
Federated Random Reshuffling with Compression and Variance Reduction

Grigory Malinovsky¹ Peter Richtárik¹

Abstract

Random Reshuffling (RR), which is a variant of Stochastic Gradient Descent (SGD) employing sampling *without* replacement, is an immensely popular method for training supervised machine learning models via empirical risk minimization. Due to its superior practical performance, it is embedded and often set as default in common machine learning software. Under the name FedRR, this method was recently shown to be applicable to federated learning (Mishchenko et al., 2021), with superior performance when compared to common baselines such as Local SGD. Inspired by this development, we design three new algorithms with the aim of further improving FedRR: compressed FedRR, and two variance reduced extensions: one for taming the variance coming from shuffling, and the other for taming the variance due to compression. The variance reduction mechanism for compression allows us to eliminate dependence on the compression parameter, and applying additional controlled linear perturbations for Random Reshuffling, introduced by Malinovsky et al. (2021) helps to eliminate variance at the optimum. We provide the first analysis of compressed local methods under standard assumptions without bounded gradient assumptions and for heterogeneous data, overcoming the limitations of the compression operator. We corroborate our theoretical results with experiments on synthetic and real data sets.

1. Introduction

In modern machine learning world the main approach for training supervised machine learning models is Empirical Risk Minimization. While the ultimate goal of supervised learning is to train models that generalize well to unseen

data, in practice only a finite data set is available during training. This leads to the following finite-sum optimization problem

$$\min_{x \in \mathbb{R}^d} \left[f(x) = \frac{1}{M} \sum_{m=1}^M g_m(x) \right], \quad (1)$$

where each function we also have finite-sum structure

$$g_m = \frac{1}{n} \sum_{i=1}^n f_{m,i}(x).$$

Big machine learning models are typically trained in a distributed setting. The training data is distributed across several workers, all of which compute local updates and then communicate them to the server. We are particularly interested in the Federated Learning setting. Federated Learning (Konečný et al., 2016) is a subarea of distributed machine learning, where the number of devices n is enormous. Usually, millions and local devices are heterogeneous to local data, computational and memory resources. Also, users want to keep their privacy, so the algorithm should do training locally. Moreover, communication between workers and server should be conducted via a trusted aggregation server, which is very expensive.

Communication as the bottleneck. Nowadays, in literature, we have two strategies to overcome communication issue in federated learning. The first one is communication compression, where our goal is to reduce the number of communicated bits using gradient compression scheme (Mishchenko et al., 2019; Gorbunov et al., 2020) and compressed iterates (Khaled & Richtárik, 2019; Chraïbi et al., 2019). There are many compression techniques such as quantization (Alistarh et al., 2017; Bernstein et al., 2018; Ramezani-Kebrya et al., 2019), sparsification (Aji & Heafield, 2017; Lin et al., 2017; Wangni et al., 2017; Alistarh et al., 2018) and other approaches (Shamir et al., 2014; Vogels et al., 2019; Wu et al., 2018). The second strategy to tackle this issue is increasing the total number of local steps between the communication rounds. The most popular algorithm — FedAvg (McMahan et al., 2017)— is based on this idea. Many papers provide theoretical justifications for special cases of FedAvg such as local GD (Khaled et al., 2019) and local SGD (Khaled et al., 2020; Gorbunov et al., 2020; Stich, 2018; Lin et al., 2018). The natural union

¹King Abdullah University of Science and Technology, Thuwal, Saudi Arabia. Correspondence to: Grigory Malinovsky <<https://grigory-malinovsky.github.io>>.

of communication compression and local computations is presented in Basu et al. (2020); Haddadpour et al. (2021). However, the theory provided in these papers is limited due to unrealistic assumptions.

Sampling without replacement. Stochastic first-order algorithms, in particular, have attracted a lot of attention in the machine learning world. Of these, stochastic gradient descent (SGD) is perhaps the best known and the most basic. SGD has a long history (Robbins & Monro, 1951) and is therefore well-studied and well-understood (Gower et al., 2019). However, methods based on data permutations (Bottou, 2009), when data points are shuffled randomly and processed in order, without replacement, show better performance than SGD (Recht & Ré, 2013). Also, this method has software implementation advantages since these methods are friendly to cache locality (Bengio, 2012). The most popular model in this class is Random Reshuffling (Recht & Ré, 2012). Shuffle-Once (Safran & Shamir, 2020) uses a similar approach, but shuffling occurs only once, at the very beginning, before the training begins. Random Reshuffling and Shuffle-Once have a long history, and many theoretical works try to show advantages of Random Reshuffling (Gürbüzbalaban et al., 2019; Haochen & Sra, 2019; Nagaraj et al., 2019) and Shuffle Once (Rajput et al., 2020).

Federated Random Reshuffling. Recent advances of Mishchenko et al. (2020) and providing extension for Federated Learning (Mishchenko et al., 2021) allow us to consider this technique as a particular variant of FedAvg with a fixed number of local computations and sampling without replacement.

Variance Reduction. Compression operators help to reduce the number of transmitted bits, but at the same time, it starts to be a source of variance, which increase the neighbourhood of the optimal solution. This variance can sufficiently slow down the algorithm. In order to overcome this challenge, we need to use a variance reduction mechanism. The idea of this approach is based on shifted compression operator, and firstly it was proposed for compressed gradients (Mishchenko et al., 2019). For compressed iterates, the variance reduction mechanism was proposed in Chraibi et al. (2019). Moreover, stochastic first-order become to be a source of variance due to their random nature. Hopefully, there is another variance reduction mechanism that can help with this type of variance. There are many methods which use sampling with replacement such as SVRG (Johnson & Zhang, 2013), L-SVRG (Kovalev et al., 2020), SAGA (Defazio et al., 2014a), SAG (Roux et al., 2012), Finito (Defazio et al., 2014b) etc. For permutation-based algorithms, we have only a few variance-reduced methods (Ying et al., 2019; Park & Ryu, 2020; Mokhtari et al., 2018). In a recent paper of Malinovsky et al. (2021) introduced linear perturbation

reformulation that allows getting better rates for variance reduced Random Reshuffling.

2. Contributions

In this section, we outline our work’s key contributions and offer explanations and clarifications regarding some of the development.

Compressed FedRR. We propose first method which combine three ideas: compression, local steps and sampling without replacement. This is compressed federated random reshuffling. Basic approach is to apply compression operator to the iterates after each epoch and then aggregate compressed updates. Applying compression to the iterates can significantly worsen convergence properties. We prove the following rate:

$$\begin{aligned} \mathbb{E}\|x_T - x_*\|^2 &\leq (1 - \gamma\mu)^{\frac{nT}{2}} \mathbb{E}\|x_0 - x_*\|^2 \\ &+ \frac{2\omega}{M} \frac{1}{\gamma\mu} \frac{1}{M} \sum_{m=1}^M \|x_{*,m}^n\|^2 \\ &+ \frac{2}{\mu} \left(1 + \frac{2\omega}{M}\right) \gamma^2 L \frac{1}{M} \sum_{m=1}^M \left(\|\nabla F_m(x_*)\|^2 + \frac{n}{4} \sigma_{*,m}^2\right) \end{aligned}$$

As we can see, we have a part with linear rate: $(1 - \gamma\mu)^{\frac{nT}{2}} \mathbb{E}\|x_0 - x_*\|^2$. Also there are three sources of variance in the optimum. The first one is $\frac{2\omega}{M} \frac{1}{\gamma\mu} \frac{1}{M} \sum_{m=1}^M \|x_{*,m}^n\|^2$ and it caused by compression. It is equal to zero only if $\omega = 0$. This term cannot be eliminated by decreasing step-sizes strategies. The second term is $\frac{2}{\mu} \left(1 + \frac{2\omega}{M}\right) \gamma^2 L \frac{1}{M} \sum_{m=1}^M \frac{n}{4} \sigma_{*,m}^2$. This source of variance is caused by stochasticity of Random Reshuffling method. This variance can be decreased by decreasing step-sizes. The third term is $\frac{2}{\mu} \left(1 + \frac{2\omega}{M}\right) \gamma^2 L \frac{1}{M} \sum_{m=1}^M \|\nabla F_m(x_*)\|^2$. This source of variance is caused by heterogeneity of data. In other words, if we have the same optimum for all functions $F_m(x)$, we can get rid of this term. In heterogenous regime we need to use additional mechanism for controlling client drift such as SCAFFOLD (Karimireddy et al., 2019).

Variance-reduced Compressed FedRR. In the previous section, it was shown that using compressed iterates causes additional variance, which cannot be vanished by decreasing step-sizes strategies. Moreover, it forces us to have an additional assumption on the compression operator. We need to have very small compression parameter $\omega \leq \frac{M\gamma\mu\epsilon}{2\|x_{*,m}^n\|^2}$. In order to fix it, we propose Variance-reduced compressed FedRR (FedCRR-VR) that utilize shifted compressed updates with learning shifts. The similar mechanism is used in Mishchenko et al. (2019) and Gorbunov et al. (2020).

Double Variance-reduced Compressed FedRR. We propose a modification of Variance-reduced Compressed Fed-

erated Random Reshuffling, which allows eliminating variance caused by stochasticity. We armed Variance-reduced Compressed FedRR with linear permutation approach proposed by Malinovsky et al. (2021). Now we get both variance reduction mechanisms in one algorithm.

3. Preliminaries

3.1. L -smooth and μ -strongly convex functions

Before introducing our convergence results, let us first formulate all concepts that we use throughout the paper. Firstly, we consider a class of μ -strongly convex and L -smooth functions.

Definition 1. A differentiable function f is μ -strongly convex if

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{\mu}{2} \|y - x\|^2$$

for $\mu > 0$ and all x, y .

Definition 2. A differentiable function f is L -smooth if

$$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{L}{2} \|y - x\|^2$$

for some $L > 0$ and all x, y .

There is the first assumption that we use in all theorems.

Assumption 1. Each $f_{m,i}$ is μ -strongly convex and L -smooth.

We also need to define Bregman divergence, which is used many times in analysis.

Definition 3. The Bregman divergence with respect to f is the mapping $D_f : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ defined as follows:

$$D_f(x, y) \stackrel{\text{def}}{=} f(x) - f(y) - \langle \nabla f(y), x - y \rangle.$$

3.2. Compression operator

In order to overcome communication issues, we apply a compression operator to the iterates. Now we are going to extend Federated Random Reshuffling using compression. Let us define the concept of compressors.

Definition 4. We say that a randomized map $\mathcal{C} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is in class $\mathbb{B}^d(\omega)$ if there exists a constant $\omega \geq 0$ such that the following relations hold for all $x \in \mathbb{R}^d$:

$$\mathbb{E}[\mathcal{C}(x)] = x, \quad \mathbb{E}[\|\mathcal{C}(x)\|^2] \leq (\omega + 1)\|x\|^2.$$

Assumption 2. All compression operators are in class $\mathbb{B}^d(\omega)$.

This class of compressor operators is classical in literature (Mishchenko et al., 2019; Horvath et al., 2019; Basu et al., 2020).

3.3. Random Reshuffling, Shuffle Once

In order to conduct analysis for sampling without replacement we need to establish specific notions. We sample a random permutation $\{\pi_0, \pi_1, \dots, \pi_{n-1}\}$ of the set $\{1, 2, \dots, n\}$, and proceed with n iterates of the form $x_{t,m}^{i+1} = x_{t,m}^i - \gamma \nabla f_{m,\pi_i}(x_{t,m}^i)$ at each machine locally. We also consider option when we have only one random permutation, at the very beginning, and then algorithm uses this permutation during the whole process.

For a constant stepsize and a fixed permutation, we define intermediate limit point:

$$x_*^i \stackrel{\text{def}}{=} x_* - \gamma \sum_{j=0}^{i-1} \nabla f_{\pi_j}(x_*), \quad i = 1, \dots, n-1.$$

To measure the closeness between x_* and x_*^n we use definition from Mishchenko et al. (2021) of Shuffling radius.

Definition 5. For given a stepsize $\gamma > 0$ and a random permutation π of $\{1, 2, \dots, n\}$ shuffling radius is defined by

$$\sigma_{\text{rad}}^2 \stackrel{\text{def}}{=} \max_{i=1, \dots, n-1} \left[\frac{1}{\gamma^2} \mathbb{E}_{\pi} [D_{f_{\pi_i}}(x_*^i, x_*)] \right].$$

We also need to define the most popular parameter for method's stochasticity.

Definition 6. Variance at the optimum:

$$\sigma_*^2 \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^n \|\nabla f_i(x_*) - \nabla f(x_*)\|^2.$$

The shuffling radius for permutation-based algorithms is natural, and it is more convenient to work with this concept. However, we need to have an upper bound in terms of σ_*^2 to compare different methods. To get an upper bound for shuffling radius, we need to use a lemma in Mishchenko et al. (2020) that bounds variance of sampling without replacement.

Theorem 1. For any stepsize $\gamma > 0$ and any random permutation π of $\{1, 2, \dots, n\}$ we have

$$\sigma_{\text{rad}}^2 \leq \frac{L_{\max}}{2} n \left(n \|\nabla f(x_*)\|^2 + \frac{1}{2} \sigma_*^2 \right).$$

In case when we have only one node we obtain that $\|\nabla f(x_*)\|^2 = 0$. However, in multiple node case we will need this term.

3.4. Lifted problem reformulation

Let us consider a bigger product space by introducing dummy variables and the constraint $x_1 = x_2 = \dots = x_M$. We need to define regularizer for this reformulation:

$$\psi(x_1, \dots, x_M) = \begin{cases} 0, & x_1 = \dots = x_M \\ +\infty, & \text{otherwise.} \end{cases}$$

Table 1. Comparison of the main features of our algorithms and results with other methods with compression.

Feature	Variance reduction for compression	Variance reduction for stochasticity	Variance reduction for client-drift	RR ⁽¹⁾	No additional assumption	Communication complexity ⁽²⁾
QSPARSE (Basu et al., 2020)	✗	✓	✓	✗	✗ ⁽³⁾	$\mathcal{O}\left(\frac{\kappa(\omega+1)}{\sqrt{\varepsilon}}\right)$
FedCOMGATE (Haddadpour et al., 2021)	✓	✓	✓	✗	✗ ⁽⁴⁾	$\tilde{\mathcal{O}}\left(\kappa\left(\frac{\omega}{M} + 1\right)\right)$
DIANA(Mishchenko et al., 2019)	✓	✗	✗	✗	✓	$\mathcal{O}\left(\kappa \max\left\{V^0, \frac{(1-\alpha_p)\sigma^2}{\mu L n \alpha_p}\right\} \frac{1}{\varepsilon}\right)$
FedCRR	✗	✗	✗	✓	✗ ⁽⁵⁾	$\tilde{\mathcal{O}}\left(\left(\kappa + \frac{\sqrt{\kappa}(\sigma_{cl} + \sigma_{st})}{\mu\sqrt{\varepsilon}}\right)\right)$ ⁽⁶⁾
FedCRR-VR	✓	✗	✗	✓	✓	$\tilde{\mathcal{O}}\left(\frac{(\omega+1)\left(1-\frac{1}{\kappa}\right)^n}{\left(1-\left(1-\frac{1}{\kappa}\right)^n\right)^2} + \frac{\sqrt{\kappa}(\sigma_{cl} + \sigma_{st})}{\mu\sqrt{\varepsilon}}\right)$
FedCRR-VR-2	✓	✓	✗	✓	✗ ⁽⁷⁾	$\tilde{\mathcal{O}}\left(\frac{(\omega+1)\left(1-\frac{1}{\kappa\sqrt{\kappa n}}\right)^{\frac{n}{2}}}{\left(1-\left(1-\frac{1}{\kappa\sqrt{\kappa n}}\right)^{\frac{n}{2}}\right)^2} + \frac{\sqrt{\kappa}(\sigma_{cl})}{\mu\sqrt{\varepsilon}}\right)$

(1) Sampling without replacement (Random Reshuffling).

(2) $\tilde{\mathcal{O}}$ notation ignores logarithmic factors.

(3) Bounded second moment: $\mathbb{E}[\|\nabla f_i(x)\|_2^2] \leq G_1^2$. Contractive compressor assumption: $\mathbb{E}[\|x - \mathcal{C}(x)\|_2^2] \leq (1-\gamma)\|x\|_2^2$ for $\gamma \in (0, 1]$.

(4) Assumption that for all $x_1, \dots, x_n \in \mathbb{R}^d$ the compressor \mathcal{C} satisfies $\mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^n \mathcal{C}(x_i)\right\|^2 - \left\|\mathcal{C}\left(\frac{1}{n}\sum_{i=1}^n x_i\right)\right\|^2\right] \leq G_2$. Classical compression operators like RandK and l_2 -quantization on \mathbb{R}^d does not satisfy this condition. As counterexample we can set $n = 2$ and $x_2 = -x_1 = t \cdot (1, 1, \dots, 1)^\top$ with arbitrary large $t > 0$.

(5) Small compression operator: $\omega \leq \frac{M\gamma\mu\varepsilon}{2\|x_{*,m}^n\|^2}$.

(6) Client drift: $\sigma_{cl} = \frac{1}{M}\sum_{m=1}^M \|F_m(x_*)\|$, sum of local variances: $\sum_{m=1}^M \frac{\sqrt{n}}{2}\sigma_{*,m}$.

(7) Big data regime: $n > \log\left(\frac{1}{1-\delta^2}\right)\left(\log\left(\frac{1}{1-\gamma\mu}\right)\right)^{-1}$, where $0 < \delta < 1$.

Using this regularizer, we can establish the reformulated problem:

$$\min_{x_1, \dots, x_M \in \mathbb{R}^d} \frac{1}{nM} \sum_{m=1}^M F_m(x_m) + \psi(x_1, \dots, x_M)$$

$$F_m(x) = \sum_{j=1}^n f_{mj}(x).$$

For this reformulated problem we need to have an upper bound for shuffling radius. First we need to define variance of method's stochasticity in distributed case.

Definition 7. Variance of local gradients:

$$\sigma_{m,*}^2 \stackrel{\text{def}}{=} \frac{1}{n} \sum_{j=1}^n \left\| \nabla f_{mj}(x_*) - \frac{1}{n} \nabla F_m(x_*) \right\|^2.$$

Now we need to use a lemma from (Mishchenko et al., 2021) to bound shuffled radius for the reformulated problem:

Lemma 1. The shuffling radius σ_{rad}^2 of lifted problem is upper bounded by

$$\sigma_{rad}^2 \leq L \sum_{m=1}^M \left(\|\nabla F_m(x_*)\|^2 + \frac{n}{4} \sigma_{m,*}^2 \right).$$

We can see that there are two parts of variance. First one depends on the sum of local variances $\sum_{m=1}^M \sigma_{m,*}^2$.

The second part depends on sum of local gradient norms $\sum_{m=1}^M \|\nabla F_m(x_*)\|^2$. Both of these terms appear in analysis of local SGD (Khaled et al., 2020).

4. Compressed Federated Random Reshuffling

In this section, we propose a direct application of compressed iterates to federated Random Reshuffling. In this procedure server distributes the current point to workers, then each worker computes the full epoch according to its sampled permutation locally. After that, the final iterate $x_{t,m}^n$ is compressed and transmitted to the server, where all updates are aggregated by taking the average. The following theorem states how the convergence of FedCRR depends on compression parameter, shuffling radius and client drift.

Theorem 2. Suppose that Assumption 2 and Assumption 1 hold. Additionally assume that compression parameter is sufficiently small: $\omega \leq \frac{M}{2} \frac{1-(1-\gamma\mu)^{\frac{n}{2}}}{(1-\gamma\mu)^{\frac{n}{2}}}$. Then provided the stepsize satisfies $\gamma \leq \frac{1}{L}$ the iterates generated by FedCRR

Algorithm 1 Federated Compressed Random Reshuffling (FedCRR) and Shuffle-Once (FedCSO)

- 1: **Parameters:** Stepsize $\gamma > 0$, initial vector $x_0 = x_0^0 \in \mathbb{R}^d$, number of epochs T
 - 2: For each m , sample permutation $\pi_{0,m}, \pi_{1,m}, \dots, \pi_{n-1,m}$ of $\{1, 2, \dots, n\}$ (Only FedCSO)
 - 3: **for** epochs $t = 0, 1, \dots, T - 1$ **do**
 - 4: **for** $m = 1, \dots, M$ locally in parallel **do**
 - 5: $x_{t,m}^0 = x_t$
 - 6: Sample permutation $\pi_{0,m}, \pi_{1,m}, \dots, \pi_{n-1,m}$ of $\{1, 2, \dots, n\}$ (Only FedCRR)
 - 7: **for** $i = 0, 1, \dots, n - 1$ **do**
 - 8: $x_{t,m}^{i+1} = x_{t,m}^i - \gamma \nabla f_{\pi_{i,m}}(x_{t,m}^i)$
 - 9: **end for**
 - 10: $q_{t,m} = \mathcal{C}(x_{t,m}^n)$
 - 11: **end for**
 - 12: $x_{t+1} = \frac{1}{M} \sum_{m=1}^M q_{t,m}$;
 - 13: **end for**
-

or FedCSO (Algorithm 1) satisfy

$$\begin{aligned} \mathbb{E} [\|x_{t+1} - x_*\|^2] &\leq (1 - \gamma\mu)^{\frac{nT}{2}} \|x_0 - x_*\|^2 \\ &+ \frac{2}{\mu} \gamma^2 L_{\max} \frac{1}{M} \sum_{m=1}^M \left(\|F_m(x_*)\|^2 + \frac{n}{4} \sigma_{*,m}^2 \right) \\ &+ \frac{2\omega}{M} \frac{1}{\gamma\mu} \frac{1}{M} \sum_{m=1}^M \|x_{*,m}^n\|^2. \end{aligned}$$

We can see that the last term makes the largest contribution to the size of the neighborhood and decreasing stepsizes cannot help. Now we establish communication complexity.

Corollary 1. *Let the assumptions in the Theorem 2 hold. Also assume that $\omega \leq \frac{M\gamma\mu\varepsilon}{2\|x_{*,m}^n\|^2}$. Then the communication complexity of Algorithm 1 is*

$$T = \tilde{\mathcal{O}} \left(\left(\kappa + \frac{\sqrt{\kappa}}{\mu\sqrt{\varepsilon}} \Delta \right) \log \left(\frac{1}{\varepsilon} \right) \right),$$

where $\Delta = \frac{1}{M} \sum_{m=1}^M (\|\nabla F_m(x_*)\| + \sqrt{n}\sigma_{*,m})$

5. Variance Reduced Compressed Federated Random Reshuffling

In this section, we introduce a variance reduction mechanism for compression in order to upgrade Algorithm 1. The main part of the algorithm remains the same. However, after each epoch, we apply the compression operator to the difference between local iterates and learning shifts. After that, at each node, we compute updates of learning shifts. To control the learning process of shifts, we use additional parameter α . To get convergence we need to

Algorithm 2 Variance Reduced Federated Compressed Random Reshuffling (FedCRR-VR) and Shuffle-Once (FedCSO-VR)

- 1: **Parameters:** Stepsize $\gamma > 0$, initial vector $x_0 = x_0^0 \in \mathbb{R}^d$, number of epochs T
 - 2: For each m , sample permutation $\pi_{0,m}, \pi_{1,m}, \dots, \pi_{n-1,m}$ of $\{1, 2, \dots, n\}$ (Only FedCSO-VR)
 - 3: **for** epochs $t = 0, 1, \dots, T - 1$ **do**
 - 4: **for** $m = 1, \dots, M$ locally in parallel **do**
 - 5: $x_{t,m}^0 = x_t$
 - 6: Sample permutation $\pi_{0,m}, \pi_{1,m}, \dots, \pi_{n-1,m}$ of $\{1, 2, \dots, n\}$ (Only FedCRR-VR)
 - 7: **for** $i = 0, 1, \dots, n - 1$ **do**
 - 8: $x_{t,m}^{i+1} = x_{t,m}^i - \gamma \nabla f_{\pi_{i,m}}(x_{t,m}^i)$
 - 9: **end for**
 - 10: $q_{t,m} = \mathcal{C}(x_{t,m}^n - h_{t,m})$
 - 11: $h_{t+1,m} = h_{t,m} + \alpha q_{t,m}$
 - 12: **end for**
 - 13: $x_{t+1} = (1 - \eta)x_t + \eta \frac{1}{M} \sum_{m=1}^M (q_{t,m} + h_{t,m})$;
 - 14: **end for**
-

satisfy $\alpha \leq \frac{1}{\omega+1}$. After that server aggregates updates by using a convex combination of previous iterate and average of updates. To control this convex combination, we use additional parameter η . The next theorem shows that this mechanism helps to get rid of compression variance and the additional assumptions. To get the convergence rate, we introduce the Lyapunov function.

Theorem 3. *Suppose that Assumption 1 and Assumption 2 hold. Then provided the stepsize satisfies $\gamma \leq \frac{1}{L}$, $\alpha \leq \frac{1}{\omega+1}$ and $\eta \leq \min \left(1, \frac{(1-(1-\gamma\mu)^n)M}{12\omega(1-\gamma\mu)^n} \right)$ the iterates generated by FedCRR-VR or FedCSO-VR (Algorithm 2) satisfy*

$$\begin{aligned} \mathbb{E} \Psi_T &\leq \left(1 - \frac{\min(\alpha, \eta(1 - (1 - \gamma\mu)^n))}{2} \right)^T \Psi_0 \\ &+ \frac{2 \left(\alpha + \eta + \frac{2\eta^2\omega}{M} \right) \gamma^3 L_{\max}}{M(\alpha, \eta(1 - (1 - \gamma\mu)^n))} \sum_{m=1}^M \delta_m, \end{aligned}$$

where Lyapunov function is defined as $\Psi_t = \|x_t - x_*\|^2 + \frac{4\eta^2\omega}{\alpha M} \frac{1}{M} \sum_{m=1}^M \|h_{t,m} - x_{*,m}^n\|^2$ and $\delta_m = \left(\|\nabla F_m(x_*)\|^2 + \frac{n}{4} \sigma_{m,*}^2 \right)$.

Now there is no compression variance term anymore. Next corollary demonstrates communication complexity.

Corollary 2. *Let the assumptions in the Theorem 3 hold. Then the communication complexity of Algorithm 2 is*

$$T = \mathcal{O} \left(\left(\frac{(\omega+1)(1-\frac{1}{\kappa})^n}{(1-(1-\frac{1}{\kappa})^n)^2} + \frac{\sqrt{\kappa}}{\mu\sqrt{\varepsilon}} \Delta \right) \log \left(\frac{1}{\varepsilon} \right) \right),$$

where $\Delta = \frac{1}{M} \sum_{m=1}^M (\|\nabla F_m(x_*)\| + \sqrt{n}\sigma_{*,m})$.

Algorithm 3 Double Variance Reduced Federated Compressed Random Reshuffling (FedCRR-VR-2) and Shuffle-Once (FedCSO-VR-2)

- 1: **Parameters:** Stepsize $\gamma > 0$, initial vector $x_0 = x_0^0 \in \mathbb{R}^d$, number of epochs T
- 2: For each m , sample permutation $\pi_{0,m}, \pi_{1,m}, \dots, \pi_{n-1,m}$ of $\{1, 2, \dots, n\}$ (Only VR-FedCSO)
- 3: **for** epochs $t = 0, 1, \dots, T - 1$ **do**
- 4: **for** $m = 1, \dots, M$ locally in parallel **do**
- 5: $x_{t,m}^0 = x_t$
- 6: $y_t = x_t$
- 7: Sample permutation $\pi_{0,m}, \pi_{1,m}, \dots, \pi_{n-1,m}$ of $\{1, 2, \dots, n\}$ (Only D-VR-FedCRR)
- 8: **for** $i = 0, 1, \dots, n - 1$ **do**
- 9: $g(x_{t,m}^i, y_t) = \nabla f_{\pi_i, m}(x_{t,m}^i) - \nabla f_{\pi_i, m}(y_t) + \frac{1}{n} \nabla F_m(y_t)$
- 10: $x_{t,m}^{i+1} = x_{t,m}^i - \gamma g(x_{t,m}^i, y_t)$
- 11: **end for**
- 12: $q_{t,m} = \mathcal{C}(x_{t,m}^n - h_{t,m})$
- 13: $h_{t+1,m} = h_{t,m} + \alpha q_{t,m}$
- 14: **end for**
- 15: $x_{t+1} = (1 - \eta)x_t + \eta \frac{1}{M} \sum_{m=1}^M (q_{t,m} + h_{t,m});$
- 16: **end for**

We have the same second term, which depends on the sum of local gradients and local variances. Also, the linear rate is slightly worse since we can use any compression operator, and we also need to learn shifts. However, the main advantage of this method is the possibility of using any compression parameter ω .

6. Double Variance Reduced Compressed Federated Random Reshuffling

This section proposes another variance reduction mechanism to eliminate local variances caused by the method's stochasticity. To achieve this goal, we need to use inner product reformulation introduced by Malinovsky et al. (2021). We can get an equivalent form of the local function. Let $a_i, \dots, a_n \in \mathbb{R}^d$ are vectors that sum to zero $\sum_{i=1}^n a_i = 0$:

$$F_m(x) = \sum_{i=1}^n (f_{i,m} + \langle a_i, x \rangle) = \sum_{i=1}^n \tilde{f}_{i,m}. \quad (2)$$

Let us consider the following gradient estimate:

$$g(x_t^i, y_t) = \nabla f_{\pi_i, m}(x_t^i) - \nabla f_{\pi_i, m}(y_t) + \frac{1}{n} \nabla F_m(y_t).$$

Obviously, the sum of these vectors is equal to zero:

$$\sum_{i=1}^n a_i = - \sum_{i=1}^n \nabla f_{\pi_i, m}(y_t) + \frac{1}{n} \sum_{i=1}^n \nabla F_m(y_t) = 0.$$

Now we are ready to formulate the theorem of convergence guarantees.

Theorem 4. *Suppose that Assumption 2 and Assumption 1 hold. Then provided the stepsize satisfies $\gamma \leq \frac{1}{2L} \sqrt{\frac{\mu}{nL}}$, $\alpha \leq \frac{1}{\omega+1}$, $\eta \leq \min\left(1, \frac{(1-(1-\gamma\mu)^{\frac{n}{2}})M}{12\omega(1-\gamma\mu)^{\frac{n}{2}}}\right)$ and $\frac{1}{2} \leq (1 - \gamma\mu)^{\frac{n}{2}} (1 - (1 - \gamma\mu)^{\frac{n}{2}})$, the iterates generated by FedCRR-VR-2 or FedCSO-VR-2 (Algorithm 3) satisfy*

$$\mathbb{E}\Psi_T \leq \left(1 - \frac{1}{2} \min(\alpha, \eta(1 - (1 - \gamma\mu)^{\frac{n}{2}}))\right)^T \Psi_0 + \frac{2\left(\alpha + \eta + \frac{2\eta^2\omega}{M}\right)\gamma^3 L \sum_{m=1}^M (\|\nabla F_m(x_*)\|^2)}{M \min(\alpha, \eta(1 - (1 - \gamma\mu)^{\frac{n}{2}}))},$$

where Lyapunov function is defined as $\Psi_t = \|x_t - x_*\|^2 + \frac{4\eta^2\omega}{\alpha M} \frac{1}{M} \sum_{m=1}^M \|h_{t,m} - x_{*,m}^n\|^2$.

We need to use smaller stepsize since we applied variance reduction mechanism. However, we managed to vanish sum of local variances. The next theorem shows the communication complexity of Algorithm 3.

Corollary 3. *Let the assumptions in the Theorem 3 hold. Then the communication complexity of Algorithm 3 is*

$$T = \mathcal{O}\left(\left(\frac{(\omega+1)\left(1 - \frac{1}{\kappa\sqrt{\kappa n}}\right)^{\frac{n}{2}}}{\left(1 - (1 - \frac{1}{\kappa\sqrt{\kappa n}})\right)^{\frac{n}{2}}}\right)^2 + \frac{\sqrt{\kappa}}{\mu\sqrt{\varepsilon}} \Delta'\right) \log\left(\frac{1}{\varepsilon}\right),$$

where $\Delta' = \frac{1}{M} \sum_{m=1}^M (\|\nabla F_m(x_*)\|)$.

7. Experiments

Model. In our experiments we solve the regularized ridge regression problem, which has the form 1 with $f_{im}(x) = \frac{1}{2} \|A_{i,:}^m x - y_i^m\|^2 + \frac{\lambda}{2} \|x\|^2$, where $A^m \in \mathbb{R}^{n \times d}$, $y^m \in \mathbb{R}^n$ and $\lambda > 0$ is regularization parameter. Consider concatenated matrix $A \in \mathbb{R}^{mn \times d}$. This problem satisfies Assumption 1 for $L = \max_i \|A_{i,:}\|^2 + \lambda$ and $\mu = \frac{\rho_{\min}(A^T A)}{n} + \lambda$, where ρ_{\min} is the smallest eigenvalue. In our experiments we set $\lambda = \frac{1}{n}$. In all plots x -axis is the number of communicated bits, and y -axis is the squared norm of difference between current iterate and solution.

Hardware and software. We use real datasets from open LIBSVM corpus (Chang & Lin, 2011) (Modified BSD License www.csie.ntu.edu.tw/~cjlin/libsvm/) and synthetic datasets from scikit-learn.datasets (Pedregosa et al., 2011) (BSD License <https://scikit-learn.org>). We implemented all algorithms in Python. All methods were evaluated on a computer with an Intel(R) Xeon(R) Gold 6146 CPU at 3.20GHz, having 24 cores. You can find more details and additional experiments in supplementary materials.

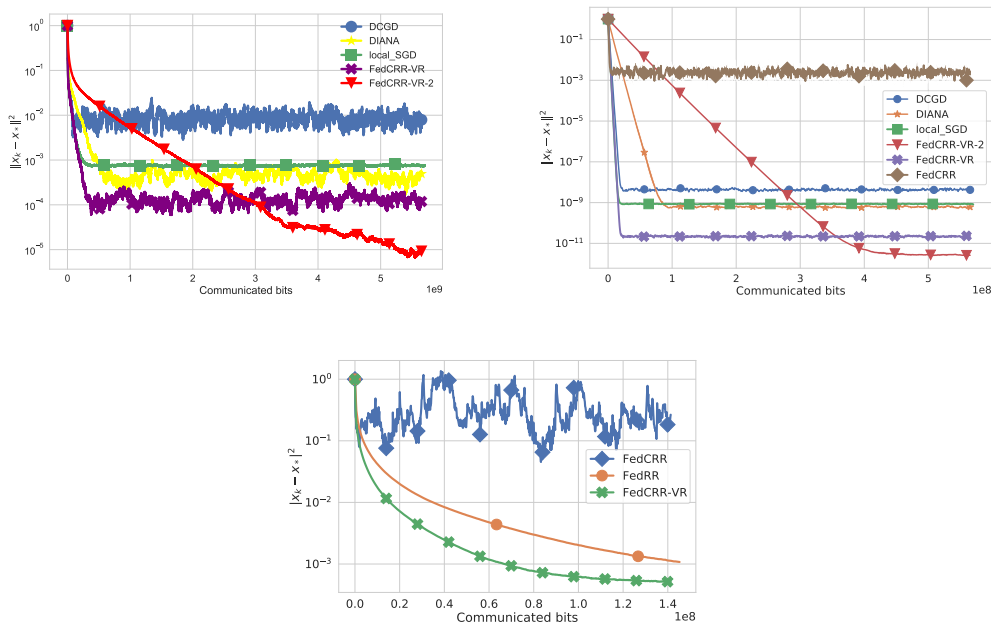


Figure 1. Comparison of distributed methods and versions of Compressed Federated Random Reshuffling: FedCRR, FedCRR-VR, FedCRR-VR-2 on synthetic datasets with the different data points. In order to have fair competition, we used the same parameter ω for all methods with compression.

Results. We have a very tight match between our theory and the numerical results. As we can see, Compressed Federated Random Reshuffling cannot get appropriate accuracy since we have a huge compression variance term. It means that this method can be used only if the required accuracy is not high. However, we can see that variance-reduced methods show better convergence. While FedRR-VR has the same linear rate, the solution’s neighbourhood is smaller in comparison to other methods. FedRR-VR-2 has a slower linear rate because of stepsize requirements, but the neighbourhood of the solution is the smallest and allows to get a much better solution of the problem.

8. Conclusion

In this work, we propose three new algorithms: Compressed Federated Random Reshuffling and two variance-reduced variants. These methods are first-of-its-kind algorithms that include three popular approaches: periodic aggregation, compressed updates and sampling without replacement. Moreover, we sequentially applied variance reduction mechanisms for compression and then for Random Reshuffling. We provide the first analysis under general assumptions. Experimental results confirm our theoretical findings. Thus, we gain a deeper theoretical understanding of how these algorithms work and hope that this will inspire researchers to develop further and analyze methods for Federated learning. In future work, we desire to get rid of the client drift term in

the neighbourhood of the solution and get an algorithm that will converge linearly to the exact solution. We also want to allow a method to have partial participation of clients since it is essential for a Federated Learning setting. We also believe that our theoretical and practical results can be applied to other aspects of machine learning and federated learning, leading to improvements in current and future applications.

References

- Aji, A. F. and Heafield, K. Sparse communication for distributed gradient descent. *arXiv preprint arXiv:1704.05021*, 2017.
- Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. Qsgd: Communication-efficient sgd via gradient quantization and encoding. *Advances in Neural Information Processing Systems*, 30:1709–1720, 2017.
- Alistarh, D., Hoeffler, T., Johansson, M., Khirirat, S., Konstantinov, N., and Renggli, C. The convergence of sparsified gradient methods. *arXiv preprint arXiv:1809.10505*, 2018.
- Basu, D., Data, D., Karakus, C., and Diggavi, S. Qsparse-local-sgd: Distributed sgd with quantization, sparsification, and local computations. *IEEE Journal on Selected Areas in Information Theory*, 1:217–226, 2020.
- Bengio, Y. Practical recommendations for gradient-based training of deep architectures. *Neural Networks: Tricks of the Trade*, pp. 437–478, 2012. ISSN 1611-3349. doi: 10.1007/978-3-642-35289-8_26.
- Bernstein, J., Wang, Y.-X., Azizadenesheli, K., and Anandkumar, A. signsgd: Compressed optimisation for non-convex problems. In *International Conference on Machine Learning*, pp. 560–569. PMLR, 2018.
- Bottou, L. Curiously fast convergence of some stochastic gradient descent algorithms. 2009.
- Chang, C.-C. and Lin, C.-J. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):1–27, 2011.
- Chraïbi, S., Khaled, A., Kovalev, D., Richtárik, P., Salim, A., and Takáč, M. Distributed fixed point methods with compressed iterates. *arXiv preprint arXiv:1912.09925*, 2019.
- Defazio, A., Bach, F., and Lacoste-Julien, S. SAGA : A fast incremental gradient method with support for non-strongly convex composite objectives. *arXiv preprint arXiv:1407.0202*, 2014a.
- Defazio, A., Domke, J., and Caetano. Finito: A faster, permutable incremental gradient method for big data problems. In Xing, E. P. and Jebara, T. (eds.), *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 1125–1133, Beijing, China, 22–24 Jun 2014b. PMLR.
- Gorbunov, E., Hanzely, F., and Richtárik, P. A unified theory of sgd: Variance reduction, sampling, quantization and coordinate descent. In *International Conference on Artificial Intelligence and Statistics*, pp. 680–690. PMLR, 2020.
- Gower, R. M., Loizou, N., Qian, X., Sailanbayev, A., Shulgin, E., and Richtárik, P. SGD: General analysis and improved rates. In *International Conference on Machine Learning*, pp. 5200–5209. PMLR, 2019.
- Gürbüzbalaban, M., Ozdaglar, A., and Parrilo, P. A. Why random reshuffling beats stochastic gradient descent. *Mathematical Programming*, pp. 1–36, 2019.
- Haddadpour, F., Kamani, M. M., Mokhtari, A., and Mahdavi, M. Federated learning with compression: Unified analysis and sharp guarantees. In *International Conference on Artificial Intelligence and Statistics*, pp. 2350–2358. PMLR, 2021.
- Haochen, J. and Sra, S. Random shuffling beats SGD after finite epochs. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning Research*, pp. 2624–2633, Long Beach, California, USA, 09–15 Jun 2019. PMLR.
- Horvath, S., Ho, C.-Y., Horvath, L., Sahu, A. N., Canini, M., and Richtárik, P. Natural compression for distributed deep learning. *arXiv preprint arXiv:1905.10988*, 2019.
- Johnson, R. and Zhang, T. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in Neural Information Processing Systems*, 26:315–323, 2013.
- Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S. J., Stich, S. U., and Suresh, A. T. SCAFFOLD: Stochastic Controlled Averaging for On-Device Federated Learning. *arXiv preprint arXiv:1910.06378*, 2019.
- Khaled, A. and Richtárik, P. Gradient descent with compressed iterates. *arXiv preprint arXiv:1909.04716*, 2019.
- Khaled, A., Mishchenko, K., and Richtárik, P. First analysis of local gd on heterogeneous data. *arXiv preprint arXiv:1909.04715*, 2019.
- Khaled, A., Mishchenko, K., and Richtárik, P. Tighter theory for local sgd on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pp. 4519–4529. PMLR, 2020.
- Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.
- Kovalev, D., Horváth, S., and Richtárik, P. Don't jump through hoops and remove those loops: SVRG and katyusha are better without the outer loop. In *Algorithmic Learning Theory*, pp. 451–467. PMLR, 2020.

- Lin, T., Stich, S. U., Patel, K. K., and Jaggi, M. Don't use large mini-batches, use local sgd. *arXiv preprint arXiv:1808.07217*, 2018.
- Lin, Y., Han, S., Mao, H., Wang, Y., and Dally, W. J. Deep gradient compression: Reducing the communication bandwidth for distributed training. *arXiv preprint arXiv:1712.01887*, 2017.
- Malinovsky, G., Sailanbayev, A., and Richtárik, P. Random reshuffling with variance reduction: New analysis and better rates. *arXiv preprint arXiv:2104.09342*, 2021.
- McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pp. 1273–1282. PMLR, 2017.
- Mishchenko, K., Gorbunov, E., Takáč, M., and Richtárik, P. Distributed learning with compressed gradient differences. *arXiv preprint arXiv:1901.09269*, 2019.
- Mishchenko, K., Khaled, A., and Richtárik, P. Random reshuffling: Simple analysis with vast improvements. *Advances in Neural Information Processing Systems*, 33, 2020.
- Mishchenko, K., Khaled, A., and Richtárik, P. Proximal and federated random reshuffling. *arXiv preprint arXiv:2102.06704*, 2021.
- Mokhtari, A., Gürbüzbalaban, M., and Ribeiro, A. Surpassing gradient descent provably: A cyclic incremental method with linear convergence rate. *SIAM Journal on Optimization*, 28(2):1420–1447, 2018.
- Nagaraj, D., Jain, P., and Netrapalli, P. SGD without replacement: sharper rates for general smooth convex functions. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4703–4711, Long Beach, California, USA, 09–15 Jun 2019. PMLR.
- Park, Y. and Ryu, E. K. Linear convergence of cyclic SAGA. *Optimization Letters*, 14(6):1583–1598, 2020.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- Rajput, S., Gupta, A., and Papailiopoulos, D. Closing the convergence gap of SGD without replacement. *arXiv preprint arXiv:2002.10400*, 2020.
- Ramezani-Kebrya, A., Faghri, F., and Roy, D. M. Nuqsgd: Improved communication efficiency for data-parallel sgd via nonuniform quantization. *arXiv preprint arXiv:1908.06077*, 2019.
- Recht, B. and Ré, C. Toward a noncommutative arithmetic-geometric mean inequality: conjectures, case-studies, and consequences. In *Conference on Learning Theory*, pp. 11–1. JMLR Workshop and Conference Proceedings, 2012.
- Recht, B. and Ré, C. Parallel stochastic gradient algorithms for large-scale matrix completion. *Mathematical Programming Computation*, 5(2):201–226, 2013.
- Robbins, H. and Monro, S. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.
- Roux, N. L., Schmidt, M., and Bach, F. A stochastic gradient method with an exponential convergence rate for finite training sets. *arXiv preprint arXiv:1202.6258*, 2012.
- Safran, I. and Shamir, O. How good is SGD with random shuffling? In *Conference on Learning Theory*, pp. 3250–3284. PMLR, 2020.
- Shamir, O., Srebro, N., and Zhang, T. Communication-efficient distributed optimization using an approximate newton-type method. In *International conference on machine learning*, pp. 1000–1008. PMLR, 2014.
- Stich, S. U. Local sgd converges fast and communicates little. *arXiv preprint arXiv:1805.09767*, 2018.
- Vogels, T., Karimireddy, S. P., and Jaggi, M. Powersgd: Practical low-rank gradient compression for distributed optimization. *arXiv preprint arXiv:1905.13727*, 2019.
- Wangni, J., Wang, J., Liu, J., and Zhang, T. Gradient sparsification for communication-efficient distributed optimization. *arXiv preprint arXiv:1710.09854*, 2017.
- Wu, J., Huang, W., Huang, J., and Zhang, T. Error compensated quantized sgd and its applications to large-scale distributed optimization. In *International Conference on Machine Learning*, pp. 5325–5333. PMLR, 2018.
- Ying, B., Yuan, K., Vlaski, S., and Sayed, A. H. Stochastic learning under random reshuffling with constant step-sizes. In *IEEE Transactions on Signal Processing*, volume 67, pp. 474–489, 2019.

Supplementary materials

A. Basic Facts

Proposition 1. Let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be continuously differentiable and let $L \geq 0$. Then the following statements are equivalent:

- f is L -smooth
- $2D_f(x, y) \leq L\|x - y\|^2$ for all $x, y \in \mathbb{R}^d$
- $\langle \nabla f(x) - \nabla f(y), x - y \rangle \leq L\|x - y\|^2$ for all $x, y \in \mathbb{R}^d$

Proposition 2. Let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be continuously differentiable and let $\mu \geq 0$. Then the following statements are equivalent:

- f is μ -strongly convex
- $2D_f(x, y) \geq \mu\|x - y\|^2$ for all $x, y \in \mathbb{R}^d$
- $\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \mu\|x - y\|^2$ for all $x, y \in \mathbb{R}^d$

Note that the $\mu = 0$ case reduces to convexity.

Proposition 3. Let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be continuously differentiable and $L > 0$. Then the following statements are equivalent:

- f is convex and L -smooth
- $0 \leq 2D_f(x, y) \leq L\|x - y\|^2$ for all $x, y \in \mathbb{R}^d$
- $\frac{1}{L}\|\nabla f(x) - \nabla f(y)\|^2 \leq 2D_f(x, y)$ for all $x, y \in \mathbb{R}^d$
- $\frac{1}{L}\|\nabla f(x) - \nabla f(y)\|^2 \leq \langle \nabla f(x) - \nabla f(y), x - y \rangle$ for all $x, y \in \mathbb{R}^d$

Proposition 4 (Jensen's inequality). Let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be a convex function, $x_1, \dots, x_m \in \mathbb{R}^d$ and $\lambda_1, \dots, \lambda_m$ be nonnegative real numbers adding up to 1. Then

$$f\left(\sum_{i=1}^m \lambda_i x_i\right) \leq \sum_{i=1}^m \lambda_i f(x_i)$$

Proposition 5. For all $a, b \in \mathbb{R}^d$ and $t > 0$ the following inequalities holds:

$$\begin{aligned} \langle a, b \rangle &\leq \frac{\|a\|^2}{2t} + \frac{t\|b\|^2}{2} \\ \|a + b\|^2 &\leq 2\|a\|^2 + 2\|b\|^2 \\ \frac{1}{2}\|a\|^2 - \|b\|^2 &\leq \|a + b\|^2 \end{aligned}$$

B. General lemmas

B.1. Proposition 1

We need to prove a basic fact which will be used later.

Proposition 6. Let us consider

$$x_{*,m}^n = x_* - \gamma \sum_{i=0}^{n-1} \nabla f_{\pi_i, m}(x_*),$$

then

$$\frac{1}{M} \sum_{m=1}^M x_{*,m}^n = x_*.$$

Proof. We start from the definition:

$$\begin{aligned}
 \frac{1}{M} \sum_{m=1}^M x_{*,m}^n &= \frac{1}{M} \sum_{m=1}^M \left(x_* - \gamma \sum_{i=0}^{n-1} \nabla f_{\pi_i, m}(x_*) \right) \\
 &= \frac{1}{M} \sum_{m=1}^M x_* - \frac{1}{M} \sum_{m=1}^M \sum_{i=0}^{n-1} \nabla f_{\pi_i, m}(x_*) \\
 &= x_* - \nabla f(x_*) \\
 &= x_*.
 \end{aligned}$$

□

B.2. Proof of Theorem 1

For completeness we include the proof of important theorem introduced in [Mishchenko et al. \(2020\)](#).

Proof.

$$\begin{aligned}
 \mathbb{E} [D_{f_{\pi_i}}(x_*^i, x_*)] &\leq \mathbb{E} \left[\frac{L}{2} \|x_*^i - x_*\|^2 \right] \leq \frac{L_{\max}}{2} \mathbb{E} \left[\|x_*^i - x_*\|^2 \right] \\
 &= \frac{\gamma^2 L_{\max}}{2} \mathbb{E} \left[\left\| \sum_{j=0}^{i-1} \nabla f_{\pi_j}(x_*) \right\|^2 \right] \\
 &= \frac{\gamma^2 L_{\max} i^2}{2} \mathbb{E} \left[\left\| \frac{1}{i} \sum_{j=0}^{i-1} \nabla f_{\pi_j}(x_*) \right\|^2 \right] \\
 &= \frac{\gamma^2 L_{\max} i^2}{2} \mathbb{E} \left[\|\bar{X}_\pi\|^2 \right],
 \end{aligned}$$

where $\bar{X}_\pi = \frac{1}{j} \sum_{j=0}^{i-1} X_{\pi_j}$ with $X_j \stackrel{\text{def}}{=} \nabla f_j(x_*)$ for $j = 1, 2, \dots, n$. Since $\bar{X} = \nabla f(x_*)$, by applying Lemma 1 in [\(Mishchenko et al., 2020\)](#).

$$\mathbb{E} \left[\|\bar{X}_\pi\|^2 \right] = \|\bar{X}\|^2 + \mathbb{E} \left[\|\bar{X}_\pi - \bar{X}\|^2 \right] = \|\nabla f(x_*)\|^2 + \frac{n-i}{i(n-1)} \sigma_*^2.$$

It remains to combine both terms and use the bounds $i^2 \leq n^2$ and $i(n-i) \leq \frac{n(n-1)}{2}$, which holds for all $i \in \{1, 2, \dots, n-1\}$, and divide both sides of the resulting inequality by γ^2 . □

B.3. Proof of Lemma 1

Proof. We start from Theorem 1. Then for reformulated problem we have

$$n\sigma_*^2 \stackrel{\text{def}}{=} \sum_{i=1}^n \|\nabla f_i(x_*) - \nabla f(x_*)\|^2 = \sum_{i=1}^n \sum_{m=1}^M \left\| \nabla f_{mi}(x_*) - \frac{1}{n} \nabla F_m(x_*) \right\|^2.$$

For inner sum we have a bound from [Mishchenko et al. \(2021\)](#):

$$\sum_{i=1}^n \left\| \nabla f_{mi}(x_*) - \frac{1}{n} \nabla F_m(x_*) \right\|^2 \leq n\sigma_{m,*}^2 + \|\nabla F_m(x_*)\|^2.$$

Also, we have

$$n^2 \|\nabla f(x_*)\|^2 = n^2 \left\| \frac{1}{n} \sum_{i=1}^n \nabla f_i(x_*) \right\|^2 = \sum_{m=1}^M \left\| \sum_{i=1}^n \nabla f_{mi}(x_*) \right\|^2 = \sum_{m=1}^M \|\nabla F_m(x_*)\|^2.$$

Plugging the last two inequalities back inside the first bound on σ_{rad}^2 , we get the lemma's statement. □

C. Analysis of Algorithm 1

C.1. Proof of Theorem 1

Proof. We start from conditional expectation

$$\begin{aligned}
 \mathbb{E} [\|x_{t+1} - x_*\|^2 \mid x_{t,m}^n] &= \mathbb{E} \left[\left\| \frac{1}{M} \sum_{m=1}^M C(x_{t,m}^n) - x_* \right\|^2 \mid x_{t,m}^n \right] \\
 &\leq \mathbb{E} \left\| \frac{1}{M} \sum_{m=1}^M C(x_{t,m}^n) - \frac{1}{M} \sum_{m=1}^M x_{t,m}^n \right\|^2 + \left\| \frac{1}{M} \sum_{m=1}^M x_{t,m}^n - x_* \right\|^2 \\
 &\leq \frac{\omega}{M^2} \sum_{m=1}^M \|x_{t,m}^n\|^2 + \frac{1}{M} \sum_{m=1}^M \|x_{t,m}^n - x_{*,m}^n\|^2 \\
 &\leq \frac{2\omega}{M^2} \sum_{m=1}^M \|x_{t,m}^n - x_{*,m}^n\|^2 + \frac{1}{M} \sum_{m=1}^M \|x_{t,m}^n - x_{*,m}^n\|^2 + \frac{2\omega}{M^2} \sum_{m=1}^M \|x_{*,m}^n\|^2 \\
 &\leq (1 - \gamma\mu)^n \left(1 + \frac{2\omega}{M} \right) \|x_t - x_*\|^2 + \frac{2\omega}{M} \frac{1}{M} \sum_{m=1}^M \|x_{*,m}^n\|^2 \\
 &\quad + 2 \left(1 + \frac{2\omega}{M} \right) \gamma^3 \sigma_{rad}^2 \left(\sum_{j=0}^{n-1} (1 - \gamma\mu)^j \right).
 \end{aligned}$$

Using tower property we get

$$\mathbb{E} [\|x_{t+1} - x_*\|^2] = \mathbb{E} [\mathbb{E} [\|x_{t+1} - x_*\|^2 \mid x_{t,m}^n]].$$

Utilizing this property we have

$$\begin{aligned}
 \mathbb{E} [\|x_{t+1} - x_*\|^2] &\leq (1 - \gamma\mu)^n \left(1 + \frac{2\omega}{M} \right) \mathbb{E} [\|x_t - x_*\|^2] + \frac{2\omega}{M} \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|x_{*,m}^n\|^2 \\
 &\quad + 2 \left(1 + \frac{2\omega}{M} \right) \gamma^3 \sigma_{rad}^2 \left(\sum_{j=0}^{n-1} (1 - \gamma\mu)^j \right).
 \end{aligned}$$

Unrolling this recursion we get

$$\begin{aligned}
 \mathbb{E} [\|x_{t+1} - x_*\|^2] &\leq \left((1 - \gamma\mu)^n \left(1 + \frac{2\omega}{M} \right) \right)^T \|x_0 - x_*\|^2 \\
 &\quad + \sum_{i=0}^{T-1} \left((1 - \gamma\mu)^n \left(1 + \frac{2\omega}{M} \right) \right)^i \frac{2\omega}{M} \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|x_{*,m}^n\|^2 \\
 &\quad + 2 \sum_{i=0}^{T-1} \left((1 - \gamma\mu)^n \left(1 + \frac{2\omega}{M} \right) \right)^i \left(1 + \frac{2\omega}{M} \right) \gamma^3 \sigma_{rad}^2 \left(\sum_{j=0}^{n-1} (1 - \gamma\mu)^j \right).
 \end{aligned}$$

Using assumption of compression operator we have

$$(1 - \gamma\mu)^n \left(1 + \frac{2\omega}{M} \right) \leq (1 - \gamma\mu)^{\frac{n}{2}}.$$

Also let us look at last term:

$$\begin{aligned} \left(1 + \frac{2\omega}{M}\right) \left(\sum_{i=0}^{T-1} (1 - \gamma\mu)^i\right) &\leq \sum_{i=0}^{T-1} (1 - \gamma\mu)^i \left(1 + \frac{2\omega}{M}\right)^i \\ &\leq \sum_{j=0}^{T-1} \left((1 - \gamma\mu) \left(1 + \frac{2\omega}{M}\right)\right)^j \\ &\leq \sum_{i=0}^{T-1} (1 - \gamma\mu)^{\frac{n_i}{2}}. \end{aligned}$$

Moreover, we have this bound for geometric sequence:

$$\left(\sum_{i=0}^{T-1} (1 - \gamma\mu)^{\frac{n_i}{2}}\right) \left(\sum_{j=0}^{n-1} (1 - \gamma\mu)^j\right) \leq \sum_{i=0}^{T-1} \sum_{j=0}^{n-1} (1 - \gamma\mu)^{\frac{n_i}{2} + j} \leq \frac{1}{\gamma\mu}.$$

The same bound we have for the second sum:

$$\left(\sum_{i=0}^{T-1} (1 - \gamma\mu)^{\frac{n_i}{2}}\right) \leq \frac{1}{\gamma\mu}.$$

Finally, we have the following:

$$\mathbb{E} [\|x_{t+1} - x_*\|^2] \leq (1 - \gamma\mu)^{\frac{n_T}{2}} \|x_0 - x_*\|^2 + \frac{2}{\mu} \gamma^2 \sigma_{rad}^2 + \frac{2\omega}{M} \frac{1}{\gamma\mu} \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|x_{*,m}^n\|^2.$$

Using Lemma we have

$$\begin{aligned} \mathbb{E} [\|x_{t+1} - x_*\|^2] &\leq (1 - \gamma\mu)^{\frac{n_T}{2}} \|x_0 - x_*\|^2 + \frac{2}{\mu} \gamma^2 L_{\max} \frac{1}{M} \sum_{m=1}^M \left(\|F_m(x_*)\|^2 + \frac{n}{4} \sigma_{*,m}^2\right) \\ &\quad + \frac{2\omega}{M} \frac{1}{\gamma\mu} \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|x_{*,m}^n\|^2. \end{aligned}$$

□

D. Analysis of Algorithm 2 and Algorithm 3

D.1. Lemma 2

For Algorithm 2 and Algorithm 3 the following inequality holds:

$$\mathbb{E} [\|x_{t+1} - x_*\|^2 \mid x_t, h_{t,m}] \leq \frac{\eta^2}{M^2} \omega \sum_{m=1}^M \|x_{t,m}^n - h_{t,m}\|^2 + (1 - \eta) \|x_t - x_*\|^2 + \eta \frac{1}{M} \sum_{m=1}^M \|x_{t,m}^n - x_{*,m}^n\|^2$$

Proof. Let us use property of compression operator:

$$\begin{aligned}
 \mathbb{E} [\|x_{t+1} - x_*\|^2 \mid x_t, h_{t,m}] &= \mathbb{E} \left[\left\| (1-\eta)x_t + \eta \frac{1}{M} \sum_{m=1}^M (C(x_{t,m}^n - h_{t,m}) + h_{t,m}) \right\|^2 \mid x_t, h_{t,m} \right] \\
 &= \mathbb{E} \left[\left\| \eta \frac{1}{M} \sum_{m=1}^M C(x_{t,m}^n - h_{t,m}) - \eta \frac{1}{M} \sum_{m=1}^M (x_{t,m}^n - h_{t,m}) \right\|^2 \mid x_t, h_{t,m} \right] \\
 &\quad + \left\| (1-\eta)x_t + \eta \frac{1}{M} \sum_{m=1}^M x_{t,m}^n - x_* \right\|^2 \\
 &\leq \frac{\eta^2}{M^2} \omega \sum_{m=1}^M \|x_{t,m}^n - h_{t,m}\|^2 + (1-\eta)\|x_t - x_*\|^2 + \eta \frac{1}{M} \sum_{m=1}^M \|x_{t,m}^n - x_{*,m}^n\|^2.
 \end{aligned}$$

□

D.2. Lemma 3

For Algorithm 2 and Algorithm 3 the following inequality holds:

$$\mathbb{E} [\|h_{t+1,m} - x_{*,m}^n\|^2] \leq (1-\alpha)\mathbb{E}\|h_{t,m} - x_{*,m}^n\|^2 + \alpha\mathbb{E}\|x_{t,m}^n - x_{*,m}^n\|^2.$$

Proof. Let us start from conditional expectation:

$$\begin{aligned}
 \mathbb{E} [\|h_{t+1,m} - x_{*,m}^n\|^2 \mid x_{t,m}^n, h_{t,m}] &= \mathbb{E} [\|h_{t,m} + \alpha q_{t,m} - x_{*,m}^n\|^2 \mid x_{t,m}^n, h_{t,m}] \\
 &\leq \|h_{t,m} - x_{*,m}^n\|^2 + 2\alpha \langle h_{t,m} - x_{*,m}^n, \mathbb{E}[q_{t,m}] \rangle + \alpha^2 \mathbb{E} [\|q_{t,m}\|^2] \\
 &\leq \|h_{t,m} - x_{*,m}^n\|^2 + 2\alpha \langle h_{t,m} - x_{*,m}^n, x_{t,m}^n - h_{t,m} \rangle \\
 &\quad + \alpha^2(\omega + 1)\|x_{t,m}^n - h_{t,m}\|^2 \\
 &\leq \|h_{t,m} - x_{*,m}^n\|^2 + 2\alpha \langle h_{t,m} - x_{*,m}^n, x_{t,m}^n - h_{t,m} \rangle \\
 &\quad + \alpha\|x_{t,m}^n - h_{t,m}\|^2 \\
 &= \|h_{t,m} - x_{*,m}^n\|^2 + \alpha \langle 2h_{t,m} - 2x_{*,m}^n + x_{t,m}^n - h_{t,m}, x_{t,m}^n - h_{t,m} \rangle.
 \end{aligned}$$

Let us consider last term:

$$\begin{aligned}
 &\langle 2h_{t,m} - 2x_{*,m}^n + x_{t,m}^n - h_{t,m}, x_{t,m}^n - h_{t,m} \rangle \\
 &= \langle h_{t,m} - x_{*,m}^n + x_{t,m}^n - x_{*,m}^n, x_{t,m}^n - x_{*,m}^n - (h_{t,m} - x_{*,m}^n) \rangle \\
 &= -\|h_{t,m} - x_{*,m}^n\|^2 + \|x_{t,m}^n - x_{*,m}^n\|^2.
 \end{aligned}$$

□

Using this result and previous inequality we get the following:

$$\mathbb{E} [\|h_{t+1,m} - x_{*,m}^n\|^2 \mid x_{t,m}^n, h_{t,m}] \leq (1-\alpha)\|h_{t,m} - x_{*,m}^n\|^2 + \alpha\|x_{t,m}^n - x_{*,m}^n\|^2.$$

Taking full expectation we finish the proof.

D.3. Lemma 6

For completeness we include the proof of important theorem introduced in [Mishchenko et al. \(2021\)](#). Suppose that each f_i is L -smooth and μ -strongly convex. Then the inner iterates satisfy

$$\mathbb{E} [\|x_t^{i+1} - x_*^{i+1}\|^2] \leq (1-\gamma\mu)\mathbb{E} [\|x_t^i - x_*^i\|^2] - 2\gamma(1-\gamma L)\mathbb{E} [D_{f_{\pi_i}}(x_t^i, x_*)] + 2\gamma^3\sigma_{\text{rad}}^2.$$

Proof. By definition of x_t^{i+1} and x_*^{i+1} , we have

$$\begin{aligned} \mathbb{E} \left[\|x_t^{i+1} - x_*^{i+1}\|^2 \right] &= \mathbb{E} \left[\|x_t^i - x_*^i\|^2 \right] - 2\gamma \mathbb{E} \left[\langle \nabla f_{\pi_i}(x_t^i) - \nabla f_{\pi_i}(x_*) , x_t^i - x_*^i \rangle \right] \\ &\quad + \gamma^2 \mathbb{E} \left[\|\nabla f_{\pi_i}(x_t^i) - \nabla f_{\pi_i}(x_*)\|^2 \right]. \end{aligned}$$

Note that the third term can be bounded as

$$\|\nabla f_{\pi_i}(x_t^i) - \nabla f_{\pi_i}(x_*)\|^2 \leq 2L \cdot D_{f_{\pi_i}}(x_t^i, x_*).$$

Using the three-point identity we get

$$\langle \nabla f_{\pi_i}(x_t^i) - \nabla f_{\pi_i}(x_*), x_t^i - x_*^i \rangle = D_{f_{\pi_i}}(x_*^i, x_t^i) + D_{f_{\pi_i}}(x_t^i, x_*) - D_{f_{\pi_i}}(x_*^i, x_*).$$

Combining these bounds we have

$$\begin{aligned} \mathbb{E} \left[\|x_t^{i+1} - x_*^{i+1}\|^2 \right] &\leq \mathbb{E} \left[\|x_t^i - x_*^i\|^2 \right] - 2\gamma \cdot \mathbb{E} \left[D_{f_{\pi_i}}(x_*^i, x_t^i) \right] + 2\gamma \cdot \mathbb{E} \left[D_{f_{\pi_i}}(x_t^i, x_*) \right] \\ &\quad - 2\gamma(1 - \gamma L) \mathbb{E} \left[D_{f_{\pi_i}}(x_t^i, x_*) \right]. \end{aligned}$$

Using μ -strong convexity of f_{π_i} , we derive

$$\frac{\mu}{2} \|x_t^i - x_*^i\|^2 \leq D_{f_{\pi_i}}(x_*^i, x_t^i).$$

Using definition of shuffling radius we have

$$\mathbb{E} \left[D_{f_{\pi_i}}(x_*^i, x_*) \right] \leq \max_{i=1, \dots, n-1} \mathbb{E} \left[D_{f_{\pi_i}}(x_*^i, x_*) \right] = \gamma^2 \sigma_{\text{rad}}^2.$$

Putting all bounds together we get result. □

D.4. Proof of Theorem 3

Proof. Let us define the Lyapunov function:

$$\Psi_t = \|x_t - x_*\|^2 + \frac{4\eta^2\omega}{\alpha M} \frac{1}{M} \sum_{m=1}^M \|h_{t,m} - x_{*,m}^n\|^2.$$

Now we use Lemma 2 and Lemma 3:

$$\begin{aligned} \mathbb{E} [\Psi_{t+1}] &= \mathbb{E} \|x_{t+1} - x_*\|^2 + \frac{4\eta^2\omega}{\alpha M} \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|h_{t+1,m} - x_{*,m}^n\|^2 \\ &\leq \frac{2\eta^2}{M^2} \omega \sum_{m=1}^M \mathbb{E} \|x_{t,m}^n - x_{*,m}^n\|^2 + \frac{2\eta^2}{M^2} \omega \sum_{m=1}^M \mathbb{E} \|h_{t,m} - x_{*,m}^n\|^2 + (1 - \eta) \mathbb{E} \|x_t - x_*\|^2 \\ &\quad + \eta \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|x_{t,m}^n - x_{*,m}^n\|^2 + \frac{4\eta^2\omega}{\alpha M} (1 - \alpha) \sum_{m=1}^M \mathbb{E} \|h_{t,m} - x_{*,m}^n\|^2 + \frac{4\eta^2\omega}{\alpha M} \alpha \sum_{m=1}^M \mathbb{E} \|x_{t,m}^n - x_{*,m}^n\|^2. \end{aligned}$$

Using Lemma 6 and Theorem 2 from [Mishchenko et al. \(2021\)](#) we have

$$\begin{aligned} \mathbb{E} [\Psi_{t+1}] &\leq \frac{4\eta^2\omega}{\alpha M} \left(1 - \frac{\alpha}{2}\right) \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|h_{t,m} - x_{*,m}^n\|^2 + \left(1 - \eta + \eta(1 - \gamma\mu)^n + \frac{6\eta^2\omega}{M}(1 - \gamma\mu)^n\right) \mathbb{E} \|x_t - x_*\|^2 \\ &\quad + \left(\alpha + \eta + \frac{2\eta^2\omega}{M}\right) 2\gamma^3 \sigma_{\text{rad}}^2 \left(\sum_{j=0}^{n-1} (1 - \gamma\mu)^j\right) \end{aligned}$$

Using the condition $\eta \leq \min\left(1, \frac{(1-(1-\gamma\mu)^n)M}{12\omega(1-\gamma\mu)^n}\right)$ we have

$$\mathbb{E}[\Psi_{t+1}] \leq \max\left(1 - \frac{\alpha}{2}, 1 - \frac{\eta(1-(1-\gamma\mu)^n)}{2}\right) \mathbb{E}[\Psi_t] + \left(\alpha + \eta + \frac{2\eta^2\omega}{M}\right) 2\gamma^3\sigma_{rad}^2 \left(\sum_{j=0}^{n-1} (1-\gamma\mu)^j\right)$$

Note that we have the following inequality:

$$\begin{aligned} 1 - \gamma\mu &\leq 1 - \frac{1}{2}\eta(1 - (1 - \gamma\mu)^n) \\ -\gamma\mu &\leq -\frac{1}{2}\eta(1 - (1 - \gamma\mu)^n) \\ \gamma\mu &\geq \frac{1}{2}\eta(1 - (1 - \gamma\mu)^n). \end{aligned}$$

We have it since $0 < \eta \leq 1$ and $n > 1$, so we have

$$\begin{aligned} \gamma\mu &\geq \frac{1}{2}(1 - (1 - \gamma\mu)^n) \\ \gamma\mu &\geq \frac{1}{2}(1 - (1 - \gamma\mu)) \\ 1 &\geq \frac{1}{2}. \end{aligned}$$

Unrolling this recursion finishes the proof. □

Lemma 4

For completeness we include the proof of important theorem introduced in [Malinovsky et al. \(2021\)](#). Suppose that the functions f_1, \dots, f_n are μ -strongly convex and L -smooth. Fix constant $0 < \delta < 1$. If the stepsize satisfies $\gamma \leq \frac{\delta}{L} \sqrt{\frac{\mu}{2nL}}$ and if number of functions is sufficiently big:

$$n > \log\left(\frac{1}{1-\delta^2}\right) \cdot \left(\log\left(\frac{1}{1-\gamma\mu}\right)\right)^{-1}$$

and

$$\begin{aligned} \delta^2 &\leq (1 - \gamma\mu)^{\frac{n}{2}} (1 - (1 - \gamma\mu)^{\frac{n}{2}}). \\ \mathbb{E}\left[\|x_t^n - x_*^n\|^2\right] &\leq (1 - \gamma\mu)^{\frac{n}{2}} \mathbb{E}\left[\|x_t - x_*\|^2\right], \end{aligned}$$

then we have

$$\mathbb{E}\left[\|x_t^n - x_*^n\|^2 \mid x_t\right] \leq (1 - \gamma\mu)^n \|x_t - x_*\|^2 + \frac{\gamma^3 Ln}{2} \sigma_*^2 \left(\sum_{i=0}^{n-1} (1 - \gamma\mu)^i\right)$$

Proof. We start from Theorem 1 in [Mishchenko et al. \(2020\)](#):

$$\mathbb{E}\left[\|x_t^n - x_*^n\|^2 \mid x_t\right] \leq (1 - \gamma\mu)^n \|x_t - x_*\|^2 + \frac{\gamma^3 Ln}{2} \sigma_*^2 \left(\sum_{i=0}^{n-1} (1 - \gamma\mu)^i\right)$$

Using property of geometric progression we can have an upper bound $\sum_{i=0}^{n-1} (1 - \gamma\mu)^i \leq \frac{1}{\gamma\mu}$:

$$\mathbb{E}\left[\|x_t^n - x_*^n\|^2 \mid x_t\right] \leq (1 - \gamma\mu)^n \|x_t - x_*\|^2 + \frac{\gamma^2 Ln}{2\mu} \sigma_*^2.$$

Using Lemma 1 in Malinovsky et al. (2021) we get

$$\mathbb{E} \left[\|x_t^n - x_*^n\|^2 \mid x_t \right] \leq \left((1 - \gamma\mu)^n + \frac{2\gamma^2 L^3 n}{\mu} \right) \|x_t - x_*\|^2.$$

Let us use $\gamma \leq \frac{\delta}{L} \sqrt{\frac{\mu}{2nL}}$. To get convergence we need

$$(1 - \gamma\mu)^n + \delta^2 < 1.$$

This leads to the following inequality:

$$n > \log \left(\frac{1}{1 - \delta^2} \right) \cdot \left(\log \left(\frac{1}{1 - \gamma\mu} \right) \right)^{-1}.$$

Also assume

$$\delta^2 \leq (1 - \gamma\mu)^{\frac{n}{2}} (1 - (1 - \gamma\mu)^{\frac{n}{2}}).$$

Putting this into bound finishes the proof. \square

D.5. Lemma 5

For completeness we include the proof of important lemma introduced in Malinovsky et al. (2021).

Assume that each f_i is L -smooth and convex. If we apply the linear perturbation reformulation 6, then the variance of reformulated problem satisfies the following inequality:

$$\tilde{\sigma}_*^2 \leq 4L^2 \|y_t - x_*\|^2.$$

Proof.

$$\tilde{\sigma}_*^2 = \frac{1}{n} \sum_{i=1}^n \|\nabla f_i(x_*) - \nabla f_i(y_t) + \nabla f(y_t) - \nabla f(x_*)\|^2$$

Using Young's inequality we have

$$\begin{aligned} \tilde{\sigma}_*^2 &\leq \frac{1}{n} \sum_{i=1}^n \left(2\|\nabla f_i(y_t) - \nabla f_i(x_*)\|^2 + 2\|\nabla f(y_t) - \nabla f(x_*)\|^2 \right) \\ &\leq \frac{1}{n} \sum_{i=1}^n 4L_i D_{f_i}(y_t, x_*) + \frac{1}{n} \sum_{i=1}^n 4LD_f(y_t, x_*) \\ &\leq 4LD_f(y_t, x_*) + 4LD_f(y_t, x_*) \\ &= 8LD_f(y_t, x_*) \\ &\leq 4L^2 \|y_t - x_*\|^2 \end{aligned}$$

\square

D.6. Proof of Theorem 4

The proof is similar to the proof of 3.

Proof. Let us define the Lyapunov function:

$$\Psi_t = \|x_t - x_*\|^2 + \frac{4\eta^2\omega}{\alpha M} \frac{1}{M} \sum_{m=1}^M \|h_{t,m} - x_{*,m}^n\|^2.$$

Now we use Lemma 2 and Lemma 3:

$$\begin{aligned}
 \mathbb{E} [\Psi_{t+1}] &= \mathbb{E} \|x_{t+1} - x_*\|^2 + \frac{4\eta^2\omega}{\alpha M} \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|h_{t+1,m} - x_{*,m}^n\|^2 \\
 &\leq \frac{2\eta^2}{M^2} \omega \sum_{m=1}^M \mathbb{E} \|x_{t,m}^n - x_{*,m}^n\|^2 + \frac{2\eta^2}{M^2} \omega \sum_{m=1}^M \mathbb{E} \|h_{t,m} - x_{*,m}^n\|^2 + (1-\eta) \mathbb{E} \|x_t - x_*\|^2 \\
 &\quad + \eta \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|x_{t,m}^n - x_{*,m}^n\|^2 + \frac{4\eta^2\omega}{\alpha M} (1-\alpha) \sum_{m=1}^M \mathbb{E} \|h_{t,m} - x_{*,m}^n\|^2 + \frac{4\eta^2\omega}{\alpha M} \alpha \sum_{m=1}^M \mathbb{E} \|x_{t,m}^n - x_{*,m}^n\|^2.
 \end{aligned}$$

Using Lemma 5 and Theorem 3 from Malinovsky et al. (2021) we have

$$\begin{aligned}
 \mathbb{E} [\Psi_{t+1}] &\leq \frac{4\eta^2\omega}{\alpha M} \left(1 - \frac{\alpha}{2}\right) \frac{1}{M} \sum_{m=1}^M \mathbb{E} \|h_{t,m} - x_{*,m}^n\|^2 + \left(1 - \eta + \eta(1 - \gamma\mu)^{\frac{n}{2}} + \frac{6\eta^2\omega}{M} (1 - \gamma\mu)^{\frac{n}{2}}\right) \mathbb{E} \|x_t - x_*\|^2 \\
 &\quad + \left(\alpha + \eta + \frac{2\eta^2\omega}{M}\right) 2\gamma^3 L \sum_{m=1}^M \|\nabla F_m(x_*)\|^2 \left(\sum_{j=0}^{n-1} (1 - \gamma\mu)^j\right)
 \end{aligned}$$

Using the condition $\eta \leq \min\left(1, \frac{(1 - (1 - \gamma\mu)^{\frac{n}{2}})M}{12\omega(1 - \gamma\mu)^{\frac{n}{2}}}\right)$ we have

$$\begin{aligned}
 \mathbb{E} [\Psi_{t+1}] &\leq \max\left(1 - \frac{\alpha}{2}, 1 - \frac{\eta(1 - (1 - \gamma\mu)^{\frac{n}{2}})}{2}\right) \mathbb{E} [\Psi_t] \\
 &\quad + \left(\alpha + \eta + \frac{2\eta^2\omega}{M}\right) 2\gamma^3 L \sum_{m=1}^M \|\nabla F_m(x_*)\|^2 \left(\sum_{j=0}^{n-1} (1 - \gamma\mu)^j\right)
 \end{aligned}$$

Unrolling this recursion as we did previously finishes the proof. \square

E. Experimental details

In all experiments we used random sparsification as compression operator:

$$C(x) = \frac{d}{k} \sum_{i \in S} x_i e_i,$$

where S is a random subset of $\{1, 2, \dots, d\}$ of cardinality k chosen uniformly at random, and e_i is the i -th standard unit basis vector in \mathbb{R}^d .

We used carefully chosen stepsizes in terms of constants to have fair competition. We include code in folder because a link to repository can deanonymize the submission.