Distributed Personalized Empirical Risk Minimization

Anonymous Author(s)*

ABSTRACT

This paper advocates a new paradigm Personalized Empirical Risk Minimization (PERM) to facilitate learning from heterogeneous data sources without imposing stringent constraints on computational resources shared by participating devices. In PERM, we aim at learning a distinct model for each client by personalizing the aggregation of local empirical losses by effectively estimating the statistical discrepancy among data distributions, which entails optimal statistical accuracy for all local distributions and overcomes the data heterogeneity issue. To learn personalized models at scale, we propose a distributed algorithm that replaces the standard model averaging with model shuffling to simultaneously optimize PERM objectives for all devices. This also allows to learn distinct model architectures (e.g., neural networks with different number of parameters) for different clients, thus confining to underlying memory and compute resources of individual clients. We rigorously analyze the convergence of proposed algorithm and conduct experiments that corroborate the effectiveness of proposed paradigm.

KEYWORDS

federated learning

ACM Reference Format:

Anonymous Author(s). 2018. Distributed Personalized Empirical Risk Minimization. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX)*. ACM, New York, NY, USA, 24 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Recently *federated learning* (FL) has emerged as an alternative paradigm to centralized learning to encourage federated model sharing and create a framework to support edge intelligence by shifting model training and inference from data centers to potentially scattered—and perhaps self-interested—systems where data is generated. While undoubtedly being a better paradigm than centralized learning, FL usually suffers from *heterogeneity* of data and compute resources among participants. To mitigate the negative effect of data heterogeneity (non-IIDness), two common approaches are clustering and personalization. The key idea behind the clustering based methods [16] is to partition the devices into clusters (coalitions) of similar data distributions and then learn a single shared model for all clients within each cluster. While appealing, the partitioning methods are limited to heuristic ideas such as clustering based on geographical distribution of devices without taking the actual data distributions into account. In personalization based methods [6, 8, 10, 17, 22], the idea is to learn a distinct *personalized model* for each device alongside the global model, which can be unified as minimizing a bi-level optimization problem [9]. Existing personalization techniques also suffer from two issues. First, as the number of clients grow, while the number of training data increases, the number of parameters to be learned increases which limits to increase the number of clients beyond a certain point to balance data and overall model complexity tradeoff– a phenomenon known as incidental parameters problem [13]. Moreover, since the knowledge transfer among data sources happens through the single global model, it might lead to suboptimal results. To see this, consider an extreme example, where half of the users have identical data distributions, say $\mathcal{D}$, while other half share a data distribution that is completely different, say $-\mathcal{D}$. In this case, the global model obtained by optimizing averaged local losses converges to a solution that suffers from low test accuracy on all local distributions and it would be preferable to learn a model for each client solely based on its local data.

The aforementioned issues lead to a fundamental question: *What is the best strategy to learn from heterogeneous data sources to achieve optimal statistical accuracy w.r.t. each data source, without imposing stringent constraints on computational resources shared by participating devices?*. To answer this question, we advocate a new paradigm dubbed as Personalized Empirical Risk Minimization (PERM) to facilitate learning from massively fragmented private data under resource constraints. Motivated by generalization bounds in multiple source domain adaptation [2, 5, 12, 18], in PERM we aim to learn a distinct model for each client by *personalizing the aggregation* of empirical losses of different data sources. That is, for each client, its personalized model is learnt on a distinct weighted-ERM problem, and the mixing weight is determined by the discrepancy between distributions. While PERM overcomes the data heterogeneity issue, the number of optimization problems (i.e., distinct personalized ERMs) to be solved scales linearly with the number of data sources. To simultaneously optimize all objectives in a scalable and computationally efficient manner, we propose a novel idea which replaces the standard *model averaging* in federated learning with *model shuffling* and establish its convergence rate. This also allows us to learn distinct model architectures (e.g., neural networks with different number of parameters) for different clients.

2 PERSONALIZED EMPIRICAL RISK MINIMIZATION

In this section we formally state the problem and introduce PERM as an ideal paradigm to learning from heterogeneous data sources. We assume there are N distributed devices where each holds a distinct data shard $\mathcal{S}_i = \{(\boldsymbol{x}_{i,j}, y_{i,j})\}_{j=1}^{n_i}$ with n_i training samples that are realized by a source distribution $\mathcal{D}_i$ over instance space $\Xi = \mathcal{X} \times \mathcal{Y}$. The data distributions across the devices are not independently and identically distributed (non-IID or *heterogeneous*), i.e., $\mathcal{D}_1 \neq \mathcal{D}_2 \neq \ldots \neq \mathcal{D}_m$, and each distribution corresponds to a local

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

generalization error or *true risk* $\mathcal{L}_i(h) = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}_i}[\ell(h(\boldsymbol{x}), y)]$ on *unseen* samples for any model $h \in \mathcal{H}$, where $\mathcal{H}$ is the hypothesis set (e.g., a linear model or a deep neural network) and a given convex or non-convex loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$. We use $\widehat{\mathcal{L}}_i(h) = (1/n_i)\sum_{(\boldsymbol{x},y)\in\mathcal{S}_i} \ell(h(\boldsymbol{x}), y))$ to denote the *local empirical risk* or training loss at ith data shard $\mathcal{S}_i$ with n_i samples. We seek to collaboratively learn a model or personalized models that entail a good generalization on all local distributions, minimizing true risk $\mathcal{L}_i(\cdot), i = 1, \ldots, N$ for all data sources (all-for-all [7]).

A simple non-personalized solution for FL aims to minimize a (weighted) *empirical risk* over all data shards in a communication-efficient manner [11]:

$$\arg\min_{h\in\mathcal{H}} \sum_{i=1}^{N} \alpha(i)\widehat{\mathcal{L}}_i(h) \text{ with } \boldsymbol{\alpha} \in \Delta_N, \qquad \text{(WERM)}$$

where $\Delta_N = \{\boldsymbol{\alpha} \in \mathbb{R}_+^N \mid \sum_{i=1}^N \alpha(i) = 1\}$ denotes the simplex set.

To motivate our proposal, let us consider the empirical loss $\sum_{i=1}^m \alpha(i)\widehat{\mathcal{L}}_i(h)$ in WERM with fixed mixing weights $\boldsymbol{\alpha} \in \Delta_N$, and denote the optimal solution by $\widehat{h}_{\boldsymbol{\alpha}}$. The excess risk of the learned model $\widehat{h}_{\boldsymbol{\alpha}}$ on ith local distribution $\mathcal{D}_i$ w.r.t. the optimal local model $h_i^* = \arg\min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_i}(h)$ (i.e., all-for-one) can be bounded by [12]

$$\mathcal{L}_i\left(\widehat{h}_{\boldsymbol{\alpha}}\right) \leq \mathcal{L}_i\left(h_i^*\right) + \sum_{j=1}^{N} \alpha(j)\mathrm{R}_j(\mathcal{H}) + 2\sum_{j=1}^{N} \alpha(j)\mathrm{disc}_{\mathcal{H}}\left(\mathcal{D}_j, \mathcal{D}_i\right) + O\left(\sqrt{\sum_{j=1}^{N} \frac{\alpha_j^2}{n_j}}\right) \qquad \text{(GEN)}$$

where $\mathrm{R}_j(\mathcal{H})$ is the empirical Radamacher complexity $\mathcal{H}$ w.r.t. $\mathcal{S}_i$, and $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_i, \mathcal{D}_j)$ is a pseudo-distance on the set of probability measures on Ξ to assess the discrepancy between the distributions $\mathcal{D}_i$ and $\mathcal{D}_j$ with respect to the hypothesis class $\mathcal{H}$ as defined below [2]:

Definition 1. *For a model space $\mathcal{H}$ and $\mathcal{D}, \mathcal{D}'$ two probability distributions on Ξ,*

$$\mathrm{disc}_{\mathcal{H}}\left(\mathcal{D}, \mathcal{D}'\right) = \sup_{h\in\mathcal{H}} |\mathbb{E}_{\xi\sim\mathcal{D}}(\ell(h,\xi)) - \mathbb{E}_{\xi'\sim\mathcal{D}'}(\ell(h,\xi'))|$$

Intuitively, the discrepancy between the two distributions is large, if there exists a predictor that performs well on one of them and badly on the other. On the other hand, if all functions in the hypothesis class perform similarly on both, then $\mathcal{D}$ and $\mathcal{D}'$ have low discrepancy.

From GEN, it can be observed that mismatch between pairs of distributions limits the benefits of ERM on all distributions. Indeed, the generalization risk will blow up when the distribution divergence term $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_i, \mathcal{D}_j)$ is large. When the discrepancys are small or almost zero, it leads to an ideal sample complexity $1/\sqrt{n}$ where $n = n_1 + n_2 + \ldots + n_N$ is the total number of samples, which could have been obtained in IID setting with $\alpha(j) = 1/N$. Also, we note that even if global model achieves small training error over union of all data (e.g., overparametrized setting), the divergence term still remains which illustrates the poor performance of global model on $\mathcal{D}_i$. This implies that even personalization of global model as in WERM can not entail a good generalization on all local distributions as no positive knowledge transfer among data sources occurs.

Interestingly the bound suggest that seeking optimal accuracy on *each* local distribution requires choosing a distinct mixing of local losses for each client i that minimizes the right-hand side of GEN. This indicates that in an *ideal setting*, we can achieve the best accuracy for each local distribution $\mathcal{D}_i$ by *personalizing* the WERM, i.e., (i) first estimating $\boldsymbol{\alpha}_i, i = 1, 2, \ldots, N$ for each client individually, then (ii) solving a variant of WERM for each client with obtained mixing parameters:

$$\arg\min_{h\in\mathcal{H}_i} \sum_{j=1}^{N} \alpha_i(j)\widehat{\mathcal{L}}_j(h) \quad \text{for} \quad i = 1, 2, \ldots, N \qquad \text{(PERM)}$$

By doing this each device achieves the optimal local generalization error by *learning who to learn with* based on number of samples at each source and mismatch between its data distribution with other clients. Compared to WERM, in PERM since we solve a different aggregated empirical loss for each client, we can pick a different model space/model architecture $\mathcal{H}_i$ for each client to meet its available computational resources.

While this two-stage method is guaranteed to entail optimal test accuracy for all local distributions $\mathcal{D}_i$, however, making it scalable requires overcoming two issues. First, estimating the statistical discrepancies between each pair of data sources (i.e., $\boldsymbol{\alpha}_i, i = 1, \ldots, N$) is a computing burden as it requires solving $O(N^2)$ difference of (non)-convex functions in a distributed manner and requires enough samples form each source to entail good accuracy on estimating pairwise discrepancies. Second, we need to solve N variants of the optimization problem in PERM, possibly each with a different model space, which is infeasible when the number of devices is huge (e.g., cross-device federated learning). In the next section, we propose a simple yet effective idea to overcome these issues in a computationally efficient manner.

3 PERM AT SCALE VIA MODEL SHUFFLING

In this section, we propose a method to efficiently estimate the empirical discrepancies among data sources, followed by model shuffling idea to simultaneously solve N versions of PERM to learn a model for each client. We first start by proposing a two-stage algorithm: estimating mixing parameters followed by model shuffling. Then, we propose a single loop unified algorithm which simutaneously optimizes the mixing weights and personalized models. We assume that the model space $\mathcal{H}$ is a parameterized by a convex set $\mathcal{W} \subseteq \mathbb{R}^d$ and use $f_i(w) := \widehat{\mathcal{L}}_i(w) = \sum_{(x,y)\in\mathcal{S}_i} \ell(w; (x, y))$ to denote the empirical loss at ith data shard.

3.1 Warmup: a two-stage algorithm

Stage 1: Mixing parameters estimation. In the first stage we aim to efficiently estimate the pairwise discrepancy among local distributions to construct mixing parameters $\boldsymbol{\alpha}_i, i = 1, 2, \ldots N$. From generalization bound GEN and Definition 1, a direct solution to estimate $\boldsymbol{\alpha}_i$ is to solve the following convex-nonconcave minimax problem for each client:

$$\arg\min_{\boldsymbol{\alpha}\in\Delta_N} \sum_{j=1}^{N} \alpha(j) \max_{w\in\mathcal{W}} |f_i(w) - f_j(w)| + \sum_{j=1}^{N} \frac{\alpha(j)^2}{n_j}. \quad (1)$$

However, solving the above minimax problem itself is already challenging: the inner maximization loop is a nonconcave (or difference of convex) problem, so most of existing minimax algorithms will

fail on this problem. To our best knowledge, the only provable deterministic algorithm is [26], and but it is hard to generalize it to stochastic and distributed fashion.

To overcome aforementioned issues, we make two reasonable relaxations. First, inspired by notion of average drift at optimum as a right metric to measure the effect of data heterogeneity in federated learning [23], we propose to measure discrepancy at the optimal solution obtained by solving averaged global objective, i.e., $\boldsymbol{w}^* := \arg\min_{\boldsymbol{w}\in\mathcal{W}}(1/N)\sum_{i=1}^N f_i(\boldsymbol{w})$. That is, we use $|f_i(\boldsymbol{w}^*) - f_j(\boldsymbol{w}^*)|$ to replace $\sup_{\boldsymbol{w}}|f_i(\boldsymbol{w}) - f_j(\boldsymbol{w})|$. By doing this, solving the supremum problem reduces to a simple minimization problem for each client, given the optimal global solution. Then we relax the difference between function values to the difference of gradients, i.e., gradient dissimilarity. To see this, for the difference of pair of losses $f_i(\boldsymbol{w}) - f_j(\boldsymbol{w})$, it is $2L$ smooth, hence we can upper bound $|f_i(\boldsymbol{w}) - f_j(\boldsymbol{w})|$ as:

$$
\begin{aligned}
&|f_i(\boldsymbol{w}) - f_j(\boldsymbol{w})| \\
&\le \left|(f_i(\boldsymbol{w}^*) - f_j(\boldsymbol{w}^*)) + \left\langle \nabla f_i(\boldsymbol{w}^*) - \nabla f_j(\boldsymbol{w}^*), \boldsymbol{w} - \boldsymbol{w}^*\right\rangle\right| + L\left\|\boldsymbol{w} - \boldsymbol{w}^*\right\|^2 \\
&\le \left|f_i(\boldsymbol{w}^*) - f_j(\boldsymbol{w}^*)\right| + \left\|\nabla f_i(\boldsymbol{w}^*) - \nabla f_j(\boldsymbol{w}^*)\right\| D + LD^2
\end{aligned}
$$

where D is the size of the domain. The above relaxation lead to solving the following tractable optimization problem to decide the per-client mixing parameters:

$$
\begin{aligned}
\boldsymbol{\alpha}_i^* = \arg\min_{\boldsymbol{\alpha}\in\Delta_N} g_i(\boldsymbol{w}^*, \boldsymbol{\alpha}) &:= \sum_{j=1}^N \alpha(j)\|\nabla f_i(w^*) - \nabla f_j(w^*)\|^2 \\
&+ \lambda\sum_{j=1}^N \alpha(j)^2/n_j
\end{aligned} \tag{2}
$$

where we added a regularization parameter λ and used the squared of gradient dissimilarity for computational convinience. Thus, obtaining all N mixing parameters requires solving a single ERM to obtain optimal global soltuion and N varaints of (2). To get the optimal solution in a communication reduced manner, we adapt the Local SGD algorithm [21] (or FedAvg [19]) and find the optimal solution in intermittent communication setting [24] where the clients work in parallel and are allowed to make K stochastic updates between two communication rounds for R consecutive rounds. The detailed steps are given in Algorithm A1 in Appendix A for completeness. After obtaining the global model $\boldsymbol{w}^R$ we optimize over $\boldsymbol{\alpha}$ in $g_i(\boldsymbol{w}^R, \boldsymbol{\alpha})$ using $T_{\boldsymbol{\alpha}}$ iterations of GD to get $\hat{\boldsymbol{\alpha}}_i$. Actually, we will show that as long as $\boldsymbol{w}^R$ converge to $\boldsymbol{w}^*$, $\hat{\boldsymbol{\alpha}}_i, i = 1, \ldots, N$ converges to solution of (2) very fast.

Definition 2 (Gradient Dissimilarity). *We define the following quantities to measure the gradient dissimilarity among local functions:*

$$
\begin{aligned}
&\zeta_{i,j}(\boldsymbol{w}) := \left\|\nabla f_i(w) - \nabla f_j(w)\right\|^2, \quad \bar{\zeta}_j(\boldsymbol{w}) := \zeta_{i,j}(\boldsymbol{w}), \\
&\zeta := \sup_{\boldsymbol{w}\in\mathcal{W}} \max_{i\in[N]} \|\nabla f_i(w) - (1/N)\sum_{j=1}^N \nabla f_j(\boldsymbol{w})\|^2.
\end{aligned}
$$

The following theorem gives the convergence rate of estimated discrepancies to optimal counterparts.

Theorem 1. *Assume each f_i is L-smooth and μ-strongly-convex. If we run Algorithm A1 on $F(\boldsymbol{w}) := \frac{1}{N}\sum_{j=1}^N f_j(\boldsymbol{w})$ with $\gamma = \Theta\left(\frac{\log(RK)}{\mu RK}\right)$ for R rounds with synchronization gap K, it holds that $\forall i \in [N]$*

$$
\mathbb{E}\|\hat{\boldsymbol{\alpha}}_i - \boldsymbol{\alpha}_i^*\|^2 \le \tilde{O}\left(\exp\left(-\frac{T_{\boldsymbol{\alpha}}}{\kappa_g}\right) + \kappa_g^2\bar{\zeta}_i(\boldsymbol{w}^*)L^2\left(\frac{D^2}{RK} + \frac{\kappa\zeta^2}{\mu^2R^2} + \frac{\delta^2}{\mu^2NRK}\right)\right)
$$

The implication of Theorem 1 is that even we solve (2) at $\boldsymbol{w}^R$, the algorithm will eventually converge to optimal solution of (2) at $\boldsymbol{w}^*$.

Stage 2: Scalable personalized optimization with model shuffling. After obtaining the per client mixing parameters, in the second stage to learn personalized models, we aim at solving N different personalized variants of PERM i.e., $\Phi(\hat{\boldsymbol{\alpha}}_1, \boldsymbol{v}), \ldots \Phi(\hat{\boldsymbol{\alpha}}_N, \boldsymbol{v})$ where

$$
\min_{\boldsymbol{v}\in\mathcal{W}} \Phi(\hat{\boldsymbol{\alpha}}_i, \boldsymbol{v}) := \frac{1}{N}\sum_{j=1}^N \hat{\alpha}_i(j) f_j(\boldsymbol{v}). \tag{3}
$$

Here we devise an iterative algorithm based on Local SGD to solve these N optimization problems in parallel with *no extra overhead*. The idea is to replace the model averaging in vanilla distributed (Local) SGD with *model shuffling*. Specifically, as shown in Algorithm 1 the algorithm proceeds for R epochs where each epoch runs for N communication rounds. At the beginning of each epoch r the server generates a random permutation σ_r over N clients. At each communication round j within the epoch, the server sends the model of client i to client $i_j = (i+j)\%N$ in the permutation σ_r along with $\alpha_i(i_j)$. After receiving a model from server, the client updates the received model for K local steps and returns it back to the server. As it can be seen, the updates of each loss $\Phi(\hat{\boldsymbol{\alpha}}_i, \boldsymbol{v}), i = 1, 2, \ldots, N$ during an epoch is equivalent to sequentially processing individual losses in (3) which can be considered as permutation-based SGD but with the difference that each component now is updated for K steps. By *interleaving the permutations*, we are able to simultaneously optimize all N objectives. The following theorem establishes the convergence rate

Theorem 2. *Assume each f_i is L-smooth, μ-strongly-convex and with gradient bounded by G, i.e., $\sup_{\boldsymbol{w}\in\mathcal{W}}\|\nabla f_i(\boldsymbol{w})\| \le G$. Let $\boldsymbol{\alpha}_i^*$ be the solution of (2). Then if we run Algorithm 1 on the $\hat{\boldsymbol{\alpha}}_i$ obtained from Algorithm A1, then Algorithm 1 with $\eta = \Theta\left(\frac{\log(NKR^3)}{\mu R}\right)$ will output the solution $\hat{\boldsymbol{v}}_i, \forall i \in [N]$, such that with probability at least $1-p$, the following statement holds:*

$$
\begin{aligned}
&\mathbb{E}[\Phi(\boldsymbol{\alpha}_i^*, \hat{\boldsymbol{v}}_i) - \Phi(\boldsymbol{\alpha}_i^*, \boldsymbol{v}^*(\boldsymbol{\alpha}_i^*))] \\
&\quad\le \tilde{O}\left(\frac{D^2L}{NKR^2} + \frac{L\delta^2}{\mu^2R} + \left(\frac{L^4+N}{\mu^4R^2}\right)LG^2N\log(1/p)\right) \\
&\quad + \kappa^2L\tilde{O}\left(\exp\left(-\frac{T_{\boldsymbol{\alpha}}}{\kappa_g}\right) + \kappa_g^2\bar{\zeta}_i(\boldsymbol{w}^*)L^2\left(\frac{D^2}{RK} + \frac{\kappa\zeta^2}{\mu^2R^2} + \frac{\delta^2}{\mu^2NRK}\right)\right),
\end{aligned}
$$

where the expectation is taken over randomness of Algorithm A1.

The above theorem shows that, even though we run the optimization on $\Phi(\hat{\boldsymbol{\alpha}}_i, \boldsymbol{v})$, our obtained model $\hat{\boldsymbol{v}}_i$ will still converge to the optimal solution of $\Phi(\boldsymbol{\alpha}_i^*, \boldsymbol{v})$. The convergence rate is contributed from two parts: convergence of $\hat{\boldsymbol{\alpha}}_i$ (Algorithm A1) and convergence of personalized model $\hat{\boldsymbol{v}}_i$ (Algorithm 1). Notice that, for the convergence rate of $\hat{\boldsymbol{v}}_i$, we roughly recover the optimal rate of shuffling SGD [1], which is $O(\frac{1}{R^2})$. However, we suffer from a $O(\frac{\delta^2}{R})$ term since each client runs vanilla SGD on their local data (the `SGD-Update` procedure in Algorithm 1). A recent work [4] also

Algorithm 1: Shuffling Local SGD

Input: Clients 1, ..., N, Number of Local Steps K , Number of Epoch R, mixing parameter $\hat{\boldsymbol{\alpha}}_1, ..., \hat{\boldsymbol{\alpha}}_N$

Epoch for $r = 0, ..., R-1$ **do**
  Server generates permutation $\sigma_r : [N] \mapsto [N]$.
  parallel for $i = 1, ..., N$ **do**
    Client i sets initial model $\boldsymbol{v}_i^{r,0} = \boldsymbol{v}_i^r$.
    for $j = 1, ..., N$ **do**
      Set indices $i_j = \sigma_r((i+j) \mod N)$.
      Server sends $\boldsymbol{v}_i^{r,j}$ to Client i_j.
      $\boldsymbol{v}_i^{r,j+1} = \texttt{SGD-Update}(\boldsymbol{v}_i^{r,j}, \eta, i_j, K, \hat{\boldsymbol{\alpha}}_i)$.
    Client i does projection: $\boldsymbol{v}_i^{r+1} = \mathcal{P}_{\mathcal{W}}(\boldsymbol{v}_i^{r,N})$.
Output: $\hat{\boldsymbol{v}}_i = \boldsymbol{v}_i^R, \forall i \in [N]$.

SGD-Update$(\boldsymbol{v}, \eta, j, K, \boldsymbol{\alpha})$
  Initialize $\boldsymbol{v}^0 = \boldsymbol{v}$
  for $t = 0, ..., K-1$ **do**
    $\boldsymbol{v}^t = \boldsymbol{v}^{t-1} - \eta\alpha(j)N\nabla f_j(\boldsymbol{v}^{t-1}; \xi_j^{t-1})$
  Output: $\boldsymbol{v}^K$

considering client-level shuffling idea. The differences are two-fold: 1) they only consider training one model while we employ shuffling idea to simultaneously learn N models 2) they work with fixed $\boldsymbol{\alpha}$ while we have toshow that the algorithm can converge to the true optimal solution of $\Phi(\boldsymbol{\alpha}_i^*, \boldsymbol{v})$ given that we only optimize on a surrogate function $\Phi(\hat{\boldsymbol{\alpha}}_i, \boldsymbol{v})$.

3.2 A unified single loop algorithm

In practice, a single-loop algorithm is perferred since it is easy to implement. We hence turn to introducing a single stage algorithm that jointly optimizes $\boldsymbol{\alpha}_i$s and $\boldsymbol{v}_i$s as depicted in Algorithm 2 by intertwining the two stages in Algorithm A1 and Algorithm 1 in a single unified method. At each communication round, the clients computes gradients on global model, on their data, after that serve collects these gradients and does one step mini-batch sgd update on global model, and then updates mixing parameters. Next we proceed to update the personalized models similar to Algorithm 1. We note that, unlike the two stages method where the mixing parameters are computed at the final global model, here the mixing parameters are updated adaptively based on intermediate global models.

Theorem 3. *Assume each f_i is L-smooth, μ-strongly-convex and with gradient bounded by G, i.e., $\sup_{\boldsymbol{w}\in\mathcal{W}} \|\nabla f_i(\boldsymbol{w})\| \leq G$. Let $\boldsymbol{\alpha}_i^*$ be the solution of (2). Then if we run Algorithm 2 with $\eta = \Theta\left(\frac{\log(NKR^3)}{\mu R}\right)$ and $\gamma = \Theta\left(\frac{\log(NKR^3)}{\mu R}\right)$, it will output the solution $\hat{\boldsymbol{v}}_i, \forall i \in [N]$, such that with probability at least $1-p$, the following statement holds:*

$$\mathbb{E}[\Phi(\boldsymbol{\alpha}_i^*, \hat{\boldsymbol{v}}_i) - \Phi(\boldsymbol{\alpha}_i^*, \boldsymbol{v}_i^*)] \leq O\left(\frac{LD^2}{NKR^3}\right) + \tilde{O}\left(\left(\frac{\kappa^4 L}{R^2} + \frac{NL}{\mu^2 R^2}\right) G^2 N \log(1/p) + \frac{L\delta^2}{\mu^2 R}\right) + \tilde{O}\left(\frac{\kappa^2\kappa_g^2 L^3 \bar{\zeta}_i(\boldsymbol{w}^*) DG}{R} + LR^2\left(1 - \frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}} + \frac{L\kappa_g^2\kappa^2\bar{\zeta}_i(\boldsymbol{w}^*)\delta^2}{\mu^2 M}\right),$$

Algorithm 2: Single Loop PERM

Input: Clients 1, ..., N, Number of Local Steps K , Number of Epoch R, Initial mixing parameter $\boldsymbol{\alpha}_1^0 =, ..., \boldsymbol{\alpha}_N^0 = \bar{\boldsymbol{\alpha}} = [1/N, ..., 1/N]$.

Epoch for $r = 0, ..., R-1$ **do**
  Server generates permutation $\sigma_r : [N] \mapsto [N]$.
  parallel for Client $i = 1, ..., N$ **do**
    Client i sets initial model $\boldsymbol{v}_i^{r,0} = \boldsymbol{v}_i^r$.
    for $j = 1, ..., N$ **do**
      Set indices $i_j = \sigma_r((i+j) \mod N)$.
      Server sends $\boldsymbol{v}_i^{r,j}$ to client i_j .
      $\boldsymbol{v}_i^{r,j+1} = \texttt{SGD-Update}(\boldsymbol{v}_i^{r,j}, \eta, i_j, K, \boldsymbol{\alpha}_i^r)$.
        `// Personalized model update`
    Client i does projection: $\boldsymbol{v}_i^{r+1} = \mathcal{P}_{\mathcal{W}}(\boldsymbol{v}_i^{r,N})$.
  $\boldsymbol{w}^{r+1} = \mathcal{P}_{\mathcal{W}}(\boldsymbol{w}^r - \gamma \frac{1}{N}\sum_{i=1}^N \frac{1}{M}\sum_{j=1}^M \nabla f_i(\boldsymbol{w}^r, \xi_{i,j}^r))$
    `// Global model update`
  Compute $\boldsymbol{\alpha}_i^{r+1}$ by running $T_{\boldsymbol{\alpha}}$ steps GD on $g_i(\boldsymbol{w}^{r+1}, \boldsymbol{\alpha})$
    `//` $\boldsymbol{\alpha}$ `update`
Output: $\hat{\boldsymbol{v}}_i = \boldsymbol{v}_i^R$, $\hat{\boldsymbol{\alpha}}_i = \boldsymbol{\alpha}_i^R, \forall i \in [N]$.

SGD-Update$(\boldsymbol{v}, \eta, j, K, \boldsymbol{\alpha})$
  Initialize $\boldsymbol{v}_j^0 = \boldsymbol{v}$
  for $t = 0, ..., K-1$ **do**
    $\boldsymbol{v}^t = \boldsymbol{v}^{t-1} - \eta\boldsymbol{\alpha}(j)N\nabla f_j(\boldsymbol{v}_j^{t-1}; \xi_j^{t-1})$
  Output: $\boldsymbol{v}^K$

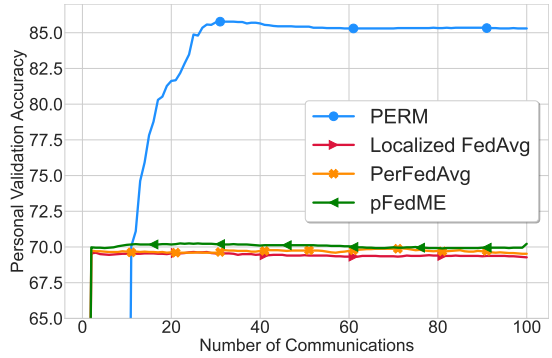
where the expectation is took over the randomness of stochastic samples in Algorithm 2.

4 EXPERIMENT

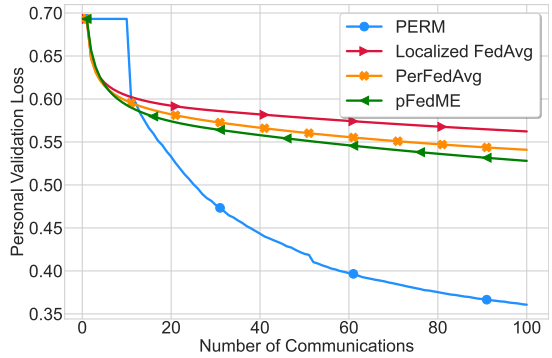
To demonstrate the superior effectiveness of our proposed single-loop PERM algorithm compared to other existing personalization methods, we conducted an experiment using synthetic data generated according to the following specifications. We consider a scenario with a total of N clients, where we draw samples from the distribution $\mathcal{N}(\mu_1, \Sigma_i)$ for half of the clients, denoted by $i \in [1, \frac{N}{2}]$, and from $\mathcal{N}(\mu_2, \Sigma_i)$ for the remaining clients, denoted by $i \in [\frac{N}{2}, N]$. Following the approach outlined in [14], we adopt a uniform variance for all samples, with $\Sigma_{k,k} = k^{-1.2}$. Subsequently, we generate a labeling model using the distribution $\mathcal{N}(\mu_w, \Sigma_w)$.

Given a data sample $\boldsymbol{x} \in \mathbb{R}^d$, the labels are generated as follows: clients 1, ..., $\frac{N}{2}$ assign labels based on $y = \text{sign}(\boldsymbol{w}^\top \boldsymbol{x})$, while clients $\frac{N}{2}+1, ..., N$ assign labels based on $y = \text{sign}(-\boldsymbol{w}^\top \boldsymbol{x})$. For this specific experiment, we set $\mu_1 = 0.2$, $\mu_2 = -0.2$, and $\mu_w = 0.1$. The data dimension is $d = 60$, and there are 2 classes in the output. We have a total of 50 clients, each generating 500 samples following the aforementioned guidelines. We train a logistic regression model on each client's data.

To demonstrate the superiority of our PERM algorithm, we conducted a performance comparison against other prominent personalized approaches, including the fined-tuned model of FedAvg [19] (referred to as localized FedAvg), perFedAg [8], and pFedME [22]. The results in Figure 1 highlight PERM's efficient learning of personalized models for individual clients. In contrast, competing methods



(a) Personalized Accuracy



(b) Personalized Loss

Figure 1: Comparative Analysis of Personalization methods, including our single-loop PERM algorithm, localized FedAvg, perFedAg, and pFedME, with synthetic data. The disparity in personalized accuracy and loss highlights PERM's capability in leveraging relevant client correlations.

relying on globally trained models struggle to match PERM's effectiveness in highly heterogeneous scenarios, as seen in personalized accuracy and loss. This showcases PERM's exceptional ability to leverage relevant client learning.

5 CONCLUSION

This paper introduces a new *data&system-aware* paradigm for learning from multiple heterogeneous data sources to achieve optimal statistical accuracy across all data distributions without imposing stringent constraints on computational resources shared by participating devices. The proposed PERM schema, though simple, provides an efficient solution to enable each client to learn a personalized model by *learning who to learn with* via personalizing the aggregation of data sources through an efficient empirical statistical discrepancy estimation module. PERM can also be employed in other learning settings with multiple sources of data such as domain adaptation and multi-task learning to entail optimal statistical accuracy. To efficiently solve all aggregated personalized losses, we propose a model shuffling idea to optimizes all losses in parallel. This also enables us to learn models with varying complexity for different devices to meet their available resources.

REFERENCES

[1] Kwangjun Ahn, Chulhee Yun, and Suvrit Sra. Sgd with shuffling: optimal rates without component convexity and large epoch requirements. *Advances in Neural Information Processing Systems*, 33:17526–17535, 2020.

[2] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79:151–175, 2010.

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

[4] Yae Jee Cho, Pranay Sharma, Gauri Joshi, Zheng Xu, Satyen Kale, and Tong Zhang. On the convergence of federated averaging with cyclic client participation. *arXiv preprint arXiv:2302.03109*, 2023.

[5] Koby Crammer, Michael Kearns, and Jennifer Wortman. Learning from multiple sources. *Journal of Machine Learning Research*, 9(8), 2008.

[6] Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020.

[7] Mathieu Even, Laurent Massoulié, and Kevin Scaman. On sample optimality in personalized collaborative and federated learning. In *NeurIPS 2022-36th Conference on Neural Information Processing System*, 2022.

[8] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. *Advances in Neural Information Processing Systems*, 2020.

[9] Filip Hanzely, Boxin Zhao, and Mladen Kolar. Personalized federated learning: A unified framework and universal optimization techniques. *arXiv preprint arXiv:2102.09743*, 2021.

[10] Yutao Huang, Lingyang Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. Personalized federated learning: An attentive collaboration approach. *arXiv preprint arXiv:2007.03797*, 2020.

[11] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[12] Nikola Konstantinov and Christoph Lampert. Robust learning from untrusted sources. In *International conference on machine learning*, pages 3488–3498. PMLR, 2019.

[13] Tony Lancaster. The incidental parameter problem since 1948. *Journal of econometrics*, 95(2):391–413, 2000.

[14] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 2018.

[15] Tianyi Lin, Chi Jin, and Michael I Jordan. On gradient descent ascent for nonconvex-concave minimax problems. *arXiv preprint arXiv:1906.00331*, 2019.

[16] Jie Ma, Guodong Long, Tianyi Zhou, Jing Jiang, and Chengqi Zhang. On the convergence of clustered federated learning. *arXiv preprint arXiv:2202.06187*, 2022.

[17] Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. Three approaches for personalization with applications to federated learning. *arXiv preprint arXiv:2002.10619*, 2020.

[18] Yishay Mansour and Mariano Schain. Robust domain adaptation. *Annals of Mathematics and Artificial Intelligence*, 71(4):365–380, 2014.

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

[20] Markus Schneider. Probability inequalities for kernel embeddings in sampling without replacement. In *Artificial Intelligence and Statistics*, pages 66–74. PMLR, 2016.

[21] Sebastian U Stich. Local sgd converges fast and communicates little. In *International Conference on Learning Representations*.

[22] Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems*, 33:21394–21405, 2020.

[23] Jianyu Wang, Rudrajit Das, Gauri Joshi, Satyen Kale, Zheng Xu, and Tong Zhang. On the unreasonable effectiveness of federated averaging with heterogeneous data. *arXiv preprint arXiv:2206.04723*, 2022.

[24] Blake Woodworth, Jialei Wang, Adam Smith, Brendan McMahan, and Nathan Srebro. Graph oracle models, lower bounds, and gaps for parallel stochastic optimization. *arXiv preprint arXiv:1805.10222*, 2018.

[25] Blake E Woodworth, Kumar Kshitij Patel, and Nati Srebro. Minibatch vs local sgd for heterogeneous distributed learning. *Advances in Neural Information Processing Systems*, 33:6281–6292, 2020.

[26] Zi Xu, Huiling Zhang, Yang Xu, and Guanghui Lan. A unified single-loop alternating gradient projection algorithm for nonconvex–concave and convex–nonconcave minimax problems. *Mathematical Programming*, pages 1–72, 2023.

Supplementary Material

The supplementary material is organized as follows:

- In Section A, we provide the proof of convergence of two-stage implementation of PERM (computing mixing parameters followed by learning personalized models via model shuffling using permutation-based variant of distributed SGD with periodic communication). The technical tools we develop to establish the convergence could be interesting by their own right.
- In Section B, we provide the proof of convergence of single loop implementation of PERM, where estimation of mixing parameters are coupled with learning of personalized models.
- In Section C, we report additional experiments on performance of PERM on EMNIST dataset along with effectiveness of proposed method to estimate mixing parameters.

A PROOF OF TWO STAGES ALGORITHM

A.1 Technical Lemmas

Lemma 1. *Define* $v^*(\boldsymbol{\alpha}) := \arg\min_{v\in\mathcal{W}} \Phi(\boldsymbol{\alpha}, v)$*, and assume* $\Phi(\boldsymbol{\alpha}, \cdot)$ *is* μ*-strongly convex and* L *smooth. Then,* $v^*(\cdot)$ *is* κ*-Lipschitz.*

Proof. The proof is similar to Lin et al's result on minimax objective [15]. First, according to optimality conditions we have:

$$\langle v - v^*(\boldsymbol{\alpha}), \nabla_v \Phi(\boldsymbol{\alpha}, v^*(\boldsymbol{\alpha}))\rangle \geq 0,$$
$$\langle v - v^*(\boldsymbol{\alpha}'), \nabla_v \Phi(\boldsymbol{\alpha}', v^*(\boldsymbol{\alpha}'))\rangle \geq 0$$

Substituting v with $v^*(\boldsymbol{\alpha}')$ and $v^*(\boldsymbol{\alpha})$ in the above first and second inequalities respectively yields:

$$\langle v^*(\boldsymbol{\alpha}') - v^*(\boldsymbol{\alpha}), \nabla_v \Phi(\boldsymbol{\alpha}, v^*(\boldsymbol{\alpha}))\rangle \geq 0,$$
$$\langle v^*(\boldsymbol{\alpha}) - v^*(\boldsymbol{\alpha}'), \nabla_v \Phi(\boldsymbol{\alpha}', v^*(\boldsymbol{\alpha}'))\rangle \geq 0$$

Adding up the above two inequalities yields:

$$\langle v^*(\boldsymbol{\alpha}') - v^*(\boldsymbol{\alpha}), \nabla_v \Phi(\boldsymbol{\alpha}, v^*(\boldsymbol{\alpha})) - \nabla_v \Phi(\boldsymbol{\alpha}', v^*(\boldsymbol{\alpha}'))\rangle \geq 0, \tag{4}$$

Since $\Phi(\boldsymbol{\alpha}, \cdot)$ is μ strongly convex, we have:

$$\langle v^*(\boldsymbol{\alpha}') - v^*(\boldsymbol{\alpha}), \nabla_y \Phi(\boldsymbol{\alpha}, v^*(\boldsymbol{\alpha}')) - \nabla_y \Phi(\boldsymbol{\alpha}, v^*(\boldsymbol{\alpha}))\rangle \geq \mu \|v^*(\boldsymbol{\alpha}') - v^*(\boldsymbol{\alpha})\|^2. \tag{5}$$

Adding up (4) and (5) yields:

$$\langle v^*(\boldsymbol{\alpha}') - v^*(\boldsymbol{\alpha}), \nabla_y \Phi(\boldsymbol{\alpha}, v^*(\boldsymbol{\alpha}')) - \nabla_y \Phi(\boldsymbol{\alpha}', v^*(\boldsymbol{\alpha}'))\rangle \geq \mu \|v^*(\boldsymbol{\alpha}') - v^*(\boldsymbol{\alpha})\|^2$$

Finally, using L smoothness of Φ will conclude the proof:

$$L\|v^*(\boldsymbol{\alpha}') - v^*(\boldsymbol{\alpha})\|\|\boldsymbol{\alpha} - \boldsymbol{\alpha}'\| \geq \mu \|v^*(\boldsymbol{\alpha}') - v^*(\boldsymbol{\alpha})\|^2$$
$$\iff \kappa\|\boldsymbol{\alpha} - \boldsymbol{\alpha}'\| \geq \|v^*(\boldsymbol{\alpha}') - v^*(\boldsymbol{\alpha})\|$$

□

Lemma 2 (Optimality Gap). *Let* $\Phi(\boldsymbol{\alpha}, v)$ *be defined in (3). If we assume each* f_i *is* L*-smooth and* μ*-strongly convex, then the following statement holds true:*

$$\Phi(\boldsymbol{\alpha}_i^*, \hat{v}_i) - \Phi(\boldsymbol{\alpha}_i^*, v_i^*) \leq L\left\|\hat{v}_i - v^*(\hat{\boldsymbol{\alpha}}_i)\right\|^2 + \kappa^2 L\left\|\hat{\boldsymbol{\alpha}}_i - \boldsymbol{\alpha}_i^*\right\|^2,$$

where $v_i^* = \arg\min_{v\in\mathcal{W}} \Phi(\boldsymbol{\alpha}_i^*, v)$.

Proof. We notice the following fact:

$$\Phi(\boldsymbol{\alpha}_i^*, \hat{v}_i) - \Phi(\boldsymbol{\alpha}_i^*, v_i^*) \leq \underbrace{\langle \nabla_v \Phi(\boldsymbol{\alpha}_i^*, v_i^*), \hat{v}_i - v_i^*\rangle}_{\leq 0} + \frac{L}{2}\|\hat{v}_i - v_i^*\|^2$$
$$\leq \frac{L}{2}\|\hat{v}_i - v_i^*\|^2.$$

Algorithm A1: Discrepancy Estimation at Optimum

Input: Number of clients N, number of local steps K , number of communications rounds R
for $r = 0, \ldots, R-1$ **do**
  parallel for client $i = 0, ..., N-1$ **do**
    Client i initializes model $\boldsymbol{w}_i^{r,0} = \boldsymbol{w}_i^r$.
    for $t = 0, ..., K-1$ **do**
      $\boldsymbol{w}_i^{r,t+1} = \boldsymbol{w}_i^{r,t} - \gamma \nabla f_i(\boldsymbol{w}_i^{r,t}; \xi_i^{r,t})$ where $\xi_i^{r,t}$ is a mini-batch sampled from $\mathcal{S}_i$.
    Client i sends $\boldsymbol{w}_i^{r,K}$ to Server.
  Server computes $\boldsymbol{w}^{r+1} = \mathcal{P}_{\mathcal{W}}\left(\frac{1}{N}\sum_{i=1}^N \boldsymbol{w}_i^{r,K}\right)$
  Server broadcasts $\boldsymbol{w}^{r+1}$ to all clients.
Server computes $\hat{\boldsymbol{\alpha}}_i, i = 1, 2, \ldots, N$ by running $T_{\boldsymbol{\alpha}}$ steps of GD on $g_i(w^R, \boldsymbol{\alpha})$.
Output: $\hat{\boldsymbol{\alpha}}_1, \ldots, \hat{\boldsymbol{\alpha}}_N$.

where we use the smoothness of Φ and optimality condition of $\boldsymbol{v}_i^*$ for fixed $\boldsymbol{\alpha}_i^*$. Next, due to the κ-Lipschitzness property of of $\boldsymbol{v}^*(\cdot)$ as shown in Lemma 1, it follows that:

$$\begin{aligned}\frac{L}{2}\|\hat{\boldsymbol{v}}_i - \boldsymbol{v}_i^*\|^2 &\le L\|\hat{\boldsymbol{v}}_i - \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}_i)\|^2 + L\|\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}_i) - \boldsymbol{v}_i^*\|^2 \\ &\le L\|\hat{\boldsymbol{v}}_i - \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}_i)\|^2 + \kappa^2 L\|\hat{\boldsymbol{\alpha}}_i - \boldsymbol{\alpha}_i^*\|^2.\end{aligned}$$

as desired. □

A.2 Proof of Convergence of Theorem 1

In this section we are going to prove the result in Theorem 1. To this end, we need to show that mixing parameters we compute by first learning the global model and then solving the optimization problem in objective (2) (as depicted in Algorithm A1) converges to optimal values. Notice that in Algorithm A1 we do not solve $g_i(\boldsymbol{w}^*, \boldsymbol{\alpha})$ directly, but optimize $g_i(\boldsymbol{w}^R, \boldsymbol{\alpha})$ on $\boldsymbol{\alpha}$ for $T_{\boldsymbol{\alpha}}$ iterations of GD. Hence, firstly we need to show that optimizing the surrogate function will also guarantee the convergence of output of algorithm $\widehat{\boldsymbol{\alpha}}$ to $\boldsymbol{\alpha}^*$ by deriving a property of the objective in (2). Formally the property is captured by the following lemma.

Lemma 3. *Let* $g(\boldsymbol{w}, \boldsymbol{\alpha}) := \sum_{j=1}^N \alpha_j \left\|\nabla f_i(\boldsymbol{w}) - \nabla f_j(\boldsymbol{w})\right\|^2 + \lambda \sum_{j=1}^N \alpha_j^2/n_j$ *and* $\boldsymbol{\alpha}_g^*(\boldsymbol{w}) = \arg\min_{\boldsymbol{\alpha}\in\Delta^N} g(\boldsymbol{w}, \boldsymbol{\alpha})$. *Let* $\boldsymbol{w}^R$ *be the output of Algorithm A1. Then the following statement holds:*

$$\left\|\boldsymbol{\alpha}_g^*(\boldsymbol{w}^R) - \boldsymbol{\alpha}_g^*(\boldsymbol{w}^*)\right\| \le \kappa_g^2 \sum_{j=1}^N \left(2\left\|\nabla f_i(\boldsymbol{w}^*) - \nabla f_j(\boldsymbol{w}^*)\right\|^2 + 4L^2\left\|\boldsymbol{w}^R - \boldsymbol{w}^*\right\|^2\right) 4L\left\|\boldsymbol{w}^R - \boldsymbol{w}^*\right\|^2$$

where κ_g *is the condition number of* g.

Proof. Define function

$$W(z, \boldsymbol{\alpha}) = \sum_{j=1}^N \alpha_j z_j + \lambda \sum_{j=1}^N \alpha_j^2/n_j \tag{6}$$

Apparently, $W(z, \boldsymbol{\alpha})$ is linear in z and strongly convex in $\boldsymbol{\alpha}$. Then, according to Proposition 1, $\boldsymbol{\alpha}_W^*(z) := \arg\min_{\boldsymbol{\alpha}\in\Delta^N} W(z, \boldsymbol{\alpha})$ is κ_g lipschitz in z where $\kappa_g = \frac{n_{\max}}{n_{\min}}$, i.e., $\left\|\boldsymbol{\alpha}_W^*(z) - \boldsymbol{\alpha}_W^*(z')\right\| \le \kappa_g \|z - z'\|$. Now, let us consider the objective (2):

$$g(\boldsymbol{w}, \boldsymbol{\alpha}) := \sum_{j=1}^N \alpha_j \left\|\nabla f_i(\boldsymbol{w}) - \nabla f_j(\boldsymbol{w})\right\|^2 + \lambda \sum_{j=1}^N \alpha_j^2/n_j$$

We define $\boldsymbol{\alpha}_g^*(\boldsymbol{w}) = \arg\min_{\boldsymbol{\alpha}\in\Delta^N} g(\boldsymbol{w}, \boldsymbol{\alpha})$.
We set

$$\begin{aligned}z^R &= \left[\left\|\nabla f_i(\boldsymbol{w}^R) - \nabla f_1(\boldsymbol{w}^R)\right\|^2, ..., \left\|\nabla f_i(\boldsymbol{w}^R) - \nabla f_N(\boldsymbol{w}^R)\right\|^2\right], \\ z^* &= \left[\left\|\nabla f_i(\boldsymbol{w}^*) - \nabla f_1(\boldsymbol{w}^*)\right\|^2, ..., \left\|\nabla f_i(\boldsymbol{w}^*) - \nabla f_N(\boldsymbol{w}^*)\right\|^2\right].\end{aligned}$$

Then we know that

$$\left\|\boldsymbol{\alpha}_g^*(\boldsymbol{w}^R)-\boldsymbol{\alpha}_g^*(\boldsymbol{w}^*)\right\|^2=\left\|\boldsymbol{\alpha}_W^*(z^R)-\boldsymbol{\alpha}_W^*(z^*)\right\|^2\leq\kappa_g^2\left\|z^R-z^*\right\|^2 \tag{7}$$

$$\leq\kappa_g^2\sum_{j=1}^N\left|\left\|\nabla f_i(\boldsymbol{w}^R)-\nabla f_j(\boldsymbol{w}^R)\right\|^2-\left\|\nabla f_i(\boldsymbol{w}^*)-\nabla f_j(\boldsymbol{w}^*)\right\|^2\right|^2 \tag{8}$$

$$\leq\kappa_g^2\sum_{j=1}^N\Big|\Big(\nabla f_i(\boldsymbol{w}^R)-\nabla f_j(\boldsymbol{w}^R)+\nabla f_i(\boldsymbol{w}^*)-\nabla f_j(\boldsymbol{w}^*)\Big)$$

$$\times\Big(\nabla f_i(\boldsymbol{w}^R)-\nabla f_j(\boldsymbol{w}^R)-\nabla f_i(\boldsymbol{w}^*)+\nabla f_j(\boldsymbol{w}^*)\Big)\Big|^2$$

$$\leq\kappa_g^2\sum_{j=1}^N\left\|\nabla f_i(\boldsymbol{w}^R)-\nabla f_j(\boldsymbol{w}^R)+\nabla f_i(\boldsymbol{w}^*)-\nabla f_j(\boldsymbol{w}^*)\right\|^2 4L^2\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2$$

Since $\left\|\nabla f_i(\boldsymbol{w}^R)-\nabla f_j(\boldsymbol{w}^R)\right\|\leq\left\|\nabla f_i(\boldsymbol{w}^*)-\nabla f_j(\boldsymbol{w}^*)\right\|+2L\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|$, we can conclude that

$$\left\|\boldsymbol{\alpha}_g^*(\boldsymbol{w}^R)-\boldsymbol{\alpha}_g^*(\boldsymbol{w}^*)\right\|\leq\kappa_g^2\sum_{j=1}^N\left(2\left\|\nabla f_i(\boldsymbol{w}^*)-\nabla f_j(\boldsymbol{w}^*)\right\|^2+4L^2\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\right)4L\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2.$$

□

With above lemma, to show the convergence of $\hat{\boldsymbol{\alpha}}$ to $\boldsymbol{\alpha}^*$, we do the following decomposition

$$\left\|\hat{\boldsymbol{\alpha}}-\boldsymbol{\alpha}^*\right\|^2\leq 2\left\|\hat{\boldsymbol{\alpha}}-\boldsymbol{\alpha}_g^*(\boldsymbol{w}^R)\right\|^2+2\left\|\boldsymbol{\alpha}_g^*(\boldsymbol{w}^R)-\boldsymbol{\alpha}_g^*(\boldsymbol{w}^*)\right\|^2$$

$$\leq 2(1-\mu\eta_{\boldsymbol{\alpha}})^K+2\kappa_g^2\sum_{j=1}^N\left(2\left\|\nabla f_i(\boldsymbol{w}^*)-\nabla f_j(\boldsymbol{w}^*)\right\|^2+4L^2\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\right)4L\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2.$$

Now it remains to show the convergence of Local SGD *last iterate* $\boldsymbol{w}^R$ to optimal solution $\boldsymbol{w}^*$. By convention, we use $\boldsymbol{w}^t=\frac{1}{N}\sum_{i=1}^N\boldsymbol{w}_i^t$ to denote the virtual average iterates.

Lemma 4 (One iteration analysis of Local SGD). *Under the condition of Theorem 1, the following statement holds true for any* $t\in[T]$*:*

$$\mathbb{E}\left\|\boldsymbol{w}^{r,t+1}-\boldsymbol{w}^*\right\|^2\leq(1-\mu\gamma)\mathbb{E}\left\|\boldsymbol{w}^{r,t}-\boldsymbol{w}^*\right\|^2-(2\gamma-4\gamma^2L)\mathbb{E}\left(F(\boldsymbol{w}^*)-F(\boldsymbol{w}^{r,t})\right)$$

$$+(\gamma L+2\gamma^2L^2)\frac{1}{N}\sum_{j=1}^N\left\|\boldsymbol{w}_i^{r,t}-\boldsymbol{w}^{r,t}\right\|^2+\gamma^2\frac{\delta^2}{N}.$$

Proof. According to updating rule in Algorithm A1, we have the following identity:

$$\mathbb{E}\left\|\boldsymbol{w}^{r,t+1}-\boldsymbol{w}^*\right\|^2=\mathbb{E}\left\|\boldsymbol{w}^{r,t}-\boldsymbol{w}^*\right\|^2-2\gamma\mathbb{E}\left\langle\frac{1}{N}\sum_{j=1}^N\nabla f_j(\boldsymbol{w}_i^{r,t};z_i^{r,t}),\boldsymbol{w}^{r,t}-\boldsymbol{w}^*\right\rangle+\gamma^2\mathbb{E}\left\|\frac{1}{N}\sum_{j=1}^N\nabla f_j(\boldsymbol{w}_i^{r,t})\right\|^2 \tag{9}$$

$$=\mathbb{E}\left\|\boldsymbol{w}^t-\boldsymbol{w}^*\right\|^2\underbrace{-2\gamma\left\langle\frac{1}{N}\sum_{j=1}^N\nabla f_j(\boldsymbol{w}_i^{r,t}),\boldsymbol{w}^{r,t}-\boldsymbol{w}^*\right\rangle}_{T_1}\underbrace{+\gamma^2\mathbb{E}\left\|\frac{1}{N}\sum_{j=1}^N\nabla f_j(\boldsymbol{w}_i^{r,t};z_i^{r,t})\right\|^2}_{T_2}+\gamma^2\frac{\delta^2}{N}. \tag{10}$$

For T_1, since each f_j is L smooth and μ strongly convex, we have:

$$-2\gamma\left\langle\frac{1}{N}\sum_{j=1}^N\nabla f_j(\boldsymbol{w}_i^{r,t}),\boldsymbol{w}^t-\boldsymbol{w}^*\right\rangle=-2\gamma\left\langle\frac{1}{N}\sum_{j=1}^N\nabla f_j(\boldsymbol{w}_i^t),\boldsymbol{w}^t-\boldsymbol{w}_j^t+\boldsymbol{w}_j^{r,t}-\boldsymbol{w}^*\right\rangle$$

$$\leq-2\gamma\left\langle\frac{1}{N}\sum_{j=1}^N\nabla f_j(\boldsymbol{w}_i^t),\boldsymbol{w}^{r,t}-\boldsymbol{w}_j^{r,t}+\boldsymbol{w}_j^{r,t}-\boldsymbol{w}^*\right\rangle$$

$$\leq 2\gamma\frac{1}{N}\sum_{j=1}^N\left(f_j(\boldsymbol{w}^*)-f_j(\boldsymbol{w}^{r,t})-\frac{\mu}{2}\left\|\boldsymbol{w}_i^{r,t}-\boldsymbol{w}^*\right\|^2+\frac{L}{2}\left\|\boldsymbol{w}_i^{r,t}-\boldsymbol{w}^{r,t}\right\|^2\right).$$

Due to Jensen's inequality we know: $-\frac{1}{N}\sum_{j=1}^{N}\frac{\mu}{2}\left\|\boldsymbol{w}_i^t-\boldsymbol{w}^*\right\|^2\leq-\frac{\mu}{2}\left\|\boldsymbol{w}^t-\boldsymbol{w}^*\right\|^2$. Hence we know:

$$-2\gamma\left\langle\frac{1}{N}\sum_{j=1}^{N}\nabla f_j(\boldsymbol{w}_i^{r,t}),\boldsymbol{w}^{r,t}-\boldsymbol{w}^*\right\rangle\leq 2\gamma\left(F(\boldsymbol{w}^*)-F(\boldsymbol{w}^{r,t})-\frac{\mu}{2}\left\|\boldsymbol{w}^{r,t}-\boldsymbol{w}^*\right\|^2+\frac{L}{2}\frac{1}{N}\sum_{j=1}^{N}\left\|\boldsymbol{w}_i^{r,t}-\boldsymbol{w}^{r,t}\right\|^2\right).$$

For T_2, we have:

$$\begin{aligned}\mathbb{E}\left\|\frac{1}{N}\sum_{j=1}^{N}\nabla f_j(\boldsymbol{w}_i^{r,t})\right\|^2&=2\mathbb{E}\left\|\frac{1}{N}\sum_{j=1}^{N}\nabla f_j(\boldsymbol{w}_i^{r,t})-\nabla F(\boldsymbol{w}^{r,t})\right\|^2+2\mathbb{E}\left\|\nabla F(\boldsymbol{w}^{r,t})\right\|^2\\&\leq 2L^2\frac{1}{N}\sum_{j=1}^{N}\mathbb{E}\left\|\boldsymbol{w}_i^{r,t}-\boldsymbol{w}^{r,t}\right\|^2+4L\left(F(\boldsymbol{w}^{r,t})-F(\boldsymbol{w}^*)\right).\end{aligned}$$

Now, plugging T_1 and T_2 back to (10) yields:

$$\begin{aligned}\mathbb{E}\left\|\boldsymbol{w}^{r,t+1}-\boldsymbol{w}^*\right\|^2\leq(1-\mu\gamma)\mathbb{E}\left\|\boldsymbol{w}^{r,t}-\boldsymbol{w}^*\right\|^2-(2\gamma-4\gamma^2L)\mathbb{E}\left(F(\boldsymbol{w}^*)-F(\boldsymbol{w}^{r,t})\right)\\+(\gamma L+2\gamma^2L^2)\frac{1}{N}\sum_{j=1}^{N}\left\|\boldsymbol{w}_i^{r,t}-\boldsymbol{w}^{r,t}\right\|^2+\gamma^2\frac{\delta^2}{N}.\end{aligned}$$

□

Lemma 5. *[25, Lemma 8] For the iterates $\{\boldsymbol{w}_i^{r,t}\}$ generated in Algorithm A1, the following statement holds true:*

$$\frac{1}{N}\sum_{j=1}^{N}\left\|\boldsymbol{w}_i^{r,t}-\boldsymbol{w}^{r,t}\right\|^2\leq 3K\gamma^2\delta^2+6K^2\gamma^2\zeta^2.$$

Lemma 6 (Last iterate convergence of Local SGD). *Under the conditions of Theorem 1, the following statement holds true for the iterates in Algorithm A1:*

$$\mathbb{E}\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\leq(1-\mu\gamma)^{RK}\mathbb{E}\left\|\boldsymbol{w}^0-\boldsymbol{w}^*\right\|^2+\frac{1}{\mu\gamma}(\gamma L+2\gamma^2L^2)\left(3K\gamma^2\delta^2+6K^2\gamma^2\zeta^2\right)+\frac{\gamma\delta^2}{\mu N}$$

Proof. We first unroll the recursion in Lemma 4 from $t=K$ to 0, within one communication round:

$$\begin{aligned}\mathbb{E}\left\|\boldsymbol{w}^{r,K}-\boldsymbol{w}^*\right\|^2&=(1-\mu\gamma)^K\mathbb{E}\left\|\boldsymbol{w}^{r,0}-\boldsymbol{w}^*\right\|^2-\sum_{t=0}^{K-1}(1-\mu\gamma)^{K-t}(2\gamma-4\gamma^2L)\mathbb{E}\left(F(\boldsymbol{w}^*)-F(\boldsymbol{w}^{r,t})\right)\\&\quad+\sum_{t=0}^{K-1}(1-\mu\gamma)^{K-t}(\gamma L+2\gamma^2L^2)\frac{1}{N}\sum_{j=1}^{N}\left\|\boldsymbol{w}_i^{r,t}-\boldsymbol{w}^{r,t}\right\|^2+\sum_{t=0}^{K-1}(1-\mu\gamma)^{K-t}\gamma^2\frac{\delta^2}{N}\end{aligned}$$

Since we choose $\gamma\leq\frac{1}{2L}$, we know $\sum_{t=0}^{K-1}(1-\mu\gamma)^{K-t}(2\gamma-4\gamma^2L)\mathbb{E}\left(F(\boldsymbol{w}^*)-F(\boldsymbol{w}^{r,t})\right)\geq 0$. Plugging in Lemma 5 yields:

$$\mathbb{E}\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2=(1-\mu\gamma)^{RK}\mathbb{E}\left\|\boldsymbol{w}^0-\boldsymbol{w}^*\right\|^2+\frac{1}{\mu\gamma}(\gamma L+2\gamma^2L^2)\left(3K\gamma^2\delta^2+6K^2\gamma^2\zeta^2\right)+\frac{\gamma\delta^2}{\mu N},$$

Plugging in $\gamma=\frac{\log(RK)}{\mu RK}$ gives the convergence rate:

$$\mathbb{E}\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\leq\tilde{O}\left(\frac{\mathbb{E}\left\|\boldsymbol{w}^0-\boldsymbol{w}^*\right\|^2}{RK}+\kappa\left(\frac{\delta^2}{\mu^2R^2K}+\frac{\zeta^2}{\mu^2R^2}\right)+\frac{\delta^2}{\mu^2NRK}\right),$$

which concludes the proof. □

Equipped with above results, we are now ready to provide the convergence of main theorem.

Proof of Theorem 1. The proof simply follows from Lemma 3:

$$\begin{aligned}\left\|\hat{\boldsymbol{\alpha}}-\boldsymbol{\alpha}^*\right\|^2&\leq 2\left\|\hat{\boldsymbol{\alpha}}-\boldsymbol{\alpha}_g^*(\boldsymbol{w}^R)\right\|^2+2\left\|\boldsymbol{\alpha}_g^*(\boldsymbol{w}^R)-\boldsymbol{\alpha}_g^*(\boldsymbol{w}^*)\right\|^2\\&\leq 2(1-\mu\eta_{\boldsymbol{\alpha}})^{T_{\boldsymbol{\alpha}}}+8L\kappa_g^2\sum_{j=1}^{N}\left(2\left\|\nabla f_i(\boldsymbol{w}^*)-\nabla f_j(\boldsymbol{w}^*)\right\|^2+4L^2\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\right)\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\\&\leq 2(1-\mu\eta_{\boldsymbol{\alpha}})^{T_{\boldsymbol{\alpha}}}+8L\kappa_g^2\left(2\bar{\zeta}_i(\boldsymbol{w}^*)+4NL^2\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\right)\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\end{aligned}$$

Algorithm A2: Shuffling Local SGD (`One Client`)

Input: Clients $0, ..., N-1$, Number of Local Steps K , Number of Epoch R, Mixing parameter $\hat{\boldsymbol{\alpha}}$
Epoch for $r = 0, ..., R-1$ **do**
  Server generates permutation $\sigma_r : [N] \mapsto [N]$.
  Client sets initial model $\boldsymbol{v}^{r,0} = \boldsymbol{v}^r$.
  for $j = 0, ..., N-1$ **do**
    Server sends $\boldsymbol{v}^{r,j}$ to Client $\sigma_r(j)$.
    $\boldsymbol{v}^{r,j+1} = \texttt{SGD-Update}(\boldsymbol{v}^{r,j}, \eta, \sigma_r(j), K, \hat{\boldsymbol{\alpha}})$.
  Client i does projection: $\boldsymbol{v}^{r+1} = \mathcal{P}_{\mathcal{W}}(\boldsymbol{v}^{r,N})$.
Output: $\hat{\boldsymbol{v}} = \boldsymbol{v}^R$.
SGD-Update$(\boldsymbol{v}, \eta, j, K, \boldsymbol{\alpha})$
  Initialize $\boldsymbol{v}^0 = \boldsymbol{v}$
  for $t = 0, ..., K-1$ **do**
    $\boldsymbol{v}^t = \boldsymbol{v}^{t-1} - \eta\alpha(j)N\nabla f_j(\boldsymbol{v}^{t-1}; \xi^{t-1})$
  Output $\boldsymbol{v}^K$

Plugging in the convergence of $\left\|\boldsymbol{w}^R - \boldsymbol{w}^*\right\|^2$ from Lemma 6, and the stepsize $\eta_{\boldsymbol{\alpha}} = \frac{1}{L_g}$ for $\boldsymbol{\alpha}$ yields:

$$\mathbb{E}\|\boldsymbol{\alpha}_i^R - \boldsymbol{\alpha}_i^*\|^2 \leq \tilde{O}\left(\exp(-\frac{T_{\boldsymbol{\alpha}}}{\kappa_g}) + \kappa_g^2\bar{\zeta}_i(\boldsymbol{w}^*)L^2\left(\frac{D^2}{RK} + \kappa\left(\frac{\delta^2}{\mu^2R^2K} + \frac{\zeta^2}{\mu^2R^2}\right) + \frac{\delta^2}{\mu^2NRK}\right)\right).$$

□

A.3 Proof of Convergence of Shuffling Local SGD

In this section, we are going to prove the convergence of proposed shuffled variant of Local SGD (Theorem 2). The whole proof framework follows the analysis of vanilla shuffling SGD, but notice that there are two differences. First, in vanilla shuffling SGD, in each epoch, algorithm only updates on each component function f_j once, while here we have to take K steps of SGD update on each component function. Second, we are considering a weighted sum objective in contrary to averaged objective in [4], which means we need to rescale the objective when we apply without-replacement concentration inequality. Even though our algorithm solves models for N clients, for the sake of simplicity, throughout the proof we only show the convergence of one client's model. The algorithm from one client point of view is described in Algorithm A2, where we drop the client index for notational convenience.

Proposition 1. *Assume a sequence* $\{\boldsymbol{w}^t\}_{t=1}^K$ *is obtained by*

$$\boldsymbol{w}^t = \boldsymbol{w}^{t-1} - \eta\alpha N\nabla f(\boldsymbol{w}^{t-1}; \xi^{t-1}),\ t = 1, \ldots, K,$$

then we have

$$\boldsymbol{w}^{t+1} = \boldsymbol{w}^0 - \left(\sum_{\tau=0}^{t}\prod_{t'=t}^{\tau+1}(\mathbf{I} - \alpha N\eta\mathbf{H}_{t'})\right)\eta\alpha N\nabla f(\boldsymbol{w}^0) - \sum_{\tau=0}^{t}\prod_{t'=t}^{\tau+1}(\mathbf{I} - \alpha N\eta\mathbf{H}_{t'})\,\eta\alpha N\boldsymbol{\delta}^t, \quad \forall 0 \leq t \leq K-1,$$

where $\boldsymbol{\delta}^t := \nabla f(\boldsymbol{w}^t; \xi^t) - \nabla f(\boldsymbol{w}^t)$*, and by convention, we define* $\prod_{j=a}^{b}\boldsymbol{A}_j = \mathbf{I}$ *if* $a < b$.

Proof. According to updating rule, we have:

$$\begin{aligned}
\boldsymbol{w}^{t+1} - \boldsymbol{w}^0 &= \boldsymbol{w}^t - \boldsymbol{w}^0 - \eta\alpha N\nabla f(\boldsymbol{w}^t; \xi^t) \\
&= \boldsymbol{w}^t - \boldsymbol{w}^0 - \eta\alpha N\nabla f(\boldsymbol{w}^t) - \eta\alpha N\boldsymbol{\delta}^t \\
&= \boldsymbol{w}^t - \boldsymbol{w}^0 - \eta\alpha N\nabla f(\boldsymbol{w}^0) - \eta\alpha N(\nabla f(\boldsymbol{w}^t) - \nabla f(\boldsymbol{w}^0)) - \eta\alpha N\boldsymbol{\delta}^t.
\end{aligned}$$

Since f is L smooth, and according to Mean Value Theorem, there is a matrix $\mathbf{H}_t$ satisfying $\mu\mathbf{I} \preceq \mathbf{H}_t \preceq L\mathbf{I}$, such that $\nabla f(\boldsymbol{w}^t) - \nabla f(\boldsymbol{w}^0) = \mathbf{H}_t(\boldsymbol{w}^t - \boldsymbol{w}^0)$. Hence we have:

$$\boldsymbol{w}^{t+1} - \boldsymbol{w}^0 = (\mathbf{I} - \eta\alpha N\mathbf{H}_t)(\boldsymbol{w}^t - \boldsymbol{w}^0) - \eta\alpha N\nabla f(\boldsymbol{w}^0) - \eta\alpha N\boldsymbol{\delta}^t.$$

Unrolling the recursion from t to 0 will conclude the proof. □

The following lemma establishes the updating rule of models between epochs r and $r+1$. For notational convenience, whenever there is no confusion, we drop the superscript r in σ^r.

Lemma 7 (One epoch updating rule). *Let $\boldsymbol{v}^r$ and $\boldsymbol{v}^{r+1}$ be two iterates generated by Shuffling Local SGD (Algorithm A2), then the following updating rule holds:*

$$\boldsymbol{v}^{r+1} = \boldsymbol{v}^r - \sum_{j=1}^{N} \prod_{j'=N}^{j+1} (\mathbf{I} - \boldsymbol{Q}_{j'} \mathbf{H}_{j'}) (\boldsymbol{Q}_j \nabla f_{\sigma(j)}(\boldsymbol{v}^r) - \boldsymbol{\delta}_j),$$

where

$$\boldsymbol{Q}_j := \left(\sum_{\tau=0}^{K-1} \prod_{t'=K-1}^{\tau+1} (\mathbf{I} - \eta \hat{\alpha}(\sigma(j)) N \mathbf{H}_{t'}) \right) \eta \hat{\alpha}(\sigma(j)) N,$$

$$\boldsymbol{\delta}_j := \sum_{\tau=0}^{K-1} \prod_{t'=t}^{\tau+1} (\mathbf{I} - \hat{\alpha}(\sigma(j)) N \eta \mathbf{H}_{t'}) \, \eta \hat{\alpha}(\sigma(j)) N \boldsymbol{\delta}^t_{\sigma(j)},$$

by convention, we define $\prod_{j=a}^{b} \boldsymbol{A}_j = \mathbf{I}$ if $a < b$.

Proof. According to Proposition 1, we have

$$\begin{aligned} \boldsymbol{v}^{r,j+1} = \boldsymbol{v}^{r,j} &- \left(\sum_{\tau=0}^{K-1} \prod_{t'=t}^{\tau+1} (\mathbf{I} - \hat{\alpha}(\sigma(j)) N \eta \mathbf{H}_{t'}) \right) \eta \hat{\alpha}(\sigma(j)) N \nabla f_{\sigma(j)}(\boldsymbol{v}^{r,j}) \\ &- \sum_{\tau=0}^{K-1} \prod_{t'=t}^{\tau+1} (\mathbf{I} - \hat{\alpha}(\sigma(j)) N \eta \mathbf{H}_{t'}) \, \eta \hat{\alpha}(\sigma(j)) N \boldsymbol{\delta}^t_{\sigma(j)}. \end{aligned}$$

Plugging our definition of $\boldsymbol{Q}_j$ and $\boldsymbol{\delta}_j$ yields:

$$\boldsymbol{v}^{r,j+1} - \boldsymbol{v}^r = \boldsymbol{v}^{r,j} - \boldsymbol{v}^r - \boldsymbol{Q}_j \nabla f_{\sigma(j)}(\boldsymbol{v}^{r,j}) - \boldsymbol{\delta}_j.$$

Following the same reasoning in the proof of Proposition 1 will conclude the proof. □

Lemma 8 (Summation by parts). *Let $\boldsymbol{A}_j$ and $\mathbf{B}_j$ be complex valued matrices. Then the following fact holds:*

$$\sum_{j=1}^{N} \boldsymbol{A}_j \mathbf{B}_j = \boldsymbol{A}_N \sum_{j=1}^{N} \mathbf{B}_j - \sum_{n=1}^{N-1} (\boldsymbol{A}_{n+1} - \boldsymbol{A}_n) \sum_{j=1}^{n} \mathbf{B}_j.$$

Proposition 2 (Spectral bound of polynomial expansion). *Given a collection of matrices $\{\boldsymbol{A}_t\}$ and $\{\mathbf{B}_t\}$, such that $\boldsymbol{A}_t \preceq L\mathbf{I}$ and $\mathbf{B}_t \preceq L\mathbf{I}$, the following bound hold:*

$$\left\| \prod_{t=l}^{h} (\mathbf{I} - a\boldsymbol{A}_t) - \mathbf{I} \right\| \leq \sum_{m=1}^{h-l} \left(\frac{e(h-l)}{m} \right)^m (aL)^m,$$

$$\left\| \prod_{t=l}^{h} (\mathbf{I} - a\boldsymbol{A}_t) - \prod_{t=l}^{h} (\mathbf{I} - b\mathbf{B}_t) \right\| \leq \sum_{m=1}^{h-l} \left(\frac{e(h-l)}{m} \right)^m (aL)^m + \sum_{m=1}^{h-l} \left(\frac{e(h-l)}{m} \right)^m (bL)^m.$$

Proof. We start with proving the first statement. Expanding the product yields:

$$\prod_{t=l}^{h} (\mathbf{I} - a\boldsymbol{A}_t) = \mathbf{I} + \sum_{m=1}^{h-l} (-1)^m a^m \sum_{|S|=m, |S| \subseteq \{l,\ldots,h\}} \prod_{m' \in S} \boldsymbol{A}_{m'}.$$

Hence we have:

$$\left\| \prod_{t=l}^{h} (\mathbf{I} - a\boldsymbol{A}_t) - \mathbf{I} \right\| = \left\| \sum_{m=1}^{h-l} (-1)^m a^m \sum_{|S|=m, |S| \subseteq \{l,\ldots,h\}} \prod_{m' \in S} \boldsymbol{A}_{m'} \right\| \leq \sum_{m=1}^{h-l} \binom{h-l}{m} (aL)^m$$

According to the upper bound for binomial coefficients: $\binom{h-l}{m} \leq \left(\frac{e(h-l)}{m} \right)^m$, we have:

$$\sum_{m=1}^{h-l} \binom{h-l}{m} (aL)^m \leq \sum_{m=1}^{h-l} \left(\frac{e(h-l)}{m} \right)^m (aL)^m.$$

Then we switch to the second one. Using the same expanding product yields:

$$\begin{aligned}
&\left\|\prod_{t=l}^{h}(\mathbf{I}-a\boldsymbol{A}_t)-\prod_{t=l}^{h}(\mathbf{I}-b\mathbf{B}_t)\right\| \\
&=\left\|\sum_{m=1}^{h-l}(-1)^m a^m \sum_{|S|=m,|S|\subseteq\{l,\ldots,h\}}\prod_{m'\in S}\boldsymbol{A}_{m'}-\sum_{m=1}^{h-l}(-1)^m b^m \sum_{|S|=m,|S|\subseteq\{l,\ldots,h\}}\prod_{m'\in S}\mathbf{B}_{m'}\right\| \\
&\leq \sum_{m=1}^{h-l}\binom{h-l}{m}(aL)^m+\sum_{m=1}^{h-l}\binom{h-l}{m}(bL)^m \\
&\leq \sum_{m=1}^{h-l}\left(\frac{e(h-l)}{m}\right)^m(aL)^m+\sum_{m=1}^{h-l}\left(\frac{e(h-l)}{m}\right)^m(bL)^m.
\end{aligned}$$

□

The following concentration result is the key to bound variance during shuffling updating. The original result holds for average of gradients, and we will later on generalize it to an arbitrary weighted sum of gradients.

Lemma 9 ([20, Theorem 2]). *Suppose $n \geq 2$. Let $\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_n \in \mathbb{R}^d$ satisfy $\left\|\boldsymbol{g}_j\right\| \leq G$ for all j. Let $\bar{\boldsymbol{g}} = \frac{1}{n}\sum_{j=1}^{n}\boldsymbol{g}_j$. Let $\sigma \in S_n$ be a uniform random permutation of n elements. Then, for $i \leq n$, with probability at least $1-p$, we have*

$$\left\|\frac{1}{i}\sum_{j=1}^{i}\boldsymbol{g}_{\sigma(j)}-\bar{\boldsymbol{g}}\right\| \leq G\sqrt{\frac{8(1-\frac{i-1}{n})\log\frac{2}{p}}{i}}.$$

Lemma 10 (Concentration of partial sum of gradients). *Given a uniformly randomly generated permutation σ, and simplex vector $\boldsymbol{\alpha}$, if we assume each $\sup_{\boldsymbol{v}\in\mathcal{W}}\|\nabla f_j(\boldsymbol{v})\| \leq G$, then the following statement holds true:*

$$\left\|\sum_{j=0}^{n}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \leq G\sqrt{8n\log(1/p)}+\frac{n}{N}\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|.$$

Proof. The proof works by re-writing weighted sum of vectors to average of the these vectors:

$$\begin{aligned}
\left\|\sum_{j=0}^{n}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| &= \frac{1}{N}\left\|\sum_{j=0}^{n}\hat{\alpha}(\sigma(j))N\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \\
&= \frac{1}{N}\left(\left\|\sum_{j=0}^{n}\hat{\alpha}(\sigma(j))N\nabla f_{\sigma(j)}(\boldsymbol{v}^r)-n\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|+n\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|\right) \\
&\leq G\sqrt{8n\log(1/p)}+\frac{n}{N}\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|.
\end{aligned}$$

□

Proposition 3 (Spectral norm bound of $\boldsymbol{Q}$). *Let $\boldsymbol{Q}_j$ be defined in (11). Then the following bound for the spectral norm of $\boldsymbol{Q}_j$ holds true for all $j \in [N]$:*

$$\left\|\boldsymbol{Q}_j\right\| \leq \eta\hat{\alpha}(\sigma(j))NK(1+\eta NL)^K$$

Proof. The proof can be completed by writing down the definitin of $\boldsymbol{Q}_j$ and applying Cauchy-Schwartz inequality:

$$\begin{aligned}
\left\|\boldsymbol{Q}_j\right\| &= \left\|\left(\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}\left(\mathbf{I}-\eta\hat{\alpha}(\sigma(j))N'\mathbf{H}_{t'}\right)\right)\eta\hat{\alpha}(\sigma(j))N\right\| \\
&\leq \eta\hat{\alpha}(\sigma(j))N\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}\left\|\left(\mathbf{I}-\eta\hat{\alpha}(\sigma(j))N'\mathbf{H}_{t'}\right)\right\| \\
&\leq \eta\hat{\alpha}(\sigma(j))N\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}(1+\eta\hat{\alpha}(\sigma(j))N'L) \\
&\leq \eta\hat{\alpha}(\sigma(j))NK(1+\eta NL)^K.
\end{aligned}$$

The last step is due to $\eta NL \leq \frac{1}{K}$. □

The following lemma establishes the bound regarding cumulative update between two epochs, namely, $\boldsymbol{v}^{r+1} - \boldsymbol{v}^r$. In particular, Lemma 11 below shows that: (a) in shuffling Local SGD, our update from $\boldsymbol{v}^r$ to $\boldsymbol{v}^{r+1}$ approximates performing NK times of gradient descent with $\hat{\alpha}(j)N\nabla f_{\sigma(j)}(\boldsymbol{v}^r)$, namely, the bias is controlled, and (b) the update itself is bounded, and can be related to the norm of full gradient.

Lemma 11. *During the dynamic of Algorithm A2, the following statements hold true with probability at least* $1-p$*:*
(a)

$$\left\|\sum_{j=1}^{N} Q_j \nabla f_{\sigma(j)}(\boldsymbol{v}^r) - \eta NK \sum_{j=1}^{N} \hat{\alpha}(j) \nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2 \leq 10\eta^2 N^2 K^2 \left(\frac{e}{4R-e}\right)^2 \left\|\nabla \Phi(\hat{\boldsymbol{\alpha}}, \boldsymbol{v}^r)\right\|^2 + 128\eta^2 N^3 K^2 \left(\frac{e}{4R-e}\right)^2 G^2 \log(1/p).$$

(b) for any N' *such that* $0 \leq N' < N$

$$\left\|\sum_{j=1}^{N'-1} Q_j \nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \leq 3e\eta NK \left(\left\|\nabla \Phi(\hat{\boldsymbol{\alpha}}, \boldsymbol{v}^r)\right\| + G\sqrt{8N\log(1/p)}\right),$$

where

$$Q_j := \left(\sum_{\tau=0}^{K-1} \prod_{t'=K-1}^{\tau+1} \left(\mathbf{I} - \eta\hat{\alpha}(\sigma(j))N'\mathbf{H}_{t'}\right)\right) \eta\hat{\alpha}(\sigma(j))N. \tag{11}$$

Proof. We start with proving statement (a). Let $\boldsymbol{A}_j = \frac{Q_j}{\hat{\alpha}(\sigma(j))}$ and $\mathbf{B}_j = \hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)$, applying the identity of summation by parts yields:

$$\sum_{j=1}^{N} Q_j \nabla f_{\sigma(j)}(\boldsymbol{v}^r) = \frac{Q_{N-1}}{\hat{\alpha}(\sigma(N-1))} \sum_{j=1}^{N} \hat{\alpha}(\sigma(j)) \nabla f_{\sigma(j)}(\boldsymbol{v}^r) - \sum_{n=1}^{N-1} \left(\frac{Q_{n+1}}{\hat{\alpha}(\sigma(n+1))} - \frac{Q_n}{\hat{\alpha}(\sigma(n))}\right) \sum_{j=1}^{n} \hat{\alpha}(\sigma(j)) \nabla f_{\sigma(j)}(\boldsymbol{v}^r)$$

$$\left\|\sum_{j=1}^{N} Q_j \nabla f_{\sigma(j)}(\boldsymbol{v}) - \eta NK \sum_{j=1}^{N} \hat{\alpha}(j) \nabla f_j(\boldsymbol{v})\right\|^2 \leq 2 \underbrace{\left\|\left(\frac{Q_{N-1}}{\hat{\alpha}(\sigma(N-1))} - \eta NK\mathbf{I}\right) \sum_{j=1}^{N} \hat{\alpha}(\sigma(j)) \nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2}_{T_1} + 2 \underbrace{\left\|\sum_{n=1}^{N-1} \left(\frac{Q_{n+1}}{\hat{\alpha}(\sigma(n+1))} - \frac{Q_n}{\hat{\alpha}(\sigma(n))}\right) \sum_{j=1}^{n} \hat{\alpha}(\sigma(j)) \nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2}_{T_2}.$$

According to Proposition 2, we have:

$$\left\|\prod_{t'=K-1}^{\tau+1} (\mathbf{I} - \eta\hat{\alpha}(\sigma(j))N\mathbf{H}_{t'}) - \mathbf{I}\right\| \leq \sum_{m=1}^{K-2-\tau} \left(\frac{e(K-2-\tau)}{m} \eta\hat{\alpha}(\sigma(j))NL\right)^m.$$

Since we choose $\eta \leq \frac{1}{4NKRL}$, we have:

$$\left\|\prod_{t'=K-1}^{\tau+1} (\mathbf{I} - \eta\hat{\alpha}(\sigma(j))N\mathbf{H}_{t'}) - \mathbf{I}\right\| \leq \sum_{m=1}^{K-2-\tau} \left(\frac{e}{4Rm}\right)^m \leq \frac{e}{4R-e}, \tag{12}$$

where we use the fact that $\sum_{m=1}^{K-2-\tau}\left(\frac{e}{4Rm}\right)^m \le \sum_{m=1}^{K-2-\tau}\left(\frac{e}{4R}\right)^m \le \frac{e}{4R}\frac{1}{1-e/4R}$. Hence we know:

$$\begin{aligned}
T_1 &\le \left\|\left(\frac{Q_{N-1}}{\hat{\alpha}(\sigma(N-1))} - \eta NK\mathbf{I}\right)\right\|^2 \left\|\sum_{j=1}^{N}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2 \\
&\le \left\|\left(\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}(\mathbf{I}-\eta\hat{\alpha}(\sigma(N-1))N\mathbf{H}_{t'})\right)\eta N - \eta NK\mathbf{I}\right\|^2 \left\|\sum_{j=1}^{N}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2 \\
&\le \eta^2N^2K\sum_{\tau=0}^{K-1}\left\|\prod_{t'=K-1}^{\tau+1}(\mathbf{I}-\eta\hat{\alpha}(\sigma(N-1))N\mathbf{H}_{t'}) - \mathbf{I}\right\|^2 \left\|\sum_{j=1}^{N}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2 \\
&\le \eta^2N^2K^2\left(\frac{e}{4R-e}\right)^2\left\|\sum_{j=1}^{N}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2.
\end{aligned}$$

Thus we have:

$$T_1 \le \eta^2N^2K^2\left(\frac{e}{4R-e}\right)^2\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|^2.$$

For T_2, we first examine the bound of $\frac{Q_{n+1}}{\hat{\alpha}(\sigma(n+1))} - \frac{Q_n}{\hat{\alpha}(\sigma(n))}$:

$$\begin{aligned}
\left\|\frac{Q_{n+1}}{\hat{\alpha}(\sigma(n+1))} - \frac{Q_n}{\hat{\alpha}(\sigma(n))}\right\| &= \left\|\left(\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}(\mathbf{I}-\eta\hat{\alpha}(\sigma(n+1))N\mathbf{H}_{t'})\right)\eta N - \left(\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}(\mathbf{I}-\eta\hat{\alpha}(\sigma(n))N\mathbf{H}_{t'})\right)\eta N\right\| \\
&= \eta N\left\|\left(\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}(\mathbf{I}-\eta\hat{\alpha}(\sigma(n+1))N\mathbf{H}_{t'})\right) - \left(\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}(\mathbf{I}-\eta\hat{\alpha}(\sigma(n))N\mathbf{H}_{t'})\right)\right\| \\
&= \eta N\left\|\left(\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}(\mathbf{I}-\eta\hat{\alpha}(\sigma(n+1))N\mathbf{H}_{t'})\right) - \left(\sum_{\tau=0}^{K-1}\prod_{t'=K-1}^{\tau+1}(\mathbf{I}-\eta\hat{\alpha}(\sigma(n))N\mathbf{H}_{t'})\right)\right\| \\
&\le \eta N\sum_{\tau=0}^{K-1}\left(\sum_{m=1}^{K-2-\tau}\left(\frac{e(K-2-\tau)}{m}\eta\hat{\alpha}(\sigma(n))NL\right)^m + \sum_{m=1}^{K-2-\tau}\left(\frac{e(K-2-\tau)}{m}\eta\hat{\alpha}(\sigma(n+1))NL\right)^m\right).
\end{aligned}$$

where we evoke Proposition 2 at last step. Given that $\eta \le \frac{1}{4NKRL}$ we have:

$$\begin{aligned}
\left\|\frac{Q_{n+1}}{\hat{\alpha}(\sigma(n+1))} - \frac{Q_n}{\hat{\alpha}(\sigma(n))}\right\| &\le \eta N\sum_{\tau=0}^{K-1}\left(\sum_{m=1}^{K-2-\tau}\left(\frac{e}{4Rm}\hat{\alpha}(\sigma(n))\right)^m + \sum_{m=1}^{K-2-\tau}\left(\frac{e}{4Rm}\hat{\alpha}(\sigma(n+1))L\right)^m\right) \\
&\le \eta NK\left(\frac{\hat{\alpha}(\sigma(n))e}{4R-e} + \frac{\hat{\alpha}(\sigma(n+1))e}{4R-e}\right).
\end{aligned}$$

where we use the reasoining in (12). Hence for $\sqrt{T_2}$:

$$\begin{aligned}
\sqrt{T_2} &\le \eta NK\sum_{n=1}^{N-1}\left(\frac{\hat{\alpha}(\sigma(n))e}{4R-e} + \frac{\hat{\alpha}(\sigma(n+1))e}{4R-e}\right)\left\|\sum_{j=1}^{n}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \\
&\le \eta NK\frac{e}{4R-e}\sum_{n=1}^{N-1}\left(\hat{\alpha}(\sigma(n)) + \hat{\alpha}(\sigma(n+1))\right)\left(G\sqrt{8n\log(1/p)} + \frac{n}{N}\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|\right) \\
&\le \eta NK\frac{2e}{4R-e}\left(G\sqrt{8N\log(1/p)} + \left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|\right).
\end{aligned}$$

where at last step we evoke Lemma 10. So we can conclude $T_2 \le 2\eta^2N^2K^2\left(\frac{2e}{4R-e}\right)^2\left(G^2 8N\log(1/p) + \|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\|^2\right)$. Putting the bounds of T_1 and T_2 together will conclude the proof for (a).

Now we switch to proving (b). Once again by the summation of parts identity we have:

$$\sum_{j=1}^{N'}Q_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r) = \frac{Q_{N'}}{\hat{\alpha}(\sigma(N'))}\sum_{j=1}^{N'}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r) - \sum_{n=1}^{N'-1}\left(\frac{Q_{n+1}}{\hat{\alpha}(\sigma(n+1))} - \frac{Q_n}{\hat{\alpha}(\sigma(n))}\right)\sum_{j=1}^{n}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r).$$

Taking the norm of both side yields:

$$\left\|\sum_{j=1}^{N'} Q_j \nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| = \underbrace{\left\|\frac{Q_{N'}}{\hat{\alpha}(\sigma(N'))}\sum_{j=1}^{N'}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|}_{B} + \underbrace{\left\|\sum_{n=1}^{N'-1}\left(\frac{Q_{n+1}}{\hat{\alpha}(\sigma(n+1))}-\frac{Q_n}{\hat{\alpha}(\sigma(n))}\right)\sum_{j=1}^{n}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|}_{C}.$$

Plugging our developed bound for $\left\|Q_{N'}\right\|$ and $\left\|\sum_{n=1}^{N'-1}\left(\frac{Q_{n+1}}{\hat{\alpha}(\sigma(n+1))}-\frac{Q_n}{\hat{\alpha}(\sigma(n))}\right)\right\|$ yields:

$$\begin{aligned} B &\le \left\|\frac{Q_{N'}}{\hat{\alpha}(\sigma(N'))}\right\|\left\|\sum_{j=1}^{N'}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \\ &\le \eta NK(1+\eta NL)^K\left(G\sqrt{8N'\log(1/p)}+\frac{N'}{N}\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|\right). \end{aligned}$$

where at last step we evoke Lemma 10. And for C, we use the similar reasoning:

$$\begin{aligned} C &\le \sum_{n=1}^{N'-1}\left\|\left(\frac{Q_{n+1}}{\hat{\alpha}(\sigma(n+1))}-\frac{Q_n}{\hat{\alpha}(\sigma(n))}\right)\right\|\left\|\sum_{j=1}^{n}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \\ &\le \sum_{n=1}^{N'-1}\eta NK\frac{e}{4R-e}\left(\hat{\alpha}(\sigma(n+1))+\hat{\alpha}(\sigma(n))\right)\left(G\sqrt{8n\log(1/p)}+\frac{n}{N}\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|\right) \\ &\le 2\eta NK\frac{e}{4R-e}\left(G\sqrt{8N\log(1/p)}+\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|\right). \end{aligned}$$

Putting these pieces together yields:

$$\left\|\sum_{j=1}^{N'} Q_j \nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \le 3e\eta NK\left(\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|+G\sqrt{8N\log(1/p)}\right).$$

□

Lemma 12. *During the dynamic of Algorithm A2, the following statements hold true with probability at least* $1-p$*:*

$$\left\|\sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I}-Q_{j'}\mathbf{H}_{j'})\right)Q_{n+1}\mathbf{H}_{n+1}\sum_{j=1}^{n}Q_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2 \le 18e^6\eta^4N^4K^4L^4\left(\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|^2+8G^2N\log(1/p)\right)$$

Proof. We first apply Cauchy-Schwartz inequality:

$$\begin{aligned} &\left\|\sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I}-Q_{j'}\mathbf{H}_{j'})\right)Q_{n+1}\mathbf{H}_{n+1}\sum_{j=1}^{n}Q_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \\ &\le \sum_{n=1}^{N-1}\left\|\left(\prod_{j'=N}^{n+2}(\mathbf{I}-Q_{j'}\mathbf{H}_{j'})\right)\right\|\left\|Q_{n+1}\mathbf{H}_{n+1}\right\|\left\|\sum_{j=1}^{n}Q_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \\ &\le \left(1+\eta NK+\eta^2N^2KL\right)^{2N}\eta NLK(1+\eta NL)^KL\sum_{n=1}^{N-1}\hat{\alpha}(\sigma(n+1))\left\|\sum_{j=1}^{n}Q_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\| \\ &\le e^2\eta NKL^2\sum_{n=1}^{N-1}\hat{\alpha}(\sigma(n+1))\left\|\sum_{j=1}^{n}Q_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|. \end{aligned}$$

We proceed by applying the bound from Lemma 11 (b):

$$\left\|\sum_{j=1}^{n} Q_j \nabla f_{\sigma(j)}(v^r)\right\| \le 3e\eta NK\left(\left\|\nabla\Phi(\hat{\alpha}, v^r)\right\| + G\sqrt{8N\log(1/p)}\right).$$

Therefore, it follows that:

$$\begin{aligned}
&\left\|\sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I}-Q_{j'}\mathbf{H}_{j'})\right)Q_{n+1}\mathbf{H}_{n+1}\sum_{j=1}^{n}Q_j\nabla f_{\sigma(j)}(v^r)\right\| \\
&\le e^2\eta NKL^2\sum_{n=1}^{N-1}\hat{\alpha}(\sigma(n+1))\cdot 3e\eta NK\left(\left\|\nabla\Phi(\hat{\alpha}, v^r)\right\| + G\sqrt{8N\log(1/p)}\right) \\
&\le 3e^3\eta^2N^2K^2L^2\left(\left\|\nabla\Phi(\hat{\alpha}, v^r)\right\| + G\sqrt{8N\log(1/p)}\right)
\end{aligned}$$

□

Lemma 13 (Noise bound). *During the dynamic of Algorithm A2, the following statement for gradient noises holds true with probability at least* $1-p$*:*

$$\mathbb{E}\left\|\sum_{j=1}^{N}\prod_{j'=N}^{j+1}(\mathbf{I}-Q_{j'}\mathbf{H}_{j'})\boldsymbol{\delta}_j\right\| \le \eta NKe^2\delta,$$

where

$$\boldsymbol{\delta}_j := \sum_{\tau=0}^{K-1}\prod_{t'=t}^{\tau+1}(\mathbf{I}-\hat{\alpha}(\sigma(j))N\eta\mathbf{H}_{t'})\,\eta\hat{\alpha}(\sigma(j))N\boldsymbol{\delta}^t_{\sigma(j)}.$$

Proof. According to triangle and Cauchy-Schwartz inequalities we have:

$$\begin{aligned}
\left\|\sum_{j=1}^{N}\prod_{j'=N}^{j+1}(\mathbf{I}-Q_{j'}\mathbf{H}_{j'})\boldsymbol{\delta}_j\right\| &\le \sum_{j=1}^{N}\prod_{j'=N}^{j+1}\left\|(\mathbf{I}-Q_{j'}\mathbf{H}_{j'})\right\|\left\|\boldsymbol{\delta}_j\right\| \\
&\le \sum_{j=1}^{N}\left(1+(\eta\hat{\alpha}(\sigma(j))NK(1+\eta NL)^K)L\right)^N\left\|\boldsymbol{\delta}_j\right\| \\
&\le \sum_{j=1}^{N}\underbrace{\left(1+(\eta\hat{\alpha}(\sigma(j))NK(1+\eta NL)^K)L\right)^N}_{\le e}\cdot\eta\hat{\alpha}(\sigma(j))NK\underbrace{(1+\eta NL)^K}_{\le e}\delta \\
&\le \eta NKe^2\delta.
\end{aligned}$$

□

A.4 Proof of Theorem 2

Proof. For notational convenience, let us define

$$\begin{aligned}
g^r &:= \sum_{j=1}^{N}\prod_{j'=N}^{j+1}(\mathbf{I}-Q_{j'}\mathbf{H}_{j'})Q_j\nabla f_{\sigma(j)}(v^r), \\
\boldsymbol{\delta}^r &:= \sum_{j=1}^{N}\prod_{j'=N}^{j+1}(\mathbf{I}-Q_{j'}\mathbf{H}_{j'})\boldsymbol{\delta}_j.
\end{aligned}$$

Then we recall the updating rule of v (Lemma 7):

$$v^{r+1} = \mathcal{P}_{\mathcal{W}}\left(v^r - g^r - \boldsymbol{\delta}^r\right)$$

Hence we have:

$$\begin{aligned}\mathbb{E}\left\|\boldsymbol{v}^{r+1}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2 &= \mathbb{E}\left\|\mathcal{P}_{\mathcal{W}}\left(\boldsymbol{v}^r-\boldsymbol{g}^r-\boldsymbol{\delta}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right)\right\|^2\\ &\le \mathbb{E}\left\|\boldsymbol{v}^r-\boldsymbol{g}^r-\boldsymbol{\delta}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2\\ &\le \mathbb{E}\left\|\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2-2\mathbb{E}\langle\boldsymbol{g}^r,\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\rangle+\mathbb{E}\left\|\boldsymbol{g}^r\right\|^2+\mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2\\ &\le \mathbb{E}\left\|\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2-2\mathbb{E}\langle\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r),\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\rangle-2\mathbb{E}\langle\boldsymbol{g}^r-\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r),\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\rangle\\ &\quad+\mathbb{E}\left\|\boldsymbol{g}^r\right\|^2+\mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2.\end{aligned}$$

Now, applying strongly convexity of $\Phi(\hat{\boldsymbol{\alpha}},\cdot)$ and Cauchy-Schwartz inequality yields:

$$\begin{aligned}\mathbb{E}\left\|\boldsymbol{v}^{r+1}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2 &\le (1-\mu\eta NK)\mathbb{E}\left\|\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2-\eta NK\mathbb{E}[\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)-\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))]\\ &\quad+\frac{1}{2}\left(\frac{1}{\mu\eta NK}\mathbb{E}\|\boldsymbol{g}^r-\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\|^2+\mu\eta NK\mathbb{E}\|\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\|^2\right)\\ &\quad+\mathbb{E}\left\|\boldsymbol{g}^r\right\|^2+\mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2\\ &\le (1-\frac{1}{2}\mu\eta NK)\mathbb{E}\left\|\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2-\eta NK\mathbb{E}[\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)-\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))]\\ &\quad+\frac{1}{2\mu\eta NK}\mathbb{E}\|\boldsymbol{g}^r-\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\|^2\\ &\quad+2\mathbb{E}\left\|\boldsymbol{g}^r-\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|^2+2\mathbb{E}\left\|\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\right\|^2+\mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2.\end{aligned}$$

Since $\Phi(\hat{\boldsymbol{\alpha}},\cdot)$ is L smooth, we have: $\mathbb{E}\,\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\|^2\le 2L\mathbb{E}[\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)-\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))]$. Therefore, we have:

$$\mathbb{E}\left\|\boldsymbol{v}^{r+1}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2\le(1-\frac{1}{2}\mu\eta NK)\mathbb{E}\left\|\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2-(\eta NK-4\eta^2N^2K^2L)\mathbb{E}[\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)-\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))] \tag{13}$$

$$+\left(\frac{1}{2\mu\eta NK}+2\right)\mathbb{E}\|\boldsymbol{g}^r-\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\|^2+\mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2 \tag{14}$$

Now, we examine the term $\|\boldsymbol{g}^r-\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\|^2$. First according to summation by part (Lemma 8) by letting $\boldsymbol{A}_j:=\prod_{j'=N}^{j+1}(\mathbf{I}-\boldsymbol{Q}_{j'}\mathbf{H}_{j'})$ and $\mathbf{B}_j=\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)$, we have:

$$\begin{aligned}\boldsymbol{g}^r&=\sum_{j=1}^{N}\prod_{j'=N}^{j+1}(\mathbf{I}-\boldsymbol{Q}_{j'}\mathbf{H}_{j'})\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\\ &=\sum_{j=1}^{N}\boldsymbol{A}_j\mathbf{B}_j=\sum_{j=1}^{N}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)-\sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I}-\boldsymbol{Q}_{j'}\mathbf{H}_{j'})-\prod_{j'=N}^{n+1}(\mathbf{I}-\boldsymbol{Q}_{j'}\mathbf{H}_{j'})\right)\sum_{j=1}^{n}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\\ &=\sum_{j=1}^{N}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)-\sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I}-\boldsymbol{Q}_{j'}\mathbf{H}_{j'})\right)\boldsymbol{Q}_{n+1}\mathbf{H}_{n+1}\sum_{j=1}^{n}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r).\end{aligned}$$

Hence we have:

$$\begin{aligned}&\|\boldsymbol{g}^r-\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\|^2\\ &=\left\|\eta NK\sum_{j=1}^{N}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)-\left(\sum_{j=1}^{N}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)-\sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I}-\boldsymbol{Q}_{j'}\mathbf{H}_{j'})\right)\boldsymbol{Q}_{n+1}\mathbf{H}_{n+1}\sum_{j=1}^{n}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right)\right\|^2\\ &\overset{(1)}{=}2\left\|\left(\eta NK\sum_{j=1}^{N}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r)-\sum_{j=1}^{N}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right)\right\|^2+2\left\|\sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I}-\boldsymbol{Q}_{j'}\mathbf{H}_{j'})\right)\boldsymbol{Q}_{n+1}\mathbf{H}_{n+1}\sum_{j=1}^{n}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2\\ &\overset{(2)}{\le}\left(20\eta^2N^2K^2\left(\frac{e}{4R-e}\right)^2+36e^6\eta^4N^4K^4L^4\right)\|\nabla\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)\|^2+256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2G^2\log(1/p)\\ &\quad+244e^6\eta^4N^4K^4L^4G^2N\log(1/p)\\ &\overset{(3)}{\le}\left(20\eta^2N^2K^2\left(\frac{e}{4R-e}\right)^2+36e^6\eta^4N^4K^4L^4\right)2L\left(\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)-\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))\right)\\ &\quad+\left(244e^6\eta^4N^4K^4L^4+256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p),\end{aligned}$$

where in (1) we apply Jensen's inequality, in (2) we plug in Lemma 11 (a), and Lemma 12, and in (3) we use the L-smoothness of Φ. Plugging above bound back in (19) yields:

$$\begin{aligned}
&\mathbb{E}\left\|\boldsymbol{v}^{r+1}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2 \\
&\leq (1-\frac{1}{2}\mu\eta NK)\mathbb{E}\left\|\boldsymbol{v}^{r}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2+\eta^2N^2K^2e^4\delta^2 \\
&\underbrace{-\left(\eta NK-4\eta^2N^2K^2L-\left(\frac{1}{2\mu\eta NK}+2\right)\left(20\eta^2N^2K^2\left(\frac{e}{4R-e}\right)^2-36e^6\eta^4N^4K^4L^4\right)\right)}_{T_1}\mathbb{E}[\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)-\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))] \\
&+\left(\frac{1}{2\mu\eta NK}+2\right)\left(244e^6\eta^4N^4K^4L^4+256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p).
\end{aligned}$$

Since we choose $\eta=\frac{4\log(\sqrt{NK}R)}{\mu NKR}$, and large enough epoch number:

$$R\geq\max\left\{\left(\frac{40}{\mu}+1\right)e,16\log(\sqrt{NK}R),64\kappa\log(\sqrt{NK}R)\right\},$$

we know that $T_1\leq 0$. We thus have:

$$\begin{aligned}
&\mathbb{E}\left\|\boldsymbol{v}^{r+1}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2 \\
&\leq (1-\frac{1}{2}\mu\eta NK)\mathbb{E}\left\|\boldsymbol{v}^{r}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2+\eta^2N^2K^2e^4\delta^2+\left(\frac{1}{2\mu\eta NK}+2\right)\left(244e^6\eta^4N^4K^4L^4+256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p)
\end{aligned}$$

Unrolling the recursion from $r=R$ to 0:

$$\begin{aligned}
&\mathbb{E}\left\|\boldsymbol{v}^{R}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2 \\
&\leq (1-\frac{1}{2}\mu\eta NK)^R\mathbb{E}\left\|\boldsymbol{v}^{0}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2+\frac{2}{\mu}\eta NKe^4\delta^2 \\
&\quad+\frac{1}{\mu}\left(\frac{1}{2\mu\eta NK}+2\right)\left(488e^6\eta^3N^3K^3L^4+512\eta N^2K\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p)
\end{aligned}$$

Plugging in our choice of η will conclude the proof:

$$\mathbb{E}\left\|\boldsymbol{v}^{R}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2\leq\tilde{O}\left(\frac{\mathbb{E}\left\|\boldsymbol{v}^{0}-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2}{NKR^2}+\frac{\delta^2}{\mu^2R}+\left(\frac{L^4+N}{\mu^4R^2}\right)G^2N\log(1/p)\right)$$

Finally, according to Lemma 2 we can complete the proof:

$$\begin{aligned}
\Phi(\boldsymbol{\alpha}_i^*,\hat{\boldsymbol{v}}_i)-\Phi(\boldsymbol{\alpha}_i^*,\boldsymbol{v}_i^*)&\leq L\left\|\hat{\boldsymbol{v}}_i-\hat{\boldsymbol{v}}_i^*\right\|^2+\kappa^2L\|\hat{\boldsymbol{\alpha}}_i-\boldsymbol{\alpha}^*\|^2 \\
&\leq\tilde{O}\left(\frac{LD^2}{NKR^2}+\frac{L\delta^2}{\mu^2R}+\left(\frac{L^4+N}{\mu^4R^2}\right)LG^2N\log(1/p)\right) \\
&\quad+\tilde{O}\left(\exp\left(-\frac{T_{\boldsymbol{\alpha}}}{\kappa_g}\right)+\kappa_g^2\bar{\zeta}_i(\boldsymbol{w}^*)L^2\left(\frac{D^2}{RK}+\frac{\kappa\zeta^2}{\mu^2R^2}+\frac{\delta^2}{\mu^2NRK}\right)\right),
\end{aligned}$$

where we plug in the convergence result from Theorem 1 at last step. □

B PROOF OF CONVERGENCE OF SINGLE LOOP ALGORITHM

In this section, we turn to presenting the proof of single loop PERM algorithm (Algorithm 2) where the learning of mixing parameters and personalized models are coupled. Compared to Algorithm A2, here during the optimization of model, the mixing parameters are also being updated. As a result, we need to decouple the two updates which makes the analysis more involved. We begin with some technical lemmas that support the proof of main result.

B.1 Technical Lemmas

Proposition 4 (Basic Properties of SGD on Smooth Strongly Convex Function). *Let $\boldsymbol{w}^t$ to be the tth iterate of minibatch SGD on smooth and strongly convex function F, with minibatch size M and learning rate γ. Also assume the variance is bounded by δ. Then the following statements hold true after T iterations of SGD:*

$$\mathbb{E}\|\nabla F(\boldsymbol{w}^T)\|^2\leq 2L\left(1-\mu\gamma\right)^T\left(F(\boldsymbol{w}^0)-F(\boldsymbol{w}^*)\right)+\frac{2\gamma\kappa\delta^2}{M} \tag{15}$$

$$\mathbb{E}\|\boldsymbol{w}^{T+1}-\boldsymbol{w}^{T}\|^2 \leq 2\gamma^2 L\left(1-\mu\gamma\right)^{T}\left(F(\boldsymbol{w}^0)-F(\boldsymbol{w}^*)\right)+\frac{2\gamma^3\kappa\delta^2}{M}+\frac{\gamma^2\delta^2}{M} \tag{16}$$

$$\mathbb{E}\|\boldsymbol{w}^{T}-\boldsymbol{w}^{*}\|^2 \leq \frac{2}{\mu}\left(1-\mu\gamma\right)^{T}\left(F(\boldsymbol{w}^0)-F(\boldsymbol{w}^*)\right)+2\gamma\frac{\delta^2}{\mu^2 M}. \tag{17}$$

Lemma 14 (Bounded iterates difference of $\boldsymbol{\alpha}$). *Let $\{\boldsymbol{\alpha}_i^r\}$ be iterates generated by Algorithm 2, then under conditions of Theorem 3, the following statement holds:*

$$\|\boldsymbol{\alpha}_i^r-\boldsymbol{\alpha}_i^{r-1}\|^2 \leq 6\left(1-\frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}}+O\left(\kappa_g^2 L^2\bar{\zeta}_i(\boldsymbol{w}^*)\right)\left(\gamma^2 L\left(1-\mu\gamma\right)^{r}\left(F(\boldsymbol{w}^0)-F(\boldsymbol{w}^*)\right)+\frac{\gamma^3\kappa\delta^2}{M}+\frac{\gamma^2\delta^2}{M}\right)$$

Proof. Define

$$\boldsymbol{z}^r=\left[\left\|\nabla f_i(\boldsymbol{w}^r)-\nabla f_1(\boldsymbol{w}^r)\right\|^2,\ldots,\left\|\nabla f_i(\boldsymbol{w}^r)-\nabla f_N(\boldsymbol{w}^r)\right\|^2\right].$$

According to updating rule of $\boldsymbol{\alpha}$ in Algorithm 2 and Lemma 3 we have:

$$\begin{aligned}
\|\boldsymbol{\alpha}_i^r-\boldsymbol{\alpha}_i^{r-1}\|^2 &\leq 3\|\boldsymbol{\alpha}_i^r-\boldsymbol{\alpha}_{g_i}^*(\boldsymbol{w}^r)\|^2+3\left\|\boldsymbol{\alpha}_{g_i}^*(\boldsymbol{w}^{r-1})-\boldsymbol{\alpha}_{g_i}^*(\boldsymbol{w}^r)\right\|^2+3\|\boldsymbol{\alpha}_{g_i}^*(\boldsymbol{w}^{r-1})-\boldsymbol{\alpha}_i^{r-1}\|^2\\
&\leq 6(1-\mu_g\eta_{\boldsymbol{\alpha}})^{T_{\boldsymbol{\alpha}}}+3\left\|\boldsymbol{\alpha}_{g_i}^*(\boldsymbol{w}^{r-1})-\boldsymbol{\alpha}_{g_i}^*(\boldsymbol{w}^r)\right\|^2\\
&\leq 6(1-\mu_g\eta_{\boldsymbol{\alpha}})^{T_{\boldsymbol{\alpha}}}+3\kappa_g^2\sum_{j=1}^{N}\left\|z_j^{r-1}-z_j^r\right\|^2\\
&\leq 6(1-\mu_g\eta_{\boldsymbol{\alpha}})^{T_{\boldsymbol{\alpha}}}+3\kappa_g^2\sum_{j=1}^{N}\left\|\nabla f_i(\boldsymbol{w}^r)-\nabla f_j(\boldsymbol{w}^r)+\nabla f_i(\boldsymbol{w}^{r-1})-\nabla f_j(\boldsymbol{w}^{r-1})\right\|^2 4L^2\left\|\boldsymbol{w}^r-\boldsymbol{w}^{r-1}\right\|^2
\end{aligned}$$

where the second inequality follows from (8). Since $\left\|\nabla f_i(\boldsymbol{w}^r)-\nabla f_j(\boldsymbol{w}^r)\right\| \leq \left\|\nabla f_i(\boldsymbol{w}^*)-\nabla f_j(\boldsymbol{w}^*)\right\|+2L\left\|\boldsymbol{w}^r-\boldsymbol{w}^*\right\|$, we can conclude that

$$\begin{aligned}
\|\boldsymbol{\alpha}_i^r-\boldsymbol{\alpha}_i^{r-1}\|^2 &\leq 6(1-\mu_g\eta_{\boldsymbol{\alpha}})^{T_{\boldsymbol{\alpha}}}\\
&\quad+12L^2\kappa_g^2\sum_{j=1}^{N}\left(8\left\|\nabla f_i(\boldsymbol{w}^*)-\nabla f_j(\boldsymbol{w}^*)\right\|^2+8L^2\left\|\boldsymbol{w}^r-\boldsymbol{w}^*\right\|^2+8L^2\left\|\boldsymbol{w}^{r-1}-\boldsymbol{w}^*\right\|^2\right)\left\|\boldsymbol{w}^r-\boldsymbol{w}^{r-1}\right\|^2\\
&\leq 6\left(1-\frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}}+O\left(\kappa_g^2L^2\bar{\zeta}_i(\boldsymbol{w}^*)\left\|\boldsymbol{w}^r-\boldsymbol{w}^{r-1}\right\|^2\right)\\
&\leq 6\left(1-\frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}}+O\left(\kappa_g^2L^2\bar{\zeta}_i(\boldsymbol{w}^*)\right)\left(\gamma^2L\left(1-\mu\gamma\right)^{r}\left(F(\boldsymbol{w}^0)-F(\boldsymbol{w}^*)\right)+\frac{\gamma^3\kappa\delta^2}{M}+\frac{\gamma^2\delta^2}{M}\right)
\end{aligned}$$

where at last step we plug in Proposition 4 (16). □

Lemma 15 (Convergence of $\boldsymbol{\alpha}$). *Let $\{\hat{\boldsymbol{\alpha}}_i\}_{i=1}^N$ be the mixing parameters generated by Algorithm 2. Then under the conditions of Theorem 3, the following statement holds:*

$$\left\|\hat{\boldsymbol{\alpha}}_i-\boldsymbol{\alpha}^*\right\|^2 \leq 2(1-\frac{1}{\kappa_g})^{T_{\boldsymbol{\alpha}}}+O\left(\kappa_g^2\bar{\zeta}_i(\boldsymbol{w}^*)L^2\frac{2}{\mu}\left(1-\mu\gamma\right)^{T}+2\gamma\frac{\delta^2}{\mu^2M}\right),\ i\in[N]$$

Proof. We notice the following decomposition:

$$\begin{aligned}
\left\|\hat{\boldsymbol{\alpha}}_i-\boldsymbol{\alpha}^*\right\|^2 &= \left\|\boldsymbol{\alpha}_i^R-\boldsymbol{\alpha}_g^*(\boldsymbol{w}^*)\right\|^2\\
&\leq 2\left\|\boldsymbol{\alpha}_i^R-\boldsymbol{\alpha}_g^*(\boldsymbol{w}^R)\right\|^2+2\left\|\boldsymbol{\alpha}_{g_i}^*(\boldsymbol{w}^R)-\boldsymbol{\alpha}_{g_i}^*(\boldsymbol{w}^*)\right\|^2\\
&\leq 2(1-\frac{1}{\kappa_g})^{T_{\boldsymbol{\alpha}}}+O\left(\kappa_g^2\left(\bar{\zeta}_i(\boldsymbol{w}^*)+NL^2\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\right)4L\left\|\boldsymbol{w}^R-\boldsymbol{w}^*\right\|^2\right)\\
&\leq 2(1-\frac{1}{\kappa_g})^{T_{\boldsymbol{\alpha}}}+O\left(\kappa_g^2\bar{\zeta}_i(\boldsymbol{w}^*)L^2\frac{2}{\mu}\left(1-\mu\gamma\right)^{T}+2\gamma\frac{\delta^2}{\mu^2M}\right),
\end{aligned}$$

where in the second inequality we apply Lemma 3, and in the third inequality we use Proposition 4 (17). □

B.2 Proof of Theorem 3

Proof. According to Lemma 2, we have:

$$\Phi(\boldsymbol{\alpha}_i^*, \hat{\boldsymbol{v}}_i) - \Phi(\boldsymbol{\alpha}_i^*, \boldsymbol{v}_i^*) \leq L \left\|\hat{\boldsymbol{v}}_i - \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}_i)\right\|^2 + \kappa^2 L \left\|\hat{\boldsymbol{\alpha}}_i - \boldsymbol{\alpha}_i^*\right\|^2$$

We first examine the convergence of $\|\hat{\boldsymbol{v}}_i - \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}_i)\|^2$. Applying Cauchy-Schwartz inequality yields:

$$\begin{aligned}
\|\boldsymbol{v}^{r+1} - \boldsymbol{v}^*(\boldsymbol{\alpha}^{r+1})\|^2 &\leq \left(1 + \frac{1}{4a-2}\right) \|\boldsymbol{v}^{r+1} - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\|^2 + (1+4a-2) \|\boldsymbol{v}^*(\boldsymbol{\alpha}^{r+1}) - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\|^2 \\
&\leq \left(1 + \frac{1}{4a-2}\right) \|\boldsymbol{v}^{r+1} - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\|^2 + (1+4a-2)\, \kappa^2 \|\boldsymbol{\alpha}^{r+1} - \boldsymbol{\alpha}^r\|^2 \qquad (18)
\end{aligned}$$

where $a = \frac{1}{\mu\eta NK}$. Similar to the proof of Theorem 2, we first define

$$\begin{aligned}
\boldsymbol{g}^r &:= \sum_{j=1}^{N} \prod_{j'=N-1}^{j+1} (\mathbf{I} - Q_{j'} \mathbf{H}_{j'}) Q_j \nabla f_{\sigma(j)}(\boldsymbol{v}^r), \\
\boldsymbol{\delta}^r &:= \sum_{j=1}^{N} \prod_{j'=N-1}^{j+1} (\mathbf{I} - Q_{j'} \mathbf{H}_{j'}) \boldsymbol{\delta}_j.
\end{aligned}$$

Then we recall the updating rule of $\boldsymbol{v}$:

$$\boldsymbol{v}^{r+1} = \mathcal{P}_{\mathcal{W}} \left(\boldsymbol{v}^r - \boldsymbol{g}^r - \boldsymbol{\delta}^r\right)$$

Hence we have:

$$\begin{aligned}
\mathbb{E}\left\|\boldsymbol{v}^{r+1} - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 &= \mathbb{E}\left\|\mathcal{P}_{\mathcal{W}}\left(\boldsymbol{v}^r - \boldsymbol{g}^r - \boldsymbol{\delta}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right)\right\|^2 \\
&\leq \mathbb{E}\left\|\boldsymbol{v}^r - \boldsymbol{g}^r - \boldsymbol{\delta}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 \\
&\leq \mathbb{E}\left\|\boldsymbol{v}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 - 2\mathbb{E}\langle \boldsymbol{g}^r, \boldsymbol{v}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\rangle + \mathbb{E}\left\|\boldsymbol{g}^r\right\|^2 + \mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2 \\
&\leq \mathbb{E}\left\|\boldsymbol{v}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 - 2\mathbb{E}\langle \eta NK \nabla\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r), \boldsymbol{v}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\rangle \\
&\quad - 2\mathbb{E}\langle \boldsymbol{g}^r - \eta NK \nabla\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r), \boldsymbol{v}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\rangle + \mathbb{E}\left\|\boldsymbol{g}^r\right\|^2 + \mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2.
\end{aligned}$$

Now, applying strongly convexity of $\Phi(\boldsymbol{\alpha}^r, \cdot)$ and Cauchy-Schwartz inequality yields:

$$\begin{aligned}
\mathbb{E}\left\|\boldsymbol{v}^{r+1} - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 &\leq (1 - \mu\eta NK)\mathbb{E}\left\|\boldsymbol{v}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 - \eta NK \mathbb{E}[\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r) - \Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))] \\
&\quad + \frac{1}{2}\left(\frac{1}{\mu\eta NK}\mathbb{E}\|\boldsymbol{g}^r - \eta NK \nabla\Phi(\hat{\boldsymbol{\alpha}}, \boldsymbol{v}^r)\|^2 + \mu\eta NK \mathbb{E}\|\boldsymbol{v}^r - \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\|^2\right) \\
&\quad + \mathbb{E}\left\|\boldsymbol{g}^r\right\|^2 + \mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2 \\
&\leq \left(1 - \frac{1}{2}\mu\eta NK\right)\mathbb{E}\left\|\boldsymbol{v}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 - \eta NK \mathbb{E}[\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r) - \Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))] \\
&\quad + \frac{1}{2\mu\eta NK}\mathbb{E}\|\boldsymbol{g}^r - \eta NK \nabla\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r)\|^2 \\
&\quad + 2\mathbb{E}\left\|\boldsymbol{g}^r - \eta NK \nabla\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r)\right\|^2 + 2\mathbb{E}\left\|\eta NK \nabla\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r)\right\|^2 + \mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2.
\end{aligned}$$

where in the first inequality we applied Cauchy-Schwartz inequality and strongly convexity. Since $\Phi(\boldsymbol{\alpha}^r, \cdot)$ is L smooth, we have: $\mathbb{E}\|\nabla\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r)\|^2 \leq 2L\mathbb{E}[\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r) - \Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^*(\boldsymbol{\alpha}^r))]$. Therefore, it follows that:

$$\begin{aligned}
\mathbb{E}\left\|\boldsymbol{v}^{r+1} - \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2 &\leq \left(1 - \frac{1}{2}\mu\eta NK\right)\mathbb{E}\left\|\boldsymbol{v}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 - (\eta NK - 4\eta^2 N^2 K^2 L)\mathbb{E}[\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r) - \Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^*(\boldsymbol{\alpha}^r))] \\
&\quad + \left(\frac{1}{2\mu\eta NK} + 2\right)\mathbb{E}\|\boldsymbol{g}^r - \eta NK \nabla\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r)\|^2 + \mathbb{E}\left\|\boldsymbol{\delta}^r\right\|^2 \qquad (19)
\end{aligned}$$

Now, we examine the term $\|\boldsymbol{g}^r - \eta NK\nabla\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r)\|^2$ in the right hand side of abovee inequality. First according to summation by part (Lemma 8): we let $\boldsymbol{A}_j := \prod_{j'=N-1}^{j+1}(\mathbf{I} - \boldsymbol{Q}_{j'}\mathbf{H}_{j'})$ and $\mathbf{B}_j = \boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)$, then we have:

$$
\begin{aligned}
\boldsymbol{g}^r &= \sum_{j=1}^{N}\prod_{j'=N}^{j+1}(\mathbf{I} - \boldsymbol{Q}_{j'}\mathbf{H}_{j'})\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r) \\
&= \sum_{j=1}^{N}\boldsymbol{A}_j\mathbf{B}_j = \sum_{j=1}^{N}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r) - \sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I} - \boldsymbol{Q}_{j'}\mathbf{H}_{j'}) - \prod_{j'=N}^{n+1}(\mathbf{I} - \boldsymbol{Q}_{j'}\mathbf{H}_{j'})\right)\sum_{j=1}^{n}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r) \\
&= \sum_{j=1}^{N}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r) - \sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I} - \boldsymbol{Q}_{j'}\mathbf{H}_{j'})\right)\boldsymbol{Q}_{n+1}\mathbf{H}_{n+1}\sum_{j=1}^{n}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r).
\end{aligned}
$$

Hence we have:

$$
\begin{aligned}
&\|\boldsymbol{g}^r - \eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}}, \boldsymbol{v}^r)\|^2 \\
&= \left\|\eta NK\nabla\Phi(\hat{\boldsymbol{\alpha}}, \boldsymbol{v}^r) - \sum_{j=1}^{N}\prod_{j'=N-1}^{j+1}(\mathbf{I} - \boldsymbol{Q}_{j'}\mathbf{H}_{j'})\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2 \\
&= \left\|\eta NK\sum_{j=1}^{N}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r) - \left(\sum_{j=1}^{N}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r) - \sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I} - \boldsymbol{Q}_{j'}\mathbf{H}_{j'})\right)\boldsymbol{Q}_{n+1}\mathbf{H}_{n+1}\sum_{j=0}^{n}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right)\right\|^2 \\
&\overset{(1)}{\leq} 2\left\|\left(\eta NK\sum_{j=1}^{N}\hat{\alpha}(\sigma(j))\nabla f_{\sigma(j)}(\boldsymbol{v}^r) - \sum_{j=1}^{N}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right)\right\| \\
&\quad + 2\left\|\sum_{n=1}^{N-1}\left(\prod_{j'=N}^{n+2}(\mathbf{I} - \boldsymbol{Q}_{j'}\mathbf{H}_{j'})\right)\boldsymbol{Q}_{n+1}\mathbf{H}_{n+1}\sum_{j=1}^{n}\boldsymbol{Q}_j\nabla f_{\sigma(j)}(\boldsymbol{v}^r)\right\|^2 \\
&\overset{(2)}{\leq} \left(20\eta^2N^2K^2\left(\frac{e}{4R-e}\right)^2 + 36e^6\eta^4N^4K^4L^4\right)\left\|\nabla\Phi(\hat{\boldsymbol{\alpha}}, \boldsymbol{v}^r)\right\|^2 + 256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2G^2\log(1/p) \\
&\quad + 244e^6\eta^4N^4K^4L^4G^2N\log(1/p) \\
&\overset{(3)}{\leq} \left(20\eta^2N^2K^2\left(\frac{e}{4R-e}\right)^2 + 36e^6\eta^4N^4K^4L^4\right)2L\left(\Phi(\hat{\boldsymbol{\alpha}}, \boldsymbol{v}^r) - \Phi(\hat{\boldsymbol{\alpha}}, \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))\right) \\
&\quad + \left(244e^6\eta^4N^4K^4L^4 + 256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p)
\end{aligned}
$$

where in (1) we apply Jensen's inequality, in (2) we plug in Lemma 11 (a), and Lemma 12, and in (3) we use the L-smoothness of Φ. Plugging above bound back in (19) yields:

$$
\begin{aligned}
&\mathbb{E}\left\|\boldsymbol{v}^{r+1} - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 \\
&\leq (1 - \frac{1}{2}\mu\eta NK)\mathbb{E}\left\|\boldsymbol{v}^r - \boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 + \eta^2N^2K^2e^4\delta^2 \\
&\quad - \left(\eta NK - 4\eta^2N^2K^2L - \left(\frac{1}{2\mu\eta NK} + 2\right)\left(20\eta^2N^2K^2\left(\frac{e}{4R-e}\right)^2 - 36e^6\eta^4N^4K^4L^4\right)\right) \\
&\quad \times \mathbb{E}[\Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^r) - \Phi(\boldsymbol{\alpha}^r, \boldsymbol{v}^*(\boldsymbol{\alpha}^r))] \\
&\quad + \left(\frac{1}{2\mu\eta NK} + 2\right)\left(244e^6\eta^4N^4K^4L^4 + 256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p).
\end{aligned}
$$

Since we choose $\eta = \frac{4\log(\sqrt{NK}R)}{\mu NKR}$, and

$$
R \geq \max\left\{\frac{3}{8}e, \sqrt[3]{\frac{64\kappa^2\log(\sqrt{NK}R)e^6}{9\mu}}\right\}
$$

Hence we have:

$$
\begin{aligned}
&\mathbb{E}\left\|\boldsymbol{v}^{r+1}-\boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2 \\
&\leq (1-\frac{1}{2}\mu\eta NK)\mathbb{E}\left\|\boldsymbol{v}^r-\boldsymbol{v}^*(\boldsymbol{\alpha}^r)\right\|^2+\eta^2N^2K^2e^4\delta^2-\frac{1}{2}\eta NK\underbrace{\mathbb{E}[\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^r)-\Phi(\hat{\boldsymbol{\alpha}},\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}))]}_{\geq 0} \\
&\quad+\left(\frac{1}{2\mu\eta NK}+2\right)\left(244e^6\eta^4N^4K^4L^4+256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p) \\
&\leq \left(1-\frac{1}{2}\mu\eta NK\right)\mathbb{E}\left\|\boldsymbol{v}^r-\boldsymbol{v}^*(\hat{\boldsymbol{\alpha}})\right\|^2+\eta^2N^2K^2e^4\delta^2 \\
&\quad+\left(\frac{1}{2\mu\eta NK}+2\right)\left(244e^6\eta^4N^4K^4L^4+256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p).
\end{aligned}
$$

Putting above inequality back to (18) yields:

$$
\begin{aligned}
\|\boldsymbol{v}^{r+1}-\boldsymbol{v}^*(\boldsymbol{\alpha}^{r+1})\|^2 &\leq \left(1-\frac{1}{4a}\right)\|\boldsymbol{v}^{r+1}-\boldsymbol{v}^*(\boldsymbol{\alpha}^r)\|^2+2\eta^2N^2K^2e^4\delta^2+4a\|\boldsymbol{v}^*(\boldsymbol{\alpha}^{r+1})-\boldsymbol{v}^*(\boldsymbol{\alpha}^r)\|^2 \\
&\quad+2\left(\frac{1}{2\mu\eta NK}+2\right)\left(244e^6\eta^4N^4K^4L^4+256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p) \\
&\leq \left(1-\frac{1}{4a}\right)\|\boldsymbol{v}^{r+1}-\boldsymbol{v}^*(\boldsymbol{\alpha}^r)\|^2+2\eta^2N^2K^2e^4\delta^2 \\
&\quad+2\left(\frac{1}{2\mu\eta NK}+2\right)\left(244e^6\eta^4N^4K^4L^4+256\eta^2N^3K^2\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p) \\
&\quad+O\left(\frac{1}{\mu\eta NK}\left(\left(1-\frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}}+\kappa_g^2L^2\bar{\zeta}_i(\boldsymbol{w}^*)\left(\gamma^2L\left(1-\mu\gamma\right)^rDG+\frac{\gamma^3\kappa\delta^2}{M}+\frac{\gamma^2\delta^2}{M}\right)\right)\right)
\end{aligned}
$$

where at second inequality we plug in Lemma 14. Unrolling the recursion from $r=R$ to 0, and plugging in $\eta=\frac{4\log(NKR^3)}{\mu NKR}$ yields:

$$
\begin{aligned}
&\|\boldsymbol{v}^R-\boldsymbol{v}^*(\boldsymbol{\alpha}^R)\|^2 \\
&\leq \left(1-\frac{1}{4}\mu\eta NK\right)^R\|\boldsymbol{v}^0-\boldsymbol{v}^*(\boldsymbol{\alpha}^0)\|^2+\frac{1}{\mu}\eta NKe^4\delta^2 \\
&+8\frac{1}{\mu}\left(\frac{1}{2\mu\eta NK}+2\right)\left(244e^6\eta^3N^3K^3L^4+256\eta N^2K\left(\frac{e}{4R-e}\right)^2\right)G^2N\log(1/p) \\
&+O\left(\frac{1}{\mu\eta NK}\sum_{r=0}^{R}\left(1-\frac{1}{4a}\right)^{R-r}\left(\left(1-\frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}}+\kappa_g^2L^2\bar{\zeta}_i(\boldsymbol{w}^*)\left(\gamma^2L\left(1-\mu\gamma\right)^rDG+\frac{\gamma^3\kappa\delta^2}{M}+\frac{\gamma^2\delta^2}{M}\right)\right)\right) \\
&\leq O\left(\frac{\|\boldsymbol{v}^0-\boldsymbol{v}^*(\boldsymbol{\alpha}^0)\|^2}{NKR^3}\right)+\tilde{O}\left(\left(\frac{\kappa^4}{R^2}+\frac{N}{\mu^2R^2}\right)G^2N\log(1/p)+\frac{\delta^2}{\mu R}\right) \\
&+O\left(\frac{1}{\mu\eta NK}\sum_{r=0}^{R}\left(1-\frac{\log(NKR^3)}{R}\right)^{R-r}\left(\left(1-\frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}}+\kappa_g^2L^2\bar{\zeta}_i(\boldsymbol{w}^*)\left(\gamma^2L\left(1-\mu\gamma\right)^rDG+\frac{\gamma^3\kappa\delta^2}{M}+\frac{\gamma^2\delta^2}{M}\right)\right)\right)
\end{aligned}
$$

Plugging in $\gamma = \frac{\log(NKR^3)}{\mu R}$ yields:

$$\begin{aligned}
\|\boldsymbol{v}^R - \boldsymbol{v}^*(\boldsymbol{\alpha}^R)\|^2 &\leq O\left(\frac{\|\boldsymbol{v}^0 - \boldsymbol{v}^*(\boldsymbol{\alpha}^0)\|^2}{NKR^3}\right) + \tilde{O}\left(\left(\frac{\kappa^4}{R^2} + \frac{N}{\mu^2 R^2}\right) G^2 N \log(1/p) + \frac{\delta^2}{\mu R}\right) \\
&\quad + \tilde{O}\left(R\left(\gamma^2 L^2 \sum_{r=0}^{R}\left(1 - \frac{\log(NKR^3)}{R}\right)^R \kappa_g^2 L^2 \bar{\zeta}_i(\boldsymbol{w}^*) DG\right.\right. \\
&\quad \left.\left. + \sum_{r=0}^{R}\left(1 - \frac{\log(NKR^3)}{R}\right)^{R-r}\left(\left(1 - \frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}} + \frac{\kappa_g^2 L^2 \bar{\zeta}_i(\boldsymbol{w}^*)\gamma^2\delta^2}{M}\right)\right)\right) \\
&\leq O\left(\frac{D^2}{NKR^3}\right) + \tilde{O}\left(\left(\frac{\kappa^4}{R^2} + \frac{N}{\mu^2 R^2}\right) G^2 N \log(1/p) + \frac{\delta^2}{\mu R}\right) \\
&\quad + \tilde{O}\left(\frac{\kappa^2 \kappa_g^2 L^2 \bar{\zeta}_i(\boldsymbol{w}^*) DG}{R} + R^2\left(\left(1 - \frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}} + \frac{\kappa_g^2 \kappa^2 \bar{\zeta}_i(\boldsymbol{w}^*)\delta^2}{\mu^2 M R^2}\right)\right)
\end{aligned}$$

Since $\hat{\boldsymbol{v}}_i = \boldsymbol{v}^R$ and $\hat{\boldsymbol{\alpha}}_i = \boldsymbol{\alpha}^R$, we have the convergence of $\|\hat{\boldsymbol{v}}_i - \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}_i)\|^2$. Plugging this convergence rate together with the convergence of $\|\hat{\boldsymbol{\alpha}}_i - \boldsymbol{\alpha}^*\|^2$ from Lemma 15:

$$\begin{aligned}
\left\|\hat{\boldsymbol{\alpha}}_i - \boldsymbol{\alpha}^*\right\|^2 &\leq O\left(2(1 - \frac{1}{\kappa_g})^{T_{\boldsymbol{\alpha}}} + O\left(\kappa_g^2 \bar{\zeta}_i(\boldsymbol{w}^*) L^2 \frac{2}{\mu}(1 - \mu\gamma)^R + 2\gamma\frac{\delta^2}{\mu^2 M}\right)\right) \\
&\leq \tilde{O}\left(\left(1 - \frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}} + O\left(\kappa_g^2 \bar{\zeta}_i(\boldsymbol{w}^*) L^2 \frac{2}{\mu}\frac{1}{R} + \frac{\delta^2}{\mu^3 RM}\right)\right)
\end{aligned}$$

together with applying Lemma 2 leads to:

$$\begin{aligned}
\Phi(\boldsymbol{\alpha}_i^*, \hat{\boldsymbol{v}}_i) - \Phi(\boldsymbol{\alpha}_i^*, \boldsymbol{v}_i^*) &\leq L\left\|\hat{\boldsymbol{v}}_i - \boldsymbol{v}^*(\hat{\boldsymbol{\alpha}}_i)\right\|^2 + \kappa^2 L\left\|\hat{\boldsymbol{\alpha}}_i - \boldsymbol{\alpha}_i^*\right\|^2 \\
&\leq O\left(L\frac{D^2}{NKR^3}\right) + \tilde{O}\left(\left(\frac{\kappa^4 L}{R^2} + \frac{NL}{\mu^2 R^2}\right) G^2 N \log(1/p) + \frac{L\delta^2}{\mu R}\right) \\
&\quad + \tilde{O}\left(\frac{\kappa^2 \kappa_g^2 L^3 \bar{\zeta}_i(\boldsymbol{w}^*) DG}{R} + LR^2\left(1 - \frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}} + \frac{L\kappa_g^2 \kappa^2 \bar{\zeta}_i(\boldsymbol{w}^*)\delta^2}{\mu^2 M}\right) \\
&\quad + \kappa^2 L\tilde{O}\left(\left(1 - \frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}} + \frac{\kappa_g^2 \bar{\zeta}_i(\boldsymbol{w}^*)\kappa L}{R} + \frac{\delta^2}{\mu^3 RM}\right) \\
&\leq O\left(L\frac{D^2}{NKR^3}\right) + \tilde{O}\left(\left(\frac{\kappa^4 L}{R^2} + \frac{NL}{\mu^2 R^2}\right) G^2 N \log(1/p) + \frac{L\delta^2}{\mu R}\right) \\
&\quad + \tilde{O}\left(\frac{\kappa^2 \kappa_g^2 L^3 \bar{\zeta}_i(\boldsymbol{w}^*) DG}{R} + LR^2\left(1 - \frac{1}{\kappa_g}\right)^{T_{\boldsymbol{\alpha}}} + \frac{L\kappa_g^2 \kappa^2 \bar{\zeta}_i(\boldsymbol{w}^*)\delta^2}{\mu^2 M}\right).
\end{aligned}$$

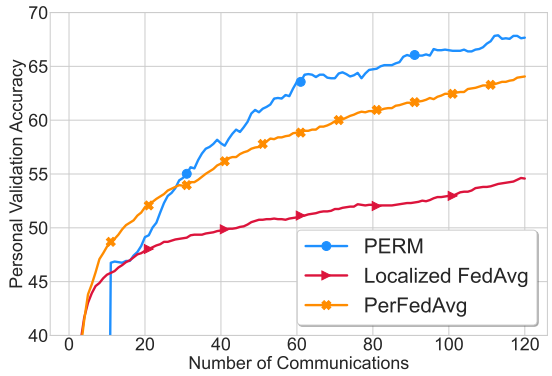
thus completing the proof as desired. □

C ADDITIONAL EXPERIMENTS

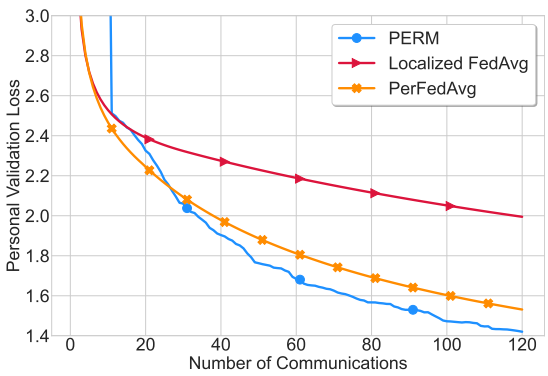
In addition to the synthetic dataset discussed in the main body, we run experiments on the EMNIST dataset [3], which is naturally distributed in a federated setting. In this case, we chose 50 clients and use a 2-layer MLP model, each with 200 neurons. We compare the PERM algorithm with the localized model in FedAvg and perFedAvg [8]. As it can be seen in Figure 2, PERM can learn a better personalized model by attending to each client's data according to the similarity of the data distribution between clients. The learned values of $\boldsymbol{\alpha}$ show that the clients are learning from each others' data, and not focused on their own data only. This signifies that the distribution of data among clients in this dataset is not highly heterogeneous. Note that, since we are using a subset of clients in the EMNIST dataset for the training (only 50 clients for 100 rounds of communication), the results would be sub-optimal. Nonetheless, the experiments are designed to show the effectiveness of different algorithms.

C.1 The effectiveness of learned mixture weights

To show the effectiveness of two-stage PERM algorithm, as well as the effects of heterogeneity on distribution of data among clients on the learned weights $\boldsymbol{\alpha}$ in the algorithm, we run this algorithm on MNIST dataset. We use 50 clients, and the model is an MLP, similar to the EMNIST experiment. In this case, once we distribute the data randomly across clients (homogeneous) and once only allocating 1 class per client (highly heterogeneous). As it can be seen from Figure 4, when the data distribution is homogeneous the learned values of $\boldsymbol{\alpha}$ as

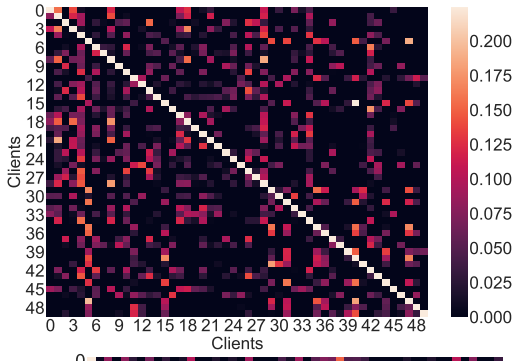


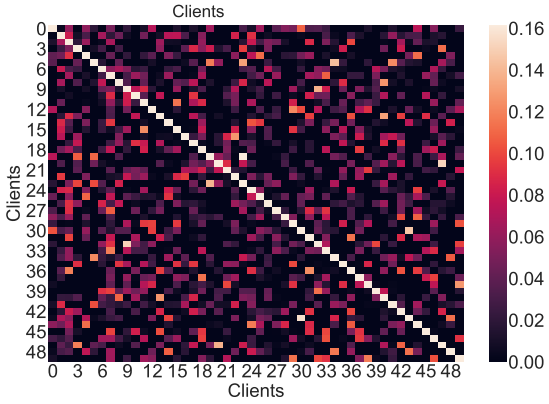
(a) Personalized Accuracy



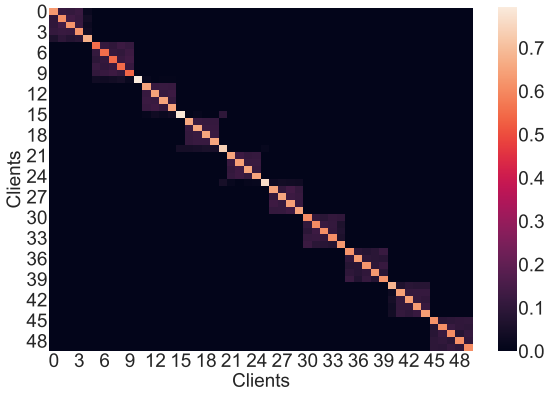
(b) Personalized Loss

Figure 2: Comparative Analysis of Personalization methods, including our single-loop PERM algorithm, localized FedAvg, and perFedAg, with EMNIST dataset. The disparity in personalized accuracy and loss highlights PERM's capability in leveraging relevant client correlations.





(a) Homogeneous distribution



(b) Highly heterogeneous distribution

Figure 3: The heat map of the learned α values for the PERM algorithms. The weights signify that clients mutually benefiting from one another's data, which also highlight that the distribution of data is not significantly heterogeneous in this dataset.

Figure 4: Comparing the performance of two-stage PERM algorithm in learning α values on heterogeneous and homogeneous data distributions. WE use MNIST dataset across 50 clients with homogeneous and heterogeneous distributions.

diffused across clients. However, when the data is highly heterogeneous, the learned $\boldsymbol{\alpha}$ values will be highly sparse, indicating that each client is mostly learning from its own data and some other clients with partial distribution similarity.