EFFICIENT DISTRIBUTED ALGORITHMS: BETTER THEORY AND COMMUNICATION COMPRESSION

By

Yao Li

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Applied Mathematics – Doctor of Philosophy
Computational Mathematics, Science and Engineering – Dual Major

2022

ABSTRACT

EFFICIENT DISTRIBUTED ALGORITHMS: BETTER THEORY AND COMMUNICATION COMPRESSION

By

Yao Li

Large-scale machine learning models are often trained by distributed algorithms over either centralized or decentralized networks. The former uses a central server to aggregate the information of local computing agents and broadcast the averaged parameters in a master-slave architecture. The latter considers a connected network formed by all agents. The information can only be exchanged with accessible neighbors with a mixing matrix of communication weights encoding the network's topology.

Compared with centralized optimization, decentralization facilitates data privacy and reduces the communication burden of the single central agent due to model synchronization, but the connectivity of the communication network weakens the theoretical convergence complexity of the decentralized algorithms. Therefore, there are still gaps between decentralized and centralized algorithms in terms of convergence conditions and rates.

In the first part of this dissertation, we consider two decentralized algorithms: EXTRA and NIDS, which both converge linearly with strongly convex objective functions and answer two questions regarding them. *What are the optimal upper bounds for their stepsizes? Do decentralized algorithms require more properties on the functions for linear convergence than centralized ones?* More specifically, we relax the required conditions for linear convergence of both algorithms. For EXTRA, we show that the stepsize is comparable to that of centralized algorithms. For NIDS, the upper bound of the stepsize is shown to be exactly the same as the centralized ones. In addition, we relax the requirement for the objective functions and the mixing matrices. We provide the linear convergence results for both algorithms under the weakest conditions.

As the number of computing agents and the dimension of the model increase, the communication cost of parameter synchronization becomes the major obstacle to efficient learning. Communication compression techniques have exhibited great potential as an antidote to accelerate distributed machine learning by mitigating the communication bottleneck.

In the rest of the dissertation, we propose compressed residual communication frameworks for both centralized and decentralized optimization and design different algorithms to achieve efficient communication.

For centralized optimization, we propose DORE, a modified parallel stochastic gradient descent method with a bidirectional residual compression, to reduce over 95% of the overall communication. Our theoretical analysis demonstrates that the proposed strategy has superior convergence properties for both strongly convex and nonconvex objective functions.

Existing works mainly focus on smooth problems and compressing DGD-type algorithms for decentralized optimization. The class of smooth objective functions and the sublinear convergence rate under relatively strong assumptions limit these algorithms' application and practical performance. Motivated by primal-dual algorithms, we propose Prox-LEAD, a linear convergent decentralized algorithm with compression, to tackle strongly convex problems with a nonsmooth regularizer. Our theory describes the coupled dynamics of the inexact primal and dual update as well as compression error without assuming bounded gradients. The superiority of the proposed algorithm is demonstrated through the comparison with state-of-the-art algorithms in terms of convergence complexities and numerical experiments. Our algorithmic framework also generally enlightens the compressed communication on other primal-dual algorithms by reducing the impact of inexact iterations.

Dedicated to my parents and in memory of my grandmother, a woman of strength, kindness and love, who passed away before seeing me graduate.

ACKNOWLEDGEMENTS

There are many people who encouraged me and shared great times with me throughout my doctoral career. Please accept my apologies if there are any names that have been forgotten.

First of all, I would like to express my deep gratitude to my advisor and dissertation director, Professor Ming Yan. With his dedicated guidance and support over the past five years, I have developed a good taste and a broad vision of research. Again, thanks to my friend-like relationship with him, I could maintain a work-life balance and alleviate the pressure from life and studies. I would also like to thank all the other committee members, Professor Ekaterina Rapinchuk, Professor Kalyanmoy Deb, and Professor Mark Iwen, for their guidance and advice. They provided helpful feedback and suggestions on my path to graduation.

I want to thank my most important research partner of mine, Dr. Xiaorui Liu. We collaborated with each other, discussed and shared insightful research ideas, and completed many good papers. Without him, I would not have expanded my knowledge in deep learning and bridged the gap between theoretic algorithmic research and machine learning applications so quickly. I would also like to thank Dr. Brendt Wohlberg and Dr. Youzuo Lin for their short-term guidance in LANL.

I also want to thank my friends Jing Huang, Jian Song, Mengzhi Chen, Menglun Wang, Ningyu Sha, Kai Huang, Jialin Qu, Runze Su, Rui Wang and Tao Feng for the precious time they spent with me.

Last but not least, I would like to express my greatest gratitude to my parents, Youzhu Li and Yiyan Liu. Their selfless love has always been the greatest motivation for me to complete my doctoral career.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

CHAPTER 1

INTRODUCTION

This chapter briefly overviews the problems considered and the results developed in the following separated chapters. The detailed introduction and claimed notations are contained throughout each chapter.

1.1 Decentralized Optimization

Decentralized optimization problem is to minimize $\bar{f}(x) \coloneqq \frac{1}{n}\sum_{i=1}^{n} f_i(x)$ collaboratively over a network of n agents. We consider convex and differentiable function $f_i \colon \mathbb{R}^p \to \mathbb{R}$, which is known only by the corresponding agent i. The whole system is decentralized because each agent has an estimation of the global variable and can only exchange the estimation with their accessible neighbors during each iteration. A symmetric mixing matrix $\mathbf{W} \in \mathbb{R}^{n\times n}$ is used to encode the communication weights between the agents and enforce the consensus. The minimum condition for $\mathbf{W}$ has one eigenvalue 1 with the all-one vector $\mathbf{1}$ being a corresponding eigenvector. All other eigenvalues of $\mathbf{W}$ are less than 1. We provide new and stronger linear convergence results for two state-of-the-art algorithms: EXTRA in [1] and NIDS in [2]. More specifically, by assuming L-smoothness of each f_i,

- We show the linear convergence of EXTRA under the strong convexity of $\bar{f}$ and the relaxed condition $\lambda_{\min}(\mathbf{W}) > -5/3$. The upper bound of the stepsize can be as large as $\frac{5+3\lambda_{\min}(\mathbf{W})}{4L}$, which is shown to be optimal in [3] for general convex problems;
- We show the linear convergence of NIDS under the same condition on $\bar{f}$ and $\mathbf{W}$ as EXTRA with any network-independent stepsize $\alpha \in (0, 2/L)$.

1.2 Parallel SGD with Bidirectional Communication Compression

We first consider centralized optimization with efficient communication over a parameter server architecture and propose algorithms based on the well-known parallel stochastic gradient descent (SGD) to achieve efficient communication. Stochastic gradient algorithms [4] are efficient at minimizing the objective function $f : \mathbb{R}^d \to \mathbb{R}$ which is usually defined as $f(\mathbf{x}) := \mathbb{E}_{\xi\sim\mathcal{D}}[\ell(\mathbf{x}, \xi)]$, where $\ell(\mathbf{x}, \xi)$ is the objective function defined on data sample ξ and model parameter $\mathbf{x}$. A basic stochastic gradient descent repeats the gradient "descent" step $\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma \mathbf{g}(\mathbf{x}^k)$ where $\mathbf{x}_k$ is the current iteration and γ is the step size. The stochastic gradient $\mathbf{g}(\mathbf{x}^k)$ is computed based on an i.i.d. sampled mini-batch from the training data distribution $\mathcal{D}$ and serves as the estimator of the full gradient $\nabla f(\mathbf{x}^k)$. In large-scale machine learning, the number of data samples and the model size are usually huge. Distributed learning utilizes many computers/cores to perform the stochastic algorithms to reduce the training time. It has attracted extensive attention due to the demand for highly efficient model training [5, 6, 7, 8].

We focus on the data-parallel SGD [9, 10, 11], which provides a scalable solution to speed up the training process by distributing the whole data to multiple computing nodes, and consider the following problem

$$\operatorname*{minimize}_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) + R(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{\mathbb{E}_{\xi\sim\mathcal{D}_i}[\ell(\mathbf{x}, \xi)]}_{:=f_i(\mathbf{x})} + R(\mathbf{x}), \tag{1.1}$$

where each $f_i(\mathbf{x})$ is a local objective function of the worker node i defined based on the allocated data under distribution $\mathcal{D}_i$ and $R : \mathbb{R}^d \to \mathbb{R}$ is usually a closed convex regularizer.

We propose DORE, adapted by compressed communication on both pull and push directions of Parallel SGD, to reduce the bidirectional communication cost in distributed learning and provide a complete theoretical analysis of the algorithm's behavior in both strongly convex and nonconvex settings.

1.3 Decentralized Optimization with Compression

We then return to decentralized optimization and try to generalize the communication compression scheme of DORE over networks without a parameter server. We consider the general composite problems, i.e., the smooth problems with nonsmooth regularizer, over the connected n-node network $\mathcal{G}$ in the form of

$$\underset{\mathbf{x}\in\mathbb{R}^p}{\operatorname{minimize}} \ \frac{1}{n}\sum_{i=1}^{n}\Big(\underbrace{\mathbb{E}_{\xi_i\sim\mathcal{D}_i} f_i(\mathbf{x},\xi_i)}_{=:f_i(\mathbf{x})} + r(\mathbf{x})\Big), \tag{1.2}$$

where $f_i(\mathbf{x},\xi_i)$ is the objective function at node i defined on the data ξ_i sampled from the distribution $\mathcal{D}_i$, and x denotes the model parameters. We use $f_i(\mathbf{x})$ to define the overall local objective function at node i, and we will differentiate $f_i(\mathbf{x},\xi_i)$ and $f_i(\mathbf{x})$ using different inputs. The data distributions $\{\mathcal{D}_i\}$ can be heterogeneous. In other words, the data distributions can be different from node to node, and we do not make assumptions about data heterogeneity. The function $f_i(\mathbf{x},\xi_i)$ is assumed to be convex and smooth, and $r(\mathbf{x})$ is a proper, convex, and possibly nonsmooth function shared across the nodes. The graph $\mathcal{G}$ encodes the topology of the communication network where information is exchanged along the edges.

In recent years, various communication compression techniques, such as quantization and sparsification, have been developed to reduce communication costs. Notably, extensive studies [12, 13, 14, 15, 16, 17, 18, 19] have utilized gradient compression to boost communication efficiency for centralized optimization significantly. They enable efficient large-scale optimization while maintaining comparable convergence rates and practical performance with their non-compressed counterparts. This remarkable success has suggested the potential and significance of communication compression in decentralized algorithms.

While extensive attention has been paid to centralized optimization, communication compression is relatively less studied in decentralized algorithms because the algorithm design and analysis are more challenging to cover general communication topologies.

Recent efforts are trying to push this research direction. For instance, DCD-SGD and ECD-SGD [20] introduce difference compression and extrapolation compression to reduce model compression error. [21, 22] introduce QDGD and QuanTimed-DSGD to have exact convergence with small stepsizes. DeepSqueeze [23] directly compresses the local model and compensates for the compression error in the next iteration. CHOCO-SGD [24, 25] presents a novel quantized gossip algorithm that reduces compression error by difference compression and preserves the model average. Nevertheless, most current works focus on the compression of primal-only algorithms, i.e., reduce to DGD [26, 27] or P-DSGD [28]. They are unsatisfying regarding convergence rate, stability, and the capability to handle heterogeneous data. Part of the reason is that they inherit the drawback of DGD-type algorithms, whose convergence rate is slow in heterogeneous data scenarios where the data distributions are significantly different from agent to agent.

In the literature on decentralized optimization, it has been proved that primal-dual algorithms can achieve faster converge rates and better support heterogeneous data [29, 1, 2, 30]. However, it is unknown whether communication compression is feasible for primal-dual algorithms and how fast the convergence can be with compression. This chapter attempts to bridge this gap by investigating the communication compression for primal-dual decentralized algorithms.

We design a decentralized residual compression scheme and propose a novel decentralized algorithm with compression, Prox-LEAD, to achieve linear convergence under the strongly convex assumption and efficient communication. We derive the convergence complexity of Prox-LEAD for two types of f_i, the general expectation of loss $f_i(\mathbf{x}, \xi_i)$ as defined in (1.2) and the finite-sum setting. We combine Prox-LEAD with Loopless SVRG and SAGA for the second case to achieve the exact linear convergence with stochastic gradients.

1.4 Publications

Chapters 2-4 are based on the following papers respectively:

Published papers:

- [31] Yao Li and Ming Yan. On the linear convergence of two decentralized algorithms. *Journal of Optimization Theory and Applications*, 189(1):271–290, 2021
- [19] Xiaorui Liu, Yao Li, Jiliang Tang, and Ming Yan. A double residual compression algorithm for efficient distributed learning. In *International Conference on Artificial Intelligence and Statistics*, pages 133–143. PMLR, 2020

In print:

- [32] Yao Li, Xiaorui Liu, Jiliang Tang, Ming Yan, and Kun Yuan. Decentralized composite optimization with compression. *arXiv preprint arXiv:2108.04448*, 2021

1.5 Other Papers

In addition to the above papers, I completed the following three papers during my doctoral study. The first work also considers optimization with compression, which proposes an improved algorithmic framework with a compression scheme different from the one covered in Chapter 3 to achieve efficient communication for centralized learning. The second paper proposes LEAD to solve smooth problems with compressed communication, which is the special case of Prox-LEAD in Chapter 4. The theoretical result of LEAD in [33] is extended and improved by Prox-LEAD in [32]. The last one considers the equivalence of several existing primal-dual algorithms and derives the improved bound on stepsizes for some of them, which is beyond the topic I would like to discuss here. For the consistency and the conciseness of the whole dissertation, I omit these papers.

- [34] Hanlin Tang, Yao Li, Ji Liu, and Ming Yan. ErrorCompensatedX: Error compensation for variance reduced algorithms. *Advances in Neural Information Processing Systems*, 34:18102–18113, 2021

- [33] Xiaorui Liu, Yao Li, Rongrong Wang, Jiliang Tang, and Ming Yan. Linear convergent decentralized optimization with compression. In *International Conference on Learning Representations*, 2021

- [35] Yao Li and Ming Yan. On the improved conditions for some primal-dual algorithms. *arXiv preprint arXiv:2201.00139*, 2022

CHAPTER 2

EXTRA AND NIDS: DECENTRALIZED ALGORITHMS WITH IMPROVED CONDITIONS

2.1 Introduction

Early decentralized methods based on decentralized gradient descent [26, 36, 37, 27, 38] have sublinear convergence for strongly convex objective functions, because of the diminishing stepsize that is needed to obtain a consensual and optimal solution. This sublinear convergence rate is much slower than that for centralized ones. The first decentralized algorithm with linear convergence [39] is based on Alternate Direction Multiplier Method (ADMM) [40, 41]. Note that this type of algorithms has $O(1/k)$ rate for general convex functions [42, 43, 44]. After that, many linearly convergent algorithms are proposed. Some examples are EXTRA [1], NIDS [2], DIGing [45, 46], ESOM [47], gradient tracking methods [48, 49, 50, 46, 45, 51, 52], exact diffusion [53, 54], and dual optimal [55, 56]. There are also works on composite functions, where each private function is the sum of a smooth and a nonsmooth functions [57, 2, 58, 59]. Another topic of interest is decentralized optimization over directed and dynamic graphs [60, 61, 62, 63, 45, 64, 65]. Interested readers are referred to [66] and the references therein.

This chapter focuses on two linear convergent algorithms: EXTRA and NIDS, and provides better theoretical convergence results. EXact firsT-ordeR Algorithm (EXTRA) was proposed in [1], and its iteration is described in (2.2). For the general convex case, where each f_i is convex and L-smooth (i.e., has a L-Lipschitz continuous gradient), the convergence condition in [1] is $\alpha \in \left(0, \frac{1+\lambda_{\min}(\mathbf{W})}{L}\right)$. Therefore, there is an implicit condition for $\mathbf{W}$ that the smallest eigenvalue of $\mathbf{W}$ is larger than -1. Later the condition is relaxed to $\alpha \in \left(0, \frac{5+3\lambda_{\min}(\mathbf{W})}{4L}\right)$ in [3], and the corresponding requirement for $\mathbf{W}$ is that the smallest eigenvalue of $\mathbf{W}$ is larger than $-5/3$. In addition, this condition for the stepsize is shown to be optimal, i.e., EXTRA may diverge if the condition is not satisfied. Though we can always manipulate $\mathbf{W}$ to change the smallest eigenvalue, the convergence speed of EXTRA depends on the matrix $\mathbf{W}$. In the numerical experiment, we will see that it is beneficial to choose small eigenvalues for EXTRA in certain scenarios.

The linear convergence of EXTRA requires additional conditions on the functions. There are mainly three types of conditions used in the literature: the strong convexity of $\bar{f}$ (and some weaker variants) [1], the strong convexity of each f_i (and some weaker variants) [3], and the strong convexity of one function f_i [54]. Note that the condition on $\bar{f}$ is much weaker than the other two; there are cases where $\bar{f}$ is strongly convex but none of f_i's is. E.g., $f_i = \|e_i^T x\|_2^2$ for $p = n > 1$, where e_i is the vector whose ith component is 1 and all other components are 0. If $\bar{f}$ is (restricted) strongly convex with parameter $\mu_{\bar{f}}$, the linear convergence of EXTRA is shown when $\alpha \in \left(0, \frac{\mu_{\bar{f}}(1+\lambda_{\min}(\mathbf{W}))}{L^2}\right)$ in [1]. The upper bound for the stepsize is very conservative, and the better performance with a larger stepsize was shown numerically in [1] without proof. If each f_i is strongly convex with parameter μ, the linear convergence is shown when $\alpha \in \left(0, \frac{1+\lambda_{\min}(\mathbf{W})}{L+\mu}\right)$ and $\alpha \in \left(0, \frac{5+3\lambda_{\min}(\mathbf{W})}{4L}\right)$ in [58] and [3], respectively. One contribution of this chapter is to show the linear convergence of EXTRA under the (restricted) strong convexity of $\bar{f}$ and $\alpha \in \left(0, \frac{5+3\lambda_{\min}(\mathbf{W})}{4L}\right)$.

The algorithm NIDS (Network InDependent Stepsize) was proposed in [2]. Though there is a small difference from EXTRA, NIDS can choose a stepsize that does not depend on the mixing matrices. The linear convergence of NIDS in [2] requires $\mathbf{I} \succcurlyeq \mathbf{W} \succ -\mathbf{I}$ and strong convexity of $\mathbf{f}(\mathbf{x})$. In this chapter, we relax this condition for linear convergence to (restricted) strong convexity of $\bar{f}(x)$ and the relaxed mixing matrices with $\mathbf{I} \succcurlyeq \mathbf{W} \succ -(5/3)\mathbf{I}$.

2.2 Notation

We let

$$\mathbf{f}(\mathbf{x}) := \sum_{i=1}^{n} f_i(x_i), \tag{2.1}$$

where each $x_i \in \mathbb{R}^p$ is the local copy of the global variable x and the kth iterated point is x_i^k. Since agent i has its own estimate x_i of the global variable x, we put them together and define

$$\mathbf{x} = [x_1, x_2, \cdots, x_n]^\top \in \mathbb{R}^{n \times p}.$$

The gradient of $\mathbf{f}$ is defined as

$$\nabla \mathbf{f}(\mathbf{x}) = [\nabla f_1(x_1), \nabla f_2(x_2), \cdots, \nabla f_n(x_n)]^\top \in \mathbb{R}^{n \times p}.$$

We say that $\mathbf{x}$ is consensual if $x_1 = x_2 = \cdots = x_n$, i.e., $\mathbf{x} = \mathbf{1}x^\top$, where $x \in \mathbb{R}^{p \times 1}$ and $\mathbf{1} = [1, 1, \cdots, 1]^\top \in \mathbb{R}^{n \times 1}$.

In this chapter, we use $\|\cdot\|$ and $\langle \cdot, \cdot \rangle$ to denote the Frobenious norm and the corresponding inner product, respectively. For a given matrix $\mathbf{M} \in \mathbb{R}^{n \times p}$ and any positive (semi)definite matrix $\mathbf{H}$, which is denoted as $\mathbf{H} \succ 0$ ($\mathbf{H} \succcurlyeq 0$ for positive semidefinite), we define $\|\mathbf{M}\|_{\mathbf{H}} := \sqrt{\mathrm{tr}(\mathbf{M}^\top \mathbf{H} \mathbf{M})}$. The largest and the smallest eigenvalues of a matrix $\mathbf{A}$ are defined as $\lambda_{\max}(\mathbf{A})$ and $\lambda_{\min}(\mathbf{A})$. For a symmetric positive semidefinite matrix $\mathbf{A}$, we let $\lambda_{\min}^+(\mathbf{A})$ be the smallest nonzero eigenvalue. $\mathbf{A}^\dagger$ is the pseudo inverse of $\mathbf{A}$. For a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, we say a matrix $\mathbf{B} \in \mathbb{R}^{n \times p}$ is in $\mathbf{Ker}\{\mathbf{A}\}$ if $\mathbf{AB} = \mathbf{0}_{n \times p}$, and $\mathbf{B}$ is in $\mathbf{Range}\{\mathbf{A}\}$ if there exists $\mathbf{C} \in \mathbb{R}^{n \times p}$ such that $\mathbf{B} = \mathbf{AC}$. For simplicity, we may use $\mathbf{x}^+$ and $\mathbf{x}$ to replace $\mathbf{x}^{k+1}$ and $\mathbf{x}^k$, respectively, in the proofs.

2.3 Algorithms and Prerequisites

One iteration of EXTRA can be expressed as

$$\mathbf{x}^{k+2} = (\mathbf{I} + \mathbf{W})\mathbf{x}^{k+1} - \widetilde{\mathbf{W}}\mathbf{x}^k - \alpha[\nabla \mathbf{f}(\mathbf{x}^{k+1}) - \nabla \mathbf{f}(\mathbf{x}^k)]. \tag{2.2}$$

The stepsize $\alpha > 0$, and the symmetric matrices $\mathbf{W}$ and $\widetilde{\mathbf{W}}$ satisfy $\mathbf{I} + \mathbf{W} \succcurlyeq 2\widetilde{\mathbf{W}} \succcurlyeq 2\mathbf{W}$. The initial value $\mathbf{x}^0$ is chosen arbitrarily, and $\mathbf{x}^1 = \mathbf{W}\mathbf{x}^0 - \alpha \nabla \mathbf{f}(\mathbf{x}^0)$. In practice, we usually let $\widetilde{\mathbf{W}} = \frac{\mathbf{I}+\mathbf{W}}{2}$.

One iteration of NIDS is

$$\mathbf{x}^{k+2} = \tfrac{\mathbf{I}+\mathbf{W}}{2}\left[2\mathbf{x}^{k+1} - \mathbf{x}^k \ - \alpha(\nabla \mathbf{f}(\mathbf{x}^{k+1}) - \nabla \mathbf{f}(\mathbf{x}^k))\right], \tag{2.3}$$

where $\alpha > 0$ is the stepsize. The initial value $\mathbf{x}^0$ is chosen arbitrarily, and $\mathbf{x}^1 = \frac{\mathbf{I}+\mathbf{W}}{2}[\mathbf{x}^0 - \alpha\nabla\mathbf{f}(\mathbf{x}^0)]$.

If we choose $\widetilde{\mathbf{W}} = \frac{\mathbf{I}+\mathbf{W}}{2}$ in (2.2), the difference between EXTRA and NIDS in the above mathematical forms happens only in the communicated data, i.e.,whether we exchange the gradient information or not at each step. In practice, EXTRA can gain the advantage of time overlap by parallelizing communication and gradient evaluation, while NIDS evaluates the gradient and then communicates after the gradient is added. However, this small difference brings big changes in the convergence [2]. In order for both algorithms to converge, we have the following assumptions on $\mathbf{W}$ and $\widetilde{\mathbf{W}}$.

Assumption 2.3.1 (Mixing matrix)**.** *The connected network $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ consists of a set of nodes $\mathcal{V} = \{1, 2, \cdots, n\}$ and a set of undirected edges $\mathcal{E}$. An undirected edge $(i, j) \in \mathcal{E}$ means that there is a connection between agents i and j and both agents can exchange data. The mixing matrices $\mathbf{W} = [w_{ij}] \in \mathbb{R}^{n\times n}$ and $\widetilde{\mathbf{W}} = [\widetilde{w}_{ij}] \in \mathbb{R}^{n\times n}$ satisfy:*

1. *(Decentralized property): If $i \neq j$ and $(i, j) \notin \mathcal{E}$, then $w_{ij} = \widetilde{w}_{ij} = 0$.*

2. *(Symmetry): $\mathbf{W} = \mathbf{W}^\top$, $\widetilde{\mathbf{W}} = \widetilde{\mathbf{W}}^\top$.*

3. *(Null space property): $\mathbf{Null}\{\mathbf{W} - \widetilde{\mathbf{W}}\} = \mathbf{span}\{\mathbf{1}\} \subseteq \mathbf{Null}\{\mathbf{I} - \widetilde{\mathbf{W}}\}$.*

4. *(Spectral property): $\frac{\mathbf{I}+\mathbf{W}}{2} \succcurlyeq \widetilde{\mathbf{W}} \succ -\frac{1}{3}\mathbf{I}, \quad \widetilde{\mathbf{W}} \succcurlyeq \mathbf{W}$.*

Remark 2.3.1. *Parts 2-4 imply that the spectrum of $\mathbf{W}$ is enlarged to $(-\frac{5}{3}, 1]$, while the original assumption is $(-1, 1]$ for doubly stochastic matrices. Therefore, in our assumption, $\frac{\mathbf{I}+\mathbf{W}}{2}$ does not have to be positive definite. This assumption for $\mathbf{W}$ is strictly weaker than those in [1] and [2].*

Remark 2.3.2. *From [1, Proposition 2.2], $\mathbf{Null}\{\mathbf{I} - \mathbf{W}\} = \mathbf{span}\{\mathbf{1}\}$. It is a critical result for both algorithms.*

Before showing their theoretical results, we reformulate both algorithms.

Reformulation of EXTRA: We reformulate EXTRA by introducing a variable $\mathbf{y} \in \mathbb{R}^{n\times p}$ as

$$\mathbf{x}^{k+1} = \widetilde{\mathbf{W}}\mathbf{x}^k + \mathbf{y}^k - \alpha\nabla\mathbf{f}(\mathbf{x}^k), \tag{2.4a}$$

$$\mathbf{y}^{k+1} = \mathbf{y}^k - (\widetilde{\mathbf{W}} - \mathbf{W})\mathbf{x}^{k+1}, \tag{2.4b}$$

with $\mathbf{y}^0 = -(\widetilde{\mathbf{W}} - \mathbf{W})\mathbf{x}^0$. Then (2.4) is equivalent to EXTRA (2.2).

Proposition 2.3.1. *Let the* $\mathbf{x}$*-sequence generated by* (2.4) *with* $\mathbf{y}^0 = -(\widetilde{\mathbf{W}} - \mathbf{W})\mathbf{x}^0$ *be* $\{\mathbf{x}^k\}_{k=1}^{\infty}$*, then it's identical to the sequence generated by EXTRA* (2.2) *with the same initial point* $\mathbf{x}^0$*.*

Proof. From (2.4a), we have

$$\begin{aligned}\mathbf{x}^1 =&\widetilde{\mathbf{W}}\mathbf{x}^0 + \mathbf{y}^0 - \alpha\nabla\mathbf{f}(\mathbf{x}^0) = \widetilde{\mathbf{W}}\mathbf{x}^0 - (\widetilde{\mathbf{W}} - \mathbf{W})\mathbf{x}^0 - \alpha\nabla\mathbf{f}(\mathbf{x}^0)\\ =&\mathbf{W}\mathbf{x}^0 - \alpha\nabla\mathbf{f}(\mathbf{x}^0).\end{aligned}$$

For $k \geq 0$, we have

$$\begin{aligned}\mathbf{x}^{k+2} =&\widetilde{\mathbf{W}}\mathbf{x}^{k+1} + \mathbf{y}^{k+1} - \alpha\nabla\mathbf{f}(\mathbf{x}^{k+1}) = \mathbf{W}\mathbf{x}^{k+1} + \mathbf{y}^k - \alpha\nabla\mathbf{f}(\mathbf{x}^{k+1})\\ =&(\mathbf{I} + \mathbf{W})\mathbf{x}^{k+1} - \widetilde{\mathbf{W}}\mathbf{x}^k - \alpha[\mathbf{f}(\mathbf{x}^{k+1}) - \mathbf{f}(\mathbf{x}^k)],\end{aligned}$$

where the second and last equalities are from (2.4b) and (2.4a), respectively. □

Remark 2.3.3. *By* (2.4b) *and the assumption of* $\mathbf{y}^0$, $\mathbf{y}^k \in \mathbf{Range}\{\widetilde{\mathbf{W}} - \mathbf{W}\}$ *for all* k*. Also,* $\mathbf{x}^{k+1} = (\widetilde{\mathbf{W}} - \mathbf{W})^{\dagger}(\mathbf{y}^k - \mathbf{y}^{k+1}) + \mathbf{z}^{k+1}$ *for some* $\mathbf{z}^{k+1} \in \mathbf{Ker}\{\widetilde{\mathbf{W}} - \mathbf{W}\}$*.*

Reformulation of NIDS: We adopt the following reformulation from [2]:

$$\mathbf{d}^{k+1} = \mathbf{d}^k + \tfrac{\mathbf{I}-\mathbf{W}}{2\alpha}[\mathbf{x}^k - \alpha\nabla\mathbf{f}(\mathbf{x}^k) - \alpha\mathbf{d}^k], \tag{2.5a}$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha\nabla\mathbf{f}(\mathbf{x}^k) - \alpha\mathbf{d}^{k+1}, \tag{2.5b}$$

with $\mathbf{d}^0 = \mathbf{0}$. The equivalence is shown in [2].

To establish the linear convergence of EXTRA and NIDS, we need the following two assumptions.

Assumption 2.3.2 (Uniqueness). *There is a unique minimizer x^* for $\bar{f}(x)$.*

Assumption 2.3.3 (L-smoothness and restricted strong convexity). *Each function f_i is a proper, closed and convex function with a Lipschitz continuous gradient:*

$$\|\nabla f_i(x) - \nabla f_i(\widetilde{x})\| \leq L\|x - \widetilde{x}\|, \ \forall x, \ \widetilde{x} \in \mathbb{R}^p, \tag{2.6}$$

where $L > 0$ is the Lipschitz constant. Furthermore, $\bar{f}(x)$ is (restricted) strongly convex with respect to x^*:

$$\langle x - x^*, \nabla \bar{f}(x) - \nabla \bar{f}(x^*)\rangle \geq \mu_{\bar{f}}\|x - x^*\|^2, \ \forall x \in \mathbb{R}^p. \tag{2.7}$$

From [67, Theorem 2.1.5], the inequality (2.6) is equivalent to, for any $\mathbf{x}, \ \tilde{\mathbf{x}} \in \mathbb{R}^{n\times p}$,

$$\langle \mathbf{x} - \widetilde{\mathbf{x}}, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\widetilde{\mathbf{x}})\rangle \geq L^{-1}\|\nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\widetilde{\mathbf{x}})\|^2. \tag{2.8}$$

Proposition 2.3.2 ([1, Appendix A]). *The following two statements are equivalent:*

1. *$\bar{f}(x)$ is (restricted) strongly convex with respect to x^*;*
2. *For any $\eta > 0$, $\mathbf{g}(\mathbf{x}) := \mathbf{f}(\mathbf{x}) + \frac{\eta}{2}\|\mathbf{x}\|^2_{\mathbf{I}-\mathbf{W}}$ is $\mu_\mathbf{g}$ (restricted) strongly convex with respect to $\mathbf{x}^* = \mathbf{1}(x^*)^\top$. Specially, we can characterize*

$$\mu_\mathbf{g} = \min\left\{\frac{\mu_{\bar{f}}}{2}, \frac{\mu_{\bar{f}}^2\lambda^+_{\min}(\mathbf{I}-\mathbf{W})}{\mu_{\bar{f}}^2 + 16L^2}\eta\right\}.$$

This proposition shows

$$\langle \mathbf{x} - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle + \eta\|\mathbf{x} - \mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} \geq \mu_\mathbf{g}\|\mathbf{x} - \mathbf{x}^*\|^2 \tag{2.9}$$

for any $\mathbf{x} \in \mathbb{R}^{n\times p}$.

2.4 New Linear Convergence Results for EXTRA and NIDS

Throughout this section, we assume that Assumptions 2.3.1-2.3.3 hold. Two techniques are used to show the linear convergence: a) Proposition 2.3.2 serves as a bridge to connect $\bar{f}(x)$ and $\mathbf{f}(\mathbf{x})$. It is the key to the weaker assumption on objective functions. b) Both algorithms are equivalent to the extended Proximal Alternating Predictor-Corrector (PAPC) in [3], and this equivalence is the key to relaxing the conditions on the mixing matrices $\mathbf{W}$ and $\widetilde{\mathbf{W}}$.

2.4.1 Linear Convergence of EXTRA

When $\widetilde{\mathbf{W}} = \frac{\mathbf{I}+\mathbf{W}}{2}$, EXTRA is recovered by applying the extended PAPC in [3] to the following dual form of the decentralized consensus problem

$$\underset{\mathbf{y}}{\text{minimize}}\ \mathbf{f}^*(\sqrt{\mathbf{I}-\mathbf{W}}\mathbf{y}),$$

where $\mathbf{f}^*$ is the conjugate function of $\mathbf{f}$ and $\mathbf{y}$ is the dual variable. In this case, EXTRA has the optimal bound of the stepsize over the relaxed mixing matrix $\mathbf{W} \succ -(5/3)\mathbf{I}$. This fact enlightens us on the critical Lemma 2.4.3.

For simplicity, we introduce some notations. Because of part 4 of Assumption 2.3.1, given the mixing matrices $\mathbf{W}$ and $\widetilde{\mathbf{W}}$, there is a constant

$$\theta \in \Big(\frac{3}{4}, \min\big\{\frac{1}{1-\lambda_{\min}(\widetilde{\mathbf{W}})}, 1\big\}\Big]$$

such that

$$\overline{\mathbf{W}} := \theta\widetilde{\mathbf{W}} + (1-\theta)\mathbf{I} \succ \mathbf{0}, \tag{2.10}$$

$$\mathbf{H} := \overline{\mathbf{W}} + (\theta - \tfrac{1}{2})(\mathbf{I}-\widetilde{\mathbf{W}}) = \tfrac{\mathbf{I}+\widetilde{\mathbf{W}}}{2} \succ \mathbf{0}, \tag{2.11}$$

$$\mathbf{M} := (\widetilde{\mathbf{W}} - \mathbf{W})^{\dagger} \succcurlyeq \mathbf{0}, \tag{2.12}$$

$$\mathbf{G} := \mathbf{W} + \mathbf{I} - 2\widetilde{\mathbf{W}} \succcurlyeq \mathbf{0}. \tag{2.13}$$

Based on (2.10), we have

$$\widetilde{\mathbf{W}} = \overline{\mathbf{W}} - (1-\theta)(\mathbf{I}-\widetilde{\mathbf{W}}). \tag{2.14}$$

Let $(\mathbf{x}^*, \mathbf{y}^*)$ be a fixed point of (2.4), it is straightforward to show that

$$(\widetilde{\mathbf{W}} - \mathbf{W})\mathbf{x}^* = \mathbf{0}. \tag{2.15}$$

Part 3 of Assumption 2.3.1 shows that $\mathbf{x}^*$ is consensual, i.e., $\mathbf{x}^* = \mathbf{1}(x^*)^\top$ for certain $x^* \in \mathbb{R}^p$. The $\mathbf{y}$-iteration in (2.4b) and the initialization of $\mathbf{y}^0$ show $\mathbf{y}^k \in \mathbf{Range}\{\widetilde{\mathbf{W}} - \mathbf{W}\} =$

$\mathbf{Ker}\{\mathbf{1}^\top\}$. Then we have $\mathbf{1}^\top\mathbf{y}^* = \alpha\mathbf{1}^\top\nabla\mathbf{f}(\mathbf{x}^*) = 0$. Thus, x^* is the unique minimizer of $\bar{f}(x)$.

Lemma 2.4.1 (Norm over range space [2, Lemma 3])**.** *Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be symmetric positive (semi)definite with rank r ($r \leq n$) and $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r > 0$ be its r eigenvalues. Then $\mathbf{Range}\{\mathbf{A}\}$ is a rp-dimensional subspace in $\mathbb{R}^{n\times p}$ and has a norm defined by $\|\mathbf{x}\|^2_{\mathbf{A}^\dagger} := \langle \mathbf{x}, \mathbf{A}^\dagger\mathbf{x}\rangle$, where $\mathbf{A}^\dagger$ is the pseudo inverse of $\mathbf{A}$. In addition, $\lambda_1^{-1}\|\mathbf{x}\|^2 \leq \|\mathbf{x}\|^2_{\mathbf{A}^\dagger} \leq \lambda_r^{-1}\|\mathbf{x}\|^2$ for all $\mathbf{x} \in \mathbf{Range}\{\mathbf{A}\}$.*

For simplicity, we let $\mathbf{x}^+$ and $\mathbf{x}$ stand for $\mathbf{x}^{k+1}$ and $\mathbf{x}^k$, respectively, in the proofs. The same simplification applies to $\mathbf{y}^k$.

Lemma 2.4.2 (Norm equality)**.** *Let $\{(\mathbf{x}^k, \mathbf{y}^k)\}_{k=1}^\infty$ be the sequence generated by* (2.4)*, then it satisfies*

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2_{\widetilde{\mathbf{W}}-\mathbf{W}} = \|\mathbf{y}^k - \mathbf{y}^{k+1}\|^2_{\mathbf{M}}. \tag{2.16}$$

Proof. From Remark 2.3.3, we have

$$\mathbf{x}^+ = \mathbf{M}(\mathbf{y} - \mathbf{y}^+) + \mathbf{z}^+ \tag{2.17}$$

for $\mathbf{z}^+ \in \mathbf{Ker}\{\widetilde{\mathbf{W}} - \mathbf{W}\}$. This equality and (2.15) give

$$\begin{aligned}\|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\widetilde{\mathbf{W}}-\mathbf{W}} =&\langle \mathbf{x}^+ - \mathbf{x}^*, (\widetilde{\mathbf{W}} - \mathbf{W})(\mathbf{x}^+ - \mathbf{x}^*)\rangle = \langle \mathbf{x}^+, (\widetilde{\mathbf{W}} - \mathbf{W})\mathbf{x}^+\rangle \\ =&\langle \mathbf{M}(\mathbf{y} - \mathbf{y}^+), \mathbf{y} - \mathbf{y}^+\rangle = \|\mathbf{y} - \mathbf{y}^+\|^2_{\mathbf{M}},\end{aligned}$$

where the third equality holds because of (2.12), (2.17), and $\mathbf{y} - \mathbf{y}^+ \in \mathbf{Range}(\widetilde{\mathbf{W}} - \mathbf{W})$. □

Lemma 2.4.3 (A key inequality for EXTRA)**.** *Let $\{(\mathbf{x}^k, \mathbf{y}^k)\}_{k=1}^\infty$ be generated by* (2.4)*, then we have*

$$\begin{aligned}&\|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2_{\mathbf{H}} + \|\mathbf{y}^{k+1} - \mathbf{y}^*\|^2_{\mathbf{M}} \\ \leq&\|\mathbf{x}^k - \mathbf{x}^*\|^2_{\mathbf{H}} + \|\mathbf{y}^k - \mathbf{y}^*\|^2_{\mathbf{M}} - \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2_{(\theta-\frac{3}{4})(\mathbf{I}-\widetilde{\mathbf{W}})} - \|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2_{\mathbf{G}} \\ &- \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2_{\widetilde{\mathbf{W}}} - 2\alpha\langle \mathbf{x}^{k+1} - \mathbf{x}^*, \nabla\mathbf{f}(\mathbf{x}^k) - \nabla\mathbf{f}(\mathbf{x}^*)\rangle.\end{aligned} \tag{2.18}$$

Proof. The iteration (2.4) and equation (2.14) show

$$\begin{aligned}
&2\alpha\langle \mathbf{x}^+ - \mathbf{x}^*, \nabla\mathbf{f}(\mathbf{x}) - \nabla\mathbf{f}(\mathbf{x}^*)\rangle \\
=&2\langle \mathbf{x}^+ - \mathbf{x}^*, \widetilde{\mathbf{W}}(\mathbf{x}-\mathbf{x}^+) + \widetilde{\mathbf{W}}(\mathbf{x}^+ - \mathbf{x}^*) - (\mathbf{x}^+ - \mathbf{x}^*) + (\mathbf{y}-\mathbf{y}^*)\rangle \\
=&2\langle \mathbf{x}^+ - \mathbf{x}^*, \widetilde{\mathbf{W}}(\mathbf{x}-\mathbf{x}^+) + (\widetilde{\mathbf{W}} - \mathbf{I})(\mathbf{x}^+ - \mathbf{x}^*) \\
&+ (\widetilde{\mathbf{W}} - \mathbf{W})(\mathbf{x}^+ - \mathbf{x}^*) + \mathbf{y}^+ - \mathbf{y} + \mathbf{y} - \mathbf{y}^*\rangle \\
=&2\langle \mathbf{x}^+ - \mathbf{x}^*, \widetilde{\mathbf{W}}(\mathbf{x}-\mathbf{x}^+)\rangle + 2\langle \mathbf{x}^+ - \mathbf{x}^*, \mathbf{y}^+ - \mathbf{y}^*\rangle - 2\|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\mathbf{G}} \\
=&2\langle \mathbf{x}^+ - \mathbf{x}^*, \overline{\mathbf{W}}(\mathbf{x}-\mathbf{x}^+)\rangle - 2\langle \mathbf{x}^+ - \mathbf{x}^*, (1-\theta)(\mathbf{I}-\widetilde{\mathbf{W}})(\mathbf{x}-\mathbf{x}^+)\rangle \\
&+ 2\langle \mathbf{x}^+ - \mathbf{x}^*, \mathbf{y}^+ - \mathbf{y}^*\rangle - 2\|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\mathbf{G}}, \qquad (2.19)
\end{aligned}$$

where the first equality comes from (2.4a), the second one follows (2.4b), and the last one is from (2.14). From Remark 2.3.3, $\mathbf{x}^+ - \mathbf{x}^* = \mathbf{M}(\mathbf{y}-\mathbf{y}^+) + \mathbf{z}^+ - \mathbf{x}^*$ for some $\mathbf{z}^+ \in \mathbf{Ker}\{\widetilde{\mathbf{W}} - \mathbf{W}\}$. Thus $\langle \mathbf{z}^+ - \mathbf{x}^*, \mathbf{y}^+ - \mathbf{y}^*\rangle = 0$, and the equality (2.19) can be rewritten as

$$\begin{aligned}
&2\alpha\langle \mathbf{x}^+ - \mathbf{x}^*, \nabla\mathbf{f}(\mathbf{x}) - \nabla\mathbf{f}(\mathbf{x}^*)\rangle \\
=&2\langle \mathbf{x}^+ - \mathbf{x}^*, \overline{\mathbf{W}}(\mathbf{x}-\mathbf{x}^+)\rangle - 2\langle \mathbf{x}^+ - \mathbf{x}^*, (1-\theta)(\mathbf{I}-\widetilde{\mathbf{W}})(\mathbf{x}-\mathbf{x}^+)\rangle \\
&+ 2\langle \mathbf{M}(\mathbf{y}-\mathbf{y}^+), \mathbf{y}^+ - \mathbf{y}^*\rangle - 2\|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\mathbf{G}}.
\end{aligned}$$

Using the basic equality $2\langle a-b, b-c\rangle = \|a-c\|^2 - \|a-b\|^2 - \|b-c\|^2$ and Lemma 2.4.2, we have

$$\begin{aligned}
&\|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\overline{\mathbf{W}}} - \|\mathbf{x}^+ - \mathbf{x}^*\|^2_{(1-\theta)(\mathbf{I}-\widetilde{\mathbf{W}})} + \|\mathbf{y}^+ - \mathbf{y}^*\|^2_{\mathbf{M}} \\
=&\|\mathbf{x} - \mathbf{x}^*\|^2_{\overline{\mathbf{W}}} - \|\mathbf{x} - \mathbf{x}^*\|^2_{(1-\theta)(\mathbf{I}-\widetilde{\mathbf{W}})} + \|\mathbf{y} - \mathbf{y}^*\|^2_{\mathbf{M}} \\
&- \|\mathbf{x} - \mathbf{x}^+\|^2_{\overline{\mathbf{W}}} + \|\mathbf{x} - \mathbf{x}^+\|^2_{(1-\theta)(\mathbf{I}-\widetilde{\mathbf{W}})} - \|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\widetilde{\mathbf{W}}-\mathbf{W}} \\
&- 2\|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\mathbf{G}} - 2\alpha\langle \mathbf{x}^+ - \mathbf{x}^*, \nabla\mathbf{f}(\mathbf{x}) - \nabla\mathbf{f}(\mathbf{x}^*)\rangle. \qquad (2.20)
\end{aligned}$$

Note that the following inequality holds,

$$\tfrac{1}{2}\|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\widetilde{\mathbf{W}}-\mathbf{W}} \le \|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\widetilde{\mathbf{W}}-\mathbf{W}} + \tfrac{1}{2}\|\mathbf{x} - \mathbf{x}^*\|^2_{\widetilde{\mathbf{W}}-\mathbf{W}} - \tfrac{1}{4}\|\mathbf{x} - \mathbf{x}^+\|^2_{\widetilde{\mathbf{W}}-\mathbf{W}}.$$

Adding it onto both sides of (2.20), we have

$$\begin{aligned}
&\|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{H}}^2 - \tfrac{1}{2}\|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{G}}^2 + \|\mathbf{y}^+ - \mathbf{y}^*\|_{\mathbf{M}}^2 \\
\leq &\|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{H}}^2 - \tfrac{1}{2}\|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{G}}^2 + \|\mathbf{y} - \mathbf{y}^*\|_{\mathbf{M}}^2 \\
&- \|\mathbf{x} - \mathbf{x}^+\|_{\overline{\mathbf{W}}}^2 - \|\mathbf{x} - \mathbf{x}^+\|_{(\theta - \frac{3}{4})(\mathbf{I} - \widetilde{\mathbf{W}})}^2 + \frac{1}{4}\|\mathbf{x} - \mathbf{x}^+\|_{\mathbf{G}}^2 \\
&- 2\|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{G}}^2 - 2\alpha\langle \mathbf{x}^+ - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle. \qquad (2.21)
\end{aligned}$$

Apply the inequality $\frac{1}{4}\|\mathbf{x}-\mathbf{x}^+\|_{\mathbf{G}}^2 \leq \frac{1}{2}\|\mathbf{x}-\mathbf{x}^*\|_{\mathbf{G}}^2+\frac{1}{2}\|\mathbf{x}^+-\mathbf{x}^*\|_{\mathbf{G}}^2$, then the key inequality (2.18) is obtained. □

In the following theorem, we assume $\mathbf{G} \neq \mathbf{0}$ (i.e., $\widetilde{\mathbf{W}} \neq (\mathbf{I} + \mathbf{W})/2$). It is easy to amend the proof to show the result for this special case.

Theorem 2.4.1 (Q-linear convergence of EXTRA)**.** *Under Assumptions 2.3.1-2.3.3, we define*

$$r_1 = \tfrac{4\theta - 3}{4(1-\theta)^2 \lambda_{\max}(\overline{\mathbf{W}}^{-1}(\mathbf{I} - \widetilde{\mathbf{W}}))} > 0, \qquad (2.22)$$

$$r_2 = \tfrac{1}{2\lambda_{\max}(\mathbf{G}\overline{\mathbf{W}}^{-1})} > 0, \qquad (2.23)$$

$$r_3 = \tfrac{r_1 r_2}{r_1 + r_2 + r_1 r_2} \in (0, 1), \qquad (2.24)$$

and choose two small parameters ξ and η such that

$$\xi \in \Big(0, \min\Big\{\tfrac{r_3}{4\lambda_{\max}(\overline{\mathbf{W}}\mathbf{M})}, 1\Big\}\Big), \qquad (2.25)$$

$$\eta \in \Big(0, \tfrac{\lambda_{\min}(\overline{\mathbf{W}})\xi}{4\alpha\lambda_{\min}(\overline{\mathbf{W}}) - 2\alpha^2 L}\Big). \qquad (2.26)$$

In addition, we define

$$\begin{aligned}
\mathbf{P} :=& \mathbf{H} + \tfrac{\xi}{2}(\mathbf{I} - \mathbf{W}) \succ \mathbf{0}, \\
\mathbf{Q} :=& \mathbf{M} + (r_3 - 2\xi\lambda_{\max}(\overline{\mathbf{W}}\mathbf{M}))\overline{\mathbf{W}}^{-1} \succ \mathbf{0}.
\end{aligned}$$

Then for any stepsize $\alpha \in (0, \frac{2\lambda_{\min}(\overline{\mathbf{W}})}{L})$, we have

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_{\mathbf{P}}^2 + \|\mathbf{y}^{k+1} - \mathbf{y}^*\|_{\mathbf{Q}}^2 \leq \rho(\|\mathbf{x}^k - \mathbf{x}^*\|_{\mathbf{P}}^2 + \|\mathbf{y}^k - \mathbf{y}^*\|_{\mathbf{Q}}^2), \qquad (2.27)$$

where

$$\begin{aligned}\rho := \max\Big\{ & 1 - \Big(2\alpha - \tfrac{\alpha^2 L}{\lambda_{\min}(\overline{\mathbf{W}})}\Big)\mu_{\mathbf{g}}, \Big(4\alpha - \tfrac{2\alpha^2 L}{\lambda_{\min}(\overline{\mathbf{W}})}\Big)\tfrac{\eta}{\xi}, \\ & 1 - \tfrac{r_3 - 4\xi\lambda_{\max}(\overline{\mathbf{W}}\mathbf{M})}{r_3 + (1-2\xi)\lambda_{\max}(\overline{\mathbf{W}}\mathbf{M})}\Big\} < 1.\end{aligned} \tag{2.28}$$

Proof. From (2.18) in Lemma 2.4.3, we have

$$\begin{aligned}&\|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{H}}^2 + \|\mathbf{y}^+ - \mathbf{y}^*\|_{\mathbf{M}}^2 \\ \leq & \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{H}}^2 + \|\mathbf{y} - \mathbf{y}^*\|_{\mathbf{M}}^2 - \|\mathbf{x} - \mathbf{x}^+\|_{(\theta - \frac{3}{4})(\mathbf{I} - \widetilde{\mathbf{W}})}^2 - \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{G}}^2 \\ & - \|\mathbf{x} - \mathbf{x}^+\|_{\overline{\mathbf{W}}}^2 - 2\alpha\langle \mathbf{x}^+ - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle.\end{aligned} \tag{2.29}$$

Then we find an upper bound of $-\|\mathbf{x} - \mathbf{x}^+\|_{\overline{\mathbf{W}}}^2 - 2\alpha\langle \mathbf{x}^+ - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle$.

$$\begin{aligned}& - \|\mathbf{x} - \mathbf{x}^+\|_{\overline{\mathbf{W}}}^2 - 2\alpha\langle \mathbf{x}^+ - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle \\ = & \alpha^2 \|\nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\|_{\overline{\mathbf{W}}^{-1}}^2 - 2\alpha\langle \mathbf{x} - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle \\ & - \|\overline{\mathbf{W}}(\mathbf{x} - \mathbf{x}^+) - \alpha(\nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*))\|_{\overline{\mathbf{W}}^{-1}}^2 \\ \leq & - \big(2\alpha - \tfrac{\alpha^2 L}{\lambda_{\min}(\overline{\mathbf{W}})}\big)\langle \mathbf{x} - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle \\ & - \|\overline{\mathbf{W}}(\mathbf{x} - \mathbf{x}^+) - \alpha(\nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*))\|_{\overline{\mathbf{W}}^{-1}}^2,\end{aligned}$$

where, the inequality comes from (2.8). Combining it with (2.29), we have

$$\begin{aligned}& \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{H}}^2 + \|\mathbf{y}^+ - \mathbf{y}^*\|_{\mathbf{M}}^2 - \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{H}}^2 - \|\mathbf{y} - \mathbf{y}^*\|_{\mathbf{M}}^2 \\ \leq & - \big(2\alpha - \tfrac{\alpha^2 L}{\lambda_{\min}(\overline{\mathbf{W}})}\big)\langle \mathbf{x} - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle \\ & - \|\overline{\mathbf{W}}(\mathbf{x} - \mathbf{x}^+) - \alpha(\nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*))\|_{\overline{\mathbf{W}}^{-1}}^2 \\ & - \|\mathbf{x} - \mathbf{x}^+\|_{(\theta - \frac{3}{4})(\mathbf{I} - \widetilde{\mathbf{W}})}^2 - \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{G}}^2.\end{aligned} \tag{2.30}$$

The inequality (2.30) shows that $\{(\mathbf{x}^k, \mathbf{y}^k)\}_{k=1}^{\infty}$ is a Cauchy sequence converging to the fixed point $(\mathbf{x}^*, \mathbf{y}^*)$ of (2.4). From (2.9), we can bound the first term on the right hand side of (2.30) as

$$\begin{aligned}& - \big(2\alpha - \tfrac{\alpha^2 L}{\lambda_{\min}(\overline{\mathbf{W}})}\big)\langle \mathbf{x} - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle \\ \leq & \big(2\alpha - \tfrac{\alpha^2 L}{\lambda_{\min}(\overline{\mathbf{W}})}\big)\eta \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{I} - \mathbf{W}}^2 - \big(2\alpha - \tfrac{\alpha^2 L}{\lambda_{\min}(\overline{\mathbf{W}})}\big)\mu_g \|\mathbf{x} - \mathbf{x}^*\|^2.\end{aligned} \tag{2.31}$$

Next, we bound the two terms involving successive iterated points, i.e., $-\|\overline{\mathbf{W}}(\mathbf{x}-\mathbf{x}^+)-\alpha(\nabla\mathbf{f}(\mathbf{x})-\nabla\mathbf{f}(\mathbf{x}^*))\|^2_{\overline{\mathbf{W}}^{-1}}-\|\mathbf{x}-\mathbf{x}^+\|^2_{(\theta-\frac{3}{4})(\mathbf{I}-\widetilde{\mathbf{W}})}$. Note that

$$\begin{aligned}&\overline{\mathbf{W}}(\mathbf{x}-\mathbf{x}^+)-\alpha(\nabla\mathbf{f}(\mathbf{x})-\nabla\mathbf{f}(\mathbf{x}^*))\\ =&\mathbf{G}(\mathbf{x}^+-\mathbf{x}^*)-(\mathbf{y}^+-\mathbf{y}^*)+(1-\theta)(\mathbf{I}-\widetilde{\mathbf{W}})(\mathbf{x}-\mathbf{x}^+).\end{aligned}\tag{2.32}$$

We use $T_1,\ T_2$, and T_3 to denote the three terms on the right hand side of (2.32), respectively. Using the definition of r_1 in (2.22), we have

$$\begin{aligned}&-\|T_1+T_2+T_3\|^2_{\overline{\mathbf{W}}^{-1}}-\|\mathbf{x}-\mathbf{x}^+\|^2_{(\theta-\frac{3}{4})(\mathbf{I}-\widetilde{\mathbf{W}})}\\ =&-\|T_1+T_2\|^2_{\overline{\mathbf{W}}^{-1}}-2\langle\overline{\mathbf{W}}^{-\frac{1}{2}}(T_1+T_2),\overline{\mathbf{W}}^{-\frac{1}{2}}T_3\rangle-\|T_3\|^2_{\overline{\mathbf{W}}^{-1}}\\ &-\tfrac{4\theta-3}{4(1-\theta)}\|\mathbf{x}-\mathbf{x}^+\|^2_{(1-\theta)(\mathbf{I}-\widetilde{\mathbf{W}})}\\ \le&-\|T_1+T_2\|^2_{\overline{\mathbf{W}}^{-1}}-2\langle\overline{\mathbf{W}}^{-\frac{1}{2}}(T_1+T_2),\overline{\mathbf{W}}^{-\frac{1}{2}}T_3\rangle-(1+r_1)\|T_3\|^2_{\overline{\mathbf{W}}^{-1}}\\ \le&-\tfrac{r_1}{1+r_1}\|T_1+T_2\|^2_{\overline{\mathbf{W}}^{-1}},\end{aligned}$$

where the last inequality comes from the Cauchy inequality

$$-2\langle a,b\rangle\le\tfrac{1}{1+r_1}\|a\|^2+(1+r_1)\|b\|^2.$$

Combining it with the last term $-\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{G}}$ on the right hand side of (2.30), we have

$$\begin{aligned}&-\tfrac{r_1}{1+r_1}\|T_1+T_2\|^2_{\overline{\mathbf{W}}^{-1}}-\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{G}}\\ \le&-\tfrac{r_1}{1+r_1}\|T_2\|^2_{\overline{\mathbf{W}}^{-1}}-\tfrac{2r_1}{1+r_1}\langle\overline{\mathbf{W}}^{-\frac{1}{2}}T_1,\overline{\mathbf{W}}^{-\frac{1}{2}}T_2\rangle-\tfrac{r_1}{1+r_1}\|T_1\|^2_{\overline{\mathbf{W}}^{-1}}\\ &-r_2\|T_1\|^2_{\overline{\mathbf{W}}^{-1}}-\tfrac{1}{2}\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{G}}\\ \le&-r_3\|\mathbf{y}^+-\mathbf{y}^*\|^2_{\overline{\mathbf{W}}^{-1}}-\tfrac{\xi}{2}\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{G}},\end{aligned}\tag{2.33}$$

where $\xi<1$ is a small positive parameter, and r_2 and r_3 are defined as (2.23) and (2.24), respectively. Since $\mathbf{G}=(\mathbf{I}-\mathbf{W})-2(\widetilde{\mathbf{W}}-\mathbf{W})$, we have

$$\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{G}}=\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}}-2\|\mathbf{y}-\mathbf{y}^+\|^2_{\mathbf{M}}.\tag{2.34}$$

Therefore

$$\begin{aligned}
&-\tfrac{r_1}{1+r_1}\|T_1+T_2\|^2_{\overline{\mathbf{W}}^{-1}} - \|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{G}}\\
\leq&-r_3\|\mathbf{y}^+-\mathbf{y}^*\|^2_{\overline{\mathbf{W}}^{-1}} - \tfrac{\xi}{2}\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} - \xi\|\mathbf{y}-\mathbf{y}^+\|^2_{\mathbf{M}}\\
\leq&-r_3\|\mathbf{y}^+-\mathbf{y}^*\|^2_{\overline{\mathbf{W}}^{-1}} - \tfrac{\xi}{2}\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} + 2\xi\|\mathbf{y}^+-\mathbf{y}^*\|^2_{\mathbf{M}} + 2\xi\|\mathbf{y}-\mathbf{y}^*\|^2_{\mathbf{M}}\\
\leq&-(r_3/\lambda_{\max}(\overline{\mathbf{W}}\mathbf{M})-2\xi)\|\mathbf{y}^+-\mathbf{y}^*\|^2_{\mathbf{M}} - \tfrac{\xi}{2}\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} + 2\xi\|\mathbf{y}-\mathbf{y}^*\|^2_{\mathbf{M}}.
\end{aligned} \tag{2.35}$$

Let $\xi < r_3/(4\lambda_{\max}(\overline{\mathbf{W}}\mathbf{M}))$, then we have $r_3/\lambda_{\max}(\overline{\mathbf{W}}\mathbf{M})-2\xi > 2\xi$. Putting (2.31) and (2.35) together onto (2.30), we have

$$\begin{aligned}
&\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{H}} + \tfrac{\xi}{2}\|\mathbf{x}^+-\mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} + (1+(r_3/\lambda_{\max}(\overline{\mathbf{W}}\mathbf{M})-2\xi))\|\mathbf{y}^+-\mathbf{y}^*\|^2_{\mathbf{M}}\\
\leq&\big(1-\big(2\alpha-\tfrac{\alpha^2L}{\lambda_{\min}(\overline{\mathbf{W}})}\big)\mu_g\big)\|\mathbf{x}-\mathbf{x}^*\|^2_{\mathbf{H}} + \big(2\alpha-\tfrac{\alpha^2L}{\lambda_{\min}(\overline{\mathbf{W}})}\big)\eta\|\mathbf{x}-\mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}}\\
&+(1+2\xi)\|\mathbf{y}-\mathbf{y}^*\|^2_{\mathbf{M}}.
\end{aligned}$$

Let ρ be defined as (2.28), we get (2.27). Note that the choice of ξ and η affects the definition of $\mathbf{P}$ and $\mathbf{Q}$, but not the algorithm. Hence for any $\alpha \in (0, \frac{2\lambda_{\min}(\overline{\mathbf{W}})}{L})$, Q-linear convergence is guaranteed for $(\mathbf{x}^k-\mathbf{x}^*, \mathbf{y}^k-\mathbf{y}^*)$.

Because $\|\mathbf{x}^k-\mathbf{x}^*\|^2_{\mathbf{P}} \leq \|\mathbf{x}^k-\mathbf{x}^*\|^2_{\mathbf{P}}+\|\mathbf{y}^k-\mathbf{y}^*\|^2_{\mathbf{Q}}$, the sequence $\{\|\mathbf{x}^k-\mathbf{x}^*\|^2_{\mathbf{P}}\}_{k=1}^{\infty}$ converges R-linearly to 0 at the rate of ρ. □

Two special cases are not covered by the theorem: $\theta=1$ and $\widetilde{\mathbf{W}}=\frac{\mathbf{I}+\mathbf{W}}{2}$. When $\theta=1$, we have $r_1=\infty$ and $r_3=\frac{r_2}{1+r_2}$. When $\widetilde{\mathbf{W}}=\frac{\mathbf{I}+\mathbf{W}}{2}$, i.e., $\mathbf{G}=\mathbf{0}$, we have $r_2=\infty$ and $r_3=\frac{r_1}{1+r_1}$. In both cases, the linear convergence rate is

$$\begin{aligned}
\rho=\max\Big\{&\ 1-\Big(2\alpha-\tfrac{2\alpha^2L}{2-\theta+\theta\lambda_{\min}(\mathbf{W})}\Big)\mu_{\mathbf{g}}, \Big(4\alpha-\tfrac{4\alpha^2L}{2-\theta+\theta\lambda_{\min}(\mathbf{W})}\Big)\tfrac{\eta}{\xi},\\
&1-\tfrac{\beta r_3-4\xi(2-\theta\beta)}{\beta r_3+(1-2\xi)(2-\theta\beta)}\Big\},
\end{aligned} \tag{2.36}$$

where $\beta=1-\lambda_2(\mathbf{W})$ is the spectral gap. It is exactly the limit of ρ in (2.28) with r_1 or r_2 approaching infinity.

Remark 2.4.1. *The upper bound for the stepsize α, $2(1-\theta+\theta\lambda_{\min}(\widetilde{\mathbf{W}}))/L$, is much larger than that in [1] for ensuring linear convergence, $2\mu_{\mathbf{g}}\lambda_{\min}(\widetilde{\mathbf{W}})/L^2$, when $\widetilde{\mathbf{W}}$ is positive definite. In the* *special case $\widetilde{\mathbf{W}} = (\mathbf{I}+\mathbf{W})/2$, we have $\alpha < (2-\theta+\theta\lambda_{\min}(\mathbf{W}))/L$. Since we can choose θ as close as possible to $3/4$, the upper bound of α attains $(3\lambda_{\min}(\mathbf{W})+5)/(4L)$, which coincides the optimal bound given in [3] for general convex functions. In [3], the linear convergence was shown under the strong convexity of all functions $\{f_i\}_{i=1}^n$.*

2.4.2 NIDS without Nonsmooth Term

We consider NIDS next. In the smooth case, NIDS can be recovered by PAPC applied to the primal form of the decentralized consensus problem

$$\operatorname*{minimize}_{\mathbf{x}} \ \mathbf{f}(\mathbf{x}), \quad \text{s.t.}\sqrt{\mathbf{I}-\mathbf{W}}\mathbf{x} = \mathbf{0}.$$

It motivates us to show the inequality in Lemma 2.4.5.

[2, Lemma 1] shows that, with the initialization $(\mathbf{d}^0 = \mathbf{0},\ \mathbf{x}^0)$, the fixed point $(\mathbf{d}^* \in \mathbf{Range}(\mathbf{I}-\mathbf{W}), \mathbf{x}^*)$ of (2.5) satisfies

$$\mathbf{d}^* + \nabla\mathbf{f}(\mathbf{x}^*) = \mathbf{0}, \tag{2.37a}$$

$$(\mathbf{I}-\mathbf{W})\mathbf{x}^* = \mathbf{0}, \tag{2.37b}$$

and $\mathbf{x}^*$ is the consensual solution to the problem (2.1). We will use the following important equality, which can be derived from (2.5)

$$(\mathbf{I} - \tfrac{\mathbf{I}-\mathbf{W}}{2})(\mathbf{d}^{k+1} - \mathbf{d}^k) = \tfrac{\mathbf{I}-\mathbf{W}}{2\alpha}(\mathbf{x}^{k+1} - \mathbf{x}^*). \tag{2.38}$$

Motivated by the proof of EXTRA, we introduce another matrix to measure the distance to the fixed point. We still pick $\theta \in (\frac{3}{4}, 1]$ such that

$$\theta\Big(\tfrac{\mathbf{I}+\mathbf{W}}{2}\Big) + (1-\theta)\mathbf{I} = \mathbf{I} - \theta\Big(\tfrac{\mathbf{I}-\mathbf{W}}{2}\Big) \succ \mathbf{0}. \tag{2.39}$$

Define a new symmetric matrix

$$\widetilde{\mathbf{M}} = 2(\mathbf{I}-\mathbf{W})^\dagger - \theta\mathbf{I} = \Big(\tfrac{\mathbf{I}-\mathbf{W}}{2}\Big)^\dagger - \theta\mathbf{I}. \tag{2.40}$$

Then $\widetilde{\mathbf{M}}$ is a norm over $\mathbf{Range}(\mathbf{I}-\mathbf{W})$. Note that $\widetilde{\mathbf{M}}$ is invertible because $\widetilde{\mathbf{M}}\mathbf{1} = -\theta\mathbf{1}$. In the following proofs, we use the same simplification $\mathbf{x}$ and $\mathbf{x}^+$.

Lemma 2.4.4 (Equality)**.** *Let* $\{(\mathbf{d}^k, \mathbf{x}^k)\}_{k=1}^{\infty}$ *be the sequence generated by* (2.5)*, we have the following two equalities:*

$$\langle \mathbf{x}^{k+1} - \mathbf{x}^*, \mathbf{d}^{k+1} - \mathbf{d}^* \rangle = \alpha \langle \mathbf{d}^{k+1} - \mathbf{d}^k, \mathbf{d}^{k+1} - \mathbf{d}^* \rangle_{\widetilde{\mathbf{M}} - (1-\theta)\mathbf{I}} \tag{2.41a}$$

$$\langle \mathbf{x}^{k+1} - \mathbf{x}^*, \mathbf{d}^{k+1} - \mathbf{d}^k \rangle = \alpha \| \mathbf{d}^{k+1} - \mathbf{d}^k \|^2_{\widetilde{\mathbf{M}} - (1-\theta)\mathbf{I}}. \tag{2.41b}$$

Proof. Since $\mathbf{d}^+ - \mathbf{d}^* \in \mathbf{Range}(\mathbf{I} - \mathbf{W})$, we have

$$\begin{aligned}\langle \mathbf{x}^+ - \mathbf{x}^*, \mathbf{d}^+ - \mathbf{d}^* \rangle =& \langle (\mathbf{I} - \mathbf{W})(\mathbf{x}^+ - \mathbf{x}^*), (\mathbf{I} - \mathbf{W})^\dagger (\mathbf{d}^+ - \mathbf{d}^*) \rangle \\ =& \alpha \langle (2\mathbf{I} - (\mathbf{I} - \mathbf{W}))(\mathbf{d}^+ - \mathbf{d}), (\mathbf{I} - \mathbf{W})^\dagger (\mathbf{d}^+ - \mathbf{d}^*) \rangle \\ =& \alpha \langle (2(\mathbf{I} - \mathbf{W})^\dagger - \mathbf{I})(\mathbf{d}^+ - \mathbf{d}), \mathbf{d}^+ - \mathbf{d}^* \rangle, \end{aligned} \tag{2.42}$$

where the second equality follows (2.38). Replacing $\mathbf{d}^*$ with $\mathbf{d}$ in (2.42), we get (2.41b) in the same way. □

Lemma 2.4.5 (A key inequality for NIDS)**.** *Let* $\{(\mathbf{d}^k, \mathbf{x}^k)\}_{k=1}^{\infty}$ *be generated by* (2.5)*. We have, with any* $r_4 \in (0, \theta - \frac{3}{4})$*,*

$$\begin{aligned} & \|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 + \alpha^2 \|\mathbf{d}^{k+1} - \mathbf{d}^*\|^2_{\widetilde{\mathbf{M}} + (\theta - \frac{1}{2} + 2r_4)\mathbf{I}} \\ \leq & \|\mathbf{x}^k - \mathbf{x}^*\|^2 + \alpha^2 \|\mathbf{d}^k - \mathbf{d}^*\|^2_{\widetilde{\mathbf{M}} + (\theta - \frac{1}{2} - 2r_4)\mathbf{I}} - \alpha^2 \|\mathbf{d}^k - \mathbf{d}^{k+1}\|^2_{\widetilde{\mathbf{M}} + (\theta - \frac{3}{4} - r_4)\mathbf{I}} \\ & + \alpha^2 \|\nabla \mathbf{f}(\mathbf{x}^k) - \nabla \mathbf{f}(\mathbf{x}^*)\|^2 - 2\alpha \langle \mathbf{x}^k - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}^k) - \nabla \mathbf{f}(\mathbf{x}^*) \rangle. \end{aligned} \tag{2.43}$$

Proof. The iteration (2.5) and the definition of $\widetilde{\mathbf{M}}$ in (2.40) show

$$\begin{aligned} & 2\alpha \langle \mathbf{x} - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*) \rangle \\ =& 2\langle \mathbf{x} - \mathbf{x}^*, \mathbf{x} - \mathbf{x}^+ \rangle - 2\alpha \langle \mathbf{x} - \mathbf{x}^*, \mathbf{d}^+ - \mathbf{d}^* \rangle \\ =& 2\langle \mathbf{x} - \mathbf{x}^+, \mathbf{x} - \mathbf{x}^* \rangle - 2\alpha \langle \mathbf{x} - \mathbf{x}^+, \mathbf{d}^+ - \mathbf{d}^* \rangle - 2\alpha \langle \mathbf{x}^+ - \mathbf{x}^*, \mathbf{d}^+ - \mathbf{d}^* \rangle \\ =& 2\langle \mathbf{x} - \mathbf{x}^+, \mathbf{x} - \alpha \mathbf{d}^+ - \mathbf{x}^* + \alpha \mathbf{d}^* \rangle + 2\alpha^2 \langle \mathbf{d} - \mathbf{d}^+, \mathbf{d}^+ - \mathbf{d}^* \rangle_{\widetilde{\mathbf{M}} - (1-\theta)\mathbf{I}} \\ =& 2\langle \mathbf{x} - \mathbf{x}^+, \mathbf{x}^+ - \mathbf{x}^* + \alpha \nabla \mathbf{f}(\mathbf{x}) - \alpha \nabla \mathbf{f}(\mathbf{x}^*) \rangle + 2\alpha^2 \langle \mathbf{d} - \mathbf{d}^+, \mathbf{d}^+ - \mathbf{d}^* \rangle_{\widetilde{\mathbf{M}} - (1-\theta)\mathbf{I}}, \end{aligned}$$

where the first and the last equalities use (2.5b) and the third one follows (2.41a).

From (2.5b), we obtain

$$\begin{aligned}
&2\alpha\langle \mathbf{x}-\mathbf{x}^+, \nabla\mathbf{f}(\mathbf{x})-\nabla\mathbf{f}(\mathbf{x}^*)\rangle\\
=&\|\mathbf{x}-\mathbf{x}^+\|^2+\alpha^2\|\nabla\mathbf{f}(\mathbf{x})-\nabla\mathbf{f}(\mathbf{x}^*)\|^2-\|\mathbf{x}-\mathbf{x}^+-\alpha\nabla\mathbf{f}(\mathbf{x})+\alpha\nabla\mathbf{f}(\mathbf{x}^*)\|^2\\
=&\|\mathbf{x}-\mathbf{x}^+\|^2+\alpha^2\|\nabla\mathbf{f}(\mathbf{x})-\nabla\mathbf{f}(\mathbf{x}^*)\|^2-\alpha^2\|\mathbf{d}^+-\mathbf{d}^*\|^2. \qquad (2.44)
\end{aligned}$$

Together with the basic equality $2\langle a-b, b-c\rangle = \|a-c\|^2-\|b-c\|^2-\|a-b\|^2$, we get

$$\begin{aligned}
&\|\mathbf{x}^+-\mathbf{x}^*\|^2+\alpha^2\|\mathbf{d}^+-\mathbf{d}^*\|^2_{\widetilde{\mathbf{M}}-(1-\theta)\mathbf{I}}\\
=&\|\mathbf{x}-\mathbf{x}^*\|^2+\alpha^2\|\mathbf{d}-\mathbf{d}^*\|^2_{\widetilde{\mathbf{M}}-(1-\theta)\mathbf{I}}-\alpha^2\|\mathbf{d}-\mathbf{d}^+\|^2_{\widetilde{\mathbf{M}}-(1-\theta)\mathbf{I}}-\alpha^2\|\mathbf{d}^+-\mathbf{d}^*\|^2\\
&+\alpha^2\|\nabla\mathbf{f}(\mathbf{x})-\nabla\mathbf{f}(\mathbf{x}^*)\|^2-2\alpha\langle\mathbf{x}-\mathbf{x}^*, \nabla\mathbf{f}(\mathbf{x})-\nabla\mathbf{f}(\mathbf{x}^*)\rangle. \qquad (2.45)
\end{aligned}$$

Since $r_4 < \theta-\frac{3}{4}\le 1/4$, the following inequality holds,

$$-(\tfrac{1}{2}-2r_4)\|\mathbf{d}^+-\mathbf{d}^*\|^2 \le (\tfrac{1}{2}-2r_4)\|\mathbf{d}-\mathbf{d}^*\|^2-(\tfrac{1}{4}-r_4)\|\mathbf{d}-\mathbf{d}^+\|^2.$$

Adding it onto both sides of (2.45), we get (2.43). □

Theorem 2.4.2 (Q-linear convergence for NIDS). *Under Assumptions 2.3.1-2.3.3, we define*

$$r_5 = \max\Big(2, \frac{(\lambda_{\max}(\mathbf{I}-\mathbf{W})-2)^2}{2-(\frac{3}{4}+r_4)\lambda_{\max}(\mathbf{I}-\mathbf{W})}\Big). \qquad (2.46)$$

For any stepsize $\alpha\in(0,\frac{2}{L})$, we choose $\eta\in(0,\frac{1}{\alpha(2-\alpha L)r_5})$ and define

$$\rho_3 = \max\Big\{1-\alpha(2-\alpha L)\mu_{\mathbf{g}}, \alpha(2-\alpha L)\eta r_5, 1-\tfrac{4r_4}{2\lambda_{\max}((\mathbf{I}-\mathbf{W})^+)-\frac{1}{2}+2r_4}\Big\} < 1, \qquad (2.47)$$

Then we have

$$\begin{aligned}
&\|\mathbf{x}^{k+1}-\mathbf{x}^*\|^2_{\mathbf{I}+\frac{\mathbf{I}-\mathbf{W}}{r_5}}+\alpha^2\|\mathbf{d}^{k+1}-\mathbf{d}^*\|^2_{\mathbf{Q}}\\
\le&\rho(\|\mathbf{x}^k-\mathbf{x}^*\|^2_{\mathbf{I}+\frac{\mathbf{I}-\mathbf{W}}{r_5}}+\alpha^2\|\mathbf{d}^k-\mathbf{d}^*\|^2_{\mathbf{Q}}),
\end{aligned} \qquad (2.48)$$

where $\mathbf{Q} := \widetilde{\mathbf{M}}+(\theta-\frac{1}{2}+2r_4)\mathbf{I}\succ\mathbf{0}$.

Proof. Given any $\alpha \in (0, \frac{2}{L})$, we have

$$\begin{aligned}
&\alpha^2\|\nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\|^2 - 2\alpha\langle \mathbf{x} - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle \\
\leq& -\alpha(2-\alpha L)\langle \mathbf{x} - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle \\
=& -\alpha(2-\alpha L)\langle \mathbf{x} - \mathbf{x}^*, \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{x}^*)\rangle - \alpha(2-\alpha L)\eta\|\mathbf{x} - \mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} \\
&+ \alpha(2-\alpha L)\eta\|\mathbf{x} - \mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} \\
\leq& -\alpha(2-\alpha L)\mu_{\mathbf{g}}\|\mathbf{x} - \mathbf{x}^*\|^2 + \alpha(2-\alpha L)\eta\|\mathbf{x} - \mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}},
\end{aligned}$$

where the first inequality is from (2.8) and the second one uses (restricted) strong convexity (2.9). Together with (2.43), we have

$$\begin{aligned}
&\|\mathbf{x}^+ - \mathbf{x}^*\|^2 + \alpha^2\|\mathbf{d}^+ - \mathbf{d}^*\|^2_{\widetilde{\mathbf{M}}+(\theta-\frac{1}{2}+2r_4)\mathbf{I}} \\
\leq&\|\mathbf{x} - \mathbf{x}^*\|^2 + \alpha^2\|\mathbf{d} - \mathbf{d}^*\|^2_{\widetilde{\mathbf{M}}+(\theta-\frac{1}{2}-2r_4)\mathbf{I}} - \alpha^2\|\mathbf{d} - \mathbf{d}^+\|^2_{\widetilde{\mathbf{M}}+(\theta-\frac{3}{4}-r_4)\mathbf{I}} \\
&- \alpha(2-\alpha L)\mu_{\mathbf{g}}\|\mathbf{x} - \mathbf{x}^*\|^2 + \alpha(2-\alpha L)\eta\|\mathbf{x} - \mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}},
\end{aligned} \tag{2.49}$$

The equality (2.38) gives

$$\begin{aligned}
&\|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} \\
=&\|(\mathbf{I}-\mathbf{W})(\mathbf{x}^+ - \mathbf{x}^*)\|^2_{(\mathbf{I}-\mathbf{W})^\dagger} = \alpha^2\|(2\mathbf{I}-(\mathbf{I}-\mathbf{W}))(\mathbf{d}^+ - \mathbf{d})\|^2_{(\mathbf{I}-\mathbf{W})^\dagger} \\
=&\alpha^2\|\mathbf{d} - \mathbf{d}^+\|^2_{(2\mathbf{I}-(\mathbf{I}-\mathbf{W}))(\mathbf{I}-\mathbf{W})^\dagger(2\mathbf{I}-(\mathbf{I}-\mathbf{W}))} = \alpha^2\|\mathbf{d} - \mathbf{d}^+\|^2_{4(\mathbf{I}-\mathbf{W})^\dagger-4\mathbf{I}+(\mathbf{I}-\mathbf{W})} \\
\leq&\alpha^2 r_5\|\mathbf{d} - \mathbf{d}^+\|^2_{\widetilde{\mathbf{M}}+(\theta-\frac{3}{4}-r_4)\mathbf{I}},
\end{aligned} \tag{2.50}$$

where the second equality follows (2.38), the fourth one is from $\mathbf{d} - \mathbf{d}^+ \in \mathbf{Range}(\mathbf{I}-\mathbf{W})$, and the inequality holds with the definition of r_5 in (2.46). Combing (2.49) and (2.50), we derive

$$\begin{aligned}
&\|\mathbf{x}^+ - \mathbf{x}^*\|^2 + \tfrac{1}{r_5}\|\mathbf{x}^+ - \mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} + \alpha^2\|\mathbf{d}^+ - \mathbf{d}^*\|^2_{\widetilde{\mathbf{M}}+(\theta-\frac{1}{2}+2r_4)\mathbf{I}} \\
\leq&(1-\alpha(2-\alpha L)\mu_{\mathbf{g}})\|\mathbf{x} - \mathbf{x}^*\|^2 + \alpha(2-\alpha L)\eta\|\mathbf{x} - \mathbf{x}^*\|^2_{\mathbf{I}-\mathbf{W}} \\
&+ \alpha^2\|\mathbf{d} - \mathbf{d}^*\|^2_{\widetilde{\mathbf{M}}+(\theta-\frac{1}{2}-2r_4)\mathbf{I}}.
\end{aligned} \tag{2.51}$$

Let ρ_3 be defined as (2.47), and we show (2.48). Meanwhile, the Q-linear convergence of $(\mathbf{d}^k, \mathbf{x}^k)$ implies the R-linear convergence of $\mathbf{x}^k$. □

This theorem shows that NIDS is still linearly convergent over a relaxed $\mathbf{W}$ and keeps the network-independent stepsize, which attains $\frac{2}{L}$ practically.

2.5 Numerical Experiments

In this section, we compare the performance of EXTRA and NIDS over the relaxed mixing matrices in the following two scenarios:

- Comparison of Decentralized Gradient Descent (DGD), EXTRA, and NIDS with different stepsizes for a doubly stochastic matrix $\mathbf{W}$.
- Comparison of EXTRA and NIDS with different stepsizes for a relaxed matrix $\mathbf{W}$.

We consider the following decentralized sensing problem. Each agent $i \in \{1, \cdots, n\}$ has its own private measured data $M_i \in \mathbb{R}^{m_i \times p}$ and $y_i \in \mathbb{R}^{m_i}$ based on the unknown common variable $x \in \mathbb{R}^p$. Suppose that $y_i = M_i x + e_i$ with independently identically distributed random noise $e_i \in \mathbb{R}^{m_i}$. The goal is to estimate x cooperatively over the network, and the problem is

$$\underset{x}{\text{minimize}} \;\; \bar{f}(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2}\|M_i x - y_i\|_2^2.$$

The data $\{M_i\}_{i=1}^n$ and x are generated from Gaussian distribution. We normalize each M_i such that $\|M_i^\top M_i\| = 10$, i.e., $L = 10$. In both scenarios, we set $n = 10$, $p = 5$, $\mathbf{x}^0 = \mathbf{0}$, and $\widetilde{\mathbf{W}} = \frac{\mathbf{I}+\mathbf{W}}{2}$ for EXTRA.

For the first scenario, we construct the matrix $\mathbf{W}$ based on the Metropolis constant edge weight matrix in [1, §2.4]. In this case $\widetilde{\mathbf{W}}$ is positive definite, and we can set $\theta = \frac{3}{4}$. Then $\overline{\mathbf{W}} = \frac{5\mathbf{I}+3\mathbf{W}}{8}$. We implement EXTRA with three different stepsizes: $\alpha_1 = \frac{(1+\lambda_{\min}(\mathbf{W}))\mu_{\bar{f}}}{100}$ (the stepsize for linear convergence in [1]), $\alpha_2 = \frac{1+\lambda_{\min}(\mathbf{W})}{10}$ (the stepsize for convergence only in [1]), and $\alpha_3 = \frac{5+3\lambda_{\min}(\mathbf{W})}{40}$ (our largest stepsize). For NIDS, the stepsize is set to $\alpha_4 = \frac{1}{5}$ although it is the upper bound of the stepsize which is not attainable in our proof theoretically.

The result with $m_i = 1$ is illustrated in Fig. 2.1. Because we have $n > p$, the function $\bar{f}(x)$ is strongly convex with probability one. NIDS requires the least number of iteration to attain the expected tolerance. Meanwhile, EXTRA with our proposed stepsize has better performance than that given in [1].

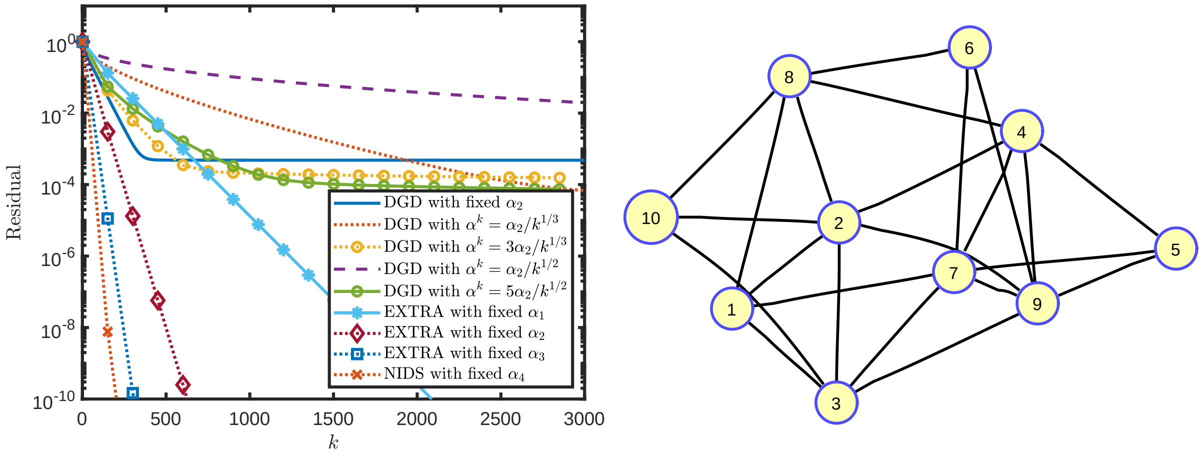


Figure 2.1: LEFT: the error $\frac{\|\mathbf{x}^k - \mathbf{x}^*\|_F}{\|\mathbf{x}^0 - \mathbf{x}^*\|_F}$ vs iterations for DGD with different stepsizes, EXTRA with three stepsizes, and NIDS. RIGHT: The random network with 10 nodes.

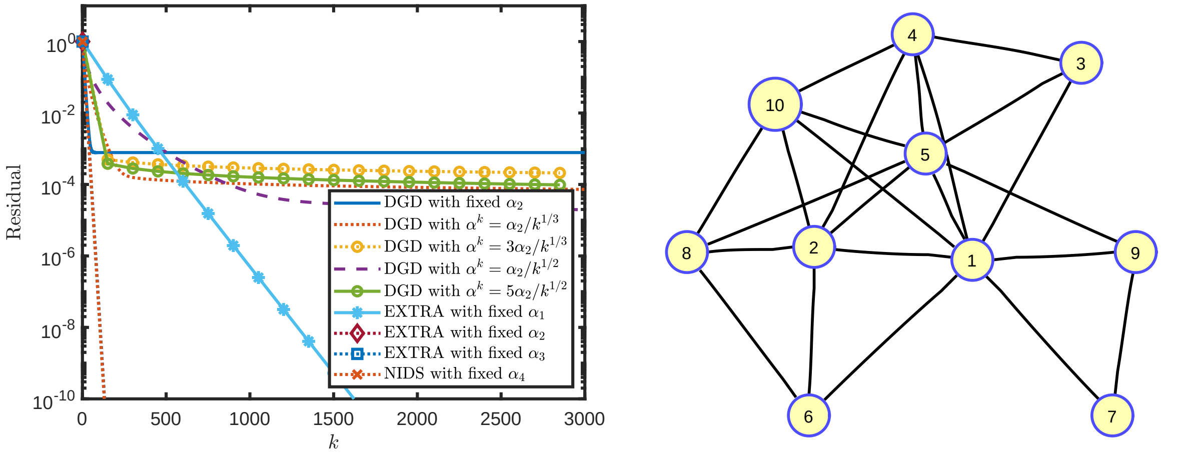


Figure 2.2: LEFT: the error $\frac{\|\mathbf{x}^k - \mathbf{x}^*\|_F}{\|\mathbf{x}^0 - \mathbf{x}^*\|_F}$ vs iterations for DGD with different stepsizes, EXTRA with three stepsizes, and NIDS. RIGHT: The random network with 10 nodes.

Then, we set $m_i = 10$ in Fig. 2.2. In this case, individual functions $f_i(x)$ and $\mathbf{f}(\mathbf{x})$ are strongly convex. NIDS and EXTRA with the largest stepsize lead the performance. Here two results of EXTRA are the same as that of NIDS although they are set with different stepsizes. The observation may indicate that there is an optimal choice of stepsize between α_2 and α_3 for both EXTRA and NIDS. By setting $\alpha_5 = \frac{5+3\lambda_{\min}(\mathbf{W})}{40+\mu_{\bar{f}}}$ for EXTRA and $\alpha_6 = \frac{2}{10+\mu_{\bar{f}}}$, we compare these algorithms in Fig. 2.3. This figure suggests that the optimal stepsize may depends on the problem/functions. How to find the optimal stepsize is an important research topic and beyond the scope of this chapter.

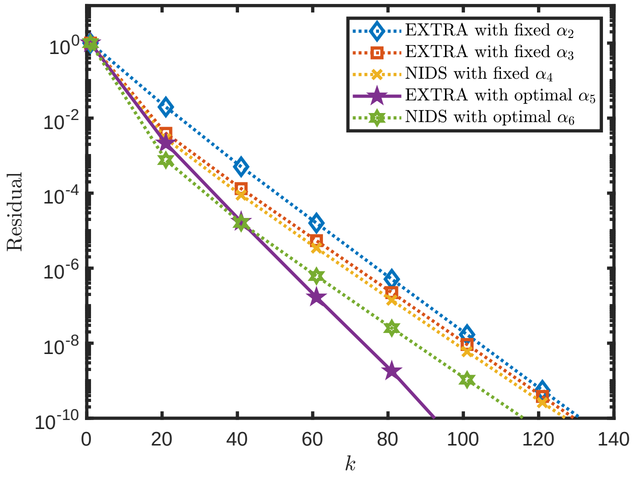


Figure 2.3: The comparison of proved stepsizes for EXTRA and NIDS with the optimal choice.

Next, we turn to the relaxed mixing matrices. Based on the previous created $\mathbf{W}$, we replace it by $\mathbf{W}_{\text{new}} = \frac{4\mathbf{W}-\mathbf{I}}{3}$ to scale the range of eigenvalues to $(-\frac{5}{3}, 1]$. In this case, some diagonal entries of $\mathbf{W}_{\text{new}}$ may be negative. We consider the worst topology of network, line topology, i.e., each agent has at most two neighbors. In this experiment, we solve the same problem using EXTRA and NIDS on the unrelaxed and relaxed mixing matrices, respectively, over the line. For NIDS, since the stepsize is network-independent, we relax the mixing matrix $\mathbf{W}$ to $\mathbf{W}_{\text{new}}$ more aggressively so that $\lambda_{\min}(\mathbf{W}_{\text{new}})$ approaches $-\frac{5}{3}$ and compare the performance with the unrelaxed case of NIDS under $\alpha = \frac{1}{5}$. For EXTRA, we set the stepsize to $\alpha = \frac{5+3\lambda_{\min}(\mathbf{W})}{40}$, and compare the performance with the relaxed one under the stepsize $\alpha = \frac{5+3\lambda_{\min}(\mathbf{W}_{\text{new}})}{40}$ where we only perturb $\mathbf{W}$ mildly so that $\lambda_{\min}(\mathbf{W}_{\text{new}})$ approaches -1. The result is shown in Fig. 2.4.

From Fig. 2.4, if the topology of network is weak, switching to the relaxed mixing matrix may offer better performance when using NIDS and EXTRA to solve the problem.

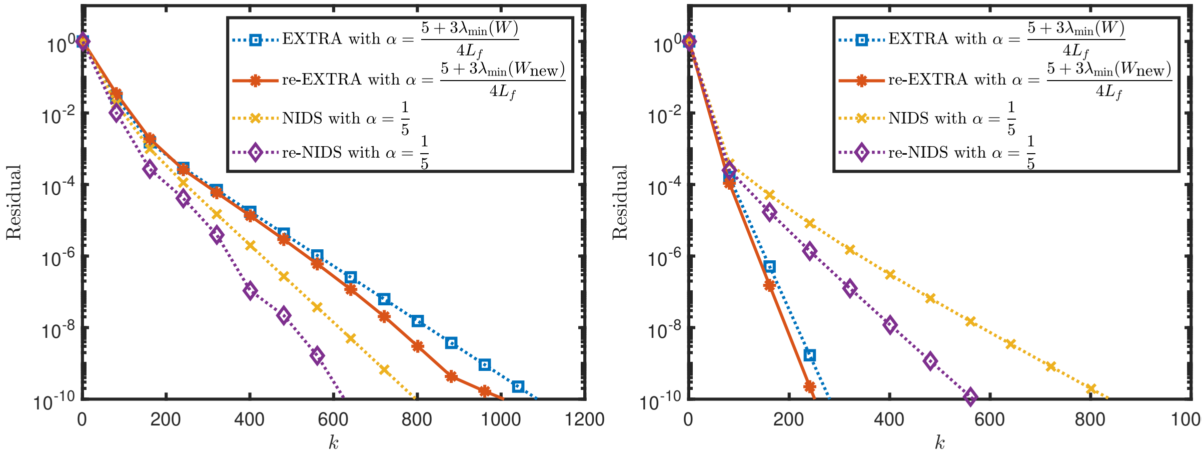


Figure 2.4: The figure of residuals $\frac{\|\mathbf{x}^k - \mathbf{x}^*\|_F}{\|\mathbf{x}^0 - \mathbf{x}^*\|_F}$ with respect to iteration. The left graph is for strongly convex $\bar{f}(x)$ and the right one is for strongly convex $\mathbf{f}(\mathbf{x})$. re-EXTRA and re-NIDS stand for implementing EXTRA and NIDS over relaxed $\mathbf{W}_{\text{new}}$.

The improvement for NIDS is more distinguished.

2.6 Conclusion

In this chapter, we relax the mixing matrices and prove the linear convergence of EXTRA and NIDS under the (restricted) strongly convexity assumption on $\bar{f}$. A larger upper bound of the stepsize is derived for EXTRA compared with that given in [1] and [58]. NIDS can choose a network-independent stepsize and this stepsize can be chosen as the same as that of centralized ones. We relax the conditions for the mixing matrices and the functions, while keeping the same stepsize.

In numerical experiments on linear regression, EXTRA with the larger stepsize converges faster than using the $\mu_{\bar{f}}$-dependent stepsize in [1]. Over the unrelaxed mixing matrix, NIDS leads the performance in most cases and is the easiest to implement. If the topology of network is weak, using the relaxed mixing matrix can accelerate NIDS. For EXTRA, in general, we may not choose the mixing matrices to be relaxed due to the tiny improvement, but the larger stepsize derived in the relaxed case is competent to be considered.

CHAPTER 3

DORE: A CENTRALIZED ALGORITHM WITH BIDIRECTIONAL COMMUNICATION COMPRESSION

3.1 Introduction

In the well-known parameter server framework [7, 11], each worker node evaluates its own stochastic gradient $\{\widetilde{\nabla} f_i(\mathbf{x}^k)\}_{i=1}^n$ and send it to the master node, which collects all gradients and calculates their average $(1/n)\sum_{i=1}^n \widetilde{\nabla} f_i(\mathbf{x}^k)$. Then the master node further takes the gradient descent step with the averaged gradient and broadcasts the new model parameter $\mathbf{x}^{k+1}$ to all worker nodes. It makes use of the computational resources from all nodes. In reality, the network bandwidth is often limited. Thus, the communication cost for the gradient transmission and model synchronization becomes the dominating bottlenecks as the number of nodes and the model size increase, which hinders the scalability and efficiency of SGD.

One common way to reduce the communication cost is to compress the gradient information by either gradient sparsification or quantization [13, 12, 15, 68, 69, 70, 71, 72] such that many fewer bits of information are needed to be transmitted. However, little attention has been paid on how to reduce the communication cost for model synchronization and the corresponding theoretical guarantees. Obviously, the model shares the same size as the gradient, so does the communication cost. Thus, merely compressing the gradient can reduce at most 50% of the communication cost, which suggests the importance of model compression. Notably, the compression of model parameters is much more challenging than gradient compression. One key obstacle is that its compression error cannot be well controlled by the step size γ and thus it cannot diminish like that in the gradient compression [20]. In this chapter, we aim to bridge this gap by investigating algorithms to compress the full communication in the optimization process and understanding their theoretical properties. Our contributions can be summarized as:

- We proposed DORE, which can compress both the gradient and the model information such that more than 95% of the communication cost can be reduced.
- We provided theoretical analyses to guarantee the convergence of DORE under strongly convex and nonconvex assumptions without the bounded gradient assumption.
- Our experiments demonstrate the superior efficiency of DORE comparing with the state-of-art baselines without degrading the convergence speed and the model accuracy.

3.2 Background

Recently, many works try to reduce the communication cost to speed up the distributed learning, especially for deep learning applications, where the size of the model is typically very large (so is the size of the gradient) while the network bandwidth is relatively limited. Below we briefly review relevant papers.

Gradient quantization and sparsification. Recent works [13, 12, 71, 17, 14] have shown that the information of the gradient can be quantized into a lower-precision vector such that fewer bits are needed in communication without loss of accuracy. [12] proposed 1Bit SGD that keeps the sign of each element in the gradient only. It empirically works well, and [14] provided theoretical analysis systematically. QSGD [13] utilizes an unbiased multi-level random quantization to compress the gradient while Terngrad [71] quantizes the gradient into ternary numbers $\{0, \pm 1\}$. In DIANA [17], the gradient difference is compressed and communicated contributing to the estimator of the gradient in the master node.

Another effective strategy to reduce the communication cost is sparsification. [70] proposed a convex optimization formulation to minimize the coding length of stochastic gradients. A more aggressive sparsification method is to keep the elements with relatively larger magnitude in gradients, such as top-k sparsification [15, 68, 73].

Model synchronization. The typical way for model synchronization is to broadcast model parameters to all worker nodes. Some works [69, 74] have been proposed to reduce model size by enforcing sparsity, but it cannot be applied to general optimization problems. Some alternatives including QSGD [13] and ECQ-SGD [72] choose to broadcast all quantized gradients to all other workers such that every worker can perform model update independently. However, all-to-all communication is not efficient since the number of transmitted bits increases dramatically in large-scale networks. DoubleSqueeze [18] applies compression on the averaged gradient with error compensation to speed up model synchronization.

Error compensation. [12] applied error compensation on 1Bit-SGD and achieved negligible loss of accuracy empirically. Recently, error compensation was further studied [72, 15, 16] to mitigate the error caused by compression. The general idea is to add the compressed error to the next compression step:

$$\hat{\mathbf{g}} = Q(\mathbf{g} + \mathbf{e}), \quad \mathbf{e} = (\mathbf{g} + \mathbf{e}) - \hat{\mathbf{g}}.$$

However, to the best of our knowledge, most of the algorithms with error compensation [72, 15, 16, 18] need to assume bounded gradient, i.e., $\mathbb{E}\|\mathrm{g}\|^2 \leq B$, and the convergence rate depends on this bound.

Contributions of DORE. The most related papers to DORE are DIANA [17] and DoubleSqueeze [18]. Similarly, DIANA compresses gradient difference on the worker side and achieves good convergence rate. However, it doesn't consider the compression in model synchronization, so at most 50% of the communication cost can be saved. DoubleSqueeze applies compression with error compensation on both worker and server sides, but it only considers nonconvex objective functions. Moreover, its analysis relies on a bounded gradient assumption, i.e., $\mathbb{E}\|\mathrm{g}\|^2 \leq B$, and the convergence error has a dependency on the gradient bound like most existed error compensation works.

In general, the uniform bound on the norm of the stochastic gradient is a strong assumption which might not hold in some cases. For example, it is violated in the strongly convex case [75, 76]. In this chapter, we design DORE, the first algorithm which utilizes gradient and model compression with error compensation without assuming bounded gradients. Unlike existing error compensation works, we provide a linear convergence rate to the $\mathcal{O}(\sigma)$ neighborhood of the optimal solution for strongly convex functions and a sublinear rate to the stationary point for nonconvex functions with linear speedup. In Table 3.1, we compare the asymptotic convergence rates of different quantized SGDs with DORE.

3.3 Double Residual Compression SGD

In this section, we introduce the proposed DOuble REsidual compression SGD (DORE) algorithm. Before that, we introduce a common assumption for the compression operator.

In this work, we adopt an assumption from [13, 71, 17] that the compression variance is linearly proportional to the magnitude.

Assumption 3.3.1. *The stochastic compression operator $Q : \mathbb{R}^d \to \mathbb{R}^d$ is unbiased, i.e.,* $\mathbb{E}Q(\mathbf{x}) = \mathbf{x}$ *and satisfies*

$$\mathbb{E}\|Q(\mathbf{x}) - \mathbf{x}\|^2 \leq C\|\mathbf{x}\|^2, \tag{3.1}$$

for a nonnegative constant C that is independent of $\mathbf{x}$*. We use $\hat{\mathbf{x}}$ to denote the compressed* $\mathbf{x}$*, i.e.,* $\hat{\mathbf{x}} \sim Q(\mathbf{x})$.

Many feasible compression operators can be applied to our algorithm since our theoretical analyses are built on this common assumption. Some examples of feasible stochastic compression operators include:

- *No Compression:* $C = 0$ when there is no compression.
- *Stochastic Quantization:* A real number $x \in [a, b], (a < b)$ is set to be a with probability $\frac{b-x}{b-a}$ and b with probability $\frac{x-a}{b-a}$, where a and b are predefined quantization levels [13]. It satisfies Assumption 3.3.1 when $ab > 0$ and $a < b$.

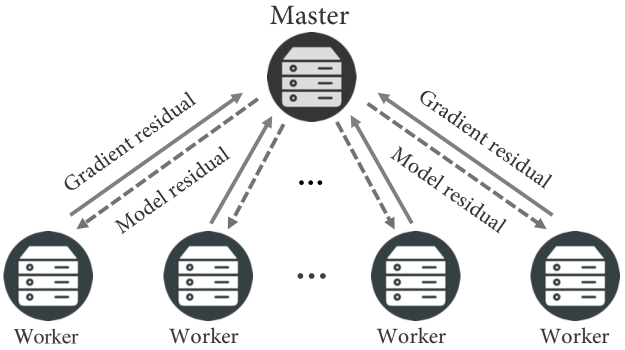


Figure 3.1: An illustration of DORE.

- *Stochastic Sparsification:* A real number x is set to be 0 with probability $1-p$ and $\frac{x}{p}$ with probability p [71]. It satisfies Assumption 3.3.1 with $C=(1/p)-1$.
- *p-norm Quantization:* A vector $\mathbf{x}$ is quantized element-wisely by $Q_p(\mathbf{x})=\|\mathbf{x}\|_p \operatorname{sign}(\mathbf{x}) \circ \xi$, where $\circ$ is the Hadamard product and ξ is a Bernoulli random vector satisfying $\xi_i \sim \text{Bernoulli}(\frac{|x_i|}{\|\mathbf{x}\|_p})$. It satisfies Assumption 3.3.1 with $C=\max_{\mathbf{x}\in\mathbb{R}^d}\frac{\|\mathbf{x}\|_1\|\mathbf{x}\|_p}{\|\mathbf{x}\|_2^2}-1$ [17]. To decrease the constant C for a higher accuracy, we can further decompose a vector $\mathbf{x}\in\mathbb{R}^d$ into blocks, i.e., $\mathbf{x}=(\mathbf{x}(1)^\top,\mathbf{x}(2)^\top,\cdots,\mathbf{x}(m)^\top)^\top$ with $\mathbf{x}(l)\in\mathbb{R}^{d_l}$ and $\sum_{l=1}^m d_l=d$, and compress the blocks independently.

3.3.1 The Proposed DORE

Many previous works [13, 12, 71] reduce the communication cost of P-SGD by quantizing the stochastic gradient before sending it to the master node, but there are several intrinsic issues.

First, these algorithms will incur extra optimization error intrinsically. Let's consider the case when the algorithm converges to the optimal point $\mathbf{x}^*$. By the first-order optimality, we have $(1/n)\sum_{i=1}^n \nabla f_i(\mathbf{x}^*)=\mathbf{0}.$, the data distributions may be different for different worker nodes in general, and thus we may have $\nabla f_i(\mathbf{x}^*)\neq\nabla f_j(\mathbf{x}^*),\forall i,j\in\{1,\ldots,n\}$ and $i\neq j$. In other words, each individual $\nabla f_i(\mathbf{x}^*)$ may be far away from zero. This will cause large compression variance according to Assumption 3.3.1, which indicates that the upper bound of compression variance $\mathbb{E}\|Q(\mathbf{x})-\mathbf{x}\|^2$ is linearly proportional to the magnitude of $\mathbf{x}$.

Second, most existing algorithms [12, 13, 71, 14, 72, 17] need to broadcast the model or gradient to all worker nodes in each iteration. It is a considerable bottleneck for efficient optimization since the amount of bits to transmit is the same as the uncompressed gradient. DoubleSqueeze [18] is able to apply compression on both worker and server sides. However, its analysis depends on a strong assumption on bounded gradient. Meanwhile, no theoretical guarantees are provided for the convex problems.

Algorithm 3.1: DORE[1]

```
Input: Stepsize $\alpha, \beta, \gamma, \eta$, initialize $\mathbf{h}^0 = \mathbf{h}_i^0 = \mathbf{0}^d$, $\hat{\mathbf{x}}_i^0 = \hat{\mathbf{x}}^0,\ \forall i \in \{1, \dots, n\}$.
for $k = 1, 2, \cdots, K-1$ do
    For each worker $i \in \{1, 2, \cdots, n\}$:
    Sample $\mathbf{g}_i^k$ such that $\mathbb{E}[\mathbf{g}_i^k|\hat{\mathbf{x}}_i^k] = \nabla f_i(\hat{\mathbf{x}}_i^k)$
    Gradient residual: $\Delta_i^k = \mathbf{g}_i^k - \mathbf{h}_i^k$
    Compression: $\hat{\Delta}_i^k = Q(\Delta_i^k)$
    $\mathbf{h}_i^{k+1} = \mathbf{h}_i^k + \alpha\hat{\Delta}_i^k$
    { $\hat{\mathbf{g}}_i^k = \mathbf{h}_i^k + \hat{\Delta}_i^k$ }
    Send $\hat{\Delta}_i^k$ to the master
    Receive $\hat{\mathbf{q}}^k$ from the master
    $\hat{\mathbf{x}}_i^{k+1} = \hat{\mathbf{x}}_i^k + \beta\hat{\mathbf{q}}^k$
    For the master:
    Receive $\{\hat{\Delta}_i^k\}$ from workers
    $\hat{\Delta}^k = 1/n\sum_i^n \hat{\Delta}_i^k$
    $\hat{\mathbf{g}}^k = \mathbf{h}^k + \hat{\Delta}^k$  $\{= 1/n\sum_i^n \hat{\mathbf{g}}_i^k$ }
    $\mathbf{x}^{k+1} = \mathbf{prox}_{\gamma R}(\hat{\mathbf{x}}^k - \gamma\hat{\mathbf{g}}^k)$
    $\mathbf{h}^{k+1} = \mathbf{h}^k + \alpha\hat{\Delta}^k$
    Model residual: $\mathbf{q}^k = \mathbf{x}^{k+1} - \hat{\mathbf{x}}^k + \eta\mathbf{e}^k$
    Compression: $\hat{\mathbf{q}}^k = Q(\mathbf{q}^k)$
    $\mathbf{e}^{k+1} = \mathbf{q}^k - \hat{\mathbf{q}}^k$
    $\hat{\mathbf{x}}^{k+1} = \hat{\mathbf{x}}^k + \beta\hat{\mathbf{q}}^k$
    Broadcast $\hat{\mathbf{q}}^k$ to workers
end for
Output: $\hat{\mathbf{x}}^K$ or any $\hat{\mathbf{x}}_i^K$
```

We proposed DORE to address all aforementioned issues. Our motivation is that the gradient should change smoothly for smooth functions so that each worker node can keep a state variable $\mathbf{h}_i^k$ to track its previous gradient information. As a result, the residual between new gradient and the state $\mathbf{h}_i^k$ should decrease, and the compression variance of the residual can be well bounded. On the other hand, as the algorithm converges, the model would only change slightly. Therefore, we propose to compress the model residual such that the compression variance can be minimized and also well bounded. We also compensate the model residual compression error into next iteration to achieve a better convergence. Due to the advantages of the proposed double residual compression scheme, we can derive the fastest convergence rate through analyses without the bounded gradient assumption. Below are some key steps of our algorithm as showed in Algorithm 3.1 and Figure 3.1:

[lines 4-9]: each worker node sends the compressed gradient residual ($\hat{\Delta}_i^k$) to the master node and updates its state $\mathbf{h}_i^k$ with $\hat{\Delta}_i^k$;

[lines 13-15]: the master node gathers the compressed gradient residual ($\{\hat{\Delta}_i^k)\}$ from all worker nodes and recovers the averaged gradient $\hat{\mathbf{g}}^k$ based on its state $\mathbf{h}^k$;

[lines 16]: the master node applies gradient descent algorithms (possibly with the proximal operator);

[lines 18-22]: the master node broadcasts the compressed model residual with error compensation ($\hat{\mathbf{q}}^k$) to all worker nodes and updates the model;

[lines 10-11]: each worker node receives the compressed model residual ($\hat{\mathbf{q}}^k$) and updates its model $\mathbf{x}_i^k$.

In the algorithm, the state $\mathbf{h}_i^k$ serves as an exponential moving average of the local gradient in expectation, i.e., $\mathbb{E}_Q \mathbf{h}_i^{k+1} = (1-\alpha)\mathbf{h}_i^k + \alpha \mathbf{g}_i^k$, as proved in Lemma A.4.1. Therefore, as the iteration approaches the optimum, $\mathbf{h}_i^k$ will also approach the local gradient $\nabla f_i(\mathbf{x}^*)$ rapidly which contributes to small gradient residual and consequently small compression variance. Similar difference compression techniques are also proposed in DIANA and its variance-reduced variant [17, 77].

[1] Equations in the curly bracket are just notations for the proof but does not need to computed actually.

3.3.2 Discussion

In this subsection, we provide more detailed discussions about DORE including model initialization, model update, the special smooth case as well as the compression rate of communication.

Initialization. It is important to take the identical initialization $\hat{\mathbf{x}}^0$ for all worker and master nodes. It is easy to be ensured by either setting the same random seed or broadcasting the model once at the beginning. In this way, although we don't need to broadcast the model parameters directly, every worker node updates the model $\hat{\mathbf{x}}^k$ in the same way. Thus we can keep their model parameters identical. Otherwise, the model inconsistency needs to be considered.

Model update. It is worth noting that although we can choose an accurate model $\mathbf{x}^{k+1}$ as the next iteration in the master node, we use $\hat{\mathbf{x}}^{k+1}$ instead. In this way, we can ensure that the gradient descent algorithm is applied based on the exact stochastic gradient which is evaluated on $\hat{\mathbf{x}}_i^k$ at each worker node. This dispels the intricacy to deal with inexact gradient evaluated on $\mathbf{x}^k$ and thus it simplifies the convergence analysis.

Smooth case. In the smooth case, i.e., $R = 0$, Algorithm 3.1 can be simplified. The master node quantizes the recovered averaged gradient with error compensation and broadcasts it to all worker nodes. The details of this simplified case can be found in Appendix A.3.

Compression rate. The compression of the gradient information can reduce at most 50% of the communication cost since it only considers compression during gradient aggregation while ignoring the model synchronization. However, DORE can further cut down the remaining 50% communication.

Taking the blockwise p-norm quantization as an example, every element of $\mathbf{x}$ can be represented by $\frac{3}{2}$ bits using the simple ternary coding $\{0, \pm 1\}$, along with one magnitude for each block. For example, if we consider the uniform block size b, the number of bits to represent a d-dimension vector of 32 bit float-point numbers can be reduced from $32d$

bits to $32\frac{d}{b} + \frac{3}{2}d$ bits. As long as the block size b is relatively large with respect to the constant 32, the cost $32\frac{d}{b}$ for storing the float-point number is relatively small such that the compression rate is close to $32d/(\frac{3}{2}d) \approx 21.3$ times (for example, 19.7 times when $b = 256$).

Applying this quantization, QSGD, Terngrad, MEM-SGD, and DIANA need to transmit $(32d+32\frac{d}{b}+\frac{3}{2}d)$ bits per iteration and thus they are able to cut down 47% of the overall $2 \times 32d$ bits per iteration through gradient compression when $b = 256$. But with DORE, we only need to transmit $2(32\frac{d}{b} + \frac{3}{2}d)$ bits per iteration. Thus DORE can reduce over 95% of the total communication by compressing both the gradient and model transmission. More efficient coding techniques such as Elias coding [78] can be applied to further reduce the number of bits per iteration.

3.4 Convergence Analysis

To show the convergence of DORE, we make the following commonly used assumptions.

Assumption 3.4.1. *Each worker node samples an unbiased estimator of the gradient stochastically with bounded variance, i.e., for $i = 1, 2, \cdots, n$ and $\forall \mathbf{x} \in \mathbb{R}^d$,*

$$\mathbb{E}[\mathbf{g}_i|\mathbf{x}] = \nabla f_i(\mathbf{x}), \quad \mathbb{E}\|\mathbf{g}_i - \nabla f_i(\mathbf{x})\|^2 \leq \sigma_i^2, \tag{3.2}$$

where $\mathbf{g}_i$ is the estimator of ∇f_i at $\mathbf{x}$. In addition, we define $\sigma^2 = \frac{1}{n}\sum_{i=1}^n \sigma_i^2$.

Assumption 3.4.2. *Each f_i is L-Lipschitz differentiable, i.e., for $i = 1, 2, \cdots, n$ and $\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$,*

$$f_i(\mathbf{x}) \leq f_i(\mathbf{y}) + \langle \nabla f_i(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2. \tag{3.3}$$

Assumption 3.4.3. *Each f_i is μ-strongly convex ($\mu \geq 0$), i.e., for $i = 1, 2, \cdots, n$ and $\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$,*

$$f_i(\mathbf{x}) \geq f_i(\mathbf{y}) + \langle \nabla f_i(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{\mu}{2}\|\mathbf{x} - \mathbf{y}\|^2. \tag{3.4}$$

For simplicity, we use the same compression operator for all worker nodes, and the master node can apply a different compression operator. We denote the constants in Assumption 3.3.1 as C_q and C_q^m for the worker and master nodes, respectively. Then we set

α and β in both algorithms to satisfy

$$\begin{aligned}\frac{1-\sqrt{1-\frac{4C_q(C_q+1)}{nc}}}{2(C_q+1)} \leq \alpha \leq \frac{1+\sqrt{1-\frac{4C_q(C_q+1)}{nc}}}{2(C_q+1)},\\ 0<\beta \leq \frac{1}{C_q^m+1},\end{aligned} \tag{3.5}$$

with $c \geq \frac{4C_q(C_q+1)}{n}$.

We consider two scenarios in the following two subsections: f is strongly convex with a convex regularizer R and f is nonconvex with $R=0$.

3.4.1 The Strongly Convex Case

Theorem 3.4.1. *Under Assumptions 3.3.1-3.4.3, if α and β in Algorithm 3.1 satisfy* (3.5), *η and γ satisfy*

$$\begin{aligned}\eta < \min\Bigg(&\frac{-C_q^m+\sqrt{(C_q^m)^2+4(1-(C_q^m+1)\beta)}}{2C_q^m},\\ &\frac{4\mu L}{(\mu+L)^2(1+c\alpha)-4\mu L}\Bigg),\end{aligned} \tag{3.6}$$

$$\frac{\eta(\mu+L)}{2(1+\eta)\mu L} \leq \gamma \leq \frac{2}{(1+c\alpha)(\mu+L)}, \tag{3.7}$$

then we have

$$\mathbf{V}^{k+1} \leq \rho^k \mathbf{V}^1 + \frac{(1+\eta)(1+nc\alpha)}{n(1-\rho)}\beta\gamma^2\sigma^2, \tag{3.8}$$

with

$$\begin{aligned}\mathbf{V}^k =&\beta(1-(C_q^m+1)\beta)\mathbb{E}\|\mathbf{q}^{k-1}\|^2+\mathbb{E}\|\hat{\mathbf{x}}^k-\mathbf{x}^*\|^2\\ &+\frac{(1+\eta)c\beta\gamma^2}{n}\sum_{i=1}^n \mathbb{E}\|\mathbf{h}_i^k-\nabla f_i(\mathbf{x}^*)\|^2,\\ \rho =&\max\Big(\frac{(\eta^2+\eta)C_q^m}{1-(C_q^m+1)\beta}, 1+\eta\beta-\frac{2(1+\eta)\beta\gamma\mu L}{\mu+L}, 1-\alpha\Big)<1.\end{aligned}$$

Corollary 3.4.1. *When there is no error compensation and we set $\eta = 0$, then $\rho = \max(1 - \frac{2\beta\gamma\mu L}{\mu+L}, 1-\alpha)$. If we further set*

$$\alpha = \frac{1}{2(C_q+1)}, \quad \beta = \frac{1}{C_q^m+1}, \quad c = \frac{4C_q(C_q+1)}{n}, \tag{3.9}$$

and choose the largest stepsize $\gamma = \frac{2}{(\mu+L)(1+2C_q/n)}$, the convergent factor is

$$(1-\rho)^{-1} = \max\Big(2(C_q+1), (C_q^m+1)\frac{(\mu+L)^2}{2\mu L}\Big(\frac{1}{2}+\frac{C_q}{n}\Big)\Big). \tag{3.10}$$

Remark 3.4.1. *In particular, suppose $\{\Delta_i\}_{i=1}^n$ are compressed using the Bernoulli p-norm quantization with the largest block size $d_{\max}$, then $C_q = \frac{1}{\alpha^w} - 1$, with*

$$\alpha^w = \min_{\mathbf{0}\neq\mathbf{x}\in\mathbb{R}^{d_{\max}}} \frac{\|\mathbf{x}\|_2^2}{\|\mathbf{x}\|_1\|\mathbf{x}\|_p} \leq 1.$$

Similarly, $\mathbf{q}$ is compressed using the Bernoulli p-norm quantization with $C_q^m = \frac{1}{\alpha^m} - 1$. Then the linear convergent factor is

$$(1-\rho)^{-1} = \max\Big\{\frac{2}{\alpha^w}, \frac{1}{\alpha^m}\frac{(\mu+L)^2}{\mu L}\Big(\frac{1}{2}-\frac{2}{n}+\frac{2}{n\alpha^w}\Big)\Big\}. \tag{3.11}$$

While the result of DIANA in [17] is $\max\Big\{\frac{2}{\alpha^w}, \frac{\mu+L}{\mu}\Big(\frac{1}{2}-\frac{1}{n}+\frac{1}{n\alpha^w}\Big)\Big\}$, which is larger than (3.11) *with $\alpha^m = 1$ (no compression for the model). When there is no compression for Δ_i, i.e., $\alpha^w = 1$, the algorithm reduces to the gradient descent, and the linear convergent factor is the same as that of the gradient descent for strongly convex functions.*

Remark 3.4.2. *Although error compensation often improves the convergence in practice, in theory, no compensation, i.e., $\eta = 0$, provides the best convergence rate. This is because we don't have much information of the error being compensated. Filling this gap will be an interesting future direction.*

3.4.2 The Nonconvex Case with $R = 0$

Theorem 3.4.2. *Under Assumptions 3.3.1-3.4.2 and the additional assumption that each worker samples the gradient from the full dataset, we set α and β according to* (3.5)*. By choosing*

$$\gamma \leq \min\Big\{\frac{-1+\sqrt{1+\frac{48L^2\beta^2(C_q^m+1)^2}{C_q^m}}}{12L\beta(C_q^m+1)}, \frac{1}{6L\beta(1+c\alpha)(C_q^m+1)}\Big\},$$

we have

$$\frac{\frac{\beta}{2} - 3(1+c\alpha)(C_q^m+1)L\beta^2\gamma}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2$$
$$\leq \frac{\Lambda^1 - \Lambda^{K+1}}{\gamma K} + \frac{3(C_q^m+1)(1+nc\alpha)L\beta^2\sigma^2\gamma}{n}, \tag{3.12}$$

where

$$\Lambda^k = (C_q^m+1)L\beta^2\|\mathbf{q}^{k-1}\|^2 + f(\hat{\mathbf{x}}^k) - f^* + 3c(C_q^m+1)L\beta^2\gamma^2\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\|\mathbf{h}_i^k\|^2. \tag{3.13}$$

Corollary 3.4.2. *Let $\alpha = \frac{1}{2(C_q+1)}$, $\beta = \frac{1}{C_q^m+1}$, and $c = \frac{4C_q(C_q+1)}{n}$, then $1+nc\alpha$ is a fixed constant. If $\gamma = \frac{1}{12L(1+c\alpha)(1+\sqrt{K/n})}$, when K is relatively large, we have*

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2 \lesssim \frac{1}{K} + \frac{1}{\sqrt{Kn}}. \tag{3.14}$$

Remark 3.4.3. *The dominant term in* (3.14) *is $O(1/\sqrt{Kn})$, which implies that the sample complexity of each worker node is $O(1/(n\epsilon^2))$ in average to achieve an ϵ-accurate solution. It shows that, same as DoubleSqueeze in [18], DORE is able to perform linear speedup. Furthermore, this convergence result is the same as the P-SGD without compression. Note that DoubleSqueeze has an extra term $(1/K)^{\frac{2}{3}}$, and its convergence requires the bounded variance of the compression operator.*

3.5 Numerical Experiments

In this section, we validate the theoretical results and demonstrate the superior performance of DORE. Our experimental results demonstrate that (1) DORE achieves similar convergence speed as full-precision SGD and state-of-art quantized SGD baselines and (2) its iteration time is much smaller than most existing algorithms, supporting the superior communication efficiency of DORE.

To make a fair comparison, we choose the same Bernoulli ∞-norm quantization as described in Section 3.3 and the quantization block size is 256 for all experiments if not

Algorithm	Direction	Compression	Linear Rate	Nonconvex Rate
QSGD	Uni.	2-norm Quantization	N/A	$\frac{1}{K} + B$
DIANA	Uni.	p-norm Quantization	✓	$\frac{1}{\sqrt{Kn}} + \frac{1}{K}$
DoubleSqueeze	Bi.	Bounded Variance	N/A	$\frac{1}{\sqrt{Kn}} + \frac{1}{K^{2/3}} + \frac{1}{K}$
DORE	Bi.	Assumption 3.3.1	✓	$\frac{1}{\sqrt{Kn}} + \frac{1}{K}$

Table 3.1: A comparison between related algorithms. The second column indicates whether algorithm compresses only the gradients or both the gradients and the model. DORE is able to converges linearly to the $\mathcal{O}(\sigma)$ neighborhood of optimal point like full-precision SGD and DIANA in the strongly convex case while achieving much better communication efficiency. DORE also admits linear speedup in the nonconvex case like DoubleSqueeze but DORE doesn't require the assumptions of bounded compression error or bounded gradient.

being explicitly stated because ∞-norm quantization is unbiased and commonly used. The parameters α, β, η for DORE are chosen to be $0.1, 1$ and 1, respectively.

The baselines we choose to compare include SGD, QSGD [13], MEM-SGD [15], DIANA [17], DoubleSqueeze and DoubleSqueeze (topk) [18]. SGD is the vanilla SGD without any compression and QSGD quantizes the gradient directly. MEM-SGD is the QSGD with error compensation. DIANA, which only compresses and transmits the gradient difference, is a special case of the proposed DORE. DoubleSqueeze quantizes both the gradient on the workers and the averaged gradient on the server with error compensation. Although DoubleSqueeze is claimed to work well with both biased and unbiased compression, in our experiment it converges much slower and suffers the loss of accuracy with unbiased compression. Thus, we also compare with DoubleSqueeze using the Top-k compression as presented in [18].

3.5.1 Strongly convex

To verify the convergence for strongly convex and smooth objective functions, we conduct the experiment on a linear regression problem: $f(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda\|\mathbf{x}\|^2$. The data matrix $\mathbf{A} \in \mathbb{R}^{1200\times 500}$ and optimal solution $\mathbf{x}_* \in \mathbb{R}^{500}$ are randomly synthesized. Then we generate the prediction $\mathbf{b}$ by sampling from a Gaussian distribution whose mean is

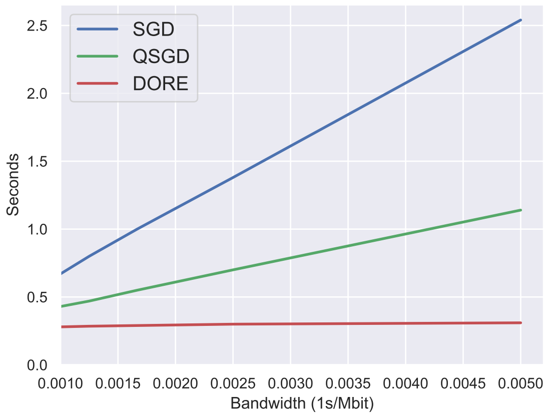


Figure 3.2: Per iteration time cost on Resnet18 for SGD, QSGD, and DORE. It is tested in a shared cluster environment connected by Gigabit Ethernet interface. DORE speeds up the training process significantly by mitigating the communication bottleneck.

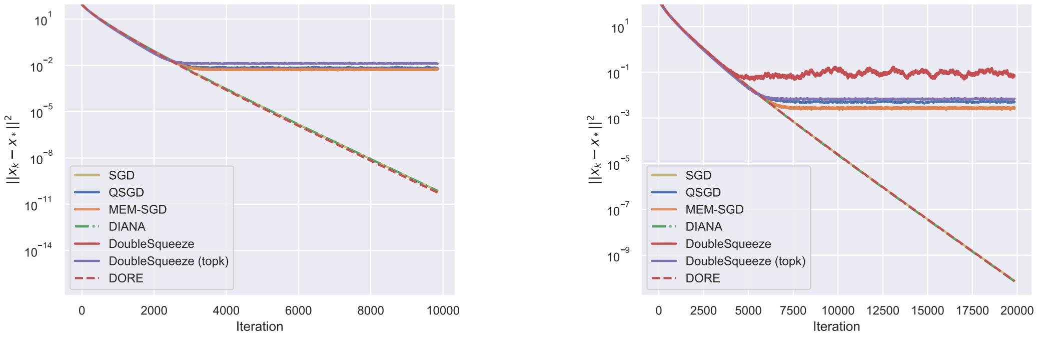
$\mathbf{A}\mathbf{x}_*$. The rows of the data matrix $\mathbf{A}$ are allocated evenly to 20 worker nodes. To better verify the linear convergence to the $\mathcal{O}(\sigma)$ neighborhood around the optimal solution, we take the full gradient in each node for all algorithms to exclude the effect of the gradient variance ($\sigma = 0$).

As showed in Figure 3.3, with full gradient and a constant learning rate, DORE converges linearly, same as SGD and DIANA, but QSGD, MEM-SGD, DoubleSqueeze, as well as DoubleSqueeze (topk) converge to a neighborhood of the optimal point. This is because these algorithms assume the bounded gradient and their convergence errors depend on that bound. Although they converge to the optimal solution using a diminishing step size, their converge rates will be much slower.

In addition, we also validate that the norms of the gradient and model residual decrease exponentially, and it explains the linear convergence behavior of DORE. For more details, please refer to Appendix A.1.

3.5.2 Nonconvex

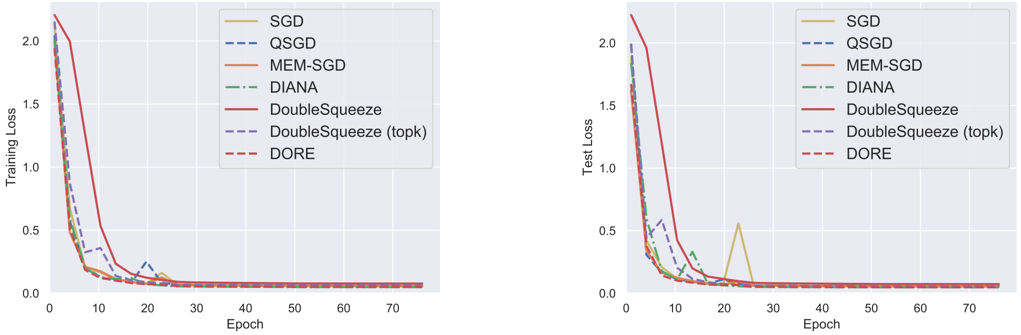
To verify the convergence in the nonconvex case, we test the proposed DORE with two classical deep neural networks on two representative datasets, i.e., LeNet [79] on MNIST and Resnet18 [80] on CIFAR10. In the experiment, we use 1 parameter server and 10 workers, each of which is equipped with an NVIDIA Tesla K80 GPU. The batch size for each worker node is 256. We use 0.1 and 0.01 as the initial learning rates for LeNet and Resnet18, and decrease them by a factor of 0.1 after every 25 and 100 epochs, respectively. All parameter settings are the same for all algorithms.



(a) Learning rate=0.05

(b) Learning rate=0.025

Figure 3.3: Linear regression on synthetic data. When the learning rate is 0.05, DoubleSqueeze diverges. In both cases, DORE, SGD, and DIANA converge linearly to the optimal point, while QSGD, MEM-SGD, DoubleSqueeze, and DoubleSqueeze (topk) only converge to the neighborhood even when full gradient is available.

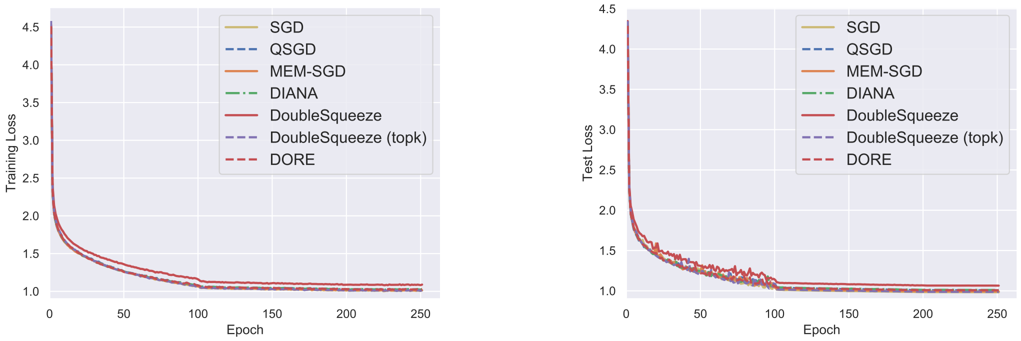


(a) Training loss

(b) Test Loss

Figure 3.4: LeNet trained on MNIST. DORE converges similarly as most baselines. It outperforms DoubleSqueeze using the same compression method while has similar performance as DoubleSqueeze (topk).

Figures 3.4 and 3.5 show the training loss and test loss for each epoch during the training of LeNet on the MNIST dataset and Resnet18 on CIFAR10 dataset. The results indicate that in the nonconvex case, even with both compressed gradient and model information, DORE can still achieve similar convergence speed as full-precision SGD and other quantized SGD variants. DORE achieves much better convergence speed than DoubleSqueeze



(a) Training Loss

(b) Test Loss

Figure 3.5: Resnet18 trained on CIFAR10. DORE achieves similar convergence and accuracy as most baselines. DoubeSuqeeze converges slower and suffers from the higher loss but it works well with topk compression.

using the same compression method and converges similarly with DoubleSqueeze with Topk compression as presented in [18]. We also validate via parameter sensitivity in Appendix A.2 that DORE performs consistently well under different parameter settings such as compression block size, α, β and η.

3.5.3 Communication Efficiency

In terms of communication cost, DORE enjoys the benefit of extremely efficient communication. As one example, under the same setting as the Resnet18 experiment described in the previous section, we test the time cost per iteration for SGD, QSGD, and DORE under varied network bandwidth. We didn't test MEM-SGD, DIANA, and DoubleSqueeze because MEM-SGD, DIANA have similar time cost as QSGD while DoubleSqueeze has similar time cost as DORE. The result showed in Figure 3.2 indicates that as the bandwidth becomes worse, with both gradient and model compression, the advantage of DORE becomes more remarkable compared to the baselines that don't apply compression for model synchronization .

3.6 Conclusion

Communication cost is the severe bottleneck for distributed training of modern large-scale machine learning models. Extensive works have compressed the gradient information to be transferred during the training process, but model compression is rather limited due to its intrinsic difficulty. In this chapter, we proposed the Double Residual Compression SGD named DORE to compress both gradient and model communication that can mitigate this bottleneck prominently. The theoretical analyses suggest good convergence rate of DORE under weak assumptions. Furthermore, DORE is able to reduce 95% of the communication cost while maintaining similar convergence rate and model accuracy compared with the full-precision SGD.

CHAPTER 4

PROX-LEAD: A LINEAR CONVERGENT CONVERGENT ALGORITHM WITH EFFICIENT COMMUNICATION FOR COMPOSITE PROBLEMS

4.1 Introduction

In recent years, the communication cost has become the bottleneck in the distributed training of machine learning models, given that the computation becomes much faster with powerful computing devices such as GPUs and TPUs. Therefore, the communication efficiency gains increasing attention in the algorithm design. On the one hand, decentralized communication has been shown to be an effective and important direction for improving communication efficiency [28]. On the other hand, various communication compression techniques, such as quantization and sparsification, which are originally developed for centralized settings [12, 13, 14, 15, 16, 17, 19, 18], have been shown to be significant in reducing the communication cost for decentralized optimization [20, 21, 24, 25, 33, 81]. These works have exhibited the great potential of decentralized optimization with communication compression in speeding up decentralized machine learning.

The composite problem (1.2) abstracts many important applications such as regularized empirical risk minimization in statistics and machine learning, and optimal control of multi-agent systems. More specifically, we consider two different settings on the smooth components of the objective function, i.e., the general stochastic setting and the finite-sum setting. In the general stochastic setting, the problem follows (1.2) where the local functions $\{f_i\}$ are defined as the expectation over the general local sample distributions $\{\mathcal{D}_i\}$. In the finite-sum setting, we consider the discrete distribution on local nodes, and the local functions $\{f_i\}$ are defined as the unweighted average over local samples, $f_i(\mathbf{x}) = \frac{1}{m}\sum_{j=1}^{m} f_{ij}(\mathbf{x})$, where each f_{ij} stands for the loss function defined on the jth batch of samples at node i. In this work, we assume that the number of batches m is the same for all nodes for simplicity. Note that it can be easily generalized to the case when the nodes have different numbers of batches.

Contribution. The main contribution of this chapter is a Proximal gradient LinEAr convergent Decentralized algorithm with compression, Prox-LEAD, which solves the problem (1.2).

Specifically, the contributions can be summarized as follows:

- We propose the first decentralized stochastic proximal gradient algorithm with compressed communication, Prox-LEAD. It converges linearly up to the neighborhood of the optimal solution in the general stochastic setting. In the finite-sum setting, we establish the linear convergence to the exact solution for Prox-LEAD's two variance reduction variants, i.e., Loopless SVRG and SAGA.

- We provide a rigorous theory of Prox-LEAD on the compressed communication and convergence complexities in different settings on the smooth component of the problem. Without the restriction on data heterogeneity and gradient boundedness, Prox-LEAD maintains a comparable convergence rate compared to the uncompressed counterpart. Our theorems indicate that Prox-LEAD works with arbitrary compression precision. Moreover, with reasonably aggressive compression, Prox-LEAD significantly reduces the communication cost almost for free.

- Our algorithmic framework builds bridges between many known algorithms. Without involving compression, it provides stochastic and variance reduction variants of some deterministic algorithms, and it has a better convergence complexity against the existing non-accelerated stochastic decentralized algorithms. When the nonsmooth regularizer is absent, it reduces to LEAD and achieves a better convergence complexity. The framework also enlightens other primal-dual algorithms to apply compressed communication and reduce the impact of inexact primal and dual iterations.

- We present comprehensive experiments to verify our theorems and the effectiveness of the proposed algorithm in different settings. The comparison with state-of-art algorithms demonstrates the superiority of Prox-LEAD and its stochastic variants. Moreover, it is robust to parameter tuning, which exhibits great advantages in practice.

The rest of this chapter is organized as follows. Section 4.1.1 and Section 4.1.2 summarize related works and introduce the notations used in this chapter, respectively. In Section 4.2, the proposed algorithm Prox-LEAD, motivation and derivation, as well as convergence complexities are presented. The assumptions on regularity are introduced in Section 4.3, and the convergence analyses and major theorems are illustrated in Section 4.4. Importantly, we also detail the connection with existing algorithms in Section B.2. Finally, numerical experiments are presented in Section 4.5.

4.1.1 Related Work

Many algorithms were proposed to solve the decentralized optimization of the average of functions defined over agents in networks. The early decentralized algorithms can be traced back to the work by [82]. DGD [26] is the most classical decentralized algorithm. It is intuitive and simple but converges slowly due to the diminishing stepsize that is needed to obtain the optimal solution [27]. Its stochastic version D-PSGD [28] has been shown effective for training nonconvex deep learning models. Algorithms based on primal-dual formulations or gradient tracking are proposed to eliminate the convergence bias in DGD-type algorithms and improve the convergence rate, such as D-ADMM [83], DLM [29], EXTRA [1], NIDS [2], D^2 [84], Exact Diffusion [53], NEXT [50], DIGing [45], Harnessing [46], SONATA [85], GSGT [86], OPTRA [87], etc. There are also dual-based methods which apply gradient methods on the dual formulation [55, 88, 56]. These algorithms are able to achieve optimal bounds but requires computing the non-trivial gradient of the dual function.

To improve the communication efficiency, communication compression is first applied to decentralized settings by [20]. It proposes two algorithms, i.e., DCD-SGD and ECD-SGD, which require compression of high accuracy and are not stable with aggressive compression. [21, 22] introduce QDGD and QuanTimed-DSGD to achieve exact convergence with small stepsize and the convergence is slow. DeepSqueeze [23] compensates the compression error to the compression in the next iteration. Motivated by the quantized average consensus algorithms, such as the work in [89], the quantized gossip algorithm Choco-Gossip [24] converges linearly to the consensual solution. Combining Choco-Gossip and D-PSGD leads to a decentralized algorithm with compression, Choco-SGD, which converges sublinearly under the strong convexity and gradient boundedness assumptions. Its nonconvex variant is further analyzed in [25]. A new compression scheme using the modulo operation is introduced in [90] for decentralized optimization. A general algorithmic framework aiming to maintain the linear convergence of distributed optimization under compressed communication is considered in [91]. It requires a contractive property that is not satisfied by many decentralized algorithms including the algorithms proposed in this chapter. A preliminary version of this work [33] proposes the first linear convergent decentralized algorithm with compression, LEAD. However, the composite problem is not considered in LEAD, and LEAD is deficient in dealing with stochastic gradients. [81] introduces a linear convergent algorithm for decentralized optimization with communication compression based on a primal-dual decentralized algorithm [92]. Communication compression is also proposed for gradient tracking algorithms [93, 94, 95], but they require double communication cost since two vectors need to be transmitted in each communication run.

Variance reduction techniques such as SAG [96], SVRG [97], SAGA [98] and Loopless SVRG [99] have been introduced to accelerate stochastic optimization problems with finite-sum structure. SVRG-type gradient estimator requires more gradient evaluation but they are memory friendly. SAGA reduces the number of gradient evaluation in each iteration but it requires more memory space. Variance reduction has been applied to decentralized optimization [100, 101, 102, 103, 104, 81].

Decentralized algorithms such as PG-EXTRA [57] and NIDS [2] are proposed for composite optimization, and the sublinear rate is proved when the smooth component is strongly convex and smooth. [58] introduces a proximal gradient algorithm (P2D2), [105] enhances the convergece rate of SONATA, and the linear convergence rate is proved in both literature when the nonsmooth component is shared across all nodes. A proximal unified decentralized algorithm (PUDA) [106] and another proximal decentralized algorithmic framework [107] unify many existing algorithms and establish linear convergence for composite optimization when the nonsmooth component is shared. A decentralized accelerated proximal gradient descent algorithm in proposed in [108]. However, communication compression and stochastic optimization are not considered in these algorithms.

4.1.2 Notation

We clarify commonly used notation in this section. We use bold lower-case letters to denote column vectors in $\mathbb{R}^p$ and bold upper-case letters for matrices in $\mathbb{R}^{n\times p}$. The lower-case letter with a subscript will be the corresponding row of a matrix denoted by the same letter in the upper-case. For example, in the algorithm, we use $\mathbf{x}_i \in \mathbb{R}^p$ for the local copy of the model parameters at node i and $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]^\top$ is the matrix, whose rows are these local copies. Throughout the chapter, without any specification, we use $\langle \cdot, \cdot \rangle$ as standard vector/matrix inner product, whose form is dependent on context. Similarly, $\|\cdot\|_{\mathbf{P}}$ is a vector/matrix norm defined as $\sqrt{\langle \cdot, \mathbf{P}(\cdot) \rangle}$ for a positive semi-definite matrix $\mathbf{P} \in \mathbb{R}^{n\times n}$, with some specific domain. We use $\mathbf{M}^\dagger$ to denote the pseudo-inverse of a matrix $\mathbf{M} \in \mathbb{R}^{n\times n}$. If $\mathbf{M}$ is symmetric, we use $\lambda_{\max}(\mathbf{M})$, $\lambda_i(\mathbf{M})$ and $\lambda_{\min}(\mathbf{M})$ to denote the largest, the ith-largest and the smallest nonzero eigenvalues of $\mathbf{M}$, respectively. For the stochastic approximate of the local gradient, we use $\nabla f_i(\mathbf{x}_i, \xi_i) \in \mathbb{R}^p$ as the estimate of the deterministic gradient $\nabla f_i(\mathbf{x}_i) \in \mathbb{R}^p$ of $\mathbf{x}_i$ at node i. With the collection of all local estimates $\{\nabla f_i(\mathbf{x}_i, \xi_i)\}_{i=1}^n$ for $\{\nabla f_i(\mathbf{x}_i)\}_{i=1}^n$, we define $\nabla \mathbf{F}(\mathbf{X}) \in \mathbb{R}^{n\times p}$ and $\nabla \mathbf{F}(\mathbf{X}, \xi) \in \mathbb{R}^{n\times p}$ as the compact matrix form of them. The commonly used all-zero vector(matrix) and all-one vector are $\mathbf{0} \in \mathbb{R}^p(\mathbb{R}^{n\times p})$ and $\mathbf{1} \in \mathbb{R}^n$ respectively. The proximal operator with parameter $\eta > 0$ of a function $r : \mathbb{R}^p \to \mathbb{R}$ is $\mathbf{prox}_{\eta r} = \arg\min_{\mathbf{z}\in\mathbb{R}^p} r(\mathbf{z}) + \frac{1}{2\eta}\|\mathbf{z} - \mathbf{x}\|^2$. Finally, we use $[n]$ to replace $\{1, \cdots, n\}$ for abbreviation.

4.2 The Proposed Algorithms

An equivalent form of the problem (1.2) reformulating the decentralized constraint via a mixing matrix is provided as follows:

$$\mathbf{X}^* = \underset{\substack{(\mathbf{I}-\mathbf{W})\mathbf{X}=\mathbf{0} \\ \mathbf{X}\in\mathbb{R}^{n\times p}}}{\arg\min} \underbrace{\sum_{i=1}^n f_i(\mathbf{x}_i)}_{=:\mathbf{F}(\mathbf{X})} + \underbrace{\sum_{i=1}^n r(\mathbf{x}_i)}_{=:\mathbf{R}(\mathbf{X})}, \tag{4.1}$$

where $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n]^\top$ is the collection of local $\mathbf{x}_i$s and $\mathbf{W}$ is a symmetric matrix which restricts the feasible region of the above problem to the subspace $\{\mathbf{1}\mathbf{x}^\top \in \mathbb{R}^{n\times p} \mid \forall \mathbf{x} \in \mathbb{R}^p\}$. The optimal solution $\mathbf{X}^* = \mathbf{1}(\mathbf{x}^*)^\top$ is consensual and provides an optimal solution $\mathbf{x}^* \in \mathbb{R}^p$ to the problem (1.2). The detailed assumptions on $\mathbf{W}$ are shown below.

Assumption 4.2.1 (Mixing matrix)**.** *The graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is undirected and connected with $\mathcal{V} = [n] := \{1, 2, \cdots, n\}$. The mixing matrix $\mathbf{W} = [w_{ij}] \in \mathbb{R}^{n\times n}$ is symmetric and satisfies*

1. $w_{ij} = 0$ *if* $i \neq j$, $(i, j) \notin \mathcal{E}$ *and* $w_{ij} > 0$ *if* $(i, j) \in \mathcal{E}$.
2. $-1 < \lambda_n(\mathbf{W}) \leq \cdots \leq \lambda_2(\mathbf{W}) < \lambda_1(\mathbf{W}) = 1$ *and* $\mathbf{W}\mathbf{1} = \mathbf{1}$.

We propose a Proximal gradient LinEAr convergent Decentralized algorithm with compressed communication, Prox-LEAD, to solve the problem (4.1). Our algorithm extends LEAD proposed in [33] to deal with the nonsmooth component via the proximal operator, and to accelerate the stochastic optimization in the finite-sum setting. Moreover, when the compression is absent and the gradient oracle returns full gradient, Prox-LEAD is shown to be covered by the proximal unified decentralized framework in [106]. Algorithm 4.1 illustrates the prototype of Prox-LEAD in the compact form, and it reduces to LEAD by setting $\mathbf{R} = \mathbf{0}$.

In Algorithm 4.1, line 6 uses the gradient from the stochastic gradient oracle to create an auxiliary variable $\mathbf{Z}$ as the information to be communicated. The COMM procedure compresses the difference between $\mathbf{Z}$ and a state variable $\mathbf{H}$ by a unbiased operator satisfying the following condition.

Assumption 4.2.2 (Unbiased compression operator)**.** *The stochastic operator* $\mathcal{Q} : \mathbb{R}^p \to \mathbb{R}^p$ *is unbiased, i.e.,* $\mathbb{E}\mathcal{Q}(\mathbf{x}) = \mathbf{x}$*, and there exists* $C \geq 0$ *such that*

$$\mathbb{E}\|\mathbf{x} - \mathcal{Q}(\mathbf{x})\|^2 \leq C\|\mathbf{x}\|^2, \ \forall \mathbf{x} \in \mathbb{R}^p.$$

In particular, when $C = 0$, *we treat* $\mathcal{Q}$ *as the identity operator.*

The benefit of this difference compression is to reduce the compression error asympototically [17, 33]. Since the following variance of the stochastic estimation is dependent on the distance between $\mathbf{Z}^{k+1}$ and $\mathbf{H}^k$,

$$\mathbb{E}\|\hat{\mathbf{Z}}^{k+1} - \mathbf{Z}^{k+1}\|^2 = \underbrace{\mathbb{E}\|\mathcal{Q}(\mathbf{Z}^{k+1} - \mathbf{H}^k) - (\mathbf{Z}^{k+1} - \mathbf{H}^k)\|^2}_{=\mathcal{O}(\|\mathbf{Z}^{k+1} - \mathbf{H}^k\|)},$$

the variance of the compression will vanish as $\mathbf{Z}$ and $\mathbf{H}$ converge to the same point. The updates in line 8 and line 9 essentially compensate the compression error locally. Note that the COMM procedure is first proposed in the preliminary version of this work [33], and please to refer to Section 3.1 in [33] for a detailed explanation for the reduced compression error and implicit error compensation.

Lines 8 and 9 of Algorithm 4.1 proceed in parallel after the compressed communication is exchanged. The proximal mapping in Line 10 is applied to the rows of $\mathbf{V}^{k+1}$

separately, which is defined as

$$\begin{bmatrix} — & (\mathbf{x}_1^{k+1})^\top & — \\ & \vdots & \\ — & (\mathbf{x}_n^{k+1})^\top & — \end{bmatrix} = \begin{bmatrix} — & \mathbf{prox}_{\eta r}(\mathbf{v}_1^{k+1})^\top & — \\ & \vdots & \\ — & \mathbf{prox}_{\eta r}(\mathbf{v}_n^{k+1})^\top & — \end{bmatrix}.$$

Algorithm 4.1: Prox-LEAD

```
Input: Stepsize $\eta$, parameter $\alpha, \gamma$, initial $\mathbf{X}^0, \mathbf{H}^1, \mathbf{D}^1 = \mathbf{0}$
Output: $\mathbf{X}^K$ or $1/n \sum_{i=1}^n \mathbf{X}_i^K$

$\mathbf{H}_w^1 = \mathbf{W}\mathbf{H}^1$
$\mathbf{Z}^1 = \mathbf{X}^0 - \eta \nabla \mathbf{F}(\mathbf{X}^0, \xi^0)$
$\mathbf{X}^1 = \mathbf{prox}_{\eta \mathbf{R}}(\mathbf{Z}^1)$
for $k = 1, 2, \cdots, K-1$ do
    $\mathbf{G}^k = \mathrm{SGO}(\mathbf{X}^k)$
    $\mathbf{Z}^{k+1} = \mathbf{X}^k - \eta \mathbf{G}^k - \eta \mathbf{D}^k$
    $\hat{\mathbf{Z}}^{k+1}, \hat{\mathbf{Z}}_w^{k+1}, \mathbf{H}^{k+1}, \mathbf{H}_w^{k+1} = \mathrm{COMM}(\mathbf{Z}^{k+1}, \mathbf{H}^k, \mathbf{H}_w^k)$
    $\mathbf{D}^{k+1} = \mathbf{D}^k + \frac{\gamma}{2\eta}(\hat{\mathbf{Z}}^{k+1} - \hat{\mathbf{Z}}_{\mathbf{W}}^{k+1})$
    $\mathbf{V}^{k+1} = \mathbf{Z}^{k+1} - \frac{\gamma}{2}(\hat{\mathbf{Z}}^{k+1} - \hat{\mathbf{Z}}_{\mathbf{W}}^{k+1})$
    $\mathbf{X}^{k+1} = \mathbf{prox}_{\eta \mathbf{R}}\left(\mathbf{V}^{k+1}\right)$
end for
```

Algorithm 4.2: Compressed Communication Procedure (COMM)

```
procedure COMM($\mathbf{Z}^{k+1}, \mathbf{H}^k, \mathbf{H}_w^k$)
    $\mathbf{Q}^k = \mathcal{Q}(\mathbf{Z}^{k+1} - \mathbf{H}^k)$                ▷ Compression
    $\hat{\mathbf{Z}}^{k+1} = \mathbf{H}^k + \mathbf{Q}^k$
    $\hat{\mathbf{Z}}_w^{k+1} = \mathbf{H}_w^k + \mathbf{W}\mathbf{Q}^k$               ▷ Communication
    $\mathbf{H}^{k+1} = (1-\alpha)\mathbf{H}^k + \alpha \hat{\mathbf{Z}}$
    $\mathbf{H}_w^{k+1} = (1-\alpha)\mathbf{H}_w^k + \alpha \hat{\mathbf{Z}}_w$
    Return: $\hat{\mathbf{Z}}^{k+1}, \hat{\mathbf{Z}}_w^{k+1}, \mathbf{H}^{k+1}, \mathbf{H}_w^{k+1}$
end procedure
```

4.2.1 Prox-LEAD as Inexact PUDA

The problem (4.1) can be reformulated into the following three operators splitting

$$\underset{\mathbf{X} \in \mathbb{R}^{n \times p}}{\text{minimize}} \ \mathbf{F}(\mathbf{X}) + \iota_{\mathbf{0}}(\mathbf{B}^{\frac{1}{2}}\mathbf{X}) + \mathbf{R}(\mathbf{X}), \tag{4.2}$$

The general setting	The finite-sum setting	
	Loopless SVRG	SAGA
Sample $\xi_i \sim \mathcal{D}_i$	Sample $l \in [m] \sim \mathcal{P}_i$ randomly	
$\mathbf{g}_i = \nabla f_i(\mathbf{x}_i, \xi_i).$	Sample $\omega \in \{0,1\} \sim$ *Bernoulli* (p), $\mathbf{g}_i = \frac{1}{mp_{il}}(\nabla f_{il}(\mathbf{x}_i) - \nabla f_{il}(\tilde{\mathbf{x}}_i)) + \nabla f_i(\tilde{\mathbf{x}}_i),$ $\tilde{\mathbf{x}}_i = \omega \cdot \mathbf{x}_i + (1-\omega) \cdot \tilde{\mathbf{x}}_i.$	$\mathbf{g}_i = \frac{1}{mp_{il}}(\nabla f_{il}(\mathbf{x}_i) - \nabla f_{il}(\tilde{\mathbf{x}}_{il})) + \frac{1}{m}\sum_{j=1}^{m} \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}),$ $\tilde{\mathbf{x}}_{il} = \mathbf{x}_i.$

Table 4.1: Stochastic gradient oracle (SGO).

where $\iota_{\mathbf{0}}$ is the indicator function taking zero value at $\mathbf{0}$ and infinity elsewhere.

Many existing schemes such as Condat-Vu in [109, 110], PDFP in [111] and PD3O in [112] are applicable to this problem but we choose to adapt the inexact PDHG with a single proximal gradient step and derive the following iteration which is not the realization of any scheme mentioned above.

$$\left\lfloor \begin{aligned} \overline{\mathbf{X}}^{k+1} &= \mathbf{X}^k - \eta \nabla \mathbf{F}(\mathbf{X}^k) - \eta \mathbf{D}^k, \\ \mathbf{D}^{k+1} &= \mathbf{D}^k + \frac{\lambda}{2}(\mathbf{I} - \mathbf{W})\overline{\mathbf{X}}^{k+1}, \\ \mathbf{V}^{k+1} &= \mathbf{X}^k - \eta \nabla \mathbf{F}(\mathbf{X}^k) - \eta \mathbf{D}^{k+1} \\ &= \Big(\mathbf{I} - \frac{\eta\lambda}{2}(\mathbf{I} - \mathbf{W})\Big)\overline{\mathbf{X}}^{k+1}, \\ \mathbf{X}^{k+1} &= \mathbf{prox}_{\eta \mathbf{R}}(\mathbf{V}^{k+1}). \end{aligned} \right. \tag{4.3}$$

Compared to LEAD, the third step is reformulated to fully depend on $\overline{\mathbf{X}}$ but the number of communication is unchanged, and a proximal map is applied on $\mathbf{V}$ directly in the final step. The iteration (4.3) can be shown as a special case of PUDA in [106] which has global linear convergence for strongly convex $\mathbf{F}$.

The benefit of this iteration is to maintain the consensus of $\overline{\mathbf{X}}$ in optimality, which further implies the consensus of $\mathbf{V}$ and $\mathbf{X}$. As shown in Section 4.4, the consensus of $\overline{\mathbf{X}}$ is the key to the linear convergence of Prox-LEAD. It also explains why we need the same nonsmooth function r for all nodes.

If we compress the only communication step involving $\overline{\mathbf{X}}$ via COMM procedure, $(\mathbf{D}, \mathbf{V})$ is updated by the inexact information with the controllable compression error. Therefore, we regard Prox-LEAD as inexact PUDA in terms of the inexact dual and proximal steps.

4.2.2 Stochastic Gradient Oracle

As shown in Algorithm 4.1, Prox-LEAD uses a stochastic gradient oracle (SGO) to estimate the gradient, and different stochastic estimators are listed in Table 4.1 for gradient estimation. The stochastic gradient oracle returns three types of estimation. In the general setting, each node uses sample distribution $\mathcal{D}_i$ to provide an unbiased stochastic gradient and the variance exists. In the finite-sum setting, we assume each node will construct a discrete distribution $\mathcal{P}_i = \{p_{il} : l \in [m]\}$ for m mini-batches and the stochastic gradient will be corrected by two different variance reduction schemes: Loopless SVRG and SAGA.

For Loopless SVRG (LSVRG), each node will have a reference point $\tilde{x}_i$ where the full gradient is evaluated after a random period. A random variable $l \in [m]$ will be sampled first with distribution $\mathcal{P}_i$, then a Bernoulli random variable ω will be sampled to determine whether the reference point will be update or not. If $\omega = 1$, the reference is replaced by the latest $\mathbf{x}_i$, otherwise unchanged. For SAGA, each node will have m reference points, $\{\tilde{x}_{ij} : j \in [m]\}$. After the index l is sampled, $\tilde{\mathbf{x}}_{il}$ will be replaced by the latest $\mathbf{x}_i$ while remaining reference points are unchanged. Both schemes use reference points to correct the gradient by variance reduction. Loopless SVRG is memory-friendly but requires more gradient evaluations. Empirically, SAGA converges faster in terms of the number of gradient evaluations, but it requires more memory space. The following Table 4.2 summarizes the convergence complexity of Prox-LEAD in different settings.

Algorithm	Convergence complexity
Prox-LEAD Theorem 4.4.1	$\widetilde{\mathcal{O}}(C + \kappa_f + \kappa_g + C\kappa_f\kappa_g)$
Prox-LEAD LSVRG Theorem 4.4.2	$\widetilde{\mathcal{O}}(C + \kappa_f + \kappa_g + C\kappa_f\kappa_g + p^{-1})$
Prox-LEAD SAGA Theorem 4.4.3	$\widetilde{\mathcal{O}}(C + \kappa_f + \kappa_g + C\kappa_f\kappa_g + m)$

Table 4.2: Summary of the convergence compleixty for Prox-LEAD to achieve ϵ-accuracy with fiexed stepsizes. The first row is the complexity with the full gradient.

We define condition numbers, κ_f and κ_g, as follows

$$\kappa_f = \frac{L}{\mu}, \;\; \kappa_g = \frac{\lambda_{\max}(\mathbf{I} - \mathbf{W})}{\lambda_{\min}(\mathbf{I} - \mathbf{W})},$$

where L and μ are regularity constants of the objective function assumed in Assumption 4.3.2. When the compression procedure and the regularizer are removed, the complexities of LEAD LSVRG and LEAD SAGA are reduced to $\widetilde{\mathcal{O}}(\kappa_f + \kappa_g + p^{-1})$ and $\widetilde{\mathcal{O}}(\kappa_f + \kappa_g + m)$ respectively, which are better than LessBit-Option D in [81] in terms of the conditional numbers, and it improves over the stochastic LEAD in [33].

4.3 Assumptions on Regularity

In the general stochastic setting, each $f_i(\mathbf{x})$ in (1.2) is the expectation of the local loss function under the sample distribution $\mathcal{D}_i$, and we make the following inter-node assumption.

Assumption 4.3.1 (Locally bounded gradient variance). *In the general stochastic setting, each local stochastic gradient $\nabla f_i(\mathbf{x}, \xi_i)$ is an unbiased estimate, i.e., $\mathbb{E}_{\xi_i} \nabla f_i(\mathbf{x}, \xi_i) = \nabla f_i(x)$, and satisfies*

$$\mathbb{E}\|\nabla f_i(\mathbf{x}^*, \xi_i) - \nabla f_i(\mathbf{x}^*)\|^2 \leq \sigma_i^2,$$

where $\mathbf{x}^*$ is the optimal solution to the problem (1.2).

This locally bounded variance at the optimal point is strictly weaker than the uniformly bounded variance assumption [33].

Given a smooth convex function f, we define the Bregman distance with respect to f as

$$\mathrm{V}_f(\mathbf{x},\mathbf{y}) = f(\mathbf{x}) - f(\mathbf{y}) - \langle \nabla f(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle, \ \ \forall \mathbf{x},\mathbf{y} \in \mathbb{R}^p.$$

With the above definition, we impose different assumptions to the regularity of smooth function component in two settings.

Assumption 4.3.2 (Strong convexity and smoothness)**.** *Each f_i is a smooth, μ-strongly convex function, i.e., $\forall \mathbf{x},\mathbf{y} \in \mathbb{R}^p$,*

$$\underbrace{f_i(\mathbf{x}) - f_i(\mathbf{y}) - \langle \nabla f_i(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle}_{=\mathrm{V}_{f_i}(\mathbf{x},\mathbf{y})} \geq \frac{\mu}{2}\|\mathbf{x}-\mathbf{y}\|^2. \tag{4.4}$$

In the general stochastic setting, each f_i is L-smooth in expectation, i.e., $\forall \mathbf{x},\mathbf{y} \in \mathbb{R}^p$,

$$\mathbb{E}\|\nabla f_i(\mathbf{x},\xi_i) - \nabla f_i(\mathbf{y},\xi_i)\|^2 \leq \underbrace{2L[f_i(\mathbf{x}) - f_i(\mathbf{y}) - \langle \nabla f_i(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle]}_{=2L\mathrm{V}_{f_i}(\mathbf{x},\mathbf{y})}. \tag{4.5}$$

In the finite-sum setting, the L-smoothness is imposed on each f_{ij} instead, i.e., $\forall \mathbf{x},\mathbf{y} \in \mathbb{R}^p$,

$$\|\nabla f_{ij}(\mathbf{x}) - \nabla f_{ij}(\mathbf{y})\|^2 \leq \underbrace{2L[f_{ij}(\mathbf{x}) - f_{ij}(\mathbf{y}) - \langle \nabla f_{ij}(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle]}_{=2L\mathrm{V}_{f_{ij}}(\mathbf{x},\mathbf{y})}. \tag{4.6}$$

Note the inequalities in (4.4) and (4.5) are equivalent to, $\forall \mathbf{X},\mathbf{Y} \in \mathbb{R}^{n\times p}$,

$$\begin{aligned}\mathrm{V}_{\mathbf{F}}(\mathbf{X},\mathbf{Y}) = \mathbf{F}(\mathbf{X}) - \mathbf{F}(\mathbf{Y}) - \langle \nabla \mathbf{F}(\mathbf{Y}), \mathbf{X}-\mathbf{Y}\rangle \geq \frac{\mu}{2}\|\mathbf{X}-\mathbf{Y}\|^2,\\ \mathbb{E}\|\nabla \mathbf{F}(\mathbf{X},\xi) - \nabla \mathbf{F}(\mathbf{Y},\xi)\|^2 \leq 2L\sum_{i=1}^n \mathrm{V}_{f_i}(\mathbf{x}_i,\mathbf{y}_i) = 2L\mathrm{V}_{\mathbf{F}}(\mathbf{X},\mathbf{Y}).\end{aligned}$$

Remark 4.3.1. *In [76], the above two types of L-smoothness assumptions are discussed, and the latter is used to derive an improved convergence rate of SGD. The expected L-smoothness is shown to be significantly weaker than the most commonly-used L-smoothness of $f_i(\mathbf{x},\xi_i)$ for all ξ_i.*

4.4 Convergence Analysis

In this section, we present the convergence of Prox-LEAD in both stochastic scenarios. We first show two fundamental lemmas regarding the conditional expectation on the compression operator, then we present the two scenarios in Sections 4.4.1 and 4.4.2, respectively. More specifically, in Section 4.4.1, the linear convergence to the neighborhood will be shown under the general stochastic setting, while in Section 4.4.2, two variance reduction schemes are used to exploit the exact linear convergence in the problems with finite-sum structure.

The stochastic actions such as compression and gradient estimation generate two sequences of σ-algebra where the stochastic variables in this procedure are adapted. We use $\mathcal{F}^k$ to denote the σ-algebra of gradient estimation at kth step and $\mathcal{H}^k$ is the σ-algebra of stochastic compression at the same step. $\{\mathcal{F}^k\}$ and $\{\mathcal{H}^k\}$ satisfy

$$\mathcal{F}^1 \subset \mathcal{H}^1 \subset \mathcal{F}^2 \subset \mathcal{H}^2 \subset \cdots \subset \mathcal{F}^k \subset \mathcal{H}^k \subset \cdots .$$

With these notations, we can clarify the stochastic dependencies among the variables generated by the algorithm. For example, tuple $(\mathbf{G}^k, \mathbf{Z}^{k+1})$ is measurable in $\mathcal{F}^k$ and tuple $(\hat{\mathbf{Z}}^{k+1}, \mathbf{H}^{k+1}, \mathbf{D}^{k+1}, \mathbf{V}^{k+1}, \mathbf{X}^{k+1})$ is measurable in $\mathcal{H}^k$.

Throughout the section, we use $\mathbb{E}$ to denote the conditional expectations $\mathbb{E}_{\mathcal{F}^k}$ and $\mathbb{E}_{\mathcal{H}^k}$ given the context for simplicity. Then we define some auxiliary constants related to the optimal solution $\mathbf{X}^*$ to the problem (4.1). $\mathbf{X}^*$ is consensual, i.e., each row $\mathbf{x}^*$ solves the problem (1.2). We let $\mathbf{z}^* = \mathbf{x}^* - (\eta/n)\sum_{i=1}^n \nabla f_i(\mathbf{x}^*)$, which is in the pre-image of $\mathbf{prox}_{\eta r}$ at $\mathbf{x}^*$. In addition, we let

$$\mathbf{Z}^* = \mathbf{1}(\mathbf{z}^*)^\top = \mathbf{X}^* - \frac{\eta}{n}\mathbf{1}\mathbf{1}^\top \nabla \mathbf{F}(\mathbf{X}^*), \tag{4.7}$$

$$\begin{aligned}\mathbf{D}^* &= \frac{1}{\eta}(\mathbf{X}^* - \mathbf{Z}^*) - \nabla \mathbf{F}(\mathbf{X}^*) \\ &= -\left(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top\right)\nabla \mathbf{F}(\mathbf{X}^*).\end{aligned} \tag{4.8}$$

To show the convergence of the proposed algorithm, we characterizes the decrease of $\|\mathbf{Z}^k - \mathbf{Z}^*\|$, $\|\mathbf{D}^k - \mathbf{D}^*\|_{(\mathbf{I}-\mathbf{W})^\dagger}$, $\|\mathbf{X}^k - \mathbf{X}^*\|$, and $\|\mathbf{H}^k - \mathbf{Z}^*\|$. The convergence of $\mathbf{H}^k$ to $\mathbf{Z}^*$ shows the decrease of the compression error to zero. The following lemma shows the connection of those values for the k and $k+1$-th iteration.

Lemma 4.4.1 (One-step progress). *Let $\{(\mathbf{Z}^k, \mathbf{D}^k, \mathbf{X}^k)\}$ be the sequence generated from Algorithm 4.1. Under Assumption 4.2.1, taking the expectation conditioned on the stochastic compression operator at the k-th iteration, we have for any $\gamma \le \frac{2}{\lambda_{\max}(\mathbf{I}-\mathbf{W})}$,*

$$\begin{aligned}\|\mathbf{Z}^{k+1} - \mathbf{Z}^*\|^2 = {} & \|\mathbf{X}^k - \mathbf{X}^* - \eta\mathbf{G}^k + \eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2 + \eta^2\|\mathbf{D}^k - \mathbf{D}^*\|^2 \\ & - 2\eta\langle\mathbf{D}^k - \mathbf{D}^*, \mathbf{X}^k - \mathbf{X}^* - \eta\mathbf{G}^k + \eta\nabla\mathbf{F}(\mathbf{X}^*\rangle, \end{aligned} \tag{4.9}$$

$$\begin{aligned}&\mathbb{E}\|\mathbf{D}^{k+1} - \mathbf{D}^*\|^2_{(\mathbf{I}-\mathbf{W})^\dagger} \\ = {} & \|\mathbf{D}^k - \mathbf{D}^*\|^2_{(\mathbf{I}-\mathbf{W})^\dagger} - \gamma\|\mathbf{D}^k - \mathbf{D}^*\|^2 + \frac{\gamma^2}{4\eta^2}\|\mathbf{Z}^{k+1} - \mathbf{Z}^*\|^2_{\mathbf{I}-\mathbf{W}} + \frac{\gamma^2}{4\eta^2}\mathbb{E}\|\hat{\mathbf{Z}}^{k+1} - \mathbf{Z}^{k+1}\|^2_{\mathbf{I}-\mathbf{W}} \\ & + \frac{\gamma}{\eta}\langle\mathbf{D}^k - \mathbf{D}^*, \mathbf{X}^k - \mathbf{X}^* - \eta\mathbf{G}^k + \eta\nabla\mathbf{F}(\mathbf{X}^*)\rangle, \end{aligned} \tag{4.10}$$

and

$$\mathbb{E}\|\mathbf{X}^{k+1} - \mathbf{X}^*\|^2 \le \|\mathbf{Z}^{k+1} - \mathbf{Z}^*\|^2_{\mathbf{I}-\frac{\gamma}{2}(\mathbf{I}-\mathbf{W})} + \frac{\gamma^2}{4}\mathbb{E}\|\hat{\mathbf{Z}}^{k+1} - \mathbf{Z}^{k+1}\|^2_{(\mathbf{I}-\mathbf{W})^2}. \tag{4.11}$$

Proof. (i) The equality (4.9) is shown from Line 6 of Algorithm 4.1 and (4.8) directly. (ii) Note that $(\mathbf{I}-\mathbf{W})\mathbf{Z}^* = \mathbf{0}$. Then from Line 8 of Algorithm 4.1, we have

$$\mathbf{D}^{k+1} - \mathbf{D}^* = \mathbf{D}^k - \mathbf{D}^* + \frac{\gamma}{2\eta}(\mathbf{I}-\mathbf{W})(\hat{\mathbf{Z}}^{k+1} - \mathbf{Z}^*),$$

which gives

$$\begin{aligned}&\mathbb{E}\|\mathbf{D}^{k+1} - \mathbf{D}^*\|^2_{(\mathbf{I}-\mathbf{W})^\dagger} \\ = {} & \|\mathbf{D}^k - \mathbf{D}^*\|^2_{(\mathbf{I}-\mathbf{W})^\dagger} + \frac{\gamma}{\eta}\langle\mathbf{D}^k - \mathbf{D}^*, \mathbf{Z}^{k+1} - \mathbf{Z}^*\rangle + \frac{\gamma^2}{4\eta^2}\|\mathbf{Z}^{k+1} - \mathbf{Z}^*\|^2_{\mathbf{I}-\mathbf{W}} \\ & + \frac{\gamma^2}{4\eta^2}\mathbb{E}\|\hat{\mathbf{Z}}^{k+1} - \mathbf{Z}^{k+1}\|^2_{\mathbf{I}-\mathbf{W}} \\ = {} & \|\mathbf{D}^k - \mathbf{D}^*\|^2_{(\mathbf{I}-\mathbf{W})^\dagger} - \gamma\|\mathbf{D}^k - \mathbf{D}^*\|^2 + \frac{\gamma}{\eta}\langle\mathbf{D}^k - \mathbf{D}^*, \mathbf{X}^k - \mathbf{X}^* - \eta\mathbf{G}^k + \eta\nabla\mathbf{F}(\mathbf{X}^*)\rangle \\ & + \frac{\gamma^2}{4\eta^2}\|\mathbf{Z}^{k+1} - \mathbf{Z}^*\|^2_{\mathbf{I}-\mathbf{W}} + \frac{\gamma^2}{4\eta^2}\mathbb{E}\|\hat{\mathbf{Z}}^{k+1} - \mathbf{Z}^{k+1}\|^2_{\mathbf{I}-\mathbf{W}}, \end{aligned}$$

where the second equality comes from Line 6 of Algorithm 4.1.

(iii) From the definition of $\mathbf{Z}^*$, we have

$$\mathbf{X}^* = \mathbf{prox}_{\eta\mathbf{R}}(\mathbf{Z}^*).$$

Therefore,

$$\begin{aligned}\mathbb{E}\|\mathbf{X}^{k+1}-\mathbf{X}^*\|^2 &= \|\mathbf{prox}_{\eta\mathbf{R}}(\mathbf{V}^{k+1})-\mathbf{prox}_{\eta\mathbf{R}}(\mathbf{Z}^*)\|^2\\ &\le \mathbb{E}\|\mathbf{V}^{k+1}-\mathbf{Z}^*\|^2\\ &= \mathbb{E}\Big\|\mathbf{Z}^{k+1}-\mathbf{Z}^*-\Big(\frac{\gamma}{2}(\mathbf{I}-\mathbf{W})(\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^*)\Big)\Big\|^2\\ &= \Big\|\Big(\mathbf{I}-\frac{\gamma}{2}(\mathbf{I}-\mathbf{W})\Big)(\mathbf{Z}^{k+1}-\mathbf{Z}^*)\Big\|^2+\frac{\gamma^2}{4}\mathbb{E}\|(\mathbf{I}-\mathbf{W})\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|^2\\ &\le \|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2_{\mathbf{I}-\frac{\gamma}{2}(\mathbf{I}-\mathbf{W})}+\frac{\gamma^2}{4}\mathbb{E}\|(\mathbf{I}-\mathbf{W})\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|^2,\end{aligned}$$

where the first inequality comes from the non-expansiveness of $\mathbf{prox}_{\eta\mathbf{R}}$ and the last inequality follows from

$$\begin{aligned}\Big(\mathbf{I}-\frac{\gamma}{2}(\mathbf{I}-\mathbf{W})\Big)^2 &= \mathbf{I}-\gamma(\mathbf{I}-\mathbf{W})+\frac{\gamma^2}{4}(\mathbf{I}-\mathbf{W})^2\\ &= \mathbf{I}-\frac{\gamma}{2}(\mathbf{I}-\mathbf{W})-\frac{\gamma}{2}(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\Big(\mathbf{I}-\frac{\gamma}{2}(\mathbf{I}-\mathbf{W})\Big)(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\\ &\preccurlyeq \mathbf{I}-\frac{\gamma}{2}(\mathbf{I}-\mathbf{W}),\end{aligned}$$

since $\frac{\gamma}{2}(\mathbf{I}-\mathbf{W})\preccurlyeq\mathbf{I}$. The inequality (4.11) is obtained. □

The next lemma builds the critical inequality for Algorithm 4.1 and serves as the key step in the proofs. We define the following two positive (semi)definite matrices

$$\mathbf{P}:=\mathbf{I}-\frac{\gamma}{2}(\mathbf{I}-\mathbf{W}),\ \ \mathbf{Q}:=(\mathbf{I}-\mathbf{W})^\dagger$$

to measure the convergence.

Lemma 4.4.2 (The key inequality)**.** *Under Assumptions 4.2.1 and 4.2.2, we choose* $\alpha\in(0,(1+C)^{-1})$ *such that* $\Delta(\alpha):=\alpha-(1+C)\alpha^2>0$ *and*

$$\eta\in\Big(0,\frac{2}{\mu}\Big),\ \ \gamma\in\Big(0,\frac{1}{\lambda_{\max}(\mathbf{I}-\mathbf{W})}\frac{2\Delta(\alpha)\eta\mu}{\Delta(\alpha)\eta\mu+4\alpha C}\Big).$$

Then the sequence $\{(\hat{\mathbf{Z}}^k, \mathbf{Z}^k, \mathbf{D}^k, \mathbf{H}^k)\}$ *generated by Algorithm 4.1 satisfies*

$$\begin{aligned}
&\Big(1-\frac{\eta\mu}{2}\Big)\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2_{\mathbf{P}}+\frac{2\eta^2}{\gamma}\mathbb{E}\|\mathbf{D}^{k+1}-\mathbf{D}^*\|^2_{\mathbf{Q}}\\
&+CM\mathbb{E}\|\mathbf{H}^{k+1}-\mathbf{Z}^*\|^2+\mathbb{E}\|\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|^2_{\mathbf{I}-\mathbf{P}}\\
\le{}&\|\mathbf{X}^k-\mathbf{X}^*-\eta\mathbf{G}^k+\eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2+\frac{2\eta^2}{\gamma}\|\mathbf{D}^k-\mathbf{D}^*\|^2_{\mathbf{Q}-\frac{\gamma}{2}\mathbf{I}}\\
&+(1-\alpha)CM\|\mathbf{H}^k-\mathbf{Z}^*\|^2,
\end{aligned}\tag{4.12}$$

where the expectation is conditioned on the stochastic compression at k*-th step and*

$$M := \frac{(1-\frac{\gamma}{2}\lambda_{\max}(\mathbf{I}-\mathbf{W}))\eta\mu}{2\alpha C} > 0.$$

In particular, when $C=0$*, i.e.,* $\hat{\mathbf{Z}}^k=\mathbf{Z}^k$, (4.12) *still holds as*

$$\begin{aligned}
&\Big(1-\frac{\eta\mu}{2}\Big)\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2_{\mathbf{P}}+\frac{2\eta^2}{\gamma}\mathbb{E}\|\mathbf{D}^{k+1}-\mathbf{D}^*\|^2_{\mathbf{Q}}+CM\mathbb{E}\|\mathbf{H}^{k+1}-\mathbf{Z}^*\|^2\\
\le{}&\|\mathbf{X}^k-\mathbf{X}^*-\eta\mathbf{G}^k+\eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2+\frac{2\eta^2}{\gamma}\|\mathbf{D}^k-\mathbf{D}^*\|^2_{\mathbf{Q}-\frac{\gamma}{2}\mathbf{I}}+(1-\alpha)CM\|\mathbf{H}^k-\mathbf{Z}^*\|^2
\end{aligned}\tag{4.13}$$

with

$$CM := \frac{(1-\frac{\gamma}{2}\lambda_{\max}(\mathbf{I}-\mathbf{W}))\eta\mu}{2\alpha} > 0.$$

Proof. Letting (4.9) + (4.10)$\times\frac{2\eta^2}{\gamma}$, we get

$$\begin{aligned}
&\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2_{\mathbf{P}}+\frac{2\eta^2}{\gamma}\mathbb{E}\|\mathbf{D}^{k+1}-\mathbf{D}^*\|^2_{\mathbf{Q}}\\
={}&\|\mathbf{X}^k-\mathbf{X}^*-\eta\mathbf{G}^k+\eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2+\frac{2\eta^2}{\gamma}\|\mathbf{D}^k-\mathbf{D}^*\|^2_{\mathbf{Q}}-\eta^2\|\mathbf{D}^k-\mathbf{D}^*\|^2\\
&+\frac{\gamma}{2}\mathbb{E}\|\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|^2_{\mathbf{I}-\mathbf{W}}\\
={}&\|\mathbf{X}^k-\mathbf{X}^*-\eta\mathbf{G}^k+\eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2+\frac{2\eta^2}{\gamma}\|\mathbf{D}^k-\mathbf{D}^*\|^2_{\mathbf{Q}-\frac{\gamma}{2}\mathbf{I}}+\mathbb{E}\|\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|^2_{\mathbf{I}-\mathbf{P}},
\end{aligned}\tag{4.14}$$

From the update of $\mathbf{H}$ in Line 5 of the compression procedure 'COMM', we obtain

$$\begin{aligned}
&\mathbb{E}\|\mathbf{H}^{k+1}-\mathbf{Z}^*\|^2\\
=&\|(1-\alpha)(\mathbf{H}^k-\mathbf{Z}^*)+\alpha(\mathbf{Z}^{k+1}-\mathbf{Z}^*)\|^2+\alpha^2\mathbb{E}\|\mathbf{Q}^k-\mathbb{E}\mathbf{Q}^k\|^2\\
=&(1-\alpha)\|\mathbf{H}^k-\mathbf{Z}^*\|^2+\alpha\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2-\alpha(1-\alpha)\|\mathbf{Z}^{k+1}-\mathbf{H}^k\|^2+\alpha^2\mathbb{E}\|\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|^2\\
\leq&(1-\alpha)\|\mathbf{H}^k-\mathbf{Z}^*\|^2+\alpha\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2-\alpha(1-\alpha)\|\mathbf{Z}^{k+1}-\mathbf{H}^k\|^2+\alpha^2C\mathbb{E}\|\mathbf{Z}^{k+1}-\mathbf{H}^k\|^2\\
=&(1-\alpha)\|\mathbf{H}^k-\mathbf{Z}^*\|^2+\alpha\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2-\Delta(\alpha)\|\mathbf{Z}^{k+1}-\mathbf{H}^k\|^2
\end{aligned}$$

where the second equality uses

$$\|(1-\alpha)\mathbf{x}+\alpha\mathbf{y}\|^2=(1-\alpha)\|\mathbf{x}\|^2+\alpha\|\mathbf{y}\|^2-\alpha(1-\alpha)\|\mathbf{x}-\mathbf{y}\|^2,$$

and the inequality is from Assumption 4.2.2.

Multiplying the $\mathbf{H}$-inequality by C and adding it to the following inequality

$$\Delta(\alpha)\mathbb{E}\|\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|^2\leq\Delta(\alpha)C\|\mathbf{Z}^{k+1}-\mathbf{H}^k\|^2,$$

we have

$$\begin{aligned}
&\Delta(\alpha)\mathbb{E}\|\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|^2+C\mathbb{E}\|\mathbf{H}^{k+1}-\mathbf{Z}^*\|^2\\
\leq&(1-\alpha)C\|\mathbf{H}^k-\mathbf{Z}^*\|^2+\alpha C\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2\\
\leq&(1-\alpha)C\|\mathbf{H}^k-\mathbf{Z}^*\|^2+\frac{\alpha C}{\lambda_{\min}(\mathbf{P})}\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2_{\mathbf{P}},
\end{aligned}\tag{4.15}$$

if $\gamma<\frac{2}{\lambda_{\max}(\mathbf{I}-\mathbf{W})}$ since $\mathbf{P}\succ\mathbf{0}$ and $\lambda_{\max}(\mathbf{P}^{-1})=\lambda_{\min}^{-1}(\mathbf{P})$.

When $C>0$, let $M=\frac{\lambda_{\min}(\mathbf{P})\eta\mu}{2\alpha C}=\frac{(1-\frac{\gamma}{2}\lambda_{\max}(\mathbf{I}-\mathbf{W}))\eta\mu}{2\alpha C}>0$, then (4.14) + $M\times$ (4.15) gives

$$\begin{aligned}
&\Big(1-\frac{\eta\mu}{2}\Big)\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|^2_{\mathbf{P}}+\frac{2\eta^2}{\gamma}\mathbb{E}\|\mathbf{D}^{k+1}-\mathbf{D}^*\|^2_{\mathbf{Q}}\\
&+CM\mathbb{E}\|\mathbf{H}^{k+1}-\mathbf{Z}^*\|^2+\mathbb{E}\|\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|^2_{(\Delta(\alpha)M-1)\mathbf{I}+\mathbf{P}}\\
\leq&\|\mathbf{X}^k-\mathbf{X}^*-\eta\mathbf{G}^k+\eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2+\frac{2\eta^2}{\gamma}\|\mathbf{D}^k-\mathbf{D}^*\|^2_{\mathbf{Q}-\frac{\gamma}{2}\mathbf{I}}\\
&+(1-\alpha)CM\|\mathbf{H}^k-\mathbf{Z}^*\|^2.
\end{aligned}$$

Notice that for the given range of γ, we have

$$\Delta(\alpha)M\mathbf{I}\succcurlyeq 2(\mathbf{I}-\mathbf{P}),$$

hence (4.12) is proved.

Lastly, when $C = 0$, we have $\hat{\mathbf{Z}}^{k+1} = \mathbf{Z}^{k+1}$ for all k. Multiplying the $\mathbf{H}$-inequality by $CM := \frac{(1-\frac{\gamma}{2}\lambda_{\max}(\mathbf{I}-\mathbf{W}))\eta\mu}{2\alpha}$ and adding it to (4.14), we complete the proof. □

4.4.1 The General Stochastic Setting

In the general stochastic setting with $f_i(\mathbf{x}_i) = \mathbb{E}_{\xi_i\sim\mathcal{D}_i} f_i(\mathbf{x}_i, \xi_i)$, the assumptions on f_i and the variance of the stochastic gradient at the optimal point allow us to show the linear convergence of Algorithm 4.1 up to a neighborhood of the optimal point.

Theorem 4.4.1 (Prox-LEAD). *Under Assumptions 4.2.1–4.3.2, let $\{(\mathbf{X}^k, \mathbf{D}^k, \mathbf{Z}^k, \hat{\mathbf{Z}}^k, \mathbf{H}^k)\}$ be the sequence generated by Algorithm 4.1 and $\sigma^2 = \frac{1}{n}\sum_{i=1}^n \sigma_i^2$. Set $\alpha \in (0, (1+C)^{-1})$ such that $\Delta(\alpha) = \alpha - (1+C)\alpha^2 > 0$ and we can choose*

$$\eta \in \Big(0, \frac{1}{2L}\Big], \quad \gamma \in \Big(0, \frac{1}{\lambda_{\max}(\mathbf{I}-\mathbf{W})}\frac{2\Delta(\alpha)\eta\mu}{\Delta(\alpha)\eta\mu + 4\alpha C}\Big),$$

then, in total expectation,

$$\frac{1}{n}\mathbb{E}\Phi^{k+1} \le \rho^k \frac{1}{n}\mathbb{E}\Phi^1 + \frac{2\eta^2\sigma^2}{1-\rho},$$

where

$$\Phi^k := \Big(1 - \frac{\eta\mu}{2}\Big)\|\mathbf{Z}^k - \mathbf{Z}^*\|_{\mathbf{P}}^2 + \frac{2\eta^2}{\gamma}\|\mathbf{D}^k - \mathbf{D}^*\|_{\mathbf{Q}}^2 + CM\|\mathbf{H}^k - \mathbf{Z}^*\|^2 + \|\hat{\mathbf{Z}}^k - \mathbf{Z}^k\|_{\mathbf{I}-\mathbf{P}}^2$$

and

$$\rho = \max\Big\{1 - \frac{\eta\mu}{2-\eta\mu}, 1 - \frac{\gamma}{2}\lambda_{\min}(\mathbf{I}-\mathbf{W}), 1-\alpha, (1-\eta\mu)\frac{\gamma\lambda_{\max}(\mathbf{I}-\mathbf{W})}{2}\Big\} < 1.$$

Proof. In Lemma 4.4.2, we derive the one-step progress inequality in expectation conditioned on the stochastic compression at kth step and we now focus on the term involving $\mathbf{X}$ and $\mathbf{G}^k$ in (4.12).

Take the conditional expectation on stochastic gradient at kth step, we have

$$
\begin{aligned}
&\mathbb{E}\|\mathbf{X}^k - \mathbf{X}^* - \eta\mathbf{G}^k + \eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2 \\
=& \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\langle\mathbf{X}^k - \mathbf{X}^*, \nabla\mathbf{F}(\mathbf{X}^k) - \nabla\mathbf{F}(\mathbf{X}^*)\rangle + \eta^2\mathbb{E}\|\nabla\mathbf{F}(\mathbf{X}^k, \xi^k) - \nabla\mathbf{F}(\mathbf{X}^*)\|^2 \\
\leq& \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\langle\mathbf{X}^k - \mathbf{X}^*, \nabla\mathbf{F}(\mathbf{X}^k) - \nabla\mathbf{F}(\mathbf{X}^*)\rangle + 2\eta^2\mathbb{E}\|\nabla\mathbf{F}(\mathbf{X}^k, \xi^k) - \nabla\mathbf{F}(\mathbf{X}^*, \xi^k)\|^2 \\
&+ 2\eta^2\mathbb{E}\|\nabla\mathbf{F}(\mathbf{X}^*, \xi^k) - \nabla\mathbf{F}(\mathbf{X}^*)\|^2 \\
\leq& \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\langle\mathbf{X}^k - \mathbf{X}^*, \nabla\mathbf{F}(\mathbf{X}^k) - \nabla\mathbf{F}(\mathbf{X}^*)\rangle + 4\eta^2 L\mathrm{V}_{\mathbf{F}}(\mathbf{X}^k, \mathbf{X}^*) + 2n\eta^2\sigma^2 \\
=& \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\langle\mathbf{X}^k - \mathbf{X}^*, \nabla\mathbf{F}(\mathbf{X}^k)\rangle + 2\eta(\mathbf{F}(\mathbf{X}^k) - \mathbf{F}(\mathbf{X}^*) - \mathrm{V}_{\mathbf{F}}(\mathbf{X}^k, \mathbf{X}^*)) \\
&+ 4\eta^2 L\mathrm{V}_{\mathbf{F}}(\mathbf{X}^k, \mathbf{X}^*) + 2n\eta^2\sigma^2 \\
=& \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta(\mathbf{F}(\mathbf{X}^*) - \mathbf{F}(\mathbf{X}^k) - \langle\mathbf{X}^* - \mathbf{X}^k, \nabla\mathbf{F}(\mathbf{X}^k)\rangle) \\
&- 2\eta(1 - 2\eta L)\mathrm{V}_{\mathbf{F}}(\mathbf{X}^k, \mathbf{X}^*) + 2n\eta^2\sigma^2 \\
\leq& (1 - \eta\mu)\|\mathbf{X}^k - \mathbf{X}^*\|^2 + 2n\eta^2\sigma^2,
\end{aligned}
$$

where the first equality uses the unbiasedness of stochastic gradient, the second inequality follows the expected Lipschitz property in Assumption 4.3.2, and the last inequality is due to the strong convexity and $\eta \leq \frac{1}{2L}$.

Now we use the power property to take the conditional expectation on the stochastic gradient for (4.12) and plug the above inequality into it, then

$$
\begin{aligned}
&\Big(1 - \frac{\eta\mu}{2}\Big)\mathbb{E}\|\mathbf{Z}^{k+1} - \mathbf{Z}^*\|^2_{\mathbf{P}} + \frac{2\eta^2}{\gamma}\mathbb{E}\|\mathbf{D}^{k+1} - \mathbf{D}^*\|^2_{\mathbf{Q}} \\
&\quad + CM\mathbb{E}\|\mathbf{H}^{k+1} - \mathbf{Z}^*\|^2 + \mathbb{E}\|\hat{\mathbf{Z}}^{k+1} - \mathbf{Z}^{k+1}\|^2_{\mathbf{I}-\mathbf{P}} \\
\leq& (1 - \eta\mu)\|\mathbf{X}^k - \mathbf{X}^*\|^2 + \frac{2\eta^2}{\gamma}\|\mathbf{D}^k - \mathbf{D}^*\|^2_{\mathbf{Q}-\frac{\gamma}{2}\mathbf{I}} \\
&\quad + (1 - \alpha)CM\|\mathbf{H}^k - \mathbf{Z}^*\|^2 + 2n\eta^2\sigma^2.
\end{aligned}
$$

Taking the total expectation and using (4.11), we get

$$\begin{aligned}&\Big(1-\frac{\eta\mu}{2}\Big)\mathbb{E}\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|_{\mathbf{P}}^2+\frac{2\eta^2}{\gamma}\mathbb{E}\|\mathbf{D}^{k+1}-\mathbf{D}^*\|_{\mathbf{Q}}^2\\&+CM\mathbb{E}\|\mathbf{H}^{k+1}-\mathbf{Z}^*\|^2+\mathbb{E}\|\hat{\mathbf{Z}}^{k+1}-\mathbf{Z}^{k+1}\|_{\mathbf{I}-\mathbf{P}}^2\\ \le\ &(1-\eta\mu)\mathbb{E}\|\mathbf{Z}^k-\mathbf{Z}^*\|_{\mathbf{P}}^2+\frac{2\eta^2}{\gamma}\mathbb{E}\|\mathbf{D}^k-\mathbf{D}^*\|_{\mathbf{Q}-\frac{\gamma}{2}\mathbf{I}}^2\\&+(1-\alpha)CM\mathbb{E}\|\mathbf{H}^k-\mathbf{Z}^*\|^2+(1-\eta\mu)\mathbb{E}\|\hat{\mathbf{Z}}^k-\mathbf{Z}^k\|_{(\mathbf{I}-\mathbf{P})^2}^2\\&+2n\eta^2\sigma^2. \qquad (4.16)\end{aligned}$$

Let Φ^k be defined as above, by (4.16), we have

$$\mathbb{E}\Phi^{k+1}\le\max\Big\{1-\frac{\eta\mu}{2-\eta\mu},1-\frac{\gamma}{2}\lambda_{\min}(\mathbf{I}-\mathbf{W}),1-\alpha,\\ (1-\eta\mu)\frac{\gamma\lambda_{\max}(\mathbf{I}-\mathbf{W})}{2}\Big\}\mathbb{E}\Phi^k+2n\eta^2\sigma^2.$$

Finally, by taking the telescopic sum, we complete the proof. □

When there is no compression, Prox-LEAD is reduced to the special case of PUDA with the stochastic gradient. Corollary 4.4.1 shows the linear convergence to the neighborhood of the optimal solution and when the gradient is deterministic, the convergence rate matches that given in [106].

Corollary 4.4.1 (Stochastic PUDA)**.** *When there is no compression, i.e., $C=0$, under Assumptions 4.2.1, 4.3.1 and 4.3.2, we can pick $\alpha=1$ and $\gamma=1$. Then, for any $\eta\in\big(0,\frac{1}{2L}\big]$*

$$\mathbb{E}\Phi^{k+1}\le\max\Big\{1-\frac{\eta\mu}{2-\eta\mu},1-\frac{\lambda_{\min}(\mathbf{I}-\mathbf{W})}{2}\Big\}\mathbb{E}\Phi^k+2n\eta^2\sigma^2,$$

with

$$\Phi^k:=\Big(1-\frac{\eta\mu}{2}\Big)\|\mathbf{Z}^k-\mathbf{Z}^*\|_{\frac{\mathbf{I}+\mathbf{W}}{2}}^2+2\eta^2\|\mathbf{D}^k-\mathbf{D}^*\|_{\mathbf{Q}}^2+\Big(1-\frac{\lambda_{\max}(\mathbf{I}-\mathbf{W})}{2}\Big)\frac{\eta\mu}{2}\|\mathbf{Z}^k-\mathbf{Z}^*\|^2.$$

Proof. Similar to the proof of Theorem 4.4.1, when $C=0$, we considers inequality (4.13) instead. Notice that in this case (4.11) becomes

$$\|\mathbf{X}^{k+1}-\mathbf{X}^*\|^2\le\|\mathbf{Z}^{k+1}-\mathbf{Z}^*\|_{\mathbf{P}}^2.$$

Hence we can get

$$\begin{aligned}
&\Big(1 - \frac{\eta\mu}{2}\Big)\mathbb{E}\|\mathbf{Z}^{k+1} - \mathbf{Z}^*\|^2_{\mathbf{P}} + \frac{2\eta^2}{\gamma}\mathbb{E}\|\mathbf{D}^{k+1} - \mathbf{D}^*\|^2_{\mathbf{Q}} \\
&+ CM\mathbb{E}\|\mathbf{H}^{k+1} - \mathbf{Z}^*\|^2 \\
\leq\ & (1 - \eta\mu)\mathbb{E}\|\mathbf{Z}^k - \mathbf{Z}^*\|^2_{\mathbf{P}} + \frac{2\eta^2}{\gamma}\mathbb{E}\|\mathbf{D}^k - \mathbf{D}^*\|^2_{\mathbf{Q}-\frac{\gamma}{2}\mathbf{I}} \\
&+ (1 - \alpha)CM\mathbb{E}\|\mathbf{H}^k - \mathbf{Z}^*\|^2 + 2n\eta^2\sigma^2,
\end{aligned}$$

where $CM = \frac{(1-\frac{\gamma}{2}\lambda_{\max}(\mathbf{I}-\mathbf{W}))\eta\mu}{2\alpha}$.

We can take $\alpha = 1 - \epsilon$ for some $\epsilon \in (0, 1)$. Note that $\Delta(\alpha)$ approaches $1 - \epsilon$ as C goes to 0. The upper bound of γ is reduced to $\frac{2}{\lambda_{\max}(\mathbf{I}-\mathbf{W})}$, which is strictly greater than 1 due to Assumption 4.2.1. Hence we can take $\gamma = 1$ and the convergence rate becomes

$$\max\left\{1 - \frac{\eta\mu}{2 - \eta\mu}, 1 - \frac{\lambda_{\min}(\mathbf{I} - \mathbf{W})}{2}, \epsilon\right\}.$$

Let ϵ approach 0 and notice that $\mathbf{H}^k = \mathbf{Z}^k$ for all k if $\alpha = 1$, then we complete the proof. □

Corollary 4.4.1 can be viewed as the limit case of Theorem 4.4.1 with

$$CM = \frac{(1 - \frac{\gamma}{2}\lambda_{\max}(\mathbf{I} - \mathbf{W}))\eta\mu}{2\alpha}, \quad \hat{\mathbf{Z}} = \mathbf{Z}, \quad \mathbf{H} = \mathbf{Z}.$$

By taking $\alpha = \frac{1}{2(1+C)}, \eta = \frac{1}{2L}$ and $\gamma = \frac{2}{(1+16C\kappa_f)\lambda_{\max}(\mathbf{I}-\mathbf{W})}$, Theorem 4.4.1 shows the convergence complexity of Prox-LEAD is $\widetilde{\mathcal{O}}(C+\kappa_f+\kappa_g+C\kappa_f\kappa_g)$ when full gradient is used. This complexity significantly improves that of LEAD, which is $\widetilde{\mathcal{O}}((1 + C)(\kappa_f + \kappa_g) + C\kappa_f\kappa_g)$. A complete comparison can be found in Table B.1.

4.4.2 Finite-sum Setting with Variance Reduction

In the finite-sum setting, we assume that each f_i is the average of m functions f_{ij} and impose the commonly assumed L-Lipschitz continuity to ∇f_{ij}. We will keep using Φ^k defined in Theorem 4.4.1. The following theorems implicitly assumes $C > 0$ while the convergence results are extendable to the case without compression. The detailed proof of two theorems can be found in supplemental material.

Theorem 4.4.2 (Loopless SVRG)**.** *Under Assumptions 4.2.1, 4.2.2, and 4.3.2, for any* $p \in (0,1)$, *set* $p_{ij} = \frac{1}{m}, \forall i \in [n], \forall j \in [m]$,

$$\eta = \frac{1}{6L}, \quad \alpha = \frac{1}{2(1+C)}, \quad \gamma = \frac{1}{\lambda_{\max}} \frac{2}{1 + 48C\kappa_f},$$

then, in total expectation, we have

$$\mathbb{E}\widetilde{\Phi}^{k+1} \le \left(1 - \left(\max\left\{12\kappa_f - 1, \kappa_g + 48C\kappa_f\kappa_g, 1+C, 6\kappa_f, \frac{2}{p}\right\}\right)^{-1}\right)^{k+1} \widetilde{\Phi}^0,$$

where $\widetilde{\Phi}^k := \Phi^k + \frac{2}{9pL}\sum_{i=1}^n \mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^k, \mathbf{x}^*)$.

Theorem 4.4.3 (SAGA)**.** *Under Assumption 4.2.1, 4.2.2 and 4.3.2, set* $p_{ij} = \frac{1}{m}, \forall i \in [n], \forall j \in [m]$,

$$\eta = \frac{1}{6L}, \quad \alpha = \frac{1}{2(1+C)}, \quad \gamma = \frac{1}{\lambda_{\max}} \frac{2}{1 + 48C\kappa_f},$$

then, in total expectation, we have

$$\mathbb{E}\widetilde{\Phi}^{k+1} \le \left(1 - \left(\max\left\{12\kappa_f - 1, \kappa_g + 48C\kappa_f\kappa_g, 1+C, 6\kappa_f, 2m\right\}\right)^{-1}\right)^{k+1} \widetilde{\Phi}^0,$$

where $\widetilde{\Phi}^k := \Phi^k + \frac{2}{9L}\sum_{i=1}^n \sum_{j=1}^m \mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^k, \mathbf{x}^*)$.

Using $\widetilde{\mathcal{O}}(\cdot)$ as the abbreviation of $\mathcal{O}((\cdot)\log(1/\epsilon))$, we simplify the above two theorems as the following corollary.

Corollary 4.4.2. *We can achieve* $\|\mathbf{x}_i^k - \mathbf{x}^*\|^2 \le \epsilon$ *in expectation on each node after the number of iterations*

- $$K = \widetilde{\mathcal{O}}\left(C + \kappa_f + \kappa_g + C\kappa_f\kappa_g + p^{-1}\right)$$

  for Loopless SVRG and

- $$K = \widetilde{\mathcal{O}}\left(C + \kappa_f + \kappa_g + C\kappa_f\kappa_g + m\right)$$

  for SAGA.

Proof. From the definition of Φ^k, we have

$$\|\mathbf{Z}^k - \mathbf{Z}^*\|_{\mathbf{P}}^2 \leq \frac{12\kappa_f}{12\kappa_f - 1}\Phi^k$$

and

$$\frac{\gamma^2}{4}\|\hat{\mathbf{Z}}^k - \mathbf{Z}^k\|_{(\mathbf{I}-\mathbf{W})^2}^2 \leq \|\hat{\mathbf{Z}}^k - \mathbf{Z}^k\|_{\mathbf{I}-\mathbf{P}}^2 \leq \Phi^k.$$

Using (4.11) in Lemma 4.4.1, we have

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\|\mathbf{x}_i^k - \mathbf{x}^*\|^2 &\leq \frac{1}{n}\mathbb{E}\|\mathbf{Z}^k - \mathbf{Z}^*\|_{\mathbf{P}}^2 + \frac{\gamma^2}{4n}\mathbb{E}\|\hat{\mathbf{Z}}^k - \mathbf{Z}^k\|_{(\mathbf{I}-\mathbf{W})^2}^2\\ &\leq \frac{24\kappa_f - 1}{n(12\kappa_f - 1)}\mathbb{E}\Phi^k.\end{aligned}$$

Lastly, the convergence complexity is proved from Theorem 4.4.2 and Theorem 4.4.3. □

Remark 4.4.1. *In particular, when there is no compression, i.e., $C = 0$ and the network is fully connected, i.e., $\kappa_g = 1$, the complexity of Loopless SVRG is reduced to $\widetilde{\mathcal{O}}\left(\kappa_f + p^{-1}\right)$, which matches that given in [99] and the complexity of SAGA is reduced to $\widetilde{\mathcal{O}}\left(\kappa_f + m\right)$ shown in [98].*

4.5 Numerical Experiments

In this section, we present numerical experiments to validate the convergence of the proposed algorithms, including LEAD in the smooth case and Prox-LEAD in the nonsmooth case, as well as their stochastic variants.

4.5.1 Experimental Setting

Baselines. To demonstrate the effectiveness of the proposed algorithms, we compare them with the following baselines: 1) two state-of-the-art decentralized algorithms with compression: Choco [24] and LessBit [81]; 2) three non-compressed algorithms: DGD [27], NIDS [2], and P2D2 [58]. Note that NIDS and P2D2 support the nonsmooth case. In the stochastic case, we also include LessBit with Option C (LessBit-SGD) and Option D (LessBit-LSVRG).

Setup. We consider eight machines connected in a ring topology network. Each agent can only exchange information with its two 1-hop neighbors. The mixing weight is simply set as $1/3$. For compression, we use the unbiased b-bits quantization method with ∞-norm

$$Q_\infty(\mathbf{x}) := \left(\frac{\|\mathbf{x}\|_\infty \mathrm{sign}(\mathbf{x})}{2^{b-1}}\right) \cdot \left\lfloor \frac{2^{(b-1)}|\mathbf{x}|}{\|\mathbf{x}\|_\infty} + \mathbf{u} \right\rfloor, \tag{4.17}$$

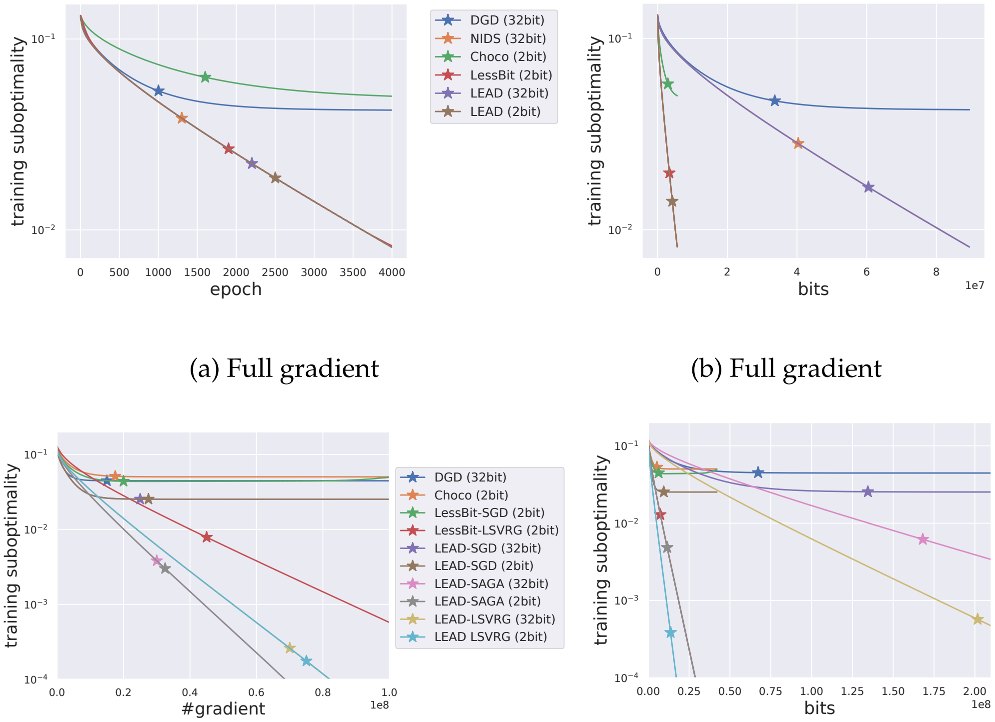
where $\cdot$ is the Hadamard product, $|\mathbf{x}|$ is the elementwise absolute value of $\mathbf{x}$, and $\mathbf{u}$ is a random vector uniformly distributed in $[0,1]^p$. Only sign($\mathbf{x}$), norm $\|\mathbf{x}\|_\infty$, and integers in the bracket need to be transmitted. Note that this quantization method is similar to the quantization used in QSGD [13] and Choco [24], but we use the ∞-norm scaling instead of the 2-norm. This small change brings significant improvement on compression precision as justified both theoretically and empirically in Appendix C in [33]. In this section, we choose 2-bit quantization and quantize the data in a blockwise manner (block size = 256).

For all experiments, we tune the stepsize η in the range $[0.01, 0.1]$. For LEAD and Prox-LEAD, we simply fix $\alpha = 0.5$ and $\gamma = 1.0$ for all experiments since they are very robust to the parameter settings. The parameters γ in Choco and θ in LessBit are tuned from $\{0.01, 0.05, 0.1, 0.2, 0.5, 0.8, 1.0\}$.

Logistic regression. We consider a regularized logistic regression problem with a cross-entropy objective function:

$$f(\mathbf{X}) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{C} \mathbf{y}_{i,j} \log(\mathbf{a}_i^\top \mathbf{X}_j) + \lambda_1\|\mathbf{X}\|_1 + \lambda_2\|\mathbf{X}\|_2^2,$$

where $\mathbf{a}_i \in \mathbb{R}^p, \mathbf{y} \in \mathbb{R}^{m\times C}, \mathbf{X} \in \mathbb{R}^{p\times C}$ and C is the number of classes. We use the MNIST dataset and distribute the samples equally to all the machines in a non-iid way, sorted by their labels. Note that this is the heterogeneous data settings where the data distribution from each agent is very different, which is more challenging than the homogeneous data setting where all the agents share the same data distribution. In the smooth case, we set $\lambda_1 = 0$ and $\lambda_2 = 0.005$, and in the nonsmooth case, we set $\lambda_1 = 0.005$ and $\lambda_2 = 0.005$. In the stochastic case, the training data in each agent are evenly divided into 15 mini-batches. The performance is measured by the the training suboptimality, i.e., $\|\mathbf{X}^k - \mathbf{X}^*\|_F^2$, where $\mathbf{X}^*$ denotes the optimal solution.



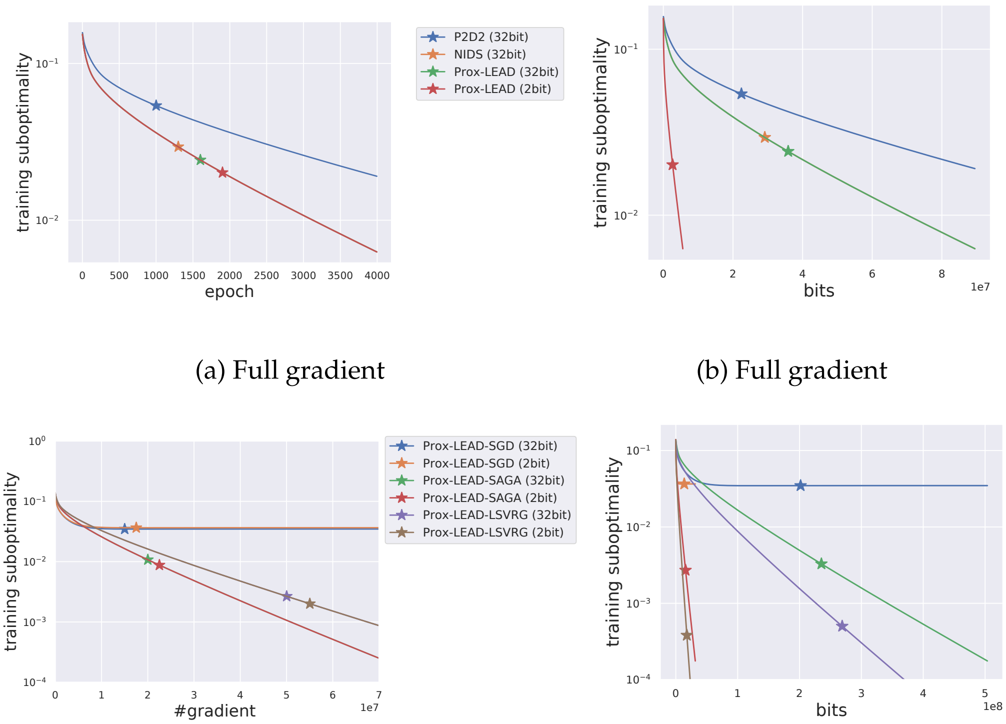
(a) Full gradient

(b) Full gradient

(c) Stochastic gradient

(d) Stochastic gradient

Figure 4.1: Smooth logistic regression problem ($\lambda_1 = 0$). In the full gradient case ((a) and (b)), LEAD (2bit) and LessBit (2bit) converge similarly as NIDS (32bit) and LEAD (32bit) in terms of epochs/iterations, but they requires much fewer bits in communication. In the stochastic case ((c) and (d)), the 2bit variants of LEAD match well with their 32bit variants in terms of the number of gradient evaluation, but they requires much fewer bits. Note that LEAD-SAGA requires more memory and more iterations/communication than LEAD-LSVRG, but it computes only one gradient in each iteration, while LEAD-LSVRG computes at least two gradients in each iteration.



(a) Full gradient

(b) Full gradient

(c) Stochastic gradient

(d) Stochastic gradient

Figure 4.2: Nonsmooth logistic regression problem ($\lambda_1 = 0.005$). In the full gradient case ((a) and (b)), Prox-LEAD (2bit) converges similarly as NIDS and Prox-LEAD (32bit) in terms of epochs/iterations, but it requires much fewer bits than the other three algorithms. In the stochastic case ((c) and (d)), the 2bit variants match well with their 32bit variants in terms of the number of gradient evaluation, but they requires much fewer bits. Note that Prox-LEAD-SAGA requires more memory and more iterations/communication than Prox-LEAD-LSVRG, but it computes only one gradient in each iteration, while Prox-LEAD-LSVRG computes at least two gradients in each iteration.

Smooth case. The experiments in the smooth case are showed in Fig. 4.1. From Fig. 4.1a, we can observe that when full gradients are available, NIDS, LessBit and LEAD converge linearly to the optimal solution, while DGD and Choco suffer from the convergence bias. Fig. 4.1b demonstrates the benefit of communication compression when considering the suboptimality with respect to the communcation bits. Note that the performance of LEAD (32bit) matches well with LEAD (2bit), which validates that the compression doesn't hurt the convergence for LEAD, while the communication bits are significantly reduced.

Fig. 4.1c and Fig. 4.1d show the performance of algorithms with stochastic gradients. We can make the following observations: 1) The performance of LEAD-SGD (2bit), LEAD-SAGA (2bit) and LEAD-LSVRG (2bit) match well with LEAD-SGD (32bit), LEAD-SAGA (32bit) and LEAD-LSVRG (32bit), respectively, which indicates that the compression in LEAD doesn't hurt the convergence; 2) The linear convergence of the variance-reduction variants, such as LEAD-SAGA (2bit) and LEAD-LSVRG (2bit), verifies our theoretical analyses[1]; 3) LEAD-SAGA (2bit) and LEAD-LSVRG (2bit) significantly outperform all baselines[2]. The benefit of communication compression can be clearly illustrated by Fig. 4.1d.

Nonsmooth case. The experiments in the nonsmooth case are showed in Fig. 4.2. Fig. 4.2a shows that Prox-LEAD (2bit) achieves linear convergence to the optimal solution with full gradient, and its performance matches well with the non-compressed version Prox-LEAD (32bit). It also converges similarly with other non-compressed baselines such as P2D2 and NIDS. Fig. 4.2b demonstrates the tremendous advantages of communication compression in Prox-LEAD (2bit) when considering the communication bits.

Fig. 4.2c and Fig. 4.2d present the performance with stochastic gradients. It can be observed from Fig. 4.2c that: 1) Prox-LEAD-SAGA (2bit) and Prox-LEAD-LSVRG (2bit) maintains linear convergence with communication compression and stochastic gradients; 2) The compressed versions of Prox-LEAD all match well with the non-compressed versions. Fig. 4.2b shows that the advantages of communication compression in Prox-LEAD are very significant in terms of communication bits.

[1] LEAD-SAGA outperform LEAD-LSVRG in terms of the number of gradient computation since LEAD-SAGA computes fewer gradient evaluations in each iteration by sacrificing the memory space. However, LEAD-LSVRG outperforms LEAD-SAGA in terms of communication bits since the extra gradient computation in LEAD-LSVRG doesn't increase communication cost but it improves the convergence speed. Similar phenomenon is also observed in the nonsmooth case in Figure 4.2c and Fig. 4.2d.

[2] LEAD-SGD (2bit) and LEAD-LSVRG (2bit) outperform LessBit-SGD (2bit) and LessBit-LSVRG (2bit), which shows the advantages of the extra gradient descent step in LEAD, as discussed in Section B.2. Though LessBit-LSVRG (2bit) has the same communication bit as LEAD-SAGA (2bit), LEAD-SAGA (2bit) requires about half of the gradient evaluation as LessBit-LSVRG (2bit).

To summarize, the experiments in this section verify the theoretical linear convergence of the proposed algorithm when the nonsmooth objective, stochastic gradients and communication compression are present. They also suggest the state-of-the-art performance in the comparison with strong baseline algorithms.

4.6 Conclusion

In this chapter, we consider the decentralized stochastic composite optimization problem. A decentralized proximal stochastic gradient algorithm with communication compression, Prox-LEAD, is proposed to improve the communication efficiency and convergence rates. We provide rigorous theoretical analyses and convergence complexities for the proposed algorithm in the general stochastic setting and the finite-sum setting. We establish the linear convergence rate with variance reduction schemes and well-controlled compression error. Both the theorems and numerical experiments demonstrate the effectiveness of Prox-LEAD in reducing the communication cost and the advantages over existing algorithms. Moreover, our algorithmic framework builds bridges between many known algorithms, and it potentially enlightens the communication compression for other primal-dual algorithms.

APPENDICES

APPENDIX A

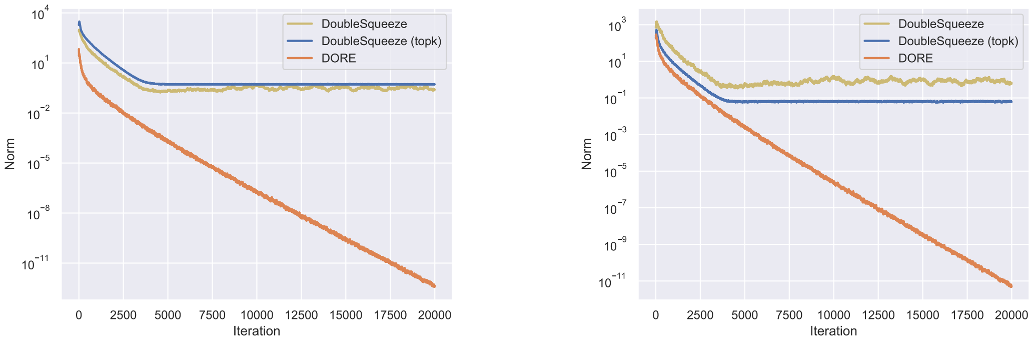
SUPPLEMENTARY OF CHAPTER 3

A.1 Compression Error

The property of the compression operator indicates that the compression error is linearly proportional to the norm of the variable being compressed:

$$\mathbb{E}\|Q(\mathbf{x}) - \mathbf{x}\|^2 \leq C\|\mathbf{x}\|^2.$$

We visualize the norm of the variables being compressed, i.e., the gradient residual (the worker side) and model residual (the master side) for DORE as well as error compensated gradient (the worker side) and averaged gradient (the master side) for DoubleSqueeze. As showed in Figure A.1, the gradient and model residual of DORE decrease exponentially and the compression errors vanish. However, for DoubleSqueeze, their norms only decrease to some certain value and the compression error doesn't vanish. It explains why algorithms without residual compression cannot converge linearly to the $\mathcal{O}(\sigma)$ neighborhood of the optimal solution in the strongly convex case.



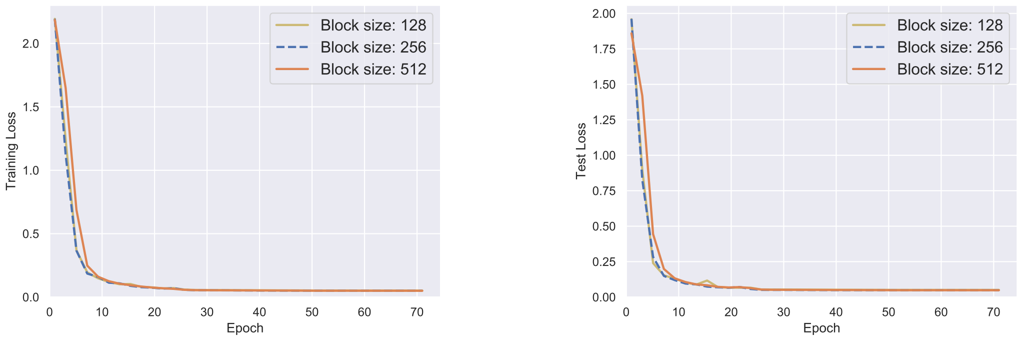
(a) Worker side

(b) Master side

Figure A.1: The norm of compressed variable in linear regression.

A.2 Parameter Sensitivity

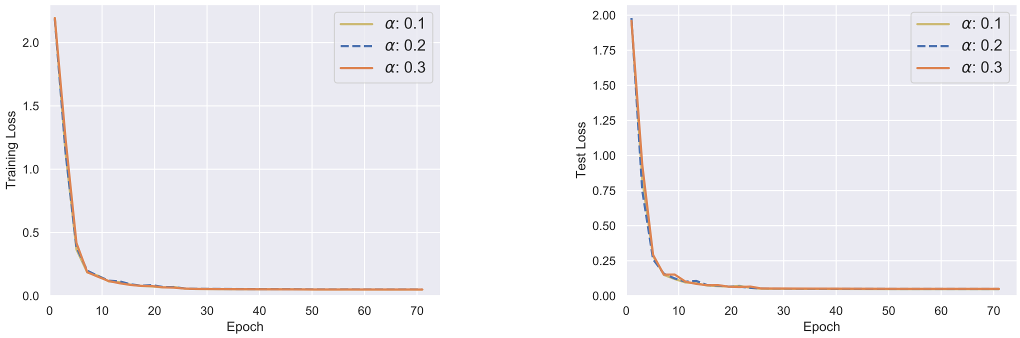
Continuing the MNIST experiment in Section 3.5, we further conduct parameter analysis on DORE. The basic setting for block size, learning rate, α, β and η are 256, 0.1, 0.1, 1, 1, respectively. We change each parameter individually. Figures A.2, A.3, A.4, and A.5 demonstrate that DORE performs consistently well under different parameter settings.



(a) Training loss

(b) Test loss

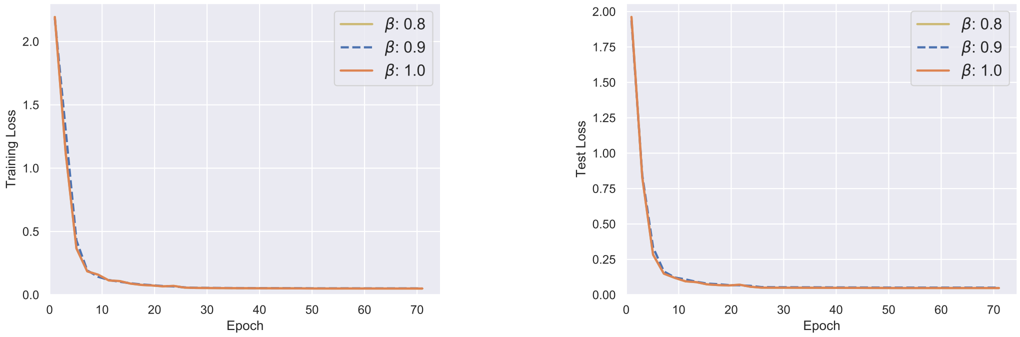
Figure A.2: Training under different compression block sizes.



(a) Training loss

(b) Test loss

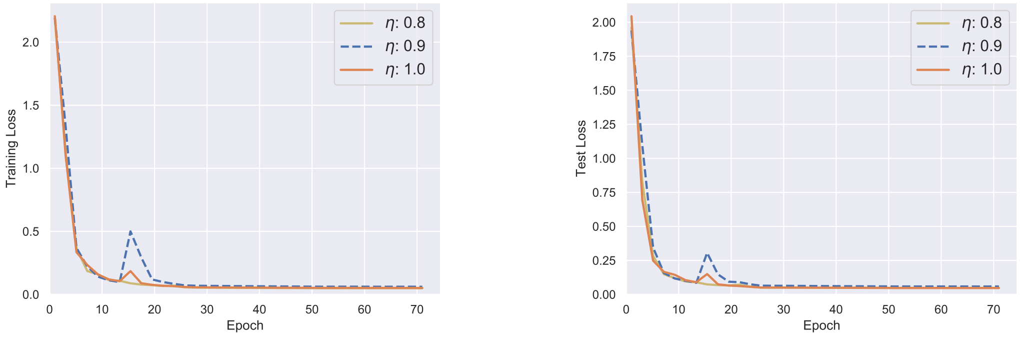
Figure A.3: Training under different α.



(a) Training loss

(b) Test loss

Figure A.4: Training under different β.



(a) Training loss

(b) Test loss

Figure A.5: Training under different η.

A.3 DORE in the Smooth Case

Algorithm A.1: DORE with $R(\mathbf{x}) = 0$

```
1:  Input: Stepsize $\alpha, \beta, \gamma, \eta$, initialize $\mathbf{h}^0 = \mathbf{h}_i^0 = \mathbf{0}^d$, $\hat{\mathbf{x}}_i^0 = \hat{\mathbf{x}}^0,\ \forall i \in \{1, \dots, n\}$.
2:  for $k = 1, 2, \cdots, K-1$ do
3:      For each worker $\{i = 1, 2, \cdots, n\}$:
4:      Sample $\mathbf{g}_i^k$ such that $\mathbb{E}[\mathbf{g}_i^k|\hat{\mathbf{x}}_i^k] = \nabla f_i(\hat{\mathbf{x}}_i^k)$
5:      Gradient residual: $\Delta_i^k = \mathbf{g}_i^k - \mathbf{h}_i^k$
6:      Compression: $\hat{\Delta}_i^k = Q(\Delta_i^k)$
7:      $\mathbf{h}_i^{k+1} = \mathbf{h}_i^k + \alpha\hat{\Delta}_i^k$
8:      $\{\ \hat{\mathbf{g}}_i^k = \mathbf{h}_i^k + \hat{\Delta}_i^k\ \}$
9:      Sent $\hat{\Delta}_i^k$ to the master
10:     Receive $\hat{\mathbf{q}}^k$ from the master
11:     $\hat{\mathbf{x}}_i^{k+1} = \hat{\mathbf{x}}_i^k + \beta\hat{\mathbf{q}}^k$
12:     For the master:
13:     Receive $\hat{\Delta}_i^k$s from workers
14:     $\hat{\Delta}^k = 1/n \sum_i^n \hat{\Delta}_i^k$
15:     $\hat{\mathbf{g}}^k = \mathbf{h}^k + \hat{\Delta}^k\ \ \{= 1/n \sum_i^n \hat{\mathbf{g}}_i^k\ \}$
16:     $\mathbf{h}^{k+1} = \mathbf{h}^k + \alpha\hat{\Delta}^k$
17:     $\mathbf{q}^k = -\gamma\hat{\mathbf{g}}^k + \eta\mathbf{e}^k$
18:     Compression: $\hat{\mathbf{q}}^k = Q(\mathbf{q}^k)$
19:     $\mathbf{e}^{k+1} = \mathbf{q}^k - \hat{\mathbf{q}}^k$
20:     Broadcast $\hat{\mathbf{q}}^k$ to workers
21: end for
22: Output: any $\hat{\mathbf{x}}_i^K$
```

A.4 Proof of Theorem 3.4.1

We first provide two lemmas. We define $\mathbb{E}_Q$, $\mathbb{E}_k$, and $\mathbb{E}$ be the expectation taken over the quantization, the kth iteration based on $\hat{\mathbf{x}}^k$, and the overall expectation, respectively.

Lemma A.4.1. *For every i, we can estimate the first two moments of $\mathbf{h}_i^{k+1}$ as*

$$\mathbb{E}_Q\mathbf{h}_i^{k+1} = (1-\alpha)\mathbf{h}_i^k + \alpha\mathbf{g}_i^k, \tag{A.1}$$

$$\mathbb{E}_Q\|\mathbf{h}_i^{k+1} - \mathbf{s}_i\|^2 \leq (1-\alpha)\|\mathbf{h}_i^k - \mathbf{s}_i\|^2 + \alpha\|\mathbf{g}_i^k - \mathbf{s}_i\|^2 + \alpha[(C_q+1)\alpha - 1]\|\Delta_i^k\|^2. \tag{A.2}$$

Proof. The first equality follows from lines 5-7 of Algorithm 3.1 and Assumption 3.3.1. For the second equation, we have the following variance decomposition

$$\mathbb{E}\|X\|^2 = \|\mathbb{E}X\|^2 + \mathbb{E}\|X - \mathbb{E}X\|^2 \tag{A.3}$$

for any random vector X. By taking $X = \mathbf{h}_i^{k+1} - \mathrm{s}_i$, we get

$$\mathbb{E}_Q\|\mathbf{h}_i^{k+1} - \mathbf{s}_i\|^2 = \|(1-\alpha)(\mathbf{h}_i^k - \mathbf{s}_i) + \alpha(\mathbf{g}_i^k - \mathbf{s}_i)\|^2 + \alpha^2\mathbb{E}_Q\|\hat{\Delta}_i^k - \Delta_i^k\|^2. \tag{A.4}$$

Using the basic equality

$$\|\lambda \mathbf{a} + (1-\lambda)\mathbf{b}\|^2 + \lambda(1-\lambda)\|\mathbf{a}-\mathbf{b}\|^2 = \lambda\|\mathbf{a}\|^2 + (1-\lambda)\|\mathbf{b}\|^2 \tag{A.5}$$

for all $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$ and $\lambda \in [0,1]$, as well as Assumption 3.3.1, we have

$$\mathbb{E}_Q\|\mathbf{h}_i^{k+1} - \mathbf{s}_i\|^2 \leq (1-\alpha)\|\mathbf{h}_i^k - \mathbf{s}_i\|^2 + \alpha\|\mathbf{g}_i^k - \mathbf{s}_i\|^2 - \alpha(1-\alpha)\|\Delta_i^k\|^2 + \alpha^2 C_q\|\Delta_i^k\|^2, \tag{A.6}$$

which is the inequality (A.2). □

Next, from the variance decomposition (A.3), we also derive Lemma A.4.2.

Lemma A.4.2. *The following inequality holds*

$$\mathbb{E}[\|\hat{\mathbf{g}}^k - \mathbf{h}^*\|^2] \leq \mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k) - \mathbf{h}^*\|^2 + \frac{C_q}{n^2}\sum_{i=1}^n \mathbb{E}\|\Delta_i^k\|^2 + \frac{\sigma^2}{n}, \tag{A.7}$$

where $\mathbf{h}^* = \nabla f(\mathbf{x}^*) = \frac{1}{n}\sum_{i=1}^n \mathbf{h}_i^*$ *and* $\sigma^2 = \frac{1}{n}\sum_{i=1}^n \sigma_i^2$*.*

Proof. By taking the expectation over the quantization of g, we have

$$\begin{aligned}\mathbb{E}\|\hat{\mathbf{g}}^k - \mathbf{h}^*\|^2 &= \mathbb{E}\|\mathbf{g}^k - \mathbf{h}^*\|^2 + \mathbb{E}\|\hat{\mathbf{g}}^k - \mathbf{g}^k\|^2 \\ &\leq \mathbb{E}\|\mathbf{g}^k - \mathbf{h}^*\|^2 + \frac{C_q}{n^2}\sum_{i=1}^n \mathbb{E}\|\Delta_i^k\|^2,\end{aligned} \tag{A.8}$$

where the inequality is from Assumption 3.3.1.

For $\|\mathbf{g}^k - \mathbf{h}^*\|$, we take the expectation over the sampling of gradients and derive

$$\begin{aligned}\mathbb{E}\|\mathbf{g}^k - \mathbf{h}^*\|^2 &= \mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k) - \mathbf{h}^*\|^2 + \mathbb{E}\|\mathbf{g}^k - \nabla f(\hat{\mathbf{x}}^k)\|^2 \\ &\leq \mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k) - \mathbf{h}^*\|^2 + \frac{\sigma^2}{n}\end{aligned} \tag{A.9}$$

by Assumption 3.4.1.

Combining (A.8) with (A.9) gives (A.7). □

Proof of Theorem 3.4.1. We consider $\mathbf{x}^{k+1} - \mathbf{x}^*$ first. Since $\mathbf{x}^*$ is the solution of (1.1), it satisfies

$$\mathbf{x}^* = \mathbf{prox}_{\gamma R}(\mathbf{x}^* - \gamma \mathbf{h}^*). \tag{A.10}$$

Hence

$$\begin{aligned}\mathbb{E}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 =& \mathbb{E}\|\mathbf{prox}_{\gamma R}(\hat{\mathbf{x}}^k - \gamma\hat{\mathbf{g}}^k) - \mathbf{prox}_{\gamma R}(\mathbf{x}^* - \gamma\mathbf{h}^*)\|^2 \\ \leq& \mathbb{E}\|\hat{\mathbf{x}}^k - \mathbf{x}^* - \gamma(\hat{\mathbf{g}}^k - \mathbf{h}^*)\|^2 \\ =& \mathbb{E}\|\hat{\mathbf{x}}^k - \mathbf{x}^*\|^2 - 2\gamma\mathbb{E}\langle\hat{\mathbf{x}}^k - \mathbf{x}^*, \hat{\mathbf{g}}^k - \mathbf{h}^*\rangle + \gamma^2\mathbb{E}\|\hat{\mathbf{g}}^k - \mathbf{h}^*\|^2 \\ =& \mathbb{E}\|\hat{\mathbf{x}}^k - \mathbf{x}^*\|^2 - 2\gamma\mathbb{E}\langle\hat{\mathbf{x}}^k - \mathbf{x}^*, \nabla f(\hat{\mathbf{x}}^k) - \mathbf{h}^*\rangle + \gamma^2\mathbb{E}\|\hat{\mathbf{g}}^k - \mathbf{h}^*\|^2, \qquad \text{(A.11)}\end{aligned}$$

where the inequality comes from the non-expansiveness of the proximal operator and the last equality is derived by taking the expectation of the stochastic gradient $\hat{\mathbf{g}}^k$.

Combining (A.7) and (A.11), we have

$$\begin{aligned}\mathbb{E}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 \leq& \mathbb{E}\|\hat{\mathbf{x}}^k - \mathbf{x}^*\|^2 - 2\gamma\mathbb{E}\langle\hat{\mathbf{x}}^k - \mathbf{x}^*, \nabla f(\hat{\mathbf{x}}^k) - \mathbf{h}^*\rangle \\ &+ \frac{\gamma^2}{n}\sum_{i=1}^n \mathbb{E}\|\nabla f_i(\hat{\mathbf{x}}^k) - \mathbf{h}_i^*\|^2 + \frac{C_q\gamma^2}{n^2}\sum_{i=1}^n \mathbb{E}\|\Delta_i^k\|^2 + \frac{\gamma^2}{n}\sigma^2. \qquad \text{(A.12)}\end{aligned}$$

Then we consider $\mathbb{E}\|\hat{\mathbf{x}}^{k+1} - \mathbf{x}^*\|^2$. According to Algorithm 3.1, we have:

$$\begin{aligned}\mathbb{E}_Q[\hat{\mathbf{x}}^{k+1} - \mathbf{x}^*] &= \hat{\mathbf{x}}^k + \beta\mathbf{q}^k - \mathbf{x}^* \\ &= (1-\beta)(\hat{\mathbf{x}}^k - \mathbf{x}^*) + \beta(\mathbf{x}^{k+1} - \mathbf{x}^* + \eta\mathbf{e}^k) \qquad \text{(A.13)}\end{aligned}$$

where the expectation is taken on the quantization of $\mathbf{q}^k$.

By variance decomposition (A.3) and the basic equality (A.5),

$$\begin{aligned}&\mathbb{E}\|\hat{\mathbf{x}}^{k+1} - \mathbf{x}^*\|^2 \\ \leq& (1-\beta)\mathbb{E}\|\hat{\mathbf{x}}^k - \mathbf{x}^*\|^2 + \beta\mathbb{E}\|\mathbf{x}^{k+1} + \eta\mathbf{e}^k - \mathbf{x}^*\|^2 - \beta(1-\beta)\mathbb{E}\|\mathbf{q}^k\|^2 + \beta^2 C_q^m\mathbb{E}\|\mathbf{q}^k\|^2 \\ \leq& (1-\beta)\mathbb{E}\|\hat{\mathbf{x}}^k - \mathbf{x}^*\|^2 + (1+\eta^2\epsilon)\beta\mathbb{E}\|\mathbf{x}^{k+1} - \mathbf{x}^*\|^2 - \beta(1-(C_q^m+1)\beta)\mathbb{E}\|\mathbf{q}^k\|^2 \\ &+ (\eta^2 + \frac{1}{\epsilon})\beta C_q^m\mathbb{E}\|\mathbf{q}^{k-1}\|^2, \qquad \text{(A.14)}\end{aligned}$$

where ϵ is generated from Cauchy inequality of inner product. For convenience, we let $\epsilon = \frac{1}{\eta}$.

Choose a β such that $0 < \beta \leq \frac{1}{1+C_q^m}$. Then we have

$$\begin{aligned}
&\beta(1-(C_q^m+1)\beta)\mathbb{E}\|\mathbf{q}^k\|^2 + \mathbb{E}\|\hat{\mathbf{x}}^{k+1}-\mathbf{x}^*\|^2 \\
\leq&(1-\beta)\mathbb{E}\|\hat{\mathbf{x}}^k-\mathbf{x}^*\|^2 + (1+\eta)\beta\mathbb{E}\|\mathbf{x}^{k+1}-\mathbf{x}^*\|^2 + (\eta^2+\eta)\beta C_q^m\mathbb{E}\|\mathbf{q}^{k-1}\|^2. \qquad \text{(A.15)}
\end{aligned}$$

Letting $\mathbf{s}_i = \mathbf{h}_i^*$ in (A.2), we have

$$\begin{aligned}
&\frac{(1+\eta)c\beta\gamma^2}{n}\sum_{i=1}^n \mathbb{E}\|\mathbf{h}_i^{k+1}-\mathbf{h}_i^*\|^2 \\
\leq&\frac{(1+\eta)(1-\alpha)c\beta\gamma^2}{n}\sum_{i=1}^n \|\mathbf{h}_i^k-\mathbf{h}_i^*\|^2 + \frac{(1+\eta)\alpha c\beta\gamma^2}{n}\sum_{i=1}^n \|\mathbf{g}_i^k-\mathbf{h}_i^*\|^2 \\
&+\frac{(1+\eta)\alpha[(C_q+1)\alpha-1]c\beta\gamma^2}{n}\sum_{i=1}^n \|\Delta_i^k\|^2. \qquad \text{(A.16)}
\end{aligned}$$

Then we let $\mathbf{R}^k = \beta(1-(C_q^m+1)\beta)\mathbb{E}\|\mathbf{q}^k\|^2$ and define $\mathbf{V}^k = \mathbf{R}^{k-1} + \mathbb{E}\|\hat{\mathbf{x}}^k-\mathbf{x}^*\|^2 + \frac{(1+\eta)c\beta\gamma^2}{n}\sum_{i=1}^n \mathbb{E}\|\mathbf{h}_i^k-\mathbf{h}_i^*\|^2$. Thus, we obtain

$$\begin{aligned}
\mathbf{V}^{k+1} \leq&(\eta^2+\eta)\beta C_q^m\mathbb{E}\|\mathbf{q}^{k-1}\|^2 + (1+\eta\beta)\mathbb{E}\|\hat{\mathbf{x}}^k-\mathbf{x}^*\|^2 \\
&-2(1+\eta)\beta\gamma\mathbb{E}\langle\hat{\mathbf{x}}^k-\mathbf{x}^*, \nabla f(\hat{\mathbf{x}}^k)-\mathbf{h}^*\rangle + \frac{(1+\eta)(1-\alpha)c\beta\gamma^2}{n}\sum_{i=1}^n \mathbb{E}\|\mathbf{h}_i^k-\mathbf{h}_i^*\|^2 \\
&+\frac{(1+\eta)\beta\gamma^2}{n^2}\Big[nc(C_q+1)\alpha^2 - nc\alpha + C_q\Big]\sum_{i=1}^n \mathbb{E}\|\Delta_i^k\|^2 \\
&+\frac{(1+\eta)(1+c\alpha)}{n}\beta\gamma^2\sum_{i=1}^n \mathbb{E}\|\nabla f_i(\hat{\mathbf{x}}^k)-\mathbf{h}_i^*\|^2 + \frac{(1+\eta)(1+nc\alpha)}{n}\beta\gamma^2\sigma^2. \qquad \text{(A.17)}
\end{aligned}$$

The $\mathbb{E}\|\Delta_i^k\|^2$-term can be ignored if $nc(C_q+1)\alpha^2 - nc\alpha + C_q \leq 0$, which can be guaranteed by $c \geq \frac{4C_q(C_q+1)}{n}$ and

$$\alpha \in \left(\frac{1-\sqrt{1-\frac{4C_q(C_q+1)}{nc}}}{2(C_q+1)}, \frac{1+\sqrt{1-\frac{4C_q(C_q+1)}{nc}}}{2(C_q+1)}\right).$$

Given that each f_i is L-Lipschitz differentiable and μ-strongly convex, we have

$$\mathbb{E}\langle\nabla f(\hat{\mathbf{x}}^k)-\mathbf{h}^*, \hat{\mathbf{x}}^k-\mathbf{x}^*\rangle \geq \frac{\mu L}{\mu+L}\mathbb{E}\|\hat{\mathbf{x}}^k-\mathbf{x}^*\|^2 + \frac{1}{\mu+L}\frac{1}{n}\sum_{i=1}^n \mathbb{E}\|\nabla f_i(\hat{\mathbf{x}}^k)-\mathbf{h}_i^*\|^2. \qquad \text{(A.18)}$$

Hence

$$\begin{aligned}
\mathbf{V}^{k+1} \leq& \rho_1 \mathbf{R}^{k-1} + (1+\eta\beta)\mathbb{E}\|\hat{\mathbf{x}}^k - \mathbf{x}^*\|^2 - 2(1+\eta)\beta\gamma\mathbb{E}\langle \hat{\mathbf{x}}^k - \mathbf{x}^*, \nabla f(\hat{\mathbf{x}}^k) - \mathbf{h}^*\rangle \\
&+ \frac{(1+\eta)(1-\alpha)c\beta\gamma^2}{n}\sum_{i=1}^n \mathbb{E}\|\mathbf{h}_i^k - \mathbf{h}_i^*\|^2 + \frac{(1+\eta)(1+c\alpha)}{n}\beta\gamma^2\sum_{i=1}^n \mathbb{E}\|\nabla f_i(\hat{\mathbf{x}}^k) - \mathbf{h}_i^*\|^2 \\
&+ \frac{(1+\eta)(1+nc\alpha)}{n}\beta\gamma^2\sigma^2 \\
\leq& \rho_1 \mathbf{R}^{k-1} + \Big[1+\eta\beta - \frac{2(1+\eta)\beta\gamma\mu L}{\mu+L}\Big]\mathbb{E}\|\hat{\mathbf{x}}^k - \mathbf{x}^*\|^2 \\
&+ \frac{(1+\eta)(1-\alpha)c\beta\gamma^2}{n}\sum_{i=1}^n \mathbb{E}\|\mathbf{h}_i^k - \mathbf{h}_i^*\|^2 \\
&+ \Big[(1+\eta)(1+c\alpha)\beta\gamma^2 - \frac{2(1+\eta)\beta\gamma}{\mu+L}\Big]\frac{1}{n}\sum_{i=1}^n \mathbb{E}\|\nabla f_i(\hat{\mathbf{x}}^k) - \mathbf{h}_i^*\|^2 \\
&+ \frac{(1+\eta)(1+nc\alpha)}{n}\beta\gamma^2\sigma^2 \\
\leq& \rho_1 \mathbf{R}^{k-1} + \rho_2\mathbb{E}\|\hat{\mathbf{x}}^k - \mathbf{x}^*\|^2 + \frac{(1+\eta)(1-\alpha)c\beta\gamma^2}{n}\sum_{i=1}^n \mathbb{E}\|\mathbf{h}_i^k - \mathbf{h}_i^*\|^2 \\
&+ \frac{(1+\eta)(1+nc\alpha)}{n}\beta\gamma^2\sigma^2 \qquad \text{(A.19)}
\end{aligned}$$

where

$$\begin{aligned}
\rho_1 =& \frac{(\eta^2+\eta)C_q^m}{1-(C_q^m+1)\beta}, \\
\rho_2 =& 1+\eta\beta - \frac{2(1+\eta)\beta\gamma\mu L}{\mu+L}.
\end{aligned}$$

Here we let $\gamma \leq \frac{2}{(1+c\alpha)(\mu+L)}$ such that $(1+\eta)(1+c\alpha)\beta\gamma^2 - \frac{2(1+\eta)\beta\gamma}{\mu+L} \leq 0$ and the last inequality holds. In order to get $\max(\rho_1, \rho_2, 1-\alpha) < 1$, we have the following conditions

$$\begin{aligned}
0 \leq (\eta^2+\eta)C_q^m \leq& 1-(C_q^m+1)\beta, \\
\eta <& \frac{2(1+\eta)\gamma\mu L}{\mu+L}.
\end{aligned}$$

Therefore, the condition for γ is

$$\frac{\eta(\mu+L)}{2(1+\eta)\mu L} \leq \gamma \leq \frac{2}{(1+c\alpha)(\mu+L)},$$

which implies an additional condition for η. Therefore, the condition for η is

$$\eta \in \left[0, \min\left(\frac{-C_q^m + \sqrt{(C_q^m)^2 + 4(1-(C_q^m+1)\beta)}}{2C_q^m}, \frac{4\mu L}{(\mu+L)^2(1+c\alpha) - 4\mu L}\right)\right).$$

where $\eta \leq \frac{4\mu L}{(\mu+L)^2(1+c\alpha)-4\mu L}$ is to ensure $\frac{\eta(\mu+L)}{2(1+\eta)\mu L} \leq \frac{2}{(1+c\alpha)(\mu+L)}$ such that we don't get an empty set for γ.

If we define $\rho = \max\{\rho_1, \rho_2, 1-\alpha\}$, we obtain

$$\mathbf{V}^{k+1} \leq \rho \mathbf{V}^k + \frac{(1+\eta)(1+nc\alpha)}{n}\beta\gamma^2\sigma^2 \tag{A.20}$$

and the proof is completed by applying (A.20) recurrently. □

A.5 Proof of Theorem 3.4.2

Proof. In Algorithm A.1, we can show

$$\begin{aligned}\mathbb{E}\|\hat{\mathbf{x}}^{k+1} - \hat{\mathbf{x}}^k\|^2 &= \beta^2\mathbb{E}\|\hat{\mathbf{q}}^k\|^2 = \beta^2\mathbb{E}\|\mathbb{E}\hat{\mathbf{q}}^k\|^2 + \beta^2\mathbb{E}\|\hat{\mathbf{q}}^k - \mathbb{E}\hat{\mathbf{q}}^k\|^2 \\ &= \beta^2\mathbb{E}\|\mathbf{q}^k\|^2 + \beta^2\mathbb{E}\|\hat{\mathbf{q}}^k - \mathbf{q}^k\|^2 \\ &\leq (1+C_q^m)\beta^2\mathbb{E}\|\mathbf{q}^k\|^2.\end{aligned} \tag{A.21}$$

and

$$\mathbb{E}\|\mathbf{q}^k\|^2 = \mathbb{E}\|-\gamma\hat{\mathbf{g}}^k + \eta\mathbf{e}^k\|^2 \leq 2\gamma^2\mathbb{E}\|\hat{\mathbf{g}}^k\|^2 + 2\eta^2\mathbb{E}\|\mathbf{e}^k\|^2 \leq 2\gamma^2\mathbb{E}\|\hat{\mathbf{g}}^k\|^2 + 2C_q^m\eta^2\mathbb{E}\|\mathbf{q}^{k-1}\|^2. \tag{A.22}$$

Using (A.21)(A.22) and the Lipschitz continuity of $\nabla f(\mathbf{x})$, we have

$$\begin{aligned}
&\mathbb{E}f(\hat{\mathbf{x}}^{k+1}) + (C_q^m + 1)L\beta^2\mathbb{E}\|\mathbf{q}^k\|^2 \\
\leq&\mathbb{E}f(\hat{\mathbf{x}}^k) + \mathbb{E}\langle\nabla f(\hat{\mathbf{x}}^k), \hat{\mathbf{x}}^{k+1} - \hat{\mathbf{x}}^k\rangle + \frac{L}{2}\mathbb{E}\|\hat{\mathbf{x}}^{k+1} - \hat{\mathbf{x}}^k\|^2 + (C_q^m + 1)L\beta^2\mathbb{E}\|\mathbf{q}^k\|^2 \\
=&\mathbb{E}f(\hat{\mathbf{x}}^k) + \beta\mathbb{E}\langle\nabla f(\hat{\mathbf{x}}^k), -\gamma\hat{\mathbf{g}}^k + \eta\mathbf{e}^k\rangle + \frac{(1 + C_q^m)L\beta^2}{2}\mathbb{E}\|\mathbf{q}^k\|^2 + (C_q^m + 1)L\beta^2\mathbb{E}\|\mathbf{q}^k\|^2 \\
=&\mathbb{E}f(\hat{\mathbf{x}}^k) + \beta\mathbb{E}\langle\nabla f(\hat{\mathbf{x}}^k), -\gamma\nabla f(\hat{\mathbf{x}}^k) + \eta\mathbf{e}^k\rangle + \frac{3(C_q^m + 1)L\beta^2}{2}\mathbb{E}\|\mathbf{q}^k\|^2 \\
\leq&\mathbb{E}f(\hat{\mathbf{x}}^k) - \beta\gamma\mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2 + \frac{\beta\eta}{2}\mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2 + \frac{\beta\eta}{2}\mathbb{E}\|\mathbf{e}^k\|^2 \\
&\quad + 3(C_q^m + 1)L\beta^2\Big[\gamma^2\mathbb{E}\|\hat{\mathbf{g}}^k\|^2 + C_q^m\eta^2\mathbb{E}\|\mathbf{q}^{k-1}\|^2\Big] \\
\leq&\mathbb{E}f(\hat{\mathbf{x}}^k) - \Big[\beta\gamma - \frac{\beta\eta}{2} - 3(C_q^m + 1)L\beta^2\gamma^2\Big]\mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2 \\
&+ \frac{3C_q(C_q^m + 1)L\beta^2\gamma^2}{n^2}\sum_{i=1}^n\mathbb{E}\|\Delta_i^k\|^2 + \frac{3(C_q^m + 1)L\beta^2\gamma^2}{n}\sigma^2 \\
&+ \Big[\frac{\beta\eta C_q^m}{2} + (3C_q^m + 1)C_q^m L\beta^2\eta^2\Big]\mathbb{E}\|\mathbf{q}^{k-1}\|^2,
\end{aligned} \tag{A.23}$$

where the last inequality is from (A.7) with $\mathbf{h}^* = \mathbf{0}$.

Letting $\mathbf{s}_i = \mathbf{0}$ in (A.2), we have

$$\mathbb{E}_Q\|\mathbf{h}_i^{k+1}\|^2 \leq (1 - \alpha)\|\mathbf{h}_i^k\|^2 + \alpha\|\mathbf{g}_i^k\|^2 + \alpha[(C_q + 1)\alpha - 1]\|\Delta_i^k\|^2. \tag{A.24}$$

Due to the assumption that each worker samples the gradient from the full dataset, we have

$$\mathbb{E}\mathbf{g}_i^k = \mathbb{E}\nabla f(\hat{\mathbf{x}}^k), \quad \mathbb{E}\|\mathbf{g}_i^k\|^2 \leq \mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2 + \sigma_i^2. \tag{A.25}$$

Define $\Lambda^k = (C_q^m + 1)L\beta^2\|\mathbf{q}^{k-1}\|^2 + f(\hat{\mathbf{x}}^k) - f^* + 3c(C_q^m + 1)L\beta^2\gamma^2\frac{1}{n}\sum_{i=1}^n\mathbb{E}\|\mathbf{h}_i^k\|^2$, and from (A.23), (A.24), and (A.25), we have

$$\begin{aligned}\mathbb{E}\Lambda^{k+1} \leq& \mathbb{E}f(\hat{\mathbf{x}}^k) - f^* + 3(1-\alpha)c(C_q^m+1)L\beta^2\gamma^2\frac{1}{n}\sum_{i=1}^n \mathbb{E}\|\mathbf{h}_i^k\|^2 \\ &- \Big[\beta\gamma - \frac{\beta\eta}{2} - 3(1+c\alpha)(C_q^m+1)L\beta^2\gamma^2\Big]\mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2 \\ &+ \frac{(C_q^m+1)L\beta^2\gamma^2}{n^2}\Big[3nc(C_q+1)\alpha^2 - 3nc\alpha + 3C_q\Big]\sum_{i=1}^n \mathbb{E}\|\Delta_i^k\|^2 \\ &+ 3(1+nc\alpha)\frac{(C_q^m+1)L\beta^2\gamma^2\sigma^2}{n} \\ &+ \Big[\frac{\beta\eta C_q^m}{2} + 3(C_q^m+1)C_q^m L\beta^2\eta^2\Big]\mathbb{E}\|\mathbf{q}^{k-1}\|^2. \end{aligned} \tag{A.26}$$

If we let $c = \frac{4C_q(C_q+1)}{n}$, then the condition of α in (3.5) gives

$$3nc(C_q+1)\alpha^2 - 3nc\alpha + 3C_q \leq 0$$

and

$$\begin{aligned}\mathbb{E}\Lambda^{k+1} \leq& \mathbb{E}f(\hat{\mathbf{x}}^k) - f^* + 3(1-\alpha)c(C_q^m+1)L\beta^2\gamma^2\frac{1}{n}\sum_{i=1}^n \mathbb{E}\|\mathbf{h}_i^k\|^2 \\ &- \Big[\beta\gamma - \frac{\beta\eta}{2} - 3(1+c\alpha)(C_q^m+1)L\beta^2\gamma^2\Big]\mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2 \\ &+ 3(1+nc\alpha)\frac{(C_q^m+1)L\beta^2\gamma^2\sigma^2}{n} \\ &+ [\frac{\beta\eta C_q^m}{2} + 3(C_q^m+1)C_q^m L\beta^2\eta^2]\mathbb{E}\|\mathbf{q}^{k-1}\|^2. \end{aligned} \tag{A.27}$$

Let $\eta = \gamma$ and $\beta\gamma \leq \frac{1}{6(1+c\alpha)(C_q^m+1)L}$, we have

$$\beta\gamma - \frac{\beta\eta}{2} - 3(1+c\alpha)(C_q^m+1)L\beta^2\gamma^2 = \frac{\beta\gamma}{2} - 3(1+c\alpha)(C_q^m+1)L\beta^2\gamma^2 \geq 0.$$

Take $\gamma \leq \min\Big\{\frac{-1+\sqrt{1+\frac{48L^2\beta^2(C_q^m+1)^2}{C_q^m}}}{12L\beta(C_q^m+1)}, \frac{1}{6L\beta(1+c\alpha)(C_q^m+1)}\Big\}$ will guarantee

$$\Big[\frac{\beta\eta C_q^m}{2} + 3(C_q^m+1)C_q^m L\beta^2\eta^2\Big] \leq (C_q^m+1)L\beta^2.$$

Hence we obtain

$$\mathbb{E}\Lambda^{k+1} \leq \mathbb{E}\Lambda^k - \Big[\frac{\beta\gamma}{2} - 3(1+c\alpha)(C_q^m+1)L\beta^2\gamma^2\Big]\mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2 + 3(1+nc\alpha)\frac{(C_q^m+1)L\beta^2\gamma^2\sigma^2}{n}. \tag{A.28}$$

Taking the telescoping sum and plugging the initial conditions, we derive (3.12). □

A.6 Proof of Corollary 3.4.2

Proof. With $\alpha = \frac{1}{2(C_q+1)}$ and $c = \frac{4C_q(C_q+1)}{n}$, $1 + nc\alpha = 1 + 2C_q$ is a constant.

We set $\beta = \frac{1}{C_q^m+1}$ and $\gamma = \min\left\{\frac{-1+\sqrt{1+\frac{48L^2}{C_q^m}}}{12L}, \frac{1}{12L(1+c\alpha)(1+\sqrt{K/n})}\right\}$. In general, C_q^m is bounded which makes the first bound negligible, i.e., $\gamma = \frac{1}{12L(1+c\alpha)(1+\sqrt{K/n})}$ when K is large enough. Therefore, we have

$$\frac{\beta}{2} - 3(1+c\alpha)(C_q^m+1)L\beta^2\gamma = \frac{1-6(1+c\alpha)L\gamma}{2(C_q^m+1)} \leq \frac{1}{4(C_q^m+1)}. \tag{A.29}$$

From Theorem 3.4.2, we derive

$$\begin{aligned}
&\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla f(\hat{\mathbf{x}}^k)\|^2 \\
\leq&\frac{4(C_q^m+1)(\mathbb{E}\Lambda^1 - \mathbb{E}\Lambda^{K+1})}{\gamma K} + \frac{12(1+nc\alpha)L\sigma^2\gamma}{n} \\
\leq&48L(C_q^m+1)(1+c\alpha)(\mathbb{E}\Lambda^1 - \mathbb{E}\Lambda^{K+1})(\frac{1}{K} + \frac{1}{\sqrt{nK}}) + \frac{(1+nc\alpha)\sigma^2}{(1+c\alpha)}\frac{1}{\sqrt{nK}},
\end{aligned} \tag{A.30}$$

which completes the proof.

□

APPENDIX B

SUPPLEMENTARY OF CHAPTER 4

B.1 LEAD: Smooth Case of Prox-LEAD

Algorithm B.1: LEAD

Input: Stepsize η, parameter $(\alpha,\ \gamma)$, $\mathbf{X}^0,\ \mathbf{H}^1,\ \mathbf{D}^1 = (\mathbf{I}-\mathbf{W})\mathbf{Z}$ for any $\mathbf{Z}$
Output: $\mathbf{X}^K$ or $1/n\sum_{i=1}^n \mathbf{X}_i^K$

```
$\mathbf{H}_w^1 = \mathbf{W}\mathbf{H}^1$
$\mathbf{X}^1 = \mathbf{X}^0 - \eta\nabla\mathbf{F}(\mathbf{X}^0;\xi^0)$
for $k = 1, 2, \cdots, K-1$ do
    $\mathbf{Y}^k = \mathbf{X}^k - \eta\nabla\mathbf{F}(\mathbf{X}^k;\xi^k) - \eta\mathbf{D}^k$
    $\hat{\mathbf{Y}}^k, \hat{\mathbf{Y}}_w^k, \mathbf{H}^{k+1}, \mathbf{H}_w^{k+1} = \textsc{Comm}(\mathbf{Y}^k, \mathbf{H}^k, \mathbf{H}_w^k)$
    $\mathbf{D}^{k+1} = \mathbf{D}^k + \frac{\gamma}{2\eta}(\hat{\mathbf{Y}}^k - \hat{\mathbf{Y}}_w^k)$
    $\mathbf{X}^{k+1} = \mathbf{X}^k - \eta\nabla\mathbf{F}(\mathbf{X}^k;\xi^k) - \eta\mathbf{D}^{k+1}$
end for
procedure $\textsc{Comm}(\mathbf{Y}, \mathbf{H}, \mathbf{H}_w)$
    $\mathbf{Q} = \textsc{Compress}\mathbf{Y} - \mathbf{H}$
    $\hat{\mathbf{Y}} = \mathbf{H} + \mathbf{Q}$
    $\hat{\mathbf{Y}}_w = \mathbf{H}_w + \mathbf{W}\mathbf{Q}$
    $\mathbf{H} = (1-\alpha)\mathbf{H} + \alpha\hat{\mathbf{Y}}$
    $\mathbf{H}_w = (1-\alpha)\mathbf{H}_w + \alpha\hat{\mathbf{Y}}_w$
    Return: $\hat{\mathbf{Y}}, \hat{\mathbf{Y}}_w, \mathbf{H}, \mathbf{H}_w$
end procedure
```

B.2 Connection with Existing Algorithms

In Section 4.2, we have discussed the motivation of LEAD and Prox-LEAD. We now turn to the relation between LEAD and some other existing algorithms from the perspective of the dual problem. We look at the problem (4.1) without the non-smooth regularizer, i.e., $\mathbf{R}(\mathbf{X}) = \mathbf{0}$. Consider the Fenchel conjugate of $\mathbf{F}$ defined as $\mathbf{F}^*(\mathbf{Y}) = \sup_{\mathbf{X}\in\mathbb{R}^{n\times p}} \mathbf{Y}^\top\mathbf{X} - \mathbf{F}(\mathbf{X})$, then we obtain the dual problem as

$$\min_{\mathbf{S}\in\mathbb{R}^{n\times p}} \mathbf{F}^*(-\sqrt{\mathbf{I}-\mathbf{W}}\mathbf{S}).$$

If we apply the gradient descent method to the above problem, we need to evaluate the gradient $-\sqrt{\mathbf{I}-\mathbf{W}}\nabla\mathbf{F}^*(-\sqrt{\mathbf{I}-\mathbf{W}}\mathbf{S})$. Let $\mathbf{D} = \sqrt{\mathbf{I}-\mathbf{W}}\mathbf{S}$, then the iteration follows

$$\mathbf{D}^{k+1} = \mathbf{D}^k + \theta(\mathbf{I}-\mathbf{W})\nabla\mathbf{F}^*(-\mathbf{D}^k).$$

When the gradient of the dual function is available, the communication proceeds after the gradient evaluation. If we compress the only communication step, the algorithm leads to Option A in [81] with the single exception that the quantization procedure is slightly different. Since $\mathbf{D}$ belongs to the dual space of the original optimized variable, we derive the solution of the primal problem via the relation

$$\mathbf{X}^{k+1} = \nabla \mathbf{F}^*(-\mathbf{D}^k) \tag{B.1}$$

and the linear convergence is guaranteed under the strongly convex assumption on $\mathbf{F}$.

In most cases, it is difficult to evaluate the conjugate function, so we rewrite the relation (B.1) into the following minimization problem

$$\mathbf{X}^{k+1} = \operatorname*{arg\,min}_{\mathbf{X} \in \mathbb{R}^{n \times p}} \mathbf{F}(\mathbf{X}) + \langle \mathbf{D}^k, \mathbf{X} \rangle,$$

and get an inexact estimate using the gradient method. Applying one step of gradient descent method to the subproblem and inserting the update into the original dual iterations, we get the following primal-dual iteration

$$\left| \begin{aligned} &\mathbf{X}^{k+1} = \mathbf{X}^k - \eta \nabla \mathbf{F}(\mathbf{X}^k) - \eta \mathbf{D}^k, \\ &\mathbf{D}^{k+1} = \mathbf{D}^k + \theta(\mathbf{I} - \mathbf{W})\mathbf{X}^{k+1}, \end{aligned} \right.$$

where η, θ are stepsizes for primal and dual update respectively. It can be shown that this iteration is a special case of the incremental primal-dual gradient method (PDGM) in [92] and the linear convergence rate can be guaranteed. Furthermore, when the communication of $\mathbf{X}^{k+1}$ in the dual update is conducted by the compression procedure (and the stochastic gradient estimation is involved), the algorithm recovers Option B (Option C) of LessBit in [81].

If we proceed one more gradient descent step in the subproblem, we get

$$\left| \begin{aligned} &\mathbf{X}^{k+1} = \mathbf{X}^k - \eta \nabla \mathbf{F}(\mathbf{X}^k) - \eta \mathbf{D}^k, \\ &\overline{\mathbf{X}}^{k+1} = \mathbf{X}^{k+1} - \eta \nabla \mathbf{F}(\mathbf{X}^{k+1}) - \eta \mathbf{D}^k, \\ &\mathbf{D}^{k+1} = \mathbf{D}^k + \theta(\mathbf{I} - \mathbf{W})\overline{\mathbf{X}}^{k+1}. \end{aligned} \right.$$

Algorithm	Non-smooth R	∇F	Compression	Convergence complexity
Dual Gradient Descent	✗	✗	✗	$\widetilde{\mathcal{O}}(\kappa_f\kappa_g)$
LessBit-Option A [81]	✗	✗	✓	$\widetilde{\mathcal{O}}(C+\kappa_f\kappa_g+C\kappa_f\widetilde{\kappa_g})$
PDGM [92]	✗	✓	✗	$\widetilde{\mathcal{O}}(\kappa_f+\kappa_f\kappa_g)$
LessBit-Option B [81]	✗	✓	✓	$\widetilde{\mathcal{O}}(C+\kappa_f\kappa_g+C\kappa_f\widetilde{\kappa_g})$
NIDS [31]	✗	✓	✗	$\widetilde{\mathcal{O}}(\kappa_f+\kappa_g)$
LEAD [33]	✗	✓	✓	$\widetilde{\mathcal{O}}((1+C)(\kappa_f+\kappa_g)+C\kappa_f\kappa_g)$
PUDA [106]	✓	✓	✗	$\widetilde{\mathcal{O}}(\kappa_f+\kappa_g)$
Prox-LEAD **Algorithm 4.1**	✓	✓	✓	$\widetilde{\mathcal{O}}(C+\kappa_f+\kappa_g+C\kappa_f\kappa_g)$

Table B.1: The comparison of algorithms mentioned in Section B.2; $\kappa_f := \frac{L}{\mu}, \kappa_g := \frac{\lambda_{\max}(\mathbf{I}-\mathbf{W})}{\lambda_{\min}(\mathbf{I}-\mathbf{W})}$ and $\widetilde{\kappa_g} := \frac{\max_{(i,j)\in\mathcal{E}} w_{ij}}{\lambda_{\min}(\mathbf{I}-\mathbf{W})}$ for nonnegative $\mathbf{W}$.

The addition of the step does not increase the computation of the gradient $\nabla\mathbf{F}$ because it can be reused in the next iteration. So, we switch the order of the iteration and derive

$$\left\lfloor \begin{aligned} \overline{\mathbf{X}}^{k+1} &= \mathbf{X}^k - \eta\nabla\mathbf{F}(\mathbf{X}^k) - \eta\mathbf{D}^k, \\ \mathbf{D}^{k+1} &= \mathbf{D}^k + \theta(\mathbf{I}-\mathbf{W})\overline{\mathbf{X}}^{k+1}, \\ \mathbf{X}^{k+1} &= \mathbf{X}^k - \eta\nabla\mathbf{F}(\mathbf{X}^k) - \eta\mathbf{D}^{k+1}. \end{aligned} \right.$$

By setting $\theta = 1$, the above algorithm recovers NIDS of the smooth problems in [2, 31]. It has been shown that the additional step in NIDS improves the linear convergence rate of the previous two algorithms in terms of the condition numbers of the objective function and the network. The detailed comparison is listed in Table B.1.

Compared to LEAD, Prox-LEAD improves the complexity by reducing $\widetilde{\mathcal{O}}(C(\kappa_f+\kappa_g))$ to $\widetilde{\mathcal{O}}(C)$. LessBit-Option B has complexity $\widetilde{\mathcal{O}}(C+\kappa_f\kappa_g+C\kappa_f\widetilde{\kappa_g})$, which has a better entangled term involving C since $\widetilde{\kappa_g}$ is less than κ_g. However, for all $C \in [0, \kappa_g/(\kappa_g-\widetilde{\kappa_g}))$, the complexity of Prox-LEAD is always better than the one of LessBit-Option B.

B.3 Proof of Theorem 4.4.2

Proof. **Linear convergence.** The following proof is applicable to the general choice of $\{p_{ij}\}$ and for simplicity, we only consider the uniform sampling case with $p_{ij} = \frac{1}{m}$.

Lemma 4.4.2 still holds with different $\mathbf{G}^k = [\mathbf{g}_1^k, \cdots, \mathbf{g}_n^k]^\top$ in procedure SGO, then we focus on the following term

$$
\begin{aligned}
&\mathbb{E}\|\mathbf{X}^k - \mathbf{X}^* - \eta\mathbf{G}^k + \eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2 \\
&= \sum_{i=1}^{n} \mathbb{E}\left\|\mathbf{x}_i^k - \mathbf{x}^* - \eta\Big(\frac{\nabla f_{il}(\mathbf{x}_i^k) - \nabla f_{il}(\tilde{\mathbf{x}}_i^k)}{mp_{il}} + \nabla f_i(\tilde{\mathbf{x}}_i^k) - \nabla f_i(\mathbf{x}^*)\Big)\right\|^2 \\
&= \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\sum_{i=1}^{n}\mathbb{E}\left\langle \mathbf{x}_i^k - \mathbf{x}^*, \frac{\nabla f_{il}(\mathbf{x}_i^k) - \nabla f_{il}(\tilde{\mathbf{x}}_i^k)}{mp_{il}} + \nabla f_i(\tilde{\mathbf{x}}_i^k) - \nabla f_i(\mathbf{x}^*)\right\rangle \\
&\quad + \eta^2\sum_{i=1}^{n}\mathbb{E}\left\|\frac{\nabla f_{il}(\mathbf{x}_i^k) - \nabla f_{il}(\tilde{\mathbf{x}}_i^k)}{mp_{il}} + \nabla f_i(\tilde{\mathbf{x}}_i^k) - \nabla f_i(\mathbf{x}^*)\right\|^2 \\
&= \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\sum_{i=1}^{n}\langle \mathbf{x}_i^k - \mathbf{x}^*, \nabla f_i(\mathbf{x}_i^k) - \nabla f_i(\mathbf{x}^*)\rangle \\
&\quad + \eta^2\sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}\left\|\frac{\nabla f_{ij}(\mathbf{x}_i^k) - \nabla f_{ij}(\tilde{\mathbf{x}}_i^k)}{mp_{ij}} + \nabla f_i(\tilde{\mathbf{x}}_i^k) - \nabla f_i(\mathbf{x}^*)\right\|^2 \\
&= \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\sum_{i=1}^{n}\langle \mathbf{x}_i^k - \mathbf{x}^*, \nabla f_i(\mathbf{x}_i^k)\rangle + 2\eta\sum_{i=1}^{n}(f_i(\mathbf{x}_i^k) - f_i(\mathbf{x}^*) - \mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*)) \\
&\quad + \eta^2\sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}\left\|\frac{\nabla f_{ij}(\mathbf{x}_i^k) - \nabla f_{ij}(\mathbf{x}^*)}{mp_{ij}} + \frac{f_{ij}(\mathbf{x}^*) - \nabla f_{ij}(\tilde{\mathbf{x}}_i^k)}{mp_{ij}} + \nabla f_i(\tilde{\mathbf{x}}_i^k) - \nabla f_i(\mathbf{x}^*)\right\|^2 \\
&\le (1 - \mu\eta)\|\mathbf{X}^k - \mathbf{X}^*\| - 2\eta\sum_{i=1}^{n}\mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) + 2\eta^2\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{1}{m^2 p_{ij}}\|\nabla f_{ij}(\mathbf{x}_i^k) - \nabla f_{ij}(\mathbf{x}^*)\|^2 \\
&\quad + 2\eta^2\sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}\left\|\frac{\nabla f_{ij}(\mathbf{x}^*) - \nabla f_{ij}(\tilde{\mathbf{x}}_i^k)}{mp_{ij}} + \nabla f_i(\tilde{\mathbf{x}}_i^k) - \nabla f_i(\mathbf{x}^*)\right\|^2,
\end{aligned}
$$

where the inequality uses the strong convexity of f_{ij} in Assumption 4.3.2 and $(a+b)^2 \le 2a^2 + 2b^2$.

Let u_i be the random variable taking values in $\left\{\frac{1}{mp_{il}}(\nabla f_{il}(\tilde{\mathbf{x}}_i^k) - \nabla f_{il}(\mathbf{x}^*)) : l \in [m]\right\}$ with distribution $\mathcal{P}_i = \{p_{il} : l \in [m]\}$, then the last term is actually the summation of the variance of u_is due to

$$\mathbb{E}u_i = \sum_{j=1}^{m} \frac{p_{ij}}{mp_{ij}} (\nabla f_{ij}(\tilde{\mathbf{x}}_i^*) - \nabla f_{ij}(\mathbf{x}^*)) = \nabla f_i(\tilde{\mathbf{x}}_i^k) - \nabla f_i(\mathbf{x}^*).$$

Applying the inequality $\mathbb{E}\|u_i - \mathbb{E}u_i\|^2 \leq \mathbb{E}\|u_i\|^2$ to the last term, we get

$$\begin{aligned}
&\mathbb{E}\|\mathbf{X}^k - \mathbf{X}^* - \eta\mathbf{G}^k + \eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2 \\
\leq\ & (1-\mu\eta)\|\mathbf{X}^k - \mathbf{X}^*\| - 2\eta\sum_{i=1}^{n} \mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) + \frac{4\eta^2 L}{mp_{\min}} \sum_{i=1}^{n} \mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) \\
&+ 2\eta^2 \sum_{i=1}^{n}\sum_{j=1}^{m} \frac{1}{m^2 p_{ij}} \|\nabla f_{ij}(\tilde{\mathbf{x}}_i^k) - \nabla f_{ij}(\mathbf{x}^*)\|^2 \\
\leq\ & (1-\mu\eta)\|\mathbf{X}^k - \mathbf{X}^*\| - 2\eta\sum_{i=1}^{n} \mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) \\
&+ \frac{4\eta^2 L}{mp_{\min}} \sum_{i=1}^{n} \mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) + \frac{4\eta^2 L}{mp_{\min}} \sum_{i=1}^{n} \mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^k, \mathbf{x}^*),
\end{aligned}$$

where $p_{\min} \coloneqq \min_{i,j}\{p_{ij}\}$ and the last inequality uses the Lipschitz smoothness of f_{ij} in Assumption 4.3.2.

From the update of $\tilde{\mathbf{x}}_i^{k+1}$, we have

$$\mathbb{E}\tilde{\mathbf{x}}_i^{k+1} = p\mathbf{x}_i^k + (1-p)\tilde{\mathbf{x}}_i^k.$$

Hence

$$\mathbb{E}\mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^{k+1}, \mathbf{x}^*) = p\mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) + (1-p)\mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^k, \mathbf{x}^*).$$

Combine them with the inequality (4.12) and (4.11), after taking the total expectation, we get

$$\begin{aligned}
&\mathbb{E}\Phi^{k+1} + \tilde{c}\sum_{i=1}^{n} \mathbb{E}\mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^{k+1}, \mathbf{x}^*) \\
\leq\ & (1-\eta\mu)\mathbb{E}\|\mathbf{Z}^k - \mathbf{Z}^*\|_{\mathbf{P}}^2 - \left(2\eta - \frac{4\eta^2 L}{mp_{\min}} - \tilde{c}p\right)\sum_{i=1}^{n} \mathbb{E}\mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) \\
&+ \frac{4\eta^2 L}{mp_{\min}} \sum_{i=1}^{n} \mathbb{E}\mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^k, \mathbf{x}^*) + (1-\alpha)CM\mathbb{E}\|\mathbf{H}^k - \mathbf{Z}^*\|^2 \\
&+ \frac{2\eta^2}{\gamma}\Big(1 - \frac{\gamma}{2}\lambda_{\min}(\mathbf{I}-\mathbf{W})\Big)\mathbb{E}\|\mathbf{D}^k - \mathbf{D}^*\|_{\mathbf{Q}}^2 \\
&+ (1-\eta\mu)\mathbb{E}\|\hat{\mathbf{Z}}^k - \mathbf{Z}^k\|_{(\mathbf{I}-\mathbf{P})^2}^2 + \tilde{c}(1-p)\sum_{i=1}^{n} \mathbb{E}\mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^k, \mathbf{x}^*),
\end{aligned}$$

where $\tilde{c} = \frac{8\eta^2 L}{pmp_{\min}}$.

By the choice of η and $\{p_{ij}\}$, we have

$$\frac{4\eta^2 L}{mp_{\min}} + \tilde{c}p = \frac{1}{18L}(2+4) = \frac{1}{3L} = 2\eta.$$

Therefore,

$$\begin{aligned}\mathbb{E}\Phi^{k+1} + \tilde{c}\sum_{i=1}^{n}\mathbb{E}\mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^{k+1}, \mathbf{x}^*) \leq (1-\eta\mu)\mathbb{E}\|\mathbf{Z}^k - \mathbf{Z}^*\|_{\mathbf{P}}^2 + (1-\alpha)CM\mathbb{E}\|\mathbf{H}^k - \mathbf{Z}^*\|^2 \\ + \frac{2\eta^2}{\gamma}\Big(1 - \frac{\gamma}{2}\lambda_{\min}(\mathbf{I}-\mathbf{W})\Big)\mathbb{E}\|\mathbf{D}^k - \mathbf{D}^*\|_{\mathbf{Q}}^2 \\ + (1-\eta\mu)\frac{\gamma\lambda_{\max}(\mathbf{I}-\mathbf{W})}{2}\mathbb{E}\|\hat{\mathbf{Z}}^k - \mathbf{Z}^k\|_{\mathbf{I}-\mathbf{P}}^2 \\ + \tilde{c}\left(1 - p + \frac{4\eta^2 L}{\tilde{c}mp_{\min}}\right)\sum_{i=1}^{n}\mathbb{E}\mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^k, \mathbf{x}^*).\end{aligned}$$

Note that

$$1 - p + \frac{4\eta^2 L}{\tilde{c}mp_{\min}} = 1 - \frac{p}{2}, \quad \tilde{c} = \frac{2}{9pL},$$

then we get

$$\mathbb{E}\Phi^{k+1} + \frac{2}{9pL}\sum_{i=1}^{n}\mathbb{E}\mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^{k+1}, \mathbf{x}^*) \leq \tilde{\rho}\Big(\mathbb{E}\Phi^k + \frac{2}{9pL}\sum_{i=1}^{n}\mathbb{E}\mathrm{V}_{f_i}(\tilde{\mathbf{x}}_i^k, \mathbf{x}^*)\Big),$$

where

$$\tilde{\rho} := \max\left\{1 - \frac{\eta\mu}{2-\eta\mu}, 1 - \frac{\gamma}{2}\lambda_{\min}(\mathbf{I}-\mathbf{W}), 1-\alpha, (1-\eta\mu)\frac{\gamma\lambda_{\max}(\mathbf{I}-\mathbf{W})}{2}, 1 - \frac{p}{2}\right\}.$$

Complexity. The linear convergence requires

$$\begin{aligned}\alpha &< \frac{1}{1+C}, \\ \gamma &\leq \frac{1}{\lambda_{\max}(\mathbf{I}-\mathbf{W})}\frac{2\Delta(\alpha)}{\Delta(\alpha) + 24\alpha C\kappa_f}.\end{aligned}$$

Take $\alpha = \frac{1}{2(1+C)}$, then $\Delta(\alpha) = \frac{\alpha}{2} = \frac{1}{4(1+C)}$ and

$$\gamma \leq \frac{1}{\lambda_{\max}(\mathbf{I}-\mathbf{W})}\frac{2}{1 + 48C\kappa_f}.$$

By taking γ equal to the upper bound and plugging η, α, γ into $\tilde{\rho}$, we get

$$\tilde{\rho} \leq 1 - \left(\max\Big\{12\kappa_f - 1, \kappa_g + 48C\kappa_f\kappa_g, 1+C, 6\kappa_f, \frac{2}{p}\Big\}\right)^{-1}$$

and the proof is complete. □

B.4 Proof of Theorem 4.4.3

Proof. **Linear convergence.** The following proof can be adapted to to show the convergence with the general $\{p_{ij}\}$ while for simplicity, we focus on uniform sampling case.

We start from the gradient term of the key inequality in Lemma 4.4.2 and replace $\nabla \mathbf{F}(\mathbf{X}^k, \xi^k)$ by $\mathbf{G}^k = [\mathbf{g}_1^k, \cdots, \mathbf{g}_n^k]^\top$,

$$
\begin{aligned}
&\mathbb{E}\|\mathbf{X}^k - \mathbf{X}^* - \eta\mathbf{G}^k + \eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2 \\
&= \sum_{i=1}^n \mathbb{E}\left\|\mathbf{x}_i^k - \mathbf{x}^* - \eta\Big(\frac{\nabla f_{il}(\mathbf{x}_i^k) - \nabla f_{il}(\tilde{\mathbf{x}}_{il}^k)}{mp_{il}} + \frac{\sum_{j=1}^m \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k) - \nabla f_{ij}(\mathbf{x}^*)}{m}\Big)\right\|^2 \\
&= \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\sum_{i=1}^n \mathbb{E}\left\langle \mathbf{x}_i^k - \mathbf{x}^*, \frac{\nabla f_{il}(\mathbf{x}_i^k) - \nabla f_{il}(\tilde{\mathbf{x}}_{il}^k)}{mp_{il}} + \frac{\sum_{j=1}^m \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k) - \nabla f_{ij}(\mathbf{x}^*)}{m}\right\rangle \\
&\quad + \eta^2\sum_{i=1}^n \mathbb{E}\left\|\frac{\nabla f_{il}(\mathbf{x}_i^k) - \nabla f_{il}(\tilde{\mathbf{x}}_{il}^k)}{mp_{il}} + \frac{\sum_{j=1}^m \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k) - \nabla f_{ij}(\mathbf{x}^*)}{m}\right\|^2 \\
&= \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\sum_{i=1}^n \langle \mathbf{x}_i^k - \mathbf{x}^*, \nabla f_i(\mathbf{x}_i^k) - \nabla f_i(\mathbf{x}^*)\rangle \\
&\quad + \eta^2\sum_{i=1}^n\sum_{j=1}^m p_{ij}\left\|\frac{\nabla f_{ij}(\mathbf{x}_i^k) - \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k)}{mp_{ij}} + \frac{\sum_{j=1}^m \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k) - \nabla f_{ij}(\mathbf{x}^*)}{m}\right\|^2 \\
&= \|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\sum_{i=1}^n \langle \mathbf{x}_i^k - \mathbf{x}^*, \nabla f_i(\mathbf{x}_i^k)\rangle + 2\eta\sum_{i=1}^n (f_i(\mathbf{x}_i^k) - f_i(\mathbf{x}^*) - \mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*)) + \\
&\eta^2\sum_{i=1}^n\sum_{j=1}^m p_{ij}\left\|\frac{\nabla f_{ij}(\mathbf{x}_i^k) - \nabla f_{ij}(\mathbf{x}^*)}{mp_{ij}} + \frac{f_{ij}(\mathbf{x}^*) - \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k)}{mp_{ij}} + \frac{\sum_{j=1}^m \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k) - \nabla f_{ij}(\mathbf{x}^*)}{m}\right\|^2.
\end{aligned}
$$

Using the strong convexity of f_{ij} in Assumption 4.3.2 and applying $(a+b)^2 \leq 2a^2 + 2b^2$ to the last term, we have

$$
\begin{aligned}
&\mathbb{E}\|\mathbf{X}^k - \mathbf{X}^* - \eta\mathbf{G}^k + \eta\nabla\mathbf{F}(\mathbf{X}^*)\|^2 \\
&\leq (1 - \mu\eta)\|\mathbf{X}^k - \mathbf{X}^*\| - 2\eta\sum_{i=1}^n \mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) + 2\eta^2\sum_{i=1}^n\sum_{j=1}^m \frac{1}{m^2 p_{ij}}\|\nabla f_{ij}(\mathbf{x}_i^k) - \nabla f_{ij}(\mathbf{x}^*)\|^2 \\
&\quad + 2\eta^2\sum_{i=1}^n\sum_{j=1}^m p_{ij}\left\|\frac{\nabla f_{ij}(\mathbf{x}^*) - \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k)}{mp_{ij}} + \frac{\sum_{j=1}^m \nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k) - \nabla f_{ij}(\mathbf{x}^*)}{m}\right\|^2.
\end{aligned}
$$

Let u_i be the random variable taking values in $\left\{\frac{1}{mp_{il}}(\nabla f_{il}(\tilde{\mathbf{x}}_{il}^k) - \nabla f_{il}(\tilde{\mathbf{x}}^*)) : l \in [m]\right\}$ with distribution $\mathcal{P}_i = \{p_{il} : l \in [m]\}$, then the last term of the above inequality can be upper bounded by

$$
\begin{aligned}
2\eta^2 \sum_{i=1}^{n} \mathbb{E}\|u_i - \mathbb{E}u_i\|^2 &\leq 2\eta^2 \sum_{i=1}^{n} \mathbb{E}\|u_i\|^2 \\
&= 2\eta^2 \sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij} \left\| \frac{1}{mp_{ij}} (\nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k) - \nabla f_{ij}(\tilde{\mathbf{x}}^*)) \right\|^2.
\end{aligned}
$$

Combining the inequality with the upper bound, we get

$$
\begin{aligned}
&\mathbb{E}\|\mathbf{X}^k - \mathbf{X}^* - \eta \mathbf{G}^k + \eta \nabla \mathbf{F}(\mathbf{X}^*)\|^2 \\
&\leq (1-\mu\eta)\|\mathbf{X}^k - \mathbf{X}^*\| - 2\eta \sum_{i=1}^{n} \mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) + \frac{4\eta^2 L}{m^2} \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{1}{p_{ij}} \mathrm{V}_{f_{ij}}(\mathbf{x}_i^k, \mathbf{x}^*) \\
&\quad + 2\eta^2 \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{1}{m^2 p_{ij}} \|\nabla f_{ij}(\tilde{\mathbf{x}}_{ij}^k) - \nabla f_{ij}(\mathbf{x}^*)\|^2 \\
&\leq (1-\mu\eta)\|\mathbf{X}^k - \mathbf{X}^*\| - 2\eta \sum_{i=1}^{n} \mathrm{V}_{f_i}(\mathbf{x}_i^k, \mathbf{x}^*) + \frac{4\eta^2 L}{m^2} \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{1}{p_{ij}} \mathrm{V}_{f_{ij}}(\mathbf{x}_i^k, \mathbf{x}^*) \\
&\quad + \frac{4\eta^2 L}{m^2} \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{1}{p_{ij}} \mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^k, \mathbf{x}^*),
\end{aligned}
$$

where the last inequality uses the Lipschitz smoothness of f_{ij} in Assumption 4.3.2.

From the update of $\tilde{\mathbf{x}}_{ij}$, we have

$$
\mathbb{E}\mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^{k+1}, \mathbf{x}^*) = p_{ij}\mathrm{V}_{f_{ij}}(\mathbf{x}_i^k, \mathbf{x}^*) + (1-p_{ij})\mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^k, \mathbf{x}^*).
$$

Similar to the proof of Theorem 4.4.2, in total expectation, we can get

$$
\begin{aligned}
&\mathbb{E}\Phi^{k+1} + \tilde{c}\sum_{i=1}^{n}\sum_{j=1}^{m}\mathbb{E}\mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^{k+1}, \mathbf{x}^*) \\
\leq\ & (1-\eta\mu)\mathbb{E}\|\mathbf{X}^k - \mathbf{X}^*\|^2 - 2\eta\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{1}{m}\mathbb{E}\mathrm{V}_{f_{ij}}(\mathbf{x}_i^k, \mathbf{x}^*) \\
&+ \frac{4\eta^2 L}{m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{1}{p_{ij}}\mathbb{E}\mathrm{V}_{f_{ij}}(\mathbf{x}_i^k, \mathbf{x}^*) + \tilde{c}\sum_{i=1}^{n}\sum_{j=1}^{m}p_{ij}\mathbb{E}\mathrm{V}_{f_{ij}}(\mathbf{x}_i^k, \mathbf{x}^*) \\
&+ \tilde{c}\sum_{i=1}^{n}\sum_{j=1}^{m}(1-p_{ij})\mathbb{E}\mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^k, \mathbf{x}^*) + \frac{4\eta^2 L}{m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{1}{p_{ij}}\mathbb{E}\mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^k, \mathbf{x}^*) \\
&+ \frac{2\eta^2}{\gamma}\left(1 - \frac{\gamma}{2}\lambda_{\min}(\mathbf{I}-\mathbf{W})\right)\mathbb{E}\|\mathbf{D}^k - \mathbf{D}^*\|_{\mathbf{Q}}^2 + (1-\alpha)CM\mathbb{E}\|\mathbf{H}^k - \mathbf{Z}^*\|^2 \\
\leq\ & (1-\eta\mu)\mathbb{E}\|\mathbf{Z}^k - \mathbf{Z}^*\|_{\mathbf{P}}^2 + \frac{2\eta^2}{\gamma}\Big(1 - \frac{\gamma}{2}\lambda_{\min}(\mathbf{I}-\mathbf{W})\Big)\mathbb{E}\|\mathbf{D}^k - \mathbf{D}^*\|_{\mathbf{Q}}^2 \\
&+ \tilde{c}\sum_{i=1}^{n}\sum_{j=1}^{m}(1-p_{ij})\mathbb{E}\mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^k, \mathbf{x}^*) + \frac{4\eta^2 L}{m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{1}{p_{ij}}\mathbb{E}\mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^k, \mathbf{x}^*) \\
&+ (1-\alpha)CM\mathbb{E}\|\mathbf{H}^k - \mathbf{Z}^*\|^2 + \frac{(1-\eta\mu)\gamma\lambda_{\max}(\mathbf{I}-\mathbf{W})}{2}\mathbb{E}\|\hat{\mathbf{Z}}^k - \mathbf{Z}^*\|_{\mathbf{I}-\mathbf{P}}^2,
\end{aligned}
$$

where the last inequality is guaranteed by

$$\frac{2\eta}{m} - \frac{4\eta^2 L}{m^2 p_{ij}} - \tilde{c}p_{ij} \geq 0.$$

Define $p_{\min} = \min_{i,j}\{p_{ij}\}$ and take $\tilde{c} = \frac{8\eta^2 L}{m^2 p_{\min}^2}$, then by the choice of $\{p_{ij}\}$ and η, the above condition is satisfied due to

$$\frac{2\eta}{m} - \frac{4\eta^2 L}{m^2 p_{ij}} - \tilde{c}p_{ij} = \frac{1}{3Lm} - \frac{1}{9Lm} - \frac{2}{9Lm} = 0.$$

Note that

$$\tilde{c}(1-p_{ij}) + \frac{4\eta^2 L}{m^2 p_{ij}} = \tilde{c}\left(1 - p_{ij} + \frac{\frac{4\eta^2 L}{m^2 p_{ij}}}{\tilde{c}}\right) = \tilde{c}\Big(1 - \frac{1}{2m}\Big).$$

Therefore,

$$\mathbb{E}\Phi^{k+1} + \frac{2}{9L}\sum_{i=1}^{n}\sum_{j=1}^{m}\mathbb{E}\mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^{k+1}, \mathbf{x}^*) \leq \tilde{\rho}\Big(\mathbb{E}\Phi^k + \frac{2}{9L}\sum_{i=1}^{n}\sum_{j=1}^{m}\mathbb{E}\mathrm{V}_{f_{ij}}(\tilde{\mathbf{x}}_{ij}^k, \mathbf{x}^*)\Big),$$

where

$$\tilde{\rho} := \max\left\{1 - \frac{\eta\mu}{2-\eta\mu}, 1 - \frac{\gamma}{2}\lambda_{\min}(\mathbf{I}-\mathbf{W}), 1-\alpha, (1-\eta\mu)\frac{\gamma\lambda_{\max}(\mathbf{I}-\mathbf{W})}{2}, 1 - \frac{1}{2m}\right\}.$$

Complexity. The complexity analysis is identical to that in Theorem 4.4.2 with the single exception that we replace p by m^{-1} so we omit it here. □

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] W. Shi, Q. Ling, G. Wu, and W. Yin. EXTRA: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.

[2] Zhi Li, Wei Shi, and Ming Yan. A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates. *IEEE Transactions on Signal Processing*, 67(17):4494–4506, 2019.

[3] Zhi Li and Ming Yan. New convergence analysis of a primal-dual algorithm with large stepsizes. *Advances in Computational Mathematics*, 47(1):1–20, 2021.

[4] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In Yves Lechevallier and Gilbert Saporta, editors, *Proceedings of COMPSTAT'2010*, pages 177–186, Heidelberg, 2010. Physica-Verlag HD.

[5] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation*, OSDI'16, pages 265–283, 2016.

[6] Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, and Zheng Zhang. MXNet: A flexible and efficient machine learning library for heterogeneous distributed systems. *CoRR*, abs/1512.01274, 2015.

[7] Mu Li, David G. Andersen, Jun Woo Park, Alexander J. Smola, Amr Ahmed, Vanja Josifovski, James Long, Eugene J. Shekita, and Bor-Yiing Su. Scaling distributed machine learning with the parameter server. In *Proceedings of the 11th USENIX Conference on Operating Systems Design and Implementation*, OSDI'14, pages 583–598, Berkeley, CA, USA, 2014. USENIX Association.

[8] Yang You, Zhao Zhang, Cho-Jui Hsieh, James Demmel, and Kurt Keutzer. ImageNet training in minutes. In *Proceedings of the 47th International Conference on Parallel Processing*, ICPP 2018, pages 1:1–1:10, New York, NY, USA, 2018. ACM.

[9] Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao, Marc'Aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, and Andrew Y. Ng. Large scale distributed deep networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, pages 1223–1231, USA, 2012.

[10] Xiangru Lian, Yijun Huang, Yuncheng Li, and Ji Liu. Asynchronous parallel stochastic gradient for nonconvex optimization. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2737–2745. Curran Associates, Inc., 2015.

[11] Martin Zinkevich, Markus Weimer, Lihong Li, and Alex J. Smola. Parallelized stochastic gradient descent. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 2595–2603. Curran Associates, Inc., 2010.

[12] Frank Seide, Hao Fu, Jasha Droppo, Gang Li, and Dong Yu. 1-bit stochastic gradient descent and application to data-parallel distributed training of speech DNNs. In *Interspeech 2014*, September 2014.

[13] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. QSGD: Communication-efficient SGD via gradient quantization and encoding. In *Advances in Neural Information Processing Systems*, pages 1709–1720, 2017.

[14] Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. signSGD: Compressed optimisation for non-convex problems. In *International Conference on Machine Learning*, pages 560–569. PMLR, 2018.

[15] Sebastian U. Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified SGD with memory. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 4452–4463, USA, 2018. Curran Associates Inc.

[16] Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Urban Stich, and Martin Jaggi. Error feedback fixes SignSGD and other gradient compression schemes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3252–3261. PMLR, 2019.

[17] Konstantin Mishchenko, Eduard Gorbunov, Martin Takáč, and Peter Richtárik. Distributed learning with compressed gradient differences. *arXiv preprint arXiv:1901.09269*, 2019.

[18] Hanlin Tang, Chen Yu, Xiangru Lian, Tong Zhang, and Ji Liu. DoubleSqueeze: Parallel stochastic gradient descent with double-pass error-compensated compression. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 6155–6165, 2019.

[19] Xiaorui Liu, Yao Li, Jiliang Tang, and Ming Yan. A double residual compression algorithm for efficient distributed learning. In *International Conference on Artificial Intelligence and Statistics*, pages 133–143. PMLR, 2020.

[20] Hanlin Tang, Shaoduo Gan, Ce Zhang, Tong Zhang, and Ji Liu. Communication compression for decentralized training. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 7663–7673, 2018.

[21] Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, and Ramtin Pedarsani. An exact quantized decentralized gradient descent algorithm. *IEEE Transactions on Signal Processing*, 67(19):4934–4947, 2019.

[22] Amirhossein Reisizadeh, Hossein Taheri, Aryan Mokhtari, Hamed Hassani, and Ramtin Pedarsani. Robust and communication-efficient collaborative learning. In *Advances in Neural Information Processing Systems*, pages 8388–8399, 2019.

[23] Hanlin Tang, Xiangru Lian, Shuang Qiu, Lei Yuan, Ce Zhang, Tong Zhang, and Ji Liu. Deepsqueeze: Decentralization meets error-compensated compression. *CoRR*, abs/1907.07346, 2019.

[24] Anastasia Koloskova, Sebastian U. Stich, and Martin Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication. In *Proceedings of the 36th International Conference on Machine Learning*, pages 3479–3487. PMLR, 2019.

[25] Anastasia Koloskova, Tao Lin, Sebastian U Stich, and Martin Jaggi. Decentralized deep learning with arbitrary communication compression. In *International Conference on Learning Representations*, 2020.

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

[27] Kun Yuan, Qing Ling, and Wotao Yin. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3):1835–1854, 2016.

[28] Xiangru Lian, Ce Zhang, Huan Zhang, Cho-Jui Hsieh, Wei Zhang, and Ji Liu. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. In *Advances in Neural Information Processing Systems*, pages 5330–5340, 2017.

[29] Qing Ling, Wei Shi, Gang Wu, and Alejandro Ribeiro. DLM: Decentralized linearized alternating direction method of multipliers. *IEEE Transactions on Signal Processing*, 63(15):4051–4064, 2015.

[30] Kun Yuan, Wei Xu, and Qing Ling. Can primal methods outperform primal-dual methods in decentralized dynamic optimization? *IEEE Transactions on Signal Processing*, 68:4466–4480, 2020.

[31] Yao Li and Ming Yan. On the linear convergence of two decentralized algorithms. *Journal of Optimization Theory and Applications*, 189(1):271–290, 2021.

[32] Yao Li, Xiaorui Liu, Jiliang Tang, Ming Yan, and Kun Yuan. Decentralized composite optimization with compression. *arXiv preprint arXiv:2108.04448*, 2021.

[33] Xiaorui Liu, Yao Li, Rongrong Wang, Jiliang Tang, and Ming Yan. Linear convergent decentralized optimization with compression. In *International Conference on Learning Representations*, 2021.

[34] Hanlin Tang, Yao Li, Ji Liu, and Ming Yan. ErrorCompensatedX: Error compensation for variance reduced algorithms. *Advances in Neural Information Processing Systems*, 34:18102–18113, 2021.

[35] Yao Li and Ming Yan. On the improved conditions for some primal-dual algorithms. *arXiv preprint arXiv:2201.00139*, 2022.

[36] S Sundhar Ram, Angelia Nedić, and Venugopal V Veeravalli. Distributed stochastic subgradient projection algorithms for convex optimization. *Journal of optimization theory and applications*, 147(3):516–545, 2010.

[37] Angelia Nedic. Asynchronous broadcast-based convex optimization over a network. *IEEE Transactions on Automatic Control*, 56(6):1337–1351, 2010.

[38] D. Jakovetic, J. Xavier, and J. Moura. Fast distributed gradient methods. *IEEE Transactions on Automatic Control*, 59:1131–1146, 2014.

[39] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin. On the linear convergence of the ADMM in decentralized consensus optimization. *IEEE Transactions on Signal Processing*, 62(7):1750–1761, 2014.

[40] Roland Glowinski and A Marroco. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique*, 9(R2):41–76, 1975.

[41] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[42] E. Wei and A. Ozdaglar. On the $O(1/k)$ convergence of asynchronous distributed alternating direction method of multipliers. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 551–554. IEEE, 2013.

[43] Tsung-Hui Chang, Mingyi Hong, and Xiangfeng Wang. Multi-agent distributed optimization via inexact consensus ADMM. *IEEE Transactions on Signal Processing*, 63(2):482–497, 2015.

[44] Mingyi Hong and Tsung-Hui Chang. Stochastic proximal gradient consensus over random networks. *IEEE Transactions on Signal Processing*, 65(11):2933–2948, 2017.

[45] Angelia Nedic, Alex Olshevsky, and Wei Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, 2017.

[46] Guannan Qu and Na Li. Harnessing smoothness to accelerate distributed optimization. *IEEE Transactions on Control of Network Systems*, 5(3):1245–1260, 2017.

[47] Aryan Mokhtari, Wei Shi, Qing Ling, and Alejandro Ribeiro. A decentralized second-order method for dynamic optimization. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 6036–6043. IEEE, 2016.

[48] Minghui Zhu and Sonia Martínez. Discrete-time dynamic average consensus. *Automatica*, 46(2):322–329, 2010.

[49] J. Xu, S. Zhu, Y. Soh, and L. Xie. Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes. In *Proceedings of the 54th IEEE Conference on Decision and Control (CDC)*, pages 2055–2060, 2015.

[50] Paolo Di Lorenzo and Gesualdo Scutari. NEXT: In-network nonconvex optimization. *IEEE Transactions on Signal and Information Processing over Networks*, 2(2):120–136, 2016.

[51] Angelia Nedić, Alex Olshevsky, Wei Shi, and César A Uribe. Geometrically convergent distributed optimization with uncoordinated step-sizes. In *American Control Conference (ACC), 2017*, pages 3950–3955. IEEE, 2017.

[52] Shi Pu, Wei Shi, Jinming Xu, and Angelia Nedić. A push-pull gradient method for distributed optimization in networks. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 3385–3390. IEEE, 2018.

[53] Kun Yuan, Bicheng Ying, Xiaochuan Zhao, and Ali H Sayed. Exact diffusion for distributed optimization and learning—part i: Algorithm development. *IEEE Transactions on Signal Processing*, 67(3):708–723, 2018.

[54] Kun Yuan, Bicheng Ying, Xiaochuan Zhao, and Ali H Sayed. Exact diffusion for distributed optimization and learning—part ii: Convergence analysis. *IEEE Transactions on Signal Processing*, 67(3):724–739, 2018.

[55] Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3027–3036. JMLR, 2017.

[56] César A Uribe, Soomin Lee, Alexander Gasnikov, and Angelia Nedić. A dual approach for optimal algorithms in distributed optimization over networks. In *2020 Information Theory and Applications Workshop (ITA)*, pages 1–37. IEEE, 2020.

[57] Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. A proximal gradient algorithm for decentralized composite optimization. *IEEE Transactions on Signal Processing*, 63(22):6013–6023, 2015.

[58] Sulaiman Alghunaim, Kun Yuan, and Ali H Sayed. A linearly convergent proximal gradient algorithm for decentralized optimization. In *Advances in Neural Information Processing Systems*, pages 2848–2858, 2019.

[59] A. Chen and A. Ozdaglar. A fast distributed proximal-gradient method. In *the 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 601–608, 2012.

[60] A. Nedić and A. Olshevsky. Distributed optimization over time-varying directed graphs. In *The 52nd IEEE Annual Conference on Decision and Control*, pages 6855–6860, 2013.

[61] C. Xi and U. Khan. On the linear convergence of distributed optimization over directed graphs. *arXiv preprint arXiv:1510.02149*, 2015.

[62] J. Zeng and W. Yin. ExtraPush for convex smooth decentralized optimization over directed networks. *Journal of Computational Mathematics, Special Issue on Compressed Sensing, Optimization, and Structured Solutions*, 35(4):381–394, 2017.

[63] Ying Sun, Gesualdo Scutari, and Daniel Palomar. Distributed nonconvex multiagent optimization over time-varying networks. In *2016 50th Asilomar Conference on Signals, Systems and Computers*, pages 788–794. IEEE, 2016.

[64] Qing Ling and Alejandro Ribeiro. Decentralized dynamic optimization through the alternating direction method of multipliers. In *2013 IEEE 14th Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 170–174. IEEE, 2013.

[65] Angelia Nedić and Alex Olshevsky. Stochastic gradient-push for strongly convex functions on time-varying directed graphs. *IEEE Transactions on Automatic Control*, 61(12):3936–3947, 2016.

[66] Angelia Nedić. Distributed optimization over networks. In *Multi-agent Optimization*, pages 1–84. Springer, 2018.

[67] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*, volume 87. Springer Science & Business Media, 2013.

[68] Nikko Strom. Scalable distributed DNN training using commodity GPU cloud computing. In *INTERSPEECH*, pages 1488–1492, 2015.

[69] Jialei Wang, Mladen Kolar, Nathan Srebro, and Tong Zhang. Efficient distributed learning with sparsity. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3636–3645, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

[70] Jianqiao Wangni, Jialei Wang, Ji Liu, and Tong Zhang. Gradient sparsification for communication-efficient distributed optimization. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 1306–1316, USA, 2018. Curran Associates Inc.

[71] Wei Wen, Cong Xu, Feng Yan, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. TernGrad: Ternary gradients to reduce communication in distributed deep learning. In *Advances in neural information processing systems*, pages 1509–1519, 2017.

[72] Jiaxiang Wu, Weidong Huang, Junzhou Huang, and Tong Zhang. Error compensated quantized SGD and its applications to large-scale distributed optimization. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5325–5333, 10–15 Jul 2018.

[73] Alham Fikri Aji and Kenneth Heafield. Sparse communication for distributed gradient descent. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 440–445, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[74] Michael I Jordan, Jason D Lee, and Yun Yang. Communication-efficient distributed statistical inference. *Journal of the American Statistical Association*, 114(526):668–681, 2019.

[75] Lam Nguyen, Phuong Ha Nguyen, Marten Dijk, Peter Richtárik, Katya Scheinberg, and Martin Takác. SGD and Hogwild! Convergence without the bounded gradients assumption. In *International Conference on Machine Learning*, pages 3750–3758. PMLR, 2018.

[76] Robert M. Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtárik. SGD: General analysis and improved rates. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:9090–9112, 2019.

[77] Samuel Horváth, Dmitry Kovalev, Konstantin Mishchenko, Sebastian Stich, and Peter Richtárik. Stochastic distributed learning with gradient quantization and variance reduction. *arXiv preprint arXiv:1904.05115*, 2019.

[78] P. Elias. Universal codeword sets and representations of the integers. *IEEE Transactions on Information Theory*, 21(2):194–203, March 1975.

[79] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.

[80] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[81] Dmitry Kovalev, Anastasia Koloskova, Martin Jaggi, Peter Richtarik, and Sebastian Stich. A linearly convergent algorithm for decentralized optimization: Sending less bits for free! In *International Conference on Artificial Intelligence and Statistics*, pages 4087–4095. PMLR, 2021.

[82] John Tsitsiklis, Dimitri Bertsekas, and Michael Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE transactions on automatic control*, 31(9):803–812, 1986.

[83] Joao FC Mota, Joao MF Xavier, Pedro MQ Aguiar, and Markus Püschel. D-ADMM: A communication-efficient distributed algorithm for separable optimization. *IEEE Transactions on Signal Processing*, 61(10):2718–2723, 2013.

[84] Hanlin Tang, Xiangru Lian, Ming Yan, Ce Zhang, and Ji Liu. D^2: Decentralized training over decentralized data. In *Proceedings of the 35th International Conference on Machine Learning*, pages 4848–4856, 2018.

[85] Gesualdo Scutari and Ying Sun. Distributed nonconvex constrained optimization over time-varying digraphs. *Mathematical Programming*, 176(1):497–544, 2019.

[86] Shi Pu and Angelia Nedić. Distributed stochastic gradient tracking methods. *Mathematical Programming*, pages 1–49, 2020.

[87] Jinming Xu, Ye Tian, Ying Sun, and Gesualdo Scutari. Accelerated primal-dual algorithms for distributed smooth convex optimization over networks. In *International Conference on Artificial Intelligence and Statistics*, pages 2381–2391. PMLR, 2020.

[88] Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Lee, and Laurent Massoulié. Optimal convergence rates for convex distributed optimization in networks. *Journal of Machine Learning Research*, 20:1–31, 2019.

[89] Ruggero Carli, Fabio Fagnani, Paolo Frasca, and Sandro Zampieri. Gossip consensus algorithms via quantized communication. *Automatica*, 46(1):70–80, 2010.

[90] Yucheng Lu and Christopher De Sa. Moniqua: Modulo quantized communication in decentralized SGD. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

[91] Sindri Magnússon, Hossein Shokri-Ghadikolaei, and Na Li. On maintaining linear convergence of distributed learning and optimization under limited communication. *IEEE Transactions on Signal Processing*, 68:6101–6116, 2020.

[92] Sulaiman A Alghunaim and Ali H Sayed. Linear convergence of primal–dual gradient methods and their performance in distributed optimization. *Automatica*, 117:109003, 2020.

[93] Zhuorui Li, Yiwei Liao, Kun Huang, and Shi Pu. Compressed gradient tracking for decentralized optimization with linear convergence. *arXiv preprint arXiv:2103.13748*, 2021.

[94] Yongyang Xiong, Ligang Wu, Keyou You, and Lihua Xie. Quantized distributed gradient tracking algorithm with linear convergence in directed networks. *arXiv preprint arXiv:2104.03649*, 2021.

[95] Zhuoqing Song, Lei Shi, Shi Pu, and Ming Yan. Compressed gradient tracking for decentralized optimization over general directed networks. *IEEE Transactions on Signal Processing*, 70:1775–1787, 2022.

[96] Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1-2):83–112, 2017.

[97] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in neural information processing systems*, 26:315–323, 2013.

[98] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. *Advances in neural information processing systems*, 27, 2014.

[99] Dmitry Kovalev, Samuel Horváth, and Peter Richtárik. Don't jump through hoops and remove those loops: Svrg and katyusha are better without the outer loop. In *Algorithmic Learning Theory*, pages 451–467. PMLR, 2020.

[100] Aryan Mokhtari and Alejandro Ribeiro. DSA: Decentralized double stochastic averaging gradient algorithm. *The Journal of Machine Learning Research*, 17(1):2165–2199, 2016.

[101] Kun Yuan, Bicheng Ying, Jiageng Liu, and Ali H Sayed. Variance-reduced stochastic learning by networked agents under random reshuffling. *IEEE Transactions on Signal Processing*, 67(2):351–366, 2018.

[102] Ran Xin, Usman A Khan, and Soummya Kar. Variance-reduced decentralized stochastic optimization with accelerated convergence. *IEEE Transactions on Signal Processing*, 68:6255–6271, 2020.

[103] Boyue Li, Shicong Cen, Yuxin Chen, and Yuejie Chi. Communication-efficient distributed optimization in networks with gradient tracking and variance reduction. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 1662–1672. PMLR, 26–28 Aug 2020.

[104] Hadrien Hendrikx, Francis Bach, and Laurent Massoulié. Dual-free stochastic decentralized optimization with variance reduction. *Advances in Neural Information Processing Systems*, 33:19455–19466, 2020.

[105] Ying Sun, Gesualdo Scutari, and Amir Daneshmand. Distributed optimization based on gradient tracking revisited: Enhancing convergence rate via surrogation. *SIAM Journal on Optimization*, 32(2):354–385, 2022.

[106] Sulaiman A Alghunaim, Ernest Ryu, Kun Yuan, and Ali H Sayed. Decentralized proximal gradient algorithms with linear convergence rates. *IEEE Transactions on Automatic Control*, 2020.

[107] Jinming Xu, Ye Tian, Ying Sun, and Gesualdo Scutari. Distributed algorithms for composite optimization: unified framework and convergence analysis. *IEEE Transactions on Signal Processing*, 69:3555–3570, 2021.

[108] Haishan Ye, Ziang Zhou, Luo Luo, and Tong Zhang. Decentralized accelerated proximal gradient descent. *Advances in Neural Information Processing Systems*, 2020, 2020.

[109] Laurent Condat. A primal–dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms. *Journal of optimization theory and applications*, 158(2):460–479, 2013.

[110] Bang Cong Vu. A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Advances in Computational Mathematics*, 38(3):667–681, 2013.

[111] Peijun Chen, Jianguo Huang, and Xiaoqun Zhang. A primal-dual fixed point algorithm for minimization of the sum of three convex separable functions. *Fixed Point Theory and Applications*, 2016(1):1–18, 2016.

[112] Ming Yan. A new primal–dual algorithm for minimizing the sum of three functions with a linear operator. *Journal of Scientific Computing*, 76(3):1698–1717, 2018.