$ and $\mathbf{w}$, we have $F_k(\mathbf{v}) \geq F_k(\mathbf{w}) + (\mathbf{v} - \mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{\mu}{2} \|\mathbf{v} - \mathbf{w}\|_2^2$.*

Assumption 3.3. *For the mini-batch ξ_k uniformly sampled at random from $\mathcal{B}_k$ from user k, the resulting stochastic gradient is unbiased, that is, $\mathbb{E}[g_k(\mathbf{w}_k, \xi_k)] = \nabla F_k(\mathbf{w}_k)$. Also, the variance of stochastic gradients is bounded: $\mathbb{E}\|g_k(\mathbf{w}_k, \xi_k) - \nabla F_k(\mathbf{w}_k)\|^2 \leq \sigma^2$ for $k = 1, ..., K$.*

Assumption 3.4. *The stochastic gradient's expected squared norm is uniformly bounded, i.e., $\mathbb{E}\|g_k(\mathbf{w}_k, \xi_k)\|^2 \leq G^2$ for $k = 1, ..., K$.*

These assumptions are common in related literature (Stich, 2019; Basu et al., 2019; Li et al., 2020b; Ruan et al., 2020). Next, we introduce two core metrics, *local-global objective gap* and *selection skew*, which feature in the convergence analysis presented in Theorem 3.1.

Definition 3.1 (**Local-Global Objective Gap**). *For the global optimum* $\mathbf{w}^* = \arg\min_{\mathbf{w}} F(\mathbf{w})$ *and local optimum* $\mathbf{w}_k^* = \arg\min_{\mathbf{w}} F_k(\mathbf{w})$ *we define the local-global objective gap as*

$$\Gamma \triangleq F^* - \sum_{k=1}^{K} p_k F_k^* = \sum_{k=1}^{K} p_k (F_k(\mathbf{w}^*) - F_k(\mathbf{w}_k^*)). \quad (4)$$

This definition was first introduced by Li et al. (2020b). Note that $\Gamma \geq 0$ is an inherent property of the local and global objective functions, and it is independent of the client selection strategy. A larger Γ implies higher data heterogeneity. If $\Gamma = 0$ then it implies that the local and global optimal values are consistent, and there is no solution bias due to the client selection strategy (see Theorem 3.1). Next, we define selection skew, which captures the effect of the client selection strategy on the local-global objective gap.

Definition 3.2 (**Selection Skew**). *For any* $k \in \mathcal{S}(\pi, \mathbf{w})$ *we define,*

$$\rho(\mathcal{S}(\pi, \mathbf{w}), \mathbf{w}') = \frac{\mathbb{E}_{\mathcal{S}(\pi,\mathbf{w})}[\frac{1}{m}\sum_{k\in\mathcal{S}(\pi,\mathbf{w})}(F_k(\mathbf{w}') - F_k^*)]}{F(\mathbf{w}') - \sum_{k=1}^{K} p_k F_k^*} \quad (5)$$

which reflects the skew of a client selection strategy π. *The first* $\mathbf{w}$ *in* $\rho(\mathcal{S}(\pi, \mathbf{w}), \mathbf{w}')$ *is the parameter vector that governs the client selection and* $\mathbf{w}'$ *is the point at which* F_k *and* F *in the numerator and denominator respectively are evaluated. Note that* $\mathbb{E}_{\mathcal{S}(\pi,\mathbf{w})}[\cdot]$ *is the expectation over the randomness from the selection strategy* π, *since there can be multiple sets* $\mathcal{S}$ *that* π *can map from a specific* $\mathbf{w}$. *It is trivial to show that* $\rho(\mathcal{S}(\pi, \mathbf{w}), \mathbf{w}') \geq 0$.

Since $\rho(\mathcal{S}(\pi, \mathbf{w}), \mathbf{w}')$ *is a function of versions of the global model* $\mathbf{w}$ *and* $\mathbf{w}'$, *which change during training, we define two related metrics that are independent of* $\mathbf{w}$ *and* $\mathbf{w}'$. *These metrics enable us to obtain a conservative error bound in the convergence analysis.*

$$\overline{\rho} \triangleq \min_{\mathbf{w},\mathbf{w}'} \rho(\mathcal{S}(\pi, \mathbf{w}), \mathbf{w}') \quad (6)$$

$$\widetilde{\rho} \triangleq \max_{\mathbf{w}} \rho(\mathcal{S}(\pi, \mathbf{w}), \mathbf{w}^*) \quad (7)$$

where $\mathbf{w}^* = \arg\min_{\mathbf{w}} F(\mathbf{w})$. *From* (6) *and* (7), *we have* $\overline{\rho} \leq \widetilde{\rho}$ *for any client selection strategy* π.

For the unbiased client selection strategy π_{rand} we have $\rho(\mathcal{S}(\pi_{\text{rand}}, \mathbf{w}), \mathbf{w}') = 1$ for all $\mathbf{w}$ and $\mathbf{w}'$ since the numerator and denominator of (5) become equal, and $\overline{\rho} = \widetilde{\rho} = 1$. For a client selection strategy π that chooses clients with higher $F_k(\mathbf{w})$ more often, $\overline{\rho}$ and $\widetilde{\rho}$ will be larger (and ≥ 1). In the convergence analysis we show that a larger $\overline{\rho}$ implies faster convergence, albeit with a potential error gap, which is proportional to $(\widetilde{\rho}/\overline{\rho} - 1)$. Motivated by this, in Section 4 we present an adaptive client selection strategy that prefers selecting clients with higher loss $F_k(\mathbf{w})$ and achieves faster convergence speed with low solution bias.

3.2 Main Convergence Result

We present the convergence for any client selection strategy π for federated averaging with partial device participation in terms of Γ and selection skew $\overline{\rho}, \widetilde{\rho}$.

Theorem 3.1 (Convergence with Decaying Learning Rate). *Under Assumptions 3.1 to 3.4, for learning rate* $\eta_t = \frac{1}{\mu(t+\gamma)}$ *and* $\gamma = \frac{4L}{\mu}$, *with any client selection strategy* π, *after* T *iterations of federated averaging with partial device participation we have the convergence as:*

$$\mathbb{E}[F(\overline{\mathbf{w}}^{(T)})] - F^* \leq \frac{1}{(T+\gamma)} \left[\frac{4L(32\tau^2 G^2 + \sigma^2/m)}{3\mu^2 \overline{\rho}} + \frac{8L^2\Gamma}{\mu^2} + \frac{L\gamma\|\overline{\mathbf{w}}^{(0)} - \mathbf{w}^*\|^2}{2} \right] + \underbrace{\frac{8L\Gamma}{3\mu}\left(\frac{\widetilde{\rho}}{\overline{\rho}} - 1\right)}_{\text{Non-vanishing bias}, Q(\overline{\rho},\widetilde{\rho})} \quad (8)$$

To the best of our knowledge, Theorem 3.1 provides the first convergence analysis of federated averaging with a biased client selection strategy π in the lens of *selection skew*. We show the results for the fixed learning rate case in Appendix A. The proof for Theorem 3.1 is presented in Appendix C. Our convergence result is a general analysis that encompasses random selection and any selection strategy π that is cognizant of the training progress. In the following paragraphs, we discuss the effect of the two terms in the RHS of (8) in detail.

Large $\overline{\rho}$ and Faster Convergence. A key insight from Theorem 3.1 is that a larger selection skew $\overline{\rho}$ results in faster convergence at the rate $\mathcal{O}(\frac{1}{T\overline{\rho}})$ as can be seen in the first term in the RHS of (8). This theoretically proves that selecting clients with higher local losses can improve the convergence rate of federated averaging with partial device participation. Since we obtain $\overline{\rho}$ by taking a minimum of the selection skew $\rho(\mathcal{S}(\pi, \mathbf{w}), \mathbf{w}')$ over $\mathbf{w}, \mathbf{w}'$, this is a conservative bound on the true convergence rate. In practice, since the selection skew $\rho(\mathcal{S}(\pi, \mathbf{w}), \mathbf{w}')$ changes during training depending on the current global model $\mathbf{w}$ and the local models $\mathbf{w}'$, the true convergence rate can be improved by a factor larger than and at least equal to $\overline{\rho}$.

Non-vanishing Bias Term. The second term $Q(\overline{\rho}, \widetilde{\rho}) = \frac{8L\Gamma}{3\mu}\left(\frac{\widetilde{\rho}}{\overline{\rho}} - 1\right)$ in the RHS of (8) denotes the solution bias, which is dependent on the selection strategy. By the definitions of $\overline{\rho}$ and $\widetilde{\rho}$, it follows that $\widetilde{\rho} \geq \overline{\rho}$,

which implies that $Q(\overline{\rho}, \widetilde{\rho}) \geq 0$. For an unbiased selection strategy, we have $\overline{\rho} = \widetilde{\rho} = 1$, $Q(\overline{\rho}, \widetilde{\rho}) = 0$, and hence (8) recovers the previous bound for unbiased selection strategy (Li et al., 2020b). For $\overline{\rho} > 1$, while we gain faster convergence rate by a factor of $\overline{\rho}$, we cannot guarantee $Q(\overline{\rho}, \widetilde{\rho}) = 0$. Thus, there is a trade-off between the convergence speed and the solution bias. In the experimental results, we show that even with biased selection strategies, the term $\frac{\widetilde{\rho}}{\overline{\rho}} - 1$ in $Q(\overline{\rho}, \widetilde{\rho})$ can be close to 0, and $Q(\overline{\rho}, \widetilde{\rho})$ has a negligible effect on the final error floor. We also show that by adaptively modulating the selection skew in the client selection strategy, we can gradually reduce the solution bias $Q(\overline{\rho}, \widetilde{\rho})$ to 0.

4 PROPOSED POWER-OF-CHOICE CLIENT SELECTION STRATEGY

In Section 4, we discover that a selection strategy π that prefers clients with larger local loss will result in a larger $\overline{\rho}$, yielding faster convergence. With this insight, a naive client selection strategy can be choosing the clients with highest local loss $F_k(\mathbf{w})$. However, a larger selection skew $\overline{\rho}$ may result in a larger $\overline{\rho}/\widetilde{\rho}$, i.e., a larger non-vanishing error term. Moreover, to find the current local loss $F_k(\mathbf{w})$, it requires sending the current global model to all K clients and having them evaluate $F_k(\mathbf{w})$ and sending it back. Such additional communication and computation cost can be prohibitively high due to the typically large number of clients and limited communication and computation capabilities.

We leverage such trade-offs amongst convergence speed, solution bias, and communication/computation cost by our proposed POWER-OF-CHOICE client selection strategy[1]. In POWER-OF-CHOICE (denoted by $\pi_{\text{pow-d}}$), the server chooses the active client set $\mathcal{S}^{(t)}$ as follows:

1. **Sample the Candidate Client Set.** The central server samples a candidate set $\mathcal{A}$ of d ($m \leq d \leq K$) clients without replacement such that client k is chosen with probability p_k, the fraction of data at the k-th client for $k = 1, \dots K$.

2. **Estimate Local Losses.** The server sends the current global model $\overline{\mathbf{w}}^{(t)}$ to the clients in set $\mathcal{A}$, and these clients compute and send back to the central server their local loss $F_k(\overline{\mathbf{w}}^{(t)})$.

3. **Select Highest Loss Clients.** From the candidate set $\mathcal{A}$, the central server constructs the active client set $\mathcal{S}^{(t)}$ by selecting $m = \max(CK, 1)$ clients with the largest values $F_k(\overline{\mathbf{w}}^{(t)})$, with ties broken at random. These $\mathcal{S}^{(t)}$ clients participate in the training during the next round, consisting of iterations $t+1, t+2, \dots t+\tau$.

Variations of $\pi_{\text{pow-d}}$. The three steps of $\pi_{\text{pow-d}}$ can be flexibly modified to reflect practical considerations. For example, intermittent client availability can be accounted for in step 1 by constructing set $\mathcal{A}$ only from the set of available clients in that round. We demonstrate the performance of $\pi_{\text{pow-d}}$ with intermittent client availability in Appendix G.3. The local computation and server-client communication cost in step 2 can be reduced or eliminated by the following proposed variants of $\pi_{\text{pow-d}}$: $\pi_{\text{cpow-d}}$ and $\pi_{\text{rpow-d}}$ (see Appendix F for their pseudo-codes). We also diminish the solution bias while still gaining convergence speed by the proposed variant $\pi_{\text{adapow-d}}$ below.

- **Computation-efficient Variant** $\pi_{\text{cpow-d}}$: Saving computation cost, instead of evaluating $F_k(\mathbf{w})$ with the entire local dataset $\mathcal{B}_k$, we use an estimate $\sum_{\xi \in \widehat{\xi}_k} f(\mathbf{w}, \xi)/|\widehat{\xi}_k|$, where $\widehat{\xi}_k$ is the mini-batch of b samples sampled uniformly at random from $\mathcal{B}_k$.

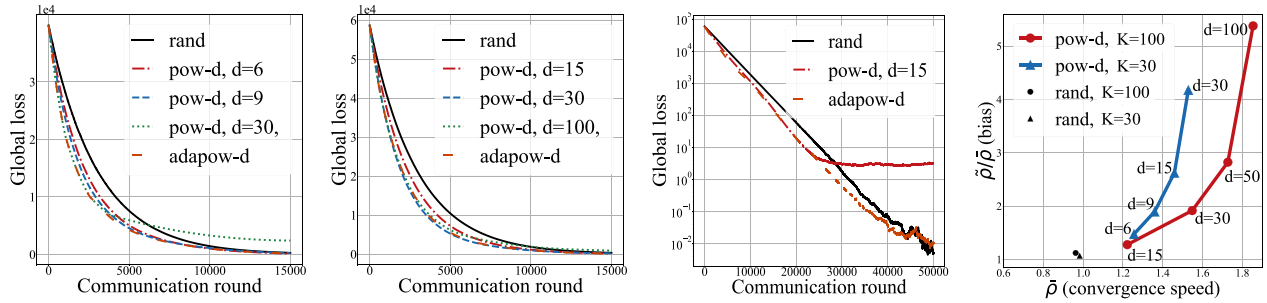
- **Communication- and Computation-efficient Variant** $\pi_{\text{rpow-d}}$: Saving both computation and communication cost, selected clients for each round send their accumulated averaged loss over local iterations, i.e., $\frac{1}{\tau |\xi_k^{(l)}|} \sum_{l=t-\tau+1}^{t} \sum_{\xi \in \xi_k^{(l)}} f(\mathbf{w}_k^{(l)}, \xi)$ when they send their local models to the server. The server uses the latest received value from each client as a proxy for $F_k(\mathbf{w})$ to select clients. For clients not yet selected, the latest value is set to ∞.

- **Adaptive Selection Skew Variant** $\pi_{\text{adapow-d}}$: To minimize the non-vanishing bias term in Theorem 3.1 while simultaneously gaining the benefit of convergence speed from $\overline{\rho}$, we gradually reduce d until $d = m$[2]. This enables convergence speed up in the initial training phase, while eventually diminishing the non-vanishing bias term when $d = m$. Which d to start with and how gradually we decrease d to m is flexible, analogous to setting the hyper-parameters.

  Intuitively, a large fixed d in $\pi_{\text{adapow-d}}$ allows the global model to move fast towards the global optimum, but at the same time can prevent the global model from actually converging to the optimum due to the non-vanishing bias. Hence, gradually decreasing d for the convergence of the global model to the optimum can be compared to the effect of modulating the learning rate through the training process.

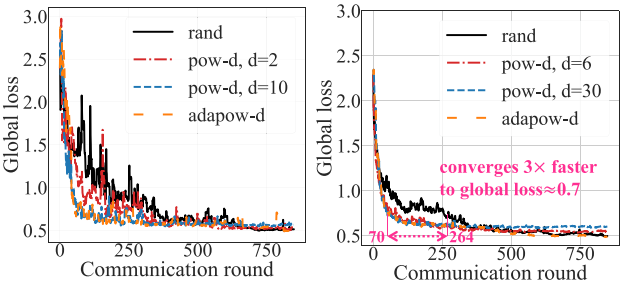
[1] POWER-OF-CHOICE is based on the power of d choices load balancing strategy (Mitzenmacher, 1996), which is extensively used in queueing systems.

[2] $d = m$ makes the POWER-OF-CHOICE strategy equivalent to an unbiased sampling strategy, which has zero non-vanishing bias.



(a) $K = 30$ (b) $K = 100$ (c) $K = 100$ (log-scale) (d) $\overline{\rho}$ vs $\overline{\rho}/\widetilde{\rho}$

Figure 2: (a)-(c): Global loss of π_{rand}, $\pi_{\text{pow-d}}$, and $\pi_{\text{adapow-d}}$ for the quadratic simulations with $C = 0.1$. $\pi_{\text{pow-d}}$ convergences faster than π_{rand} and as convergence speed increases the solution bias also increases for $\pi_{\text{pow-d}}$. $\pi_{\text{adapow-d}}$ is able to eliminate this solution bias while gaining nearly identical convergence speed to $\pi_{\text{pow-d}}$; (d): Estimated theoretical values $\overline{\rho}$ and $\overline{\rho}/\widetilde{\rho}$ for the quadratic simulations. The theoretical values of convergence speed ($\overline{\rho}$) and bias ($\widetilde{\rho}/\overline{\rho}$) are consistent with the results shown in Fig. 2(a)-(b) for π_{rand} and $\pi_{\text{pow-d}}$.



(a) $m = 1$ (b) $m = 3$

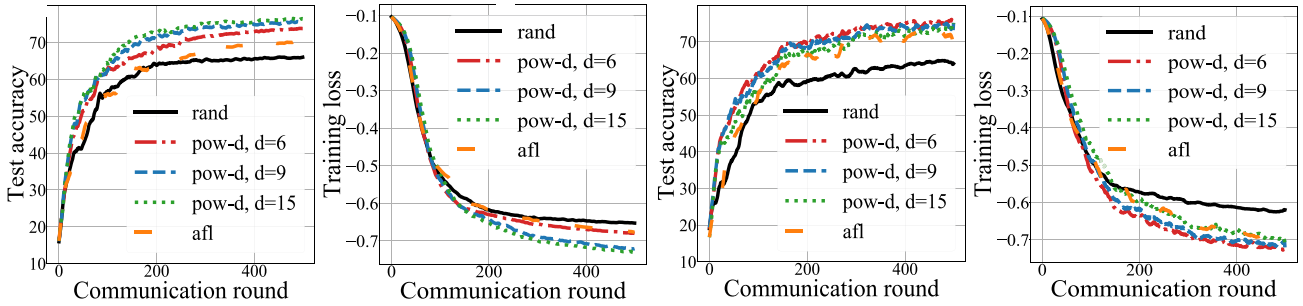
Figure 3: Logistic regression loss on `Synthetic`(1,1) for π_{rand}, $\pi_{\text{pow-d}}$, and $\pi_{\text{adapow-d}}$, with $d \in \{2m, 10m\}$, $K = 30$ and $m \in \{1, 3\}$. $\pi_{\text{pow-d}}$ converges approximately $3\times$ faster for $d = 10m$ and $2\times$ faster for $d = 2m$ than π_{rand} to the global loss ≈ 0.7. $\pi_{\text{adapow-d}}$ is able to converge to the minimum global loss $3\times$ faster than π_{rand}.

Selection Skew in Power-of-Choice. The size d of the candidate client set $\mathcal{A}$ controls the trade-off between convergence speed and solution bias. With $d = m$, Power-of-Choice becomes random client sampling without replacement in proportion of p_k. As d increases, the selection skew $\overline{\rho}$ increases, giving faster error convergence at the risk of a higher error floor. However, note that the convergence analysis replaces $\rho(\mathbf{w}, \mathbf{w}')$ with $\overline{\rho}$ to get a conservative error bound. In practice, the convergence speed and the solution bias is dictated by $\rho(\overline{\mathbf{w}}^{(\tau\lfloor t/\tau \rfloor)}, \overline{\mathbf{w}}^{(t)})$ which changes during training. With $\pi_{\text{pow-d}}$, which is biased towards higher local losses, we expect the selection skew $\rho(\mathbf{w}, \mathbf{w}')$ to reduce through the course of training. We conjecture that this is the reason for $\pi_{\text{pow-d}}$ giving faster convergence as well as little or no solution bias in our experiments for DNNs (non-convex) presented in Section 5.

5 EXPERIMENTAL RESULTS

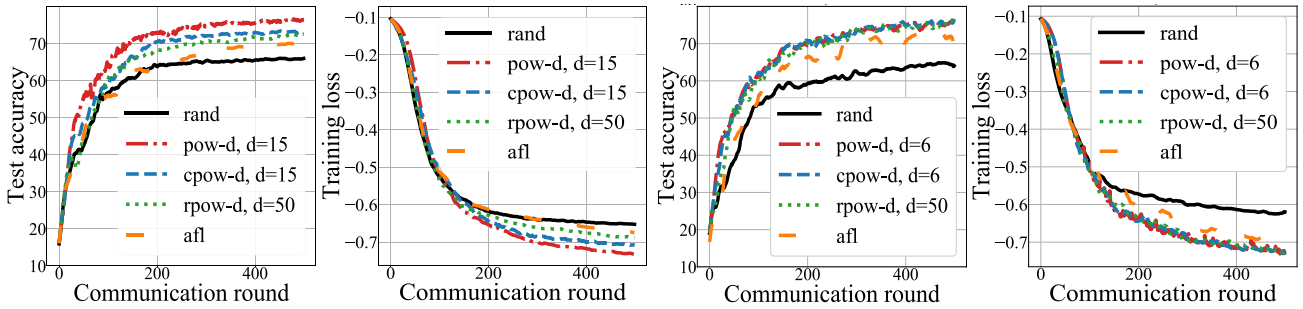
We evaluate our proposed $\pi_{\text{pow-d}}$ and its practical variants $\pi_{\text{cpow-d}}$, $\pi_{\text{rpow-d}}$, and $\pi_{\text{adapow-d}}$ by five sets of experiments: (1) quadratic optimization, (2) logistic regression on a synthetic federated dataset, `Synthetic`(1,1) (Sahu et al., 2020), (3) MLP for image classification on a non-iid partitioned FMNIST dataset (Xiao et al., 2017), (4) CNN for image classification on a non-iid partitioned CIFAR10 dataset (Krizhevsky et al., 2009), and (5) MLP for sentiment classification on the Twitter dataset (Go et al., 2009). We also benchmark the selection strategy proposed by Goetz et al. (2019), active FL, denoted as π_{afl}. Details of the experimental setup are provided in Appendix F, and the code for all experiments are shared in the supplementary material. To validate consistency in our results, we present additional experiments with MLP trained on a non-iid partitioned EMNIST (Cohen et al., 2017) dataset sorted by digits for image classification with $K = 500$ clients in Appendix G.4. Moreover we show the effect of mini-batch size and local epochs on the performance of Power-of-Choice in Appendix G.6.

Quadratic and Synthetic Simulations. In Fig. 2(a), even with few clients ($K = 30$), $\pi_{\text{pow-d}}$ converges faster than π_{rand} with nearly negligible solution bias for small d. The convergence speed increases with the increase in d, at the cost of higher error floor due to the solution bias. For $K = 100$ in Fig. 2(b), $\pi_{\text{pow-d}}$ shows convergence speed-up as with $K = 30$, but the solution bias is smaller. Fig. 2(d) shows the theoretical values $\overline{\rho}$ and $\widetilde{\rho}/\overline{\rho}$ which represents the convergence speed and the solution bias respectively in our convergence analysis. Compared to π_{rand}, $\pi_{\text{pow-d}}$ has higher $\overline{\rho}$ for all d implying higher convergence speed than π_{rand}. By varying d we can span different points on



(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

Figure 4: Test accuracy and training loss for $\pi_{\text{pow-d}}$ for varying d with $K = 100$, $C = 0.03$ for training MLP on FMNIST. For both small and large α, $\pi_{\text{pow-d}}$ achieves at least 10% test accuracy improvement than π_{rand} and the training loss converges at a much higher rate than π_{rand}.



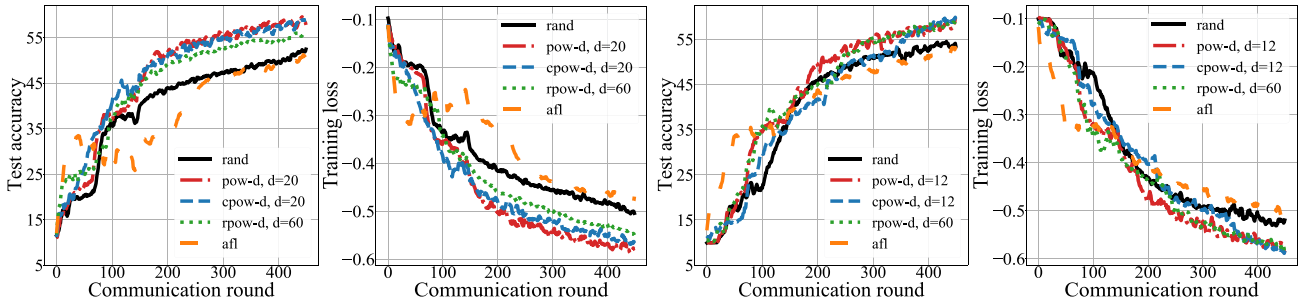
(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

Figure 5: Test accuracy and training loss for communication- and computation-efficient $\pi_{\text{cpow-d}}$ and $\pi_{\text{rpow-d}}$ with $K = 100$, $C = 0.03$ for training MLP on FMNIST. $\pi_{\text{rpow-d}}$ which requires no additional communication and minor computation, yields higher test accuracy than π_{rand} and π_{afl}.

the trade-off between the convergence speed and bias. For $d = 15$ and $K = 100$, $\widetilde{\rho}/\overline{\rho}$ of $\pi_{\text{pow-d}}$ and π_{rand} are approximately identical, but $\pi_{\text{pow-d}}$ has higher $\overline{\rho}$, implying that $\pi_{\text{pow-d}}$ can yield higher convergence speed with negligible solution bias. In Appendix G.1, we present the clients' selected frequency ratio for $\pi_{\text{pow-d}}$ and π_{rand} which gives novel insights regarding the difference between the two strategies. For the synthetic dataset simulations, we present the global losses in Fig. 3 for π_{rand} and $\pi_{\text{pow-d}}$ for different d and m. We show that $\pi_{\text{pow-d}}$ converges approximately $3\times$ faster to the global loss ≈ 0.7 than π_{rand} when $d = 10m$, with a slightly higher error floor. Even with $d = 2m$, we get $2\times$ faster convergence to global loss ≈ 0.7 than π_{rand}.

Elimination of Selection Skew. For $\pi_{\text{pow-d}}$, the selection skew is the trade-off for the convergence speed gain. We eliminate this selection skew while maintaining the benefit of convergence speed with $\pi_{\text{adapow-d}}$. In Fig. 2(a)-(b), $\pi_{\text{adapow-d}}$ shows a convergence speed similar to $\pi_{\text{pow-d}}$, $d = K$, but has no selection skew, converging to the same minimum as π_{rand} (see Fig. 2(c)). In Fig. 3, $\pi_{\text{adapow-d}}$ again shows a convergence speed similar to $\pi_{\text{pow-d}}$, $d = 10m$, but has no adversarial selection skew. In fact, $\pi_{\text{adapow-d}}$ converges to the minimum global loss value at least $3\times$ faster than π_{rand}. Hence $\pi_{\text{adapow-d}}$ gains the benefit of both worlds from biased client selection: convergence speed and elimination of selection skew.

Performance of $\pi_{\text{pow-d}}$. As elaborated in Appendix F, α determines the data heterogeneity across clients (i.e., dataset size and distribution discrepancies across the clients). Smaller α indicates larger data heterogeneity. In Fig. 4, we present the test accuracy and training losses for $\pi_{\text{pow-d}}$ and π_{rand} for the FMNIST experiments with $\alpha = 0.3$ and $\alpha = 2$. We can see that $\pi_{\text{pow-d}}$ achieves approximately 10% and 5% higher test accuracy than π_{rand} and π_{afl} respectively for both $\alpha = 2$ and $\alpha = 0.3$. For higher α, larger d performs better than smaller d. Fig. 4(a) shows that this performance improvement due to the increase of d eventually converges. For smaller α, as in Fig. 4(b), smaller $d = 6$ performs better than larger d which shows that too much solution bias is adversarial to the performance in the presence of large data heterogeneity. Moreover note that the adversarial solution bias is not present in the non-convex DNN experiments.



(a) Test accuracy and training loss for $\alpha = 2$

(b) Test accuracy and training loss for $\alpha = 0.3$

Figure 6: Test accuracy and training loss for communication- and computation-efficient $\pi_{\text{cpow-d}}$ and $\pi_{\text{rpow-d}}$ with $K = 100$, $C = 0.09$ for training CNN on CIFAR10. $\pi_{\text{rpow-d}}$ which requires no additional communication and minor computation, yields higher test accuracy than π_{rand} and π_{afl}.

Table 1: Comparison of R_{60}, t_{comp} (sec), and test accuracy (%) with $\alpha = 0.3$ for training MLP with FMNIST. In the parentheses we show the ratio of each value with that for π_{rand} with $C = 0.1$.

	$C = 0.1$	$C = 0.03$				
	rand	rand	pow-d, $d = 6$	cpow-d, $d = 6$	rpow-d, $d = 50$	afl
R_{60}	172	234(1.36)	89(0.52)	**80 (0.47)**	98(0.57)	121(0.70)
t_{comp}	0.43	0.36(0.85)	0.48(1.13)	0.38(0.88)	**0.37 (0.85)**	0.36(0.84)
Test Acc.	71.21±2.41	64.87±1.97	76.47±0.87	**76.63**±0.79	76.56±1.00	73.28±1.05

Performance of the Communication- and Computation-Efficient variants. Next, we evaluate $\pi_{\text{cpow-d}}$ and $\pi_{\text{rpow-d}}$ which were introduced in Section 4. In Fig. 5, we show for the FMNIST experiments that for $\alpha = 2$, $\pi_{\text{rpow-d}}$ and $\pi_{\text{cpow-d}}$ each yields approximately 5% and 6% higher accuracy than π_{rand}, but both yield lower accuracy than $\pi_{\text{pow-d}}$ that utilizes the highest computation and communication resources. For $\alpha = 0.3$, $\pi_{\text{cpow-d}}$ and $\pi_{\text{rpow-d}}$ perform as well as $\pi_{\text{pow-d}}$ and give a 10% accuracy improvement over π_{rand}. Moreover, $\pi_{\text{pow-d}}$, $\pi_{\text{rpow-d}}$ and $\pi_{\text{cpow-d}}$ all have higher accuracy and faster convergence than π_{afl}. In Fig. 6 we show that the results for the CIFAR10 experiments are consistent with the results for FMNIST in Fig. 5 in terms of the performance of $\pi_{\text{pow-d}}$, $\pi_{\text{cpow-d}}$, and $\pi_{\text{rpow-d}}$ over π_{rand} and π_{afl}. Moreover in Fig. 7, we demonstrate that all Power-of-Choice strategies outperform in test accuracy than that of π_{rand} and π_{afl} for the Sent140 experiment.

Communication Efficiency. We evaluate the communication and computation efficiency of Power-of-Choice by comparing different strategies in terms of R_{60}, the number of communication rounds required to reach test accuracy 60%, and t_{comp}, the average computation time (in seconds) spent per round. The computation time includes the time taken by the central server to select the clients (including the computation time for the d clients to compute their local loss values) and the time taken by selected clients to perform local updates. In Table 1, for the

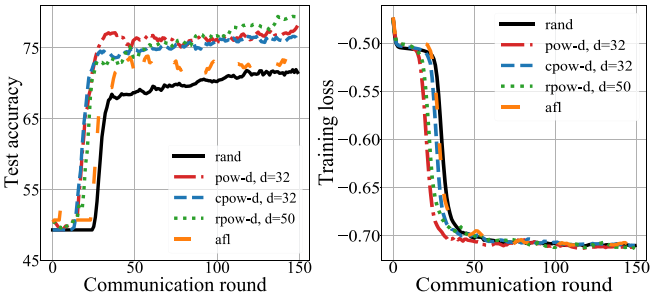


Figure 7: Test accuracy and training loss for communication- and computation-efficient $\pi_{\text{cpow-d}}$ and $\pi_{\text{rpow-d}}$ with $K = 314$, $m = 8$ for training MLP on Sent140. $\pi_{\text{rpow-d}}$ which requires no additional communication and minor computation, yields higher test accuracy than π_{rand} and π_{afl}.

FMNIST experiment with only $C = 0.03$ fraction of clients, $\pi_{\text{pow-d}}$, $\pi_{\text{cpow-d}}$, and $\pi_{\text{rpow-d}}$ have about 5% higher test accuracy than $(\pi_{\text{rand}}, C = 0.1)$. The R_{60} for $\pi_{\text{pow-d}}$, $\pi_{\text{cpow-d}}$, $\pi_{\text{rpow-d}}$ is 0.52, 0.47, 0.57 times that of $(\pi_{\text{rand}}, C = 0.1)$ respectively. This implies that even for $\pi_{\text{rpow-d}}$ which does not incur any additional communication cost for client selection, we can get a 2× reduction in the number of communication rounds using 1/3 of clients compared to $(\pi_{\text{rand}}, C = 0.1)$ and still get higher test accuracy performance. Note that the computation time t_{comp} for $\pi_{\text{cpow-d}}$ and $\pi_{\text{rpow-d}}$ with $C = 0.03$ is smaller than that of π_{rand} with $C = 0.1$. In Appendix G.2, we show that the results

for $\alpha = 2$ are consistent with the $\alpha = 0.3$ case shown in Table 1. In Appendix G.5, we show that for $C = 0.1$, the results are consistent with the $C = 0.03$ case.

6 CONCLUDING REMARKS

In this work, we present the convergence guarantees for FL with partial device participation with any biased client selection strategy. We show that biasing client selection can speed up the convergence at the rate $\mathcal{O}(\frac{1}{T\overline{\rho}})$ where $\overline{\rho}$ is the selection skew towards clients with higher local losses. From this insight, we propose the adaptive client selection strategy Power-of-Choice. Experiments on natural image and language datasets validate that Power-of-Choice yields 3× faster convergence and 10% higher test accuracy than the baseline federated averaging with random selection. Even with using fewer clients than random selection, Power-of-Choice converges 2× faster with high test performance. An interesting future direction is to improve the fairness (Li et al., 2020a; Yu et al., 2020; Lyu et al., 2020; Mohri et al., 2019) and robustness (Pillutla et al., 2019) of Power-of-Choice to use a different metric such as the clipped loss or the q-fair loss proposed by Li et al. (2020a) instead of $F_k(\mathbf{w})$.

Acknowledgements

This research was generously supported in part by NSF grants CCF-1850029, CCF-2045694, CCF-2107024, CCF-2112471, and the Doctoral Study Abroad Scholarship from the Ministry of Education of South Korea (Yae Jee Cho). We thank the reviewers for their constructive suggestions.

References

Debraj Basu, Deepesh Data, Can Karakus, and Suhas Diggavi. Qsparse-local-sgd: Distributed sgd with quantization, sparsification, and local computations. In *Advances in Neural Information Processing Systems*, pages 14695–14706, 2019.

Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konecny, Stefano Mazzocchi, H. Brendan McMahan, Timon Van Overveldt, David Petrou, Daniel Ramage, and Jason Roselander. Towards Federated Learning at Scale: System Design. *SysML*, April 2019. URL https://www.sysml.cc/doc/2019/193.pdf.

Yae Jee Cho, Samarth Gupta, Gauri Joshi, and Osman Yağan. Bandit-based communication-efficient client selection strategies for federated learning. In *2020 54th Asilomar Conference on Signals, Systems, and Computers*, pages 1066–1069. IEEE, 2020.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of MNIST to handwritten letters. *arXiv preprint arXiv:1702.05373*, 2017.

Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao, Marc'Aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, and Andrew Y. Ng. Large scale distributed deep networks. In *Proceedings of the International Conference on Neural Information Processing Systems*, pages 1223–1231, 2012.

Yann Fraboni, Richard Vidal, Laetitia Kameni, and Marco Lorenzi. Clustered sampling: Low-variance and improved representativity for clients selection in federated learning. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 2009. URL https://www-cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf.

Jack Goetz, Kshitiz Malik, Duc Bui, Seungwhan Moon, Honglei Liu, and Anuj Kumar. Active federated learning. *arXiv preprint arXiv:1909.12641*, 2019.

Farzin Haddadpour and Mehrdad Mahdavi. On the convergence of local descent methods in federated learning. *arXiv preprint arXiv:1910.14425*, 2019.

Samuel Horváth and Peter Richtárik. A better alternative to error feedback for communication-efficient distributed learning. In *International Conference on Learning Representations (ICLR)*, 2021.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. In *International Workshop on Federated Learning for User Privacy and Data Confidentiality in Conjunction with NeurIPS 2019 (FL-NeurIPS'19)*, December 2019.

Zhouyuan Huo, Qian Yang, Bin Gu, Lawrence Carin, and Heng Huang. Faster on-device training using new federated momentum algorithm. *arXiv preprint arXiv:2002.02090*, 2020.

Angela H. Jiang, Daniel L. K. Wong, Giulio Zhou, David G. Andersen, Jeffrey Dean, Gregory R. Ganger, Gauri Joshi, Michael Kaminksy, Michael Kozuch, Zachary C. Lipton, and Padmanabhan Pillai. Accelerating deep learning by focusing on the biggest losers. *arXiv preprint arXiv:1910.00762*, October 2019. URL https://arxiv.org/abs/1910.00762.

Peter Kairouz, H. Brendan McMahan, Brendan Avent, and Aurelien Bellet et. al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J Reddi, Sebastian U Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for on-device federated learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020.

A. Katharopoulos and F. Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pages 2525–2534, 2018.

A Khaled, K Mishchenko, and P Richtárik. Tighter theory for local SGD on identical and heterogeneous data. In *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS 2020)*, 2020.

Taehyeon Kim, Sangmin Bae, Jin woo Lee, and Seyoung Yun. Accurate and fast federated learning via combinatorial multi-armed bandits. *arXiv preprint arXiv:2012.03270*, Dec 2020.

Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian U Stich. A unified theory of decentralized SGD with changing topology and local updates. In *Proceedings of 37th International Conference on Machine Learning*, 2020.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Learning multiple layers of features from tiny images. *CIFAR-10 (Canadian Institute for Advanced Research)*, 2009. URL http://www.cs.toronto.edu/~kriz/cifar.html.

Tian Li, Maziar Sanjabi, and Virginia Smith. Fair resource allocation in federated learning. In *International Conference on Learning Representations (ICLR)*, 2020a.

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 (ICLR)*, July 2020b. URL https://arxiv.org/abs/1907.02189.

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*, May 2020.

Grigory Malinovsky, Dmitry Kovalev, Elnur Gasanov, Laurent Condat, and Peter Richtárik. From local SGD to local fixed point methods for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agøura y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. *International Conference on Artificial Intelligenece and Statistics (AISTATS)*, April 2017. URL https://arxiv.org/abs/1602.05629.

M. Mitzenmacher. *The power of two choices in randomized load balancing*. PhD thesis, University of California Berkeley, CA, 1996.

Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4615–4625, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

Takayuki Nishio and Ryo Yonetani. Client selection for federated learning with heterogeneous resources in mobile edge. In *IEEE International Conference on Communications*, pages 1–7, May 2019.

Reese Pathak and Martin J Wainwright. FedSplit: An algorithmic framework for fast federated optimization. In *Advances in Neural Information Processing Systems*, 2020.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. URL http://www.aclweb.org/anthology/D14-1162.

Krishna Pillutla, Sham M. Kakade, and Zaid Harchaoui. Robust aggregation for federated learning. *arXiv preprint 1912.13445*, 2019. URL https://arxiv.org/abs/1912.13445.

Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations (ICLR)*, 2021.

Mónica Ribero and Haris Vikalo. Communication-efficient federated learning via optimal client sampling. *arXiv preprint arXiv:2007.15197*, 2020.

Yichen Ruan, Xiaoxi Zhang, Shu-Che Liang, and Carlee Joe-Wong. Towards flexible device participation in federated learning for non-iid data. In *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS 2020)*, 2020.

Anit Kumar Sahu, Tian Li, Maziar Sanjabi, Manzil Zaheer, Ameet Talwalkar, and Virginia Smith. Federated optimization for heterogeneous networks. In *Proceedings of the 3rd MLSys Conference*, January 2020.

Farnood Salehi, Patrick Thiran, and Elisa Celis. Coordinate descent with bandit sampling. In *Advances in Neural Information*

Processing Systems 31, pages 9247–9257, 2018. URL http://papers.nips.cc/paper/8137-coordinate-descent-with-bandit-sampling.pdf.

Vatsal Shah, Xiaoxia Wu, and Sujay Sanghavi. Choosing the sample with lowest loss makes sgd robust. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS 2020)*, 2020. URL https://arxiv.org/abs/2001.03316.

Sebastian U Stich. Local SGD converges fast and communicates little. In *International Conference on Learning Representations (ICLR)*, 2019.

Sebastian U Stich and Sai Praneeth Karimireddy. The error-feedback framework: Better rates for SGD with delayed gradients and compressed communication. *Journal of Machine Learning Research (JMLR)*, 2020.

Jianyu Wang and Gauri Joshi. Adaptive Communication Strategies for Best Error-Runtime Trade-offs in Communication-Efficient Distributed SGD. In *Proceedings of the SysML Conference*, April 2019. URL https://arxiv.org/abs/1810.08313.

Jianyu Wang and Gauri Joshi. Cooperative SGD: A unified framework for the design and analysis of communication-efficient SGD algorithms. *Journal of Machine Learning Research (JMLR)*, 2021. URL https://arxiv.org/abs/1808.07576.

Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor. Tackling the Objective Inconsistency Problem in Heterogeneous Federated Optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. URL https://arxiv.org/abs/2007.07481.

Blake Woodworth, Kumar Kshitij Patel, Sebastian U Stich, Zhen Dai, Brian Bullins, H Brendan McMahan, Ohad Shamir, and Nathan Srebro. Is local SGD better than minibatch SGD? In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *https://arxiv.org/abs/1708.07747*, aug 2017.

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. *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, page 393–399, 2020.

Hao Yu, Sen Yang, and Shenghuo Zhu. Parallel restarted SGD for non-convex optimization with faster convergence and less communication. *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*, 2019.

Xinwei Zhang, Mingyi Hong, Sairaj Dhople, Wotao Yin, and Yang Liu. FedPD: A federated learning framework with optimal rates and adaptivity to non-IID data. *Asilomar Conference on Signals, Systems, and Cmputers.*, 2020.

Supplementary Material: Towards Understanding Biased Client Selection in Federated Learning

A Additional Theorem

Theorem A.1 (Convergence with Fixed Learning Rate)**.** *Under Assumptions 3.1 to 3.4, a fixed learning rate* $\eta \leq \min\{\frac{1}{2\mu B}, \frac{1}{4L}\}$ *where* $B = 1 + \frac{3\overline{\rho}}{8}$*, and any client selection strategy* π *as defined above, the error after* T *iterations of federated averaging with partial device participation satisfies*

$$
\begin{aligned}
&F(\overline{\mathbf{w}}^{(T)}) - F^* \\
&\leq \underbrace{\frac{L}{\mu}\left[1 - \eta\mu\left(1 + \frac{3\overline{\rho}}{8}\right)\right]^T \left(F(\overline{\mathbf{w}}^{(0)}) - F^* - \frac{4\left[\eta\left(32\tau^2G^2 + \frac{\sigma^2}{m} + 6\overline{\rho}L\Gamma\right) + 2\Gamma(\widetilde{\rho} - \overline{\rho})\right]}{8 + 3\overline{\rho}}\right)}_{\textit{Vanishing Term}} \\
&+ \underbrace{\frac{4L\eta\left(32\tau^2G^2 + \frac{\sigma^2}{m} + 6\overline{\rho}L\Gamma\right)}{\mu(8 + 3\overline{\rho})} + \frac{8L\Gamma(\widetilde{\rho} - \overline{\rho})}{\mu(8 + 3\overline{\rho})}}_{\textit{Non-vanishing bias}}
\end{aligned} \tag{9}
$$

As $T \to \infty$ the first term in (9) goes to 0 and the second term becomes the bias term for the fixed learning rate case. For a small η, we have that the bias term for the fixed learning rate case in Theorem A.1 is upper bounded by $\frac{8L\Gamma}{3\mu}\left(\frac{\widetilde{\rho}}{\overline{\rho}} - 1\right)$ which is identical to the decaying-learning rate case. The proof is presented in Appendix D.

B Preliminaries for Proof of Theorem 3.1 and Theorem A.1

We present the preliminary lemmas used for proof of Theorem 3.1 and Theorem A.1. We will denote the expectation over the sampling random source $\mathcal{S}^{(t)}$ as $\mathbb{E}_{\mathcal{S}^{(t)}}$ and the expectation over all the random sources as $\mathbb{E}$.

Lemma B.1. *Suppose* F_k *is* $L-$*smooth with global minimum at* $\mathbf{w}_k^*$*, then for any* $\mathbf{w}_k$ *in the domain of* F_k*, we have that*

$$\|\nabla F_k(\mathbf{w}_k)\|^2 \leq 2L(F_k(\mathbf{w}_k) - F_k(\mathbf{w}_k^*)) \tag{10}$$

Proof.

$$F_k(\mathbf{w}_k) - F_k(\mathbf{w}_k^*) - \langle \nabla F_k(\mathbf{w}_k^*), \mathbf{w}_k - \mathbf{w}_k^* \rangle \geq \frac{1}{2L}\|\nabla F_k(\mathbf{w}_k) - \nabla F_k(\mathbf{w}_k^*)\|^2 \tag{11}$$

$$F_k(\mathbf{w}_k) - F_k(\mathbf{w}_k^*) \geq \frac{1}{2L}\|\nabla F_k(\mathbf{w}_k)\|^2 \tag{12}$$

□

Lemma B.2 (Expected average discrepancy between $\overline{\mathbf{w}}^{(t)}$ and $\mathbf{w}_k^{(t)}$ for $k \in \mathcal{S}^{(t)}$)**.**

$$\frac{1}{m}\mathbb{E}[\sum_{k \in \mathcal{S}^{(t)}} \|\overline{\mathbf{w}}^{(t)} - \mathbf{w}_k^{(t)}\|^2] \leq 16\eta_t^2\tau^2G^2 \tag{13}$$

Proof.

$$\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}}\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}_k^{(t)}\|^2=\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}}\|\frac{1}{m}\sum_{k'\in\mathcal{S}^{(t)}}(\mathbf{w}_{k'}^{(t)}-\mathbf{w}_k^{(t)})\|^2 \tag{14}$$

$$\leq\frac{1}{m^2}\sum_{k\in\mathcal{S}^{(t)}}\sum_{k'\in\mathcal{S}^{(t)}}\|\mathbf{w}_{k'}^{(t)}-\mathbf{w}_k^{(t)}\|^2 \tag{15}$$

$$=\frac{1}{m^2}\sum_{\substack{k\neq k',\\ k,k'\in\mathcal{S}^{(t)}}}\|\mathbf{w}_{k'}^{(t)}-\mathbf{w}_k^{(t)}\|^2 \tag{16}$$

Observe from the update rule that $k,\ k'$ are in the same set $\mathcal{S}^{(t)}$ and hence the terms where $k=k'$ in the summation in (15) will be zero resulting in (16). Moreover for any arbitrary t there is a t_0 such that $0\leq t-t_0<\tau$ that $\mathbf{w}_{k'}^{(t_0)}=\mathbf{w}_k^{(t_0)}$ since the selected clients are updated with the global model at every τ. Hence even for an arbitrary t we have that the difference between $\|\mathbf{w}_{k'}^{(t)}-\mathbf{w}_k^{(t)}\|^2$ is upper bounded by τ updates. With non-increasing η_t over t and $\eta_{t_0}\leq 2\eta_t$, (16) can be further bounded as,

$$\frac{1}{m^2}\sum_{\substack{k\neq k',\\ k,k'\in\mathcal{S}^{(t)}}}\|\mathbf{w}_{k'}^{(t)}-\mathbf{w}_k^{(t)}\|^2\leq\frac{1}{m^2}\sum_{\substack{k\neq k',\\ k,k'\in\mathcal{S}^{(t)}}}\|\sum_{i=t_0}^{t_0+\tau-1}\eta_i(g_{k'}(\mathbf{w}_{k'}^{(i)},\xi_{k'}^{(i)})-g_k(\mathbf{w}_k^{(i)},\xi_k^{(i)}))\|^2 \tag{17}$$

$$\leq\frac{\eta_{t_0}^2\tau}{m^2}\sum_{\substack{k\neq k',\\ k,k'\in\mathcal{S}^{(t)}}}\sum_{i=t_0}^{t_0+\tau-1}\|(g_{k'}(\mathbf{w}_{k'}^{(i)},\xi_{k'}^{(i)})-g_k(\mathbf{w}_k^{(i)},\xi_k^{(i)}))\|^2 \tag{18}$$

$$\leq\frac{\eta_{t_0}^2\tau}{m^2}\sum_{\substack{k\neq k',\\ k,k'\in\mathcal{S}^{(t)}}}\sum_{i=t_0}^{t_0+\tau-1}[2\|g_{k'}(\mathbf{w}_{k'}^{(i)},\xi_{k'}^{(i)})\|^2+2\|g_k(\mathbf{w}_k^{(i)},\xi_k^{(i)})\|^2] \tag{19}$$

By taking expectation over (19),

$$\mathbb{E}[\frac{1}{m^2}\sum_{\substack{k\neq k',\\ k,k'\in\mathcal{S}^{(t)}}}\|\mathbf{w}_{k'}^{(t)}-\mathbf{w}_k^{(t)}\|^2]\leq\frac{2\eta_{t_0}^2\tau}{m^2}\mathbb{E}[\sum_{\substack{k\neq k',\\ k,k'\in\mathcal{S}^{(t)}}}\sum_{i=t_0}^{t_0+\tau-1}(\|g_{k'}(\mathbf{w}_{k'}^{(i)},\xi_{k'}^{(i)})\|^2+\|g_k(\mathbf{w}_k^{(i)},\xi_k^{(i)})\|^2)] \tag{20}$$

$$\leq\frac{2\eta_{t_0}^2\tau}{m^2}\mathbb{E}_{\mathcal{S}^{(t)}}[\sum_{\substack{k\neq k',\\ k,k'\in\mathcal{S}^{(t)}}}\sum_{i=t_0}^{t_0+\tau-1}2G^2] \tag{21}$$

$$=\frac{2\eta_{t_0}^2\tau}{m^2}\mathbb{E}_{\mathcal{S}^{(t)}}[\sum_{\substack{k\neq k',\\ k,k'\in\mathcal{S}^{(t)}}}2\tau G^2] \tag{22}$$

$$\leq\frac{16\eta_t^2(m-1)\tau^2G^2}{m} \tag{23}$$

$$\leq 16\eta_t^2\tau^2G^2 \tag{24}$$

where (23) is because there can be at most $m(m-1)$ pairs such that $k\neq k'$ in $\mathcal{S}^{(t)}$. □

Lemma B.3 (Upper bound for expectation over $\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2$ for any selection strategy π)**.** *With $\mathbb{E}[\cdot]$, the total expectation over all random sources including the random source from selection strategy we have the upper bound:*

$$\mathbb{E}[\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2]\leq\frac{1}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}}\|\mathbf{w}_k^{(t)}-\mathbf{w}^*\|^2] \tag{25}$$

Proof.

$$\mathbb{E}[\|\overline{\mathbf{w}}^{(t)} - \mathbf{w}^*\|^2] = \mathbb{E}[\|\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} \mathbf{w}_k^{(t)} - \mathbf{w}^*\|^2] = \mathbb{E}[\|\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} (\mathbf{w}_k^{(t)} - \mathbf{w}^*)\|^2] \tag{26}$$

$$\leq \frac{1}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}} \|\mathbf{w}_k^{(t)} - \mathbf{w}^*\|^2] \tag{27}$$

□

C Proof of Theorem 3.1

With $\overline{\mathbf{g}}^{(t)} = \frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} g_k(\mathbf{w}_k^{(t)}, \xi_k^{(t)})$ as defined in Section 2, we have that

$$\|\overline{\mathbf{w}}^{(t+1)} - \mathbf{w}^*\|^2 = \|\overline{\mathbf{w}}^{(t)} - \eta_t\overline{\mathbf{g}}^{(t)} - \mathbf{w}^*\|^2 \tag{28}$$

$$= \|\overline{\mathbf{w}}^{(t)} - \eta_t\overline{\mathbf{g}}^{(t)} - \mathbf{w}^* - \frac{\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)}) + \frac{\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)})\|^2 \tag{29}$$

$$= \|\overline{\mathbf{w}}^{(t)} - \mathbf{w}^* - \frac{\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)})\|^2 + \eta_t^2\|\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)}) - \overline{\mathbf{g}}^{(t)}\|^2$$
$$+ 2\eta_t\langle\overline{\mathbf{w}}^{(t)} - \mathbf{w}^* - \frac{\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)}), \frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)}) - \overline{\mathbf{g}}^{(t)}\rangle \tag{30}$$

$$= \|\overline{\mathbf{w}}^{(t)} - \mathbf{w}^*\|^2 \underbrace{-2\eta_t\langle\overline{\mathbf{w}}^{(t)} - \mathbf{w}^*, \frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)})\rangle}_{A_1}$$
$$\underbrace{+ 2\eta_t\langle\overline{\mathbf{w}}^{(t)} - \mathbf{w}^* - \frac{\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)}), \frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)}) - \overline{\mathbf{g}}^{(t)}\rangle}_{A_2}$$
$$\underbrace{+ \eta_t^2\|\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)})\|^2}_{A_3} + \underbrace{\eta_t^2\|\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} \nabla F_k(\mathbf{w}_k^{(t)}) - \overline{\mathbf{g}}^{(t)}\|^2}_{A_4} \tag{31}$$

First let's bound A_1.

$$-2\eta_t\langle\overline{\mathbf{w}}^{(t)}-\mathbf{w}^*,\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}}\nabla F_k(\mathbf{w}_k^{(t)})\rangle=-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}\langle\overline{\mathbf{w}}^{(t)}-\mathbf{w}^*,\nabla F_k(\mathbf{w}_k^{(t)})\rangle \tag{32}$$

$$=-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}\langle\overline{\mathbf{w}}^{(t)}-\mathbf{w}_k^{(t)},\nabla F_k(\mathbf{w}_k^{(t)})\rangle-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}\langle\mathbf{w}_k^{(t)}-\mathbf{w}^*,\nabla F_k(\mathbf{w}_k^{(t)})\rangle \tag{33}$$

$$\leq\frac{\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}\left(\frac{1}{\eta_t}\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}_k^{(t)}\|^2+\eta_t\|\nabla F_k(\mathbf{w}_k^{(t)})\|^2\right)-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}\langle\mathbf{w}_k^{(t)}-\mathbf{w}^*,\nabla F_k(\mathbf{w}_k^{(t)})\rangle \tag{34}$$

$$=\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}}\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}_k^{(t)}\|^2+\frac{\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}}\|\nabla F_k(\mathbf{w}_k^{(t)})\|^2-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}\langle\mathbf{w}_k^{(t)}-\mathbf{w}^*,\nabla F_k(\mathbf{w}_k^{(t)})\rangle \tag{35}$$

$$\begin{aligned}\leq\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}}\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}_k^{(t)}\|^2+\frac{2L\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}_k^{(t)})-F_k^*)\\-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}\langle\mathbf{w}_k^{(t)}-\mathbf{w}^*,\nabla F_k(\mathbf{w}_k^{(t)})\rangle\end{aligned} \tag{36}$$

$$\begin{aligned}\leq\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}}\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}_k^{(t)}\|^2+\frac{2L\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}_k^{(t)})-F_k^*)\\-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}\left[(F_k(\mathbf{w}_k^{(t)})-F_k(\mathbf{w}^*))+\frac{\mu}{2}\|\mathbf{w}_k^{(t)}-\mathbf{w}^*\|^2\right]\end{aligned} \tag{37}$$

$$\begin{aligned}\leq 16\eta_t^2\tau^2G^2-\frac{\eta_t\mu}{m}\sum_{k\in\mathcal{S}^{(t)}}\|\mathbf{w}_k^{(t)}-\mathbf{w}^*\|^2+\frac{2L\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}_k^{(t)})-F_k^*)\\-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}_k^{(t)})-F_k(\mathbf{w}^*))\end{aligned} \tag{38}$$

where (34) is due to the AM-GM inequality and Cauchy–Schwarz inequality, (36) is due to Lemma B.1, (37) is due to the μ-convexity of F_k, and (38) is due to Lemma B.2. Next, in expectation, $\mathbb{E}[A_2]=0$ due to the unbiased gradient. Next again with Lemma B.1 we bound A_3 as follows:

$$\eta_t^2\|\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}}\nabla F_k(\mathbf{w}_k^{(t)})\|^2=\frac{\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}}\left\|\nabla F_k(\mathbf{w}_k^{(t)})\right\|^2 \tag{39}$$

$$\leq\frac{2L\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}_k^{(t)})-F_k^*) \tag{40}$$

Lastly we can bound A_4 using the bound of variance of stochastic gradients as,

$$\mathbb{E}[\eta_t^2\|\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}}\nabla F_k(\mathbf{w}_k^{(t)})-\overline{\mathbf{g}}^{(t)}\|^2]=\eta_t^2\mathbb{E}[\|\sum_{k\in\mathcal{S}^{(t)}}\frac{1}{m}(g_k(\mathbf{w}_k^{(t)},\xi_k^{(t)})-\nabla F_k(\mathbf{w}_k^{(t)}))\|^2] \tag{41}$$

$$=\frac{\eta_t^2}{m^2}\mathbb{E}_{\mathcal{S}^{(t)}}[\sum_{k\in\mathcal{S}^{(t)}}\mathbb{E}\|g_k(\mathbf{w}_k^{(t)},\xi_k^{(t)})-\nabla F_k(\mathbf{w}_k^{(t)})\|^2] \tag{42}$$

$$\leq\frac{\eta_t^2\sigma^2}{m} \tag{43}$$

Using the bounds of A_1, A_2, A_3, A_4 above we have that the expectation of the LHS of (28) is bounded as

$$\begin{aligned}
&\mathbb{E}[\|\overline{\mathbf{w}}^{(t+1)} - \mathbf{w}^*\|^2] \\
\leq &\mathbb{E}[\|\overline{\mathbf{w}}^{(t)} - \mathbf{w}^*\|^2] - \frac{\eta_t\mu}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}} \|\mathbf{w}_k^{(t)} - \mathbf{w}^*\|^2] + 16\eta_t^2\tau^2G^2 \\
&+ \frac{\eta_t^2\sigma^2}{m} + \frac{4L\eta_t^2}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}_k^{(t)}) - F_k^*)] - \frac{2\eta_t}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}_k^{(t)}) - F_k(\mathbf{w}^*))] \qquad (44) \\
\leq &(1-\eta_t\mu)\mathbb{E}[\|\overline{\mathbf{w}}^{(t)} - \mathbf{w}^*\|^2] + 16\eta_t^2\tau^2G^2 \\
&+ \frac{\eta_t^2\sigma^2}{m} + \underbrace{\frac{4L\eta_t^2}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}_k^{(t)}) - F_k^*)] - \frac{2\eta_t}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}_k^{(t)}) - F_k(\mathbf{w}^*))]}_{A_5} \qquad (45)
\end{aligned}$$

where (45) is due to Lemma B.3. Now we aim to bound A_5 in (45). First we can represent A_5 in a different form as:

$$\begin{aligned}
&\mathbb{E}[\frac{4L\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}_k^{(t)}) - F_k^*) - \frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}_k^{(t)}) - F_k(\mathbf{w}^*))] \\
=&\mathbb{E}[\frac{4L\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}} F_k(\mathbf{w}_k^{(t)}) - \frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}} F_k(\mathbf{w}_k^{(t)}) - \frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k^* - F_k(\mathbf{w}^*)) \\
&+ \frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}} F_k^* - \frac{4L\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}} F_k^*] \qquad (46) \\
=&\mathbb{E}[\underbrace{\frac{2\eta_t(2L\eta_t - 1)}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}_k^{(t)}) - F_k^*)}_{A_6}] + 2\eta_t\mathbb{E}[\frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}^*) - F_k^*)] \qquad (47)
\end{aligned}$$

Now with $\eta_t < 1/(4L)$ and $\nu_t = 2\eta_t(1 - 2L\eta_t)$, we have that A_6 can be rewritten and bounded as

$$\begin{aligned}
&-\frac{\nu_t}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}_k^{(t)}) - F_k(\overline{\mathbf{w}}^{(t)}) + F_k(\overline{\mathbf{w}}^{(t)}) - F_k^*) \\
=&-\frac{\nu_t}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k(\mathbf{w}_k^{(t)}) - F_k(\overline{\mathbf{w}}^{(t)})) - \frac{\nu_t}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k(\overline{\mathbf{w}}^{(t)}) - F_k^*) \qquad (48) \\
\leq &-\frac{\nu_t}{m}\sum_{k\in\mathcal{S}^{(t)}} \left[\langle\nabla F_k(\overline{\mathbf{w}}^{(t)}), \mathbf{w}_k^{(t)} - \overline{\mathbf{w}}^{(t)}\rangle + \frac{\mu}{2}\|\mathbf{w}_k^{(t)} - \overline{\mathbf{w}}^{(t)}\|^2\right] - \frac{\nu_t}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k(\overline{\mathbf{w}}^{(t)}) - F_k^*) \qquad (49) \\
\leq &\frac{\nu_t}{m}\sum_{k\in\mathcal{S}^{(t)}} \left[\eta_t L(F_k(\overline{\mathbf{w}}^{(t)}) - F_k^*) + \left(\frac{1}{2\eta_t} - \frac{\mu}{2}\right)\|\mathbf{w}_k^{(t)} - \overline{\mathbf{w}}^{(t)}\|^2\right] - \frac{\nu_t}{m}\sum_{k\in\mathcal{S}^{(t)}} (F_k(\overline{\mathbf{w}}^{(t)}) - F_k^*) \qquad (50) \\
= &-\frac{\nu_t}{m}(1-\eta_t L)\sum_{k\in\mathcal{S}^{(t)}} (F_k(\overline{\mathbf{w}}^{(t)}) - F_k^*) + \left(\frac{\nu_t}{2\eta_t m} - \frac{\nu_t\mu}{2m}\right)\sum_{k\in\mathcal{S}^{(t)}} \|\mathbf{w}_k^{(t)} - \overline{\mathbf{w}}^{(t)}\|^2 \qquad (51) \\
\leq &-\frac{\nu_t}{m}(1-\eta_t L)\sum_{k\in\mathcal{S}^{(t)}} (F_k(\overline{\mathbf{w}}^{(t)}) - F_k^*) + \frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} \|\mathbf{w}_k^{(t)} - \overline{\mathbf{w}}^{(t)}\|^2 \qquad (52)
\end{aligned}$$

where (49) is due to $\mu-$convexity, (50) is due to Lemma B.1 and the AM-GM inequality and Cauchy–Schwarz

inequality, and (52) is due to the fact that $\frac{\nu_t(1-\eta_t\mu)}{2\eta_t} \leq 1$. Hence using this bound of A_6 we can upper bound A_5 as

$$\begin{aligned}
&\mathbb{E}[\frac{4L\eta_t^2}{m}\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}_k^{(t)})-F_k^*)-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}_k^{(t)})-F_k(\mathbf{w}^*))] \\
\leq&\frac{1}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}}\|\mathbf{w}_k^{(t)}-\overline{\mathbf{w}}^{(t)}\|^2]-\frac{\nu_t}{m}(1-\eta_tL)\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}}(F_k(\overline{\mathbf{w}}^{(t)})-F_k^*)] \\
&+\frac{2\eta_t}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}^*)-F_k^*)] && (53)\\
\leq&16\eta_t^2\tau^2G^2-\frac{\nu_t}{m}(1-\eta_tL)\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}}(F_k(\overline{\mathbf{w}}^{(t)})-F_k^*)]+\frac{2\eta_t}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}^*)-F_k^*)] && (54)\\
=&16\eta_t^2\tau^2G^2-\nu_t(1-\eta_tL)\mathbb{E}[\rho(\mathcal{S}(\pi,\overline{\mathbf{w}}^{(\tau\lfloor t/\tau\rfloor)}),\overline{\mathbf{w}}^{(t)})(F(\overline{\mathbf{w}}^{(t)})-\sum_{k=1}^K p_kF_k^*)] \\
&+2\eta_t\mathbb{E}[\rho(\mathcal{S}(\pi,\overline{\mathbf{w}}^{(\tau\lfloor t/\tau\rfloor)}),\mathbf{w}^*)(F^*-\sum_{k=1}^K p_kF_k^*)] && (55)\\
\leq&16\eta_t^2\tau^2G^2\underbrace{-\nu_t(1-\eta_tL)\overline{\rho}(\mathbb{E}[F(\overline{\mathbf{w}}^{(t)})]-\sum_{k=1}^K p_kF_k^*)}_{A_7}+2\eta_t\widetilde{\rho}\Gamma && (56)
\end{aligned}$$

where (55) is due to the definition of $\rho(\mathcal{S}(\pi,\mathbf{w}),\mathbf{w}')$ in Definition 3.2 and (56) is due to the definition of Γ in Definition 3.1 and the definitions of $\overline{\rho}$, $\widetilde{\rho}$ in Definition 3.2. We can expand A_7 in (56) as

$$\begin{aligned}
&-\nu_t(1-\eta_tL)\overline{\rho}(\mathbb{E}[F(\overline{\mathbf{w}}^{(t)})]-\sum_{k=1}^K p_kF_k^*) && (57)\\
=&-\nu_t(1-\eta_tL)\overline{\rho}\sum_{k=1}^K p_k(\mathbb{E}[F_k(\overline{\mathbf{w}}^{(t)}]-F^*+F^*-F_k^*) && (58)\\
=&-\nu_t(1-\eta_tL)\overline{\rho}\sum_{k=1}^K p_k(\mathbb{E}[F_k(\overline{\mathbf{w}}^{(t)}]-F^*)-\nu_t(1-\eta_tL)\overline{\rho}\sum_{k=1}^K p_k(F^*-F_k^*) && (59)\\
=&-\nu_t(1-\eta_tL)\overline{\rho}(\mathbb{E}[F(\overline{\mathbf{w}}^{(t)})]-F^*)-\nu_t(1-\eta_tL)\overline{\rho}\Gamma && (60)\\
\leq&-\frac{\nu_t(1-\eta_tL)\mu\overline{\rho}}{2}\mathbb{E}[\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2]-\nu_t(1-\eta_tL)\overline{\rho}\Gamma && (61)\\
\leq&-\frac{3\eta_t\mu\overline{\rho}}{8}\mathbb{E}[\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2]-2\eta_t(1-2L\eta_t)(1-\eta_tL)\overline{\rho}\Gamma && (62)\\
\leq&-\frac{3\eta_t\mu\overline{\rho}}{8}\mathbb{E}[\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2]-2\eta_t\overline{\rho}\Gamma+6\eta_t^2\overline{\rho}L\Gamma && (63)
\end{aligned}$$

where (61) is due to the $\mu-$convexity, (62) is due to $-2\eta_t(1-2L\eta_t)(1-\eta_tL)\leq-\frac{3}{4}\eta_t$, and (63) is due to $-(1-2L\eta_t)(1-\eta_tL)\leq-(1-3L\eta_t)$. Hence we can finally bound A_5 as

$$\begin{aligned}
&\frac{4L\eta_t^2}{m}\mathbb{E}[\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}_k^{(t)})-F_k^*)-\frac{2\eta_t}{m}\sum_{k\in\mathcal{S}^{(t)}}(F_k(\mathbf{w}_k^{(t)})-F_k(\mathbf{w}^*))] \\
\leq&-\frac{3\eta_t\mu\overline{\rho}}{8}\mathbb{E}[\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2]+2\eta_t\Gamma(\widetilde{\rho}-\overline{\rho})+\eta_t^2(6\overline{\rho}L\Gamma+16\tau^2G^2) && (64)
\end{aligned}$$

Now we can bound $\mathbb{E}[\|\overline{\mathbf{w}}^{(t+1)}-\mathbf{w}^*\|^2]$ as

$$\begin{aligned}
\mathbb{E}[\|\overline{\mathbf{w}}^{(t+1)}-\mathbf{w}^*\|^2]\leq&\left[1-\eta_t\mu\left(1+\frac{3\overline{\rho}}{8}\right)\right]\mathbb{E}[\|\overline{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2] \\
&+\eta_t^2\left(32\tau^2G^2+\frac{\sigma^2}{m}+6\overline{\rho}L\Gamma\right)+2\eta_t\Gamma(\widetilde{\rho}-\overline{\rho}) && (65)
\end{aligned}$$

By defining $\Delta_{t+1} = \mathbb{E}[\|\overline{\mathbf{w}}^{(t+1)} - \mathbf{w}^*\|^2]$, $B = 1 + \frac{3\overline{\rho}}{8}$, $C = 32\tau^2 G^2 + \frac{\sigma^2}{m} + 6\overline{\rho} L \Gamma$, $D = 2\Gamma(\widetilde{\rho} - \overline{\rho})$, we have that

$$\Delta_{t+1} \leq (1 - \eta_t \mu B)\Delta_t + \eta_t^2 C + \eta_t D \tag{66}$$

By setting $\Delta_t \leq \frac{\psi}{t+\gamma}$, $\eta_t = \frac{\beta}{t+\gamma}$ and $\beta > \frac{1}{\mu B}$, $\gamma > 0$ by induction we have that

$$\psi = \max\left\{\gamma \|\overline{\mathbf{w}}^{(0)} - \mathbf{w}^*\|^2, \frac{1}{\beta\mu B - 1}\left(\beta^2 C + D\beta(t+\gamma)\right)\right\} \tag{67}$$

Then by the L-smoothness of $F(\cdot)$, we have that

$$\mathbb{E}[F(\overline{\mathbf{w}}^{(t)})] - F^* \leq \frac{L}{2}\Delta_t \leq \frac{L}{2}\frac{\psi}{\gamma + t} \tag{68}$$

D Proof of Theorem A.1

With fixed learning rate $\eta_t = \eta$, we can rewrite (66) as

$$\Delta_{t+1} \leq (1 - \eta\mu B)\Delta_t + \eta^2 C + \eta D \tag{69}$$

and with $\eta \leq \min\{\frac{1}{2\mu B}, \frac{1}{4L}\}$ using recursion of (69) we have that

$$\Delta_t \leq (1 - \eta\mu B)^t \Delta_0 + \frac{\eta^2 C + \eta D}{\eta\mu B}(1 - (1 - \eta\mu B)^t) \tag{70}$$

Using $\Delta_t \leq \frac{2}{\mu}(F(\overline{\mathbf{w}}^{(t)}) - F^*)$ and L-smoothness, we have that

$$F(\overline{\mathbf{w}}^{(t)}) - F^* \leq \frac{L}{\mu}(1 - \eta\mu B)^t (F(\overline{\mathbf{w}}^{(0)}) - F^*) + \frac{L(\eta C + D)}{2\mu B}(1 - (1 - \eta\mu B)^t) \tag{71}$$

$$= \frac{L}{\mu}\left[1 - \eta\mu\left(1 + \frac{3\overline{\rho}}{8}\right)\right]^t (F(\overline{\mathbf{w}}^{(0)}) - F^*) + \frac{4L(\eta C + D)}{\mu(8 + 3\overline{\rho})}\left[1 - \left[1 - \eta\mu\left(1 + \frac{3\overline{\rho}}{8}\right)\right]^t\right] \tag{72}$$

E Extension: Generalization to different averaging schemes

While we considered a simple averaging scheme where $\overline{\mathbf{w}}^{(t+1)} = \frac{1}{m}\sum_{k \in \mathcal{S}^{(t)}}\left(\mathbf{w}_k^{(t)} - \eta_t g_k(\mathbf{w}_k^{(t)})\right)$, we can extend the averaging scheme to any scheme $\mathbf{q}$ such that the averaging weights q_k are invariant in time and satisfies $\sum_{k \in \mathcal{S}^{(t)}} q_k = 1$ for any t. Note that $\mathbf{q}$ includes the random sampling without replacement scheme introduced by Li et al. (2020b) where the clients are sampled uniformly at random without replacement with the averaging coefficients $q_k = p_k K/m$. With such averaging scheme $\mathbf{q}$, we denote the global model for the averaging scheme q_k as $\widehat{\mathbf{w}}^{(t)}$, where $\widehat{\mathbf{w}}^{(t+1)} \triangleq \sum_{k \in \mathcal{S}^{(t)}} q_k \left(\mathbf{w}_k^{(t)} - \eta_t g_k(\mathbf{w}_k^{(t)})\right)$, and the update rule changes to

$$\widehat{\mathbf{w}}^{(t+1)} = \widehat{\mathbf{w}}^{(t)} - \eta_t \widehat{\mathbf{g}}^{(t)} = \widehat{\mathbf{w}}^{(t)} - \eta_t \left(\sum_{k \in \mathcal{S}^{(t)}} q_k g_k(\mathbf{w}_k^{(t)}, \xi_k^{(t)})\right) \tag{73}$$

where $\widehat{\mathbf{g}}^{(t)} = \sum_{k \in \mathcal{S}^{(t)}} q_k g_k(\mathbf{w}_k^{(t)}, \xi_k^{(t)})$. We show that the convergence analysis for the averaging scheme $\mathbf{q}$ is consistent with Theorem 3.1. In the case of the averaging scheme $\mathbf{q}$, we have that Lemma B.2 and Lemma B.3 shown in Appendix B, each becomes

$$\frac{1}{m}\mathbb{E}[\sum_{k \in \mathcal{S}^{(t)}} \|\widehat{\mathbf{w}}^{(t)} - \mathbf{w}_k^{(t)}\|^2] \leq 16\eta_t^2 m(m-1)\tau^2 G^2 \tag{74}$$

$$\mathbb{E}[\|\widehat{\mathbf{w}}^{(t)} - \mathbf{w}^*\|^2] \leq m\mathbb{E}[\sum_{k \in \mathcal{S}^{(t)}} q_k \|\mathbf{w}_k^{(t)} - \mathbf{w}^*\|^2] \tag{75}$$

Then, using the same method we used for the proof of Theorem 3.1, we have that

$$\mathbb{E}[\|\widehat{\mathbf{w}}^{(t+1)}-\mathbf{w}^*\|^2] \leq \left(1-\frac{\eta_t\mu}{m}\right)\mathbb{E}[\|\widehat{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2]+\eta_t^2\sigma^2 m+16m^2(m-1)\eta_t^2\tau^2G^2+ \underbrace{\mathbb{E}\left[2L\eta_t^2(1+m)\sum_{k\in\mathcal{S}^{(t)}}q_k(F_k(\mathbf{w}_k^{(t)})-F_k^*)-2\eta_t\sum_{k\in\mathcal{S}^{(t)}}q_k(F_k(\mathbf{w}_k^{(t)})-F_k(\mathbf{w}^*))\right]}_{M} \tag{76}$$

By defining the selection skew for averaging scheme $\mathbf{q}$ similar to Definition 5 as

$$\rho_{\mathbf{q}}(\mathcal{S}(\pi,\mathbf{w}),\mathbf{w}')=\frac{\mathbb{E}_{\mathcal{S}(\pi,\mathbf{w})}[\sum_{k\in\mathcal{S}(\pi,\mathbf{w})}q_k(F_k(\mathbf{w}')-F_k^*)]}{F(\mathbf{w}')-\sum_{k=1}^K p_kF_k^*}\geq 0, \tag{77}$$

and

$$\overline{\rho}_{\mathbf{q}} \triangleq \min_{\mathbf{w},\mathbf{w}'}\rho_{\mathbf{q}}(\mathcal{S}(\pi,\mathbf{w}),\mathbf{w}') \tag{78}$$

$$\widetilde{\rho}_{\mathbf{q}} \triangleq \max_{\mathbf{w}}\rho_{\mathbf{q}}(\mathcal{S}(\pi,\mathbf{w}),\mathbf{w}^*)=\frac{\max_{\mathbf{w}}\mathbb{E}_{\mathcal{S}(\pi,\mathbf{w})}[\sum_{k\in\mathcal{S}(\pi,\mathbf{w})}q_k(F_k(\mathbf{w}^*)-F_k^*)]}{\Gamma} \tag{79}$$

With $\eta_t<1/(2L(1+m))$, using the same methodology for proof of Theorem 3.1 we have that M becomes upper bounded as

$$\mathbb{E}\left[2L\eta_t^2(1+m)\sum_{k\in\mathcal{S}^{(t)}}q_k(F_k(\mathbf{w}_k^{(t)})-F_k^*)-2\eta_t\sum_{k\in\mathcal{S}^{(t)}}q_k(F_k(\mathbf{w}_k^{(t)})-F_k(\mathbf{w}^*))\right] \tag{80}$$

$$\leq -\frac{\eta_t\mu\overline{\rho}_{\mathbf{q}}}{2}\mathbb{E}[\|\widehat{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2]+2\eta_t\Gamma(\widetilde{\rho}_{\mathbf{q}}-\overline{\rho}_{\mathbf{q}})+16m^2(m-1)\eta_t^2\tau^2G^2+2L\eta_t^2(2+m)\overline{\rho}_{\mathbf{q}}\Gamma \tag{81}$$

Finally we have that

$$\begin{aligned}\mathbb{E}[\|\widehat{\mathbf{w}}^{(t+1)}-\mathbf{w}^*\|^2] \leq \left[1-\eta_t\mu\left(\frac{1}{m}+\frac{\overline{\rho}_{\mathbf{q}}}{2}\right)\right]\mathbb{E}[\|\widehat{\mathbf{w}}^{(t)}-\mathbf{w}^*\|^2]+2\eta_t\Gamma(\widetilde{\rho}_{\mathbf{q}}-\overline{\rho}_{\mathbf{q}})\\ +\eta_t^2[32m^2(m-1)\tau^2G^2+\sigma^2 m+2L(2+m)\overline{\rho}_{\mathbf{q}}\Gamma]\end{aligned} \tag{82}$$

By defining $\widehat{\Delta}_{t+1}=\mathbb{E}[\|\widehat{\mathbf{w}}^{(t+1)}-\mathbf{w}^*\|^2]$, $\widehat{B}=\frac{1}{m}+\frac{\overline{\rho}_{\mathbf{q}}}{2}$, $\widehat{C}=32m^2(m-1)\tau^2G^2+\sigma^2 m+2L(2+m)\overline{\rho}_{\mathbf{q}}\Gamma$, $\widehat{D}=2\Gamma(\widetilde{\rho}_{\mathbf{q}}-\overline{\rho}_{\mathbf{q}})$, we have that

$$\widehat{\Delta}_{t+1}\leq(1-\eta_t\mu\widehat{B})\widehat{\Delta}_t+\eta_t^2\widehat{C}+\eta_t\widehat{D} \tag{83}$$

Again, by setting $\widehat{\Delta}_t\leq\frac{\psi}{t+\gamma}$, $\eta_t=\frac{\beta}{t+\gamma}$ and $\beta>\frac{1}{\mu\widehat{B}}$, $\gamma>0$ by induction we have that

$$\psi=\max\left\{\gamma\|\overline{\mathbf{w}}^{(0)}-\mathbf{w}^*\|^2,\frac{1}{\beta\mu\widehat{B}-1}\left(\beta^2\widehat{C}+\widehat{D}\beta(t+\gamma)\right)\right\} \tag{84}$$

Then by the L-smoothness of $F(\cdot)$, we have that

$$\mathbb{E}[F(\overline{\mathbf{w}}^{(t)})]-F^*\leq\frac{L}{2}\widehat{\Delta}_t\leq\frac{L}{2}\frac{\psi}{\gamma+t} \tag{85}$$

With $\beta=\frac{m}{\mu}$, $\gamma=\frac{4m(1+m)L}{\mu}$ and $\eta_t=\frac{\beta}{t+\gamma}$, we have that

$$\begin{aligned}&\mathbb{E}[F(\widehat{\mathbf{w}}^{(T)})]-F^*\leq\\ &\underbrace{\frac{1}{(T+\gamma)}\left[\frac{Lm^2(32m(m-1)\tau^2G^2+\sigma^2)}{\mu^2\overline{\rho}_{\mathbf{q}}}+\frac{2L^2m(m+2)\Gamma}{\mu^2}+\frac{L\gamma\|\overline{\mathbf{w}}^{(0)}-\mathbf{w}^*\|^2}{2}\right]}_{\text{Vanishing Error Term}}\\ &+\underbrace{\frac{2L\Gamma}{\overline{\rho}_{\mathbf{q}}\mu}\left(\frac{\widetilde{\rho}_{\mathbf{q}}}{\overline{\rho}_{\mathbf{q}}}-1\right)}_{\text{Non-vanishing bias}}\end{aligned} \tag{86}$$

which is consistent with Theorem 3.1.

F Experiment Details

Quadratic Model Optimization. For the quadratic model optimization, we set each local objective function as strongly convex as follows:

$$F_k(\mathbf{w}) = \frac{1}{2}\mathbf{w}^\top \mathbf{H}_k \mathbf{w} - \mathbf{e}_k^\top \mathbf{w} + \frac{1}{2}\mathbf{e}_k^\top \mathbf{H}_k^{-1}\mathbf{e}_k \tag{87}$$

$\mathbf{H}_k \in \mathbb{R}^{v\times v}$ is a diagonal matrix $\mathbf{H}_k = h_k\mathbf{I}$ with $h_k \sim \mathcal{U}(1,20)$ and $\mathbf{e}_k \in \mathbb{R}^v$ is an arbitrary vector. We set the global objective function as $F(\mathbf{w}) = \sum_{k=1}^K p_k F_k(\mathbf{w})$, where the data size p_k follows the power law distribution $P(x;a) = ax^{a-1},\ 0 \le x \le 1,\ a = 3$. We can easily show that the optimum for $F_k(\mathbf{w})$ and $F(\mathbf{w})$ is $\mathbf{w}_k^* = \mathbf{H}_k^{-1}\mathbf{e}_k$ and $\mathbf{w}^* = (\sum_{k=1}^K p_k\mathbf{H}_k)^{-1}(\sum_{k=1}^K p_k\mathbf{e}_k)$ respectively. The gradient descent update rule for the local model of client k in the quadratic model optimization is

$$\mathbf{w}_k^{(t+1)} = \mathbf{w}_k^{(t)} - \eta(\mathbf{H}_k\mathbf{w}_k^{(t)} - \mathbf{e}_k) \tag{88}$$

where the global model is defined as $\overline{\mathbf{w}}^{(t+1)} = \frac{1}{m}\sum_{k\in\mathcal{S}^{(t)}} \mathbf{w}_k^{(t+1)}$. We sample $m = KC$ clients for every round where for each round the clients perform τ gradient descent local iterations with fixed learning rate η and then these local models are averaged to update the global model. For the implementation of $\pi_{\text{adapow-d}}$, d was decreased half from $d = K$ for every 5000 rounds. For all simulations we set $\tau = 2,\ v = 5,\ \eta = 2\times 10^{-5}$.

For the estimation of $\overline{\rho}$ and $\widetilde{\rho}$ for the quadratic model, we get the estimates of the theoretical $\overline{\rho}$, $\widetilde{\rho}$ values by doing a grid search over a large range of possible $\mathbf{w}, \mathbf{w}'$ for $\rho(\mathcal{S}(\pi,\mathbf{w}),\mathbf{w}')$ and $\rho(\mathcal{S}(\pi,\mathbf{w}),\mathbf{w}^*)$ respectively. The distribution of $\mathcal{S}(\pi,\mathbf{w})$ is estimated by simulating 10000 iterations of client sampling for each π and $\mathbf{w}$.

Logistic Regression on Synthetic Dataset. We conduct simulations on synthetic data which allows precise manipulation of heterogeneity. Using the methodology constructed in Sahu et al. (2020), we use the dataset with large data heterogeneity, `Synthetic`(1,1). We assume in total 30 devices where the local dataset sizes for each device follows the power law. For the implementation of $\pi_{\text{adapow-d}}$, d was decreased to $d = m$ from $d = K$ at half the entire communication rounds. We set the mini batch-size to 50 with $\tau = 30$, and $\eta = 0.05$, where η is decayed to $\eta/2$ every 300 and 600 rounds.

DNN Experiments. For image datasets, FMNIST (the MIT License) and CIFAR10 (the MIT License), we construct the heterogeneous data partition amongst clients using the Dirichlet distribution $\text{Dir}_K(\alpha)$ (Hsu et al., 2019), where α determines the degree of the data heterogeneity across clients (the data size imbalance and degree of label skew across clients). Smaller α indicates larger data heterogeneity. We experiment with three different seeds for the randomness in the dataset partition across clients and present the averaged results. For Sent140, we randomly select 314 users (twitter accounts) that have more than or equal to 32 tweets, and the data heterogeneity across the clients is naturally set. All experiments are conducted with clusters equipped with one NVIDIA TitanX GPU. The number of clusters we use vary by C, the fraction of clients we select. The machines communicate amongst each other through Ethernet to transfer the model parameters and information necessary for client selection. Each machine is regarded as one client in the FL setting. The algorithms are implemented by PyTorch. For all datasets we divide the train/validation/test dataset into 0.8/0.05/0.15 ratio where the clients' datasets are partitioned amongst the training dataset.

- **MLP on FMNIST for Image Classification**: We train a deep multi-layer perceptron network with 2 hidden layers of units $[64, 30]$ with dropout after the first hidden layer where the input is the flattened image and the output is consisted of 10 units each of one of the 0-9 labels. For all experiments we use mini-batch size of $b = 64$, with $\tau = 30$ and $\eta = 0.005$, where η is decayed by half for every 150, 300 rounds.
- **CNN on CIFAR10 for Image Classification**: We train a deep convolutional neural network with 2 convolutional layers with max pooling and 4 hidden fully connected linear layers of units $[120, 100, 84, 50]$. The input is the flattened convolution output and the output is consisted of 10 units each of one of the 0-9 labels. For all experiments we use mini-batch size of $b = 128$, with $\tau = 64$ and $\eta = 0.5$, where η is decayed by half for every 150, 300 rounds.
- **MLP on Sent140 for Text Sentiment Analysis**: We train a deep multi-layer perceptron network with 3 hidden layers of units $[128, 86, 30]$ with pre-trained 200D average-pooled GloVe embedding (Pennington

et al., 2014). The input is the embedded 200D vector and the output is a binary classifier determining whether the tweet sentiment is positive or negative with labels 0 and 1 respectively. For all experiments we use mini-batch size of $b = 32$, with $\tau = 100$ and $\eta = 0.05$.

Pseudo-code of the variants of pow-d: cpow-d and rpow-d. We here present the pseudo-code for $\pi_{\text{cpow-d}}$ and $\pi_{\text{rpow-d}}$. Note that the pseudo-code for $\pi_{\text{cpow-d}}$ in Algorithm 1 can be generalized to the algorithm for $\pi_{\text{pow-d}}$, by changing $\frac{1}{|\widehat{\xi}_k|}\sum_{\xi\in\widehat{\xi}_k} f(\mathbf{w},\xi)$ to $F_k(\mathbf{w})$.

Algorithm 1 Pseudo code for `cpow-d`: computation efficient variant of `pow-d`

Input: $m,\ d,\ p_k$ for $k \in [K]$, mini-batch size $b = |\widehat{\xi}_k|$ for computing $\frac{1}{|\widehat{\xi}_k|}\sum_{\xi\in\widehat{\xi}_k} f(\mathbf{w},\xi)$
Output: $\mathcal{S}^{(t)}$
Initialize: empty sets $\mathcal{S}^{(t)}$ and $\mathcal{A}$
Global server do:
  Get $\mathcal{A} = \{d$ indices sampled without replacement from $[K]$ by $p_k\}$
  Send the global model $\overline{\mathbf{w}}^{(t)}$ to the d clients in $\mathcal{A}$
  Receive $\frac{1}{|\widehat{\xi}_k|}\sum_{\xi\in\widehat{\xi}_k} f(\mathbf{w},\xi)$ from all clients in $\mathcal{A}$
  Get $\mathcal{S}^{(t)} = \{m$ clients with largest $\frac{1}{|\widehat{\xi}_k|}\sum_{\xi\in\widehat{\xi}_k} f(\mathbf{w},\xi)$ (break ties randomly)$\}$
Clients in $\mathcal{A}$ in parallel do:
  Create mini-batch $\widehat{\xi}_k$ from sampling b samples uniformly at random from $\mathcal{B}_k$ and compute $\frac{1}{|\widehat{\xi}_k|}\sum_{\xi\in\widehat{\xi}_k} f(\mathbf{w},\xi)$ and send it to the server
Return: $\mathcal{S}^{(t)}$

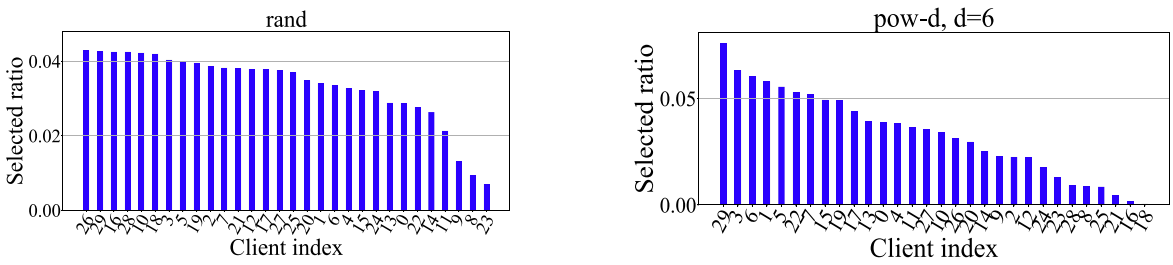
Algorithm 2 Pseudo code for `rpow-d`: computation & communication efficient variant of `pow-d`

Input: $m,\ d,\ p_k$ for $k \in [K]$
Output: $\mathcal{S}^{(t)}$
Initialize: empty sets $\mathcal{S}^{(t)}$ and $\mathcal{A}$, and list A_{tmp} with K elements all equal to `inf`
All client $k \in \mathcal{S}^{(t-1)}$ **do:**
  For t mod $\tau = 0$, send $\frac{1}{\tau b}\sum_{l=t-\tau+1}^{t}\sum_{\xi\in\xi_k^{(l)}} f(\mathbf{w}_k^{(l)},\xi)$ to the server with its local model
Global server do:
  Receive and update $A_{\text{tmp}}[k] = \frac{1}{\tau b}\sum_{l=t-\tau+1}^{t}\sum_{\xi\in\xi_k^{(l)}} f(\mathbf{w}_k^{(l)},\xi)$ for $k \in \mathcal{S}^{(t-1)}$
  Get $\mathcal{A} = \{d$ indices sampled without replacement from $[K]$ by $p_k\}$
  Get $\mathcal{S}^{(t)} = \{m$ clients with largest values in $[A_{\text{tmp}}[i]$ for $i \in \mathcal{A}]$, (break ties randomly)$\}$
Return: $\mathcal{S}^{(t)}$

G Additional Experiment Results

G.1 Selected Client Profile

We further visualize the difference between our proposed sampling strategy $\pi_{\text{pow-d}}$ and the baseline scheme π_{rand} by showing the selected frequency ratio of the clients for $K = 30,\ C = 0.1$ for the quadratic simulations in Fig. 8. Note that the selected ratio for π_{rand} reflects each client's dataset size. We show that the selected frequencies of clients for $\pi_{\text{pow-d}}$ are not proportional to the data size of the clients, and we are selecting clients frequently even when they have relatively low data size like client 6 or 22. We are also not necessarily frequently selecting the clients that have the highest data size such as client 26. This aligns well with our main motivation of Power-of-Choice that weighting the clients' importance based on their data size does not achieve the best performance, and rather considering their local loss values along with the data size better represents their importance. Note that the selected frequency for π_{rand} is less biased than $\pi_{\text{pow-d}}$.



(a) Selected client profile for π_{rand}

(b) Selected client profile for $\pi_{\text{pow-d}}$

Figure 8: Clients' selected frequency ratio for optimizing the quadratic model for π_{rand} and $\pi_{\text{pow-d}}$ with $K = 30,\ C = 0.1$. The selected ratio is sorted in the descending order.

Table 2: Comparison of $R_{60},\ t_{\text{comp}}$ (sec), and test accuracy (%) for different sampling strategies with $\alpha = 2$. The ratio R_{60} / (R_{60} for rand, $C = 0.1$) and t_{comp} / (t_{comp} for rand, $C = 0.1$) are each shown in the parenthesis.

	$C = 0.1$	$C = 0.03$				
	rand	rand	pow-d, $d = 6$	cpow-d, $d = 6$	rpow-d, $d = 50$	afl
R_{60}	135	136(1.01)	**82 (0.61)**	**89 (0.66)**	99(0.73)	131(0.97)
t_{comp}	0.42	0.36(0.85)	**0.46 (1.08)**	**0.38 (0.88)**	0.36(0.86)	0.36(085)
Test Acc.	**63.50**±2.74	66.03±1.47	**73.81**±1.14	**73.36**±1.17	72.52±0.89	70.64±1.99

G.2 Communication and Computation Efficiency with larger data heterogeneity

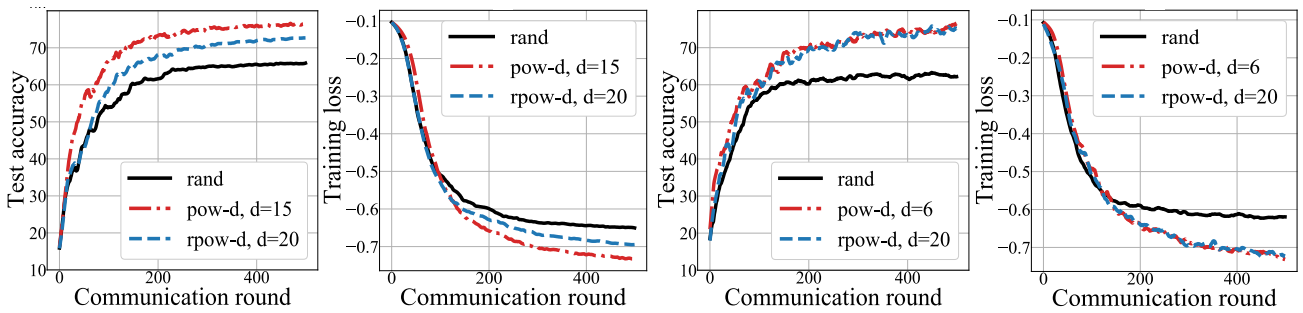
In Table 2, we show the communication and computation efficiency of Power-of-Choice for $\alpha = 2$, as we showed for $\alpha = 0.3$ in Table 1 in Section 5. With $C = 0.03$ fraction of clients, $\pi_{\text{pow-d}},\ \pi_{\text{cpow-d}}$, and $\pi_{\text{rpow-d}}$ have better test accuracy of at least approximately 10% higher test accuracy performance than ($\pi_{\text{rand}},\ C = 0.1$). R_{60} for $\pi_{\text{pow-d}},\ \pi_{\text{cpow-d}},\ \pi_{\text{rpow-d}}$ is 0.61, 0.66, 0.73 times that of ($\pi_{\text{rand}},\ C = 0.1$) respectively. This indicates that we can reduce the number of communication rounds by at least 0.6 using 1/3 of clients compared to ($\pi_{\text{rand}},\ C = 0.1$) and still get higher test accuracy performance. The computation time t_{comp} for $\pi_{\text{cpow-d}}$ and $\pi_{\text{rpow-d}}$ with $C = 0.03$ is smaller than that of ($\pi_{\text{rand}},\ C = 0.1$).

G.3 Intermittent Client Availability

In real world scenarios, certain clients may not be available due to varying availability of resources such as battery power or wireless connectivity. Hence we experiment with a virtual scenario, where amongst K clients, for each communication round, we select clients alternately from one group out of two fixed groups, where each group has $0.5K$ clients. This altering selection reflects a more realistic client selection scenario where, for example, we have different time zones across clients. For each communication round, we select 0.1 portion of clients from the corresponding group uniformly at random and exclude them from the client selection process. This random exclusion of certain clients represents the randomness in the client availability within that group for cases such as low battery power or wireless connectivity. In Fig. 9 we show that $\pi_{\text{pow-d}}$ and $\pi_{\text{rpow-d}}$ achieves 10% and 5% test accuracy improvement respectively compared to π_{rand} for $\alpha = 2$. For $\alpha = 3$, both $\pi_{\text{pow-d}}$ and $\pi_{\text{rpow-d}}$ shows 10% improvement. Therefore, we demonstrate that Power-of-Choice also performs well in a realistic scenario where clients are available intermittently.

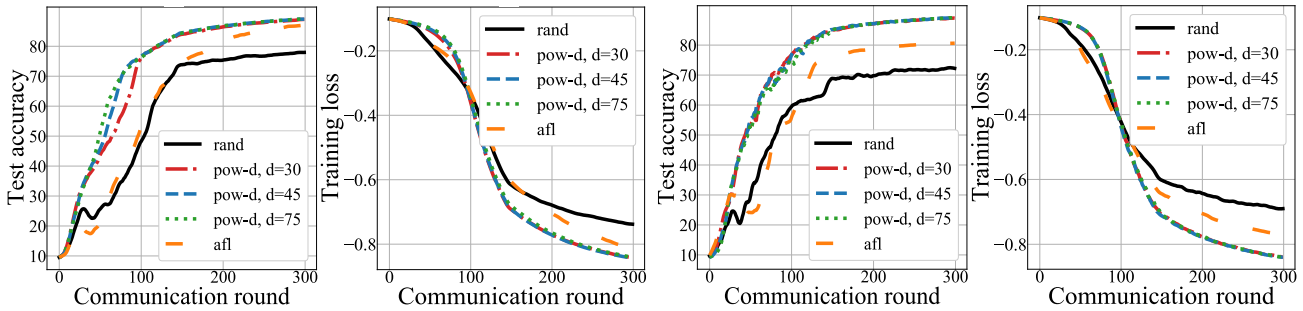
G.4 Results for DNN on Non-iid Partitioned EMNIST Dataset

To provide validation of the consistency in our results of $\pi_{\text{pow-d}}$ and its variants on the FMNIST dataset, we present additional experiment results on the EMNIST dataset sorted by digits with $K = 500,\ C = 0.03$. We train a deep multi-layer perceptron network with two hidden layers on the dataset partitioned heterogeneously across the clients in the same way as for the FMNIST dataset. For all experiments, we use $b = 64,\ \tau = 30$, and $\eta = 0.005$ where η is decayed by half at round 300.



(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

Figure 9: Test accuracy and training loss in the virtual environment where clients have intermittent availability for $K = 100$, $C = 0.03$ with π_{rand}, $\pi_{\text{pow-d}}$, and $\pi_{\text{rpow-d}}$ on the FMNIST dataset. For both $\alpha = 2$ and $\alpha = 3$, $\pi_{\text{pow-d}}$ achieves approximately 10% higher test accuracy than π_{rand}.



(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

Figure 10: Test accuracy and training loss for different sampling strategies for $K = 500$, $C = 0.03$ with π_{rand}, $\pi_{\text{pow-d}}$, and π_{afl} on the EMNIST dataset.

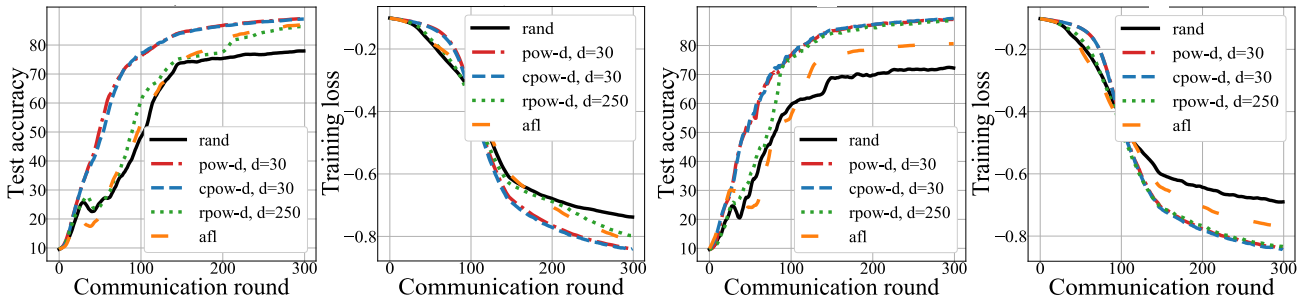
In Fig. 10, we show that $\pi_{\text{pow-d}}$ performs with significantly higher test accuracy than π_{rand} for varying d for both $\alpha = 2$ and 0.3. For $\alpha = 2$, π_{afl} is able to follow the performance of $\pi_{\text{pow-d}}$ in the later communication rounds, but is slower in achieving the same test accuracy than $\pi_{\text{pow-d}}$. Moreover, in Fig. 11, we show that $\pi_{\text{cpow-d}}$ works as good as $\pi_{\text{pow-d}}$ for both large and small data heterogeneity. The performance of $\pi_{\text{rpow-d}}$ falls behind $\pi_{\text{pow-d}}$ and $\pi_{\text{cpow-d}}$ for smaller data heterogeneity, whereas for larger data heterogeneity, $\pi_{\text{rpow-d}}$ is able to perform similarly with $\pi_{\text{pow-d}}$ and $\pi_{\text{cpow-d}}$.

G.5 Effect of the fraction of selected clients

In Fig. 12, for larger $C = 0.1$ with $\alpha = 2$, the test accuracy improvement for $\pi_{\text{pow-d}}$ is even higher than the case of $C = 0.03$ with approximately 15% improvement. $\pi_{\text{cpow-d}}$ performs slightly lower in test accuracy than $\pi_{\text{pow-d}}$ but still performs better than π_{rand} and π_{afl}. $\pi_{\text{rpow-d}}$ performs as well as π_{afl}. For $\alpha = 0.3$, $\pi_{\text{pow-d}}$, $\pi_{\text{cpow-d}}$, and $\pi_{\text{rpow-d}}$ have approximately equal test accuracy performance, higher than π_{rand} by 5%. The Power-of-Choice strategies all perform slightly better than π_{afl}. Therefore we show that Power-of-Choice performs well for selecting a larger fraction of clients, i.e., when we have larger $C = 0.1 > 0.03$.

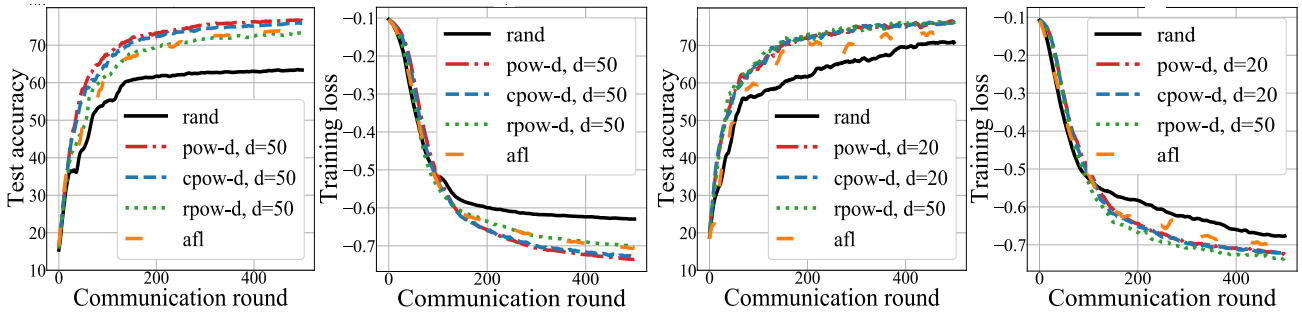
G.6 Effect of Mini-batch Size and Local Epochs

We evaluate the effect of mini-batch size b and local epochs τ on the FMNIST experiments with different sets of hyper-parameters: $(b, \tau) \in \{(128, 30), (64, 100)\}$. Note that $(b, \tau) = (64, 30)$ is the hyper-parameter setting used for the results in Fig. 4 and Fig. 5. For $b = 128$, we observe that the performance improvement of $\pi_{\text{pow-d}}$ and its variants over π_{rand} and π_{afl} is consistent with $b = 64$ (see Fig. 13 and Fig. 14). In Fig. 15 and Fig. 16, for $\tau = 100$, with smaller data heterogeneity, the performance gap between π_{rand} and $\pi_{\text{pow-d}}$ and its variants is consistent with that of $\tau = 30$. For larger data heterogeneity, however, increasing the local epochs results in π_{rand} and $\pi_{\text{pow-d}}$



(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

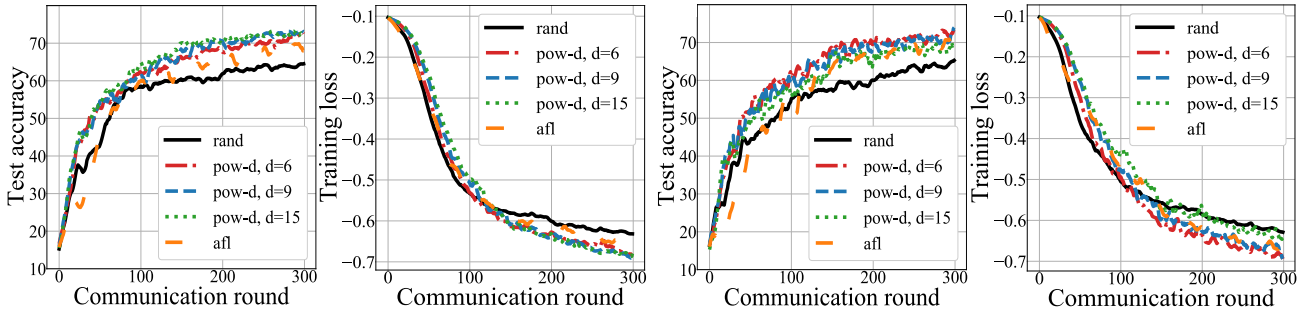
Figure 11: Test accuracy and training loss for different sampling strategies for $K = 500$, $C = 0.03$ with π_{rand}, $\pi_{\text{pow-d}}$, $\pi_{\text{cpow-d}}$, $\pi_{\text{rpow-d}}$, and π_{afl} on the EMNIST dataset.



(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

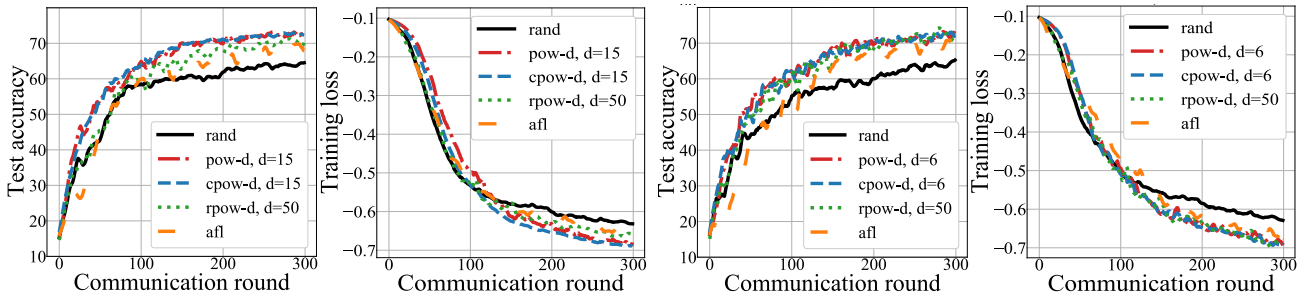
Figure 12: Test accuracy and training loss for different sampling strategies for $K = 100$, $C = 0.1$ with π_{rand}, $\pi_{\text{pow-d}}$, $\pi_{\text{cpow-d}}$, $\pi_{\text{rpow-d}}$, and π_{afl} on the FMNIST dataset. For larger $C = 0.1$, $\pi_{\text{pow-d}}$ performs with 15% and 5% higher test accuracy than π_{rand} for $\alpha = 2$ and $\alpha = 0.3$ respectively.

$\pi_{\text{pow-d}}$ and its variants performing similarly. This shows that with larger data heterogeneity, larger τ results in increasing the selection skew towards specific clients, and weakens generalization.



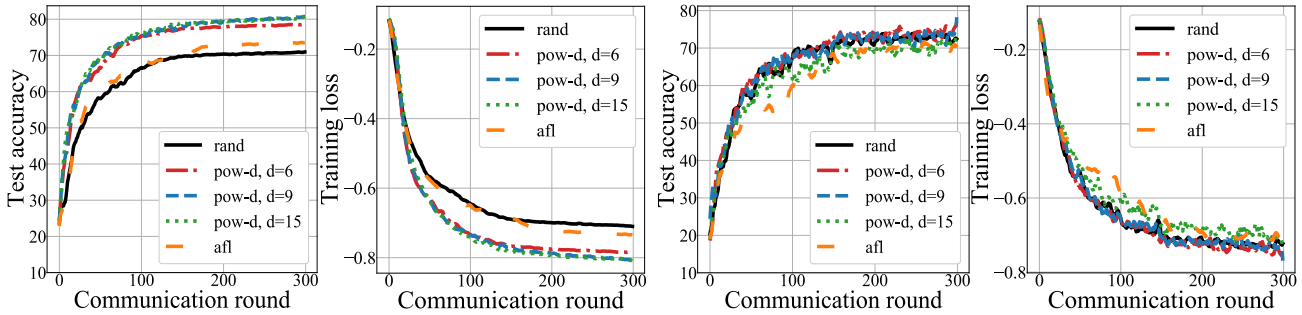
(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

Figure 13: Test accuracy and training loss for π_{rand}, $\pi_{\text{pow-d}}$, and π_{afl} for $K = 100$, $C = 0.03$ on the FMNIST dataset with mini-batch size $b = 128$ and $\tau = 30$.



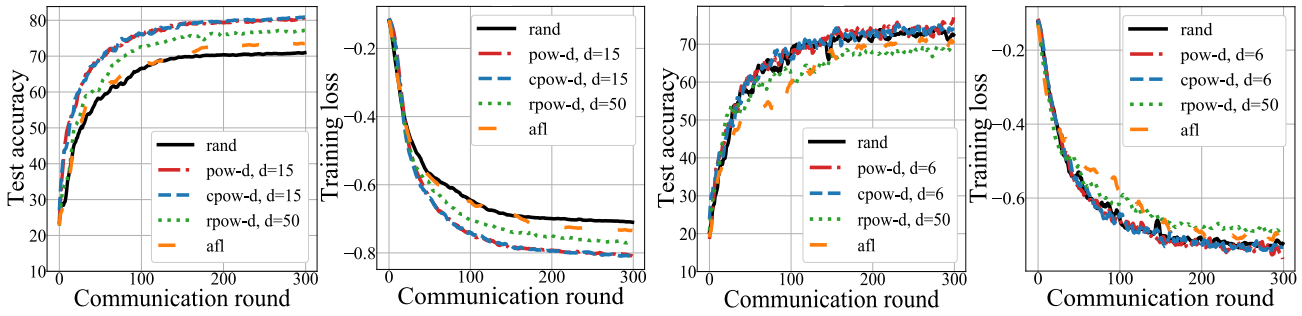
(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

Figure 14: Test accuracy and training loss for π_{rand}, $\pi_{\text{pow-d}}$, $\pi_{\text{cpow-d}}$, $\pi_{\text{rpow-d}}$, and π_{afl} for $K = 100$, $C = 0.03$ on the FMNIST dataset with mini-batch size $b = 128$ and $\tau = 30$.



(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

Figure 15: Test accuracy and training loss for π_{rand}, $\pi_{\text{pow-d}}$, and π_{afl} for $K = 100$, $C = 0.03$ on the FMNIST dataset with mini-batch size $b = 64$ and $\tau = 100$.



(a) Test accuracy and training loss for $\alpha = 2$ (b) Test accuracy and training loss for $\alpha = 0.3$

Figure 16: Test accuracy and training loss for π_{rand}, $\pi_{\text{pow-d}}$, $\pi_{\text{cpow-d}}$, $\pi_{\text{rpow-d}}$, and π_{afl} for $K = 100$, $C = 0.03$ on the FMNIST dataset with mini-batch size $b = 64$ and $\tau = 100$.