Decentralization Meets Quantization

Hanlin Tang*[1], Ce Zhang†[2], Shaoduo Gan‡[2], Tong Zhang§[3], and Ji Liu¶[3,1]

[1]Department of Computer Science, University of Rochester
[2]Department of Computer Science, ETH Zurich
[3]Tencent AI Lab

April 23, 2018

Abstract

Optimizing distributed learning systems is an art of balancing between computation and communication. There have been two lines of research that try to deal with slower networks: quantization *for low bandwidth networks, and* decentralization *for high latency networks. In this paper, we explore a natural question:* can the combination of both decentralization and quantization lead to a system that is robust to both bandwidth and latency?

Although the system implication of such combination is trivial, the underlying theoretical principle and algorithm design is challenging: simply quantizing data sent in a decentralized training algorithm would accumulate the error. In this paper, we develop a framework of quantized, decentralized training and propose two different strategies, which we call extrapolation compression *and* difference compression. *We analyze both algorithms and prove both converge at the rate of $O(1/\sqrt{nT})$ where n is the number of workers and T is the number of iterations, matching the convergence rate for full precision, centralized training. We evaluate our algorithms on training deep learning models, and find that our proposed algorithm outperforms the best of merely decentralized and merely quantized algorithm significantly for networks with* both *high latency and low bandwidth.*

1 Introduction

When training machine learning models in a distributed fashion, the underlying constraints of how workers (or nodes) communication have a significant impact on the training algorithm. When workers cannot form a fully connected communication topology (maybe due to physical constraints), or the communication latency is high, decentralizing the communication topology comes to the rescue. On the other hand, when the amount of data sent through the network is an optimization objective (maybe to lower the cost or energy consumption), or the network bandwidth is low, quantizing the traffic to a low precision representation

*htang14@ur.rochester.edu
†ce.zhang@inf.ethz.ch
‡ganshaoduo@gmail.com
§tongzhang@tongzhang-ml.org
¶ji.liu.uwisc@gmail.com

is a popular strategy. In this paper, our goal is to develop a novel framework that works robustly in an environment that *both* decentralization and quantization could be beneficial.

Both decentralized training and quantized training have attracted intensive interests recently. Decentralized algorithms usually exchange local models among nodes, which consumes the main communication budget; on the other hands, quantized algorithms usually exchange quantized gradient, and update an un-quantized model. A straightforward idea to combine these two is to directly quantize the models sent through the network during decentralized training. However, this simple strategy does not converge to the right solution as the quantization error would accumulate during training. The technical contribution of this paper is to develop novel algorithms that combine *both* decentralized training and quantized training together.

Problem Formulation. We consider the following decentralized optimization:

$$\min_{x\in\mathbb{R}^N} \quad f(x) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{\mathbb{E}_{\xi\sim\mathcal{D}_i} F_i(x;\xi)}_{=:f_i(x)}, \tag{1}$$

where n is the number of node and $\mathcal{D}_i$ is the local data distribution for node i. n nodes form a connected graph and each node can only communicate with its neighbors. Here we only assume $f_i(x)$'s are with L-Lipschitzian gradients.

Summary of Technical Contributions. In this paper, we propose two decentralized parallel stochastic gradient descent algorithms (D-PSGD): extrapolation compression D-PSGD (ECD-PSGD) and difference compression D-PSGD (DCD-PSGD). Both algorithms can be proven to converge in the rate roughly $O(1/\sqrt{nT})$ where T is the number of iterations. The convergence rates are consistent with two special cases: centralized parallel stochastic gradient descent (C-PSGD) and D-PSGD.

The key difference between ECD-PSGD and DCD-PSGD is that DCD-PSGD quantizes the *difference* between the last two local models, and ECD-PSGD quantizes the *extrapolation* between the last two local models, and send to the neighbors. DCD-PSGD admits a slightly better convergence rate than ECD-PSGD when the data variation among nodes is very large. On the other hand, ECD-PSGD is more robust to more aggressive quantization, as extremely low precision quantization can cause DCD-PSGD to diverge, since DCD-PSGD has strict constraint on quantization. In this paper, we analyze both algorithms, and empirically validate our theory. We also show that when the underlying network has both high latency and low bandwidth, both algorithms outperform state-of-the-arts significantly.

Definitions and notations Throughout this paper, we use following notations and definitions:

- $\nabla f(\cdot)$ denotes the gradient of a function f.
- f^* denotes the optimal solution of (1).
- $\lambda_i(\cdot)$ denotes the i-th largest eigenvalue of a matrix.
- $\mathbf{1} = [1, 1, \cdots, 1]^\top \in \mathbb{R}^n$ denotes the full-one vector.
- $\|\cdot\|$ denotes the l_2 norm for vector.
- $\|\cdot\|_F$ denotes the vector Frobenius norm of the matrix.

2 Related work

Stochastic gradient descent The *Stocahstic Gradient Descent* (**SGD**) [Ghadimi and Lan, 2013, Moulines and Bach, 2011, Nemirovski et al., 2009] - a stochastic variant of the gradient descent method - has been widely used for solving large scale machine learning problems [Bottou, 2010]. It admits the optimal convergence rate $O\left(\frac{1}{\sqrt{T}}\right)$ for non-convex functions.

Centralized algorithms The centralized algorithms is a widely used scheme for parallel computation, such as Tensorflow [Abadi et al., 2016], MXNet [Chen et al., 2015], and CNTK [Seide and Agarwal, 2016]. It uses a central node to control all leaf nodes. For *Centralized Parallel Stochastic Gradient Descent* (**C-PSGD**), the central node performs parameter updates and leaf nodes compute stochastic gradients based on local information in parallel. In Agarwal and Duchi [2011], Zinkevich et al. [2010], the effectiveness of C-PSGD is studied with latency taken into consideration. The distributed mini-batches SGD, which requires each leaf node to compute the stochastic gradient more than once before the parameter update, is studied in Dekel et al. [2012]. Recht et al. [2011] proposed a variant of C-PSGD, HOGWILD, and proved that it would still work even if we allow the memory to be shablack and let the private mode to be overwriten by others. The asynchronous non-convex C-PSGD optimization is studied in Lian et al. [2015]. Zheng et al. [2016] proposed an algorithm to improve the performance of the asynchronous C-PSGD. In Alistarh et al. [2017], De Sa et al. [2017], a quantized SGD is proposed to save the communication cost for both convex and non-convex object functions. The convergence rate for C-PSGD is $O\left(\frac{1}{\sqrt{Tn}}\right)$.

Decentralized algorithms Recently, decentralized training algorithms have attracted significantly amount of attentions. Decentralized algorithms are mostly applied to solve the consensus problem [Lian et al., 2017a, Sirb and Ye, 2016, Zhang et al., 2017b], where the network topology is decentralized. A recent work shows that decentralized algorithms could outperform the centralized counterpart for distributed training [Lian et al., 2017a]. The main advantage of decentralized algorithms over centralized algorithms lies on avoiding the communication traffic in the central node. In particular, decentralized algorithms could be much more efficient than centralized algorithms when the network bandwidth is small and the latency is large.

The decentralized algorithm (also named gossip algorithm in some literature under certain scenarios [Colin et al., 2016]) only assume a connect computational network, without using the central node to collect information from all nodes. Each node owns its local data and can only exchange information with its neighbors. The goal is still to learn a model over all distributed data. The decentralized structure can applied in solving of multi-task multi-agent reinforcement learning [Mhamdi et al., 2017, Omidshafiei et al., 2017]. Boyd et al. [2006] uses a randomized weighted matrix and studied the effectiveness of the weighted matrix in different situations. Two methods [Li et al., 2017, Shi et al., 2015] were proposed to blackuce the steady point error in decentralized gradient descent convex optimization. Dobbe et al. [2017] applied an information theoretic framework for decentralize analysis. The performance of the decentralized algorithm is dependent on the second largest eigenvalue of the weighted matrix.

Decentralized parallel stochastic gradient descent The *Decentralized Parallel Stochastic Gradient Descent* (**D-PSGD**) [Nedic and Ozdaglar, 2009, Yuan et al., 2016] requires each node to exchange its own stochastic gradient and update the parameter using the information it receives. In Nedic and Ozdaglar [2009], the convergence rate for a time-varying topology was proved when the maximum of the subgradient is assumed to be bounded. In Lan et al. [2017], a new decentralized primal-dual type method is proposed with a computational complexity of $O\left(\sqrt{\frac{n}{T}}\right)$ for general convex objectives. The linear speedup of D-PSGD is proved in Lian et al. [2017a], where the computation complexity is $O\left(\sqrt{\frac{1}{nT}}\right)$. The asynchronous variant of D-PSGD is studied in Lian et al. [2017b].

Compression To guarantee the convergence and correctness, this paper only considers using the unbiased stochastic compression techniques. Existing methods include randomized quantization [Suresh et al., 2017, Zhang et al., 2017a] and randomized sparsification [Wangni et al., 2017]. Other compression methods can be found in Kashyap et al. [2007], Lavaei and Murray [2012], Nedic et al. [2009], but may not be suitable for our problem since they require the model on each work needs to be quantized.

3 Preliminary: decentralized parallel stochastic gradient descent (D-PSGD)

Unlike the traditional (centralized) parallel stochastic gradient descent (C-PSGD), which requires a central node to compute the average value of all leaf nodes, the decentralized parallel stochastic gradient descent (D-PSGD) algorithm does not need such a central node. Each node (say node i) only exchanges its local model $\boldsymbol{x}^{(i)}$ with its neighbors to take weighted average, specifically,

$$\boldsymbol{x}^{(i)} = \sum_{j=1}^{n} W_{ij}\boldsymbol{x}^{(j)}. \tag{2}$$

where $W_{ij} \geq 0$ in general and $W_{ij} = 0$ means that node i and node j is not connected. Each iteration of D-PSGD consists of three steps (t is the iteration number and i is the node index):

- Each node computes the stochastic gradient $\nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)})$, where $\xi^{(i)}$ is the samples from its local data set and $\boldsymbol{x}_t^{(i)}$ is the local model on node i.
- Each node queries its neighbors' variables and updates its local model using (2).
- Each node updates its local model $\boldsymbol{x}_t^{(i)} \leftarrow \boldsymbol{x}_t^{(i)} - \gamma\nabla F_i\left(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)}\right)$ using stochastic gradient, where γ_t is the learning rate.

To look at the D-PSGD from a global view, by defining

$$\begin{aligned}
X :=& [\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \cdots, \boldsymbol{x}^{(n)}] \in \mathbb{R}^{N\times n},\\
G(X;\xi) :=& [\nabla F_1(x^{(1)};\xi^{(1)}), \cdots, \nabla F_n(x^{(n)};\xi^{(n)})]\\
\nabla f(\overline{X}) :=& \sum_{i=1}^{n} \frac{1}{n}\nabla f_i(X\frac{\mathbf{1}}{n}) = \sum_{i=1}^{n}\frac{1}{n}\nabla f_i\left(\frac{1}{n}\sum_{i=1}^{n} x^{(i)}\right),\\
\overline{\nabla f}(X) :=& \mathbb{E}_{\xi}\overline{G}(X;\xi_t) = \frac{1}{n}\sum_{i=1}^{n}\nabla f_i(x^{(i)}),
\end{aligned}$$

the D-PSGD can be summarized into the following form

$$X_{t+1} = X_t W - \gamma_t G(X_t; \xi_t),$$

The convergence rate of D-PSGD can be shown to be $O\left(\frac{\sigma}{\sqrt{nT}} + \frac{n^{\frac{1}{3}}\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}}\right)$ (without assuming strong convexity) where both σ and ζ are the stochastic variance (please refer to Assumption 1 for detailed definitions), if the learning rate is chosen appropriately.

4 Quantized, Decentralized Algorithms

In this section, we introduce two quantized decentralized algorithms that compress information exchanged between nodes. Note that all communication cost for decentralized algorithms is exchanging local models $\boldsymbol{x}^{(i)}$ among all nodes.

To reduce the communication cost, a straightforward idea is to let nodes exchange low precision $\boldsymbol{x}^{(i)}$ (e.g., one can use unbiased random quantization[1] [Zhang et al., 2017a] or sparsification[2] [Wangni et al., 2017] to compress local $\boldsymbol{x}^{(i)}$ before exchanging).

Straightforward compression does not work Let the quantized or sparsified version X_t be

$$\tilde{X}_t = X_t + Q_t,$$

where $Q_t = [\boldsymbol{q}_t^{(1)}, \boldsymbol{q}_t^{(2)}, \cdots, \boldsymbol{q}_t^{(n)}]$, and $\boldsymbol{q}_t^{(i)} = \tilde{\boldsymbol{x}}_t^{(i)} - \boldsymbol{x}_t^{(i)}$ is the random noise to represent random quantization or sparsification. Then the update iteration becomes

$$\begin{aligned} X_{t+1} =& \tilde{X}_t W - \gamma_t G(X_t; \xi_t) \\ =& X_t W + \underbrace{Q_t W}_{\text{not diminish}} - \gamma_t G(X_t; \xi_t). \end{aligned}$$

However, this idea does not work, because the quantization error Q_t does not diminish unlike the stochastic gradient variance that can be controlled by γ_t either decays to zero or is chosen to be small enough. Next we will propose two approaches to fix this issue.

4.1 Extrapolation compression approach

Since the information exchanged between workers has been compressed, there is no way to obtain the exact models from neighbors. Then the problem becomes how to effectively estimate the neighbors' models from compressed information. We introduce an extrapolation compression approach in the following.

Instead of sending the local model $\boldsymbol{x}_t^{(i)}$ directly to neighbors, we send a z-value that is extrapolated from $\boldsymbol{x}_t^{(i)}$ and $\boldsymbol{x}_{t-1}^{(i)}$ at each iteration. Each node (say, node i) estimates its neighbor's values $\boldsymbol{x}_t^{(j)}$ from compressed

[1] A real number is randomly quantized into one of closest thresholds, for example, givens the thresholds $\{0, 0.3, 0.8, 1\}$, the number "0.5" will be quantized to 0.3 with probability 40% and to 0.8 with probability 60%. Here, we assume that all numbers have been normalized into the range $[0, 1]$.

[2] A real number z is set to 0 with probability $1 - p$ and to z/p with probability p.

z-value at t-th iteration. The following will show how to estimate $\boldsymbol{x}_t^{(j)}$ by node i (j is the neighbor of node i), which can ensure diminishing estimate error, in particular, $\mathbb{E}\|\tilde{\boldsymbol{x}}_t^{(j)} - \boldsymbol{x}_t^{(j)}\|^2 \leq \mathcal{O}\left(t^{-1}\right)$.

At tth iteration, node i performs the following steps to estimate $\boldsymbol{x}_t^j$ by $\tilde{\boldsymbol{x}}_t^j$:

- The node j, computes the z-value that is obtained through extrapolation

$$\boldsymbol{z}_t^{(j)} = \left(1 - \frac{t}{2}\right)\boldsymbol{x}_{t-1}^{(j)} + \frac{t}{2}\boldsymbol{x}_t^{(j)}, \tag{3}$$

  where t refers to the iteration number.

- Compress $\boldsymbol{z}_t^{(j)}$ and send it to its neighbors:

$$\boldsymbol{C}(\boldsymbol{z}_t^{(j)}) = \boldsymbol{z}_t^{(j)} + \boldsymbol{\delta}_t^{(j)}$$

  where $\boldsymbol{C}(\cdot)$ denotes the compression operation, and $\boldsymbol{\delta}_t^{(j)}$ is the noise induced by the random quantization or sparsification at tth iteration;

- Node i estimates $\boldsymbol{x}_t^{(j)}$ by $\tilde{\boldsymbol{x}}_t^{(j)}$ using

$$\tilde{\boldsymbol{x}}_t^{(j)} = \left(1 - 2t^{-1}\right)\tilde{\boldsymbol{x}}_{t-1}^{(j)} + 2t^{-1}\boldsymbol{C}(\boldsymbol{z}_t^{(j)}). \tag{4}$$

One can see that

$$\tilde{\boldsymbol{x}}_t^{(j)} - \boldsymbol{x}_t^{(j)} = (1-2t^{-1})(\tilde{\boldsymbol{x}}_{t-1}^{(j)} - \boldsymbol{x}_{t-1}^{(j)}) + 2t^{-1}\boldsymbol{\delta}_t^{(j)},$$

and

$$\begin{aligned}&\mathbb{E}\|\tilde{\boldsymbol{x}}_t^{(j)} - \boldsymbol{x}_t^{(j)}\|^2\\ =&\left(1-2t^{-1}\right)^2\mathbb{E}\|\tilde{\boldsymbol{x}}_{t-1}^{(j)} - \boldsymbol{x}_{t-1}^{(j)}\|^2 + 4t^{-2}\mathbb{E}\|\boldsymbol{\delta}_t^{(j)}\|^2 + 2t^{-1}\left(1-2t^{-1}\right)\mathbb{E}\left\langle \tilde{\boldsymbol{x}}_{t-1}^{(j)} - \boldsymbol{x}_{t-1}^{(j)}, \mathbb{E}_{\boldsymbol{\delta}_t^{(j)}}\boldsymbol{\delta}_t^{(j)}\right\rangle\end{aligned}$$

It's very important to keep the unbiasedness in each iteration, which means $\mathbb{E}_{\boldsymbol{\delta}_t^{(j)}}\boldsymbol{C}(\boldsymbol{z}_t^{(j)}) = \boldsymbol{z}_t^{(j)}$ must be satisfied, so we have $\mathbb{E}_{\boldsymbol{\delta}_t^{(j)}}\boldsymbol{\delta}_t^{(j)} = 0$. So we have

$$\mathbb{E}\|\tilde{\boldsymbol{x}}_t^{(j)} - \boldsymbol{x}_t^{(j)}\|^2 = \left(1-2t^{-1}\right)^2\mathbb{E}\|\tilde{\boldsymbol{x}}_{t-1}^{(j)} - \boldsymbol{x}_{t-1}^{(j)}\|^2 + 4t^{-2}\mathbb{E}\|\boldsymbol{\delta}_t^{(j)}\|^2. \tag{5}$$

Using Lemma 9 (see Supplemental Materials), if the compression noise $\boldsymbol{\delta}_t^{(j)}$ is globally bounded variance by $\tilde{\sigma}^2$, we have

$$\mathbb{E}(\|\tilde{\boldsymbol{x}}_t^{(j)} - \boldsymbol{x}_t^{(j)}\|^2) \leq \frac{\tilde{\sigma}^2}{t}.$$

Using this way to estimate the neighbors' local models leads to the following equivalent updating form

$$\begin{aligned}X_{t+1} =& \tilde{X}_t W - \gamma_t G(X_t;\xi_t)\\ =& X_t W + \underbrace{Q_t W}_{\text{diminished estimate error}} - \gamma_t G(X_t;\xi_t).\end{aligned} \tag{6}$$

Algorithm 1 ECD-PSGD

Input: Initial point $\boldsymbol{x}_1^{(i)} = \boldsymbol{x}_1$, initial estimate $\boldsymbol{y}_1^{(i)} = \boldsymbol{x}_1$, iteration step length γ, weighted matrix W, and number of total iterations T.

for $t = 1, 2, \cdots, T$ **do**

Randomly sample $\xi_t^{(i)}$ from local data of the ith node

Compute a local stochastic gradient based on $\xi_t^{(i)}$ and current optimization variable $\boldsymbol{x}_t^{(i)} : \nabla F_i(\boldsymbol{x}_t^{(i)}, \xi_t^{(i)})$

Compute the neighborhood weighted average by using the estimate value of the connected neighbors [a]:

$$\boldsymbol{x}_{t+\frac{1}{2}}^{(i)} = \sum_{j=1}^{n} W_{ij} \boldsymbol{y}_t^{(j)}$$

where $\boldsymbol{y}_t^{(j)}$ is the estimate for node j's value after compression.

Update the local model

$$\boldsymbol{x}_{t+1}^{(i)} \leftarrow \boldsymbol{x}_{t+\frac{1}{2}}^{(i)} - \gamma \nabla F_i(\boldsymbol{x}_t^{(i)}, \xi_t^{(i)})$$

Each node computes the z-value of itself:

$$\boldsymbol{z}_{t+1}^{(i)} = \left(1 - \frac{t}{2}\right) \boldsymbol{x}_t^{(i)} + \frac{t}{2} \boldsymbol{x}_{t+1}^{(i)}$$

and compress this $\boldsymbol{z}_t^{(i)}$ into $\boldsymbol{C}(\boldsymbol{z}_t^{(j)})$.

Each node updates the estimate for its connected neighbors:

$$\boldsymbol{y}_{t+1}^{(j)} = \left(1 - \frac{2}{t}\right) \boldsymbol{y}_t^{(j)} + \frac{2}{t} \boldsymbol{C}(\boldsymbol{z}_t^{(j)})$$

end for

Output: $\frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_T^{(i)}$

[a] The connected neighbors of one node i here refers to all the nodes that satisfy $W_{i,j} \neq 0$

Still we define $Q_t = [\boldsymbol{q}_t^{(1)}, \boldsymbol{q}_t^{(2)}, \cdots, \boldsymbol{q}_t^{(n)}]$ and $\boldsymbol{q}_t^{(i)} = \tilde{\boldsymbol{x}}_t^{(i)} - \boldsymbol{x}_t^{(i)}$. The full extrapolation compression D-PSGD (ECD-PSGD) algorithm is summarized in Algorithm 1.

Below we will show that EDC-PSGD algorithm would admit the same convergence rate and the same computation complexity as D-PSGD.

We first make some common assumptions for analyzing decentralized stochastic algorithms Lian et al. [2017b].

Assumption 1. *Throughout this paper, we make the following commonly used assumptions:*

1. ***Lipschitzian gradient:*** *All function $f_i(\cdot)$'s are with L-Lipschitzian gradients.*

2. ***Symmetric double stochastic matrix:*** *The weighted matrix W is a real double stochastic matrix that satisfies $W = W^\top$ and $W\mathbf{1} = W$.*

3. ***Spectral gap:*** *Given the symmetric doubly stochastic matrix W, we define $\rho := \max\{|\lambda_2(W)|, |\lambda_n(W)|\}$ and assume $\rho < 1$.*

4. ***Bounded variance:*** *Assume the variance of stochastic gradient*

$$\mathbb{E}_{\xi \sim \mathcal{D}_i} \|\nabla F_i(\boldsymbol{x}; \xi) - \nabla f_i(\boldsymbol{x})\|^2 \leqslant \sigma^2, \quad \forall i, \forall \boldsymbol{x},$$

$$\frac{1}{n}\sum_{i=1}^{n}\|\nabla f_i(\boldsymbol{x})-\nabla f(\boldsymbol{x})\|^2 \leqslant \zeta^2, \quad \forall i, \forall \boldsymbol{x},$$

is bounded for any x in each node i.

5. ***Start from 0:*** *We assume $X_1 = 0$. This assumption simplifies the proof w.l.o.g.*

For ECD-PSGD, it's reasonable to assume that the noise brought by compression is bounded. So we make the following assumption.

Assumption 2. *(**Bounded compression noise**) Assume the noise due to compression is unbiased and its variance is bounded, that is,*

$$\mathbb{E}(\delta_t^{(i)}) = 0, \quad \mathbb{E}\|\delta_t^{(i)}\|^2 \le \frac{\tilde{\sigma}^2}{2}, \quad \forall i, \forall \boldsymbol{x}, \forall t.$$

Theorem 1 (Convergence of Algorithm 1)**.** *Under Assumptions 1 and 2, choosing γ_t in Algorithm 1 to be a constant γ satisfying $1 - 6C_1L^2\gamma^2 > 0$, we have the following convergence rate for Algorithm 1*

$$\begin{aligned}
&\sum_{t=1}^{T}\left((1-C_3)\,\mathbb{E}\|\nabla f(\overline{X}_t)\|^2 + C_4\mathbb{E}\|\overline{\nabla f}(X_t)\|^2\right)\\
\le&\frac{2(f(0)-f^*)}{\gamma} + \frac{L\log T}{n\gamma}\tilde{\sigma}^2 + \frac{LT\gamma}{n}\sigma^2 + \frac{4C_2\tilde{\sigma}^2L^2}{1-\rho^2}\log T + 4L^2C_2\left(\sigma^2+3\zeta^2\right)C_1T\gamma^2.
\end{aligned} \tag{7}$$

where

$$\begin{aligned}
C_1 :=&\frac{1}{(1-\rho)^2},\\
C_2 :=&\frac{1}{1-6\rho^{-2}C_1L^2\gamma^2},\\
C_3 :=&12L^2C_2C_1\gamma^2,\\
C_4 :=&1-L\gamma.
\end{aligned}$$

To make the result more clear, we choose the steplength in the following:

Corollary 2. *In Algorithm 1 choose the steplength $\gamma = \left(6\sqrt{C_2C_1}L + \frac{\sigma}{\sqrt{n}}T^{\frac{1}{2}} + \zeta^{\frac{2}{3}}T^{\frac{1}{3}}\right)^{-1}$. Then it admits*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla f(\overline{X}_t)\|^2 \lesssim \frac{\sigma}{\sqrt{nT}} + \frac{\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}} + \frac{1}{T} + \frac{\sigma\tilde{\sigma}^2\log T}{n\sqrt{nT}} + \frac{\zeta^{\frac{2}{3}}\tilde{\sigma}^2\log T}{nT^{\frac{2}{3}}} + \frac{\tilde{\sigma}^2\log T}{T}, \tag{8}$$

and

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \lesssim \frac{n\tilde{\sigma}^2\log T}{T} + \frac{n\sqrt{n}}{T} + \frac{n\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}} + \frac{1}{T^2} + \frac{\tilde{\sigma}^2\sigma\log T}{T^{\frac{2}{3}}} + \frac{\zeta^{\frac{3}{2}}\tilde{\sigma}^2\log T}{T^{\frac{5}{3}}},$$

where we treat $f(0) - f^*$, L, and ρ constants.

This result suggests the algorithm converges roughly in the rate $(1/\sqrt{nT})$. The followed analysis will bring more detailed interpretation to show the tightness of our result.

Algorithm 2 DCD-PSGD

Input: Initial point $x_1^{(i)} = x_1$, initial replica $\tilde{x}_1^{(i)} = x_1$, iteration step length γ, weighted matrix W, and number of total iterations T

for t = 1,2,...,T **do**

Randomly sample $\xi_t^{(i)}$ from local data of the ith node

Compute a local stochastic gradient based on $\xi_t^{(i)}$ and current optimization variable $\boldsymbol{x}_t^{(i)}$: $\nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)})$

Update the local model using local stochastic gradient and the weighted average of its connected neighbors' replica[a]:

$$\boldsymbol{x}_{t+\frac{1}{2}}^{(i)} = \sum_{j=1}^{n} W_{ij}\boldsymbol{x}_t^{(j)} - \gamma\nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)}),$$

where $\tilde{\boldsymbol{x}}_t^{(j)}$ is the replica of node j's value.

Each node computes the difference between $\boldsymbol{x}_t^{(i)}$ and $\boldsymbol{x}_{t+\frac{1}{2}}^{(i)}$:

$$\boldsymbol{z}_t^{(i)} = \boldsymbol{x}_{t+\frac{1}{2}}^{(i)} - \boldsymbol{x}_t^{(i)},$$

and compress this $\boldsymbol{z}_t^{(i)}$ into $\boldsymbol{C}(\boldsymbol{z}_t^{(i)})$.

Update the local optimization variables $\boldsymbol{x}_{t+1}^{(i)} \leftarrow \boldsymbol{x}_t^{(i)} + \boldsymbol{C}(\boldsymbol{z}_t^{(i)})$.[b]

Send $\boldsymbol{C}(\boldsymbol{z}_t^{(i)})$ to its connected neighbors, and update the replicas of its connected neighbors' values:

$$\tilde{\boldsymbol{x}}_{t+1}^{(j)} = \tilde{\boldsymbol{x}}_t^{(j)} + \boldsymbol{C}(\boldsymbol{z}_t^{(i)}).^b$$

end for

Output: $\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_T^{(i)}$

[a]The connected neighbors of one node i here refers to all the nodes that satisfy $W_{i,j} \neq 0$

[b]Decompressing the quantized value first, then add them in same precision

Linear speedup Since the leading term of the convergence rate is $O\left(\frac{1}{\sqrt{nT}}\right)$ when T is large, which is consistent with the convergence rate of C-PSGD, this indicates that we would achieve a linear speed up with respect to the number of nodes.

Consistence with D-PSGD Setting $\tilde{\sigma} = 0$ to match the scenario of D-PSGD, ECD-PSGD admits the rate $O\left(\frac{\sigma}{\sqrt{nT}} + \frac{\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}}\right)$, that is slightly better the rate of D-PSGD proved in Lian et al. [2017b] $O\left(\frac{\sigma}{\sqrt{nT}} + \frac{n^{\frac{1}{3}}\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}}\right)$.

Better compression performance with more nodes The compression noise leading term in (8) is $O\left(\frac{\tilde{\sigma}^2}{n\sqrt{nT}}\right)$ that becomes minor if the number of nodes is large enough.

4.2 Difference compression approach

This section introduces another approach, namely, difference compression D-PSGD (DCD-PSGD), to compress the exchanged information such that the noise from compression diminishes over iterations.

The DCD-PSGD basically follows the framework of D-PSGD, except that nodes exchange the compressed difference of local models between two successive iterations instead of exchanging local models. More specifically, each node needs to store its neighbors' models in last iteration $\{\boldsymbol{x}_t^{(j)} : j \text{ is node } i\text{'s neighbor}\}$ and follow the following steps

- take the weighted average and apply stochastic gradient descent step

$$\boldsymbol{x}_{t+\frac{1}{2}}^{(i)} = \sum_{j=1}^{n} W_{ij}\tilde{\boldsymbol{x}}_t^{(j)} - \gamma \nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)}),$$

  where $\tilde{\boldsymbol{x}}_t^{(j)}$ is just the replica of $\boldsymbol{x}_t^{(j)}$ but is stoblack on node i[3];

- compress the difference between $\boldsymbol{x}_t^{(i)}$ and $\boldsymbol{x}_{t+\frac{1}{2}}^{(i)}$ and update the local model:

$$\begin{aligned}
\boldsymbol{z}_t^{(i)} =& \boldsymbol{x}_{t+\frac{1}{2}}^{(i)} - \boldsymbol{x}_t^{(i)} \\
\boldsymbol{x}_{t+1}^{(i)} =& \boldsymbol{x}_t^{(i)} + \boldsymbol{C}(\boldsymbol{z}_t^{(i)}),
\end{aligned} \tag{9}$$

  here $\boldsymbol{C}(\boldsymbol{z}_t^{(i)})$ is the compressed value of $\boldsymbol{z}_t^{(i)}$;

- send $\boldsymbol{C}(\boldsymbol{z}_t^{(i)})$ and query neighbors' $\boldsymbol{C}(\boldsymbol{z}_t)$'s to update the local replica

$$\tilde{\boldsymbol{x}}_{t+1}^{(j)} = \tilde{\boldsymbol{x}}_t^{(j)} + \boldsymbol{C}(\boldsymbol{z}_t^{(j)}) \quad \forall j \text{ is node } i\text{'s neighbor}.$$

The full DCD-PSGD algorithm is described in Algorithm 2.

To make the DCD-PSGD algorithm work, we need to make some restriction on the compression operator $\boldsymbol{C}(\cdot)$. Again this compression operator could be random quantization or random sparsification or any other operators. We introduce the definition of the signal-to-noise related parameter α. Let

$$\begin{aligned}
Z :=& [\boldsymbol{z}_1, \boldsymbol{z}, \cdots, \boldsymbol{z}_n] \in \mathbb{R}^{N\times n} \\
C :=& [\boldsymbol{C}(\boldsymbol{z}_1), \boldsymbol{C}(\boldsymbol{z}_2), \cdots, \boldsymbol{C}(\boldsymbol{z}_n)] \in \mathbb{R}^{N\times n} \\
Q :=& C - Z
\end{aligned}$$

and define

$$\alpha := \sqrt{\sup_{Z\neq 0} \|Q\|_F^2 / \|Z\|_F^2}.$$

Theorem 3. *Under the Assumption 1, if α satisfies $(1-\rho)^2 - 4\mu^2\alpha^2 > 0$, choosing γ satisfying $1 - 3D_1L^2\gamma^2 > 0$, we have the following convergence rate for Algorithm 2*

$$\begin{aligned}
&\sum_{t=1}^{T} \left((1-D_3)\,\mathbb{E}\|\nabla f(\overline{X}_t)\|^2 + D_4\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 \right) \\
\leq& \frac{2(f(0)-f^*)}{\gamma} + \frac{L\gamma T\sigma^2}{n} + \left(\frac{T\gamma^2 LD_2}{2} + \frac{\left(4L^2 + 3L^3D_2\gamma^2\right) D_1T\gamma^2}{1-3D_1L^2\gamma^2} \right)\sigma^2 + \frac{\left(4L^2 + 3L^3D_2\gamma^2\right) 3D_1T\gamma^2}{1-3D_1L^2\gamma^2}\zeta^2,
\end{aligned} \tag{10}$$

where

$$\mu := \max_{i\in\{2,\cdots,n\}} |\lambda_i - 1|$$

[3]Actually all neighbors of node j will have a replica of $\boldsymbol{x}_t^{(j)}$.

$$
\begin{aligned}
D_1 :=& \frac{2\alpha^2}{1-\rho^2}\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right)+\frac{1}{(1-\rho)^2}\\
D_2 :=& 2\alpha^2\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right)\\
D_3 :=& \frac{\left(4L^2+3L^3D_2\gamma^2\right)3D_1\gamma^2}{1-3D_1L^2\gamma^2}+\frac{3LD_2\gamma^2}{2}\\
D_4 :=& (1-L\gamma).
\end{aligned}
$$

To make the result more clear, we appropriately choose the steplength in the following:

Corollary 4. *Choose*

$$
\gamma = \left(6\sqrt{D_1}L + 6\sqrt{D_2 L} + \frac{\sigma}{\sqrt{n}}T^{\frac{1}{2}} + \zeta^{\frac{2}{3}}T^{\frac{1}{3}}\right)^{-1}
$$

in Algorithm 2. If α is small enough that satisfies

$$
\alpha \leq \min\left\{\frac{1-\rho}{2\sqrt{2}\mu}, \frac{1}{2}\right\}
$$

Then we have

$$
\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla f(\overline{X}_t)\|^2 \lesssim \frac{\sigma}{\sqrt{Tn}} + \frac{\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}} + \frac{1}{T},
$$

and

$$
\frac{1}{T}\sum_{i=1}^{n}\sum_{t=1}^{T}\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \lesssim \frac{n\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}} + \frac{n\sqrt{n}}{T} + \frac{n}{T^2}.
$$

where D_1, D_2 follow to same definition in Theorem 3 and we treat $f(0)-f^*$, L, and ρ constants.

The leading term of the convergence rate is $O\left(\frac{1}{\sqrt{Tn}}\right)$, we shall see the tightness of our result in the following discussion.

Linear speedup Since the leading term of the convergence rate is $O\left(\frac{1}{\sqrt{nT}}\right)$ when T is large, which is consistent with the convergence rate of C-PSGD, this indicates that we would achieve a linear speed up with respect to the number of nodes.

Consistence with D-PSGD Setting $\alpha = 0$ to match the scenario of D-PSGD, ECD-PSGD admits the rate $O\left(\frac{\sigma}{\sqrt{nT}} + \frac{\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}}\right)$, that is slightly better the rate of D-PSGD proved in Lian et al. [2017b] $O\left(\frac{\sigma}{\sqrt{nT}} + \frac{n\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}}\right)$.

Comparison to ECD-PSGD On one side, the convergence rate of DCD-PSGD is slightly better than that of ECD-PSGD, since the additional terms in ECD-PSGD $O\left(\frac{\sigma\tilde{\sigma}^2\log T}{n\sqrt{nT}} + \frac{\zeta^{\frac{2}{3}}\tilde{\sigma}^2\log T}{nT^{\frac{2}{3}}} + \frac{\tilde{\sigma}^2\log T}{T}\right)$ could dominate the convergence rate when $\tilde{\sigma}$ or ζ is extremely large. On the other side, DCD-PSGD does not allow too aggressive compression or quantization and may lead to diverge due to the hard constraint for $\alpha \leq \min\left\{\frac{1-\rho}{2\sqrt{2}\mu}, \frac{1}{2}\right\}$, while ECD-PSGD is quite robust to aggressive compression or quantization.

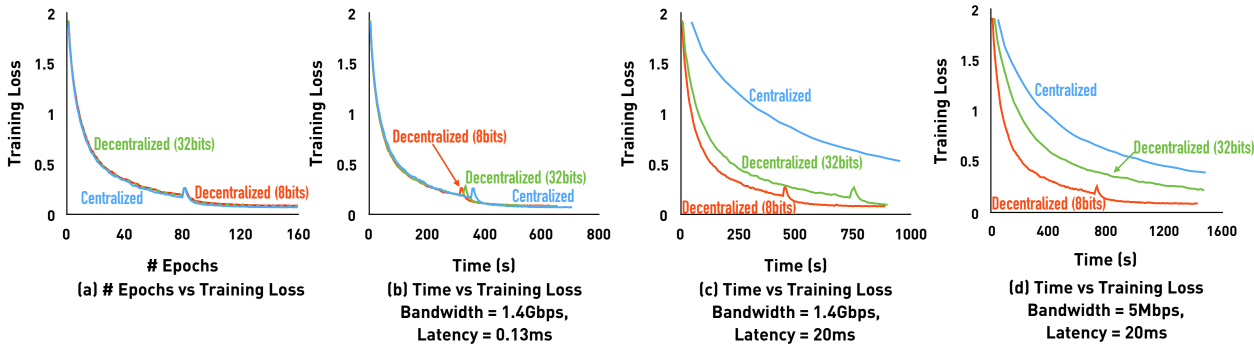

Figure 1: Performance Comparison between Decentralized and Allblackuce implementations (Decentralized algorithm used here is Algorithm 1, and Algorithm 2 has similar curves with Algorithm 1.)

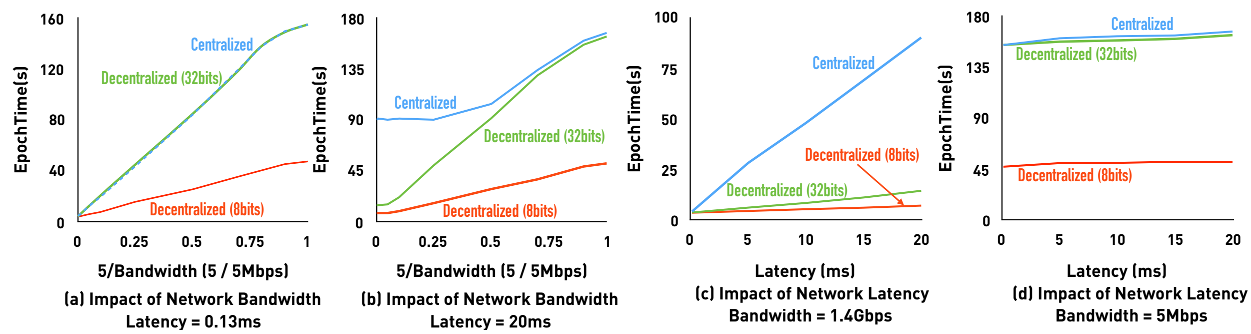

Figure 2: Performance Comparison in Diverse Network Conditions (Decentralized algorithm used here is algorithm 1, and algorithm 2 has similar curves with algorithm 1.)

5 Experiments

In this section we evaluate two decentralized algorithms by comparing with an Allblackuce implementation of centralized SGD. We run experiments under diverse network conditions and show that, decentralized algorithms with low precision can speed up training without hurting convergence.

5.1 Experimental Setup

We choose the image classification task as a benchmark to evaluate our theory. We train ResNet-20 [He et al., 2016] on CIFAR-10 dataset which has 50,000 images for training and 10,000 images for testing. Two proposed algorithms are implemented in Microsoft CNTK and compablack with CNTK's original implementation of distributed SGD:

- **Centralized:** This implementation is based on MPI Allblackuce primitive with full precision (32 bits). It is the standard training method for multiple nodes in CNTT.
- **Decentralized_32bits/8bits:** The implementation of the proposed decentralized approach with OpenMPI. The full precision is 32 bits, and the compressed precision is 8 bits.
- In this paper, we omit the comparison with quantized centralized training because the difference between Decentralized 8bits and Centralized 8bits would be similar to the original decentralized

training paper Lian et al. [2017a] – when the network latency is high, decentralized algorithm outperforms centralized algorithm in terms of the time for each epoch.

We run all experiments on 8 Amazon *p*2.*xlarge* EC2 instances, each of which has one Nvidia K80 GPU. We use each GPU as a node. In decentralized cases, 8 nodes are connected in a ring topology, which means each node just communicates with its two neighbors. The batch size for each node is same as the default configuration in CNTT. We also tune learning rate for each variant.

5.2 Convergence and Run Time Performance

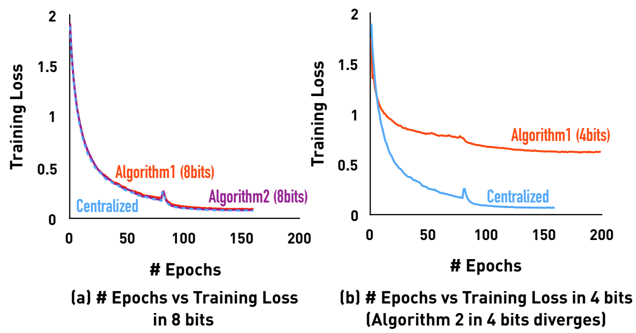

Figure 3: Comparison of Algorithm 1 and Algorithm 2

We first study the convergence of our proposed algorithms. Figure 1(a) shows the convergence w.r.t # epochs of centralized and decentralized cases. We only show ECD-PSGD in the figure (and call it Decentralized) because DCD-PSGD has almost identical convergence behavior in this experiment. We can see that with our algorithms, decentralization and compression would not hurt the convergence rate.

We then compare the runtime performance. Figure 1(b, c, d) demonstrates how training loss decreases with the run time under different network conditions. We use *tc* command to change bandwidth and latency of the underlying network. By default, 1.4 Gbps bandwidth and 0.13 ms latency is the best network condition we can get in this cluster. On this occasion, all implementations have a very similar runtime performance because communication is not the bottleneck for system. When the latency is high, however, decentralized algorithms outperform the Allblackuce because of fewer number of communications. In comparison with decentralized full precision cases, low precision methods exchange much less amount of data and thus can outperform full precision cases in low bandwidth situation, as is shown in Figure 1(d).

5.3 Speedup in Diverse Network Conditions

To better understand the influence of bandwidth and latency on speedup, we compare the time of one epoch under various of network conditions. Figure 2(a, b) shows the trend of epoch time with bandwidth decreasing from 1.4 Gbps to 5 Mbps. When the latency is low (Figure 2(a)), low precision algorithm is faster than its full precision counterpart because it only needs to exchange around one fourth of full precision method's data amount. Note that although in a decentralized way, full precision case has no advantage over Allblackuce in this situation, because they exchange exactly the same amount of data. When it comes to high latency shown in Figure 2(b), both full and low precision cases are much better than Allblackuce in the beginning. But also, full precision method gets worse dramatically with the decline of bandwidth.

Figure 2(c, d) shows how latency influences the epoch time under good and bad bandwidth conditions. When bandwidth is not the bottleneck (Figure 2(c)), decentralized approaches with both full and low precision have similar epoch time because they have same number of communications. As is expected, Allblackuce is slower in this case. When bandwidth is very low (Figure 2(d)), only decentralized algorithm with low precision can achieve best performance among all implementations.

5.4 Discussion

Our previous experiments validate the efficiency of the decentralized algorithms in 8 bits. However, we wonder if we can compress the exchanged data even more aggressively in order to achieve better performance. In other words, we would like to understand the limitation of our compression method. Therefore, we compress the data into 4 bits in decentralized algorithms. As is exhibited in Figure 3, algorithm 1 and algorithm 2 in 8 bits basically have same convergence rate as Allblackuce. However, they can not be comparable to Allblackuce with 4 bits. What is noteworthy is that these two compression approaches have quite different behaviors in 4 bits. For algorithm 1, although it converges much slower than Allblackuce, its training loss keeps blackucing. However, algorithm 2 just diverges in the beginning of training. This observation is also consistent with our theoretical analysis in former section.

6 Conclusion

In this paper, we studied the problem of combining two tricks of training distributed stochastic gradient descent under imperfect network conditions: quantization and decentralization. We developed two novel algorithms or quantized, decentralized training, analyze the theoretical property of both algorithms, and empirically study their performance in a various settings of network conditions. We found that when the underlying communication networks has *both* high latency and low bandwidth, quantized, decentralized algorithm outperforms other strategies significantly.

References

M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

A. Agarwal and J. C. Duchi. Distributed delayed stochastic optimization. In *Advances in Neural Information Processing Systems*, pages 873–881, 2011.

D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding. In *Advances in Neural Information Processing Systems*, pages 1707–1718, 2017.

L. Bottou. Large-scale machine learning with stochastic gradient descent. *Proc. of the International Conference on Computational Statistics (COMPSTAT)*, 2010.

S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE/ACM Trans. Netw.*, 14 (SI):2508–2530, June 2006. ISSN 1063-6692. doi: 10.1109/TIT.2006.874516.

T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, and Z. Zhang. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. *arXiv preprint arXiv:1512.01274*, 2015.

I. Colin, A. Bellet, J. Salmon, and S. Clémençon. Gossip dual averaging for decentralized optimization of pairwise functions. In *International Conference on Machine Learning*, pages 1388–1396, 2016.

C. De Sa, M. Feldman, C. Ré, and K. Olukotun. Understanding and optimizing asynchronous low-precision stochastic gradient descent. In *Proceedings of the 44th Annual International Symposium on Computer Architecture*, pages 561–574. ACM, 2017.

O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13(Jan):165–202, 2012.

R. Dobbe, D. Fridovich-Keil, and C. Tomlin. Fully decentralized policies for multi-agent systems: An information theoretic approach. In *Advances in Neural Information Processing Systems*, pages 2945–2954, 2017.

S. Ghadimi and G. Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013. doi: 10.1137/120880811.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

A. Kashyap, T. Başar, and R. Srikant. Quantized consensus. *Automatica*, 43(7):1192–1203, 2007.

G. Lan, S. Lee, and Y. Zhou. Communication-efficient algorithms for decentralized and stochastic optimization. 01 2017.

J. Lavaei and R. M. Murray. Quantized consensus by means of gossip algorithm. *IEEE Transactions on Automatic Control*, 57(1):19–32, 2012.

Z. Li, W. Shi, and M. Yan. A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates. *arXiv preprint arXiv:1704.07807*, 2017.

X. Lian, Y. Huang, Y. Li, and J. Liu. Asynchronous parallel stochastic gradient for nonconvex optimization. In *Advances in Neural Information Processing Systems*, pages 2737–2745, 2015.

X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. 05 2017a.

X. Lian, W. Zhang, C. Zhang, and J. Liu. Asynchronous decentralized parallel stochastic gradient descent. 10 2017b.

E. Mhamdi, E. Mahdi, H. Hendrikx, R. Guerraoui, and A. D. O. Maurer. Dynamic safe interruptibility for decentralized multi-agent reinforcement learning. Technical report, EPFL, 2017.

E. Moulines and F. R. Bach. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 451–459. Curran Associates, Inc., 2011.

A. Nedic and A. Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009.

A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis. On distributed averaging algorithms and quantization effects. *IEEE Transactions on Automatic Control*, 54(11):2506–2517, 2009.

A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009. doi: 10.1137/070704277.

S. Omidshafiei, J. Pazis, C. Amato, J. P. How, and J. Vian. Deep decentralized multi-task multi-agent rl under partial observability. *arXiv preprint arXiv:1703.06182*, 2017.

B. Recht, C. Re, S. Wright, and F. Niu. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *Advances in neural information processing systems*, pages 693–701, 2011.

F. Seide and A. Agarwal. Cntk: Microsoft's open-source deep-learning toolkit. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 2135–2135, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4232-2. doi: 10.1145/2939672.2945397.

W. Shi, Q. Ling, G. Wu, and W. Yin. A proximal gradient algorithm for decentralized composite optimization. *IEEE Transactions on Signal Processing*, 63(22):6013–6023, 2015.

B. Sirb and X. Ye. Consensus optimization with delayed and stochastic gradients on decentralized networks. In *2016 IEEE International Conference on Big Data (Big Data)*, pages 76–85, 2016.

A. T. Suresh, F. X. Yu, S. Kumar, and H. B. McMahan. Distributed mean estimation with limited communication. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3329–3337, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL `http://proceedings.mlr.press/v70/suresh17a.html`.

J. Wangni, J. Wang, J. Liu, and T. Zhang. Gradient sparsification for communication-efficient distributed optimization. *arXiv preprint arXiv:1710.09854*, 2017.

K. Yuan, Q. Ling, and W. Yin. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3):1835–1854, 2016. doi: 10.1137/130943170.

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

W. Zhang, P. Zhao, W. Zhu, S. C. Hoi, and T. Zhang. Projection-free distributed online learning in networks. In *International Conference on Machine Learning*, pages 4054–4062, 2017b.

S. Zheng, Q. Meng, T. Wang, W. Chen, N. Yu, Z.-M. Ma, and T.-Y. Liu. Asynchronous stochastic gradient descent with delay compensation for distributed deep learning. *arXiv preprint arXiv:1609.08326*, 2016.

M. Zinkevich, M. Weimer, L. Li, and A. J. Smola. Parallelized stochastic gradient descent. In *Advances in neural information processing systems*, pages 2595–2603, 2010.

Supplemental Materials: Proofs

For the convenience of analysis, let us reformulate the updating rule of both proposed algorithms. Both algorithms can be generalized into the following form

$$X_{t+1} = X_t W - \gamma_t G(X_t;\xi_t) + Q_t, \tag{11}$$

where Q_t is the noise caused by compression.

More specifically, for ECD-PSGD, from (3) and (4), we have

$$Q_t = \tilde{X}_t - X_t,$$

where $\tilde{X}_t$ is the estimation of X_t computed using extrapolation. For DCD-PSGD, from (9), we have

$$\begin{aligned}\Delta X_t :=& X_t W - \gamma F(X_t;\xi_t) - X_t\\ Q_t =& \boldsymbol{C}\left(\Delta X_t\right) - \Delta X_t,\end{aligned}$$

where $\boldsymbol{C}(\cdot)$ is the compression operator.

Proof organization In section A, we first provide the properties of updating rule (11). Sections B and C will next specify Q_t and show the upper bounds for $\|Q_t\|^2$ in Algorithms 1 and 2 respectively.

Notations We define some additional notations throughout the following proof

$$\begin{aligned}\overline{X}_t :=& X_t \frac{\mathbf{1}}{n} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_t^{(i)},\\ \overline{Q}_t :=& Q_t \frac{\mathbf{1}}{n} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{q}_t^{(i)},\\ \overline{G}(X_t;\xi_t) :=& G(X_t,\xi_t)\frac{\mathbf{1}}{n} = \frac{1}{n}\sum_{i=1}^{n} \nabla F_i(\boldsymbol{x}_t^{(i)};\xi_t^{(i)}).\end{aligned}$$

A General bound with compression noise

In this section, to see the influence of the compression more clearly, we are going to prove two general bounds (see see Lemma 7 and Lemma 8) for compressed D-PSGD that has the same updating rule like (11). Those bounds are very helpful for the following proof of our algorithms.

The most challenging part of a decentralized algorithm, unlike the centralized algorithm, is that we need to ensure the local model on each node to converge to the average value $\overline{X}_t$. So we start with an analysis of the quantity $\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2$ and its influence on the final convergence rate. For both ECD-PSGD and DCD-PSGD, we are going to prove that

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \leq \frac{2}{1-\rho^2}\sum_{t=1}^{T}\|Q_t\|_F^2 + \frac{2}{(1-\rho)^2}\sum_{t=1}^{T}\gamma_t^2\|G(X_t;\xi_t)\|_F^2,$$

and

$$\mathbb{E}\|\nabla f(\overline{X}_t)\|^2 + (1-L\gamma_t)\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 \leq \frac{2}{\gamma_t}\left(\mathbb{E}f(\overline{X}_t) - \mathbb{E}f(\overline{X}_{t+1})\right) + \frac{L^2}{n}\mathbb{E}\sum_{i=1}^{n}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 + \frac{L}{\gamma_t}\mathbb{E}\|\overline{Q}_t\|^2 + \frac{L\gamma_t}{n}\sigma^2.$$

From the above two inequalities, we can see that the extra noise term decodes the convergence efficiency of $\boldsymbol{x}_t^{(i)}$ to the average $\overline{X}_t$.

The proof of the general bound for (11) is divided into two parts. In subsection A.1, we provide a new perspective in understanding decentralization, which can be very helpful for the simplicity of our following proof. In subsection A.2, we give the detail proof for the general bound.

A.1 A more intuitive way to understand decentralization

To have a better understanding of how decentralized algorithms work, and how can we ensure a consensus from all local variables on each node. We provide a new perspective to understand decentralization using coordinate transformation, which can simplify our analysis in the following.

The confusion matrix W satisfies $W = \sum_{i=1}^{n}\lambda_i \boldsymbol{v}^i\left(\boldsymbol{v}^i\right)^\top$ is doubly stochastic, so we can decompose it into $W = P\Lambda P^\top$, where $P = \left(\boldsymbol{v}^1, \boldsymbol{v}^2, \cdots, \boldsymbol{v}^n\right)$ that satisfies $P^\top P = PP^\top = I$. Without the loss of generality, we can assume $\lambda_1 \geq \lambda_2 \geq \cdots, \geq \lambda_n$. Then we have the following equalities:

$$\begin{aligned}
X_{t+1} =& X_t W - \gamma_t G(X_t;\xi_t) + Q_t,\\
X_{t+1} =& X_t P\Lambda P^\top - \gamma_t G(X_t;\xi_t) + Q_t,\\
X_{t+1}P =& X_t P\Lambda - G(X_t;\xi_t)P + Q_t P.
\end{aligned}$$

Consider the coordinate transformation using P as the base change matrix, and denote $Y_t = X_t P$, $H(X_t;\xi_t) = G(X_t;\xi_t)P$, $R_t = Q_t P$. Then the above equation can be rewritten as

$$Y_{t+1} = Y_t\Lambda - H(X_t;\xi_t) + R_t, . \tag{12}$$

Since Λ is a diagonal matrix, so we use $\boldsymbol{y}_t^{(i)}$, $\boldsymbol{h}_t^{(i)}$, $\boldsymbol{r}_t^{(i)}$ to indicate the i-th column of Y_t, $H(X_t;\xi_t)$, R_t. Then (12) becomes

$$\boldsymbol{y}_{t+1}^{(i)} = \lambda_i \boldsymbol{y}_t^{(i)} - \boldsymbol{h}_t^{(i)} + \boldsymbol{r}_t^{(i)}, \quad \forall i \in \{1, \cdots, n\}. \tag{13}$$

(13) offers us a much intuitive way to analysis the algorithm. Since all eigenvalues of W, except λ_1, satisfies $|\lambda_i| < 1$, so the corresponding $\boldsymbol{y}_t^{(i)}$ would "decay to zero" due to the scaling factor λ_i.

Moreover, since the eigenvector corresponding to λ_1 is $\frac{1}{\sqrt{n}}(1, 1, \cdots, 1)$, then we have $\boldsymbol{y}_t^{(1)} = \overline{X}_t\sqrt{n}$. So, if $t \to \infty$, intuitively we can set $\boldsymbol{y}_t^{(i)} \to \mathbf{0}$ for $i \neq 1$, then $Y_t \to (\overline{X}_t\sqrt{n}, 0, \cdots, 0)$ and $X_t \to \overline{X}_t\frac{\mathbf{1}}{n}$. This whole process shows how the confusion works under a coordinate transformation.

A.2 Analysis for the general updating form in (11)

Lemma 5. *For any matrix $X_t \in \mathbb{R}^{N\times n}$, decompose the confusion matrix W as $W = \sum_{i=1}^n \lambda_i \boldsymbol{v}^{(i)} \left(\boldsymbol{v}^{(i)}\right)^\top = P\Lambda P^\top$, where $P = (\boldsymbol{v}^{(1)}, \boldsymbol{v}^{(2)}, \cdots, \boldsymbol{v}^{(n)}) \in \mathbb{R}^{N\times n}$, $\boldsymbol{v}^{(i)}$ is the normalized eigenvector of λ_i, and Λ is a diagonal matrix with λ_i be its ith element. We have*

$$\sum_{i=1}^n \left\| X_t W^t \boldsymbol{e}^{(i)} - X_t \frac{\mathbf{1}_n}{n} \right\|^2 = \left\| X_t W^t - X_t \boldsymbol{v}^{(1)} \left(\boldsymbol{v}^{(1)}\right)^\top \right\|_F^2 \le \left\| \rho^t X_t \right\|_F^2,$$

where ρ follows the defination in Theorem 1.

Proof. Since $W^t = P\Lambda^t P^\top$, we have

$$\begin{aligned}
\sum_{i=1}^n \left\| X_t W^t \boldsymbol{e}^{(i)} - X_t \frac{\mathbf{1}_n}{n} \right\|^2 &= \sum_{i=1}^n \left\| \left(X_t W^t - X_t \frac{\mathbf{1}_n \mathbf{1}_n^\top}{n} \right) \boldsymbol{e}^{(i)} \right\|^2 \\
&= \left\| X_t W^t - X_t \boldsymbol{v}^{(1)} \left(\boldsymbol{v}^{(1)}\right)^\top \right\|_F^2 \\
&= \left\| X_t P \Lambda^t P^\top - X_t P \begin{pmatrix} 1, 0, \cdots, 0 \\ 0, 0, \cdots, 0 \\ \cdots \\ 0, 0, \cdots, 0 \end{pmatrix} P^\top \right\|_F^2 \\
&= \left\| X_t P \Lambda^t - X_t P \begin{pmatrix} 1, 0, \cdots, 0 \\ 0, 0, \cdots, 0 \\ \cdots \\ 0, 0, \cdots, 0 \end{pmatrix} \right\|_F^2 \\
&= \left\| X_t P \begin{pmatrix} 0, & 0, & 0, & \cdots, & 0 \\ 0, & \lambda_2^t, & 0, & \cdots, & 0 \\ 0, & 0, & \lambda_3^t, & \cdots, & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0, & 0, & 0, & \cdots, & \lambda_n^t \end{pmatrix} \right\|_F^2 \\
&\le \left\| \rho^t X_t P \right\|_F^2 \\
&= \left\| \rho^t X_t \right\|_F^2.
\end{aligned}$$

Specifically, when $t = 0$, we have

$$\begin{aligned}
\sum_{i=1}^n \left\| X_t \boldsymbol{e}^{(i)} - X_t \frac{\mathbf{1}_n}{n} \right\|^2 &= \left\| X_t P \begin{pmatrix} 0, & 0, & 0, & \cdots, & 0 \\ 0, & 1, & 0, & \cdots, & 0 \\ 0, & 0, & 1, & \cdots, & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0, & 0, & 0, & \cdots, & 1 \end{pmatrix} \right\|_F^2 \\
&= \sum_{i=2}^n \left\| \boldsymbol{y}_t^{(i)} \right\|^2,
\end{aligned} \tag{14}$$

where $\boldsymbol{y}_t^{(i)} = X_t P \boldsymbol{e}^{(i)}$. $\square$

Lemma 6. *Given two non-negative sequences* $\{a_t\}_{t=1}^{\infty}$ *and* $\{b_t\}_{t=1}^{\infty}$ *that satisfying*

$$a_t = \sum_{s=1}^{t} \rho^{t-s} b_s, \tag{15}$$

with $\rho \in [0,1)$, *we have*

$$S_k := \sum_{t=1}^{k} a_t \le \sum_{s=1}^{k} \frac{b_s}{1-\rho},$$
$$D_k := \sum_{t=1}^{k} a_t^2 \le \frac{1}{(1-\rho)^2} \sum_{s=1}^{k} b_s^2.$$

Proof. From the definition, we have

$$S_k = \sum_{t=1}^{k}\sum_{s=1}^{t} \rho^{t-s} b_s = \sum_{s=1}^{k}\sum_{t=s}^{k} \rho^{t-s} b_s = \sum_{s=1}^{k}\sum_{t=0}^{k-s} \rho^{t} b_s \le \sum_{s=1}^{k} \frac{b_s}{1-\rho}, \tag{16}$$

$$\begin{aligned}
D_k &= \sum_{t=1}^{k}\sum_{s=1}^{t} \rho^{t-s} b_s \sum_{r=1}^{t} \rho^{t-r} b_r \\
&= \sum_{t=1}^{k}\sum_{s=1}^{t}\sum_{r=1}^{t} \rho^{2t-s-r} b_s b_r \\
&\le \sum_{t=1}^{k}\sum_{s=1}^{t}\sum_{r=1}^{t} \rho^{2t-s-r} \frac{b_s^2 + b_r^2}{2} \\
&= \sum_{t=1}^{k}\sum_{s=1}^{t}\sum_{r=1}^{t} \rho^{2t-s-r} b_s^2 \\
&\le \frac{1}{1-\rho} \sum_{t=1}^{k}\sum_{s=1}^{t} \rho^{t-s} b_s^2 \\
&\le \frac{1}{(1-\rho)^2} \sum_{s=1}^{k} b_s^2. \quad \text{(due to (16))}
\end{aligned}$$

□

Lemma 5 shows us an overall understanding about how the confusion matrix works, while Lemma 6 is a very important tool for analyzing the sequence in Lemma 5. Next we are going to give a upper bound for the difference between the local modes and the global mean mode.

Lemma 7. *Under Assumption 1, we have*

$$\sum_{t=1}^{T}\sum_{i=1}^{n} \mathbb{E}\left\| \overline{X}_t - \boldsymbol{x}_t^{(i)} \right\|^2 \le \frac{2}{1-\rho^2} \sum_{t=1}^{T} \|Q_s\|_F^2 + \frac{2}{(1-\rho)^2} \sum_{t=1}^{T} \gamma_t^2 \|G(X_t;\xi_t)\|_F^2.$$

Proof. From (6), we have

$$X_t = \sum_{s=1}^{t-1} \gamma_s G\left(X_s;\xi_s\right) W^{t-s-1} + \sum_{s=1}^{t-1} Q_s W^{t-s},$$

$$
\begin{aligned}
\overline{X}_t &= \sum_{s=1}^{t-1} \gamma_s G\left(X_s;\xi_s\right) W^{t-s-1}\frac{\mathbf{1}}{n} + \sum_{s=1}^{t-1} Q_s W^{t-s}\frac{\mathbf{1}}{n} \\
&= \sum_{s=1}^{t-1} \gamma_s \overline{G}\left(X_s;\xi_s\right) + \sum_{s=1}^{t-1} \overline{Q}_s. \quad (\text{due to } W\frac{\mathbf{1}}{n} = \frac{\mathbf{1}}{n})
\end{aligned}
$$

Therefore it yields

$$
\begin{aligned}
\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 =& \sum_{i=1}^{n}\mathbb{E}\left\|\sum_{s=1}^{t-1}\left(Q_s W^{t-s}\boldsymbol{e}^{(i)} - \overline{Q}_s\right) - \sum_{s=1}^{t-1}\gamma_s\left(G(X_s;\xi_s)W^{t-s-1}\boldsymbol{e}^{(i)} - \overline{G}(X_s;\xi_s)\right)\right\|^2 \\
\leq& 2\sum_{i=1}^{n}\mathbb{E}\left\|\sum_{s=1}^{t-1}\left(Q_s W^{t-s}\boldsymbol{e}^{(i)} - \overline{Q}_s\right)\right\|^2 + 2\sum_{i=1}^{n}\mathbb{E}\left\|\sum_{s=1}^{t-1}\gamma_s\left(G(X_s;\xi_s)W^{t-s-1}\boldsymbol{e}^{(i)} - \overline{G}(X_s;\xi_s)\right)\right\|^2 \\
=& 2\sum_{i=1}^{n}\mathbb{E}\left\|\sum_{s=1}^{t-1}\left(Q_s W^{t-s}\boldsymbol{e}^{(i)} - \overline{Q}_s\right)\right\|^2 + 2\sum_{i=1}^{n}\mathbb{E}\left\|\sum_{s=1}^{t-1}\gamma_s\left(G(X_s;\xi_s)W^{t-s-1}\boldsymbol{e}^{(i)} - \overline{G}(X_s;\xi_s)\right)\right\|^2 \\
=& 2\sum_{i=1}^{n}\sum_{s=1}^{t-1}\mathbb{E}\left\|\left(Q_s W^{t-s}\boldsymbol{e}^{(i)} - \overline{Q}_s\right)\right\|^2 + 2\sum_{i=1}^{n}\mathbb{E}\left\|\sum_{s=1}^{t-1}\gamma_s\left(G(X_s;\xi_s)W^{t-s-1}\boldsymbol{e}^{(i)} - \overline{G}(X_s;\xi_s)\right)\right\|^2 \\
&+ 4\sum_{i=1}^{n}\sum_{s\neq s'}\mathbb{E}\left\langle \mathbb{E}_{Q_s} Q_s W^{t-s}\boldsymbol{e}^{(i)} - \mathbb{E}_{Q_s}\overline{Q}_s, \mathbb{E}_{Q_{s'}} Q_{s'} W^{t-s'} e^{(i)} - \mathbb{E}_{Q_{s'}}\overline{Q}_{s'}\right\rangle \\
(\text{due to } \mathbb{E}_{Q_s} Q_s = \mathbf{0}_{n\times n}) \quad =& 2\sum_{i=1}^{n}\sum_{s=1}^{t-1}\mathbb{E}\left\|\left(Q_s W^{t-s}\boldsymbol{e}^{(i)} - \overline{Q}_s\right)\right\|^2 + 2\sum_{i=1}^{n}\mathbb{E}\left\|\sum_{s=1}^{t-1}\gamma_s\left(G(X_s;\xi_s)W^{t-s-1}\boldsymbol{e}^{(i)} - \overline{G}(X_s;\xi_s)\right)\right\|^2 \\
=& 2\sum_{s=1}^{t-1}\mathbb{E}\left\|\left(Q_s W^{t-s} - Q_s \boldsymbol{v}_1\boldsymbol{v}_1^\top\right)\right\|_F^2 + 2\mathbb{E}\left\|\sum_{s=1}^{t-1}\gamma_s\left(G(X_s;\xi_s)W^{t-s-1} - G(X_s;\xi_s)\boldsymbol{v}_1\boldsymbol{v}_1^\top\right)\right\|_F^2 \\
\leq& 2\mathbb{E}\sum_{s=1}^{t-1}\left\|\rho^{t-s}Q_s\right\|_F^2 + 2\mathbb{E}\left(\sum_{s=1}^{t-1}\gamma_s\rho^{t-s-1}\left\|G(X_s;\xi_s)\right\|_F\right)^2, \quad (\text{due to Lemma 5})
\end{aligned}
$$

We can see that $\sum_{s=1}^{t-1}\rho^{2(t-s)}\left\|Q_s\right\|_F^2$ and $\sum_{s=1}^{t-1}\gamma_s\rho^{t-s-1}\left\|G(X_s;\xi_s)\right\|_F$ has the same structure with the sequence in Lemma 6, which leads to

$$
\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \leq \frac{2}{1-\rho^2}\mathbb{E}\sum_{t=1}^{T}\|Q_s\|_F^2 + \frac{2}{(1-\rho)^2}\mathbb{E}\sum_{t=1}^{T}\gamma_t^2\|G(X_t;\xi_t)\|_F^2.
$$

□

Lemma 8. *Following the Assumption 1, we have*

$$
\begin{aligned}
\frac{\gamma_t}{2}\mathbb{E}\|\nabla f(\overline{X}_t)\|^2 + (\frac{\gamma_t}{2} - \frac{L\gamma_t^2}{2})\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 \leq& \mathbb{E}f(\overline{X}_t) - \mathbb{E}f(\overline{X}_{t+1}) + \frac{L^2\gamma_t}{2n}\mathbb{E}\sum_{i=1}^{n}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \\
&+ \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2 + \frac{L\gamma_t^2}{2n}\sigma^2. \qquad (17)
\end{aligned}
$$

Proof. From the updating rule, we have

$$
X_{t+1} = \tilde{X}_t W - \gamma_t G(X_t;\xi_t) = X_t W + Q_t W - \gamma_t G(X_t;\xi_t),
$$

which implies

$$\begin{aligned}\overline{X}_{t+1} =& \left(X_t W + Q_t W - \gamma_t G(X_t;\xi_t)\right)\frac{\mathbf{1}}{n}\\ =& \frac{X_t\mathbf{1}}{n} + \frac{Q_t\mathbf{1}}{n} - \gamma_t\frac{G(X_t;\xi_t)\mathbf{1}}{n}\\ =& \overline{X}_t + \overline{Q}_t - \gamma_t\overline{G}(X_t;\xi_t).\end{aligned}$$

From the Lipschitzian condition for the objective function f_i, we know that f also satisfies the Lipschitzian condition. Then we have

$$\begin{aligned}\mathbb{E}f(\overline{X}_{t+1}) \leq& \mathbb{E}f(\overline{X}_t) + \mathbb{E}\left\langle\nabla f(\overline{X}_t), -\gamma_t\overline{G}(X_t;\xi_t) + \overline{Q}_t\right\rangle + \frac{L}{2}\mathbb{E}\left\|-\gamma_t\overline{G}(X_t;\xi_t) + \overline{Q}_t\right\|^2\\ =& \mathbb{E}f(\overline{X}_t) + \mathbb{E}\langle\nabla f(\overline{X}_t), -\gamma_t\overline{G}(X_t;\xi_t) + \mathbb{E}_{Q_t}\overline{Q}_t\rangle\\ &+ \frac{L}{2}(\mathbb{E}\|\gamma_t\overline{G}(X_t;\xi_t)\|^2 + \mathbb{E}\|\overline{Q}_t\|^2 + \mathbb{E}\langle-\gamma_t\overline{G}(X_t;\xi_t), \mathbb{E}_{Q_t}\overline{Q}_t\rangle)\\ =& \mathbb{E}f(\overline{X}_t) + \mathbb{E}\langle\nabla f(\overline{X}_t), -\gamma_t\mathbb{E}_{\xi_t}\overline{G}(X_t;\xi_t)\rangle + \frac{L\gamma_t^2}{2}\mathbb{E}\|\overline{G}(X_t;\xi_t)\|^2 + \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2 \quad \text{(due to } \mathbb{E}_{Q_t}\overline{Q}_t = \mathbf{0}\text{)}\\ =& \mathbb{E}f(\overline{X}_t) - \gamma_t\mathbb{E}\langle\nabla f(\overline{X}_t), \overline{\nabla f}(X_t)\rangle + \frac{L\gamma_t^2}{2}\mathbb{E}\|(\overline{G}(X_t;\xi_t) - \overline{\nabla f}(X_t)) + \overline{\nabla f}(X_t)\|^2 + \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2\\ =& \mathbb{E}f(\overline{X}_t) - \gamma_t\mathbb{E}\langle\nabla f(\overline{X}_t), \overline{\nabla f}(X_t)\rangle + \frac{L\gamma_t^2}{2}\mathbb{E}\|\overline{G}(X_t;\xi_t) - \overline{\nabla f}(X_t)\|^2 + \frac{L\gamma_t^2}{2}\mathbb{E}\|\overline{\nabla f}(X_t)\|^2\\ &+ L\gamma_t^2\mathbb{E}\langle\mathbb{E}_{\xi_t}\overline{G}(X_t;\xi_t) - \overline{\nabla f}(X_t), \overline{\nabla f}(X_t)\rangle + \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2\\ =& \mathbb{E}f(\overline{X}_t) - \gamma_t\mathbb{E}\langle\nabla f(\overline{X}_t), \overline{\nabla f}(X_t)\rangle + \frac{L\gamma_t^2}{2}\mathbb{E}\|\overline{G}(X_t;\xi_t) - \overline{\nabla f}(X_t)\|^2 + \frac{L\gamma_t^2}{2}\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 + \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2\\ =& \mathbb{E}f(\overline{X}_t) - \gamma_t\mathbb{E}\langle\nabla f(\overline{X}_t), \overline{\nabla f}(X_t)\rangle + \frac{L\gamma_t^2}{2n^2}\mathbb{E}\left\|\sum_{i=1}^n\left(\nabla F_i(x_t^{(i)};\xi_t^{(i)}) - \nabla f_i(x_t^{(i)})\right)\right\|^2\\ &+ \frac{L\gamma_t^2}{2}\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 + \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2\\ =& \mathbb{E}f(\overline{X}_t) - \gamma_t\mathbb{E}\langle\nabla f(\overline{X}_t), \overline{\nabla f}(X_t)\rangle + \frac{L\gamma_t^2}{2n^2}\sum_{i=1}^n\mathbb{E}\left\|\nabla F_i(x_t^{(i)};\xi_t^{(i)}) - \nabla f_i(x_t^{(i)})\right\|^2\\ &+ \sum_{i\neq i'}^n\mathbb{E}\left\langle\mathbb{E}_{\xi_t}\nabla F_i(x_t^{(i)};\xi_t^{(i)}) - \nabla f_i(x_t^{(i)}), \nabla\mathbb{E}_{\xi_t}F_{i'}(x_t^{(i')};\xi_t^{(i')}) - \nabla f_{i'}(x_t^{(i')})\right\rangle\\ &+ \frac{L\gamma_t^2}{2}\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 + \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2\\ \leq& \mathbb{E}f(\overline{X}_t) - \gamma_t\mathbb{E}\langle\nabla f(\overline{X}_t), \overline{\nabla f}(X_t)\rangle + \frac{L\gamma_t^2}{2n}\sigma^2 + \frac{L\gamma_t^2}{2}\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 + \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2\\ =& \mathbb{E}f(\overline{X}_t) - \frac{\gamma_t}{2}\mathbb{E}\|\nabla f(\overline{X}_t)\|^2 - \frac{\gamma_t}{2}\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 + \frac{\gamma_t}{2}\mathbb{E}\|\nabla f(\overline{X}_t) - \overline{\nabla f}(X_t)\|^2 + \frac{L\gamma_t^2}{2}\mathbb{E}\|\overline{\nabla f}(X_t)\|^2\\ &+ \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2 + \frac{L\gamma_t^2}{2n}\sigma^2 \quad \text{(due to } 2\langle\boldsymbol{a},\boldsymbol{b}\rangle = \|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2 - \|\boldsymbol{a}-\boldsymbol{b}\|^2\text{)}\\ =& \mathbb{E}f(\overline{X}_t) - \frac{\gamma_t}{2}\mathbb{E}\|\nabla f(\overline{X}_t)\|^2 - (\frac{\gamma_t}{2} - \frac{L\gamma_t^2}{2})\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 + \frac{\gamma_t}{2}\mathbb{E}\|\nabla f(\overline{X}_t) - \overline{\nabla f}(X_t)\|^2\\ &+ \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2 + \frac{L\gamma_t^2}{2n}\sigma^2.\end{aligned}\tag{18}$$

To estimate the upper bound for $\mathbb{E}\|\nabla f(\overline{X}_t) - \overline{\nabla f}(X_t)\|^2$, we have

$$
\begin{aligned}
\mathbb{E}\|\nabla f(\overline{X}_t) - \overline{\nabla f}(X_t)\|^2 =& \frac{1}{n^2}\mathbb{E}\left\|\sum_{i=1}^{n}\left(\nabla f_i(\overline{X}_t) - \nabla f_i(\boldsymbol{x}_t^{(i)})\right)\right\|^2 \\
\leq& \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left\|\nabla f_i(\overline{X}_t) - \nabla f_i(\boldsymbol{x}_t^{(i)})\right\|^2 \\
\leq& \frac{L^2}{n}\mathbb{E}\sum_{i=1}^{n}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2.
\end{aligned} \tag{19}
$$

Combining (18) and (19) together, we have

$$
\begin{aligned}
\frac{\gamma_t}{2}\mathbb{E}\|\nabla f(\overline{X}_t)\|^2 + \left(\frac{\gamma_t}{2} - \frac{L\gamma_t^2}{2}\right)\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 \leq& \mathbb{E}f(\overline{X}_t) - \mathbb{E}f(\overline{X}_{t+1}) + \frac{L^2\gamma_t}{2n}\mathbb{E}\sum_{i=1}^{n}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \\
&+ \frac{L}{2}\mathbb{E}\|\overline{Q}_t\|^2 + \frac{L\gamma_t^2}{2n}\sigma^2.
\end{aligned}
$$

which completes the proof. $\square$

B Analysis for Algorithm 1

We are going to prove that by using (3) and (4) in Algorithm 1, the upper bound for the compression noise would be

$$
\mathbb{E}\|Q_t\|_F^2 \leq \frac{n\tilde{\sigma}^2}{t}.
$$

Therefore, combing this with Lemma 8 and Lemma 7, we would be able to prove the convergence rate for ALgorithm 1.

$$
\begin{aligned}
\sum_{t=1}^{T}\sum_{i=1}^{n}\left(1 - 6nC_1L^2\gamma_t^2\right)\mathbb{E}\left\|\overline{X}_t - x_t^{(i)}\right\|^2 \leq& \frac{2n^2\tilde{\sigma}^2}{1-\rho^2}\log T + 2\left(\sigma^2 + 3\zeta^2\right)n^2C_1\sum_{t=1}^{T-1}\gamma_t^2 \\
&+ 6C_1n^2\sum_{t=1}^{T-1}\mathbb{E}\gamma_t^2\left\|\nabla f(\overline{X}_t)\right\|^2,
\end{aligned}
$$

which ensures that all nodes would converge to the same value.

Lemma 9. *For any non-negative sequences $\{a_n\}_{n=1}^{+\infty}$ and $\{b_n\}_{n=1}^{+\infty}$ that satisfying*

$$
\begin{aligned}
&a_t = \left(1 - \frac{2}{t}\right)^2 a_{t-1} + \frac{4}{t^2}b_t \\
&b_t \leq \frac{\tilde{\sigma}^2}{2} \quad \forall t \in \{1, 2, 3, \cdots\} \\
&a_1 = 0,
\end{aligned}
$$

we have

$$
a_t \leq \frac{\tilde{\sigma}^2}{t}.
$$

Proof. We use induction to prove the lemma. Since $a_1 = 0 \leq \tilde{\sigma}^2$, suppose the lemma holds for $t \leq k$, which means $a_t \leq \frac{\tilde{\sigma}^2}{t}$, for $\forall t \leq k$. Then it leads to

$$\begin{aligned}
a_{k+1} =& \left(1 - \frac{2}{k}\right)^2 a_k + \frac{4}{k^2} b_t \\
\leq& \left(1 - \frac{2}{k}\right)^2 a_k + \frac{2\tilde{\sigma}^2}{k^2} \\
\leq& \left(1 - \frac{4}{k} + \frac{4}{k^2}\right) \frac{\tilde{\sigma}^2}{k} + \frac{2\tilde{\sigma}^2}{k^2} \\
=& \tilde{\sigma}^2 \left(\frac{1}{k} - \frac{1}{k+1}\right) - \frac{2\tilde{\sigma}^2}{k^2} + \frac{4\tilde{\sigma}^2}{k^3} + \frac{\tilde{\sigma}^2}{k+1} \\
=& \frac{\tilde{\sigma}^2}{k^2(k+1)} \left(k - 2(k+1) + \frac{4}{k}\right) + \frac{\tilde{\sigma}^2}{k+1} \\
=& \frac{\tilde{\sigma}^2}{k^2(k+1)} \left(-k - 2 + \frac{4}{k}\right) + \frac{\tilde{\sigma}^2}{k+1} \\
\leq& \frac{\tilde{\sigma}^2}{k+1}. \quad \text{(due to } k \geq 2\text{)}
\end{aligned}$$

It completes the proof. □

Lemma 10. *Under the Assumption 1, when using Algorithm 1, we have*

$$\begin{aligned}
\mathbb{E}\left\|G(X_t, \xi_t)\right\|_F^2 \leq& n\sigma^2 + 3L^2 \sum_{i=1}^n \mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 + 3n\zeta^2 + 3n\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2, \\
\mathbb{E}\left\|Q_t\right\|_F^2 \leq& \frac{n\tilde{\sigma}^2}{t}, \\
\mathbb{E}\left\|\overline{Q}_t\right\|_F^2 \leq& \frac{\tilde{\sigma}^2}{nt}.
\end{aligned}$$

Proof. Notice that

$$\mathbb{E}\left\|G(X_t, \xi_t)\right\|_F^2 = \sum_{i=1}^n \mathbb{E}\left\|\nabla F_i(x_t^{(i)}; \xi_t^{(i)})\right\|^2.$$

We next estimate the upper bound of $\mathbb{E}\left\|\nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)})\right\|^2$ in the following

$$\begin{aligned}
\mathbb{E}\left\|\nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)})\right\|^2 =& \mathbb{E}\left\|\left(\nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)}) - \nabla f_i(\boldsymbol{x}_t^{(i)})\right) + \nabla f_i(\boldsymbol{x}_t^{(i)})\right\|^2 \\
=& \mathbb{E}\left\|\nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)}) - \nabla f_i(\boldsymbol{x}_t^{(i)})\right\|^2 + \mathbb{E}\left\|\nabla f_i(\boldsymbol{x}_t^{(i)})\right\|^2 \\
&+ 2\mathbb{E}\left\langle \mathbb{E}_{\xi_t} \nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)}) - \nabla f_i(\boldsymbol{x}_t^{(i)}), \nabla f_i(\boldsymbol{x}_t^{(i)})\right\rangle \\
=& \mathbb{E}\left\|\nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)}) - \nabla f_i(\boldsymbol{x}_t^{(i)})\right\|^2 + \mathbb{E}\left\|\nabla f_i(\boldsymbol{x}_t^{(i)})\right\|^2 \\
\leq& \sigma^2 + \mathbb{E}\left\|\left(\nabla f_i(\boldsymbol{x}_t^{(i)}) - \nabla f_i(\overline{X}_t)\right) + \left(\nabla f_i(\overline{X}_t) - \nabla f(\overline{X}_t)\right) + \nabla f(\overline{X}_t)\right\|^2 \\
\leq& \sigma^2 + 3\mathbb{E}\left\|\nabla f_i(\boldsymbol{x}_t^{(i)}) - \nabla f_i(\overline{X}_t)\right\|^2 + 3\mathbb{E}\left\|\nabla f_i(\overline{X}_t) - \nabla f(\overline{X}_t)\right\|^2 + 3\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2
\end{aligned}$$

$$\leq \sigma^2 + 3L^2 \mathbb{E} \left\| X_t - \boldsymbol{x}_t^{(i)} \right\|^2 + 3\zeta^2 + 3\mathbb{E} \left\| \nabla f(\overline{X}_t) \right\|^2,$$

which means

$$\mathbb{E} \left\| G(X_t, \xi_t) \right\|^2 \leq \sum_{i=1}^{n} \left\| \nabla F_i(\boldsymbol{x}_t^{(i)}; \xi_t^{(i)}) \right\|^2 \leq n\sigma^2 + 3L^2 \sum_{i=1}^{n} \mathbb{E} \left\| \overline{X}_t - \boldsymbol{x}_t^{(i)} \right\|^2 + 3n\zeta^2 + 3n\mathbb{E} \left\| \nabla f(\overline{X}_t) \right\|^2.$$

Meanwhile, (5) indicates that

$$\mathbb{E} \left\| \boldsymbol{q}_t^{(i)} \right\|^2 \leq \left(1 - \frac{2}{t}\right)^2 \mathbb{E} \left\| \boldsymbol{q}_{t-1}^{(i)} \right\|^2 + \frac{4}{t^2} \frac{\tilde{\sigma}^2}{2}. \tag{20}$$

So applying Lemma 10 into (20), we have

$$\mathbb{E} \left\| \boldsymbol{q}_t^{(i)} \right\|^2 \leq \frac{\tilde{\sigma}^2}{t}.$$

Therefore

$$\begin{aligned}
\mathbb{E} \left\| Q_t \right\|_F^2 =& \sum_{i=1}^{n} \mathbb{E} \left\| \boldsymbol{q}_t^{(i)} \right\|^2 \leq \frac{n\tilde{\sigma}^2}{t}, \quad \left(\text{due to } \mathbb{E} \left\| \boldsymbol{q}_t^{(i)} \right\|^2 \leq \frac{\tilde{\sigma}^2}{t}\right) \\
\mathbb{E} \left\| \overline{Q}_t \right\|_F^2 =& \frac{1}{n^2} \mathbb{E} \left\| \sum_{i=1}^{n} \boldsymbol{q}_t^{(i)} \right\|^2 \\
=& \frac{1}{n^2} \sum_{i=1}^{n} \mathbb{E} \left\| \boldsymbol{q}_t^{(i)} \right\|^2 + \sum_{i \neq i'}^{n} \mathbb{E} \left\langle \boldsymbol{q}_t^{(i)}, \boldsymbol{q}_t^{(i')} \right\rangle \\
\leq& \frac{\tilde{\sigma}^2}{nt}. \quad \left(\text{due to } \mathbb{E}\boldsymbol{q}_t^{(i)} = 0 \text{ for } \forall i \in \{1, \cdots, n\}\right)
\end{aligned}$$

□

Lemma 11. *Under Assumption 1, when using Algorithm 1, we have*

$$\begin{aligned}
\sum_{t=1}^{T} \sum_{i=1}^{n} \left(1 - 6C_1 L^2 \gamma_t^2\right) \mathbb{E} \left\| \overline{X}_t - \boldsymbol{x}_t^{(i)} \right\|^2 \leq& \frac{2n\tilde{\sigma}^2}{1 - \rho^2} \log T + 2\left(\sigma^2 + 3\zeta^2\right) nC_1 \sum_{t=1}^{T-1} \gamma_t^2 \\
&+ 6C_1 n \sum_{t=1}^{T-1} \mathbb{E}\gamma_t^2 \left\| \nabla f(\overline{X}_t) \right\|^2,
\end{aligned} \tag{21}$$

where C_1 is defined in Theorem 1.

Proof. From Lemma 7 and Lemma 10, we have

$$\begin{aligned}
\sum_{i=1}^{n} \sum_{t=1}^{T} \mathbb{E} \left\| \overline{X}_t - \boldsymbol{x}_t^{(i)} \right\|^2 \leq& \frac{2n\tilde{\sigma}^2}{1 - \rho^2} \sum_{t=1}^{T-1} \frac{1}{t} + 2C_1 \sum_{t=1}^{T-1} \gamma_t^2 \mathbb{E} \left\| G(X_t; \xi_t) \right\|^2 \\
\leq& \frac{2n\tilde{\sigma}^2}{1 - \rho^2} \log T + 2\left(n\sigma^2 + 3n\zeta^2\right) C_1 \sum_{t=1}^{T-1} \gamma_t^2 + 6C_1 n \sum_{t=1}^{T-1} \gamma_t^2 \mathbb{E} \left\| \nabla f(\overline{X}_t) \right\|^2 \\
&+ 6C_1 L^2 \sum_{t=1}^{T-1} \sum_{i=1}^{n} \gamma_t^2 \mathbb{E} \left\| \overline{X}_t - \boldsymbol{x}_t^{(i)} \right\|^2. \quad \text{(due to Lemma 10)}
\end{aligned}$$

Rearranging it obtain the following

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\left(1-6C_1L^2\gamma_t^2\right)\mathbb{E}\left\|\overline{X}_t-\boldsymbol{x}_t^{(i)}\right\|^2\leq\frac{2n\tilde{\sigma}^2}{1-\rho^2}\log T+2\left(\sigma^2+3\zeta^2\right)nC_1\sum_{t=1}^{T-1}\gamma_t^2$$
$$+6C_1n\sum_{t=1}^{T-1}\mathbb{E}\gamma_t^2\left\|\nabla f(\overline{X}_t)\right\|^2, \tag{22}$$

which completing the proof. □

Proof to Theorem 1

Proof. Setting $\gamma_t=\gamma$, then from Lemma 8, we have

$$\mathbb{E}\|\nabla f(\overline{X_t})\|^2+(1-L\gamma)\,\mathbb{E}\|\overline{\nabla f}(X_t)\|^2\leq\frac{2}{\gamma}\left(\mathbb{E}f(\overline{X}_t)-f^*-\left(\mathbb{E}f(\overline{X}_{(t+1)})-f^*\right)\right)+\frac{L^2}{n}\sum_{i=1}^{n}\|\overline{X}_t-x_t^{(i)}\|^2$$
$$+\frac{L\tilde{\sigma}^2}{nt\gamma}+\frac{L\gamma}{n}\sigma^2. \tag{23}$$

From Lemma 11, we have

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\left(1-6C_1L^2\gamma^2\right)\mathbb{E}\left\|\overline{X}_t-x_t^{(i)}\right\|^2\leq\frac{2n\tilde{\sigma}^2}{1-\rho^2}\log T+2\left(\sigma^2+3\zeta^2\right)nC_1T\gamma^2$$
$$+6C_1n\sum_{t=1}^{T-1}\mathbb{E}\gamma^2\left\|\nabla f(\overline{X}_t)\right\|^2.$$

If γ is not too large that satisfies $1-6C_1L^2\gamma^2>0$, we have

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t-x_t^{(i)}\right\|^2\leq\frac{1}{1-6C_1L^2\gamma^2}\left(\frac{2n\tilde{\sigma}^2}{1-\rho^2}\log T+2\left(\sigma^2+3\zeta^2\right)nC_1T\gamma^2\right)$$
$$+\frac{6C_1n}{1-6C_1L^2\gamma^2}\sum_{t=1}^{T-1}\mathbb{E}\gamma^2\left\|\nabla f(\overline{X}_t)\right\|^2. \tag{24}$$

Summarizing both sides of (23) and applying (24) yields

$$\begin{aligned}
&\sum_{t=1}^{T}\left(\mathbb{E}\|\nabla f(\overline{X_t})\|^2+(1-L\gamma)\,\mathbb{E}\|\overline{\nabla f}(X_t)\|^2\right)\\
\leq&\frac{2\mathbb{E}f(\overline{X}_1)-2f^*}{\gamma}+\frac{L^2}{n}\sum_{t=1}^{T-1}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t-x_t^{(i)}\right\|^2+\frac{L\log T}{n\gamma}\tilde{\sigma}^2+\frac{LT\gamma}{n}\sigma^2\\
\leq&\frac{2(f(0)-f^*)}{\gamma}+\frac{L\log T}{n\gamma}\tilde{\sigma}^2+\frac{LT\gamma}{n}\sigma^2+\frac{4C_2\tilde{\sigma}^2L^2}{1-\rho^2}\log T\\
&+4L^2C_2\left(\sigma^2+3\zeta^2\right)C_1T\gamma^2+12L^2C_2C_1\sum_{t=1}^{T-1}\mathbb{E}\gamma^2\left\|\nabla f(\overline{X}_t)\right\|^2,
\end{aligned}$$

where $C_2=\frac{1}{1-6C_1L^2\gamma^2}$. It implies

$$\sum_{t=1}^{T}\left((1-C_3)\,\mathbb{E}\|\nabla f(\overline{X}_t)\|^2+C_4\mathbb{E}\|\overline{\nabla f}(X_t)\|^2\right)$$

$$\leq \frac{2(f(0)-f^*)}{\gamma} + \frac{L\log T}{n\gamma}\tilde{\sigma}^2 + \frac{LT\gamma}{n}\sigma^2 + \frac{4C_2\tilde{\sigma}^2L^2}{1-\rho^2}\log T + 4L^2C_2\left(\sigma^2+3\zeta^2\right)C_1T\gamma^2,$$

where $C_3 = 12L^2C_2C_1\gamma^2$ and $C_4 = (1-L\gamma)$. It completes the proof. □

Proof to Corollary 2

Proof. when $\gamma = \frac{1}{6\sqrt{C_2C_1}L + \frac{\sigma}{\sqrt{n}}T^{\frac{1}{2}} + \zeta^{\frac{2}{3}}T^{\frac{1}{3}}}$, we have

$$\begin{aligned}
1-\gamma \geq & 0,\\
1-6C_1L^2\gamma^2 > & 0,\\
12L^2C_2C_1\gamma^2 \leq & \frac{1}{2}.
\end{aligned}$$

So we can remove the $(1-L\gamma)\,\mathbb{E}\|\overline{\nabla f}(X_t)\|^2$ term on the left side of (7) and substitute $12L^2C_2C_1\gamma^2$ with $\frac{1}{2}$, then we have

$$\begin{aligned}
\sum_{t=1}^{T}\frac{1}{2}\mathbb{E}\|\nabla f(\overline{X_t})\|^2 \leq & 2(f(0)-f^*)6\sqrt{C_2C_1}L + \frac{2(f(0)-f^*)\sigma)}{\sqrt{n}}T^{\frac{1}{2}} + 2(f(0)-f^*)\zeta^{\frac{2}{3}}T^{\frac{1}{3}}\\
& + \frac{6\sqrt{C_2C_1}L^2\tilde{\sigma}^2}{n}\log T + \frac{L\log T\sigma\tilde{\sigma}^2}{n\sqrt{n}}T^{\frac{1}{2}} + \frac{L\log T\tilde{\sigma}^2\zeta^{\frac{2}{3}}}{n}T^{\frac{1}{3}}\\
& + \frac{L\sigma T^{\frac{1}{2}}}{\sqrt{n}} + \frac{4C_2\tilde{\sigma}^2L^2\log T}{1-\rho^2}\\
& + \frac{4nL^2C_2\sigma^2C_1T}{36C_1C_2nL^2+\sigma^2T} + \frac{12nL^2C_2\zeta^2C_1T}{36C_1C_2nL^2+\sigma^2T+n\zeta^{\frac{4}{3}}T^{\frac{2}{3}}}.
\end{aligned}$$

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla f(\overline{X_t})\|^2 \leq & \sigma\frac{4(f(0)-f^*)+4L}{\sqrt{nT}} + \zeta^{\frac{2}{3}}\frac{4(f(0)-f^*)+24L^2C_1}{T^{\frac{2}{3}}} + \tilde{\sigma}^2\frac{10L^2\log T}{(1-\rho^2)T}\\
& + \sigma\tilde{\sigma}^2\frac{2L\log T}{n\sqrt{nT}} + \zeta^{\frac{2}{3}}\tilde{\sigma}^2\frac{L\log T}{nT^{\frac{2}{3}}} + \frac{2(f(0)-f^*)L}{T} + \sigma^2\frac{4nL^2C_2C_1}{36nC_1C_2L^2+\sigma^2T},
\end{aligned}$$

which means

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla f(\overline{X_t})\|^2 \lesssim \frac{\sigma}{\sqrt{nT}} + \frac{\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}} + \frac{1}{T} + \frac{\tilde{\sigma}^2\sigma\log T}{n\sqrt{nT}} + \frac{\zeta^{\frac{2}{3}}\tilde{\sigma}^2\log T}{nT^{\frac{2}{3}}} + \frac{\tilde{\sigma}^2\log T}{T}. \tag{25}$$

From Lemma 11, we have

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n}\left(1-6C_1L^2\gamma_t^2\right)\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \leq & \frac{2n\tilde{\sigma}^2\log T}{(1-\rho^2)T} + 2\left(\sigma^2+3\zeta^2\right)nC_1\gamma^2\\
& + 6C_1n\gamma^2\frac{1}{T}\sum_{t=1}^{T-1}\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2.
\end{aligned}$$

Combing it with (25), we have

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \lesssim \frac{n\tilde{\sigma}^2\log T}{T} + \frac{n\sqrt{n}}{T} + \frac{n\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}} + \frac{1}{T^2} + \frac{\tilde{\sigma}^2\sigma\log T}{T^{\frac{3}{2}}} + \frac{\zeta^{\frac{2}{3}}\tilde{\sigma}^2\log T}{T^{\frac{5}{3}}}.$$

□

C Analysis for Algorithm 2

In Algorithm 2, we have

$$Z_t = X_t(W_t - I) - \gamma F\left(X_t;\xi_t\right). \tag{26}$$

We will prove that

$$\sum_{t=1}^{T} E_{qt}\|Q_t\|_F^2 \leq 2\alpha^2\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right)\sum_{t=1}^{T}\gamma_t^2\|G\left(X_t;\xi_t\right)\|_F^2,$$

which leads to

$$\sum_{i=1}^{n}\sum_{t=1}^{T}\left(1-3D_1L^2\gamma_t^2\right)^2\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \leq 2nD_1(\sigma^2+3\zeta^2)\sum_{t=1}^{T}\gamma_t^2 + 6nD_1\sum_{t=1}^{T}\gamma_t^2\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2.$$

D_1, μ and ρ are defined in Theorem 3.

Lemma 12. *Under Assumption 1, when using Algorithm 2, we have*

$$\sum_{t=1}^{T} E_{qt}\|Q_t\|_F^2 \leq 2\alpha^2\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right)\sum_{t=1}^{T}\gamma_t^2\|G\left(X_t;\xi_t\right)\|_F^2$$

when $(1-\rho)^2 - 4\mu^2\alpha^2 > 0$*, where*

$$\mu = \max_{i\in\{2\cdots n\}}|\lambda_i - 1|$$

Proof. In the proof below, we use $[A]^{(i,j)}$ to indicate the (i,j) element of matrix A.

For the noise induced by quantization, we have

$$\boldsymbol{r}_t^{(i)} = R_t\boldsymbol{e}^{(i)} = Q_tP\boldsymbol{e}^{(i)},$$

so

$$\begin{aligned}\|\boldsymbol{r}_t^{(i)}\|^2 =& \boldsymbol{e}^{(i)\top}P^\top Q_t^\top Q_tP\boldsymbol{e}^{(i)}\\ =& \left(\boldsymbol{v}^{(i)}\right)^\top Q_t^\top Q_t\boldsymbol{v}^{(i)}.\end{aligned}$$

As for $Q_t^\top Q_t$, the expectation of non-diagonal elements would be zero because the compression noise on node i is independent on node j, which leads to

$$\mathbb{E}_{q_t}\left[Q_t^\top Q_t\right]^{(i,j)} = \mathbb{E}_{q_t}\sum_{k=1}^{N}Q_t^{(k,i)}Q_t^{(k,j)}$$

$$=\tau_{ij}\sum_{k=1}^{N}\mathbb{E}_{q_t}\left(Q_t^{(k,i)}\right)^2, \quad (\text{due to } \mathbb{E}_{q_t}Q_t^{(k,i)}=0 \text{ for } \forall i\in\{1\cdots n\})$$

where $\tau_{ij}=1$ if $i=j$, else $\tau_{ij}=0$. Then

$$\begin{aligned}\mathbb{E}_{q_t}\|\boldsymbol{r}_t^{(i)}\|^2 =&\mathbb{E}_{q_t}\left(\boldsymbol{v}^{(i)}\right)^\top Q_t^\top Q_t\boldsymbol{v}^{(i)}\\ =&\sum_{j=1}^{n}\sum_{k=1}^{N}\left(\boldsymbol{v}_j^{(i)}\right)^2\mathbb{E}_{q_t}\left(Q_t^{(k,j)}\right)^2,\end{aligned}$$

where $\boldsymbol{v}_j^{(i)}$ is the jth element of $\boldsymbol{v}^{(i)}$. So we have

$$\begin{aligned}\sum_{i=2}^{n}\mathbb{E}_{q_t}\|\boldsymbol{r}_t^{(i)}\|^2 =&\sum_{i=2}^{n}\sum_{j=1}^{n}\sum_{k=1}^{N}\left(\boldsymbol{v}_j^{(i)}\right)^2\mathbb{E}_{q_t}\left(Q_t^{(k,j)}\right)^2\\ \leq&\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{N}\left(\boldsymbol{v}_j^{(i)}\right)^2\mathbb{E}_{q_t}\left(Q_t^{(k,j)}\right)^2\\ \leq&\sum_{j=1}^{n}\sum_{k=1}^{N}\mathbb{E}_{q_t}\left(Q_t^{(k,j)}\right)^2 \quad (\text{due to } \sum_{i=1}^{n}\left(\boldsymbol{v}_j^{(i)}\right)^2=1)\\ =&\mathbb{E}_{q_t}\|Q_t\|_F^2. \end{aligned}\tag{27}$$

$\mathbb{E}_{q_t}\left(Q_t^{(k,j)}\right)^2$ is the noise brought by quantization, which satisfies

$$\begin{aligned}\mathbb{E}_{q_t}\left(Q_t^{(k,j)}\right)^2 \leq&\alpha^2\left(z_t^{(k,j)}\right)^2\\ =&\alpha^2\left([X_t(W-I)]^{(k,j)}+[G(X_t;\xi_t)]^{(k,j)}\right)^2 \quad (\text{due to (26)})\\ =&2\alpha^2\left([X_t(W-I)]^{(k,j)}\right)^2+2\alpha^2\left([G(X_t;\xi_t)]^{(k,j)}\right)^2\\ =&2\alpha^2\left(\left[X_tP(\Lambda-I)P^\top\right]^{(k,j)}\right)^2+2\alpha^2\left([G(X_t;\xi_t)]^{(k,j)}\right)^2\\ =&2\alpha^2\left(\left[Y(\Lambda-I)P^\top\right]^{(k,j)}\right)^2+2\alpha^2\left([G(X_t;\xi_t)]^{(k,j)}\right)^2,\end{aligned}$$

then

$$\begin{aligned}\mathbb{E}_{q_t}\|Q_t\|_F^2\leq&2\alpha^2\left\|Y(\Lambda-I)P^\top\right\|_F^2+2\alpha^2\|G(X_t;\xi_t)\|_F^2\\ =&2\alpha^2\|Y(\Lambda-I)\|_F^2+2\alpha^2\|G(X_t;\xi_t)\|_F^2\\ =&2\alpha^2\left\|Y\begin{pmatrix}0, & 0, & \cdots, & 0\\ 0, & \lambda_2-1, & \cdots, & 0\\ \cdots & \cdots & \cdots & \cdots\\ 0, & 0, & \cdots, & \lambda_n-1\end{pmatrix}\right\|_F^2+2\alpha^2\|G(X_t;\xi_t)\|_F^2\\ \leq&2\alpha^2\sum_{i=2}^{n}(\lambda_i-1)^2\|\boldsymbol{y}_t^{(i)}\|^2+2\alpha^2\|G(X_t;\xi_t)\|_F^2\\ \leq&2\alpha^2\mu^2\sum_{i=2}^{n}\left\|\boldsymbol{y}_t^{(i)}\right\|^2+2\alpha^2\|G(X_t;\xi_t)\|_F^2. \quad (\text{due to } \mu=\max_{i\in\{2\cdots n\}}|\lambda_i-1|)\end{aligned}\tag{28}$$

Together with (27), it comes to

$$\sum_{i=2}^{n} \mathbb{E}_{q_t} \left\| \boldsymbol{r}_t^{(i)} \right\|^2 \leq 2\alpha^2\mu^2 \sum_{i=2}^{n} \left\| \boldsymbol{y}_t^{(i)} \right\|^2 + 2\alpha^2 \left\| G\left(X_t; \xi_t\right) \right\|_F^2 . \tag{29}$$

From (13), we have

$$\begin{aligned} \boldsymbol{y}_t^{(i)} &= \sum_{s=1}^{t-1} \lambda_i^{t-s-1} \left(-\boldsymbol{h}_s^{(i)} + \boldsymbol{r}_s^{(i)} \right), \\ \left\| \boldsymbol{y}_t^{(i)} \right\|^2 &\leq \left(\sum_{s=1}^{t-1} |\lambda_i|^{t-s-1} \left\| -\boldsymbol{h}_s^{(i)} + \boldsymbol{r}_s^{(i)} \right\| \right)^2 . \quad \text{(due to } \|\boldsymbol{a}+\boldsymbol{b}\|^2 \leq \|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2\text{)} \end{aligned}$$

Denote $m_t^{(i)} = \sum_{s=1}^{t-1} |\lambda_i|^{t-1-s} \left\| -\boldsymbol{h}_s^{(i)} + \boldsymbol{r}_s^{(i)} \right\|$, we can see that $m_t^{(i)}$ has the same structure of the sequence in Lemma 6, Therefore

$$\begin{aligned} \sum_{i=2}^{n} \sum_{t=1}^{T} \left(m_t^{(i)} \right)^2 &\leq \sum_{i=2}^{n} \sum_{t=1}^{T} \frac{1}{(1-|\lambda_i|)^2} \left\| -\boldsymbol{h}_t^{(i)} + \boldsymbol{r}_t^{(i)} \right\|^2 \\ &\leq \sum_{i=2}^{n} \sum_{t=1}^{T} \frac{2}{(1-|\lambda_i|)^2} \left(\left\| \boldsymbol{h}_t^{(i)} \right\|^2 + \left\| \boldsymbol{r}_t^{(i)} \right\|^2 \right) \\ &= \sum_{i=2}^{n} \sum_{t=1}^{T} \frac{2}{(1-|\lambda_i|)^2} \left(\gamma_t^2 \left\| G\left(X_t; \xi_t\right) \boldsymbol{v}^{(i)} \right\|^2 + \left\| \boldsymbol{r}_t^{(i)} \right\|^2 \right) \\ &= \sum_{i=2}^{n} \sum_{t=1}^{T} \frac{2\gamma_t^2 \left\| G\left(X_t; \xi_t\right) \boldsymbol{v}^{(i)} \right\|^2}{(1-|\lambda_i|)^2} + \sum_{i=2}^{n} \sum_{t=1}^{T} \frac{2}{(1-|\lambda_i|)^2} \left\| \boldsymbol{r}_t^{(i)} \right\|^2 \\ &\leq \sum_{i=2}^{n} \sum_{t=1}^{T} \frac{2\gamma_t^2 \left\| G\left(X_t; \xi_t\right) \boldsymbol{v}^{(i)} \right\|^2}{(1-|\lambda_i|)^2} + \sum_{i=2}^{n} \sum_{t=1}^{T} \frac{2}{(1-|\lambda_i|)^2} \left\| \boldsymbol{r}_t^{(i)} \right\|^2 \\ &\leq \frac{2}{(1-\rho)^2} \sum_{i=2}^{n} \sum_{t=1}^{T} \gamma_t^2 \left\| G\left(X_t; \xi_t\right) \boldsymbol{v}^{(i)} \right\|^2 + \frac{2}{(1-\rho)^2} \sum_{i=2}^{n} \sum_{t=1}^{T} \left\| \boldsymbol{r}_t^{(i)} \right\|^2 \\ &\leq \frac{2}{(1-\rho)^2} \sum_{i=2}^{n} \sum_{t=1}^{T} \gamma_t^2 \left\| G\left(X_t; \xi_t\right) P \right\|^2 + \frac{2}{(1-\rho)^2} \sum_{i=2}^{n} \sum_{t=1}^{T} \left\| \boldsymbol{r}_t^{(i)} \right\|^2 \\ &\leq \frac{2}{(1-\rho)^2} \sum_{t=1}^{T} \gamma_t^2 \left\| G\left(X_t; \xi_t\right) \right\|_F^2 + \frac{2}{(1-\rho)^2} \sum_{i=2}^{n} \sum_{t=1}^{T} \left\| \boldsymbol{r}_t^{(i)} \right\|^2 . \end{aligned} \tag{30}$$

Combining (29) and (30) together, we have

$$\sum_{i=2}^{n} \sum_{t=1}^{T} \left\| \boldsymbol{y}_t^{(i)} \right\|^2 \leq \sum_{i=2}^{n} \sum_{t=1}^{T} \left(m_t^{(i)} \right)^2 \leq \frac{2(1+2\alpha^2)}{(1-\rho)^2} \sum_{t=1}^{T} \gamma_t^2 \left\| G\left(X_t; \xi_t\right) \right\|_F^2 + \frac{4\mu^2\alpha^2}{(1-\rho)^2} \sum_{l=2}^{n} \sum_{t=1}^{T} \left\| \boldsymbol{y}_t^{(l)} \right\|^2 ,$$

It follows thtat

$$\begin{aligned} \sum_{i=2}^{n} \sum_{t=1}^{T} \left\| \boldsymbol{y}_t^{(i)} \right\|^2 &\leq \frac{2(1+2\alpha^2)}{(1-\rho)^2} \sum_{t=1}^{T} \gamma_t^2 \left\| G\left(X_t; \xi_t\right) \right\|_F^2 + \frac{4\mu^2\alpha^2}{(1-\rho)^2} \sum_{t=1}^{T} \sum_{l=2}^{n} \|\boldsymbol{y}_t^{(l)}\|^2, \\ \left(1 - \frac{4\mu^2\alpha^2}{(1-\rho)^2} \right) \sum_{i=2}^{n} \sum_{t=1}^{T} \left\| \boldsymbol{y}_t^{(i)} \right\|^2 &\leq \frac{2(1+2\alpha^2)}{(1-\rho)^2} \sum_{t=1}^{T} \gamma_t^2 \left\| G\left(X_t; \xi_t\right) \right\|_F^2 . \end{aligned}$$

If α is small enough that satisfies $(1-\rho)^2 - 4\mu^2\alpha^2 > 0$, then we have

$$\sum_{i=2}^{n}\sum_{t=1}^{T}\left\|\boldsymbol{y}_t^{(i)}\right\|^2 \leq \frac{2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}\sum_{t=1}^{T}\gamma_t^2\left\|G\left(X_t;\xi_t\right)\right\|_F^2. \tag{31}$$

Together (31) with (28) , we have

$$\begin{aligned}\sum_{t=1}^{T} E_{qt}\|Q_t\|_F^2 \leq& 2\alpha^2\mu^2\sum_{t=1}^{T}\sum_{l=2}^{n}\|\boldsymbol{y}_t^{(l)}\|^2 + 2\alpha^2\sum_{t=1}^{T}\gamma_t^2\left\|G\left(X_t;\xi_t\right)\right\|_F^2\\ \leq& 2\alpha^2\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right)\sum_{t=1}^{T}\gamma_t^2\|G\left(X_t;\xi_t\right)\|_F^2.\end{aligned}$$

Moreover, setting $\gamma_t = \gamma$ and denote $2\alpha^2\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right) = D_2$. Applying Lemma 10 to bound $\|G\left(X_t;\xi_t\right)\|_F^2$, we have

$$\sum_{t=1}^{T} E_{qt}\|Q_t\|_F^2 \leq n\sigma^2 D_2\gamma^2 T + 3L^2 D_2\gamma^2\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 + 3n\zeta^2 D_2\gamma^2 T + 3nD_2\gamma^2\mathbb{E}\sum_{t=1}^{T}\left\|\nabla f(\overline{X}_t)\right\|^2. \tag{32}$$

□

Lemma 13. *Under Assumption 1, we have*

$$\sum_{i=1}^{n}\sum_{t=1}^{T}\left(1-3D_1L^2\gamma_t^2\right)^2\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \leq 2nD_1(\sigma^2+3\zeta^2)\sum_{t=1}^{T}\gamma_t^2 + 6nD_1\sum_{t=1}^{T}\gamma_t^2\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2.$$

when $1-2\alpha^2\mu^2 C_\lambda > 0$, *where*

$$\begin{aligned}\mu =& \max_{l\in\{2\cdots n\}}|\lambda_l - 1|\\ \rho =& \max_{l\in\{2\cdots n\}}|\lambda_l|\\ D_1 =& \frac{2\alpha^2}{1-\rho^2}\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right)+\frac{1}{(1-\rho)^2}.\end{aligned}$$

Proof. From Lemma 7, we have

$$\begin{aligned}\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \leq& \frac{2}{1-\rho^2}\sum_{t=1}^{T}\|Q_s\|_F^2 + 2\frac{1}{(1-\rho)^2}\sum_{t=1}^{T}\gamma_t^2\|G(X_t;\xi_t)\|_F^2\\ \leq& 2\left(\frac{2\alpha^2}{1-\rho^2}\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right)+\frac{1}{(1-\rho)^2}\right)\sum_{t=1}^{T}\gamma_t^2\|G(X_t;\xi_t)\|^2. \quad \text{(due to Lemma 12)}\end{aligned}$$

Denote $\frac{2\alpha^2}{1-\rho^2}\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right)+\frac{1}{(1-\rho)^2} = D_1$,then from Lemma 10, we have

$$\begin{aligned}\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \leq& 2nD_1(\sigma^2+3\zeta^2)\sum_{t=1}^{T}\gamma_t^2 + 6nD_1\sum_{t=1}^{T}\gamma_t^2\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2\\ &+ 3D_1L^2\sum_{i=1}^{n}\gamma_t^2\mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2,\end{aligned}$$

$$\sum_{t=1}^{T}\left(1-3D_1L^2\gamma_t^2\right)^2\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t-\boldsymbol{x}_t^{(i)}\right\|^2 \leq 2nD_1(\sigma^2+3\zeta^2)\sum_{t=1}^{T}\gamma_t^2+6nD_1\sum_{t=1}^{T}\gamma_t^2\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2.$$

If $1-3D_1L^2\gamma_t^2>0$, then $\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t-\boldsymbol{x}_t^{(i)}\right\|^2$ would be bounded. $\square$

Proof to Theorem 3

Proof. Combining Lemma 8 and (32), we have

$$\begin{aligned}\mathbb{E}\|\nabla f(\overline{X}_t)\|^2+(1-L\gamma)\,\mathbb{E}\|\overline{\nabla f}(X_t)\|^2 \leq & \frac{2}{\gamma}\left(\mathbb{E}f(\overline{X}_t)-f^*-\left(\mathbb{E}f(\overline{X}_{(t+1)})-f^*\right)\right)\\ &+\left(\frac{2L^2}{n}+\frac{3L^3D_2\gamma^2}{2n}\right)\sum_{i=1}^{n}\|\overline{X}_t-x_t^{(i)}\|^2\\ &+\left(\frac{\gamma^2LD_2}{2}+\frac{L\gamma}{n}\right)T\sigma^2+\frac{3LD_2\gamma^2\zeta^2T}{2}\\ &+\frac{3LD_2\gamma^2}{2}\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2. \end{aligned}\tag{33}$$

From Lemma 5, we have

$$\left(1-3D_1L^2\gamma^2\right)^2\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t-\boldsymbol{x}_t^{(i)}\right\|^2 \leq 2nD_1(\sigma^2+3\zeta^2)T\gamma^2+6nD_1\gamma^2\sum_{t=1}^{T}\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2.$$

If γ is not too large that satisfies $1-3D_1L^2\gamma^2>0$, we have

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\mathbb{E}\left\|\overline{X}_t-\boldsymbol{x}_t^{(i)}\right\|^2 \leq \frac{2nD_1(\sigma^2+3\zeta^2)T\gamma^2}{1-3D_1L^2\gamma^2}+\frac{6nD_1\gamma^2}{1-3D_1L^2\gamma^2}\sum_{t=1}^{T}\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2. \tag{34}$$

Summarizing both sides of (33) and applying (34) yields

$$\begin{aligned}&\sum_{t=1}^{T}\left(\mathbb{E}\|\nabla f(\overline{X}_t)\|^2+(1-L\gamma)\,\mathbb{E}\|\overline{\nabla f}(X_t)\|^2\right)\\ \leq&\frac{2(f(0)-f^*)}{\gamma}+\left(\frac{T\gamma^2LD_2}{2}+\frac{TL\gamma}{n}+\frac{\left(4L^2+3L^3D_2\gamma^2\right)D_1T\gamma^2}{1-3D_1L^2\gamma^2}\right)\sigma^2\\ &+\frac{\left(4L^2+3L^3D_2\gamma^2\right)3D_1T\gamma^2}{1-3D_1L^2\gamma^2}\zeta^2+\frac{3LD_2\gamma^2T}{2}\zeta^2\\ &+\left(\frac{\left(4L^2+3L^3D_2\gamma^2\right)3D_1\gamma^2}{1-3D_1L^2\gamma^2}+\frac{3LD_2\gamma^2}{2}\right)\sum_{t=1}^{T}\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2.\end{aligned}$$

It implies

$$\begin{aligned}&\sum_{t=1}^{T}\left(\left(1-\frac{\left(4L^2+3L^3D_2\gamma^2\right)3D_1\gamma^2}{1-3D_1L^2\gamma^2}-\frac{3LD_2\gamma^2}{2}\right)\mathbb{E}\|\nabla f(\overline{X}_t)\|^2+(1-L\gamma)\,\mathbb{E}\|\overline{\nabla f}(X_t)\|^2\right)\\ \leq&\frac{2(f(0)-f^*)}{\gamma}+\left(\frac{T\gamma^2LD_2}{2}+\frac{L\gamma T}{n}+\frac{\left(4L^2+3L^3D_2\gamma^2\right)D_1T\gamma^2}{1-3D_1L^2\gamma^2}\right)\sigma^2\end{aligned}$$

$$+\left(\frac{\left(4L^2+3L^3D_2\gamma^2\right)3D_1T\gamma^2}{1-3D_1L^2\gamma^2}+\frac{3LD_2\gamma^2T}{2}\right)\zeta^2.$$

Denote $D_3=\frac{\left(4L^2+3L^3D_2\gamma^2\right)3D_1\gamma^2}{1-3D_1L^2\gamma^2}+\frac{3LD_2\gamma^2}{2}$ and $D_4=1-L\gamma$, we have

$$\begin{aligned}
&\sum_{t=1}^{T}\left((1-D_3)\,\mathbb{E}\|\nabla f(\overline{X}_t)\|^2+D_4\mathbb{E}\|\overline{\nabla f}(X_t)\|^2\right)\\
\leq&\frac{2(f(0)-f^*)}{\gamma}+\left(\frac{T\gamma^2LD_2}{2}+\frac{L\gamma T}{n}+\frac{\left(4L^2+3L^3D_2\gamma^2\right)D_1T\gamma^2}{1-3D_1L^2\gamma^2}\right)\sigma^2\\
&+\left(\frac{\left(4L^2+3L^3D_2\gamma^2\right)3D_1T\gamma^2}{1-3D_1L^2\gamma^2}+\frac{3LD_2\gamma^2T}{2}\right)\zeta^2.
\end{aligned}$$

It completes the proof. □

Proof to Corollary 4

Proof. Setting $\gamma=\frac{1}{6\sqrt{D_1}L+6\sqrt{D_2L}+\frac{\sigma}{\sqrt{n}}T^{\frac{1}{2}}+\zeta^{\frac{2}{3}}T^{\frac{1}{3}}}$, then we can verify

$$\begin{aligned}
3D_1L^2\gamma^2\leq&\frac{1}{12}\\
3LD_2\gamma^2\leq&\frac{1}{12}\\
D_3\leq&\frac{1}{2}\\
D_4\geq&0.
\end{aligned}$$

So we can remove the $\|\overline{\nabla f}(X_t)\|^2$ on the LHS and substitute $(1-D_3)$ with $\frac{1}{2}$. Therefore (10) becomes

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla f(\overline{X}_t)\|^2\leq&4(f(0)-f^*)\frac{\sigma}{\sqrt{Tn}}+\frac{4L\sigma}{\sqrt{Tn}}+\frac{\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}}(4LD_2+30L^2D_1+4(f(0)-f^*))\\
&+\frac{n\sigma^2}{n(D_5)^2L^2+\sigma^2T}\left(5D_1L^2+\frac{LD_2}{2}\right)+\frac{4(f(0)-f^*)(6\sqrt{D_1}L+6\sqrt{D_2L})}{T}.
\end{aligned}\tag{35}$$

From Lemma 13, we have

$$\begin{aligned}
\frac{1}{T}\sum_{i=1}^{n}\sum_{t=1}^{T}\left(1-3D_1L^2\gamma^2\right)^2\mathbb{E}\left\|\overline{X}_t-\boldsymbol{x}_t^{(i)}\right\|^2\leq&2nD_1(\sigma^2+3\zeta^2)\gamma^2+6nD_1\gamma^2\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2\\
\leq&\frac{2n\sqrt{n}D_1}{T}+\frac{6n\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}}+6nD_1\gamma^2\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2.
\end{aligned}$$

$$\frac{1}{T}\sum_{i=1}^{n}\sum_{t=1}^{T}\mathbb{E}\left\|\overline{X}_t-\boldsymbol{x}_t^{(i)}\right\|^2\leq\frac{4n\sqrt{n}D_1}{T}+\frac{12n\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}}+12nD_1\gamma^2\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left\|\nabla f(\overline{X}_t)\right\|^2.\tag{36}$$

If $\alpha^2\leq\min\left\{\frac{(1-\rho)^2}{8\mu^2},\frac{1}{4}\right\}$, then

$$D_2=2\alpha^2\left(\frac{2\mu^2(1+2\alpha^2)}{(1-\rho)^2-4\mu^2\alpha^2}+1\right)\leq\frac{3\mu^2\alpha^2}{(1-\rho)^2}+2\alpha^2,$$

$$D_1 = \frac{D_2}{1-\rho^2} + \frac{1}{(1-\rho)^2} \leq \frac{D_2+1}{(1-\rho)^2}, \quad \text{(due to } \rho < 1\text{)}$$

which means $D_2 = O\left(\alpha^2\right)$ and $D_1 = O\left(\alpha^2 + 1\right)$.
So (28) becomes

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\|\nabla f(\overline{X}_t)\|^2 \lesssim \frac{\sigma}{\sqrt{nT}} + \frac{\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}} + \frac{1}{T}.$$

Combing the inequality above and (36), we have

$$\frac{1}{T}\sum_{i=1}^{n}\sum_{t=1}^{T} \mathbb{E}\left\|\overline{X}_t - \boldsymbol{x}_t^{(i)}\right\|^2 \lesssim \frac{n\zeta^{\frac{2}{3}}}{T^{\frac{2}{3}}} + \frac{n\sqrt{n}}{T} + \frac{n}{T^2}.$$

$\square$