Federated Learning with Regularized Client Participation

Grigory Malinovsky [1] **Samuel Horváth** [2] **Konstantin Burlachenko** [1 3] **Peter Richtárik** [1]

Abstract

Federated Learning (FL) is a distributed machine learning approach where multiple clients work together to solve a machine learning task. One of the key challenges in FL is the issue of partial participation, which occurs when a large number of clients are involved in the training process. The traditional method to address this problem is randomly selecting a subset of clients at each communication round. In our research, we propose a new technique and design a novel regularized client participation scheme. Under this scheme, each client joins the learning process every R communication rounds, which we refer to as a meta epoch. We have found that this participation scheme leads to a reduction in the variance caused by client sampling. Combined with the popular FedAvg algorithm (McMahan et al., 2017), it results in superior rates under standard assumptions. For instance, the optimization term in our main convergence bound decreases linearly with the product of the number of communication rounds and the size of the local dataset of each client, and the statistical term scales with step size quadratically instead of linearly (the case for client sampling with replacement), leading to better convergence rate $\mathcal{O}(1/T^2)$ compared to $\mathcal{O}(1/T)$, where T is the total number of communication rounds. Furthermore, our results permit arbitrary client availability as long as each client is available for training once per each meta epoch. Finally, we corroborate our results with experiments.

[1]King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia [2]Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, United Arab Emirates [3]Center of Excellence in Data Science and Artificial Intelligence (SDAIA-KAUST AI), Thuwal, Saudi Arabia. Correspondence to: Grigory Malinovsky <grigorii.malinovskii@kaust.edu.sa>, Samuel Horváth < samuel.horvath@mbzuai.ac.ae>.

Workshop of Federated Learning and Analytics in Practice, colocated with 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. Copyright 2023 by the author(s).

1. Introduction

Federated learning (FL) aims to train models in a decentralized manner, preserving the privacy of client personal data by leveraging local computational capabilities. Clients' data never leave their devices; instead, the clients communicate with a server via updates intended for immediate aggregation to train a global model. Due to such advantages and promises, FL is now deployed in a variety of applications (Hard et al., 2018; Apple, 2019) and is a promising direction for smart healthcare, where privacy is of an essential importance (Rieke et al., 2020; Sheller et al., 2020).

In this paper, we consider the standard FL problem formulation of solving an empirical risk minimization problem over the data available from all participating clients, i.e.,

$$\min_{x\in\mathbb{R}^d}\left[f(x) \stackrel{\text{def}}{=} \frac{1}{M}\sum_{m=1}^{M} f_m(x)\right], \tag{1}$$

where

$$f_m(x) \stackrel{\text{def}}{=} \frac{1}{N}\sum_{j=1}^{N} f_m^j(x).$$

The term f_m^j corresponds to the local loss of the current model parameterized by $x \in \mathbb{R}^d$ evaluated for the j-th data point on the dataset belonging to the m-th client. [1]

We assume that there is a large number of small clients. This setting is often referred to as cross-device FL. Cross-device FL leverages edge devices such as mobile phones and different Internet of Things (IoT) devices to exploit data distributed over potentially millions of data sources (Bonawitz et al., 2017; Bhowmick et al., 2018; Niu et al., 2020). As such, it brings unique challenges compared to standard distributed learning. For instance, optimization methods must contend with issues related to edge computing (Lim et al., 2020; Xia et al., 2021), participant selection (Yang et al., 2021; Chen et al., 2022; Cho et al., 2020; Ribero & Vikalo, 2020), system heterogeneity (Diao et al., 2020; Horváth et al., 2021) and communication constraints such as low network bandwidth and high latency (Arjevani & Shamir,

[1]Although we focus on the setting, where each client has the same amount of data, our results can be extended using techniques introduced by Mishchenko et al. (2021).

2015; McMahan et al., 2017; Stich, 2018; Yu et al., 2019; Horváth & Richtárik, 2020).

In this work, we focus on *partial participation*, i.e., clients only intermittently participate in the collaborative training process (Bonawitz et al., 2017). As previously discussed, in cross-device FL, the number of participating devices M can be in the order of millions. At this scale, *client sampling* (i.e., using a subset of clients for each update) is a necessity since it is impractical for all devices to participate in every round because it would require computation and communication that can consume a large amount of energy and also lead to network congestion. In addition, clients are only sometimes available. For example, suppose the client devices are smartphones. In that case, they may be willing to participate in FL only when they are charging and connected to a high-speed network (usually during night hours) to avoid draining the battery and creating a negative user experience. Finally, each device may only participate once or a few times during the entire training process, so *stateless* methods (which do not rely on each client maintaining and updating local state throughout training) are particularly interesting.

A typical approach to limit the number of clients participating in each is to employ uniform sampling (McMahan et al., 2017). In each round, the orchestrator (global server) picks C clients sampled uniformly at random that perform local training. A more general approach is to select clients with a given importance-sampling-based probability distribution that is independent across rounds (Fraboni et al., 2021a;b; Chen et al., 2022; Horváth et al., 2022). In this work, we introduce a novel client sampling strategy, where we do not sample clients independently. Instead, we propose a regularized participation strategy, where each client participates once during a period we refer to as a meta-epoch. Our motivation stems from centralized training, where it is now well-understood that random reshuffling, i.e., data sampling without replacement, has a variance-reducing effect (Mishchenko et al., 2020). Therefore, we propose to apply the random reshuffling procedure at the client level. We discuss random reshuffling and client sampling in detail in the related work section.

2. Contributions

The key contributions of our work are the following

- We design a novel client participation strategy based on regularized participation, where each client participates once during each meta epoch.
- We combine the proposed client selection scheme with the FedAvg-like method that, apart from partial participation, incorporates local steps, local dataset shuffling, and server and client step sizes. We refer to this new method as RR-CLI. We provide rigorous convergence guarantees and show that in the considered setups, our results give state-of-the-art performance by providing the best scaling in terms of both the linearly decreasing optimization term and the statistical term proportional to squared step sizes.
- The theoretical analysis is corroborated by the experimental evaluation that validates our findings.

3. Description of Algorithm

The backbone of the proposed algorithm is based on the celebrated FedAvg (McMahan et al., 2017) further inspired by recent advances (Horváth et al., 2022; Malinovsky et al., 2021a). Our method combines previously considered local steps, local dataset reshuffling, and server and client step sizes, but, in addition, we also introduce regularized client participation, i.e., sampling without replacement of clients, to the algorithm. This extra feature is the main algorithmic contribution of our work.

Each meta epoch t starts with the partitioning of all M clients into M/C cohorts $S_t = \left\{S_t^{\lambda_r}\right\}_{r=0}^{R-1}$, each $S_t^{\lambda_r}$ with size C. These cohorts are either obtained using the without-replacement sampling of clients, i.e., the outer loop of the double shuffling procedure or given by client availability. In our main theoretical part, we assume cohorts S_t to be obtained using the without-replacement sampling of clients, but we also provide an extension that works with any (including deterministic) reshuffling. Client shuffling could be the same or resampled for each meta epoch, our theory handles both cases, and they lead to the same convergence bound.

We also use permutations locally, i.e., shuffling of local client's data points, which corresponds to the inner loop of the double shuffling procedure. For both permutations, we admit two options. Either we sample one permutation at the beginning that is used in each step, we call this option Shuffle-Once, or we resample new permutations in each meta epoch, which we refer to as Random Reshuffling. Similarly to the previous case, both options lead to the same convergence bound.

Each meta epoch contains $R = M/C$ communication rounds. For each communication round r, the server sends the current model estimate x_t^r to clients, which belong to cohort $S_t^{\lambda_r}$. Each client $m \in S_t^{\lambda_r}$ sets $x_{m,t}^{r,0} = x_t^r$ and proceeds with N local steps using permutation of datapoints π_m^j, i.e.,

$$x_{m,t}^{r,j+1} = x_{m,t}^{r,j} - \gamma \nabla f_m^{\pi_m^j}\left(x_{m,t}^{r,j}\right).$$

Once the local epoch (N local steps) is finished, each client

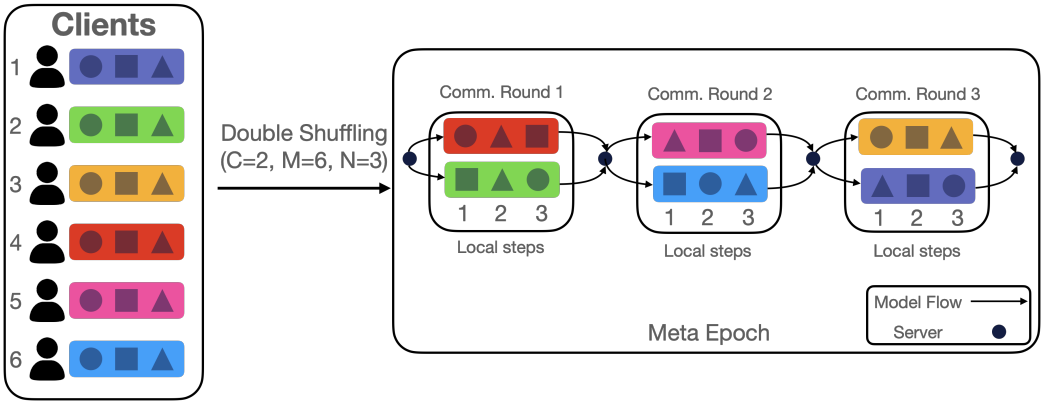

Figure 1: Visualization of double-shuffling procedure for 6 clients, each with 3 datapoints. Two clients are sampled in each communication round.

$m \in S_t^{\lambda_r}$ forms the following local gradient estimator

$$g_{m,t}^r = \frac{1}{\gamma N}\left(x_t^r - x_{m,t}^{r,N}\right).$$

All clients in the cohort $S_t^{\lambda_r}$ send their estimated directions $g_{m,t}^r$ to the server, and these updates are aggregated using averaging operator

$$g_t^r = \frac{1}{C}\sum_{m \in S_t^{\lambda_r}} g_{m,t}^r.$$

The aggregated gradient estimator g_t^r is used to perform the server-side step, which is taken from the initial point x_t^r

$$x_t^{r+1} = x_t^r - \eta g_t^r.$$

Note that if we set $\eta = \gamma N$, then we have that the updated model x_t^{r+1} equal to the average of models on the clients from the current cohort $S_t^{\lambda_r}$.

$$\begin{aligned} x_t^{r+1} &= x_t^r - \eta g_t^r = x_t^r - \eta \frac{1}{C}\sum_{m \in S_t^{\lambda_r}} g_{m,t}^r \\ &= x_t^r - \eta \frac{1}{C}\sum_{m \in S_t^{\lambda_r}} \frac{1}{\gamma N}\left(x_t^r - x_{m,t}^{r,N}\right) \\ &= x_t^r - \frac{\eta}{\gamma\mu}\frac{1}{C}\sum_{m \in S_t^{\lambda_r}} x_t^r + \frac{\eta}{\gamma\mu}\frac{1}{C}\sum_{m \in S_t^{\lambda_r}} x_{m,t}^{r,N} \\ &= \frac{1}{C}\sum_{m \in S_t^{\lambda_r}} x_{m,t}^{r,N}. \end{aligned}$$

Similar to the local update, the server constructs the global update $\frac{x_t - x_t^R}{\eta R}$ at the end of the meta epoch t, when all clients participated and apply this update for its global step to obtain new estimator in the form

$$x_{t+1} = x_t - \gamma \frac{x_t - x_t^R}{\eta R}.$$

Analogously to server-side steps after local epochs, if we set $\gamma = \eta R$, then the new model x_{t+1} is equal to x_t^R

$$\begin{aligned} x_{t+1} &= x_t - \gamma \frac{x_t - x_t^R}{\eta R} \\ &= x_t - \eta R \frac{x_t - x_t^R}{\eta R} \\ &= x_t - x_t + x_t^R = x_t^R. \end{aligned}$$

The pseudocode for this procedure is provided in Algorithm 1. The introduction of extra global step sizes η and γ is a useful algorithmic trick that will enable faster rates in scenarios, where we can't directly analyze local improvement, e.g., see (Karimireddy et al., 2019a). Our main result (Theorem C.1) does not require this trick and provides the fastest convergence guarantees.

Due to space limitation all theoretical results are presented in the appendix.

4. Numerical experiments

We conduct numerical experiments in which we analyzed how RR-CLI algorithm (Algorithm 1) compares with its closest competitors, namely NASTYA (Malinovsky et al., 2021a) and FED-AVG (McMahan et al., 2017). For the shuffling, we use the Shuffle Once option.

4.1. Computing and software environment

We performed experiments on a cluster with nodes running CentOS Linux release 7.9.2009 and Linux Kernel 3.10.0-1160 x86_64. Each experiment runs on a compute node with an NVIDIA GPU and 40 GBytes of virtual memory for the Python interpreter. We use double precision (fp64) float point arithmetic. The distributed environment within each experiment is simulated using Python software suite `FL_PyTorch` (Burlachenko et al., 2021).

Algorithm 1 RR-CLI: Federated optimization with server step sizes and global step sizes, random shuffling and partial participation with shuffling

Input: client step size $\gamma > 0$; server step size $\eta > 0$; global step size $\theta > 0$; cohort size $C \in \{1, 2, \ldots, M\}$; number of rounds $R = M/C$; initial iterate/model $x_0 \in \mathbb{R}^d$; number of epochs $T \geq 1$
Client-Shuffle-Once option: sample a permutation $\lambda = (\lambda_0, \lambda_1, \ldots, \lambda_R)$ of $[R]$
Data-Shuffle-Once option: for each client m, sample a permutation $\pi_m = (\pi_m^0, \pi_m^1, \ldots, \pi_m^{N-1})$ of $[N]$
for meta-epoch $t = 0, 1, \ldots, T-1$ **do**
  Client-Reshuffling option: sample a permutation $\lambda = (\lambda_0, \lambda_1, \ldots, \lambda_R)$ of $[R]$
  for communication rounds $r = 0, \ldots, R-1$ **do**
    Send model x_t^r to all participating clients $m \in S_t^{\lambda_r}$ (server broadcasts x_t^r to all clients $m \in S_t^{\lambda_r}$)
    for all clients $m \in S_t^{\lambda_r}$, locally in parallel **do**
      $x_{m,t}^{r,0} = x_t^r$ (client m initializes local training using the latest global model x_t^r)
      Data-Random-Reshuffling option: sample a permutation $\pi_m = (\pi_m^0, \pi_m^1, \ldots, \pi_m^{N-1})$ of $[N]$
      for all local training data points $j = 0, 1, \ldots, N-1$ **do**
        $x_{m,t}^{r,j+1} = x_{m,t}^{r,j} - \gamma \nabla f_m^{\pi_m^j}(x_{m,t}^{r,j})$ (client m makes one pass over its local training data in the order dictated by π_m)
      end for
      $g_{m,t}^r = \frac{1}{\gamma n}(x_t^r - x_{m,t}^{r,N})$ (client m computes local update direction $g_{m,t}$)
    end for
    $g_t^r = \frac{1}{C} \sum_{m \in S_t^{\lambda_r}} g_{m,t}^r$ (server aggregates the local update directions $g_{m,t}$ discovered by the cohort S_t of clients)
    $x_t^{r+1} = x_t^r - \eta g_t^r$ (server updates the model using the aggregated direction g_t and applying server step size η)
  end for
  $x_{t+1} = x_t - \theta \frac{x_t - x_t^R}{\eta R}$ global step after all communication rounds during meta-epoch
end for

4.2. Optimization problem and experimental setup

We consider the following formulation. Each $f_m(x)$ is in the form of empirical risk minimization objective for logistic loss with additive L_2 regularization term for local data D_i

$$f_m(x) = \frac{1}{N} \sum_{j=1}^{N} l_m^j(x, a^{(j)}, b^{(j)}) + \frac{\alpha}{2}\|x\|^2,$$

$$l_m^j(x, a, b) = \log(1 + \exp(-b_m^{(j)} \cdot x^\top a_m^{(j)})).$$

For our experiments, we have considered three LIBSVM datasets (Chang & Lin, 2011) – `phishing`, `w3a` and `a3a`. To distribute the dataset across clients, we reshuffle this dataset D uniformly and split it into $M = 12$ clients so that each client has $N = \lfloor \frac{|D|}{M} \rfloor$ data samples. Along different runs, we have fixed the split of data across the clients and the initial iterate x_0. In the main part, we include only the `phishing` dataset. The remaining experiments are provided in Appendix J.

Since we use logistic regression, we can exactly compute the smoothness parameters. We set the regularizer $\alpha = 5e-4$. This results in having the condition number $\kappa \approx 10^4$ for `phishing`. Quantities of our interest are the rate of decreasing distance to the optimal point $\|x_k - x_\star\|^2$ and functional gap $f(x_k) - f(x_\star)$. We have pre-computed numerically $x_\star$ such that $\|\nabla f(x_\star)\| \leq 10^{-14}$. All our experiments involve 5 independent runs to obtain estimates of $\mathbb{E}[f(x_k) - f^\star]$ and $\mathbb{E}[\|x_k - x_\star\|^2]$.

4.3. Theoretical step sizes

Firstly, we use theoretical step sizes for all methods described in the original papers, except FED-AVG for which we have used theoretical step size from more recent paper (Karimireddy et al., 2019b). The FED-AVG uses uniform sampling without replacement with 10% of local data points sampled in each local step. Both FED-AVG and NASTYA sample 3 clients from a total of $n = 12$ clients in each round uniformly at random. The RR-CLI samples clients using the random reshuffling procedure with 4 cohorts, each with 3 clients. All algorithms make 10 local steps before communication with the central server. This makes all the algorithms equivalent in terms of local computations and communication. Since different algorithms use different sampling strategies, we measure the number of epochs, where one epoch is the amount of computation equivalent to the computation of the full gradient across the whole dataset. Our results are presented in Figure 4. One can note that RR-CLI outperforms both baselines. Firstly, as theory predicts, RR-CLI dominates NASTYA due to smaller variance from partial participation since the variance terms scaled with γ^2 (RR-CLI) instead of γ (NASTYA). FED-AVG exhibit poor behaviour its theory only permits small

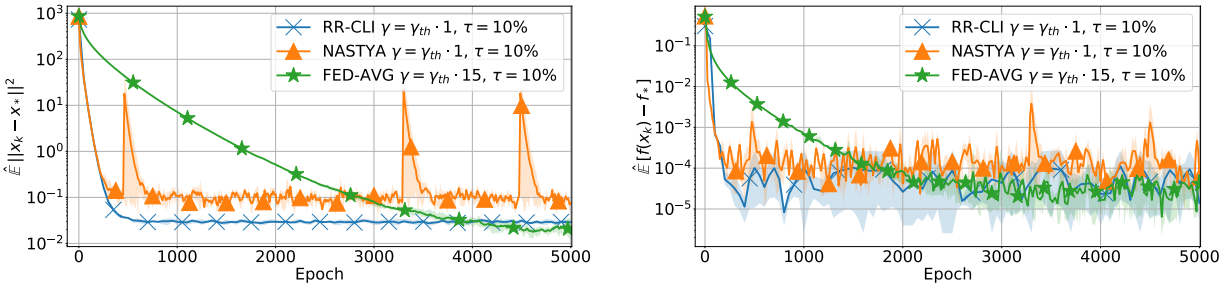

Figure 2: Training `Logistic Regression` on `phishing` with $n = 12$ clients. Theoretical global step size and tune local step sizes. Partial participation with 3 clients per round with 10 local steps.

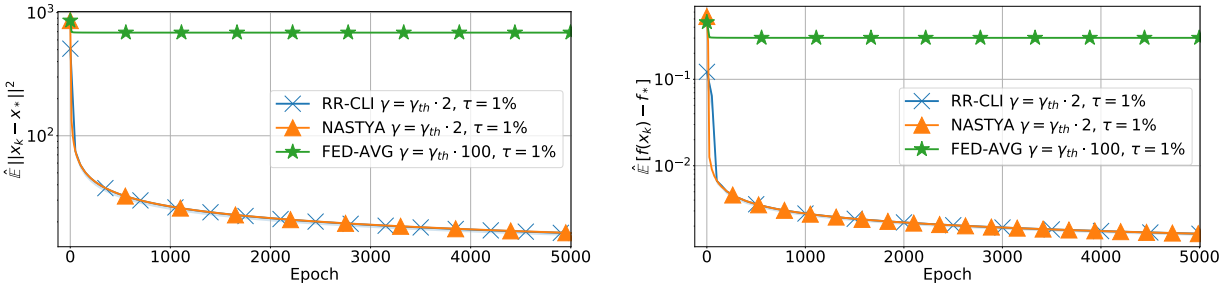

Figure 3: Training `Logistic Regression` on `phishing` with $n = 12$ clients with heoretical global step size and tuned local step sizes. Partial participation with 3 clients per round, 100 local step. Local and global step size are decaying $\propto \frac{1}{1+\#\text{passedepochs}}$. Local gradient estimators are computed with 1% of local samples.

step size $\gamma = \dfrac{1}{L + \alpha}$.

4.4. Tuned step sizes

Since we provide the worst-case analysis, the step size estimates might be too conservative. The goal of this ablation is to analyze how much we can increase local step sizes in practice. To test this, we consider the list of multipliers for theoretical local step size: $\{1, 2, 3, 5, 10, 12, 15\}$. In Figure 2, we showcase a comparison of the three considered algorithms using the tuned local step sizes. We note that RR-CLI still outperforms both baselines.

4.5. Tuned step sizes with decaying

In the last experiment, we consider decaying local and global step size with factor $\propto \frac{1}{\#\text{passed epochs}+1}$. In this experiment, we again analyze several possible multipliers of theoretical step size, concretely $\{1, 2, 3, 5, 10, 12, 15, 50, 100\}$. In this experiment, we increase the number of local steps to 100 by decreasing the local batch size by a factor of 10. Results are presented in Figure 3. We note that both NASTYA and RR-CLI follow a similar trend, while FED-AVG lacks in performance.

5. Conclusion and Future Work

In conclusion, we propose a new technique for addressing the issue of partial participation in Federated Learning. We design a novel regularized client participation scheme. By having each client join the learning process once during each meta epoch, the proposed scheme leads to a reduction in the variance caused by client sampling that is reflected both in theoretical analysis and practical performance.

For future work, we are interested in combining the proposed algorithm with momentum or different adaptive methods to make it more practical.

References

Ahn, K. and Sra, S. On tight convergence rates of without-replacement SGD. *arXiv preprint arXiv:2004.08657*, 2020.

Ahn, K., Yun, C., and Sra, S. Sgd with shuffling: optimal rates without component convexity and large epoch requirements. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 17526–17535. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/cb8acb1dc9821bf74e6ca9068032d623-Paper.pdf.

Ajalloeian, A. and Stich, S. U. On the convergence of sgd with biased gradients. *arXiv preprint arXiv:2008.00051*, 2020.

Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. QSGD: Communication-efficient SGD via gradient quantization and encoding. *Advances in Neural Information Processing Systems*, 30:1709–1720, 2017.

Apple. Designing for privacy (video and slide deck). Apple WWDC, https://developer.apple.com/videos/play/wwdc2019/708, 2019.

Arjevani, Y. and Shamir, O. Communication complexity of distributed convex learning and optimization. *arXiv preprint arXiv:1506.01900*, 2015.

Beznosikov, A. and Takáč, M. Random-reshuffled sarah does not need a full gradient computations. *arXiv preprint arXiv:2111.13322*, 2021.

Bhowmick, A., Duchi, J., Freudiger, J., Kapoor, G., and Rogers, R. Protection against reconstruction and its applications in private federated learning. *arXiv preprint arXiv:1812.00984*, 2018.

Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., and Seth, K. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1175–1191, 2017.

Bottou, L. Curiously fast convergence of some stochastic gradient descent algorithms. In *Proceedings of the symposium on learning and data science, Paris*, volume 8, pp. 2624–2633, 2009.

Bottou, L. Stochastic gradient descent tricks, volume 7700 of lecture notes in computer science (lncs), 2012.

Burlachenko, K., Horváth, S., and Richtárik, P. Fl_pytorch: optimization research simulator for federated learning. In *Proceedings of the 2nd ACM International Workshop on Distributed Machine Learning*, pp. 1–7, 2021.

Chang, C. C. and Lin, C. J. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2011.

Chen, W., Horváth, S., and Richtárik, P. Optimal client sampling for federated learning. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=8GvRCWKHIL.

Cho, Y. J., Wang, J., and Joshi, G. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. *arXiv preprint arXiv:2010.01243*, 2020.

Diao, E., Ding, J., and Tarokh, V. Heterofl: Computation and communication efficient federated learning for heterogeneous clients. *arXiv preprint arXiv:2010.01264*, 2020.

Fraboni, Y., Vidal, R., Kameni, L., and Lorenzi, M. Clustered sampling: Low-variance and improved representativity for clients selection in federated learning. In *International Conference on Machine Learning*, pp. 3407–3416. PMLR, 2021a.

Fraboni, Y., Vidal, R., Kameni, L., and Lorenzi, M. On the impact of client sampling on federated learning convergence. *arXiv preprint arXiv:2107.12211*, 2021b.

Grudzień, M., Malinovsky, G., and Richtárik, P. Can 5th generation local training methods support client sampling? yes! *arXiv preprint arXiv:2212.14370*, 2022.

Gu, X., Huang, K., Zhang, J., and Huang, L. Fast federated learning in the presence of arbitrary device unavailability. *Advances in Neural Information Processing Systems*, 34: 12052–12064, 2021.

Gürbüzbalaban, M., Ozdaglar, A., and Parrilo, P. A. Why random reshuffling beats stochastic gradient descent. *Mathematical Programming*, 186(1):49–84, 2021.

Hanzely, F. and Richtárik, P. Federated learning of a mixture of global and local models. *arXiv:2002.05516*, 2020.

Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., and Ramage, D. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*, 2018.

Horváth, S. and Richtárik, P. A better alternative to error feedback for communication-efficient distributed learning. *arXiv preprint arXiv:2006.11077*, 2020.

Horváth, S., Ho, C.-Y., Horváth, L., Sahu, A. N., Canini, M., and Richtárik, P. Natural compression for distributed deep learning. *arXiv preprint arXiv:1905.10988*, 2019.

Horváth, S., Laskaridis, S., Almeida, M., Leontiadis, I., Venieris, S., and Lane, N. D. FjORD: Fair and accurate federated learning under heterogeneous targets with ordered dropout. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=4fLr7H5D_eT.

Horváth, S., Sanjabi, M., Xiao, L., Richtárik, P., and Rabbat, M. Fedshuffle: Recipes for better use of local work in federated learning. *Transactions of Machine Learning Research*, 2022. URL https://openreview.net/forum?id=Lgs5pQ1v30.

Huang, J., Qian, F., Guo, Y., Zhou, Y., Xu, Q., Mao, Z. M., Sen, S., and Spatscheck, O. An in-depth study of LTE: Effect of network protocol and application behavior on performance. *SIGCOMM Computer Communication Review*, 43:363–374, 2013.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S. J., Stich, S. U., and Suresh, A. T. Scaffold: Stochastic controlled averaging for on-device federated learning. 2019a.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S. J., Stich, S. U., and Theertha Suresh, A. SCAFFOLD: Stochastic controlled averaging for federated learning. *arXiv e-prints*, pp. arXiv–1910, 2019b.

Khaled, A., Mishchenko, K., and Richtárik, P. Tighter theory for local SGD on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pp. 4519–4529. PMLR, 2020.

Koloskova, A., Loizou, N., Boreiri, S., Jaggi, M., and Stich, S. A unified theory of decentralized sgd with changing topology and local updates. In *International Conference on Machine Learning*, pp. 5381–5393. PMLR, 2020.

Konečný, J. and Richtárik, P. Randomized distributed mean estimation: Accuracy vs. communication. *Frontiers in Applied Mathematics and Statistics*, 4:62, 2018.

Lai, F., Zhu, X., Madhyastha, H. V., and Chowdhury, M. Oort: Efficient federated learning via guided participant selection. In *15th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 21)*, pp. 19–35, 2021.

Lim, W. Y. B., Luong, N. C., Hoang, D. T., Jiao, Y., Liang, Y.-C., Yang, Q., Niyato, D., and Miao, C. Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22(3): 2031–2063, 2020.

Lin, T., Stich, S. U., Patel, K. K., and Jaggi, M. Don't use large mini-batches, use local SGD. *arXiv preprint arXiv:1808.07217*, 2018.

Luo, B., Xiao, W., Wang, S., Huang, J., and Tassiulas, L. Tackling system and statistical heterogeneity for federated learning with adaptive client sampling. In *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*, pp. 1739–1748. IEEE, 2022.

Malinovskiy, G., Kovalev, D., Gasanov, E., Condat, L., and Richtarik, P. From local SGD to local fixed-point methods for federated learning. In *International Conference on Machine Learning*, pp. 6692–6701. PMLR, 2020.

Malinovsky, G. and Richtárik, P. Federated random reshuffling with compression and variance reduction. 2022.

Malinovsky, G., Mishchenko, K., and Richtárik, P. Server-side stepsizes and sampling without replacement provably help in federated optimization. *arXiv preprint arXiv:2201.11066*, 2021a.

Malinovsky, G., Sailanbayev, A., and Richtárik, P. Random reshuffling with variance reduction: New analysis and better rates. *arXiv preprint arXiv:2104.09342*, 2021b.

Malinovsky, G., Yi, K., and Richtárik, P. Variance reduced proxskip: Algorithm, theory and application to federated learning. *arXiv preprint arXiv:2207.04338*, 2022.

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

Mishchenko, K., Hanzely, F., and Richtárik, P. 99% of parallel optimization is inevitably a waste of time. *arXiv preprint arXiv:1901.09437*, 2019.

Mishchenko, K., Khaled Ragab Bayoumi, A., and Richtárik, P. Random reshuffling: Simple analysis with vast improvements. *Advances in Neural Information Processing Systems*, 33, 2020.

Mishchenko, K., Khaled, A., and Richtárik, P. Proximal and federated random reshuffling. *arXiv preprint arXiv:2102.06704*, 2021.

Mishchenko, K., Malinovsky, G., Stich, S., and Richtárik, P. Proxskip: Yes! local gradient steps provably lead to communication acceleration! finally! *arXiv preprint arXiv:2202.09357*, 2022.

Nguyen, H. T., Sehwag, V., Hosseinalipour, S., Brinton, C. G., Chiang, M., and Poor, H. V. Fast-convergent federated learning. *IEEE Journal on Selected Areas in Communications*, 39(1):201–218, 2020.

Niu, C., Wu, F., Tang, S., Hua, L., Jia, R., Lv, C., Wu, Z., and Chen, G. Billion-scale federated learning on mobile clients: A submodel design with tunable privacy. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, pp. 1–14, 2020.

Ramezani-Kebrya, A., Faghri, F., and Roy, D. M. NUQSGD: Improved communication efficiency for data-parallel SGD via nonuniform quantization. *arXiv preprint arXiv:1908.06077*, 2019.

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

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., Bakas, S., Galtier, M. N., Landman, B. A., Maier-Hein, K., et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):1–7, 2020.

Ruan, Y., Zhang, X., Liang, S.-C., and Joe-Wong, C. Towards flexible device participation in federated learning for non-iid data. *arXiv preprint arXiv:2006.06954*, 2020.

Sadiev, A., Malinovsky, G., Gorbunov, E., Sokolov, I., Khaled, A., Burlachenko, K., and Richtárik, P. Federated optimization algorithms with random reshuffling and gradient compression. *arXiv preprint arXiv:2206.07021*, 2022.

Sheller, M. J., Edwards, B., Reina, G. A., Martin, J., Pati, S., Kotrotsou, A., Milchenko, M., Xu, W., Marcus, D., Colen, R. R., et al. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific reports*, 10(1):1–12, 2020.

Stich, S. U. Local SGD converges fast and communicates little. *arXiv preprint arXiv:1805.09767*, 2018.

Stich, S. U. and Karimireddy, S. P. The error-feedback framework: Better rates for SGD with delayed gradients and compressed communication. *ICLR 2020 - International Conference on Learning Representations*, 2020.

Stich, S. U., Cordonnier, J.-B., and Jaggi, M. Sparsified SGD with memory. In *Advances in Neural Information Processing Systems*, pp. 4447–4458, 2018.

Van Berkel, C. Multi-core for mobile phones. In *Conference on Design, Automation and Test in Europe*, 2009.

Vogels, T., Karimireddy, S. P., and Jaggi, M. PowerSGD: Practical low-rank gradient compression for distributed optimization. 2019.

Wang, S. and Ji, M. A unified analysis of federated learning with arbitrary client participation. *arXiv preprint arXiv:2205.13648*, 2022.

Wangni, J., Wang, J., Liu, J., and Zhang, T. Gradient sparsification for communication-efficient distributed optimization. In *Advances in Neural Information Processing Systems*, pp. 1299–1309, 2018.

Wen, W., Xu, C., Yan, F., Wu, C., Wang, Y., Chen, Y., and Li, H. Terngrad: Ternary gradients to reduce communication in distributed deep learning. *arXiv preprint arXiv:1705.07878*, 2017.

Xia, Q., Ye, W., Tao, Z., Wu, J., and Li, Q. A survey of federated learning for edge computing: Research problems and solutions. *High-Confidence Computing*, 1(1):100008, 2021.

Yan, Y., Niu, C., Ding, Y., Zheng, Z., Wu, F., Chen, G., Tang, S., and Wu, Z. Distributed non-convex optimization with sublinear speedup under intermittent client availability. *arXiv preprint arXiv:2002.07399*, 2020.

Yang, H., Fang, M., and Liu, J. Achieving linear speedup with partial worker participation in non-iid federated learning. *arXiv preprint arXiv:2101.11203*, 2021.

Yang, H., Zhang, X., Khanduri, P., and Liu, J. Anarchic federated learning. In *International Conference on Machine Learning*, pp. 25331–25363. PMLR, 2022.

Yu, H., Yang, S., and Zhu, S. Parallel restarted SGD with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 5693–5700, 2019.

Yun, C., Rajput, S., and Sra, S. Minibatch vs local SGD with shuffling: Tight convergence bounds and beyond. *arXiv preprint arXiv:2110.10342*, 2021.

Zhang, H., Li, J., Kara, K., Alistarh, D., Liu, J., and Zhang, C. Zipml: Training linear models with end-to-end low precision, and a little bit of deep learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 4035–4043. JMLR. org, 2017.

Appendix

A. Related Work

Cross-device FL can be hindered by communication costs, as edge devices such as mobile phones and IoT devices often have limited bandwidth and connectivity (Van Berkel, 2009; Huang et al., 2013). These limitations can make wireless connections and internet connections expensive and unreliable. Additionally, limitations in the capacity of the aggregating master and other FL system factors can restrict the number of clients allowed to participate in each communication round. To address these issues, there is significant interest in reducing the communication bandwidth of FL systems through techniques such as local updates, communication compression, and client selection methods. Our work primarily focuses on client selection techniques, but it is worth noting that these approaches can be combined to achieve a more effective outcome.

A.1. Local Methods

This strategy involves reducing the frequency of communication and emphasising local computation, where each device performs multiple local steps before communicating its updates back to the central node. A prototypical method in this category is the Federated Averaging (`FedAvg`) algorithm (McMahan et al., 2017), an adaption of local-update to parallel `SGD`, where each client runs some predefined number of `SGD` steps based on its local data before local updates are averaged to form the global pseudo-gradient update for the global model on the master node. Recently, there has been significant interest and attempts to provide theoretical guarantees for this method, or its variants (Stich, 2018; Lin et al., 2018; Karimireddy et al., 2019a; Stich & Karimireddy, 2020; Khaled et al., 2020; Hanzely & Richtárik, 2020; Malinovskiy et al., 2020; Koloskova et al., 2020; Mishchenko et al., 2022; Malinovsky et al., 2022) as the original work was a heuristic, offering no theoretical guarantees.

A.2. Communication Compression Methods

Another popular technique works by reducing the size of the updates communicated from clients to the master. This approach is referred to as communication compression. In this approach, instead of transmitting the full-dimensional update vector $g \in \mathbb{R}^d$, each client only transmits a compressed vector $\mathcal{C}(g)$, where $\mathcal{C} : \mathbb{R}^d \to \mathbb{R}^d$ is a (possibly random) operator chosen such that $\mathcal{C}(g)$ can be represented using fewer bits than g, for instance, by using limited bit representation (quantization) (Alistarh et al., 2017; Wen et al., 2017; Zhang et al., 2017; Horváth et al., 2019; Ramezani-Kebrya et al., 2019) or by enforcing sparsity (sparsification) (Wangni et al., 2018; Konečný & Richtárik, 2018; Stich et al., 2018; Mishchenko et al., 2019; Vogels et al., 2019).

A.3. Client Sampling/Selection Methods

On top of the uniform (McMahan et al., 2017; Karimireddy et al., 2019a; Grudzień et al., 2022) or arbitrary sampling (Horváth et al., 2022), several proposed approaches focus on a careful selection of the participating clients to improve communication complexity (Cho et al., 2020; Nguyen et al., 2020; Ribero & Vikalo, 2020; Lai et al., 2021; Luo et al., 2022; Chen et al., 2022). These techniques rely on the extra partial information, such as the client's loss or the norms of the updates, to speed up the training by selecting more informative updates. Another stream of works tackles convergence under arbitrary client participation patterns (Yang et al., 2022; Wang & Ji, 2022; Gu et al., 2021; Yan et al., 2020; Ruan et al., 2020). In contrast, our proposed method selects clients using regularized sampling strategy based on client reshuffling. We note that such sampling is not independent across communication rounds and is not arbitrary, i.e., our goal is not to provide bounds for arbitrary participation patterns, as we assume we have access to client sampling to provide a better practical and theoretical sampling strategy. To account for the standard practice of FL training, where clients are only available during certain hours when their device is on charge and connected to the high-speed network, we also provide the convergence rates under non-random deterministic client shuffling that can still guarantee convergence under this challenging scenario; see Remark C.4.

A.4. Random reshuffling

A particularly successful technique to optimize the empirical risk minimization objective is to randomly permute (i.e., reshuffle) the training data at the beginning of every epoch (Bottou, 2012) instead of randomly sampling a data point (or a

subset of them) with replacement at each step, as in the standard analysis of SGD. This process is repeated several times, and the resulting method is usually referred to as Random Reshuffling (RR). RR is often observed to exhibit faster convergence than sampling with replacement, which can be intuitively attributed to the fact that RR is guaranteed to process each training sample exactly once every epoch. In contrast, with-replacement sampling needs more steps than the equivalent of one epoch to see every sample with a high probability. Correctly understanding the random reshuffling trick and why it works has been a challenging open problem (Bottou, 2009; Ahn & Sra, 2020; Gürbüzbalaban et al., 2021) until recent advances in Mishchenko et al. (2020) introduced a significant simplification of the convergence analysis technique. The difficulty of analyzing RR stems from the fact that updates conditioned on the previous iterate result in biased gradient estimates, unlike with-replacement sampling. The subsequent works provide better convergence guarantees for RR in different settings (Ahn et al., 2020; Malinovsky et al., 2021b; Beznosikov & Takáč, 2021). To our knowledge, in terms of FL, RR was only explored in terms of local steps. Initial works (Mishchenko et al., 2021; Yun et al., 2021; Malinovsky & Richtárik, 2022; Sadiev et al., 2022) require full participation in each communication round. The partial participation framework was considered in the following works (Malinovsky et al., 2021a; Horváth et al., 2022), but the authors only consider unbiased client participation. In this work, we fill this missing gap and show that RR employment on the client level leads to superior theoretical and practical performance in FL.

B. Notation and Assumptions

The loss function for client m is made up of individual losses $f_m^j(x)$ for each local data point j, where x is a parameter that we want to optimize. We assume that client i has access to an oracle that, when given input (j, x), returns the gradient $\nabla f_m^j(x)$. We denote $[l] = \{1, 2, \ldots, l\}$ for any $l \in \mathbb{N}$. To show the convergence of our method, we make certain standard assumptions.

Assumption B.1. The function h is μ**-convex** for $\mu \geq 0$; i.e.,

$$\langle \nabla h(x), y - x \rangle \leq -\Big(h(x) - h(y) + \frac{\mu}{2} \|x - y\|^2 \Big) . \tag{2}$$

We say that h is μ-strongly convex if $\mu > 0$, and otherwise h is (general) convex.

Assumption B.2. The function h are L**-smooth**; i.e., there is $L > 0$

$$\|\nabla h(x) - \nabla h(y)\| \leq L \|x - y\| . \tag{3}$$

We also define the Bregman divergence

$$D_h(x, y) \stackrel{\text{def}}{=} h(x) - h(y) - \langle \nabla f(y), x - y \rangle . \tag{4}$$

Next, we proceed with the definition of double shuffling sampling, which plays a key role in our theoretical analysis.

Definition B.3 (Double Shuffling)**.** Let $\tilde{\pi} = \{\tilde{\pi}_1, \tilde{\pi}_2, \ldots, \tilde{\pi}_M\}$ be a random permutation of $[M]$ and $\{\hat{\pi}_m = \{\hat{\pi}_m^1, \hat{\pi}_m^2, \ldots, \hat{\pi}_m^N\}\}_{m=1}^M$ is a set of M independent random permutations of $[N]$. Then the **double-shuffling** procedure $\pi = \{\pi_1, \pi_2, \ldots, \pi_{MN}\}$ is defined as

$$\pi_k = \hat{\pi}^{j_k}_{\tilde{\pi}_{m_k}} \stackrel{\text{def}}{=} \pi_m^j, \quad \forall k \in [MN], \tag{5}$$

where $m_k = \lfloor \frac{k}{N} \rfloor$ and $j_k = k - m_k N$, i.e., j_k and m_k are quotient and remainder of k with respect to M.

For the case of mini-batching with batch size C, we, for simplicity, assume $M = CR$. We first split M clients into C equisized groups $G_1, G_2, \ldots, G_C$ obtained by without-replacement sampling from $[N]$, i.e., $\bigcup_{o=1}^N G_o = [N]$ and $G_i \cap G_j = \emptyset,\ \forall i \neq j$. To obtain C samples in steps $1, \ldots, RN$, we apply double-shuffling within each group.

The visualization of the double-shuffling process is displayed in Figure 1. Equipped with these definitions, we proceed with the proposed algorithm and the main results.

C. Convergence Guarantees

Before proceeding with our theoretical results, we first define the variance quantities that commonly appear in the convergence analysis of stochastic methods

$$\begin{aligned}\tilde{\sigma}_\star^2 &\stackrel{\text{def}}{=} \frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2,\\ \sigma_\star^2 &\stackrel{\text{def}}{=} \frac{1}{MN}\sum_{m=1}^{M}\sum_{j=1}^{N}\left\|\nabla f_m^j(x_\star)\right\|^2,\end{aligned} \tag{6}$$

and the star sequence

$$\begin{aligned}x_\star^0 &= x_{m,\star}^{0,0} = x_\star,\ \forall m \in S_t^{\lambda_0}\\ x_{m,\star}^{r,j+1} &= x_{m,\star}^{r,j} - \gamma\nabla f_m^{\pi_m^j}(x_\star),\ \forall m \in [M],\\ x_\star^{r+1} &= \frac{1}{C}\sum_{m\in S_t^{\lambda_r}} x_{m,\star}^{r,N}.\end{aligned} \tag{7}$$

that corresponds to running our algorithm with the optimal solution $x_\star$ being the starting point and all the local gradients are estimated at $x_\star$. Note that for this sequence, $x_\star^R = x^\star$. Equipped with these definitions, we proceed with the analysis.

We start with the main result, where we assume each function f_m^j to be μ-strongly convex. In this case, we show that the optimization term decreases linearly with the power that is a product of the number of local data points N, the number of communications rounds in each meta-epoch R and the number of meta-epochs T. In addition, due to applying sampling without replacement of both data points and cohorts, the statistical term scales proportionally to the squared step size γ^2. The following theorem formulates the claim.

Theorem C.1. *Suppose each function f_m^j is μ-strongly convex and L-smooth. Then for Algorithm 1 with constant step sizes $\gamma \le \frac{1}{L}$, $\eta = \gamma N$, $\theta = \eta R$, the iterates generated by the Algorithm 1 satisfy*

$$\begin{aligned}\mathrm{E}\left[\|x_T - x_\star\|^2\right] &\le (1-\gamma\mu)^{NRT}\|x_0 - x_\star\|^2\\ &\quad + \frac{2\gamma^2}{\mu}\max_{r,m}\sigma_{m,DS}^2,\end{aligned}$$

where $\sigma_{m,DS}^2 \stackrel{\text{def}}{=} \frac{1}{\gamma^2}\mathrm{E}\left[D_{f_m^{\pi_m^j}}\left(x_{m,\star}^{r,j}, x_\star\right)\right]$.

To our knowledge, this bound is the first result in FL literature which combines exponentially fast decaying in the first term (optimization term), which is proportional to the number of all gradient steps (# of data × # of round × # of meta-epochs) and the second term (variance/statistical term), which is proportional to γ^2. This result is possible due to our careful algorithmic construction that involves sampling without the replacement of both clients and local data points. Let us now analyze the variance term $\sigma_{m,\mathrm{DS}}^2$. Lemma E.4 in the appendix gives the following upper bound

$$\max_{m\in[M]}\sigma_{m,\mathrm{DS}}^2 \le L\left(\frac{MN^2}{2C^2} + 2N^2\right)\tilde{\sigma}_\star^2 + \frac{LN}{2}\sigma_\star^2,$$

which is independent of γ and T. Therefore, the final convergence rate also scales with $\mathcal{O}(1/T^2)$ compared to $\mathcal{O}(1/T)$ to any method that samples clients uniformly at random in each step, e.g., FedAvg (McMahan et al., 2017) and SCAFFOLD (Karimireddy et al., 2019b).

The next set of results is slightly weaker than the theorem above since we analyse problem (1) under the weaker assumptions, where we assume that only local functions f_m are μ-strongly convex while individual loss functions f_m^j are general convex and L-smooth. In this regime, we can't guarantee a linear decrease in each local step. Therefore, the linear part of the convergence result has power, which is the product of the number of communications rounds in meta-epoch R and the number of meta-epochs T. In addition, the linear term depends on server-side step size η. The variance term is decoupled

into two parts. The first part, which is proportional to η^2 (note that this is still quadratic dependence), is related to sampling without replacement of clients. The second part, which is proportional to γ^2 is related to the reshuffling of data points. Due to the biased nature of updates and lack of individual strong convexity, the step size should be significantly small with the condition $\gamma \leq \frac{1}{8NL\sqrt{\kappa}}$. This is not surprising and is consistent with the analysis of biased SGD (Ajalloeian & Stich, 2020). The formal statement of the theorem follows.

Theorem C.2. *Assume that each f_m is μ-strongly convex. Also, assume that each f_m^j is convex and L-smooth. Let $\eta \leq \frac{1}{4L}$ and $\gamma \leq \frac{1}{8NL\sqrt{\kappa}}$, then for iterates generated by Algorithm 1, we have*

$$\begin{aligned}\mathrm{E}\left[\|x_T - x_\star\|^2\right] &\leq (1-\eta\mu)^{RT}\|x_0 - x_\star\|^2 \\ &+ \frac{4}{\mu}\eta^2 \max_{r,m} \sigma_{m,CS}^2 \\ &+ 12\kappa^2\gamma^2 N^2 \sigma_\star^2 + 12\gamma^2\kappa^2 N \sigma_\star^2,\end{aligned}$$

where $\sigma_{m,CS}^2 = \frac{1}{\eta^2}\mathrm{E}\left[D_{f_m}\left(x_\star^r, x_\star\right)\right]$.

Note that the last two terms can be as small as we want by taking γ^2 small enough. For the first two terms, we first give the upper bound on $\sigma_{m,\mathrm{CS}}^2$ using Lemma E.4 from the appendix that gives

$$\max_{m\in[M]} \sigma_{m,\mathrm{CS}}^2 \leq \frac{LM}{2C^2}\tilde{\sigma}_\star^2.$$

which is independent of η or T. Therefore, similar to the prior case, the final convergence rate scales with $\mathcal{O}(1/T^2)$.

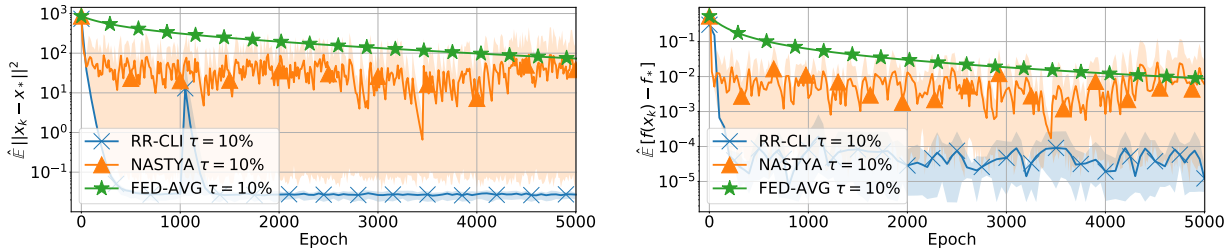

Figure 4: Training `Logistic Regression` on `phishing`, with $n = 12$ clients. Theoretical local and global step sizes, 3 clients per round with 10 local steps.

The final theorem analyzes the most restrictive case when only the function f is μ strongly convex and individual losses are general convex and L-smooth. In this case, the trick with three step sizes is suitable as this helps us to decompose the bound into three parts. The linear part depends on global step size θ, the variance coming from the sampling of data points, which depends on γ, and the variance coming from the sampling of clients, which depends on η. This is summarized in the following theorem.

Theorem C.3. *Suppose that each f_m^j is convex and L-smooth, f is μ-strongly convex. Then provided the step size satisfies $\gamma NR \leq \eta R \leq \theta \leq \frac{1}{16L}$ the final iterate generated by Algorithm 1 satisfies*

$$\begin{aligned}\mathrm{E}\left[\|x_T - x_\star\|^2\right] &\leq \left(1 - \frac{\theta\mu}{2}\right)^T \|x_0 - x_\star\|^2 \\ &+ 16\gamma^2\kappa N^2\sigma_\star^2 \\ &+ 16\gamma^2\kappa N\tilde{\sigma}_\star^2 \\ &+ 16\eta^2 \frac{\kappa}{N^2 R}\frac{M-C}{(M-1)C}\sigma_\star^2.\end{aligned}$$

Note that because of our decomposition, the last three terms can be arbitrarily small by taking γ and η sufficiently small. For the first term, we can take $\theta = \frac{1}{16L}$ to obtain linear convergence.

Remark C.4. All these results can be also adjusted to work with the deterministic client shuffling with the minor change of the rates and convergence analysis; see Section I in the appendix for a detailed discussion.

D. Notation

For the ease of the reader, we provide the table with the used notation below.

Table 1: Notation

Symbol	Description
M	the total number of clients.
N	the total number of data points per client.
C	the cohort size of clients selected for participation.
$R = M/C$	the number of communication rounds during meta-epoch.
r	index of communication round in meta-epoch.
j	index of data point in local epoch.
λ_r	permutation of cohorts for communication round r.
π_j	permutation of data point with index j.
$S_t^{\lambda_r}$	set of C clients in cohort corresponding to communication round r during meta epoch t.
x_t^r	the initial point for round r in epoch t.
$x_{m,t}^{r,j}$	the intermediate point for round r in epoch t on client m and for j-th data point.
g_t^r	the approximation of gradient for round r in epoch t.

E. Variance bounds

In this section, we present bounds that can be used to bound the variance of the gradient estimators used in this work. We start by introducing standard variance decomposition and then presenting two lemmas.

For random variable X and any $y \in \mathbb{R}^d$, the variance can be decomposed as

$$\mathbf{E}\left[\|X - \mathbf{E}[X]\|^2\right] = \mathbf{E}\left[\|X - y\|^2\right] - \|\mathbf{E}[X] - y\|^2. \tag{8}$$

The following lemma bounds the variance of the estimator obtained using the sampling without replacement with respect to both clients and local data points.

Lemma E.1. *Let $\{\{\zeta_m^j\}_{m=1}^M\}_{j=1}^N \in \mathbb{R}^d$ be fixed vectors, and*

$$\tilde{\zeta} \overset{def}{=} \frac{1}{MN}\sum_{m=1}^M\sum_{j=1}^N \zeta_m^j, \qquad \tilde{\zeta}_m \overset{def}{=} \frac{1}{N}\sum_{j=1}^M \zeta_m^j$$

be their averages, and

$$\sigma^2 \overset{def}{=} \frac{1}{NM}\sum_{m=1}^M\sum_{j=1}^N\left\|\zeta_m^j - \tilde{\zeta}\right\|^2, \qquad \tilde{\sigma}^2 \overset{def}{=} \frac{1}{M}\sum_{m=1}^M\left\|\tilde{\zeta}_m - \tilde{\zeta}\right\|^2,$$

be the population variances.

1. *Fix any $k \in \{1, \dots, MN\}$, let $\zeta_{\pi_1}, \dots \zeta_{\pi_k}$ be sampled using double-shuffling procedure (see Definition B.3) from $\{\{\zeta_m^j\}_{j=1}^m\}_{i=1}^n$ and $\tilde{\zeta}_\pi^k$ be their average. Then, the following holds for the sample average and variance*

$$\begin{aligned} \mathbf{E}\left[\tilde{\zeta}_\pi^k\right] &= \tilde{\zeta}, \\ \bar{\sigma}^2(k) \overset{def}{=} \mathbf{E}\left[\left\|\tilde{\zeta}_\pi^k - \tilde{\zeta}\right\|^2\right] &= \frac{j_k(N-j_k)}{k^2(N-1)}\sigma^2 + \left(\frac{(m_k N^2 + j_k^2)(MN-1)}{k^2(N-1)(M-1)} - \frac{N}{k(N-1)} - \frac{1}{M-1}\right)\tilde{\sigma}^2. \end{aligned} \tag{9}$$

2. *Let $N = CR$ and $G_1, G_2, \dots, G_C$ be sets with R elements each, obtained by sampling uniformly at random without replacement from $[N]$, i.e., $\bigcup_{o=1}^b G_o = [N]$ and $G_1 \cap G_j = \{\}$, $\forall i \neq j$. Fix any $k \in \{1, \dots, NR\}$, let $\zeta_{\pi_1}^o, \dots \zeta_{\pi_k}^o$ be*

sampled using double-shuffling procedure (see Definition B.3) from $\{\{\zeta_m^j\}_{j=1}^m\}_{m\in G_o}$ *for* $o \in [R]$. *Let* $k_N \overset{def}{=} \lfloor \frac{k}{N} \rfloor N$ *and*

$$\tilde{\zeta}_{\pi,o}^k \overset{def}{=} \frac{1}{k}\left(\sum_{i=1}^{k_N}\frac{1}{C}\sum_{p=1}^{C}\zeta_{\pi_i}^p + \sum_{i=k_N+1}^{k}\zeta_{\pi_i}^o\right). \tag{10}$$

Then the following holds for the expectation and variance of $\tilde{\zeta}_{\pi,o}^k$

$$\mathbf{E}\left[\tilde{\zeta}_{\pi,o}^k\right] = \tilde{\zeta}, \quad \mathbf{E}\left[\left\|\tilde{\zeta}_{\pi,o}^k - \tilde{\zeta}\right\|^2\right] = \left(\frac{k_N}{k}\right)^2\bar{\sigma}^2(k_N C) + \left(\frac{k-k_N}{k}\right)^2\bar{\sigma}^2(k-k_N) - \frac{2k_N(k-k_N)}{k^2C(M-1)}\tilde{\sigma}^2. \tag{11}$$

Proof. The first claim follows from the linearity of expectation and uniformity of sampling with respect to both permutations. Therefore,

$$\mathbf{E}\left[\tilde{\zeta}_{\pi}^k\right] = \frac{1}{k}\sum_{i=1}^{k}\zeta_{\pi_k} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=1}^{N}\zeta_m^j = \tilde{\zeta}.$$

To prove the second claim, let us first establish the identities for $\mathrm{cov}(\zeta_{\pi_r}, \zeta_{\pi_s})$ for any $r \neq s$. Firstly, we consider the case such that $\lfloor \frac{r}{N} \rfloor = \lfloor \frac{s}{N} \rfloor$. Then,

$$\begin{aligned}
\mathrm{cov}[\zeta_{\pi_r}, \zeta_{\pi_s}] &= \mathbf{E}\left[\left\langle \zeta_{\pi_r} - \tilde{\zeta}, \zeta_{\pi_s} - \tilde{\zeta}\right\rangle\right] = \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N(N-1)}\sum_{k=1}^{N}\sum_{l=1,k\neq l}^{N}\left\langle \zeta_{mk} - \tilde{\zeta}, \zeta_{ml} - \tilde{\zeta}\right\rangle \\
&= \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N(N-1)}\sum_{k=1}^{N}\sum_{l=1}^{N}\left\langle \zeta_{mk} - \tilde{\zeta}, \zeta_{ml} - \tilde{\zeta}\right\rangle - \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N(N-1)}\sum_{k=1}^{N}\left\|\zeta_{mk} - \tilde{\zeta}\right\|^2 \\
&= \frac{1}{MN(N-1)}\sum_{m=1}^{M}\left(N^2\left\|\tilde{\zeta}_m - \tilde{\zeta}\right\| - \sum_{k=1}^{N}\left\|\zeta_{mk} - \tilde{\zeta}\right\|^2\right) \\
&= \frac{1}{N-1}\left(N\tilde{\sigma}^2 - \sigma^2\right)
\end{aligned}$$

For the case, $\lfloor \frac{r}{N} \rfloor \neq \lfloor \frac{s}{N} \rfloor$, we have

$$\begin{aligned}
\mathrm{cov}[\zeta_{\pi_r}, \zeta_{\pi_s}] &= \mathbf{E}\left[\left\langle \zeta_{\pi_r} - \tilde{\zeta}, \zeta_{\pi_s} - \tilde{\zeta}\right\rangle\right] = \frac{1}{M(M-1)}\sum_{m=1}^{M}\sum_{o=1,i\neq o}^{M}\frac{1}{N^2}\sum_{k=1}^{N}\sum_{l=1}^{N}\left\langle \zeta_{mk} - \tilde{\zeta}, \zeta_{ol} - \tilde{\zeta}\right\rangle \\
&= \frac{1}{M(M-1)}\sum_{m=1}^{M}\sum_{o=1}^{M}\frac{1}{N^2}\sum_{k=1}^{N}\sum_{l=1}^{N}\left\langle \zeta_{mk} - \tilde{\zeta}, \zeta_{ol} - \tilde{\zeta}\right\rangle - \frac{1}{M(M-1)}\sum_{m=1}^{M}\frac{1}{N^2}\sum_{k=1}^{N}\sum_{l=1}^{N}\left\langle \zeta_{mk} - \tilde{\zeta}, \zeta_{ml} - \tilde{\zeta}\right\rangle \\
&= \frac{1}{M(M-1)}\frac{1}{N^2}\left\langle \sum_{m=1}^{N}\sum_{k=1}^{N}\zeta_{mk} - \tilde{\zeta}, \sum_{o=1}^{M}\sum_{l=1}^{N}\zeta_{ol} - \tilde{\zeta}\right\rangle - \frac{1}{M(M-1)}\sum_{m=1}^{M}\left\langle \frac{1}{N}\sum_{k=1}^{M}\left(\zeta_{mk} - \tilde{\zeta}\right), \frac{1}{N}\sum_{l=1}^{M}\left(\zeta_{ml} - \tilde{\zeta}\right)\right\rangle \\
&= -\frac{1}{M(M-1)}\sum_{m=1}^{M}\left\|\tilde{\zeta}_m - \tilde{\zeta}\right\|^2 = -\frac{\tilde{\sigma}^2}{M-1}.
\end{aligned}$$

Therefore, these identities help us to establish the formula for the sample variance

$$
\begin{aligned}
\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi}^{k}-\tilde{\zeta}\right\|^{2}\right] &= \frac{1}{k^{2}}\sum_{i=1}^{k}\sum_{j=1}^{k}\operatorname{cov}\left[\zeta_{\pi_i},\zeta_{\pi_j}\right] \\
&= \frac{1}{k^{2}}\left(\mathbf{E}\left[\sum_{i=1}^{k}\left\|\zeta_{\pi_i}-\tilde{\zeta}\right\|^{2}\right]+\sum_{r=1}^{k}\sum_{s=1,\left\lfloor\frac{r}{N}\right\rfloor=\left\lfloor\frac{s}{N}\right\rfloor\ \&\ j_r\neq j_s}^{k}\operatorname{cov}[\zeta_{\pi_r},\zeta_{\pi_s}]+\sum_{r=1}^{k}\sum_{s=1,\left\lfloor\frac{r}{N}\right\rfloor\neq\left\lfloor\frac{s}{N}\right\rfloor}^{k}\operatorname{cov}[\zeta_{\pi_r},\zeta_{\pi_s}]\right) \\
&= \frac{1}{k^{2}}\left(k\sigma^{2}+\frac{m_kN^{2}+j_k^{2}-k}{N-1}\left(N\tilde{\sigma}^{2}-\sigma^{2}\right)-\frac{k^{2}-m_kN^{2}-j_k^{2}}{M-1}\tilde{\sigma}^{2}\right) \\
&= \frac{j_k(N-j_k)}{k^{2}(N-1)}\sigma^{2}+\left(\frac{(m_kN^{2}+j_k^{2})(MN-1)}{k^{2}(N-1)(M-1)}-\frac{N}{k(N-1)}-\frac{1}{M-1}\right)\tilde{\sigma}^{2}.
\end{aligned}
\tag{12}
$$

For the second part, we have

$$
\mathbf{E}\left[\tilde{\zeta}_{\pi,o}^{k}\right]=\mathbf{E}\left[\frac{1}{k}\left(\sum_{i=1}^{k_N}\frac{1}{C}\sum_{p=1}^{C}\zeta_{\pi_i}^{p}+\sum_{i=k_N+1}^{k}\zeta_{\pi_i}^{o}\right)\right]=\frac{1}{k}\left(\sum_{i=1}^{k_N}\frac{1}{C}\sum_{p=1}^{C}\tilde{\zeta}+\sum_{i=k_N+1}^{k}\tilde{\zeta}\right)=\tilde{\zeta}.
$$

Then, for the variance

$$
\begin{aligned}
\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi,o}^{k}-\tilde{\zeta}\right\|^{2}\right] &= E\left\|\frac{1}{k}\left(\sum_{i=1}^{k_N}\frac{1}{C}\sum_{p=1}^{C}\left(\zeta_{\pi_i}^{p}-\tilde{\zeta}\right)+\sum_{i=k_N+1}^{k}\left(\zeta_{\pi_i}^{o}-\tilde{\zeta}\right)\right)\right\|^{2} \\
&= \left(\frac{k_N}{k}\right)^{2}E\left\|\frac{1}{k_NC}\sum_{i=1}^{k_N}\sum_{p=1}^{C}\left(\zeta_{\pi_i}^{p}-\tilde{\zeta}\right)\right\|^{2}+\left(\frac{k-k_N}{k}\right)^{2}\mathbf{E}\left[\left\|\frac{1}{k-k_N}\sum_{i=k_N+1}^{k}\left(\zeta_{\pi_i}^{o}-\tilde{\zeta}\right)\right\|^{2}\right] \\
&\quad+\frac{2}{k^{2}C^{2}}\mathbf{E}\left[\left\langle\sum_{i=1}^{k_N}\sum_{p=1}^{C}\left(\zeta_{\pi_i}^{p}-\tilde{\zeta}\right),\sum_{i=k_N+1}^{k}\left(\zeta_{\pi_i}^{o}-\tilde{\zeta}\right)\right\rangle\right] \\
&= \left(\frac{k_N}{k}\right)^{2}\bar{\sigma}^{2}(k_NC)+\left(\frac{k-k_N}{k}\right)^{2}\bar{\sigma}^{2}(k-k_N)-\frac{2k_N(k-k_N)}{k^{2}C(M-1)}\tilde{\sigma}^{2}
\end{aligned}
$$

□

Let us now analyse the obtained results. Firstly, one can notice that in the case $C=1$, (11) is equivalent to (9) since $k_N=m_kN$ and $k-k_N=j_k$. Next, we link the obtained result with the existing works. In the special case $M=1$, we have $\tilde{\sigma}^2=0$. Therefore, $\bar{\sigma}^2(k)=\frac{k(N-k)}{k^2(N-1)}\sigma^2$ that recovers the variance bound of (Mishchenko et al., 2020, Lemma 1) for simple random reshuffling. In the full participation case, i.e., $M=C$, we have $k_N=m_k=0$ and $j_k=k$. Therefore,

$$
\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi}^{k}-\tilde{\zeta}\right\|^{2}\right]=\bar{\sigma}^{2}(k)=\frac{k(N-k)}{k^{2}(N-1)}\sigma^{2}+\frac{N}{N-1}\left(1-\frac{1}{k}\right)\tilde{\sigma}^{2}\leq\frac{N}{2k^{2}}\sigma^{2}+2\tilde{\sigma}^{2}.
$$

The expression above can be used to recover the variance bound for full participation algorithm FedRR (Mishchenko et al., 2021, Theorem 1). The next step of the analysis is to give an upper bound on the quantity $k^2\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi}^{k}-\tilde{\zeta}\right\|^{2}\right]$, which is the key quantity that we use to bound the variance due to double reshuffling sampling procedure.

Lemma E.2. *Let the settings of Lemma E.1, then*

$$
k^{2}\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi}^{k}-\tilde{\zeta}\right\|^{2}\right]\leq\left(\frac{M}{2C^{2}}+2\right)N^{2}\tilde{\sigma}^{2}+\frac{N}{2}\sigma^{2}
\tag{13}
$$

Proof. First, we recall the definition of $\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi}^{k}-\tilde{\zeta}\right\|^{2}\right]$. We have

$$\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi,o}^{k}-\tilde{\zeta}\right\|^{2}\right]=\left(\frac{k_N}{k}\right)^{2}\bar{\sigma}^{2}(k_N C)+\left(\frac{k-k_N}{k}\right)^{2}\bar{\sigma}^{2}(k-k_N)-\frac{2k_N(k-k_N)}{C(M-1)}\tilde{\sigma}^{2}.$$

Since $k_N = m_k N$ and $k - k_N = j_k$, then

$$\begin{aligned}
k^2\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi,o}^{k}-\tilde{\zeta}\right\|^{2}\right] &= k_N^2\left(\frac{k_N CN(MN-1)}{k_N^2C^2(N-1)(M-1)}-\frac{N}{k_N C(N-1)}-\frac{1}{M-1}\right)\tilde{\sigma}^2 \\
&\quad + j_k^2\left(\frac{j_k(N-j_k)}{j_k^2(N-1)}\sigma^2+\left(\frac{j_k^2(MN-1)}{j_k^2(N-1)(M-1)}-\frac{N}{j_k(N-1)}-\frac{1}{M-1}\right)\tilde{\sigma}^2\right)-\frac{2k_N j_k}{k^2C(M-1)}\tilde{\sigma}^2 \\
&= \left(\frac{k_N N(MN-1)}{C(N-1)(M-1)}-\frac{k_N N}{C(N-1)}-\frac{k_N^2}{M-1}\right)\tilde{\sigma}^2 \\
&\quad + \frac{j_k(N-j_k)}{(N-1)}\sigma^2+\left(\frac{j_k^2(MN-1)}{(N-1)(M-1)}-\frac{j_k N}{(N-1)}-\frac{j_k^2}{M-1}\right)\tilde{\sigma}^2-\frac{2k_N j_k}{C(M-1)}\tilde{\sigma}^2 \\
&= \left(\frac{k_N MN}{C(M-1)}-\frac{k_N^2}{M-1}\right)\tilde{\sigma}^2+\frac{j_k(N-j_k)}{(N-1)}\sigma^2+\left(\frac{j_k^2 M}{M-1}-\frac{j_k N}{(N-1)}\right)\tilde{\sigma}^2-\frac{2k_N j_k}{C(M-1)}\tilde{\sigma}^2.
\end{aligned}$$

To obtain the upper bound, we first use $\frac{j_k(N-j_k)}{(N-1)} \leq \frac{N}{2}$. Therefore,

$$k^2\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi,o}^{k}-\tilde{\zeta}\right\|^{2}\right] \leq \left(\frac{k_N MN}{C(M-1)}-\frac{k_N^2}{M-1}\right)\tilde{\sigma}^2+\frac{N}{2}\sigma^2+\left(\frac{j_k^2 M}{M-1}-\frac{j_k N}{(N-1)}\right)\tilde{\sigma}^2-\frac{2k_N j_k}{C(M-1)}\tilde{\sigma}^2.$$

The first part of the first term is a quadratic function with respect to k_N, so we can estimate its maximum by equating its derivative to zero. For the term $\frac{j_k^2 M}{M-1}$, we have $\frac{j_k^2 M}{M-1} \leq 2N^2$ for $M \geq 2$. We ignore other negative terms. This yields the following upper bound

$$k^2\mathbf{E}\left[\left\|\tilde{\zeta}_{\pi,o}^{k}-\tilde{\zeta}\right\|^{2}\right] \leq \left(\frac{MN^2}{2C^2}+2N^2\right)\tilde{\sigma}^2+\frac{N}{2}\sigma^2,$$

which concludes the proof. □

In the next step, we will use this result to upper bound the following sequence

$$\begin{aligned}
x_\star^0 &= x_{m,\star}^{0,0} = x_\star,\ \forall m \in S_t^{\lambda_0} \\
x_{m,\star}^{r,j+1} &= x_{m,\star}^{r,j} - \gamma\nabla f_m^{\pi_m^j}\left(x_\star\right),\ \forall m \in [M], \\
x_\star^{r+1} &= \frac{1}{C}\sum_{m\in S_t^{\lambda_r}} x_{m,\star}^{r,N}.
\end{aligned} \tag{14}$$

Concretely, we are interested in upper bounding the distance of this sequence from the optimal solution $x_\star$ as this quantity will be useful to provide the upper bound for the statistical term in our convergence analysis. Note that $x_\star^{r+1} = x_\star$. Our result is summarized in the next lemma.

Lemma E.3. *The distance from the optimum of the iterates defined by* (7) *is bounded, i.e.,*

$$\mathbf{E}\left[\left\|x_{m,\star}^{r,j+1}-x_\star\right\|^2\right] \leq \gamma^2\left(\frac{MN^2}{2C^2}+2N^2\right)\tilde{\sigma}^2+\frac{\gamma^2 N}{2}\sigma^2, \tag{15}$$

where $\tilde{\sigma}_\star^2 \overset{def}{=} \frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2$ *and* $\sigma_\star^2 \overset{def}{=} \frac{1}{MN}\sum_{m=1}^{M}\sum_{j=1}^{N}\left\|\nabla f_m^j(x_\star)\right\|^2$.

Proof. Using (7), we get

$$\mathbf{E}\left[\left\|x_{m,\star}^{r,j+1}-x_\star\right\|^2\right] = \gamma^2\mathbf{E}\left[\left\|\frac{1}{C}\sum_{q=1}^{r-1}\sum_{z\in S_t^{\lambda_r}}\sum_{i=1}^{N}\nabla f_z^{\pi_z^i}\left(x_\star\right)+\sum_{l=0}^{j}\nabla f_m^{\pi_m^j}\left(x_\star\right)\right\|^2\right]$$

This is exactly the setup of the double shuffling procedure discussed in Lemma E.1. Therefore, we can apply Lemma E.2 that yields

$$\mathbf{E}\left[\left\|x_{m,\star}^{r,j+1}-x_\star\right\|^2\right] \le \gamma^2\left(\frac{MN^2}{2C^2}+2N^2\right)\tilde{\sigma}^2+\frac{\gamma^2 N}{2}\sigma^2,$$

where the corresponding value for $\tilde{\sigma}^2 = \frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2$ and for $\sigma^2 = \frac{1}{MN}\sum_{m=1}^{M}\sum_{j=1}^{N}\left\|\nabla f_m^j(x_\star)\right\|^2$, which concludes the proof. □

Using the above results, we can upper bound both quantities, based on the Bregman divergence, that appear in our analysis, i.e., $\sigma_{m,\text{DS}}^2$ and $\sigma_{m,\text{CS}}^2$. The lemma follows.

Lemma E.4. *The variance introduced by Algorithm 1 is bounded, i.e.,*

$$\begin{aligned}\max_{m\in[M]}\sigma_{m,DS}^2 &\le L\left(\frac{MN^2}{2C^2}+2N^2\right)\tilde{\sigma}_\star^2+\frac{LN}{2}\sigma_\star^2,\\ \max_{m\in[M]}\sigma_{m,CS}^2 &\le \frac{LM}{2C^2}\tilde{\sigma}_\star^2.\end{aligned} \tag{16}$$

Proof. First, we recall the definition of $\sigma_{m,\text{DS}}^2$ and apply the smoothness assumption.

$$\sigma_{m,\text{DS}}^2 = \frac{1}{\gamma^2}\mathrm{E}\left[D_{f_m^{\pi_j}}\left(x_{m,\star}^{r,j},x_\star\right)\right] \overset{(3)}{\le} \frac{L}{\gamma^2}\mathrm{E}\left[\left\|x_{m,\star}^{r,j}-x_\star\right\|^2\right].$$

Then, we apply (15) that gives desired result

$$\sigma_{m,\text{DS}}^2 \le L\left(\frac{MN^2}{2C^2}+2N^2\right)\tilde{\sigma}_\star^2+\frac{LN}{2}\sigma_\star^2.$$

For the second term, we again use its definition and the smoothness assumption. Therefore,

$$\sigma_{m,\text{CS}}^2 = \frac{1}{\eta^2}\mathrm{E}\left[D_{f_m}\left(x_{m,\star}^{r},x_\star\right)\right] \overset{(3)}{\le} \frac{L}{\eta^2}\mathrm{E}\left[\left\|x_{m,\star}^{r}-x_\star\right\|^2\right].$$

This setup is also reflected in Lemma E.1, but with $N=1$. Therefore, $\sigma^2=\tilde{\sigma^2}$. Furthermore, we can't apply (9) directly as it is not defined for $N=1$. We can instead derive the variance bound for the case $N=1$ using the proof techniques provided in the proof of Lemma E.1, where we ignore the middle term in (12) as the number of summands satisfying condition $s=1$, $\left\lfloor\frac{r}{N}\right\rfloor \neq \left\lfloor\frac{s}{N}\right\rfloor$ is zero for $N=1$. This, combined with the fact that $m_k=k$ and $j_k=0$, yields The above result together with (15) yields $\bar{\sigma}^2(k)=\frac{\tilde{\sigma}^2(M-k)}{k(M-1)}$ for $k\in[\frac{M}{C}]$. We can plug this equality to (11), where $k_1=k$. Therefore,

$$\begin{aligned}\sigma_{m,\text{DS}}^2 &\le L\frac{L}{\gamma^2}\mathrm{E}\left[\left\|x_{m,\star}^{r}-x_\star\right\|^2\right] \le Lk^2\bar{\sigma}^2(kC)\\ &= \frac{L\tilde{\sigma}_\star^2 k(M-kC)}{C(M-1)} = \frac{L\tilde{\sigma}_\star^2}{C^2}\frac{kC(M-kC)}{kC(M-1)} \le \frac{L\tilde{\sigma}_\star^2 M}{2C^2},\end{aligned}$$

which concludes the proof. □

Equipped with the bounds for the variance terms, we are ready to proceed with the exact convergence bounds.

F. Proof in case of f_m^j's strong convexity

In this section, we need to use the following equalities to prove convergence guarantees:

$$x_t^{r+1} = \frac{1}{C} \sum_{m \in S_t^{\lambda_r}} x_{m,t}^{r,n},$$

$$x_{m,t}^{r,j+1} = x_{m,t}^{r,j} - \gamma \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), \qquad x_{m,t}^{r,j} = x_t^r - \gamma \sum_{k=1}^{j-1} \nabla f_m^{\pi_k}\left(x_{m,t}^{r,k}\right),$$

$$x_t^{r+1} = x_t^r - \gamma \frac{1}{C} \sum_{m \in S_t^{\lambda_r}} \sum_{j=1}^{N-1} \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right).$$

These equations are necessary for one-step, local, and meta-epoch analysis. Let us start from the case when all individual functions are μ-strongly convex.

Theorem F.1. *Suppose that the functions $f_1, \ldots, f_n$ are μ-strongly convex and L-smooth. Then for Algorithm 1 with a constant stepsize $\gamma \leq \frac{1}{L}$, the iterates generated by either of the algorithm satisfy*

$$\mathrm{E}\left[\|x_T - x_\star\|^2\right] \leq (1-\gamma\mu)^{NRT} \|x_0 - x_\star\|^2 + \frac{2\gamma^2}{\mu} \max_{r,m} \sigma_{m,DS}^2. \tag{17}$$

Proof. We start our proof by analyzing the distance between intermediate point $x_{m,t}^{r,j+1}$ and a point of auxiliary sequence:

$$\begin{aligned}
\mathrm{E}\left[\left\|x_{m,t}^{r,j+1} - x_{m,\star}^{r,j+1}\right\|^2\right] &= \mathrm{E}\left[\left\|x_{m,t}^{r,j} - \gamma \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) - \left(x_{m,\star}^{r,j} - \gamma \nabla f_m^{\pi_j}\left(x_\star\right)\right)\right\|^2\right] \\
&= \mathrm{E}\left[\left\|x_{m,t}^{r,j} - x_{m,\star}^{r,j} - \gamma\left(\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) - \gamma \nabla f_m^{\pi_j}\left(x_\star\right)\right)\right\|^2\right] \\
&= \mathrm{E}\left[\left\|x_{m,t}^{r,j} - x_{m,\star}^{r,j}\right\|^2 - 2\gamma\left\langle \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) - \nabla f_m^{\pi_j}\left(x_\star\right), x_{m,t}^{r,j} - x_{m,\star}^{r,j}\right\rangle\right] \\
&\quad + \gamma^2 \mathrm{E}\left[\left\|\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) - \nabla f_m^{\pi_j}\left(x_\star\right)\right\|^2\right].
\end{aligned} \tag{18}$$

Using a three-point identity, we have

$$\left\langle \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) - \nabla f_m^{\pi_j}\left(x_\star\right), x_{m,t}^{r,j} - x_{m,\star}^{r,j}\right\rangle = D_{f_m^{\pi_j}}\left(x_{m,\star}^{r,j}, x_{m,t}^{r,j}\right) + D_{f_m^{\pi_j}}\left(x_{m,t}^{r,j}, x_\star\right) - D_{f_m^{\pi_j}}\left(x_{m,\star}^{r,j}, x_\star\right). \tag{19}$$

Plugging (19) into (18) we obtain

$$\begin{aligned}
\mathrm{E}\left[\left\|x_{m,t}^{r,j+1} - x_{m,\star}^{r,j+1}\right\|^2\right] &= \mathrm{E}\left[\left\|x_{m,t}^{r,j} - x_{m,\star}^{r,j}\right\|^2\right] - 2\gamma \mathrm{E}\left[D_{f_m^{\pi_j}}\left(x_{m,\star}^{r,j}, x_{m,t}^{r,j}\right)\right] - 2\gamma \mathrm{E}\left[D_{f_m^{\pi_j}}\left(x_{m,t}^{r,j}, x_\star\right)\right] \\
&\quad + 2\gamma \mathrm{E}\left[D_{f_m^{\pi_j}}\left(x_{m,\star}^{r,j}, x_\star\right)\right] + \gamma^2 \mathrm{E}\left[\left\|\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) - \nabla f_m^{\pi_j}\left(x_\star\right)\right\|^2\right].
\end{aligned}$$

Using μ-convexity and L-smoothness, we have

$$\frac{\mu}{2}\left\|x_{m,\star}^{r,j} - x_{m,t}^{r,j}\right\|^2 \leq D_{f_m^{\pi_j}}\left(x_{m,\star}^{r,j}, x_{m,t}^{r,j}\right),$$

$$\mathrm{E}\left[\left\|\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) - \nabla f_m^{\pi_j}\left(x_\star\right)\right\|^2\right] \leq 2L D_{f_m^{\pi_j}}\left(x_{m,t}^{r,j}, x_\star\right).$$

Using this inequality, we have

$$\begin{aligned}\mathrm{E}\left[\left\|x_{m,t}^{r,j+1}-x_{m,\star}^{r,j+1}\right\|^2\right] &\le (1-\gamma\mu)\mathrm{E}\left[\left\|x_{m,t}^{r,j}-x_{m,\star}^{r,j}\right\|^2\right]-2\gamma\mathrm{E}\left[D_{f_m^{\pi_j}}\left(x_{m,t}^{r,j},x_\star\right)\right]\\ &\quad+2\gamma\mathrm{E}\left[D_{f_m^{\pi_j}}\left(x_{m,\star}^{r,j},x_\star\right)\right]+2L\gamma^2 D_{f_m^{\pi_j}}\left(x_{m,t}^{r,j},x_\star\right)\\ &\le (1-\gamma\mu)\mathrm{E}\left[\left\|x_{m,t}^{r,j}-x_{m,\star}^{r,j}\right\|^2\right]+2\gamma\mathrm{E}\left[D_{f_m^{\pi_j}}\left(x_{m,\star}^{r,j},x_\star\right)\right]\\ &\quad-2\gamma\left(1-\gamma L\right)\mathrm{E}\left[D_{f_m^{\pi_j}}\left(x_{m,t}^{r,j},x_\star\right)\right].\end{aligned}$$

Using $\gamma \le \frac{1}{L}$ and definition of $\sigma_{m,\mathrm{CS}}^2$, we get the following bound:

$$\mathrm{E}\left[\left\|x_{m,t}^{r,j+1}-x_{m,\star}^{r,j+1}\right\|^2\right] \le (1-\gamma\mu)\mathrm{E}\left[\left\|x_{m,t}^{r,j}-x_{m,\star}^{r,j}\right\|^2\right]+2\gamma^3\sigma_{m,\mathrm{CS}}^2.$$

Unrolling this recursion, we obtain

$$\begin{aligned}\mathrm{E}\left[\left\|x_{m,t}^{r,n}-x_{m,\star}^{r,n}\right\|^2\right] &\le (1-\gamma\mu)^N\mathrm{E}\left[\left\|x_{m,t}^{r,0}-x_{m,\star}^{r,0}\right\|^2\right]+2\gamma^3\sigma_{m,\mathrm{CS}}^2\sum_{j=0}^{N-1}(1-\gamma\mu)^j\\ &= (1-\gamma\mu)^N\mathrm{E}\left[\left\|x_t^r-x_\star^r\right\|^2\right]+2\gamma^3\sigma_{m,\mathrm{CS}}^2\sum_{j=0}^{N-1}(1-\gamma\mu)^j.\end{aligned}$$

Now we need to establish recursion for rounds of communication:

$$\begin{aligned}\mathrm{E}\left[\left\|x_t^{r+1}-x_\star^{r+1}\right\|^2\right] &= \mathrm{E}\left[\left\|\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}x_{m,t}^{r,n}-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}x_{m,\star}^{r,n}\right\|^2\right]\\ &\le \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\mathrm{E}\left[\left\|x_{m,t}^{r,n}-x_{m,\star}^{r,n}\right\|^2\right]\\ &\le \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\left((1-\gamma\mu)^N\mathrm{E}\left[\left\|x_t^r-x_\star^r\right\|^2\right]+2\gamma^3\sigma_{m,\mathrm{CS}}^2\sum_{j=0}^{N-1}(1-\gamma\mu)^j\right)\\ &= (1-\gamma\mu)^N\mathrm{E}\left[\left\|x_t^r-x_\star^r\right\|^2\right]+2\gamma^3\max_m\sigma_{m,\mathrm{CS}}^2\sum_{j=0}^{N-1}(1-\gamma\mu)^j.\end{aligned}$$

Using the fact that $x_{t+1}=x_t^R$ and $x_\star=x_\star^R$, we can unroll this recursion again for index r:

$$\begin{aligned}\mathrm{E}\left[\left\|x_{t+1}-x_\star\right\|^2\right] &= \mathrm{E}\left[\left\|x_t^R-x_\star^R\right\|^2\right]\\ &\le (1-\gamma\mu)^{NR}\mathrm{E}\left[\left\|x_t-x_\star\right\|^2\right]+2\gamma^3\max_{r,m}\sigma_{m,\mathrm{DS}}^2\sum_{j=0}^{N-1}(1-\gamma\mu)^j\sum_{r=0}^{R-1}(1-\gamma\mu)^{rN}.\end{aligned}$$

Unrolling this recursion again for index t and using tower property, we have

$$\begin{aligned}\mathrm{E}\left[\left\|x_T-x_\star\right\|^2\right] &\le (1-\gamma\mu)^{NRT}\left\|x_0-x_\star\right\|^2\\ &\quad+2\gamma^3\max_{r,m}\sigma_{m,\mathrm{DS}}^2\sum_{j=0}^{N-1}(1-\gamma\mu)^j\sum_{r=0}^{R-1}(1-\gamma\mu)^{rN}\sum_{t=0}^{T-1}(1-\gamma\mu)^{tRN}\\ &\le (1-\gamma\mu)^{NRT}\left\|x_0-x_\star\right\|^2+2\gamma^3\sum_{j=0}^{NRT-1}(1-\gamma\mu)^j\max_{r,m}\sigma_{m,\mathrm{DS}}^2\\ &\le (1-\gamma\mu)^{NRT}\left\|x_0-x_\star\right\|^2+\frac{2\gamma^2}{\mu}\max_{r,m}\sigma_{m,\mathrm{DS}}^2.\end{aligned}$$

□

G. Proof in case of f_i's strong convexity

This section provides convergence bounds for the case when only f_m is μ-strongly convex. In this regime, we cannot use the trick with the additional sequence as we do in the previous section. Due to the biased nature of gradient updates, we cannot take expectations directly, and we need to get an error bound of gradient approximation. Formally, we have a lemma for this below.

Lemma G.1. *Assume that each f_m^j is L-smooth, then we have*

$$\mathrm{E}\left[\left\|g_t^r - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}} \nabla f_m(x_t^r)\right\|^2\right] \leq L^2 V_t^r,$$

where V_t^r is defined as

$$V_t^r = \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\left\|x_{m,t}^{r,j} - x_t^r\right\|^2.$$

Proof. We start from definition of g_t^r and then we apply Jensen's inequality and L-smooth assumption:

$$\begin{aligned}
\mathrm{E}\left[\left\|g_t^r - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}} \nabla f_m(x_t^r)\right\|^2\right] &= \mathrm{E}\left[\left\|\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}(x_{m,t}^{r,j}) - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)\right\|^2\right] \\
&= \mathrm{E}\left[\left\|\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}(x_{m,t}^{r,j}) - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}(x_t^r)\right\|^2\right] \\
&= \mathrm{E}\left[\left\|\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\left(\nabla f_m^{\pi_j}(x_{m,t}^{r,j}) - \nabla f_m^{\pi_j}(x_t^r)\right)\right\|^2\right] \\
&\overset{\text{Y}}{\leq} \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\left\|\nabla f_m^{\pi_j}(x_{m,t}^{r,j}) - \nabla f_m^{\pi_j}(x_t^r)\right\|^2\right] \\
&\overset{\text{L}}{\leq} L^2\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\left\|x_{m,t}^{r,j} - x_t^r\right\|^2 \\
&= L^2 V_t^r.
\end{aligned}$$

□

We manage to get an error bound using the sum of distances between intermediate point $x_{m,t}^{r,j}$ and starting point x_t^r. Now we need to provide bounds for such sums V_t. The following lemma does it formally.

Lemma G.2. *Assume that each $f_{m,j}$ is L-smooth, then we have*

$$\mathrm{E}\left[V_t^r\right] \leq 8\gamma^2 n^2 L\frac{1}{M}\sum_{m=1}^{M} D_{f_m}\left(x_t^r, x_\star\right) + 2\gamma^2 n^2\frac{1}{M}\sum_{m=1}^{M}\left\|\nabla f_m(x_\star)\right\|^2 + 2\gamma^2 n\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\left\|\nabla f_m^{\pi_j}(x_\star)\right\|^2. \quad (20)$$

Proof. We start with the update rule:

$$x_{m,t}^{r,j} = x_t^r - \gamma\sum_{k=0}^{j-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,k}\right).$$

Using this form, we get

$$
\begin{aligned}
\mathbf{E}\left[\left\|x_{m,t}^{r,j}-x_t^r\right\|^2\right] &= \gamma^2\mathbf{E}\left[\left\|\sum_{k=0}^{j-1}\nabla f_m^{\pi_k}\left(x_{m,t}^{r,k}\right)\right\|^2\right] \\
&\le 2\gamma^2\mathbf{E}\left[\left\|\sum_{k=0}^{j-1}\left(\nabla f_m^{\pi_k}\left(x_{m,t}^{r,k}\right)-\nabla f_m^{\pi_k}\left(x_t^r\right)\right)\right\|^2\right]+2\gamma^2\mathbf{E}\left[\left\|\sum_{k=0}^{j-1}\nabla f_m^{\pi_k}\left(x_t^r\right)\right\|^2\right] \\
&\le 2\gamma^2 j\sum_{k=0}^{j-1}\mathbf{E}\left[\left\|\nabla f_m^{\pi_k}\left(x_{m,t}^{r,k}\right)-\nabla f_m^{\pi_k}\left(x_t^r\right)\right\|^2\right]+2\gamma^2\mathbf{E}\left[\left\|\sum_{k=0}^{j-1}\nabla\nabla f_m^{\pi_k}\left(x_t^r\right)\right\|^2\right] \\
&\le 2\gamma^2 jL^2\sum_{k=0}^{j-1}\mathbf{E}\left[\left\|x_{m,t}^{r,k}-x_t^r\right\|^2\right]+2\gamma^2\mathbf{E}\left[\left\|\sum_{k=0}^{j-1}\nabla f_m^{\pi_k}\left(x_t^r\right)\right\|^2\right] \\
&\le 2\gamma^2 jL^2\sum_{k=0}^{j-1}\mathbf{E}\left[\left\|x_{m,t}^{r,k}-x_t^r\right\|^2\right]+2\gamma^2\left(j^2\left\|\nabla f_m^{\lambda_r}(x_t^r)\right\|^2+\frac{j(N-j)}{N-1}\left(\sigma_{m,t}\right)^2\right).
\end{aligned}
$$

Now we can sum these inequalities:

$$
\begin{aligned}
\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\mathbf{E}\left[\left\|x_{m,t}^{r,j}-x_t^r\right\|^2\right] &\le 2\gamma^2L^2\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}j\sum_{k=0}^{j-1}\mathbf{E}\left[\left\|x_{m,t}^{r,k}-x_t^r\right\|^2\right] \\
&\quad+2\gamma^2\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\left(j^2\left\|\nabla f_m(x_t^r)\right\|^2+\frac{j(N-j)}{N-1}\left(\sigma_{m,t}\right)^2\right).
\end{aligned}
$$

□

Now we are ready to prove the theorem.

Theorem G.3. *Assume that each f_m is μ-strongly convex. Also, assume that each f_m^j is convex and L-smooth. Let $\eta \le \frac{1}{4L}$ and $\gamma \le \frac{1}{8NL\sqrt{\kappa}}$, then for iterates generated by Algorithm 1, we have*

$$
\mathrm{E}\left[\|x_T-x_\star\|^2\right] \le (1-\eta\mu)^{RT}\|x_0-x_\star\|^2+\frac{4}{\mu}\eta^2\max_{r,m}\sigma_{m,CS}^2+12\kappa^2\gamma^2N^2\sigma_\star^2+12\gamma^2\kappa^2N\sigma_\star^2,
$$

where $\sigma_{m,CS}^2=\frac{1}{\eta^2}\mathrm{E}\left[D_{f_m}\left(x_\star^r,x_\star\right)\right]$.

Proof. We start from the following equations:

$$
\begin{aligned}
x_t^{r+1} &= x_t^r-\eta\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right)=x_t^r-\eta g_t^r, \\
g_t^r &= \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}(x_{m,t}^{r,j}), \\
x_\star^r &= x_\star-\eta\sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m\left(x_{m,t}^{r,j}\right), \quad x_\star^{r+1}=x_\star^r-\eta\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star).
\end{aligned}
$$

We start from the distance to the solution:

$$
\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1}-x_\star^{r+1}\right\|^2\right] &= \mathrm{E}\left[\left\|x_t^r-\eta g_t^r-\left(x_\star^r-\eta\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_*)\right)\right\|^2\right] \\
&= \mathrm{E}\left[\|x_t^r-x_\star^r\|^2-2\eta\left\langle g_t^r-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star),x_t^r-x_\star^r\right\rangle+\eta^2\left\|g_t^r-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star)\right\|^2\right] \\
&= \mathrm{E}\left[\|x_t^r-x_\star^r\|^2+\eta^2\left\|g_t^r+\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star)\right\|^2\right] \\
&\quad -2\eta\mathrm{E}\left[\left\langle g_t^r+\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star),x_t^r-x_\star^r\right\rangle\right].
\end{aligned}
$$

Using Young's inequality we have

$$
\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1}-x_\star^{r+1}\right\|^2\right] &= \mathrm{E}\left[\|x_t^r-x_\star^r\|^2+\eta^2\left\|g_t^r+\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star)\right\|^2\right] \\
&\quad -2\eta\mathrm{E}\left[\left\langle g_t^r+\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star),x_t^r-x_\star^r\right\rangle\right].
\end{aligned}
$$

Applying Young's inequality again, we have

$$
\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1}-x_\star^{r+1}\right\|^2\right] &\le \mathrm{E}\left[\|x_t^r-x_\star^r\|^2+2\eta^2\left\|g_t^r-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)\right\|^2\right] \\
&\quad +2\eta^2\mathrm{E}\left[\left\|\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star)\right\|^2\right] \\
&\quad -2\eta\mathrm{E}\left[\left\langle g_t^r-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r),x_t^r-x_\star^r\right\rangle\right] \\
&\quad -2\eta\mathrm{E}\left[\left\langle \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star),x_t^r-x_\star^r\right\rangle\right].
\end{aligned}
$$

Using Young's inequality for inner product $-2\left\langle a,b\right\rangle \leq \frac{1}{t}\|a\|^2 + t\|b\|^2$ we get

$$
\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1} - x_\star^{r+1}\right\|^2\right] \leq \mathrm{E}&\left[\|x_t^r - x_\star^r\|^2 + 2\eta^2\left\|g_t^r - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)\right\|^2\right] \\
&+ 2\eta^2\mathrm{E}\left[\left\|\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r) - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star)\right\|^2\right] \\
&+ \eta\mathrm{E}\left[\frac{\mu}{2}\left\|x_t^r - x_\star^r\right\|^2 + \frac{2}{\mu}\left\|g_t^r - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star)\right\|^2\right] \\
&- 2\eta\mathrm{E}\left[\left\langle\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r) - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star), x_t^r - x_\star^r\right\rangle\right].
\end{aligned}
$$

Rearraging terms leads to

$$
\mathrm{E}\left[\left\|x_t^{r+1} - x_\star^{r+1}\right\|^2\right] \leq \mathrm{E}\left[\left(1+\frac{\mu\eta}{2}\right)\left\|x_t^r - x_\star^r\right\|^2 + \left(2\eta^2 + 2\frac{\eta}{\mu}\right)\left\|g_t^r - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)\right\|^2\right] \tag{21}
$$

$$
+ 2\eta^2\mathrm{E}\left[\left\|\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r) - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star)\right\|^2\right] \tag{22}
$$

$$
- 2\eta\mathrm{E}\left[\left\langle\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r) - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star), x_t^r - x_\star^r\right\rangle\right]. \tag{23}
$$

Let us consider the last term:

$$
\begin{aligned}
\left\langle\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r) - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star), x_t^r - x_\star^r\right\rangle &= \left\langle\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\left(\nabla f_m(x_t^r) - \nabla f_m(x_\star)\right), x_t^r - x_\star^r\right\rangle \\
&= \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\left\langle\nabla f_m(x_t^r) - \nabla f_m(x_\star), x_t^r - x_\star^r\right\rangle.
\end{aligned}
$$

Let us look at inner product and use three-point identity:

$$
\left\langle\nabla f_m(x_t^r) - \nabla f_m(x_\star), x_t^r - x_\star^r\right\rangle = D_{f_m}\left(x_\star^r, x_t^r\right) + D_{f_m}\left(x_t^r, x_\star\right) - D_{f_m}\left(x_\star^r, x_\star\right). \tag{24}
$$

By μ-strong convexity of f_m, the first term in (24) satisfies

$$
\frac{\mu}{2}\left\|x_t^r - x_\star^r\right\|^2 \overset{(2)}{\leq} D_{f_m}\left(x_\star^r, x_t^r\right). \tag{25}
$$

Combining (25),(27) and using in (24) we have

$$
\begin{aligned}
-2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\langle \nabla f_m(x_t^r)-\nabla f_m(x_\star), x_t^r - x_\star^r\rangle\right] &= -2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_\star^r, x_t^r\right) + D_{f_m}\left(x_t^r, x_\star\right)\right] \\
&\quad + 2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_\star^r, x_\star\right)\right] \\
&\leq -\eta\mu \mathrm{E}\left[\|x_t^r - x_\star^r\|^2\right] \\
&\quad - 2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_t^r, x_\star\right)\right] \\
&\quad + 2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_\star^r, x_\star\right)\right]. \qquad (26)
\end{aligned}
$$

Applying (26) in (21) we obtain

$$
\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1} - x_\star^{r+1}\right\|^2\right] \leq \mathrm{E}&\left[\left(1+\frac{\mu\eta}{2}\right)\|x_t^r - x_\star^r\|^2 + \left(2\eta^2 + 2\frac{\eta}{\mu}\right)\left\|g_t^r - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)\right\|^2\right] \\
&+ 2\eta^2 \mathrm{E}\left[\left\|\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r) - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_\star)\right\|^2\right] - \eta\mu \mathrm{E}\left[\|x_t^r - x_\star^r\|^2\right] \\
&- 2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_t^r, x_\star\right)\right] + 2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_\star^r, x_\star\right)\right].
\end{aligned}
$$

We can bound first term in second line using L-smoothness:

$$
\frac{1}{2L}\left\|\nabla f_m\left(x_t^r\right) - \nabla f_m\left(x_\star\right)\right\|^2 \overset{(3)}{\leq} D_{f_m}\left(x_t^r, x_\star\right). \qquad (27)
$$

Applying (27) we get

$$
\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1} - x_\star^{r+1}\right\|^2\right] \leq \mathrm{E}&\left[\left(1+\frac{\mu\eta}{2}\right)\|x_t^r - x_\star^r\|^2 + \left(2\eta^2 + 2\frac{\eta}{\mu}\right)\left\|g_t^r - \frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)\right\|^2\right] \\
&+ 4L\eta^2 \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_t^r, x_\star\right)\right] - \eta\mu \mathrm{E}\left[\|x_t^r - x_\star^r\|^2\right] \\
&- 2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_t^r, x_\star\right)\right] + 2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_\star^r, x_\star\right)\right].
\end{aligned}
$$

Rearranging the terms we have

$$\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1}-x_\star^{r+1}\right\|^2\right] &\leq \mathrm{E}\left[\left(1-\frac{\mu\eta}{2}\right)\|x_t^r-x_\star^r\|^2+\left(2\eta^2+2\frac{\eta}{\mu}\right)\left\|g_t^r-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)\right\|^2\right] \\
&\quad+2\eta\left(2\eta L-1\right)\mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}D_{f_m}\left(x_t^r,x_\star\right)\right]+2\eta\mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}D_{f_m}\left(x_\star^r,x_\star\right)\right] \\
&=\mathrm{E}\left[\left(1-\frac{\mu\eta}{2}\right)\|x_t^r-x_\star^r\|^2+2\eta\left(\eta+\frac{1}{\mu}\right)\left\|g_t^r-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)\right\|^2\right] \\
&\quad+2\eta\left(2\eta L-1\right)\mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}D_{f_m}\left(x_t^r,x_\star\right)\right]+2\eta\mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}D_{f_m}\left(x_\star^r,x_\star\right)\right].
\end{aligned}$$

Since $L\geq\mu$ and $\eta\leq\frac{1}{2L}$ we have $\eta\leq\frac{1}{2\mu}$. Using this we get

$$\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1}-x_\star^{r+1}\right\|^2\right] &\leq \mathrm{E}\left[\left(1-\frac{\mu\eta}{2}\right)\|x_t^r-x_\star^r\|^2+\frac{3\eta}{\mu}\left\|g_t^r-\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\nabla f_m(x_t^r)\right\|^2\right] \\
&\quad+2\eta\left(2\eta L-1\right)\mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}D_{f_m}\left(x_t^r,x_\star\right)\right]+2\eta\mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}D_{f_m}\left(x_\star^r,x_\star\right)\right].
\end{aligned}$$

Applying lemma we have

$$\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1}-x_\star^{r+1}\right\|^2\right] &\leq \mathrm{E}\left[\left(1-\frac{\mu\eta}{2}\right)\|x_t^r-x_\star^r\|^2\right] \\
&\quad+\frac{3\eta}{\mu}L^2\left(8\gamma^2n^2L\frac{1}{M}\sum_{m=1}^{M}D_{f_m}\left(x_t^r,x_\star\right)+2\gamma^2n^2\frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2\right) \\
&\quad+\frac{3\eta}{\mu}L^2\left(2\gamma^2n\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\|\nabla f_m^{\pi_j}(x_\star)\|^2\right) \\
&\quad+2\eta\left(2\eta L-1\right)\frac{1}{M}\sum_{m=1}^{M}D_{f_m}\left(x_t^r,x_\star\right)+2\eta\mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}D_{f_m}\left(x_\star^r,x_\star\right)\right].
\end{aligned}$$

Rearraging terms leads to

$$\begin{aligned}
\mathrm{E}\left[\left\|x_t^{r+1}-x_\star^{r+1}\right\|^2\right] &\leq \mathrm{E}\left[\left(1-\frac{\mu\eta}{2}\right)\|x_t^r-x_\star^r\|^2\right] \\
&\quad+\frac{3\eta}{\mu}L^2\left(2\gamma^2n^2\frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2\right)+\frac{3\eta}{\mu}L^2\left(2\gamma^2n\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\|\nabla f_m^{\pi_j}(x_\star)\|^2\right) \\
&\quad+2\eta\left(2\eta L-1-12\gamma^2N^2L^2\kappa\right)\frac{1}{M}\sum_{m=1}^{M}D_{f_m}\left(x_t^r,x_\star\right)+2\eta\mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}D_{f_m}\left(x_\star^r,x_\star\right)\right].
\end{aligned}$$

Using $\eta \leq \frac{1}{4L}$ and $\gamma \leq \frac{1}{5NL\sqrt{\kappa}}$ we have $2\eta L - 1 - 12\gamma^2 N^2 L^2 \kappa \geq 0$. Finally, we get

$$\mathrm{E}\left[\left\|x_t^{r+1} - x_\star^{r+1}\right\|^2\right] \leq \mathrm{E}\left[\left(1 - \frac{\mu\eta}{2}\right)\|x_t^r - x_\star^r\|^2\right] + 2\eta \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_\star^r, x_\star\right)\right]$$
$$+ \frac{3\eta}{\mu}L^2\left(2\gamma^2 n^2 \frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2\right) + \frac{3\eta}{\mu}L^2\left(2\gamma^2 n \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\|\nabla f_m^{\pi_j}(x_\star)\|^2\right).$$

Using $x_{t+1} = x_t^R$, $x_\star^R = x_\star$, $x_t^0 = x_t$ and $x_\star^0 = x_\star$ we can unroll the recursion:

$$\begin{aligned}
\mathrm{E}\left[\|x_{t+1} - x_\star\|^2\right] &\leq (1-\eta\mu)^R\|x_t - x_\star\|^2 \\
&+ \sum_{r=0}^{R-1}(1-\eta\mu)^r\left(\frac{3\eta}{\mu}L^2\left(2\gamma^2 n^2\frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2\right)\right) \\
&+ \sum_{r=0}^{R-1}(1-\eta\mu)^r\left(\frac{3\eta}{\mu}L^2\left(2\gamma^2 n\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\|\nabla f_m^{\pi_j}(x_\star)\|^2\right)\right) \\
&+ \sum_{r=0}^{R-1}(1-\eta\mu)^r\left(2\eta\max_r \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_\star^r, x_\star\right)\right]\right).
\end{aligned}$$

Unrolling this recursion across T epochs, we obtain

$$\begin{aligned}
\mathrm{E}\left[\|x_T - x_\star\|^2\right] &\leq \left(1 - \frac{1}{2}\eta\mu\right)^{RT}\|x_0 - x_\star\|^2 \\
&+ \sum_{t=0}^{T-1}\left(1-\frac{1}{2}\eta\mu\right)^{Rt}\sum_{r=0}^{R-1}\left(1-\frac{1}{2}\eta\mu\right)^r\left(\frac{3\eta}{\mu}L^2\left(2\gamma^2 n^2\frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2\right)\right) \\
&+ \sum_{t=0}^{T-1}\left(1-\frac{1}{2}\eta\mu\right)^{Rt}\sum_{r=0}^{R-1}\left(1-\frac{1}{2}\eta\mu\right)^r\left(\frac{3\eta}{\mu}L^2\left(2\gamma^2 n\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\|\nabla f_m^{\pi_j}(x_\star)\|^2\right)\right) \\
&+ \sum_{t=0}^{T-1}\left(1-\frac{1}{2}\eta\mu\right)^{Rt}\sum_{r=0}^{R-1}\left(1-\frac{1}{2}\eta\mu\right)^r\left(2\eta\max_r \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_\star^r, x_\star\right)\right]\right).
\end{aligned}$$

Note that

$$\sum_{t=0}^{T-1}\left(1-\frac{1}{2}\eta\mu\right)^{Rt}\sum_{r=0}^{R-1}\left(1-\frac{1}{2}\eta\mu\right)^r = \sum_{t=0}^{T-1}\sum_{r=0}^{R-1}\left(1-\frac{1}{2}\eta\mu\right)^{tR+r} \leq \frac{2}{\eta\mu}.$$

Applying this inequality leads to

$$\begin{aligned}
\mathrm{E}\left[\|x_T - x_\star\|^2\right] &\leq (1-\eta\mu)^{RT}\|x_0 - x_\star\|^2 \\
&+ 12\kappa^2\gamma^2 N^2\frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2 \\
&+ 12\kappa^2\gamma^2 N\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\|\nabla f_m^{\pi_j}(x_\star)\|^2 \\
&+ \frac{4}{\mu}\max_r \mathrm{E}\left[\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} D_{f_m}\left(x_\star^r, x_\star\right)\right].
\end{aligned}$$

Using definition we obtain

$$\mathrm{E}\left[\|x_T - x_\star\|^2\right] \leq (1-\eta\mu)^{RT}\|x_0 - x_\star\|^2 + 12\kappa^2\gamma^2N^2\sigma_\star^2 + 12\kappa^2\gamma^2N\frac{1}{M}\sum_{m=1}^{M}\sigma_{m,\star}^2 + \frac{4}{\mu}\eta^2\max_{r,m}\sigma_{\mathrm{m,CS}}^2.$$

□

H. Proof in case of f's strong convexity

In this section we prove the bound for the most general case. We need to bound the second moment of the gradient approximation.

Lemma H.1. *Assume that each f_m^j is L-smooth function, then we have the following bound:*

$$\left\|\frac{1}{R}\sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right)\right\|^2 \leq 2L^2V_t + 4L(f(x_t) - f(x_\star)).$$

Proof. We use Young's inequality and L-smoothness to obtain the following bound:

$$\begin{aligned}
\left\|\frac{1}{R}\sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right)\right\|^2 &\leq 2\left\|\frac{1}{R}\sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) - \nabla f(x_\star)\right\|^2 + 2\|\nabla f(x_t)\|^2 \\
&\leq \frac{2L^2}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}\left\|x_t - x_{m,t}^{r,j}\right\|^2 + 4L\left(f(x_t) - f(x_\star)\right) \\
&\leq 2L^2V_t + 4L(f(x_t) - f(x_\star)).
\end{aligned}$$

□

As previously, we need to bound the sum of distances V_t:

Lemma H.2. *Assume that each f_m^j is L-smooth function, then we have the following bound:*

$$\mathbb{E}\left[V_t\right] \leq 8\eta^2L(2+R^2)D_f(x_t, x_\star) + 4\eta^2\frac{R}{N^2}\frac{M-C}{(M-1)C}\sigma_\star^2 + 4\gamma^2N^2\frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2 + 4\gamma^2N\frac{1}{M}\sum_{m=1}^{M}\sigma_{m,\star}^2.$$

Proof. We start from definition of $\left\|x_t - x_{m,t}^{r,j}\right\|^2$:

$$\begin{aligned}
\left\|x_t - x_{m,t}^{r,j}\right\|^2 &= \left\|x_{m,t}^{r,j} - x_t\right\|^2 \\
&= \left\|-\eta\sum_{k=0}^{r-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_k}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{k,j}\right) - \gamma\sum_{l=0}^{j-1}\nabla f_m^{\pi_l}\left(x_{m,t}^{r,l}\right)\right\|^2 \\
&= \left\|\eta\sum_{k=0}^{r-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_k}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{k,j}\right) + \gamma\sum_{l=0}^{j-1}\nabla f_m^{\pi_l}\left(x_{m,t}^{r,l}\right)\right\|^2 \\
&\leq 2\left\|\eta\sum_{k=0}^{r-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_k}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{k,j}\right)\right\|^2 + 2\left\|\gamma\sum_{l=0}^{j-1}\nabla f_m^{\pi_l}\left(x_{m,t}^{r,l}\right)\right\|^2.
\end{aligned}$$

Using Young's inequality, we obtain

$$\begin{aligned}\left\|x_t - x_{m,t}^{r,j}\right\|^2 \leq 4 &\left\|\eta \sum_{k=0}^{r-1} \frac{1}{C} \sum_{m \in S_t^{\lambda_k}} \frac{1}{N} \sum_{j=0}^{N-1} \left(\nabla f_m^{\pi_j}\left(x_{m,t}^{k,j}\right) - \nabla f_m^{\pi_j}\left(x_t\right)\right)\right\|^2 \\ &+ 4 \left\|\eta \sum_{k=0}^{r-1} \frac{1}{C} \sum_{m \in S_t^{\lambda_k}} \frac{1}{N} \sum_{j=0}^{N-1} \nabla f_m^{\pi_j}\left(x_t\right)\right\|^2 \\ &+ 4 \left\|\gamma \sum_{l=0}^{j-1} \left(\nabla f_m^{\pi_l}\left(x_{m,t}^{r,l}\right) - f_m^{\pi_l}\left(x_t\right)\right)\right\|^2 \\ &+ 4 \left\|\gamma \sum_{l=0}^{j-1} \nabla f_m^{\pi_l}\left(x_t\right)\right\|^2 .\end{aligned}$$

Using L-smoothness and lemma we have

$$\begin{aligned}\left\|x_t - x_{m,t}^{r,j}\right\|^2 \leq 4\eta^2 r^2 L^2 \frac{1}{rCN} \sum_{k=0}^{r-1} \sum_{m \in S_t^{\lambda_r}} \sum_{j=0}^{N-1} \left\|x_t - x_{m,t}^{k,j}\right\|^2 + 4\gamma^2 L^2 j \sum_{l=0}^{j-1} \left\|x_{m,t}^{r,l} - x_t\right\|^2 \\ + 4\gamma^2 \left(j^2 \|\nabla f_m(x_t)\|^2 + \frac{j(N-j)}{N-1}\sigma_{m,t}^2\right) + 4\eta^2 \frac{1}{N^2C^2}\left(N^2C^2r^2\|\nabla f(x_t)\|^2 + \frac{Cr(M-Cr)}{M-1}\sigma_t^2\right).\end{aligned}$$

Using this bound we obtain

$$\begin{aligned}\mathbb{E}\left[V_t\right] &= \frac{1}{CRN} \sum_{r=0}^{R-1} \sum_{m \in S_t^{\lambda_r}} \sum_{j=0}^{N-1} \mathbb{E}\left\|x_{m,t}^{k,j} - x_t\right\|^2 \\ &\leq \frac{1}{CRN} \sum_{r=0}^{R-1} \sum_{m \in S_t^{\lambda_r}} \sum_{j=0}^{N-1} \left(4\eta^2 r^2 L^2 \frac{1}{rCN} \sum_{k=0}^{r-1} \sum_{m \in S_t^{\lambda_r}} \sum_{j=0}^{N-1} \left\|x_t - x_{m,t}^{k,j}\right\|^2 + 4\gamma^2 L^2 j \sum_{l=0}^{j-1} \left\|x_{m,t}^{r,l} - x_t\right\|^2\right) \\ &+ \frac{1}{CRN} \sum_{r=0}^{R-1} \sum_{m \in S_t^{\lambda_r}} \sum_{j=0}^{N-1} \left(4\gamma^2 \left(j^2 \|\nabla f_m(x_t)\|^2 + \frac{j(N-j)}{N-1}\sigma_{m,t}^2\right)\right) \\ &+ \frac{1}{CRN} \sum_{r=0}^{R-1} \sum_{m \in S_t^{\lambda_r}} \sum_{j=0}^{N-1} \left(4\eta^2 \frac{1}{N^2C^2}\left(N^2C^2r^2\|\nabla f(x_t)\|^2 + \frac{Cr(M-Cr)}{M-1}\sigma_t^2\right)\right).\end{aligned}$$

Using sums over indices we get

$$\begin{aligned}\mathbb{E}\left[V_t\right] &\leq \frac{R(R-1)}{2} 4\eta^2 L^2 \mathbb{E}\left[V_t\right] + \frac{N(N-1)}{2} 4\gamma^2 L^2 \mathbb{E}\left[V_t\right] \\ &+ \frac{2}{3}\gamma^2 \frac{1}{M} \sum_{m=1}^{M} \|\nabla f_m(x_t)\|^2 (N-1)(2N-1) + \frac{2}{3}\gamma^2 (N+1) \frac{1}{M} \sum_{m=1}^{M} \sigma_{m,t}^2 \\ &+ \frac{2}{3}\eta^2 \|\nabla f(x_t)\|^2 (R-1)(2R-1) + \frac{2}{3} \frac{M-C}{(M-1)C} \eta^2 \frac{R+1}{N^2} \sigma_t^2 \\ &\leq 2\eta^2 L^2 (1+R^2) V_t + \frac{2}{3}\gamma^2 \frac{1}{M} \sum_{m=1}^{M} \|\nabla f_m(x_t)\|^2 (N-1)(2N-1) \\ &+ \frac{2}{3}\eta^2 \|\nabla f(x_t)\|^2 (R-1)(2R-1) + \frac{2}{3}\gamma^2 (N+1) \frac{1}{M} \sum_{m=1}^{M} \sigma_{m,t}^2 + \frac{2}{3}\eta^2 \frac{R+1}{N^2} \frac{M-C}{(M-1)C} \sigma_t^2.\end{aligned}$$

To extract the $\mathbb{E}\left[V_t\right]$ we need to assume that $\gamma NR \leq \eta R \leq \theta \leq \frac{1}{16L}$ to have $1 - 2\eta^2 L^2(1+R^2) > 0$. This leads to

$$
\begin{aligned}
\mathbb{E}\left[V_t\right] &\leq 2\gamma^2 N^2 \frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_t)\|^2 + 2\eta^2 R^2 \|\nabla f(x_t)\|^2 + 2\gamma^2 N \frac{1}{M}\sum_{m=1}^{M}\sigma_{m,t}^2 + 2\eta^2 \frac{R}{N^2}\frac{M-C}{(M-1)C}\sigma_t^2 \\
&\leq 4\gamma^2 N^2 \frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_t) - \nabla f_m(x_\star)\|^2 + 4\gamma^2 N^2 \frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2 \\
&+ 4\gamma^2 N \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{j=0}^{N-1}\|\nabla f_m^{\pi_j}(x_t) - \nabla f_m^{\pi_j}(x_\star)\|^2 + 4\gamma^2 N \frac{1}{M}\sum_{m=1}^{M}\sigma_{m,\star}^2 \\
&+ 2\eta^2 R^2 \|\nabla f(x_t) - \nabla f(x_\star)\|^2 + 4\eta^2 R \frac{M-C}{(M-1)C}\frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_t) - \nabla f_m(x_\star)\|^2 + 4\eta^2 \frac{R}{N^2}\frac{M-C}{(M-1)C}\sigma_\star^2 \\
&\leq 8\eta^2 L(2+R^2) D_f(x_t, x_\star) + 4\eta^2 \frac{R}{N^2}\frac{M-C}{(M-1)C}\sigma_\star^2 + 4\gamma^2 N^2 \frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2 + 4\gamma^2 N \frac{1}{M}\sum_{m=1}^{M}\sigma_{m,\star}^2.
\end{aligned}
$$

□

We also need to bound the inner product.

Lemma H.3. *Assume that each f_m^j is L-smooth function and f is μ-strongly convex, then we have the following bound:*

$$
-2\theta\left\langle \frac{1}{R}\sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), x_t - x_\star\right\rangle \leq -\frac{\theta\mu}{2}\|x_t - x_\star\|^2 - \theta\left(f(x_t) - f(x_\star)\right) + \theta L V_t.
$$

Proof. We start from initial term:

$$
-2\theta\left\langle \frac{1}{R}\sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), x_t - x_\star\right\rangle = -2\theta\frac{1}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}\left\langle \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), x_t - x_\star\right\rangle.
$$

Let us consider $\left\langle \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), x_t - x_\star\right\rangle$:

$$
\begin{aligned}
\left\langle \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), x_t - x_\star\right\rangle &= f_m^{\pi_j}(x_t) - f_m^{\pi_j}(x_\star) + f_m^{\pi_j}(x_\star) - f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) + \left\langle \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), x_{m,t}^{r,j} - x_\star\right\rangle - f_m^{\pi_j}(x_t) \\
&+ f_m^{\pi_j}\left(x_{m,t}^{r,j}\right) + \left\langle \nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), x_t - x_{m,t}^{r,j}\right\rangle \\
&= \left(f_m^{\pi_j}(x_t) - f_m^{\pi_j}(x_\star)\right) + D_{f_m^{\pi_j}}\left(x_\star, x_{m,t}^{r,j}\right) - D_{f_m^{\pi_j}}\left(x_t, x_{m,t}^{r,j}\right).
\end{aligned}
$$

Plugging this identity we get

$$
\begin{aligned}
&-2\theta\frac{1}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}\left\langle\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right),x_t-x_\star\right\rangle\\
=&-2\theta\frac{1}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}\left\langle\left(f_m^{\pi_j}(x_t)-f_m^{\pi_j}(x_\star)\right)+D_{f_m^{\pi_j}}\left(x_\star,x_{m,t}^{r,j}\right)-D_{f_m^{\pi_j}}\left(x_t,x_{m,t}^{r,j}\right)\right\rangle\\
\leq&-2\theta\left(f(x_t)-f(x_\star)\right)-\frac{2\theta}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}D_{f_m^{\pi_j}}\left(x_\star,x_{m,t}^{r,j}\right)+\frac{\theta L}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}\left\|x_t-x_{m,t}^{r,j}\right\|^2\\
=&-\theta\left(f(x_t)-f(x_\star)\right)-\theta\left(f(x_t)-f(x_\star)\right)-\frac{2\theta}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}D_{f_m^{\pi_j}}\left(x_\star,x_{m,t}^{r,j}\right)\\
&+\frac{\theta L}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}\left\|x_t-x_{m,t}^{r,j}\right\|^2\\
\leq&-\frac{\theta\mu}{2}\|x_t-x_\star\|^2-\theta\left(f(x_t)-f(x_\star)\right)+\frac{\theta L}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}\left\|x_t-x_{m,t}^{r,j}\right\|^2\\
\leq&-\frac{\theta\mu}{2}\|x_t-x_\star\|^2-\theta\left(f(x_t)-f(x_\star)\right)+\theta LV_t,
\end{aligned}
$$

where

$$
V_t=\frac{1}{RCN}\sum_{r=0}^{R-1}\sum_{m\in S_t^{\lambda_r}}\sum_{j=0}^{N-1}\left\|x_t-x_{m,t}^{r,j}\right\|^2.
$$

□

Now we are ready to formulate the final theorem.

Theorem H.4. *Suppose that each f_m^j is convex and L-smooth, f is μ-strongly convex. Then provided the step size satisfies $\gamma NR\leq\eta R\leq\theta\leq\frac{1}{16L}$ the final iterate generated by Algorithm 1 satisfies*

$$
\begin{aligned}
\mathrm{E}\left[\|x_T-x_\star\|^2\right]\leq&\left(1-\frac{\theta\mu}{2}\right)^T\|x_0-x_\star\|^2+16\gamma^2\kappa n\frac{1}{M}\sum_{m=1}^{M}\left(n\|\nabla f_m(x_\star)\|^2+\sigma_{m,\star}^2\right)\\
&+16\eta^2\frac{\kappa}{n^2R}\frac{M-C}{(M-1)C}\sigma_\star^2.
\end{aligned}
$$

Proof. We start from definition of the points x_t^R and $x_{m,t}^{r,j}$:

$$
\begin{aligned}
x_t^R&=x_t-\eta\sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right),\\
x_{m,t}^{r,j}&=x_t-\eta\sum_{k=0}^{r-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_k}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{k,j}\right)-\gamma\sum_{l=0}^{j-1}\nabla f_m^{\pi_l}\left(x_{m,t}^{r,l}\right),\\
x_{t+1}&=x_t-\frac{\theta}{\eta R}(x_t-x_t^R).
\end{aligned}
$$

Let us start from distance to the solution:

$$\begin{aligned}
\|x_{t+1} - x_\star\|^2 &= \left\| x_t - \frac{\theta}{\eta R}(x_t - x_t^R) - x_\star \right\|^2 \\
&= \|x_t - x_\star\|^2 - 2\frac{\theta}{\eta R}\left\langle x_t - x_t^R, x_t - x_\star \right\rangle + \frac{\theta^2}{\eta^2 R^2}\|x_t - x_t^R\|^2 \\
&= \|x_t - x_\star\|^2 - 2\frac{\theta}{\eta R}\left\langle \eta \sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), x_t - x_\star \right\rangle \\
&+ \frac{\theta^2}{\eta^2 R^2}\left\| \eta \sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right)\right\|^2 .
\end{aligned}$$

Finally, we have

$$\begin{aligned}
\|x_{t+1} - x_\star\|^2 = \|x_t - x_\star\|^2 &- 2\theta\left\langle \frac{1}{R}\sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right), x_t - x_\star \right\rangle \\
&+ \theta^2\left\| \frac{1}{R}\sum_{r=0}^{R-1}\frac{1}{C}\sum_{m\in S_t^{\lambda_r}}\frac{1}{N}\sum_{j=0}^{N-1}\nabla f_m^{\pi_j}\left(x_{m,t}^{r,j}\right)\right\|^2 .
\end{aligned}$$

Let us apply Lemma H.3 and Lemma H.1 and we obtain

$$\begin{aligned}
\|x_{t+1} - x_\star\|^2 &\le \|x_t - x_\star\|^2 - \frac{\theta\mu}{2}\|x_t - x_\star\|^2 - \theta\left(f(x_t) - f(x_\star)\right) + \theta L V_t + \theta^2\left(2L^2 V_t + 4L(f(x_t) - f(x_\star))\right) \\
&\le \left(1 - \frac{\theta\mu}{2}\right)\|x_t - x_\star\|^2 - \theta\left(f(x_t) - f(x_\star)\right)(1 - 4L\theta) + \theta L V_t(1 + 2L\theta).
\end{aligned}$$

Using lemma we have

$$\begin{aligned}
\|x_{t+1} - x_\star\|^2 &\le \left(1 - \frac{\theta\mu}{2}\right)\|x_t - x_\star\|^2 - \theta\left(f(x_t) - f(x_\star)\right)(1 - 4L\theta) \\
&+ \theta L(1 + 2L\theta)\left(8\eta^2 L(2 + R^2)D_f(x_t, x_\star) + 4\eta^2\frac{R}{N^2}\frac{M - C}{(M-1)C}\sigma_\star^2\right) \\
&+ \theta L(1 + 2L\theta)\left(4\gamma^2 N^2\frac{1}{M}\sum_{m=1}^{M}\|\nabla f_m(x_\star)\|^2 + 4\gamma^2 N\frac{1}{M}\sum_{m=1}^{M}\sigma_{m,\star}^2\right) \\
&\le \left(1 - \frac{\theta\mu}{2}\right)\|x_t - x_\star\|^2 - \theta\left(f(x_t) - f(x_\star)\right)\left(1 - 4L\theta - 8L(1 + 2L\theta)\eta^2 L(2 + R^2)\right) \\
&+ 4\theta L(1 + 2L\theta)\eta^2\frac{R}{N^2}\frac{M - C}{(M-1)C}\sigma_\star^2 \\
&+ 4\gamma^2 N\theta L(1 + 2L\theta)\left(\frac{1}{M}\sum_{m=1}^{M}N\|\nabla f_m(x_\star)\|^2 + \frac{1}{M}\sum_{m=1}^{M}\sigma_{m,\star}^2\right).
\end{aligned}$$

Using $\gamma N R \le \eta R \le \theta \le \frac{1}{16L}$ we have that $\left(1 - 4L\theta - 8L(1 + 2L\theta)\eta^2 L(2 + R^2)\right) \ge 0$, it leads to

$$\begin{aligned}
\|x_{t+1} - x_\star\|^2 \le \left(1 - \frac{\theta\mu}{2}\right)\|x_t - x_\star\|^2 &+ 8\theta L\eta^2\frac{R}{N^2}\frac{M - C}{(M-1)C}\sigma_\star^2 \\
&+ 8\gamma^2 N\theta L\left(\frac{1}{M}\sum_{m=1}^{M}N\|\nabla f_m(x_\star)\|^2 + \frac{1}{M}\sum_{m=1}^{M}\sigma_{m,\star}^2\right).
\end{aligned}$$

Unrolling this recursion leads to

$$\|x_T - x_\star\|^2 \leq \left(1 - \frac{\theta\mu}{2}\right)^T \|x_0 - x_\star\|^2 + 16\frac{L}{\mu}\eta^2 \frac{R}{N^2}\frac{M-C}{(M-1)C}\sigma_\star^2 + 16\gamma^2 N \frac{L}{\mu}\left(\frac{1}{M}\sum_{m=1}^{M} N\|\nabla f_m(x_\star)\|^2 + \frac{1}{M}\sum_{m=1}^{M}\sigma_{m,\star}^2\right).$$

This finishes the proof. □

I. Deterministic client shuffling

In this section, we discuss how to extend our method's applicability beyond random reshuffling to any (including deterministic) reshuffling. We discuss each setup individually.

I.1. f_m^j is strongly convex

In the provided analysis, we do not specify the type of reshuffling, which means that this result can be applied to any, including deterministic, type of client shuffling. However, in this case, we have to slightly adjust the analysis as we cannot take expectations because client sampling is not necessarily random. The absence of randomization means that we need to consider the worst-case scenario instead of the average and the bound of $\frac{1}{C}\sum_{m\in S_t^{\lambda_r}} \sigma_{m,\text{Shuffle}}^2$ will be more significant.

I.2. f_m is strongly convex

Similarly, in the analysis of the case when only f_m is μ-convex, we do not specify the type of reshuffling. Therefore, we can apply any, including deterministic, shuffling of clients. As in the previous case, we cannot use expectations, and the bound of $\sigma_{r,\text{client}}^2$ will be up to R times larger, similarly to (Mishchenko et al., 2020, Theorem 5 (option 1)), but applied to the shuffling of clients.

I.3. f is strongly convex

Finally, our analysis uses Lemma 1 from Mishchenko et al. (2020) for the most restrictive case. In the case of any, including deterministic, permutations, the term connected to client shuffling $16\eta^2\frac{\kappa}{n^2 R}\frac{M-C}{(M-1)C}\sigma_\star^2$ will also be up to R times larger, and we would have $16\eta^2\frac{\kappa}{n^2}\frac{M-C}{(M-1)C}\sigma_\star^2$ appearing in our bound of Theorem C.3 since we cannot take expectations.

J. Extra Numerical Experiments

In this section, we provide additional numerical experiments missing in the main part. The setup is exactly the same as described before. We note that the observations that we can make for the extra experiments are consistent with the conclusions provided in the main paper.

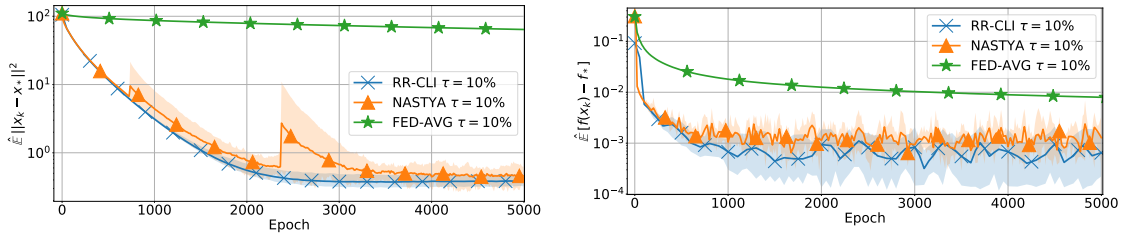

Figure 5: Training `Logistic Regression` on `a3a`, with $n = 12$ clients. Theoretical global step size and tuned local step sizes. Partial participation with 3 clients per round with 10 local steps.

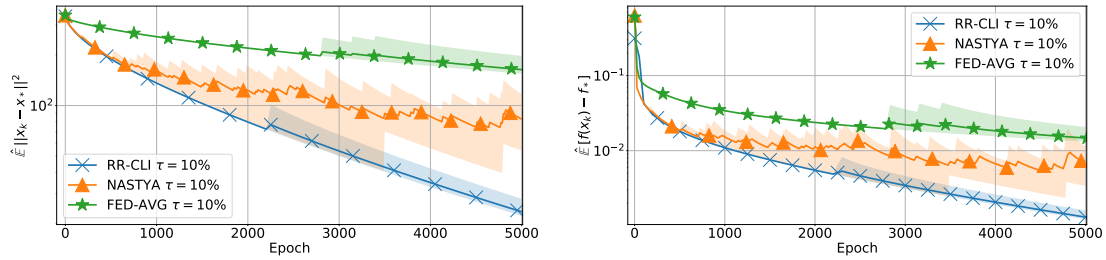

Figure 6: Training `Logistic Regression` on `w3a` with $n = 12$ clients. Theoretical global step size and tuned local step sizes. Partial participation with 3 clients per round with 10 local steps.

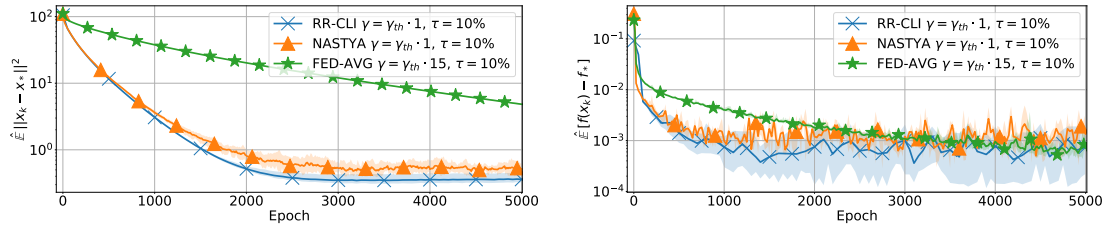

Figure 7: Training `Logistic Regression` on `a3a` with $n = 12$ clients. Theoretical global step size. Local step sizes are multipliers of theoretical. PP with 3 clients per round with, 10 local steps.

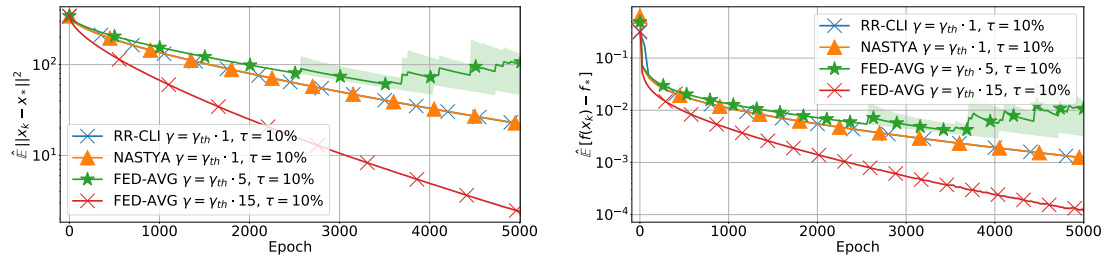

Figure 8: Training `Logistic Regression` on `w3a` with $n = 12$ clients. Theoretical global step size and tuned local step sizes. Partial participation with 3 clients per round with 10 local steps.

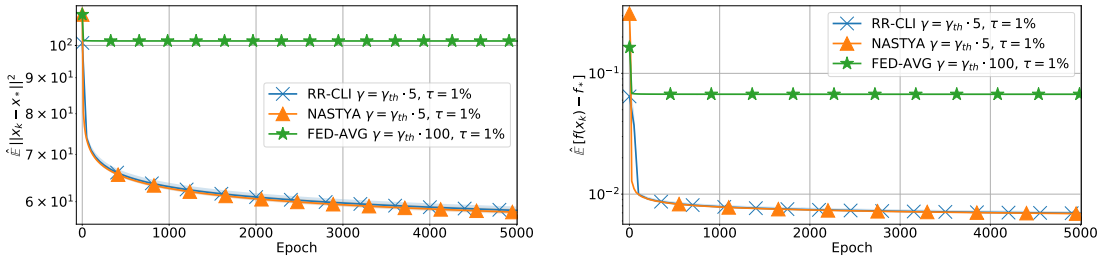

Figure 9: Training `Logistic Regression` on `w3a`, with $n = 12$ clients. Theoretical global step size. Local step sizes are multipliers of theoretical. Partial participation with 3 clients per round, 100 local step. Local and global step size are decaying $\propto \frac{1}{1+\#\text{passedepochs}}$. Local gradient estimators are computed with 1% of local samples.

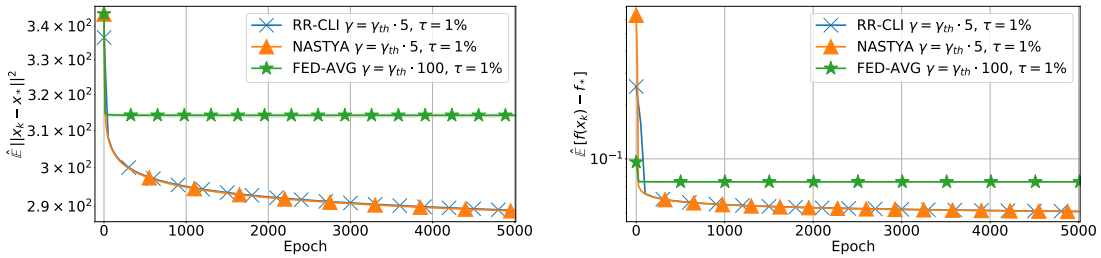

Figure 10: Training `Logistic Regression` on `w3a`, with $n = 12$ clients. Theoretical global step size. Local step sizes are multipliers of theoretical. Partial participation with 3 clients per round, 100 local step. Local and global step size are decaying $\propto \frac{1}{1+\#\text{passedepochs}}$. Local gradient estimators are computed with 1% of local samples.