Provably Efficient Algorithms for Decentralized Optimization

by

Changxin Liu

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Mechanical Engineering

Provably Efficient Algorithms for Decentralized Optimization

by

Changxin Liu

Supervisory Committee

Dr. Yang Shi, Supervisor
(Department of Mechanical Engineering)

Dr. Daniela Constantinescu, Departmental Member
(Department of Mechanical Engineering)

Dr. Jane Ye, Outside Member
(Department of Mathematics and Statistics)

ABSTRACT

Decentralized multi-agent optimization has emerged as a powerful paradigm that finds broad applications in engineering design including federated machine learning and control of networked systems. In these setups, a group of agents are connected via a network with general topology. Under the communication constraint, they aim to solving a global optimization problem that is characterized collectively by their individual interests. Of particular importance are the computation and communication efficiency of decentralized optimization algorithms. Due to the heterogeneity of local objective functions, fostering cooperation across the agents over a possibly time-varying network is challenging yet necessary to achieve fast convergence to the global optimum. Furthermore, real-world communication networks are subject to congestion and bandwidth limit. To relieve the difficulty, it is highly desirable to design communication-efficient algorithms that proactively reduce the utilization of network resources. This dissertation tackles four concrete settings in decentralized optimization, and develops four provably efficient algorithms for solving them, respectively.

Chapter 1 presents an overview of decentralized optimization, where some preliminaries, problem settings, and the state-of-the-art algorithms are introduced. Chapter 2 introduces the notation and reviews some key concepts that are useful throughout this dissertation. In Chapter 3, we investigate the non-smooth cost-coupled decentralized optimization and a special instance, that is, the dual form of constraint-coupled decentralized optimization. We develop a decentralized subgradient method with double averaging that guarantees the last iterate convergence, which is crucial to solving decentralized dual Lagrangian problems with convergence rate guarantee. Chapter 4 studies the composite cost-coupled decentralized optimization in stochastic networks, for which existing algorithms do not guarantee linear convergence. We propose a new decentralized dual averaging (DDA) algorithm to solve this problem. Under a rather mild condition on stochastic networks, we show that the proposed DDA attains an $\mathcal{O}(1/t)$ rate of convergence in the general case and a global linear rate of convergence if each local objective function is strongly convex. Chapter 5 tackles the smooth cost-coupled decentralized constrained optimization problem. We leverage the extrapolation technique and the average consensus protocol to develop an accelerated DDA algorithm. The rate of convergence is proved to be $\mathcal{O}\left(\frac{1}{t^2}+\frac{1}{t(1-\beta)^2}\right)$, where β denotes the second largest singular value of the mixing matrix. To proactively reduce the utilization of network resources, a communication-efficient decentralized primal-

dual algorithm is developed based on the event-triggered broadcasting strategy in Chapter 6. In this algorithm, each agent locally determines whether to generate network transmissions by comparing a pre-defined threshold with the deviation between the iterates at present and lastly broadcast. Provided that the threshold sequence is summable over time, we prove an $\mathcal{O}(1/t)$ rate of convergence for convex composite objectives. For strongly convex and smooth problems, linear convergence is guaranteed if the threshold sequence is diminishing geometrically. Finally, Chapter 7 provides some concluding remarks and research directions for future study.

Table of Contents

Supervisory Committee **ii**

Abstract **iii**

Table of Contents **v**

List of Tables **viii**

List of Figures **ix**

Acronyms **x**

Acknowledgements **xi**

1 Introduction **1**

1.1 Overview of Decentralized Optimization 1

1.1.1 Cost-Coupled Decentralized Optimization 2

1.1.2 Constraint-Coupled Decentralized Optimization 6

1.1.3 Communication-Efficient Decentralized Optimization 9

1.2 Organization and Contributions 12

2 Preliminaries **15**

2.1 Notation 15

2.2 Communication Model 15

2.2.1 Network Topology 15

2.2.2 Mixing Matrix 16

2.3 Optimization Background 16

3 Decentralized Subgradient Methods with Double Averaging **18**

3.1 Introduction 18

3.2 Problem Setup and Preliminaries 20
3.2.1 Basic Setup 20
3.2.2 Communication Network 20
3.2.3 Subgradient Method with Double Averaging 21
3.3 Algorithm and Convergence Results 21
3.4 Proofs of Convergence Results 22
3.5 Extension to Constraint-Coupled Decentralized Optimization 29
3.6 Experiment 36
3.7 Conclusion 38

4 Decentralized Dual Averaging Methods 39
4.1 Introduction 39
4.2 Related Work 41
4.3 Problem Setup and Preliminaries 42
4.3.1 Basic Setup 42
4.3.2 Stochastic Communication Network 43
4.3.3 Centralized Dual Averaging Method 44
4.4 Algorithm and Convergence Results 48
4.5 Proofs of Convergence Results 54
4.6 Proofs of Supporting Lemmas for Theorem 4.2 61
4.6.1 Proof of Lemma 4.2 61
4.6.2 Proof of Lemma 4.3 62
4.6.3 Proof of Lemma 4.4 71
4.7 Experiments 72
4.7.1 Decentralized Logistic Regression 73
4.7.2 Decentralized LASSO 76
4.8 Conclusion 77

5 Accelerated Decentralized Dual Averaging Method 79
5.1 Introduction 79
5.2 Problem Setup and Preliminaries 81
5.2.1 Problem Setup 81
5.2.2 Centralized Accelerated Dual Averaging 81
5.3 Algorithm and Convergence Result 82
5.4 Proof of Convergence Result 85

5.4.1 Notations and Supporting Lemmas 85
5.4.2 Proof of Theorem 5.1 92
5.5 Experiments 95
5.5.1 Case I: Real Dataset 95
5.5.2 Case II: Synthetic Dataset 96
5.6 Conclusion 96

6 Communication-Efficient Decentralized Primal-Dual Algorithms 98
6.1 Introduction 98
6.2 Problem Setup and Preliminaries 100
6.2.1 Basic Setup 100
6.2.2 Primal-Dual Formulation 100
6.3 Algorithm and Convergence Results 101
6.3.1 Algorithm Development 101
6.3.2 Convergence Results 103
6.4 Proofs of Convergence Results 107
6.4.1 Proof of Theorem 6.1 107
6.4.2 Proof of Theorem 6.2 114
6.5 Experiments 117
6.5.1 Decentralized l_1-l_2 Minimization 117
6.5.2 Decentralized Logistic Regression 120
6.6 Conclusion 121

7 Conclusion and Future Directions 122
7.1 Conclusions 122
7.2 Future Work 124
7.2.1 Privacy-Preserving and Resilient Decentralized Optimization . 124
7.2.2 Dual Averaging Methods for Decentralized Online Optimization 124
7.2.3 Rate Analysis of DDA Methods Under Error Bound Conditions 125

Appendix A Publications 126

Bibliography 128

List of Tables

Table 1.1 An overview of cost-coupled decentralized convex optimization algorithms. 6
Table 1.2 An overview of constraint-coupled decentralized convex optimization algorithms. 9
Table 1.3 An overview of communication-efficient decentralized convex optimization algorithms. 12

Table 6.1 The time spent per iteration for COCA and event-triggered LALM119

List of Figures

Figure 1.1 Centralized network versus decentralized network 2

Figure 3.1 Trajectories of the primal objective error $\left|\sum_{i=1}^{50} c_i x_i^{(t)} - \sum_{i=1}^{50} c_i x_i^*\right|$ (left-hand side) and the quadratic penalty for the coupled constraint $\left\|\left(b - \sum_{i=1}^{50} d_i \log(1 + x_i^{(t)})\right)_+\right\|^2$ 38

Figure 4.1 Comparison results for decentralized logistic regression in different network configurations. 75

Figure 4.2 Comparison results for decentralized LASSO in different network configurations. 78

Figure 5.1 Comparison of objective error in Case I. 97

Figure 5.2 Comparison of objective value in Case II. 97

Figure 6.1 Objective error $|\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*)|$ versus iteration number and broadcasting times when $r = 0.4$. 119

Figure 6.2 Objective error $|\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*)|$ versus iteration number and broadcasting times in different random networks. 119

Figure 6.3 RSE versus iteration number and broadcasting times when $r = 0.04$. 121

Acronyms

MAS multi-agent system

DGD decentralized gradient descent

DSA$_2$ decentralized subgradient with double averaging

DDA decentralized dual averaging

ADDA accelerated decentralized dual averaging

ALM augmented Lagrangian method

LALM linearized augmented Lagrangian method

ADMM alternating direction method of multipliers

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor Prof. Yang Shi for giving me the opportunity to work under his supervision at UVic. Throughout my PhD, his enthusiasm in pursuing fundamental research problems has stimulated me to do so as well; his guidance and feedback have greatly sharpened my thinking and brought my research work to a higher level. I am also deeply indebted to him for his encouragement and support whenever I was frustrated, and for his invaluable advice for my career development.

I wish to thank the thesis committee members, Prof. Jane Ye and Prof. Daniela Constantinescu, for their willingness to serve on the committee. I am also grateful for their warm support when I was TA for their courses. I would like to thank the External Examiner Prof. Na Li from Harvard University for her time in evaluating the thesis.

I also want to express my deep appreciation to Prof. Huiping Li from Northwestern Polytechnical University for the mentorship over the years. I am especially thankful for his valuable suggestions on my research work, writing, and presentation, and for his encouragement for me to pursue academic life.

I am thankful for Yong Zhang and Zirui Zhou, the mentors for my intership at Huawei Canada. I greatly thank them for the fruitful discussions that have broadened my horizons, and for their insightful comments that have improved our work.

I would like to thank the ACIPL team: Kunwu Zhang, Qi Sun, Qian Zhang, Henglai Wei, Tianyu Tan, Tianxiang Lu, Xinxin Shang, Xiang Sheng, and Chonghan Ma. I am grateful for the sincere friendship, research discussions, and those uncounted coffee gatherings.

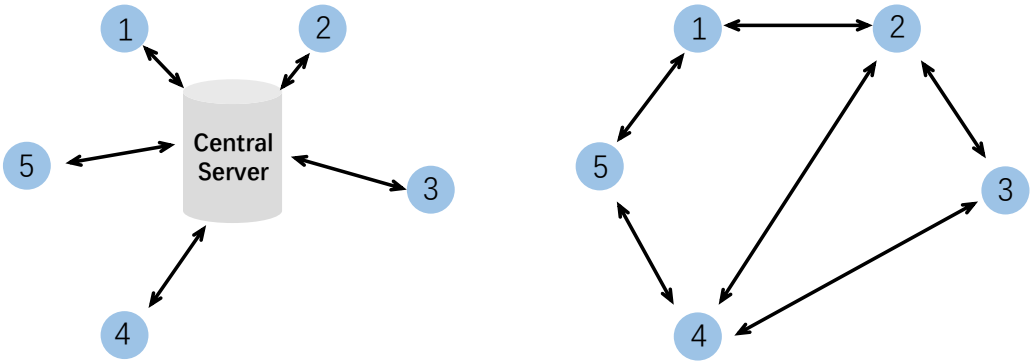
Finally, I am utmostly grateful to my parents and sister for their support throughout my life. They are so considerate and attentive. I would also like to give special thanks to our new family member–my niece Yishu– for the joyful moments she brought to us last year.

Chapter 1

Introduction

1.1 Overview of Decentralized Optimization

Multi-agent optimization has received increasing attention lately, primarily because it imparts balanced computation, privacy preservation, and communication efficiency to modern large-scale machine learning [46,62]. It refers to the optimization problems where a group of agents (e.g., processor, robots) aim to solve a common optimization problem in a collaborative manner. For example, in supervised machine learning, a set of parameters that characterize the mapping function from data to labels are determined by minimizing a loss function that penalizes the fitting error. However, it is usually inconvenient to perform such a task on one single machine due to concerns about data privacy and/or computational inefficiency. One attempt, referred to as the *distributed* solution [15, 42], to solving this problem is to use a central server to coordinate multiple agents to perform optimization collaboratively, as demonstrated in Figure 1.1a. Although it helps secure data privacy and facilitate parallel computing, this approach still suffers from several disadvantages. First, the star network topology renders the system sensitive to network changes, as possible disconnections lead to loss of training data and significant performance degradation. Second, the requirement on the bandwidth around the central server is high, in the sense that the computing efficiency can be largely declined if timely communication between the server and the agents is not guaranteed. Therefore, it is pivotal to pursue fully *decentralized* solutions where a central server is removed and each agent only exchanges information with its immediate neighbors – see Figure 1.1b, which is the main theme of this dissertation. We continue with an overview of decentralized optimization in different settings.



(a) Centralized network (b) Decentralized network

Figure 1.1: Centralized network versus decentralized network

1.1.1 Cost-Coupled Decentralized Optimization

Consider a multi-agent system (MAS) consisting of n agents, each of which, say i, has access to a local objective function f_i and the common constraint set $\mathcal{X} \subseteq \mathbb{R}^m$. They are connected via a general communication network and they aim to solving the following optimization problem:

$$\min_{x \in \mathcal{X}} \left\{ f(x) := \frac{1}{n} \sum_{i=1}^{n} f_i(x) \right\} \tag{1.1}$$

in a decentralized manner. That is, when solving (1.1), a pair of agents can exchange information only if they are connected in the communication network. This problem is referred to as *cost-coupled decentralized optimization* in the literature and finds broad applications in optimal control of MAS [76], sensor networks [75], and machine learning [46], to name a few.

As an example, we consider training parametric linear models for classification in machine learning. A set of n users, each of which possesses a proprietary dataset, are interested in achieving this task. Let the model of interest be $y = \langle x, M \rangle + b$ where $x \in \mathbb{R}^m$ is the parameter and $b \in \mathbb{R}$ the bias. In particular, each user i has a collection of m_i labeled samples; for each sample i the features and the label are denoted as $M_j^i \in \mathbb{R}^m$ and $y_j^i \in \{1, -1\}$, $j = 1, \cdots, m_i$, respectively. Performing training on the datasets of all the users generally leads to superior performance than that with a local dataset. Thus, the following logistic regression problem is constructed based on the

overall dataset

$$\min_{x\in\mathbb{R}^m} \frac{1}{n}\sum_{i=1}^{n} f_i(x), \quad f_i(x) = \frac{1}{m_i}\sum_{j=1}^{m_i} \ln\left(1+\exp\left(-(M_j^{i\mathrm{T}}x+b)y_j^i\right)\right) + \frac{\mu}{2}\|x\|^2, \quad (1.2)$$

where the loss function $\frac{1}{n}\sum_{i=1}^{n} f_i$ penalizes the fitting error and $\mu > 0$ characterizes the regularization term that prevents overfitting. Clearly, the problem in (1.2) is a cost-coupled decentralized optimization problem.

In the following, we provide an overview of existing algorithms applicable to Problem (1.1); see Table 1.1. We begin by presenting an equivalent reformulation of (1.1):

$$\min_{x_1,\ldots,x_n\in\mathcal{X}} \left\{ \mathbf{f}(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} f_i(x_i) \right\} \quad (1.3a)$$

$$\text{s.t.} \quad x_1 = x_2 \cdots = x_n, \quad (1.3b)$$

where $\mathbf{x} = [x_1^{\mathrm{T}}, \ldots, x_n^{\mathrm{T}}]^{\mathrm{T}}$. Depending on how the consensus requirement in (1.3b) is enforced, existing algorithms can be roughly categorized into two classes, that is, consensus-based decentralized optimization methods and decentralized primal-dual methods.

Consensus-based decentralized optimization methods. In this class, the average consensus protocols [122] are leveraged to amend the local search direction within each agent, such that consensus and optimization can be achieved simultaneously. In particular, a doubly stochastic matrix $P \in [0,1]^{n\times n}$ [4] is usually used to encode the network topology and the weights of connected links. Based on the local search philosophy, we further categorize the consensus-based methods into the following three subgroups.

i) Consensus-based decentralized gradient descent (DGD). In consensus-based DGD, each agent i generates two sequences of variables $\{x_i^{(t)}\}_{t\geq 1}$ and $\{y_i^{(t)}\}_{t\geq 1}$ in an iterative manner by imitating centralized gradient descent. For example, in the

seminal work [65], the iteration rule at time t reads

$$\begin{aligned} y_i^{(t+1)} &= \sum_{j\in\mathcal{N}_i\bigcup\{i\}} p_{ij}x_j^{(t)} - a_t\nabla f_i(x_i^{(t)}) \\ x_i^{(t+1)} &= \operatorname*{argmin}_{x\in\mathcal{X}} \ \|x - y_i^{(t+1)}\|^2, \end{aligned} \tag{1.4}$$

where ∇f_i denotes of gradient of f_i, a_t denotes the step size that is decreasing over time, and $\mathcal{N}_i$ is the set of neighbors of agent i. Since $\sum_{j\in\mathcal{N}_i\bigcup\{i\}} p_{ij} = 1$, one can verify that the movement of $y_i^{(t)}$ in DGD is together guided by the local gradient ∇f_i and the consensus error vector $\sum_{j\in\mathcal{N}_i} p_{ij}(x_j^{(t)} - x_i^{(t)})$, which is key to achieving consensus and optimization simultaneously. When the problem is unconstrained, i.e., $\mathcal{X} \equiv \mathbb{R}^m$, and smooth, there are several attempts in the literature to speed up the algorithm in [65]. For example, the authors in [83] proposed the EXTRA algorithm that adds a cumulative correction term to conventional DGD such that a constant step size can be used to accelerate the convergence. Specifically, EXTRA has an $\mathcal{O}(1/t)$ rate of convergence when the problem is convex and a linear rate of convergence if the problem is strongly convex. Alternatively, an additional gradient-tracking process based on the dynamic average consensus scheme in [122] can be used to equip each agent with an estimation of $\frac{1}{n}\sum_{i=1}^n \nabla f_i(x_i^{(t)})$ to facilitate local search, which also validates the use of a constant step size in DGD [73, 111]. This methodology was later extended to decentralized optimization over stochastic networks [111]. Also based on this idea, a decentralized Nesterov gradient descent was proposed in [74].

It is worth to mention that in (1.4) the local estimates $\{x_i^{(t-1)} : i = 1, \cdots, n\}$, which are obtained via a projection operator at time $t-1$, are averaged at time t. As documented in [18], such a nonlinear consensus-projection coupling makes the convergence rate analysis challenging, especially when the network is not static. This leads to the technical difficulty of developing consensus-based *projected* gradient methods that can exploit the smoothness property of the objective functions. Notably, the authors in [84] overcame this challenge; they developed a decentralized proximal gradient method and proved an $\mathcal{O}(1/t)$ rate of convergence in terms of the norm of the difference of two consecutive iterates.

ii) Consensus-based decentralized dual averaging (DDA). Different from DGD, the

update of consensus-based DDA algorithms [18] at time t is written as

$$z_i^{(t+1)} = \sum_{j=1}^{n} p_{ij}^{(t)} z_j^{(t)} + \nabla f_i(x_i^{(t)}) \tag{1.5a}$$

$$x_i^{(t+1)} = \operatorname*{argmin}_{x \in \mathcal{X}} \left\{ a_t \left\langle z_i^{(t+1)}, x \right\rangle + \frac{1}{2}\|x\|^2 \right\}, \tag{1.5b}$$

where each agent seeks consensus among local dual variables $\{z_i^{(t)} : i = 1, \ldots, n\}$. Under the standard assumption of bounded (sub)gradient, the dynamics in (1.5a) is decoupled from the projection operation and purely linear. This essentially facilitates the analysis of DDA-type algorithms even the network is time-varying and random. The $\mathcal{O}(1/\sqrt{t})$ rate of convergence for DDA is firstly established in [18] for non-smooth problems. Although this strategy was later extended to handle more general settings, e.g., directed communication network [48, 89], and nonseparable global objectives [38], they both considered general non-smooth problems and obtained an $\mathcal{O}(1/\sqrt{t})$ sublinear rate of convergence.

iii) Other consensus-based methods. Several other first-order optimization methods such as the Frank-Wolfe method [96] and the conjugate gradient method [108] have also been used to develop consensus-based decentralized algorithms. The authors in [94] proposed a decentralized Newton-Raphson method, where the Hessian and gradient of the overall objective function are estimated via two separate dynamic average consensus schemes. Recently, the authors in [117] developed a Newton tracking algorithm to avoid exchanging Hessian among agents. For decentralized second-order methods, more restrictive assumptions, e.g., the local objective functions are twice differentiable, are usually required for guaranteeing convergence.

Decentralized primal-dual methods. This type of methods are inspired by another equivalent form of (1.1)

$$\begin{aligned} \min_{x_1,\ldots,x_n \in \mathcal{X}} \ & \left\{ \mathbf{f}(\mathbf{x}) := \frac{1}{n} \sum_{i=1}^{n} f_i(x_i) \right\} \\ \text{s.t.} \quad & (\mathcal{L} \otimes I)\mathbf{x} = \mathbf{0} \end{aligned} \tag{1.6}$$

where $\mathcal{L}$ denotes the Laplacian matrix associated with the communication network, I is an identity matrix of size $m \times m$, $\otimes$ denotes the Kronecker product, and $\mathbf{x} = [x_1^{\mathrm{T}}, \ldots, x_n^{\mathrm{T}}]^{\mathrm{T}}$. Since (4.2) is a linearly constrained optimization problem, centralized primal-dual optimization paradigms such as the alternating direction method of multipliers (ADMM) [5], in company with proper coordinate change of dual variables, can be used to design decentralized algorithms. In contrast to consensus-based methods, constraints can be conveniently handled in this framework. However, since $\mathcal{L}$ needs to be explicitly given in the formulation (4.2), those algorithms and the associated linear convergence results cannot be extended to stochastic communication networks, where the network topology is random and not ensured to be connected at each time instant.

Algorithms		Constrained/ Composite	Stochastic comm.	Convergence rate	
				Cvx	Strongly cvx
Consensus -based	[55]	×	√	-	-
	[18]	√	√	$\mathcal{O}(1/\sqrt{t})$	-
	[73]	×	×	$\mathcal{O}(1/t)$	Linear
	[111]	×	√	-	Linear
	[96]	√	×	$\mathcal{O}(1/t)$	$\mathcal{O}(1/t^2)$
	Ch. 4	√	√	$\mathcal{O}(1/t)$	Linear
	Ch. 5	√	×	$\mathcal{O}(1)(\frac{1}{t^2} + \frac{1}{t(1-\beta)^2})$	-
Primal-dual	[83]	×	×	$\mathcal{O}(1/t)$	Linear
	[84]	√	×	$\mathcal{O}(1/t)$	-
	[1]	√	×	-	Linear
	[110]	√	×	$\mathcal{O}(1/t)$	Linear

Table 1.1: An overview of cost-coupled decentralized convex optimization algorithms.

1.1.2 Constraint-Coupled Decentralized Optimization

For this class of problems, each agent holds its own decision variable x_i, objective function J_i, and constraint $\mathcal{X}_i$. In addition, all the agents are coupled via global constraints, e.g., $\sum_{i=1}^n q_i(x_i) \leq 0$. Formally, the minimization problem is given by

$$\begin{aligned} \min_{\{x_i \in \mathcal{X}_i\}_{i=1}^n} & \sum_{i=1}^n J_i(x_i) \\ \text{s.t.} \quad & \sum_{i=1}^n q_i(x_i) \leq 0. \end{aligned} \tag{1.7}$$

Such a problem finds applications in resource allocation [90], decentralized charging control of plug-in electric vehicles (PEVs) [19, 95], and distributed control of constraint-coupled multi-agent systems [100].

For example, in charging control of PEVs we consider a fleet of n PEVs that shall be charged by drawing power from one electricity distribution network. The problem is concerned with finding an optimal charging schedule that fulfills several constraints, e.g., the preferred final state of charge for each PEV and the maximum power flow of the network, which can be formulated as the following optimization problem:

$$\begin{aligned} \min_{\{x_i \in \mathcal{X}_i\}_{i=1}^n} \quad & \sum_{i=1}^n c_i^{\mathrm{T}} x_i \\ \text{s.t.} \quad & \sum_{i=1}^n \left(A_i x_i - \frac{b}{n} \right) \leq 0 \end{aligned}$$

where the entries in vector x_i denote the charging rate in certain time slots for PEV i, c_i the corresponding costs with unitary charging rate, $\mathcal{X}_i$ represents the local constraint for PEV i, and $\sum_{i=1}^n \left(A_i x_i - \frac{b}{n} \right) \leq 0$ expresses the network-wide power constraints.

The approaches developed recently are mostly based on Lagrangian duality [14, 64, 68]. We continue with a survey of existing algorithms applicable to solving (1.7); see Table 1.2.

Dual decomposition methods. Define the Lagrangian function

$$\sum_{i=1}^n \left(J_i(x_i) + \langle \lambda, q_i(x_i) \rangle \right),$$

where $\lambda \geq 0$ represents the dual variable associated with the coupled constraint. Then, the dual problem of (1.7) is

$$\max_{\lambda \geq 0} \left\{ \psi(\lambda) := \sum_{i=1}^n \psi_i(\lambda) \right\} \tag{1.8}$$

where

$$\psi_i(\lambda) = \min_{x_i \in \mathcal{X}_i} \left\{ J_i(x_i) + \langle \lambda, q_i(x_i) \rangle \right\}.$$

Standard dual decomposition methods [64, 68] require a fusion center that is able to communicate with all the agents to determine the gradient of the dual objective function. Note that the dual Lagrangian problem in (1.8) has the same structure with the cost-coupled decentralized optimization problem. Therefore, existing cost-coupled decentralized optimization algorithms can be used to solve the outer problem in (1.8) in a decentralized manner. For example, the work in [100] formed a double-loop algorithm that combines the accelerated gradient method and a finite time consensus scheme to tackle the dual problem. The authors in [47] theoretically validated the use of a constant step size for the case where the objective and the constraint functions are smooth. In non-smooth settings, recent work in [70] properly relaxed the constraint-coupled problem and explored the duality principle twice to design a decentralized algorithm. Alternatively, the authors in [19, 59, 86] employed the consensus-based decentralized subgradient methods. In particular, the authors in [82] used ADMM and the primal-dual method of multipliers (PDMM) to solve the dual problem; however the convergence results are missing. The works in [86] and [59] considered the settings with constant step size and decaying step sizes, respectively, under the assumption that a Slater point exists and is known to all agents. To get such an assumption satisfied, a decentralized method is provided in [59] to find a Slater point. The framework considered in [19] relaxed this assumption, but the requirement on the step size is more restrictive, i.e., square summable step sizes.

When the coupled constraint is characterized by a special linear equality, several methods with improved convergence results were reported in the literature. For example, the authors applied the splitting technique to the dual problem and came up with an algorithm that has an $\mathcal{O}(1/t)$ rate of convergence if the cost function is convex and a linear rate of convergence when the cost is smooth and strongly convex [113]. For unbalanced communication networks and nonconvex cost functions, a decentralized algorithm was proposed based on a similar methodology in [119].

Augmented Lagrangian methods. Dual decomposition methods may suffer from several advantages including slow convergence and non-uniqueness of solutions. To tackle these problems, regularization techniques have been used. When the coupled constraint is a linear equality, the authors in [7] proposed an accelerated distributed augmented Lagrangian method (ALM). The method was later extended to handle nonconvex problems in [8]. These algorithms need a central server to update dual variables. To achieve decentralized implementation, a local version of the dual update

within the server was incorporated into each agent, where the gradient of the dual function is estimated via dynamic average consensus [20].

Algorithms		Additional assumptions	Coupled constraint		Convergence rate
			Equality	Inequality	
Dual decomposition methods	[68]	Central server needed	$\surd$	$\surd$	$\mathcal{O}(1/\sqrt{t})$
	[64]	Slater point known	$\surd$	$\surd$	$\mathcal{O}(1/t)$
	[19]	-	$\surd$	$\surd$	-
	[86]	Slater point known	$\surd$	$\surd$	$\mathcal{O}(1/t)$
	[59]	Slater point known	$\surd$	$\surd$	$\mathcal{O}(1/\sqrt{t})$
	[47]	Smoothness	$\surd$	$\surd$	-
	Ch. 3	-	$\surd$	$\surd$	$\mathcal{O}(1/\sqrt{t})$
Augmented Lagrangian methods	[7]	Central server needed	$\surd$	$\times$	$\mathcal{O}(1/t)$
	[20]	-	$\surd$	$\times$	-
	[121]	Multiple comm. rounds	$\surd$	$\times$	$\mathcal{O}(1/t)$

Table 1.2: An overview of constraint-coupled decentralized convex optimization algorithms.

1.1.3 Communication-Efficient Decentralized Optimization

In the above subsections, there is one underlying assumption that the communication between agents is perfect. However, this is rarely the case in practice. Indeed, real-world communication networks are subject to congestion and bandwidth limit. To relieve the difficulty, compressing the traffic via sparsification or quantization is a promising solution, which has been actively explored lately. In the following, we provide a survey of communication-efficient decentralized optimization algorithms in the literature; see Table 1.3.

Decentralized optimization with event-triggered broadcasting. Over the past decade, event-triggered broadcasting has emerged as a promising communication-efficient approach for scheduling data transmission in large-scale networked control systems [2, 88, 98]. The idea is to generate network transmission only when the information conveyed by the message is deemed innovative to the system, and whether it is innovative is determined via a user-defined function that takes the deviation between the actual system state and the state just broadcast as an argument. The hope of event-triggered control is to reduce the communication load while largely preserving the control performance. To exploit this attractive feature in decentralized

optimization, event-triggered communication has been incorporated into decentralized optimization algorithms lately [10, 23, 30, 41, 51, 54]. For example, the authors developed their event-triggered variants based on the decentralized optimization algorithm in [65]. Although reductions in communication were observed in numerical experiments, due to the use of diminishing step sizes, the convergence rates are rather slow: $\mathcal{O}\left(\frac{\log t}{\sqrt{t}}\right)$ in [41] and $\mathcal{O}\left(\frac{1}{\log t}\right)$ in [30]. To speed up the convergence, constant step sizes were used in event-triggered DGD [51]. Based on [73], the authors in [23] developed an event-triggered algorithm for strongly convex and smooth objective functions, where an additional event-triggered dynamic average consensus scheme is used to track the mean of local gradients. Recent work in [54] presented an event-triggered decentralized ADMM method that only requires each agent to broadcast the primal variable to its neighbors, and prove the convergence of the algorithm when the objective function is general convex. Convergence rates are analyzed for strongly convex and smooth objective functions. Furthermore, it is remarked in [54] that the event-triggered zero-gradient-sum decentralized optimization method in [10] can be seen as an event-triggered version of dual decomposition that is empirically slower than ADMM. In these schemes, each agent at every generic time instant is required to exactly solve a subproblem, which may be not practical in most cases.

Decentralized optimization with quantization/compression. In digital signal processing, quantization refers to the process of mapping input values from a continuous set to output values in a countable set. Typical examples of quantization processes include rounding and truncation. Quantization has been incorporated into the design of decentralized averaging protocols [32, 43], where the focus was placed on minimizing the effect of quantization error on the performance of algorithms. Recently, significant efforts have been devoted to designing quantized decentralized optimization algorithms. For example, the authors in [115] developed a quantized decentralized subgradient algorithm in undirected networks. Using a random quantization strategy, a decentralized gradient method was proposed in [16]. However, they have sublinear rates of convergence even when the objective functions are strongly convex. To achieve linear convergence, the authors in [34, 53] developed the DQOA and LEAD algorithm, respectively. Under the assumption that the random quantizer is unbiased and δ-contracted, i.e., $\mathbb{E}[Q(x)] = x$ and $\mathbb{E}[\|Q(x) - x\|^2] \leq \delta\|x\|^2$ for all $x \in \mathbb{R}^m$, the algorithms are proved to converge linearly. Reference [57] investigated the tradeoff between the convergence speed and the communication cost.

Linearly convergent quantized algorithms have also been extended to handle directed networks in [107]. Note that event-triggered broadcasting and quantization are orthogonal; they have been combined to design communication-efficient decentralized optimization methods [87].

Other communication-efficient decentralized optimization algorithms. Besides the above two strategies, some other types of asynchronous decentralized optimization algorithms have been reported in the literature to alleviate the communication burden. For instance, the authors in [61] considered the DGD operated in a network with random communication link failures, and established convergence rate and error bound for decaying and constant step sizes, respectively. Using a similar idea, reference [25] presented an asynchronous DGD where only a randomized set of working agents choose to update their local iterates at each time instant. The authors proved that the local estimates converge to a neighborhood of the minimizer provided that the activation probability grows to one asymptotically. The works [6, 102] developed asynchronous ADMM methods, and proved their rates of convergence. However, in these methods each agent still needs to exactly solve a subproblem at each local iteration. Recently, reference [104] designed an asynchronous decentralized consensus optimization algorithm based on [83] for a network of agents where communication delays may occur, and proved the convergence of the algorithm. Another communication-efficient decentralized gradient method was reported in [118]; its novelty may lie in the use of only signs of relative variable information between immediate neighbors. However, the convergence is rather slow, i.e., $\mathcal{O}\left(\frac{\log t}{\sqrt{t}}\right)$, due to diminishing step sizes. A random walk incremental strategy was used in [58] to design a communication-efficient asynchronous decentralized optimization algorithm, where a constant step size is used to achieve fast convergence to the global optimum exactly. The authors in [9] considered a communication scenario where a central server does not periodically request gradients from all workers in decomposable convex optimization. The authors in [35] co-designed the primal-dual decentralized optimization algorithm in outer loop and the subproblem-solving process in inner loop to save communication resources.

Algorithms	Communication strategy		Convergence rate	
	Event-triggered	Compression	Cvx	Strongly cvx
[41]	$\surd$	$\times$	$\mathcal{O}(\frac{\log t}{\sqrt{t}})$	-
[30]	$\surd$	$\times$	$\mathcal{O}(\frac{1}{\log t})$	-
[10, 23, 51, 54]	$\surd$	$\times$	-	Linear
Ch. 6	$\surd$	$\times$	$\mathcal{O}(1/t)$	Linear
[115]	$\times$	$\surd$	-	-
[16]	$\times$	$\surd$	$\mathcal{O}(1/\sqrt[4]{t})$	$\mathcal{O}(1/\sqrt[3]{t})$
[34, 53]	$\times$	$\surd$	-	Linear
[87]	$\surd$	$\surd$	-	$\mathcal{O}(1/t)$

Table 1.3: An overview of communication-efficient decentralized convex optimization algorithms.

1.2 Organization and Contributions

In this dissertation, four concrete settings in decentralized optimization are considered. The outline and main contributions of this dissertation are summarized below:

- In **Chapter** 2, we introduce the notation and review some preliminaries that are useful throughout this work.

- In **Chapter** 3, we consider *non-smooth* cost-coupled and constraint-coupled decentralized optimization in networks. Most decentralized non-smooth optimization algorithms cannot generate a convergent sequence of local variables. For decentralized dual Lagrangian problems where the local dual variable is further used to coordinate subproblems, they become not applicable. To relieve the difficulty, we proposed a decentralized subgradient method with double averaging (DSA_2) that is able to generate a convergent sequence of local iterates. Thanks to this property, an extension of DSA_2 is made to decentralized dual Lagrangian problems. Sublinear rates of convergence are established for both settings. Numerical experiment and comparison are conducted to illustrate the advantage of DSA_2 and validate our theoretical findings.

- In **Chapter** 4, we study *composite* cost-coupled decentralized optimization in stochastic networks, for which existing algorithms do not guarantee linear convergence. We propose a new DDA algorithm to solve this problem. Under a rather mild condition on stochastic networks, we show that the proposed algo-

rithm attains an $\mathcal{O}(1/t)$ rate of convergence in the general case and a global linear rate of convergence if each local objective function is strongly convex. Our algorithm substantially improves the existing DDA-type algorithms as the latter were only known to converge sublinearly prior to our work. The key to achieving the improved rate is the design of a novel dynamic averaging consensus protocol for DDA, which intuitively leads to more accurate local estimates of the global dual variable. Numerical results are also presented to support our design and analysis.

- In **Chapter** 5, we study accelerated decentralized optimization for *smooth* cost-coupled problems. We develop an accelerated DDA (ADDA) algorithm, where each agent employs the first-order dynamic average consensus to estimate the average of local gradients. Upon scaling the estimates with monotonically increasing weights and accumulating the resultant variable over time, each agent generates a local dual variable. Then, the convex conjugate of a 1-strongly convex function over the dual variable is identified and used to construct two sequences of primal variables in an iterative manner based on the extrapolation technique and the average consensus protocol. The rate of convergence is proved to be $\mathcal{O}(1)\left(\frac{1}{t^2}+\frac{1}{t(1-\beta)^2}\right)$, where β denotes the second largest singular value of the mixing matrix. Notably, the condition for the algorithmic parameter to guarantee convergence does not rely on the mixing matrix. Establishing such a condition that is independent on the mixing matrix offers the appealing advantage of convenient verification in practical applications. Finally, numerical results are presented to demonstrate the efficiency of ADDA.

- In **Chapter** 6, we investigate the communication-efficient decentralized optimization problem. Upon modeling decentralized optimization as a linearly constrained problem, we leverage the linearized augmented Lagrangian method (LALM) and the event-triggered broadcasting strategy to design a communication-efficient decentralized optimization algorithm that only requires light local computation at generic time instants and peer-to-peer communication at sporadic triggering time instants. The triggering time instants for each agent are locally determined by comparing the deviation between true and broadcast primal variables with time-varying triggering thresholds. Provided that the threshold is summable over time, we prove an $\mathcal{O}(1/t)$ rate of convergence for convex composite problems. Stronger convergence results are

obtained for strongly convex and smooth problems, that is, the iterates linearly converge when the triggering thresholds are geometrically diminishing. Finally, the developed strategy is examined with two common optimization problems; comparison results illustrate its performance and superiority in exploiting communication resources.

- In **Chapter** 7, we conclude the dissertation and present several avenues for future research.

Chapter 2

Preliminaries

In this chapter, we introduce the main notation, the communication model, and some basic concepts in optimization, which are useful for the subsequent analysis.

2.1 Notation

$\mathbb{R}$, $\mathbb{R}^m$, and $\mathbb{R}^{m\times m}$ denote the set of real valued numbers, vectors, and matrices, respectively. Column vectors are considered as the default orientation unless otherwise stated. We let $\mathbf{1}$ be a vector with all entries equal to one, where the dimension should be understood from the context. Notation '$\geq$' is element-wise when applied to vectors. All norms are 2-norms unless otherwise stated. Given a vector $x \in \mathbb{R}^m$ and a positive semi-definite matrix $P \in \mathbb{R}^{m\times m}$, the notation $\|x\|_P^2$ denotes $x^{\mathrm{T}}Px$. We use $\mathrm{diag}\{\eta_i\}_{i=1}^n$ to denote a diagonal matrix where the diagonal entries are $\eta_1, \cdots \eta_n$. For matrix P, the i-th largest singular value (eigenvalue) is written as $\sigma_i(P)(\lambda_i(P))$. Let $\rho(P) = \max_{i=1,\cdots m}|\lambda_i(P)|$. The Kronecker product is denoted by $\otimes$. Given a real number a, we let $\lceil a \rceil$ be the ceiling function that maps a to the least integer greater than or equal to a. We denote by $\mathcal{O}(\alpha)$ the values in the order of the scalar α, e.g., $\mathcal{O}(\alpha) = a\alpha$ for some constant a independent of α.

2.2 Communication Model

2.2.1 Network Topology

In this dissertation, we consider solving finite-sum optimization problems in a decentralized manner. That is, each agent is able to communicate with other agents at

time t only if they are connected in the communication network at t. To describe the network topology at time t, we use a bidirectional graph $\mathcal{G}^{(t)} = \{\mathcal{V}, \mathcal{E}^{(t)}\}$ (we omit the superscript for t when the graph is time-invariant), where $\mathcal{V} = \{1, \cdots, n\}$ denotes the set of n agents and $\mathcal{E}^{(t)} \subseteq \mathcal{V} \times \mathcal{V}$ represents the set of links, i.e., $(i, j) \in \mathcal{E}^{(t)}$ indicates that nodes i and j can send information to each other at time t. Agent j is said to be a neighbor of i at t if there exists a link between them at t, and the set of i's neighbors at t is denoted by $\mathcal{N}_i^{(t)} = \{j \in \mathcal{V} | (j, i) \in \mathcal{E}^{(t)}\}$. For $\mathcal{G}^{(t)}$, three $n \times n$ matrices are defined: The adjacency matrix $\mathcal{A}^{(t)} = [a_{ij}^{(t)}]$ where each entry $a_{ij}^{(t)} = 1$ if $(i, j) \in \mathcal{E}^{(t)}$ and $a_{ij}^{(t)} = 0$ otherwise, the diagonal degree matrix $\mathcal{D}^{(t)} = \text{diag}\{|\mathcal{N}_i^{(t)}|\}_{i=1}^n$, and the graph Laplacian $\mathcal{L}^{(t)} = \mathcal{D}^{(t)} - \mathcal{A}^{(t)}$. For undirected graphs, the matrix $\mathcal{L}^{(t)}$ is ensured to be positive semi-definite.

2.2.2 Mixing Matrix

Given a communication graph $\mathcal{G}^{(t)}$, for each pair $(i, j) \in \mathcal{E}^{(t)}$ we assign a positive weight $p_{ij}^{(t)} > 0$ to agent i to weigh the information received from j. We let $p_{ij}^{(t)} = 0$ if j is not an immediate neighbor of agent i. Denote the mixing matrix constructed from these weights by

$$P^{(t)} := [p_{ij}^{(t)}] \in [0, 1]^{n \times n}.$$

Given a graph $\mathcal{G}^{(t)}$, there exist many rules to determine the weights in a decentralized manner [63, 105]. For example, one can use the Metropolis rule [60] as follows

$$p_{ij}^{(t)} = \begin{cases} \frac{1}{1+\max\{|\mathcal{N}_i|, |\mathcal{N}_j|\}}, & \text{if } (j, i) \in \mathcal{E}^{(t)}, \\ 1 - \sum_{k \in \mathcal{N}_i} p_{ik}, & \text{if } j = i, \\ 0, & \text{otherwise.} \end{cases}$$

2.3 Optimization Background

In this section, some basic optimization concepts [3] are briefly reviewed.

Definition 2.1. (Convex set) *A set $\mathcal{C}$ is said to be convex if for every pair of points x and y in $\mathcal{C}$, the entire line segment connecting x and y is also contained in $\mathcal{C}$.*

Definition 2.2. (Convex function) *A function $f : \mathbb{R}^m \to \mathbb{R}$ is said to be convex if for any $\lambda \in [0, 1]$ and $x, y \in \mathbb{R}^m$*

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda) f(y)$$

Definition 2.3. (μ-strongly convex function) *A function $f : \mathbb{R}^m \to \mathbb{R}$ is said to be μ-strongly convex if for any $\lambda \in [0, 1]$ and $x, y \in \mathbb{R}^m$*

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y) - \frac{\mu}{2}\lambda(1-\lambda)\|x-y\|^2.$$

If f is also differentiable, then μ-strong convexity leads to

$$f(y) \geq f(x) + \langle \nabla f(x), y - x\rangle + \frac{\mu}{2}\|y-x\|^2. \tag{2.1}$$

Definition 2.4. (Lipschitz continuity) *A function $f : \mathcal{C} \to \mathbb{R}$ is called G-Lipschitz over a set $\mathcal{C}$ if for any $x, y \in \mathcal{C}$*

$$|f(x) - f(y)| \leq G\|x - y\|.$$

A differentiable function f is said to have L-Lipschitz continuous gradient or equivalently L**-smooth** if

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x-y\|, \forall x, y \in \mathbb{R}^m. \tag{2.2}$$

If f is also convex, one has

$$f(y) \leq f(x) + \langle \triangledown f(x), y - x\rangle + \frac{L}{2}\|y-x\|^2. \tag{2.3}$$

Definition 2.5. (Subdifferential) *The subdifferential of a function $f : \mathbb{R}^m \to \mathbb{R}$ at some $x \in \mathbb{R}^m$ is the set of all the subgradients*

$$\partial f(x) = \{g_x | f(y) \geq f(x) + \langle g_x, y - x\rangle, \forall y \in \mathbb{R}^m\}.$$

Chapter 3

Decentralized Subgradient Methods with Double Averaging

3.1 Introduction

In this chapter, we consider the cost-coupled decentralized optimization problem, where n agents connected via a bidirectional network aim to collaboratively solving the following constrained optimization problem:

$$\min_{x\in\mathcal{X}} \left\{ f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x) \right\} \tag{3.1}$$

where f_i denotes the local non-smooth objective function of agent i, and $\mathcal{X} \subseteq \mathbb{R}^m$ represents the constraint set shared by all the agents.

Generally speaking, the design of decentralized optimization algorithms [18, 63, 73,83,85,111,112,116] consists of two crucial steps. In the first step, one assigns local copies of the global variable to each agent such that each agent has a local version of the optimization variable to work with, and imposes a consensus constraint on all the local variables to guarantee the equivalence between the problems before and after transformation. Then, a local iteration rule associated with an appropriate consensus-building mechanism is designed. Existing methods essentially differ from each other in terms of the second step. For example, the algorithms reported in [63,73,83,85,111, 112,116] produced local iterates based on primal methods that generate points in the feasible set that is contained in the primal space of variables. One typical example of primal methods is the projected subgradient method, where the iterates are generated

by gradually shifting them along the opposite directions of subgradients, followed by a projection step; see [66] for more details. In this class of methods, consensus among local iterates is usually enforced by distributed averaging based on doubly stochastic mixing matrices. There are also some decentralized optimization algorithms available in the literature [18, 81] where the local iteration rule imitates dual methods, e.g., dual averaging [69]. It is shown in [18] that agreeing on the linear model of the global objective function can alleviate some technical difficulties faced by primal methods due to the consensus-projection coupling.

When the objective function is non-smooth, most existing decentralized optimization algorithms may not be able to generate *a convergent sequence of iterates*. Indeed, they only guarantee the convergence of the objective error over the running average of local iterates, i.e., ergodic convergence properties. This essentially allows undesired jumps of the objective function values at some iterations, possibly threatening the stability of the decentralized system. In centralized optimization, this problem may be mitigated by further considering the best iterates achieved so far. This procedure, however, may be not implementable in decentralized scenarios since it requires the global objective function that is not available locally.

Contribution. The main contributions of this chapter are summarized in the following.

i) For non-smooth cost-coupled optimization, we propose a decentralized subgradient method with double averaging (abbreviated as DSA_2) which ensures convergence in non-ergodic sense, i.e, each local sequence of iterates is convergent. Compared with existing decentralized dual averaging methods [18], we introduce an averaging step to the iteration scheme and theoretically show that it is this additional averaging step that makes the sequence of local test points convergent. We prove an $\mathcal{O}(1/\sqrt{t})$ rate of convergence for DSA_2.

ii) Extension is made to solving constraint-coupled decentralized optimization by combining dual decomposition and DSA_2. In particular, the coupling in constraints of the primal problem is transformed into that in objective functions of the dual problem by following Lagrangian relaxation. Then, a primal-dual sequence is constructed by solving the dual problem via DSA_2 and using the local dual iterates to determine the corresponding primal variables. We proved that the dual objective error and the quadratic penalty for the violation of coupled

constraints have $\mathcal{O}(1/\sqrt{t})$ subliear rates of convergence, and the primal objective error vanishes asymptotically. Numerical experiment results are provided to verify our theoretical findings.

3.2 Problem Setup and Preliminaries

3.2.1 Basic Setup

We consider the finite-sum optimization problem in (3.1), in which $\mathcal{X}$ is a closed convex constraint set and f_i satisfies the following assumptions for all $i = 1, \cdots, n$.

Assumption 3.1. *i) f_i is convex on $\mathcal{X}$;*

ii) f_i is G-Lipschitz continuous on $\mathcal{X}$.

Throughout this chapter, we denote by x^* an optimal solution of Problem (3.1). Assumption 3.1 is satisfied for a host of functions, e.g., any convex function on a closed domain or polyhedral function on an arbitrary domain. A consequence of Assumption 3.1-ii) is that any subgradient $g_i \in \partial f_i(x)$ for any $\forall x \in \mathcal{X}$ is bounded [18], i.e.,

$$\|g_i\| \leq G.$$

3.2.2 Communication Network

We consider solving Problem (3.1) in a decentralized manner. That is, each agent i holds a local objective function f_i and is able to communicate with other agents only if they are connected in the communication network. To model the decentralized communication, we consider a fixed bidirectional graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ and a mixing matrix $P = [p_{ij}]$. We make the following standard assumption for them.

Assumption 3.2. *i) The graph $\mathcal{G}$ is connected;*

ii) P has a strictly positive diagonal, i.e., $p_{ii} > 0$;

iii) P is doubly stochastic, i.e., $P\mathbf{1} = \mathbf{1}$ and $\mathbf{1}^{\mathrm{T}}P = \mathbf{1}^{\mathrm{T}}$.

Assumption 3.2 ensures $\sigma_2(P) < 1$. In particular, Assumptions 3.2 (i) and 3.2 (ii) make the matrix $P^{\mathrm{T}}P$ irreducible and primitive, respectively. This fact together with Assumption 3.2 (iii) gives that $P^{\mathrm{T}}P$ has a unique Perron-Frobenius eigenvalue which is 1, meaning that $\sigma_2(P) < 1$.

3.2.3 Subgradient Method with Double Averaging

Let $d : \mathcal{X} \to \mathbb{R}$ be a 1-strongly convex function on $\mathcal{X}$ such that

$$x^{(0)} = \operatorname*{argmin}_{x \in \mathcal{X}} d(x) \text{ and } d(x^{(0)}) = 0 \tag{3.2}$$

The centralized subgradient method with double averaging (SA_2) [67] generates $\{x^{(t)}\}_{t \geq 0}$ iteratively according to

$$\hat{x}^{(t)} = \operatorname*{argmin}_{x \in \mathcal{X}} \left\{ \left\langle \sum_{\tau=0}^{t-1} g^{(\tau)}, x \right\rangle + \gamma_{t-1} d(x) \right\} \tag{3.3a}$$

$$x^{(t)} = \frac{t}{t+1} x^{(t-1)} + \frac{1}{t+1} \hat{x}^{(t)}, \tag{3.3b}$$

where $g^{(t)} \in \partial f(x^{(t)})$ with $f = \frac{1}{n} \sum_{i=1}^{n} f_i$, γ_t is a non-decreasing sequence of positive parameters. Compared with the dual averaging method [18, 69], this scheme has an averaging step in (3.3b) that makes the sequence $\{x^{(t)}\}_{t \geq 0}$ convergent in a non-ergodic sense [67].

3.3 Algorithm and Convergence Results

In this section, we develop the DSA_2 algorithm and present its convergence results.

From (3.3), we observe that the update of $\hat{x}^{(t)}$ depends on the subgradient accumulated over time, e.g., $\sum_{\tau=0}^{t-1} g^{(\tau)}$. Then, $\hat{x}^{(t)}$ is averaged over time to update $x^{(t)}$. To imitate the update (3.3) in decentralized optimization, a local estimate of $\sum_{\tau=0}^{t-1} g^{(\tau)}$ may be necessary. And it is reasonable to expect that, if the estimate is sufficiently accurate, then decentralized optimization can be fulfilled.

We employ the following consensus scheme to estimate $\sum_{\tau=0}^{t-1} g^{(\tau)}$:

$$z_i^{(t)} = \sum_{j=1}^{n} p_{ij} z_j^{(t-1)} + g_i^{(t-1)}, \tag{3.4}$$

where $g_i^{(t)} \in \partial f_i(x_i^{(t)})$. Equipped with $z_i^{(t)}$, each agent is able to run an *inexact* version

of (3.3)

$$\hat{x}_i^{(t)} = \operatorname*{argmin}_{x\in\mathcal{X}} \left\{ \left\langle z_i^{(t)}, x \right\rangle + \gamma_{t-1} d(x) \right\} \tag{3.5a}$$

$$x_i^{(t)} = \frac{t}{t+1} x_i^{(t-1)} + \frac{1}{t+1} \hat{x}_i^{(t)}, \tag{3.5b}$$

where $x_i^{(t)}$ represents the local iterate updated by agent i at time instant t. We take $\gamma_{-1} = \gamma_0$ by convention. The entire algorithm is summarized in Algorithm 1.

Algorithm 1 Decentralized Subgradient Method with Double Averaging (DSA_2)

1: **Input:** $\{\gamma_t\}_{t\geq 0}$, $x^{(0)} \in \mathcal{X}$, and a strongly convex function d with parameter 1 on $\mathcal{X}$ such that (3.2) holds
2: **Initialize:** $x_i^{(0)} = x^{(0)}$, and $z_i^{(0)} = 0$ for all $i = 1, \cdots, n$
3: **for** $t = 1, 2, \cdots$ **do**
4: *In parallel (task for agent i, $i = 1, \cdots, n$)*
5: collect $z_j^{(t)}$ from all agents $j \in \mathcal{N}_i$
6: update $z_i^{(t)}$ by (3.4)
7: update $x_i^{(t)}$ by (3.5)
8: broadcast $z_i^{(t)}$ to all agents $j \in \mathcal{N}_i$
9: **end for**

Theorem 3.1. *Suppose that $d(x^*) \leq R^2$ and $\gamma_t = \gamma\sqrt{t+1}$ where $\gamma > 0$, and Assumptions 3.1, 3.2 hold. For the sequences $\{x_i^{(t)}\}_{t\geq 0}$ generated by Algorithm 1, we have*

$$f(x_i^{(t)}) - f(x^*) \leq \frac{1}{\sqrt{t+1}} \left(\frac{G^2}{\gamma} \left(\frac{6\sqrt{n}}{1-\sigma_2(P)} + 13 \right) + \gamma R^2 \right). \tag{3.6}$$

3.4 Proofs of Convergence Results

Motivated by the literature regarding consensus-based decentralized optimization, we set up an auxiliary sequence $\{y^{(t)}\}_{t\geq 0}$ whose update obeys the following

$$\begin{aligned} \hat{y}^{(t)} &= \arg\min_{x\in\mathcal{X}} \left\{ \left\langle \overline{z}^{(t)}, x \right\rangle + \gamma_{t-1} d(x) \right\} \\ y^{(t)} &= \frac{t}{t+1} y^{(t-1)} + \frac{1}{t+1} \hat{y}^{(t)}, \end{aligned} \tag{3.7}$$

where $\overline{z}^{(t)} = \frac{1}{n}\sum_{i=1}^{n} z_i^{(t)}$ and $y^{(0)} = x^{(0)}$.

Before proving Theorem 3.1, we present three technical lemmas.

Lemma 3.1. *For the sequence $\{\hat{x}_i^{(t)} : i = 1, \ldots, n\}_{t\geq 0}$ generated by Algorithm 1 and the auxiliary sequence $\{\hat{y}^{(t)}\}_{t\geq 0}$ in (3.7), one has that for all $t \geq 0$ and $i = 1, \cdots, n$,*

$$\|\hat{x}_i^{(t)} - \hat{y}^{(t)}\| \leq \frac{1}{\gamma_{t-1}}\|z_i^{(t)} - \overline{z}^{(t)}\|. \tag{3.8}$$

Proof of Lemma 3.1. For $t = 0$, the inequality holds because both sides of (3.8) equal 0. Now, suppose that $t \geq 1$. Recall that d is strongly convex with modulus 1. Let the mapping $R : \mathbb{R}^m \to \mathbb{R}^m$ be defined as

$$R(\omega) := \operatorname*{argmin}_{x\in\mathcal{X}} \left\{\langle \omega, x\rangle + \gamma_{t-1} d(x)\right\}. \tag{3.9}$$

By (3.5a) and (3.7), we have

$$\hat{y}^{(t)} = R(\overline{z}^{(t)}), \quad \hat{x}_i^{(t)} = R(z_i^{(t)}), \quad \forall i = 1, \ldots, n$$

The mapping R is Lipschitz continuous with Lipschitz constant γ_{t-1}^{-1}; see, e.g., Proposition 4.9 in [29]. Therefore (3.8) holds. □

Before establishing the relation between $\{z_i^{(t)} : i = 1, \ldots, n\}_{t\geq 0}$ and $\{\overline{z}^{(t)}\}_{t\geq 0}$, we introduce the following notation:

$$\tilde{z}^{(t)} = z_i^{(t)} - \overline{z}^{(t)}, \quad \overline{g}^{(t)} = \frac{1}{n}\sum_{i=1}^{n} g_i^{(t)}, \tag{3.10}$$

$$\mathbf{z}^{(t)} = \begin{bmatrix} z_1^{(t)} \\ \vdots \\ z_n^{(t)} \end{bmatrix}, \quad \tilde{\mathbf{z}}^{(t)} = \begin{bmatrix} \tilde{z}_1^{(t)} \\ \vdots \\ \tilde{z}_n^{(t)} \end{bmatrix}, \quad \mathbf{g}^{(t)} = \begin{bmatrix} g_1^{(t)} \\ \vdots \\ g_n^{(t)} \end{bmatrix}. \tag{3.11}$$

Equipped with these notation, we can re-write the update rule (3.4) in the following compact form:

$$\mathbf{z}^{(t)} = \mathbf{P}\mathbf{z}^{(t-1)} + \mathbf{g}^{(t-1)}, \tag{3.12}$$

where $\mathbf{P} = P \otimes I$ with I being an identity matrix of size $m \times m$.

Lemma 3.2. *For the sequence* $\{z_i^{(t)} : i = 1, \ldots, n\}_{t\geq 0}$ *and* $\{\overline{z}^{(t)}\}_{t\geq 0}$*, we have*

$$\|\tilde{z}_i^{(t)}\| \leq \frac{\sqrt{n}G}{1-\sigma_2(P)} + 2G. \tag{3.13}$$

Proof of Lemma 3.2. We start by iterating (3.12)

$$\mathbf{z}^{(t)} = \mathbf{P}^t \mathbf{z}^{(0)} + \sum_{\tau=0}^{t-2} \mathbf{P}^{t-1-\tau} \mathbf{g}^{(\tau)} + \mathbf{g}^{(t-1)}. \tag{3.14}$$

Summing over (3.4) from $i = 1$ to $i = n$, we obtain

$$\overline{z}^{(\tau)} = \frac{1}{n}\sum_{i=1}^{n} z_i^{(\tau)} = \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{n} p_{ij} z_j^{(\tau-1)} + g_i^{(\tau)}\right) = \overline{z}^{(\tau-1)} + \overline{g}^{(\tau-1)},$$

which implies

$$\overline{z}^{(t)} = \overline{z}^{(0)} + \sum_{\tau=0}^{t-2} \overline{g}^{(\tau)} + \overline{g}^{(t-1)}. \tag{3.15}$$

Upon subtracting $\mathbf{1} \otimes \overline{z}^{(t)}$ on both sides of (3.14), and using (3.15) and

$$z_i^{(0)} = 0, \ i = 1, \cdots, n,$$

we obtain

$$\begin{aligned}
\tilde{\mathbf{z}}^{(t)} &= \sum_{\tau=0}^{t-2}\left(\mathbf{P}^{t-1-\tau} - \left(\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n} \otimes I\right)\right)\mathbf{g}^{(\tau)} + \mathbf{g}^{(t-1)} - \mathbf{1} \otimes \overline{g}^{(t-1)} \\
&= \sum_{\tau=0}^{t-2}\left(\left(P^{t-1-\tau} - \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}\right) \otimes I\right)\mathbf{g}^{(\tau)} + \mathbf{g}^{(t-1)} - \mathbf{1} \otimes \overline{g}^{(t-1)}
\end{aligned}$$

where $\mathbf{1}$ is an all-one column vector of dimension n. Therefore

$$\tilde{z}_i^{(t)} = \sum_{\tau=0}^{t-2}\sum_{j=1}^{n}\left([P^{t-1-\tau}]_{ij} - 1/n\right) g_j^{(\tau)} + g_i^{(t-1)} - \overline{g}^{(t-1)}$$

where $[P^{t-1-\tau}]_{ij}$ represents the (i,j)-th entry of $P^{t-1-\tau}$. Taking norm on both sides

gives rise to

$$\begin{aligned}\|\tilde{z}_i^{(t)}\| &= \left\| \sum_{\tau=0}^{t-2} \sum_{j=1}^{n} \left([P^{t-1-\tau}]_{ij} - 1/n \right) g_j^{(\tau)} + g_i^{(t-1)} - \overline{g}^{(t-1)} \right\| \\ &\leq \sum_{\tau=0}^{t-2} \sum_{j=1}^{n} \left\| [P^{t-1-\tau}]_{ij} - 1/n \right\| \|g_j^{(\tau)}\| + \|g_i^{(t-1)} - \overline{g}^{(t-1)}\| \\ &\leq \sum_{\tau=0}^{t-2} \sum_{j=1}^{n} \left| [P^{t-1-\tau}]_{ij} - 1/n \right| G + 2G \\ &\leq \sum_{\tau=0}^{t-2} \left\| [P^{t-1-\tau}]_i - \mathbf{1}^{\mathrm{T}}/n \right\|_1 G + 2G.\end{aligned}$$

Recall that for a stochastic matrix P one has $\left\|[P^{t-1-\tau}]_i - \mathbf{1}/n\right\|_1 \leq \sigma_2(P)^{t-1-\tau}\sqrt{n}$ [18]. Then the inequality in (3.13) follows, thereby concluding the proof. □

We present a slightly modified result in dual averaging (Theorem 2 in [69], Lemma 3 in [18]).

Lemma 3.3. *Suppose Assumptions 3.1, 3.2 hold true. For any non-decreasing sequence $\{\gamma_t\}_{t\geq 0}$ of positive parameters, and $x \in \mathcal{X}$, we have*

$$\sum_{\tau=0}^{t} \langle \overline{g}^{(\tau)}, \hat{y}^{(\tau)} - x \rangle \leq \frac{1}{2} \sum_{\tau=0}^{t} \frac{1}{\gamma_{\tau-1}} \|\overline{g}^{(\tau)}\|^2 + \gamma_t d(x). \tag{3.16}$$

A proof of Lemma 3.3 is presented here for completeness.

Proof of Lemma 3.3. Define

$$\Psi_{\gamma_\tau}^*(w) = \sup_{x \in \mathcal{X}} \{\langle w, x \rangle - \gamma_\tau d(x)\}$$

Recall (3.15) that $\overline{z}^{(\tau)} = \sum_{k=0}^{\tau-1} \overline{g}^{(k)}$. Since γ_τ is non-decreasing,

$$\Psi_{\gamma_\tau}^*(-\overline{z}^{(\tau+1)}) \leq \Psi_{\gamma_{\tau-1}}^*(-\overline{z}^{(\tau+1)}) = \Psi_{\gamma_{\tau-1}}^*(-\overline{z}^{(\tau)} - \overline{g}^{(\tau)}). \tag{3.17}$$

Note that

$$\nabla \Psi_{\gamma_{\tau-1}}^*(-\overline{z}^{(\tau)}) = R(\overline{z}^{(\tau)}) = \hat{y}^{(\tau)}$$

where R is defined in (3.9). Therefore $\Psi_{\gamma_{\tau-1}}^*(z)$ has $\gamma_{\tau-1}^{-1}$-Lipschitz continuous gradi-

ent, and

$$\Psi^*_{\gamma_{\tau-1}}(-\overline{z}^{(\tau+1)}) \leq \Psi^*_{\gamma_{\tau-1}}(-\overline{z}^{(\tau)}) - \langle \hat{y}^{(\tau)}, \overline{g}^{(\tau)} \rangle + \frac{1}{2\gamma_{\tau-1}} \|\overline{g}^{(\tau-1)}\|^2.$$

Upon substituting the above inequality into (3.17), we obtain

$$\langle \hat{y}^{(\tau)}, \overline{g}^{(\tau)} \rangle \leq \Psi^*_{\gamma_{\tau-1}}(-\overline{z}^{(\tau)}) - \Psi^*_{\gamma_\tau}(-\overline{z}^{(\tau+1)}) + \frac{1}{2\gamma_{\tau-1}} \|\overline{g}^{(\tau-1)}\|^2.$$

By further summing up the above inequality from $\tau = 0$ to $\tau = t$, it follows

$$\sum_{\tau=0}^{t} \langle \overline{g}^{(\tau)}, \hat{y}^{(\tau)} \rangle \leq \Psi^*_{\gamma_{-1}}(-\overline{z}^{(0)}) - \Psi^*_{\gamma_\tau}(-\overline{z}^{(t+1)}) + \sum_{\tau=0}^{t} \frac{1}{2\gamma_{\tau-1}} \|\overline{g}^{(\tau-1)}\|^2.$$

Because

$$\begin{aligned} \sum_{\tau=0}^{t} \langle \overline{g}^{(\tau)}, -x \rangle &\leq \sup_{x \in \mathcal{X}} \left\{ \sum_{\tau=0}^{t} \langle \overline{g}^{(\tau)}, -x \rangle - \gamma_t d(x) \right\} + \gamma_t d(x) \\ &= \Psi^*_{\gamma_t}(-\overline{z}^{(t+1)}) + \gamma_t d(x), \forall x \in \mathcal{X} \end{aligned}$$

we have

$$\sum_{\tau=0}^{t} \langle \overline{g}^{(\tau)}, \hat{y}^{(\tau)} - x \rangle \leq \Psi^*_{\gamma_{-1}}(-\overline{z}^{(0)}) + \sum_{\tau=0}^{t} \frac{1}{2\gamma_{\tau-1}} \|\overline{g}^{(\tau-1)}\|^2 + \gamma_t d(x), \forall x \in \mathcal{X}$$

which together with (3.2) and $z_i^{(0)} = 0, i = 1, \cdots, n$ leads to (3.16) as desired. □

Now we are ready to prove Theorem 3.1.

Proof of Theorem 3.1. By convexity of f_j, we have

$$\begin{aligned} &(t+1) f_j(x_j^{(t)}) - \sum_{\tau=0}^{t} f_j(x_j^{(\tau)}) = t f_j(x_j^{(t)}) - \sum_{\tau=0}^{t-1} f_j(x_j^{(\tau)}) = \sum_{\tau=1}^{t} \tau \left(f_j(x_j^{(\tau)}) - f_j(x_j^{(\tau-1)}) \right) \\ &\leq \sum_{\tau=1}^{t} \tau \left\langle g_j^{(\tau)}, x_j^{(\tau)} - x_j^{(\tau-1)} \right\rangle \end{aligned} \tag{3.18}$$

and

$$f_j(x_j^{(\tau)}) - f_j(x) \leq \left\langle g_j^{(\tau)}, x_j^{(\tau)} - x \right\rangle. \tag{3.19}$$

Using inequalities (3.18) and (3.19), we consider

$$\begin{aligned}
&(t+1)\left(f_j(x_j^{(t)}) - f_j(x)\right) \\
&= (t+1) f_j(x_j^{(t)}) - \sum_{\tau=0}^{t} f_j(x_j^{(\tau)}) + \sum_{\tau=0}^{t}\left(f_j(x_j^{(\tau)}) - f_j(x)\right) \\
&\leq \sum_{\tau=1}^{t}\left\langle g_j^{(\tau)}, (\tau+1)x_j^{(\tau)} - \tau x_j^{(\tau-1)} - x \right\rangle + \left\langle g_j^{(0)}, x_j^{(0)} - x \right\rangle, \forall x \in \mathcal{X}
\end{aligned} \tag{3.20}$$

which in conjunction with an equivalent expression of (3.5b)

$$(\tau+1)x_j^{(\tau)} = \tau x_j^{(\tau-1)} + \hat{x}_j^{(\tau)}$$

leads to

$$(t+1)\left(f_j(x_j^{(t)}) - f_j(x)\right) \leq \sum_{\tau=0}^{t}\left\langle g_j^{(\tau)}, \hat{x}_j^{(\tau)} - x \right\rangle.$$

Upon summing the above inequality from $j = 1$ to $j = n$, we obtain

$$\begin{aligned}
&(t+1)\sum_{j=1}^{n}\left(f_j(x_j^{(t)}) - f_j(x)\right) \\
&\leq \sum_{j=1}^{n}\sum_{\tau=0}^{t}\left(\left\langle g_j^{(\tau)}, \hat{x}_j^{(\tau)} - \hat{y}^{(\tau)} \right\rangle + \left\langle g_j^{(\tau)}, \hat{y}^{(\tau)} - x \right\rangle\right) \\
&= \sum_{\tau=0}^{t}\left(\sum_{j=1}^{n}\left\langle g_j^{(\tau)}, \hat{x}_j^{(\tau)} - \hat{y}^{(\tau)} \right\rangle + n\left\langle \overline{g}^{(\tau)}, \hat{y}^{(\tau)} - x \right\rangle\right).
\end{aligned} \tag{3.21}$$

Due to $f = \frac{1}{n}\sum_{j=1}^{n} f_j$, we have

$$\begin{aligned}
&f(x_i^{(t)}) - f(x) = f(x_i^{(t)}) - f(y^{(t)}) + f(y^{(t)}) - f(x) \\
\leq& L\|x_i^{(t)} - y^{(t)}\| + \frac{1}{n}\left(\sum_{j=1}^{n}\left(f_j(y^{(t)}) - f_j(x_j^{(t)})\right) + \sum_{j=1}^{n}\left(f_j(x_j^{(t)}) - f_j(x)\right)\right) \\
\leq& L\left(\|x_i^{(t)} - y^{(t)}\| + \frac{1}{n}\sum_{j=1}^{n}\|x_j^{(t)} - y^{(t)}\|\right) + \frac{1}{n}\sum_{j=1}^{n}\left(f_j(x_j^{(t)}) - f_j(x)\right) \\
\leq& L\left(\|x_i^{(t)} - y^{(t)}\| + \frac{1}{n}\sum_{j=1}^{n}\|x_j^{(t)} - y^{(t)}\|\right) \\
&+ \frac{1}{t+1}\sum_{\tau=0}^{t}\left(\frac{1}{n}\sum_{j=1}^{n}\left\langle g_j^{(\tau)}, \hat{x}_j^{(\tau)} - \hat{y}^{(\tau)}\right\rangle + \left\langle \overline{g}^{(\tau)}, \hat{y}^{(\tau)} - x\right\rangle\right)
\end{aligned} \tag{3.22}$$

where we use the G-Lipschitz continuity of f and f_i to derive the first and second inequality, respectively, and (3.21) for the third inequality. Since

$$y^{(t)} = (t+1)^{-1}\left(y^{(0)} + \sum_{\tau=1}^{t}\hat{y}^{(\tau)}\right), \quad x_i^{(t)} = (t+1)^{-1}\left(x_i^{(0)} + \sum_{\tau=1}^{t}\hat{x}_i^{(\tau)}\right)$$

and $y^{(0)} = x_i^{(0)}$, we obtain

$$\begin{aligned}
f(x_i^{(t)}) - f(x) \leq& \frac{G}{t+1}\sum_{\tau=1}^{t}\left(\|\hat{x}_i^{(\tau)} - \hat{y}^{(\tau)}\| + \frac{1}{n}\sum_{j=1}^{n}\|\hat{x}_j^{(\tau)} - \hat{y}^{(\tau)}\|\right) \\
&+ \frac{1}{t+1}\sum_{\tau=0}^{t}\left(\frac{1}{n}\sum_{j=1}^{n}\langle \nabla f_j(x_j^{(\tau)}), \hat{x}_j^{(\tau)} - \hat{y}^{(\tau)}\rangle + \langle \overline{g}^{(\tau)}, \hat{y}^{(\tau)} - x\rangle\right) \\
\leq& \frac{1}{t+1}\left(G\sum_{\tau=1}^{t}\left(\|\hat{x}_i^{(\tau)} - \hat{y}^{(\tau)}\| + \frac{2}{n}\sum_{j=1}^{n}\|\hat{x}_j^{(\tau)} - \hat{y}^{(\tau)}\|\right) + \sum_{k=0}^{t}\langle \overline{g}^{(\tau)}, \hat{y}^{(\tau)} - x\rangle\right).
\end{aligned}$$

It follows from Lemmas 3.1, 3.2, 3.3 and the boundedness of $\|\overline{g}^{(\tau)}\|^2$ that

$$\begin{aligned}
&f(x_i^{(t)}) - f(x) \\
\leq &\frac{1}{t+1}\left(\sum_{\tau=1}^{t} G\Big(\|\hat{x}_i^{(\tau)} - \hat{y}^{(\tau)}\| + \frac{2}{n}\sum_{j=1}^{n}\|\hat{x}_j^{(\tau)} - \hat{y}^{(\tau)}\|\Big) + \sum_{\tau=0}^{t}\frac{1}{2\gamma_{\tau-1}}\|\overline{g}^{(\tau)}\|^2 + \gamma_t d(x)\right) \\
\leq &\frac{1}{t+1}\left(3G\left(\frac{\sqrt{n}G}{1-\sigma_2(P)} + 2G\right)\sum_{\tau=1}^{t}\frac{1}{\gamma_{\tau-1}} + \sum_{\tau=0}^{t}\frac{1}{2\gamma_{\tau-1}}\|\overline{g}^{(\tau)}\|^2 + \gamma_t d(x)\right) \\
\leq &\frac{1}{t+1}\left(\left(\frac{3\sqrt{n}G^2}{1-\sigma_2(P)} + 6G^2\right)\sum_{\tau=1}^{t}\frac{1}{\gamma_{\tau-1}} + \frac{G^2}{2}\sum_{\tau=0}^{t}\frac{1}{\gamma_{\tau-1}} + \gamma_t d(x)\right).
\end{aligned} \tag{3.23}$$

Due to

$$\sum_{\tau=0}^{t}\frac{1}{\gamma_{\tau-1}} = \frac{1}{\gamma_0} + \sum_{\tau=0}^{t-1}\frac{1}{\gamma_\tau} = \frac{1}{\gamma} + \frac{1}{\gamma}\sum_{\tau=0}^{t-1}\frac{1}{\sqrt{\tau+1}} \leq \frac{2}{\gamma}\sqrt{t+1},$$

we get

$$f(x_i^{(t)}) - f(x) \leq \frac{1}{\sqrt{t+1}}\left(\frac{G^2}{\gamma}\left(\frac{6\sqrt{n}}{1-\sigma_2(P)} + 13\right) + \gamma d(x)\right).$$

We arrive at (3.6) as desired by using the assumption that $d(x^*) \leq R^2$. □

3.5 Extension to Constraint-Coupled Decentralized Optimization

Consider the following constraint-coupled optimization problem

$$\begin{aligned}
\min_{\{x_i \in \mathcal{X}_i\}_{i=1}^n} \quad & \sum_{i=1}^{n} J_i(x_i) \\
\text{s.t.} \quad & \sum_{i=1}^{n} q_i(x_i) \leq 0,
\end{aligned} \tag{3.24}$$

where the set $\mathcal{X}_i \subset \mathbb{R}^{s_i}$, and the functions $J_i : \mathcal{X}_i \to \mathbb{R}$ and $q_i : \mathcal{X}_i \to \mathbb{R}^m$. We make the following assumption for Problem (3.24).

Assumption 3.3. *i) Each function J_i is convex, and each $\mathcal{X}_i$ is a nonempty compact convex set.*

ii) Each q_i is a componentwise convex function, i.e., for all $j = 1, \cdots, s_i$, each component q_{ij} is a convex function.

Assumption 3.4. *There exists $\overline{x}_1 \in \mathcal{X}_1$, $\cdots$, $\overline{x}_n \in \mathcal{X}_n$ such that $\sum_{i=1}^n q_i(\overline{x}_i) < 0$.*

Assumptions 3.3, 3.4 are standard and ensure that (3.24) has at least one optimal solution. We denote by $\{x_i^*\}_{i=1}^n$ one of the optimal solutions to Problem (3.24) and $J^* = \sum_{i=1}^n J_i(x^*)$ the minimal function value.

One powerful methodology for solving this problem is to alternatively consider the corresponding dual Lagrangian problem. In doing so, the coupling in constraints can be transformed into that in objective functions of the dual problem, thus allowing us to solve it via DSA_2. The Lagrangian of (3.24) is

$$\sum_{i=1}^n \left(J_i(x_i) + \langle \lambda, q_i(x_i) \rangle \right),$$

where $\lambda \geq 0$ represents the dual variable associated with the coupled constraint, and the dual Lagrangian problem is

$$\max_{\lambda \geq 0} \min_{\{x_i \in \mathcal{X}_i\}_{i=1}^n} \left\{ \sum_{i=1}^n J_i(x_i) + \langle \lambda, q_i(x_i) \rangle \right\},$$

which is equivalent to

$$\min_{\lambda \geq 0} \max_{\{x_i \in \mathcal{X}_i\}_{i=1}^n} -\left\{ \sum_{i=1}^n J_i(x_i) + \langle \lambda, q_i(x_i) \rangle \right\}.$$

Let

$$\psi_i(\lambda) = \max_{x_i \in \mathcal{X}_i} \left\{ -J_i(x_i) - \langle \lambda, q_i(x_i) \rangle \right\}$$

and rewrite the dual Lagrangian problem as

$$\min_{\lambda \geq 0} \psi(\lambda) := \sum_{i=1}^n \psi_i(\lambda). \tag{3.25}$$

Clearly, the dual Lagrangian problem has the same structure with (3.1). In this section, we denote one of the optimal dual variables by λ^*.

We use Algorithm 1 to solve the dual problem in (3.25), where $d(\lambda) = \|\lambda\|^2/2$ is chosen as the prox-function. The steps are detailed in the following. Each agent initializes the algorithm by setting $\lambda_i^{(0)} = 0$, and

$$x_i(\lambda_i^{(0)}) = \underset{x_i \in \mathcal{X}_i}{\operatorname{argmax}} \left\{ -J_i(x_i) \right\}. \tag{3.26}$$

At time $t = 1, 2, \cdots$, each agent updates its Lagrangian dual variable according to

$$\hat{\lambda}_i^{(t)} = \underset{\lambda \geq 0}{\operatorname{argmin}} \left\{ \langle z_i^{(t)}, \lambda \rangle + \gamma_{t-1} \|\lambda\|^2/2 \right\} \tag{3.27a}$$

$$\lambda_i^{(t)} = \frac{t}{t+1} \lambda_i^{(t-1)} + \frac{1}{t+1} \hat{\lambda}_i^{(t)}. \tag{3.27b}$$

where

$$z_i^{(t)} = \sum_{j=1}^{n} p_{ij} z_j^{(t-1)} - q_i(x_i(\lambda_i^{(t-1)})). \tag{3.28}$$

It is worth to mention that $-q_i(x_i(\lambda_i^{(t-1)})) \in \partial \psi_i(\lambda_i^{(t-1)})$ by Danskin's Theorem [68]. Then, based on the dual update, the primal variable is determined in the following way:

$$x_i(\lambda_i^{(t)}) = \underset{x_i \in \mathcal{X}_i}{\operatorname{argmax}} \left\{ -J_i(x_i) - \left\langle \lambda_i^{(t)}, q_i(x_i) \right\rangle \right\} \tag{3.29a}$$

$$x_i^{(t)} = \frac{t}{t+1} x_i^{(t-1)} + \frac{1}{t+1} x_i(\lambda_i^{(t)}). \tag{3.29b}$$

The step in (3.29b) can be seen as the primal recovery step that is common in dual decomposition algorithms [19, 86]. This step is needed since the dual objective function at the optimum is typically non-smooth, and the optimal dual variable does not necessarily lead to an optimal primal solution [68]. The overall algorithm is summarized in Algorithm 2.

Algorithm 2 DSA_2-based Dual Decomposition

Input: $\{\gamma_t\}_{t\geq 0}$, $d(\lambda) = \|\lambda\|^2/2$
Initialize: $\lambda_i^{(0)} = 0$, $z_i^{(0)} = 0$, and $x_i(\lambda_i^{(0)})$ according to (3.26) for all $i = 1, \cdots, n$
for $t = 1, 2, \cdots$ **do**
  In parallel (task for agent i, $i = 1, \cdots, n$)
  collect $z_j^{(t-1)}$ from all agents $j \in \mathcal{N}_i$
  update $z_i^{(t)}$ by (3.28)
  update $\lambda_i^{(t)}$ by (3.27)
  update $x_i^{(t)}$ by (3.29)
  broadcast $z_i^{(t)}$ to all agents $j \in \mathcal{N}_i$
end for

Theorem 3.2. *Suppose Assumptions 3.2, 3.3, and 3.4 hold true. Let the sequences $\{\lambda_i^{(t)}\}_{t\geq 0}$ and $\{x_i^{(t)}\}_{t\geq 0}$ be generated by Algorithm 2. If $\gamma_t = \gamma\sqrt{t+1}$ where $\gamma > 0$, then the dual objective error*

$$\sum_{j=1}^{n} \big(\psi_j(\lambda_i^{(t)}) - \psi_j(\lambda^*)\big) \leq \frac{n}{\sqrt{t+1}} \left(\frac{\big(\frac{6\sqrt{n}}{1-\sigma_2(P)} + 13\big)D}{\gamma} + \frac{\gamma\|\lambda^*\|^2}{2} \right), \tag{3.30}$$

the quadratic penalty for the coupled constraint

$$\big\| \big(\sum_{j=1}^{n} q_j(x_j^{(t)}) \big)_+ \big\|^2 \leq \frac{4n\big(\frac{\sqrt{n}}{1-\sigma_2(P)} + \frac{5}{2}\big)D}{t+1} + \frac{2\gamma C}{\sqrt{t+1}}, \tag{3.31}$$

and the primal objective error

$$-\|\lambda^*\| \sqrt{\frac{2n^2\big(\frac{2\sqrt{n}}{1-\sigma_2(P)} + 5\big)D}{t+1} + \frac{2n\gamma C}{\sqrt{t+1}}} \leq \sum_{j=1}^{n} J_j(x_j^{(t)}) - J^* \leq \frac{n\left(\frac{2\sqrt{n}}{1-\sigma_2(P)} + 5\right)D}{\gamma\sqrt{t+1}},$$

where $D = \max_{j\in\{1,\cdots,n\}} \max_{x_j\in\mathcal{X}_j} \|q_j(x_j)\|^2$ and $C = J^* - \min_{\{x_j\in\mathcal{X}_j\}_{j=1}^n} \sum_{j=1}^n J_j(x_j)$ are constants.

Proof. We begin by recalling

$$-q_j(x_j(\lambda_j^{(\tau)})) \in \partial\psi_j(\lambda_j^{(\tau)}).$$

Then, in light of (3.18), we readily have

$$(t+1)\psi_j(\lambda_j^{(t)}) - \sum_{\tau=0}^{t}\psi_j(\lambda_j^{(\tau)}) \le \sum_{\tau=1}^{t}\tau\left\langle -q_j(x_j(\lambda_j^{(\tau)})), \lambda_j^{\tau} - \lambda_j^{(\tau-1)}\right\rangle.$$

By adding $\sum_{\tau=0}^{t}\langle -q_j(x_j(\lambda_j^{(\tau)})), \lambda_j^{(\tau)} - \lambda\rangle, \forall\lambda \ge 0$ on both sides of the above inequality, we obtain

$$\begin{aligned}
&(t+1)\psi_j(\lambda_j^{(t)}) - \sum_{\tau=0}^{t}\left(\left\langle -q_j(x_j(\lambda_j^{(\tau)})), \lambda - \lambda_j^{(\tau)}\right\rangle + \psi_j(\lambda_j^{(\tau)})\right)\\
\le&\sum_{\tau=1}^{t}\left\langle -q_j(x_j(\lambda_j^{(\tau)})), (\tau+1)\lambda_j^{(\tau)} - \tau\lambda_j^{(\tau-1)} - \lambda\right\rangle + \left\langle -q_j(x_j(\lambda_j^{(0)})), \lambda_j^{(0)} - \lambda\right\rangle \quad (3.32)\\
=&\sum_{\tau=0}^{t}\left\langle -q_j(x_j(\lambda_j^{(\tau)})), \hat{\lambda}_j^{(\tau)} - \lambda\right\rangle,
\end{aligned}$$

where (3.27b) is used to get the last equality. Due to

$$\begin{aligned}
&\left\langle -q_j(x_j(\lambda_j^{(\tau)})), \lambda - \lambda_j^{(\tau)}\right\rangle + \psi_j(\lambda_j^{(\tau)})\\
=&\left\langle -q_j(x_j(\lambda_j^{(\tau)})), \lambda - \lambda_j^{(\tau)}\right\rangle - J_j\big(x(\lambda_j^{(\tau)})\big) - \left\langle \lambda_j^{(\tau)}, -q_j(x_j(\lambda_j^{(\tau)}))\right\rangle\\
=&-\left\langle q_j(x_j(\lambda_j^{(\tau)})), \lambda\right\rangle - J_j\big(x_j(\lambda_j^{\tau})\big),
\end{aligned}$$

we obtain

$$\begin{aligned}
&\sum_{\tau=0}^{t}\left(J_j\big(x_j(\lambda_j^{(\tau)})\big) + \left\langle q_j\big(x_j(\lambda_j^{(\tau)})\big), \lambda\right\rangle\right) + (t+1)\psi_j(\lambda_j^{(t)})\\
&\le \sum_{\tau=0}^{t}\left\langle -q_j(x_j(\lambda_j^{(\tau)})), \hat{\lambda}_j^{(\tau)} - \lambda\right\rangle.
\end{aligned} \quad (3.33)$$

Upon using

$$x_j^{(t)} = \frac{1}{t+1}\sum_{\tau=0}^{t}x_j(\lambda_j^{(\tau)})$$

and convexity of J_j and q_j, we obtain

$$(t+1)\left(J_j(x_j^{(t)}) + \left\langle q_j(x_j^{(t)}), \lambda\right\rangle + \psi_j(\lambda_j^{(t)})\right) \le \sum_{\tau=0}^{t}\left\langle -q_j(x_j(\lambda_j^{(\tau)})), \hat{\lambda}_j^{(\tau)} - \lambda\right\rangle.$$

By summing up the above inequality from $j=1$ to $j=n$ and following the same line with (3.21)-(3.23), we have

$$\begin{aligned}&(t+1)\sum_{j=1}^{n}\left(J_j(x_j^{(t)})+\left\langle q_j(x_j^{(t)}),\lambda\right\rangle+\psi_j(\lambda_j^{(t)})\right)\\ \leq& n\left(\sum_{\tau=0}^{t}\frac{D}{\gamma_{\tau-1}}\left(\frac{\sqrt{n}}{1-\sigma_2(P)}+\frac{5}{2}\right)+\frac{\gamma_t}{2}\|\lambda\|^2\right).\end{aligned}$$

Rewrite the above inequality as

$$\begin{aligned}&\sum_{j=1}^{n}\left(J_j(x_j^{(t)})-\left(-\psi_j(\lambda_j^{(t)})\right)\right)\\ \leq&\frac{1}{t+1}\left(\sum_{\tau=0}^{t}\frac{\left(\frac{\sqrt{n}}{1-\sigma_2(P)}+\frac{5}{2}\right)nD}{\gamma_{\tau-1}}+\min_{\lambda\geq 0}\left\{\frac{n\gamma_t}{2(t+1)}\|\lambda\|^2-\left\langle\sum_{j=1}^{n}q_j(x_j^{(t)}),\lambda\right\rangle\right\}\right),\end{aligned}$$

Due to

$$\min_{\lambda\geq 0}\left\{\frac{n\gamma_t}{2(t+1)}\|\lambda\|^2-\left\langle\sum_{j=1}^{n}q_j(x_j^{(t)}),\lambda\right\rangle\right\}=-\frac{t+1}{2n\gamma_t}\left\|\left(\sum_{j=1}^{n}q_j(x_j^{(t)})\right)_+\right\|^2$$

we have

$$\begin{aligned}&\sum_{j=1}^{n}\left(J_j(x_j^{(t)})-\left(-\psi_j(\lambda_j^{(t)})\right)\right)+\frac{t+1}{2n\gamma_t}\left\|\left(\sum_{j=1}^{n}q_j(x_j^{(t)})\right)_+\right\|^2\\ &\leq\frac{1}{t+1}\sum_{\tau=0}^{t}\frac{n\left(\frac{\sqrt{n}}{1-\sigma_2(P)}+\frac{5}{2}\right)D}{\gamma_{\tau-1}}\leq\frac{n\left(\frac{2\sqrt{n}}{1-\sigma_2(P)}+5\right)D}{\gamma\sqrt{t+1}}.\end{aligned}\tag{3.34}$$

Recall the saddle point inequality

$$J^*\leq\sum_{j=1}^{n}\left(J_j(x_j^{(t)})+\left\langle\lambda^*,q_j(x_j^{(t)})\right\rangle\right).\tag{3.35}$$

Upon adding $(t+1)\left\|\left(\sum_{j=1}^{n}q_j(x_j^{(t)})\right)_+\right\|^2/(2n\gamma_t)$ and subtracting $\sum_{j=1}^{n}\left(-\psi_j(\lambda_j^{(t)})\right)$

on both sides, we obtain

$$\begin{aligned}
&\sum_{j=1}^{n}\Big(J_j^*-\Big(-\psi_j(\lambda_j^{(t)})\Big)-\Big\langle\lambda^*,q_j(x_j^{(t)})\Big\rangle\Big)+\frac{t+1}{2n\gamma_t}\left\|\left(\sum_{j=1}^{n}q_j(x_j^{(t)})\right)_+\right\|^2\\
\leq&\sum_{j=1}^{n}\Big(J_j(x_j^{(t)})-\big(-\psi_j(\lambda_j^{(t)})\big)\Big)+\frac{t+1}{2n\gamma_t}\left\|\left(\sum_{j=1}^{n}q_j(x_j^{(t)})\right)_+\right\|^2\\
\leq&\frac{n\big(\frac{2\sqrt{n}}{1-\sigma_2(P)}+5\big)D}{\gamma\sqrt{t+1}}.
\end{aligned}$$

Since

$$\begin{aligned}
-\left\langle\lambda^*,\sum_{j=1}^{n}q_j(x_j^{(t)})\right\rangle+\frac{t+1}{2n\gamma_t}\left\|\left(\sum_{j=1}^{n}q_j(x_j^{(t)})\right)_+\right\|^2&\geq\min_{z\in\mathbb{R}^m}\left\{-\langle\lambda^*,z\rangle+\frac{t+1}{2n\gamma_t}\|(z)_+\|^2\right\}\\
&=-\frac{n\gamma_t}{2(t+1)}\|\lambda^*\|^2=-\frac{n\gamma\|\lambda^*\|^2}{2\sqrt{t+1}}
\end{aligned}$$

and $J^*=\sum_{j=1}^{n}-\psi_j(\lambda^*)$, we have

$$\sum_{j=1}^{n}\big(\psi_j(\lambda_j^{(t)})-\psi_j(\lambda^*)\big)\leq\frac{n}{\sqrt{t+1}}\left(\frac{\big(\frac{2\sqrt{n}}{1-\sigma_2(P)}+5\big)D}{\gamma}+\frac{\gamma\|\lambda^*\|^2}{2}\right).$$

By a similar reasoning to (3.22), we further have (3.30). To establish the upper bound on the violation of the coupled constraint, we consider $\forall\lambda\geq 0$,

$$J^*\geq\sum_{j=1}^{n}\big(J_j^*+\langle\lambda,q_j(x_j^*)\rangle\big)\geq\min_{\{x_j\in\mathcal{X}_j\}_{j=1}^n}\sum_{j=1}^{n}\left(J_j(x_j)+\langle\lambda,q_j(x_j)\rangle\right)=-\sum_{j=1}^{n}\psi_j(\lambda).\tag{3.36}$$

Upon using (3.34), we obtain

$$\frac{t+1}{2n\gamma_t}\left\|\left(\sum_{j=1}^{n}q_j(x_j^{(t)})\right)_+\right\|^2\leq\frac{n\big(\frac{2\sqrt{n}}{1-\sigma_2(P)}+5\big)D}{\gamma\sqrt{t+1}}+J^*-\min_{\{x_j\in\mathcal{X}_j\}_{j=1}^n}\sum_{j=1}^{n}J_j(x_j).$$

Therefore, (3.31) holds. By (3.34) and (3.36), we readily have

$$\sum_{j=1}^{n}\big(J_j(x_j^{(t)})-J_j^*\big)\leq\frac{n\big(\frac{2\sqrt{n}}{1-\sigma_2(P)}+5\big)D}{\gamma\sqrt{t+1}}.$$

Again, by the saddle point inequality (6.16), the fact

$$\big(\sum_{j=1}^{n}q_j(x_j^{(t)})\big)_+\geq\sum_{j=1}^{n}q_j(x_j^{(t)}),$$

and $\lambda^*\geq 0$, one obtains

$$\sum_{j=1}^{n}J_j(x_j^{(t)})-J^*\geq-\|\lambda^*\|\big\|\big(\sum_{j=1}^{n}q_j(x_j^{(t)})\big)_+\big\|\geq-\|\lambda^*\|\sqrt{\frac{2n^2\big(\frac{2\sqrt{n}}{1-\sigma_2(P)}+5\big)D}{t+1}+\frac{2n\gamma C}{\sqrt{t+1}}}.$$

This completes the proof. □

3.6 Experiment

In this section, we verify our theoretical findings by applying them to a setting where an MAS of $n=50$ agents aim to solving the following constraint-coupled decentralized optimization problem [59]:

$$\begin{aligned}&\min_{x_i\in[0,1]}\sum_{i=1}^{50}c_ix_i\\&\text{s.t.}\quad\sum_{i=1}^{50}-d_i\log(1+x_i)\leq-b.\end{aligned}$$

As explained in Section 3.5, we consider its Lagrangian

$$\sum_{i=1}^{n}\left(c_ix_i+\left\langle\lambda,\frac{b}{50}-d_i\log(1+x_i)\right\rangle\right)$$

and the corresponding dual problem

$$\min_{\lambda\geq0}\sum_{i=1}^{n}\psi_i(\lambda),$$

where

$$\psi_i(\lambda) = \max_{x_i \in [0,1]} \left\{ -c_i x_i - \left\langle \lambda, \frac{b}{50} - d_i \log(1 + x_i) \right\rangle \right\}.$$

In this simulation, the parameters c_i and d_i for each agent $i \in \{1, \cdots, 50\}$ are randomly chosen from a uniform distribution, and b is set as 5. We use the solver *fmincon* with the interior point algorithm in *Optimization Toolbox* in MATLAB to identify the optimal solution. The communication topology among agents is characterized by a fixed connected small world graph [101], and the weighting matrix P is selected as the Metropolis constant edge weight matrix [106]. The prox-function is chosen as $d(\lambda) = \|\lambda\|^2/2$. Set $\gamma_t = 0.2\sqrt{t+1}$. In the simulation, the bound for the subgradient of the dual objective is identified as 0.55, $\sigma_2(P)$ is calculated as 0.9788, and C is estimated as 27.2067. Therefore, the following theoretical bounds can be calculated based on Theorem 3.2:

$$\left| \sum_{j=1}^{n} J_j(x_j^{(t)}) - J^* \right| \leq \max \left\{ \frac{5.0849 \times 10^4}{\sqrt{t+1}}, \frac{91.5467}{\sqrt{t+1}} + \frac{6.9856}{(t+1)^{\frac{1}{4}}} \right\},$$

$$\left\| \left(\sum_{j=1}^{n} h_j(x_j^{(t)}) \right)_+ \right\|^2 \leq \frac{2.0340 \times 10^4}{t+1} + \frac{10.8827}{\sqrt{t+1}}.$$

For comparison, we also simulate the consensus-based dual decomposition strategies in [19,59,86]. For [19], according to the sufficient conditions for ensuring convergence, the step size is chosen as $10/(t+1)$. For [86], we use a constant step size 0.05. For [59], we derive the critical feasible step size for consensus-building according to Proposition 4 therein as $\sigma = 0.1103$, and use the Slater vector $(\mathbf{1}, 0)$ to get the bound on the optimal dual set as $D = 3.3130$. The step size is chosen by following the Doubling Trick scheme. All the algorithms are initialized with $\lambda_i^{(0)} = 0$ for all the agents.

The simulation results are illustrated in Figure 3.1. In particular, the left and the right plot present the trajectories of the primal objective error and the quadratic penalty for violation of the coupled constraint by all algorithms, respectively. Note that for the algorithms in [19,59,86], the performance is evaluated over the running average of the primal variables, which is in line with the theoretical results. We observe that the proposed algorithm demonstrates a slightly better performance than [59]. The trajectories of the proposed algorithm are within the theoretical upper bounds. Among the three algorithms, the algorithm in [19] has the slowest convergence. This

may be because that the step sizes for [59] and the proposed one are of order $1/\sqrt{t}$ while the step size for [19] is chosen to be of order $1/t$ to fulfill the conditions for convergence. The method in [86] does not ensure exact convergence partially due to using a constant step size, . Note that the value of $\sum_{i=1}^{50} c_i x_{i,t} - \sum_{i=1}^{50} c_i x_i^*$ can take both negative and positive values due to possible violation of the coupled constraint. When it jumps from negative to positive, the trajectory of $\left|\sum_{i=1}^{50} c_i x_{i,t} - \sum_{i=1}^{50} c_i x_i^*\right|$ presents a peak. This phenomenon is typically observed in dual Lagrangian problems.

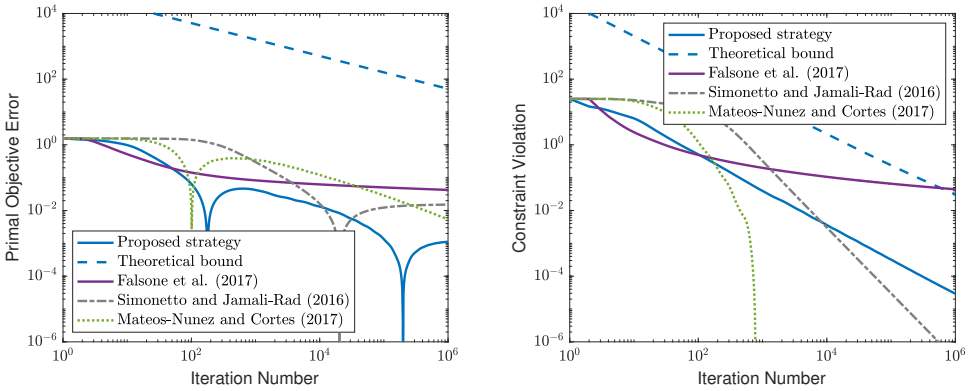


Figure 3.1: Trajectories of the primal objective error $\left|\sum_{i=1}^{50} c_i x_i^{(t)} - \sum_{i=1}^{50} c_i x_i^*\right|$ (left-hand side) and the quadratic penalty for the coupled constraint $\left\|\left(b - \sum_{i=1}^{50} d_i \log(1 + x_i^{(t)})\right)_+\right\|^2$.

3.7 Conclusion

In this chapter, we have proposed a decentralized subgradient method with double averaging, termed as DSA_2, for non-smooth cost-coupled optimization problems defined over networks. We proved a non-ergodic convergence rate of $\mathcal{O}(1/\sqrt{t})$ in terms of objective error for DSA_2. Furthermore, we have developed a DSA_2-based dual decomposition strategy for solving constraint-coupled decentralized optimization problems. We proved that the dual objective error and the quadratic penalty for violation of the coupled constraint converge at rate $\mathcal{O}(1/\sqrt{t})$. Simulation experiments and comparisons have been performed to verify the advantages of the proposed methods.

Chapter 4

Decentralized Dual Averaging Methods

4.1 Introduction

Consider a group of n agents, each of which has its own objective function. They are connected via a bidirectional communication network and aim to cooperatively solving the following convex composite optimization problem in a decentralized manner:

$$\min_{x\in\mathbb{R}^m}\left\{F(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x) + h(x)\right\}, \tag{4.1}$$

where f_i is the local smooth objective function of agent i and h is a non-smooth regularization term that is shared across all the agents. Problem (4.1) is referred to as decentralized convex composite optimization [1,84] and finds broad applications in optimal control of multi-agent systems [76], resource allocation [92], and large-scale machine learning [46], just to name a few.

In this chapter, we focus on solving Problem (4.1) when the communication network is *stochastic*. There are many practical reasons that promote the consideration of stochastic communication networks. Indeed, communication in real networks is usually subject to congestion, errors, and random dropouts, which is typically modeled as a stochastic process. Besides, stochastic networks are useful for proactively reducing communication cost. For instance, the gossip protocol [4] and Bernoulli protocol [31], which randomly choose a subset of communication links from an underlying dense graph in each iteration, have been widely regarded as effective strategies

to avoid high communication cost and network congestion. Therefore, it is highly desirable to develop decentralized algorithms that solve Problem (4.1) over stochastic communication networks and attain a favorable convergence rate.

Over the past decade, many algorithms have been proposed for solving Problem (4.1). Some of them exploit the composite structure in (4.1) and attain global *linear* convergence if Problem (4.1) is strongly convex (see, e.g., [1, 36]), which is the fastest rate of convergence that one can expect from a first-order decentralized algorithm. However, such linear convergence results are limited to *time-invariant* communication networks, because the design of these algorithms inherently requires knowledge of network topology *a priori*. Indeed, these algorithms are typically developed upon leveraging centralized primal-dual optimization paradigms, such as the alternating direction method of multipliers [5], to solve the following problem that is equivalent to (4.1):

$$\min_{x_1,\ldots,x_n\in\mathbb{R}^m} \ \frac{1}{n}\sum_{i=1}^{n}\Big(f_i(x_i)+h(x_i)\Big) \quad \text{s.t. } (\mathcal{L}\otimes I)\mathbf{x}=0, \tag{4.2}$$

where $\mathbf{x}=[x_1^T,\ldots,x_n^T]^T$, I is an identity matrix of size $m\times m$, and $\mathcal{L}$ denotes the graph Laplacian associated with the communication network. Since $\mathcal{L}$ needs to be explicitly given in formulation (4.2), these algorithms and their associated linear convergence results cannot be extended to stochastic communication networks, where the network topology is time-varying and random.

Among the existing decentralized optimization methods, the decentralized dual averaging (DDA) algorithm proposed by [18] and its later extensions [12, 38, 89] have been recognized as a powerful framework that can handle stochastic networks. However, the convergence rates of existing DDA-type algorithms are rather slow. In fact, even for decentralized convex *smooth* optimization in time-invariant networks, which is deemed to be much simpler than Problem (4.1) in stochastic networks, these algorithms were only known to converge *sublinearly*. Specifically, existing DDA-type algorithms, when applied to Problem (4.1), only attain an $\mathcal{O}(1/\sqrt{t})$ sublinear rate of convergence. For the special case of Problem (4.1) with $h\equiv 0$, [50] recently showed that the convergence rate can be improved to $\mathcal{O}(1/t)$. Nevertheless, it remains open whether a DDA-type algorithm can attain linear rate of convergence.

Contribution. In this chapter, we propose a new DDA algorithm that solves Problem (4.1) in stochastic networks. Under a rather mild condition on the stochastic network, we show that the proposed algorithm has an $\mathcal{O}(1/t)$ rate of convergence in the general case and a global linear rate of convergence if each local objective function is strongly convex. Our work contributes to the literature of decentralized optimization in the following two aspects:

i) We develop the first decentralized algorithm that attains global linear convergence for solving Problem(4.1) in stochastic networks. Existing linearly convergent decentralized algorithms for Problem (4.1) only work in time-invariant networks and cannot be extended to stochastic networks because they inherently need knowledge of network topology *a priori*. Our algorithm is based on a DDA framework that is fundamentally different from these algorithms.

ii) Our algorithmic design and convergence analysis shed new light on DDA-type algorithms. Notably, it is the first DDA-type algorithm that attains linear convergence. Prior to our work, even for decentralized convex *smooth* optimization in time-invariant networks, which is deemed to be much simpler than Problem (4.1) in stochastic networks, existing DDA-type algorithms were only known to converge sublinearly. The key to achieving the improved rate is the design of a novel dynamic averaging consensus protocol for DDA, which intuitively leads to more accurate local estimates of the global dual variable.

4.2 Related Work

Decentralized algorithms for Problem (4.1) **in time-invariant networks.** Due to its broad applications, Problem (4.1) has received attention in the community of decentralized optimization for many years; see, e.g., [84] for an early attempt. It is only until recently that linearly convergent decentralized algorithms have been developed for solving Problem (4.1) in time-invariant networks. [1] developed a decentralized proximal gradient method, where the diffusion step and the proximal step are designed differently from [84] such that not only the fixed point meets the global optimality condition but also linear convergence can be attained for strongly convex problems. [36] proposed a distributed algorithm based on randomized block-coordinate proximal method, which exhibits an asymptotic linear convergence if the monotone operator associated with Problem (4.1) is metrically subregular (a much

weaker condition than strong convexity). Very recently, [110] proposed a unified decentralized algorithmic framework based on the operator splitting theory, which attains linear convergence for the strongly convex case. Nevertheless, these algorithms are only applicable to time-invariant networks and cannot be extended to the stochastic networks, which motivates the new algorithm development and convergence analysis in this paper.

Decentralized optimization in stochastic networks. The study of decentralized algorithms over stochastic networks dates back to [55], who proposed a subgradient-based algorithm with diminishing step sizes. The decentralized dual averaging algorithm, which combines dual averaging method [69] and consensus-seeking, was reported by [18] and can handle stochastic networks with an $\mathcal{O}(1/\sqrt{t})$ sublinear rate of convergence. The decentralized accelerated gradient algorithm with a random network model was proposed by [27], where an $\mathcal{O}(\frac{\log t}{t})$ sublinear convergence rate is obtained for smooth problems. Later, [111] validated the use of a constant step size in decentralized gradient descent over stochastic networks, leading to a global linear rate of convergence for strongly convex and smooth problems. Recently, [33] developed a unified framework for decentralized stochastic gradient descent over stochastic networks. It is worth mentioning that the aforementioned studies either consider general non-smooth problems or focus on smooth problems. In particular, they cannot exploit the composite structure of Problem (4.1), partially due to the technical difficulty caused by the so-called *projection-consensus coupling* [18] for methods integrating consensus-seeking and projected/proximal gradient descent.

In summary, to the best of our knowledge, no existing methods can solve or can be easily extended to solve Problem (4.1) in stochastic networks with global linear convergence.

4.3 Problem Setup and Preliminaries

4.3.1 Basic Setup

We consider the finite-sum optimization problem (4.1), in which $h : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a proper closed convex function with its domain, denoted by $\mathrm{dom}(h) := \{x \in \mathbb{R}^m | h(x) < +\infty\}$, being non-empty and f_i satisfies the following assumptions for all $i = 1, \ldots, n$:

Assumption 4.1. *i) f_i is continuously differentiable on an open set that contains $dom(h)$;*

ii) f_i is (strongly) convex with modulus $\mu \geq 0$ on $dom(h)$;

iii) ∇f_i is Lipschitz continuous on $dom(h)$ with Lipschitz constant $L > 0$.

The above assumptions are standard in the study of decentralized algorithms for convex optimization problems. It is worth noting that we allow $\mu = 0$ in Assumption 4.1(ii), which reduces to the general convex case. Throughout the paper, we denote by x^* an optimal solution of Problem (4.1).

4.3.2 Stochastic Communication Network

We consider solving Problem (4.1) in a decentralized manner, that is, each agent i holds a local objective function $F_i := f_i + h$ and a pair of agents can exchange information only if they are connected in the communication network. Similar to existing works, we use a doubly stochastic matrix $P^{(t)} \in [0,1]^{n\times n}$ to encode the network topology and the averaging weights of connected links at time t. We focus on the fairly general setting of stochastic communication network, i.e., $P^{(t)}$ is a random matrix for every t. For the convergence of the proposed decentralized algorithm, we make the following assumption on $P^{(t)}$.

Assumption 4.2. *For every $t \geq 0$, it holds that*

i) $P^{(t)}\mathbf{1} = \mathbf{1}$ and $\mathbf{1}^T P^{(t)} = \mathbf{1}^T$;

ii) $P^{(t)}$ is independent of the random events that occur up to time $t-1$;

iii) there exists a constant $\beta \in (0,1)$ such that

$$\sqrt{\rho\left(\mathbb{E}_t\left[{P^{(t)}}^T P^{(t)}\right] - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)} \leq \beta, \tag{4.3}$$

where $\rho(\cdot)$ denotes the spectral radius and the expectation $\mathbb{E}_t[\cdot]$ is taken with respect to the distribution of $P^{(t)}$ at time t.

Assumption 4.2 has been used for analyzing the convergence of a host of decentralized algorithms; see, e.g., [4, 33, 111]. An example of $\{P^{(t)}\}_{t\geq 0}$ that satisfies

Assumption 4.2 is in the random gossip setting, where at time t, one communication link (i, j) is sampled from an underlying graph $\mathcal{G}$. Suppose that we take $P^{(t)} = I - \frac{1}{2}(e_i - e_j)(e_i - e_j)^T$, where I is the identity matrix and $e_i \in \mathbb{R}^n$ is a vector with 1 in the i-th position and 0 otherwise. Then, it is known that Assumption 4.2 is satisfied provided that the underlying graph $\mathcal{G}$ is connected; see, e.g., [4].

4.3.3 Centralized Dual Averaging Method

Our algorithm is based on the dual averaging method that was originally proposed in [69]. It can be directly applied to solving the considered Problem (4.1) in a *centralized* manner. In particular, let d be a strongly convex function with modulus 1 on $\mathrm{dom}(h)$ such that

$$x^{(0)} = \operatorname*{argmin}_{x \in \mathbb{R}^m} d(x) \in \mathrm{dom}(h) \quad \text{and} \quad d(x^{(0)}) = 0. \tag{4.4}$$

Then, the dual averaging method starts with $x^{(0)}$ and iteratively generates $\{x^{(t)}\}_{t \geq 1}$ according to

$$x^{(t)} = \operatorname*{argmin}_{x \in \mathbb{R}^m} \left\{ \sum_{\tau=0}^{t-1} a_{\tau+1} \ell(x; x^{(\tau)}) + d(x) \right\}, \tag{4.5}$$

where

$$a_t = \frac{a}{(1 - a\mu)^t}, \quad t = 1, 2, \ldots \tag{4.6}$$

for some constant $a > 0$, $\ell : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ is defined as

$$\ell(y; z) := f(z) + \langle \nabla f(z), y - z \rangle + \frac{\mu}{2} \|y - z\|^2 + h(y) \tag{4.7}$$

for any $y, z \in \mathbb{R}^m$, and $f = \frac{1}{n} \sum_{i=1}^n f_i$. It is worth noting that for the strongly convex case (i.e., $\mu > 0$), the sequence $\{a_t\}_{t \geq 1}$ is geometrically increasing; for the general convex case (i.e., $\mu = 0$), the sequence $\{a_t\}_{t \geq 1}$ equals the constant a. Moreover, both (4.5) and (4.6) requires the modulus μ of strong convexity. In practice, one can use a lower bound of μ or simply set $\mu = 0$ in (4.5) and (4.6) if no valid lower bound is available.

The following theorem summarizes the convergence property of the above dual averaging method, which is a direct extension of Theorem 3.2 in [56] to problems with non-smooth regularization terms. For completeness, we provide a proof of Theorem 4.1.

Theorem 4.1. *Suppose that Assumption 4.1 is satisfied. Let $\{x^{(t)}\}_{t\geq 0}$ be the sequence of iterates generated by the dual averaging method (4.5). If $a \leq L^{-1}$, then*

$$F(\tilde{x}^{(t)}) - F(x^*) \leq \frac{d(x^*)}{A_t}, \quad t = 1, 2, \ldots,$$

where $A_t = \sum_{\tau=1}^{t} a_\tau$ and $\tilde{x}^{(t)} = A_t^{-1} \sum_{\tau=1}^{t} a_\tau x^{(\tau)}$. Moreover, the following estimates on A_t^{-1} holds:

i) If $\mu > 0$, then

$$\frac{1}{A_t} \leq \frac{(1 - a\mu)^t}{a}.$$

ii) If $\mu = 0$, then

$$\frac{1}{A_t} = \frac{1}{at}.$$

From Theorem 4.1, one can observe that the dual averaging method, when applied to solving Problem (4.1) in a centralized manner, attains global linear convergence if $\mu > 0$ and global $\mathcal{O}(1/t)$ convergence rate if $\mu = 0$.

Before presenting the proof of Theorem 4.1, we first provide a technical lemma.

Lemma 4.1. *Suppose that the premise of Theorem 4.1 holds. For the sequence $\{x^{(t)}\}_{t\geq 0}$ generated by (4.5), it holds that*

$$\begin{aligned} &\sum_{\tau=1}^{t} a_\tau \left(\langle \nabla f(x^{(\tau-1)}), x^{(\tau)} - x^* \rangle + h(x^{(\tau)}) - h(x^*) \right) \\ &\leq d(x^*) - \frac{1}{2} \sum_{\tau=1}^{t} \left(1 + \mu A_\tau\right) \|x^{(\tau)} - x^{(\tau-1)}\|^2 - \frac{\mu}{2} \sum_{\tau=1}^{t} a_\tau \|x^{(\tau-1)} - x^*\|^2. \end{aligned} \tag{4.8}$$

Proof. We define

$$r_t(x) := \sum_{\tau=0}^{t-1} a_{\tau+1} \ell(x; x^{(\tau)}) + d(x), \quad t = 0, 1, \ldots,$$

where $r_0(x) = d(x)$ and $\ell(x; x^{(\tau)})$ is defined in (4.7). It then follows that for any $\tau \geq 1$,

$$r_\tau(x) = r_{\tau-1}(x) + a_\tau \left(\langle \nabla f(x^{(\tau-1)}), x \rangle + \frac{\mu}{2} \|x - x^{(\tau-1)}\|^2 + h(x) \right). \tag{4.9}$$

By (4.5), we know that $x^{(\tau-1)} = \operatorname{argmin}_{x\in\mathbb{R}^m} r_{\tau-1}(x)$. Moreover, $r_{\tau-1}(x)$ is strongly convex with modulus $1+\mu A_{\tau-1}$. Then, we obtain

$$r_{\tau-1}(x) - r_{\tau-1}(x^{(\tau-1)}) \geq \frac{1}{2}(1+\mu A_{\tau-1})\|x - x^{(\tau-1)}\|^2, \quad \forall x \in \operatorname{dom}(h).$$

Therefore,

$$\begin{aligned}
0 \leq & r_{\tau-1}(x^{(\tau)}) - r_{\tau-1}(x^{(\tau-1)}) - \frac{1}{2}\big(1+\mu A_{\tau-1}\big)\|x^{(\tau)} - x^{(\tau-1)}\|^2 \\
= & r_\tau(x^{(\tau)}) - a_\tau\left(\langle \nabla f(x^{(\tau-1)}), x^{(\tau)}\rangle + \frac{\mu}{2}\|x^{(\tau)} - x^{(\tau-1)}\|^2 + h(x^{(\tau)})\right) \\
& - r_{\tau-1}(x^{(\tau-1)}) - \frac{1}{2}\big(1+\mu A_{\tau-1}\big)\|x^{(\tau)} - x^{(\tau-1)}\|^2,
\end{aligned}$$

where the equality follows from (4.9). This, together with $A_\tau = A_{\tau-1} + a_\tau$, leads to

$$a_\tau\left(\langle \nabla f(x^{(\tau-1)}), x^{(\tau)}\rangle + h(x^{(\tau)})\right) \leq r_\tau(x^{(\tau)}) - r_{\tau-1}(x^{(\tau-1)}) - \frac{1}{2}\big(1+\mu A_\tau\big)\|x^{(\tau)} - x^{(\tau-1)}\|^2.$$

Summing up the above inequality from $\tau = 1$ to $\tau = t$ yields

$$\begin{aligned}
& \sum_{\tau=1}^{t} a_\tau\left(\langle \nabla f(x^{(\tau-1)}), x^{(\tau)}\rangle + h(x^{(\tau)})\right) \\
& \leq r_t(x^{(t)}) - r_0(x^{(0)}) - \sum_{\tau=1}^{t}\frac{1}{2}\big(1+\mu A_\tau\big)\|x^{(\tau)} - x^{(\tau-1)}\|^2 \qquad (4.10) \\
& = r_t(x^{(t)}) - \sum_{\tau=1}^{t}\frac{1}{2}\big(1+\mu A_\tau\big)\|x^{(\tau)} - x^{(\tau-1)}\|^2,
\end{aligned}$$

where the equality follows from $r_0(x) = d(x)$ and (5.2). Then, we turn to consider

$$\begin{aligned}
&\sum_{\tau=1}^{t} a_\tau \langle \nabla f(x^{(\tau-1)}), -x^* \rangle \\
\leq &\max_{x\in\mathbb{R}^m} \left\{ \sum_{\tau=1}^{t} a_\tau \left(\langle \nabla f(x^{(\tau-1)}), -x \rangle - \frac{\mu}{2}\|x - x^{(\tau-1)}\|^2 - h(x) \right) - d(x) \right\} \\
&+ d(x^*) + \sum_{\tau=1}^{t} a_\tau \left(\frac{\mu}{2}\|x^{(\tau-1)} - x^*\|^2 + h(x^*) \right) \\
= &- \min_{x\in\mathbb{R}^m} \left\{ \sum_{\tau=1}^{t} a_\tau \left(\langle \nabla f(x^{(\tau-1)}), x \rangle + \frac{\mu}{2}\|x - x^{(\tau-1)}\|^2 + h(x) \right) + d(x) \right\} \\
&+ d(x^*) + \sum_{\tau=1}^{t} a_k \left(\frac{\mu}{2}\|x^{(k-1)} - x^*\|^2 + h(x^*) \right) \\
= &- r_t(x^{(t)}) + d(x^*) + \sum_{\tau=1}^{t} a_\tau \left(\frac{\mu}{2}\|x^{(\tau-1)} - x^*\|^2 + h(x^*) \right).
\end{aligned} \tag{4.11}$$

Upon summing up (4.10) and the above inequality, we obtain (4.8) as desired. □

Proof of Theorem 4.1. Recall that $f = \frac{1}{n}\sum_{i=1}^{n} f_i$. Using (2.3) and (2.1) sequentially, we have

$$\begin{aligned}
&a_\tau \left(f(x^{(\tau)}) - f(x^*) \right) \\
&\leq a_\tau \left(\frac{L}{2}\|x^{(\tau)} - x^{(\tau-1)}\|^2 + f(x^{(\tau-1)}) + \langle \nabla f(x^{(\tau-1)}), x^{(\tau)} - x^{(\tau-1)} \rangle - f(x^*) \right) \\
&\leq a_\tau \Big(\frac{L}{2}\|x^{(\tau)} - x^{(\tau-1)}\|^2 + \left\langle \nabla f(x^{(\tau-1)}), x^{(\tau)} - x^* \right\rangle - \frac{\mu}{2}\|x^{(\tau-1)} - x^*\|^2 \Big).
\end{aligned}$$

Upon summing up the above inequality from $\tau = 1$ to $\tau = t$ and using Lemma 4.1 and $F = f + h$, we obtain

$$\begin{aligned}
&\sum_{\tau=1}^{t} a_\tau \left(F(x^{(\tau)}) - F(x^*) \right) \\
&\leq \sum_{\tau=1}^{t} \left(\frac{a_\tau}{2} \left(L\|x^{(\tau)} - x^{(\tau-1)}\|^2 - \frac{1 + \mu A_\tau}{a_\tau}\|x^{(\tau)} - x^{(\tau-1)}\|^2 \right) \right) + d(x^*).
\end{aligned}$$

According to (4.6) and $A_t = \sum_{\tau=1}^{t} a_\tau$, one has

$$\frac{1+\mu A_\tau}{a_\tau} = \frac{(\frac{1}{1-a\mu})^\tau}{\frac{a}{1-a\mu}\left(\frac{1}{1-a\mu}\right)^{\tau-1}} = \frac{1}{a}. \tag{4.12}$$

By substituting this into the above inequality and using the condition $a \leq L^{-1}$, we obtain

$$\sum_{\tau=1}^{t} a_\tau \left(F(x^{(\tau)}) - F(x^*)\right) \leq \left(L - \frac{1}{a}\right) \sum_{\tau=1}^{t} \frac{a_\tau}{2} \|x^{(\tau)} - x^{(\tau-1)}\|^2 + d(x^*) \leq d(x^*).$$

Upon dividing both sides of the above inequality by A_t and using the convexity of F and $\tilde{x}^{(t)} = A_t^{-1} \sum_{\tau=1}^{t} a_\tau x^{(\tau)}$, we obtain

$$F(\tilde{x}^{(t)}) - F(x^*) \leq \frac{d(x^*)}{A_t}.$$

Now it remains to show the statements i) and ii) in Theorem 4.1. By the definitions of a_t and A_t, we readily have $A_t = at$ when $\mu = 0$ and

$$\frac{1}{A_t} = \frac{\mu}{(\frac{1}{1-a\mu})^t - 1}$$

when $\mu > 0$. Moreover, by $0 < a < L^{-1}$ and $L \geq \mu$, one has $0 < a\mu < 1$ when $\mu > 0$. This, together with the above identity, yields that when $\mu > 0$,

$$\frac{1}{A_t} = \frac{\mu}{(\frac{1}{1-a\mu})^t - 1} = \frac{\mu(1-a\mu)^t}{1-(1-a\mu)^t} \leq \frac{\mu(1-a\mu)^t}{1-(1-a\mu)} = \frac{(1-a\mu)^t}{a}.$$

This completes the proof. □

4.4 Algorithm and Convergence Results

The standard dual averaging method (4.5) requires the computation of $\sum_{i=1}^{n} \nabla f_i(x^{(t)})$ at every iteration t. Thus, it cannot be executed in a *decentralized* manner, where communication can only occur between each connected pair of agents. In this section, we propose a *decentralized dual averaging* (DDA) method for solving Problem (4.1) and show that it has a nice convergence guarantee that is similar to its centralized

counterpart. In particular, we show that if $\mu > 0$, then the proposed DDA method attains a global linear rate of convergence. To the best of our knowledge, this is the first global linear convergence result for decentralized composite optimization problems in stochastic networks.

To motivate the design of DDA, observe that by letting $A_t = \sum_{\tau=1}^{t} a_\tau$ and

$$z^{(t)} = \sum_{\tau=0}^{t-1} a_{\tau+1} \left(\frac{1}{n} \sum_{i=1}^{n} \nabla f_i(x^{(\tau)}) - \mu x^{(\tau)} \right),$$

the update rule (4.5) can be written as

$$x^{(t)} = \operatorname*{argmin}_{x \in \mathbb{R}^m} \left\{ \langle z^{(t)}, x \rangle + A_t \left(\frac{\mu}{2} \|x\|^2 + h(x) \right) + d(x) \right\}.$$

Thus, it is sensible for each agent to locally estimate the global variable $z^{(t)}$ to fulfill decentralization. To this end, we propose the following consensus-based estimation protocol:

$$z_i^{(t)} = \sum_{j=1}^{n} p_{ij}^{(t-1)} \left(z_j^{(t-1)} + a_t s_j^{(t-1)} \right), \tag{4.13a}$$

$$s_i^{(t)} = \sum_{j=1}^{n} p_{ij}^{(t-1)} s_j^{(t-1)} + \left(\nabla f_i(x_i^{(t)}) - \mu x_i^{(t)} \right) - \left(\nabla f_i(x_i^{(t-1)}) - \mu x_i^{(t-1)} \right),$$

where $p_{ij}^{(t)}$ is the (i,j)-th element in the mixing matrix $P^{(t)}$, $z_i^{(t)}$ is the i-th agent's local estimate of $z^{(t)}$ at time t and $s_i^{(t)}$ is an auxiliary vector for reducing consensus error. Equipped with these, each agent i can perform a local computation to update its estimate of the global variable $x^{(t)}$:

$$x_i^{(t)} = \operatorname*{argmin}_{x \in \mathbb{R}^m} \left\{ \langle z_i^{(t)}, x \rangle + A_t \left(\frac{\mu}{2} \|x\|^2 + h(x) \right) + d(x) \right\}. \tag{4.14}$$

We denote by $\mathcal{N}_i^{(t)}$ the set of agents that are connected with agent i at time t. Then, the entire algorithm can be summarized in Algorithm 3.

Before proceeding, we make some remarks on Algorithm 3:

i) Algorithm 3 provides a unified treatment for both general convex and strongly convex cases. In particular, if $\mu = 0$, we simply set $a_t = a$ and $A_t = at$ for all t.

ii) To satisfy the condition in (5.2), one can choose an arbitrary $x^{(0)} \in \mathrm{dom}(h)$ and

Algorithm 3 Decentralized Dual Averaging for Problem (4.1)

Input: $\mu \geq 0$, $a > 0$, $x^{(0)} \in \mathrm{dom}(h)$ and a strongly convex function d with modulus 1 on $\mathrm{dom}(h)$ such that (5.2) holds
Initialize: $a_0 = a$, $A_0 = 0$, $x_i^{(0)} = x^{(0)}$, $z_i^{(0)} = 0$, and $s_i^{(0)} = \nabla f_i(x^{(0)}) - \mu x^{(0)}$ for all $i = 1, \ldots, n$
for $t = 1, 2, \cdots$ **do**
  set $a_t = a_{t-1}/(1 - a\mu)$ and $A_t = A_{t-1} + a_t$
  In parallel (task for agent i, $i = 1, \ldots, n$)
  collect $z_j^{(t-1)}$ and $s_j^{(t-1)}$ from all agents $j \in \mathcal{N}_i^{(t-1)}$
  update $z_i^{(t)}$ and $s_i^{(t)}$ by (4.13)
  compute $x_i^{(t)}$ by (4.14)
  broadcast $z_i^{(t)}$ and $s_i^{(t)}$ to all agents $j \in \mathcal{N}_i^{(t)}$
end for

let

$$d(x) := \tilde{d}(x) - \tilde{d}(x^{(0)}) - \langle \nabla \tilde{d}(x^{(0)}), x - x^{(0)} \rangle,$$

where $\tilde{d}$ is any strongly convex function with modulus 1, e.g., $\tilde{d}(x) = \|x\|^2/2$. It is easy to verify that such $x^{(0)}$ and d satisfy (5.2).

iii) Similar to the standard dual averaging method, we assume that the subproblem (4.14) can be computed easily. This holds for a host of applications. For example, if we choose $d(x) = \|x - x^{(0)}\|^2$, then the subproblem (4.14) reduces to computing the proximal operator of h, which admits a closed-form solution in many applications.

iv) Compared to existing decentralized optimization algorithms, a different dynamic consensus protocol is tailored within the dual averaging framework. With it, each agent is able to track $\frac{1}{n}\sum_{i=1}^n \nabla f_i(x^{(t)})$ and $\mu x^{(t)}$ simultaneously. Another notable feature is that when $\mu > 0$, the estimated information $\{s_j^{(t-1)} : j \in \mathcal{N}_i^{(t-1)}\}$ enters the consensus model $z_i^{(t)}$ with exponentially increasing weights in (4.13a), and therefore necessitates new analysis in quantifying the consensus error.

Next, we present the convergence guarantee of Algorithm 3. To proceed, we introduce the following 2×2 matrix:

$$\mathbf{M} = \begin{bmatrix} \beta & \beta \\ \frac{a(L+\mu)}{1-a\mu}\left(\beta + \frac{1}{1-a\mu}\right) & \frac{\beta + a\beta(L+\mu)}{1-a\mu} \end{bmatrix}, \tag{4.15}$$

where L and μ are given in Assumption 4.1, $\beta \in (0,1)$ is defined in Assumption 4.2, and a is an input of Algorithm 3. The matrix $\mathbf{M}$ is key to our convergence analysis as it defines the dynamic of the iterates generated by Algorithm 3. Let $\rho(\mathbf{M})$ be the spectral radius of $\mathbf{M}$. To facilitate the presentation of our convergence analysis, we define

$$\begin{aligned} \nu &:= \rho(\mathbf{M})\sqrt{1-a\mu}, \\ \eta &:= (1-a\mu)(1-\nu)^2, \\ \theta &:= (1-a\mu)(1-\nu^2). \end{aligned} \tag{4.16}$$

The following result on ν, η, and θ is fundamental to our convergence analysis, whose proof can be found in Section 4.6.1.

Lemma 4.2. *The value of ν monotonically increases with a if $a \in (0, 1/\mu)$. Moreover, if*

$$\frac{1}{a} > \frac{\beta(2L+3\mu)}{(1-\beta)^2} + \mu, \tag{4.17}$$

then $\nu < 1$. Consequently, η and θ are both positive and monotonically decrease with a if (4.17) *is satisfied.*

Equipped with Lemma 4.2, we are ready to present the main results of this paper, which pertain to the convergence property of Algorithm 3. Similar to some existing works, we first present the convergence property of an auxiliary sequence $\{y^{(t)}\}_{t\geq 0}$, which would then immediately imply the convergence property of the sequence $\{x_i^{(t)} : i = 1, \ldots, n\}_{t\geq 0}$ generated by Algorithm 3. In particular, we define

$$y^{(t)} = \operatorname*{argmin}_{x\in\mathbb{R}^m} \left\{ \langle \overline{z}^{(t)}, x\rangle + A_t \left(\frac{\mu}{2}\|x\|^2 + h(x) \right) + d(x) \right\}, \tag{4.18}$$

where $y^{(0)} = x^{(0)}$, $\overline{z}^{(t)} = \frac{1}{n}\sum_{i=1}^n z_i^{(t)}$ and $\{z_i^{(t)} : i = 1, \ldots, n\}_{t\geq 0}$ are generated by Algorithm 3.

Theorem 4.2. *Suppose that Assumptions 4.1 and 4.2 are satisfied. Besides, suppose that the constant a in Algorithm 3 satisfies* (4.17) *and*

$$\gamma := \frac{1}{a} - 2L + \mu - \frac{4L - 2\mu}{\eta} > 0, \tag{4.19}$$

where η is defined in (4.16)*. Then, for all $t \geq 1$, it holds that*

$$\mathbb{E}[F(\tilde{y}^{(t)})] - F(x^*) \leq \frac{C}{A_t}, \tag{4.20}$$

where $\tilde{y}^{(t)} = A_t^{-1}\sum_{\tau=1}^{t} a_\tau y^{(\tau)}$ *with* $y^{(\tau)}$ *defined in* (4.18),

$$C := d(x^*) + \frac{a\big(2L-\mu\big)\sigma^2}{n\theta(L+\mu)^2} > 0,$$

and σ^2 *is the variance of local gradients at* $t = 0$, *i.e.,*

$$\sigma^2 = \sum_{i=1}^{n}\left\|\nabla f_i(x^{(0)}) - \frac{1}{n}\sum_{j=1}^{n}\nabla f_j(x^{(0)})\right\|^2.$$

Moreover, for all $t \geq 1$ *and* $i = 1, \ldots, n$, *we have*

$$\mathbb{E}[\|\tilde{x}_i^{(t)} - \tilde{y}^{(t)}\|^2] \leq \frac{D}{A_t}, \tag{4.21}$$

where $\tilde{x}_i^{(t)} = A_t^{-1}\sum_{\tau=1}^{t} a_\tau x_i^{(\tau)}$ *and*

$$D := \frac{4nC}{\eta\gamma} + \frac{2a\sigma^2}{\theta(L+\mu)^2} > 0.$$

Theorem 4.2 can be regarded as a decentralized counterpart of Theorem 4.1. Due to the presence of consensus error in the decentralized setting, Theorem 4.2 requires a more delicate choice of a for convergence. Nevertheless, an a that satisfies the condition in Theorem 4.2 always exists. In particular, upon using Lemma 4.2, it is not hard to observe that there exists an $\bar{a} \in (0, \mu^{-1})$ such that (4.17) and (4.19) are both satisfied for any $a \in (0, \bar{a})$. While finding the largest possible $\bar{a}$ is difficult because it requires solving a nonlinear equation associated with (4.19), a conservative estimation of $\bar{a}$ can be obtained. Recall $\rho(\mathbf{M}) = \lambda_1 = (\xi_1 + \xi_2)/2$, where ξ_1, ξ_2 are defined in (4.40). Then, one can verify that by taking $a = 1/(2\mu)$, we have

$$\eta\left(\frac{1}{2\mu}\right) = \frac{\left(1 - \beta(\sqrt{2} + \frac{L}{2\sqrt{2}\mu}) - \sqrt{\frac{\beta^2L^2}{8\mu^2} + \beta(\beta+1)(1+\frac{L}{\mu})}\right)^2}{2}.$$

We have shown in Lemma 4.2 that η decreases with a if (4.17) is satisfied, so

$$\eta(a) > \eta\left(\frac{1}{2\mu}\right)$$

for all a satisfying $0 < a < 1/(2\mu)$ and (4.17). Then, as long as a satisfies

$$\begin{aligned}
&\frac{1}{a} > \frac{\beta(2L+3\mu)}{(1-\beta)^2} + \mu, \\
&\frac{1}{a} > 2L - \mu + \frac{4L-2\mu}{\eta(\frac{1}{2\mu})} = 2L - \mu + \frac{8L-4\mu}{\left(1-\beta(\sqrt{2}+\frac{L}{2\sqrt{2}\mu}) - \sqrt{\frac{\beta^2L^2}{8\mu^2} + \beta(\beta+1)(1+\frac{L}{\mu})}\right)^2}, \\
&\frac{1}{a} > 2\mu,
\end{aligned} \tag{4.22}$$

then a also satisfies (4.17) and (4.19). This implies that we can take

$$\overline{a} = \min\left\{ \frac{1}{2\mu}, \frac{1}{\frac{\beta(2L+3\mu)}{(1-\beta)^2}+\mu}, \frac{\left(1-\beta(\sqrt{2}+\frac{L}{2\sqrt{2}\mu}) - \sqrt{\frac{\beta^2L^2}{8\mu^2} + \beta(\beta+1)(1+\frac{L}{\mu})}\right)^2}{(2L-\mu)\left(4+\left(1-\beta(\sqrt{2}+\frac{L}{2\sqrt{2}\mu}) - \sqrt{\frac{\beta^2L^2}{8\mu^2} + \beta(\beta+1)(1+\frac{L}{\mu})}\right)^2\right)} \right\}.$$

It would be interesting to estimate the order of $\bar{a}$ when the condition number $\kappa = L/\mu$ goes to ∞ and the β, which relates to the connectivity of the stochastic network, goes to 1. By the standard limiting argument, one can verify that the dominating term inside the above brace is the second term, which is in the order $\mathcal{O}((1-\beta)^2/L)$.

As a consequence of Theorem 4.2, we show that Algorithm 3 attains a global linear rate of convergence if $\mu > 0$.

Corollary 4.1. *Suppose that the premise of Theorem 4.2 holds. If $\mu > 0$, then for all $t \geq 1$ and $i = 1, \ldots, n$, we have*

$$\mathbb{E}[\|\tilde{x}_i^{(t)} - x^*\|^2] \leq \frac{2}{a}\left(\frac{2C}{\mu} + D\right)(1-a\mu)^t, \tag{4.23}$$

where $\tilde{x}_i^{(t)} = A_t^{-1}\sum_{\tau=1}^t a_\tau x_i^{(\tau)}$ and C, D are positive constants given in Theorem 4.2.

Moreover, for the case $\mu = 0$, Theorem 4.2 implies that Algorithm 3 has a global $\mathcal{O}(1/t)$ rate of convergence.

Corollary 4.2. *Suppose that the premise of Theorem 4.2 holds. If $\mu = 0$, then for all $t \geq 1$ and $i = 1, \ldots, n$, we have*

$$\mathbb{E}[F(\tilde{y}^{(t)})] - F(x^*) \leq \frac{C}{at}, \tag{4.24}$$

$$\mathbb{E}[\|\tilde{x}_i^{(t)} - \tilde{y}^{(t)}\|^2] \leq \frac{D}{at}, \tag{4.25}$$

where $\tilde{y}^{(t)} = \frac{1}{t}\sum_{\tau=1}^{t} y^{(\tau)}$, $\tilde{x}_i^{(t)} = \frac{1}{t}\sum_{\tau=1}^{t} x_i^{(\tau)}$, *and* C, D *are positive constants given in Theorem 4.2. In addition, if* $h \equiv 0$ *in Problem* (4.1), $d(x) = \|x\|^2/2$, *and*

$$\frac{1}{a} > 2L \cdot \max\left\{\frac{\beta}{(1-\beta)^2}, 1 + \frac{6}{(1-\nu)^2}\right\}, \tag{4.26}$$

where β *and* ν *are given in* (4.3) *and* (4.16), *respectively, then we further have*

$$\mathbb{E}[F(\tilde{x}_i^{(t)})] - F(x^*) \leq \frac{1}{t}\left(\frac{n\|x^*\|^2}{2a} + \frac{6\sigma^2}{L\big(1-\nu^2\big)}\right). \tag{4.27}$$

Note that when $h \not\equiv 0$, we can only ensure the $\mathcal{O}(1/t)$ rate for the objective value at the auxiliary sequence $\{\tilde{y}^{(t)}\}_{t\geq 1}$ and the distance of each agent's local estimate $\tilde{x}_i^{(t)}$ to $\tilde{y}^{(t)}$; see (4.24) and (4.25) respectively. It remains open whether the $\mathcal{O}(1/t)$ rate for the objective value at $\{\tilde{x}_i^{(t)}\}_{t\geq 1}$, as in (4.27), can be established when $h \not\equiv 0$.

4.5 Proofs of Convergence Results

In this section, we provide the proofs of Theorem 4.2, Corollary 4.1, and Corollary 4.2. Throughout this section, we assume that Assumptions 4.1 and 4.2 are satisfied. Before proceeding, we introduce the following notation:

$$\mathbf{x}^{(t)} = \begin{bmatrix} x_1^{(t)} \\ \vdots \\ x_n^{(t)} \end{bmatrix}, \mathbf{y}^{(t)} = \begin{bmatrix} y^{(t)} \\ \vdots \\ y^{(t)} \end{bmatrix},$$

$$\Delta\mathbf{x}^{(t-1)} = \mathbf{x}^{(t)} - \mathbf{x}^{(t-1)},\ \Delta\mathbf{y}^{(t-1)} = \mathbf{y}^{(t)} - \mathbf{y}^{(t-1)},\ \overline{x}^{(t)} = \frac{1}{n}\sum_{i=1}^{n} x_i^{(t)},\ \overline{g}^{(t)} = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(x_i^{(t)}).$$

To start, we show the following result that quantifies the deviation between local estimates $\{x_i^{(t)}\}_{t\geq 0}$ and the auxiliary sequence $\{y^{(t)}\}_{t\geq 0}$.

Lemma 4.3. *Suppose that* a *satisfies* (4.17). *Then, for all* $t \geq 0$, *it holds that*

$$\sum_{\tau=0}^{t} a_{\tau+1}\mathbb{E}[\|\mathbf{x}^{(\tau)} - \mathbf{y}^{(\tau)}\|^2] \leq \frac{2}{\eta}\sum_{\tau=0}^{t-1} a_{\tau+1}\mathbb{E}[\|\Delta\mathbf{y}^{(\tau)}\|^2] + \frac{2a\sigma^2}{\theta(L+\mu)^2}, \tag{4.28}$$

where σ *is defined in Theorem 4.2,* η *and* θ *are given in* (4.16), *and both* η *and* θ *are positive due to* (4.17) *and Lemma 4.2.*

Lemma 4.3 states that if a satisfies (4.17), then the accumulative deviation between $\mathbf{y}^{(t)}$ and $\mathbf{x}^{(t)}$ admits an upper bound constituted by the successive change of $\mathbf{y}^{(t)}$ plus a constant.

Next, we present the following lemma that pertains to a descent-like property of Algorithm 3.

Lemma 4.4. *For all $t \geq 1$, it holds that*

$$\begin{aligned}
&\sum_{\tau=1}^{t} a_\tau \left(\langle \overline{g}^{(\tau-1)}, y^{(\tau)} - x^* \rangle + h(y^{(\tau)}) - h(x^*) \right) \\
&\quad \leq \frac{\mu}{2} \sum_{\tau=1}^{t} a_\tau \left(\|\overline{x}^{(\tau-1)} - x^*\|^2 - \|\overline{x}^{(\tau-1)} - y^{(\tau)}\|^2 \right) \\
&\qquad - \frac{1}{2} \sum_{\tau=1}^{t} (1 + \mu A_{\tau-1}) \|y^{(\tau)} - y^{(\tau-1)}\|^2 + d(x^*). \qquad (4.29)
\end{aligned}$$

Equipped with the above two technical lemmas, we are ready to present the proof of Theorem 4.2.

Proof of Theorem 4.2. For all $\tau \geq 0$, one has

$$\begin{aligned}
&\frac{1}{n} \sum_{i=1}^{n} a_\tau \left(f_i(y^{(\tau)}) - f_i(x^*) \right) \\
&\leq \frac{1}{n} \sum_{i=1}^{n} a_\tau \Big(f_i(x_i^{(\tau-1)}) - f_i(x^*) + \frac{L}{2} \|y^{(\tau)} - x_i^{(\tau-1)}\|^2 + \langle \nabla f_i(x_i^{(\tau-1)}), y^{(\tau)} - x_i^{(\tau-1)} \rangle \Big) \\
&\leq \frac{1}{n} \sum_{i=1}^{n} a_\tau \Big(\frac{L}{2} \|y^{(\tau)} - x_i^{(\tau-1)}\|^2 - \frac{\mu}{2} \|x_i^{(\tau-1)} - x^*\|^2 + \langle \nabla f_i(x_i^{(\tau-1)}), y^{(\tau)} - x^* \rangle \Big) \\
&= \frac{1}{n} \sum_{i=1}^{n} a_\tau \Big(\frac{L}{2} \|y^{(\tau)} - x_i^{(\tau-1)}\|^2 - \frac{\mu}{2} \|x_i^{(\tau-1)} - x^*\|^2 \Big) + a_\tau \Big\langle \overline{g}^{(\tau-1)}, y^{(\tau)} - x^* \Big\rangle, \qquad (4.30)
\end{aligned}$$

where the two inequalities follow from (2.3) and (2.1), respectively, and the equality uses the definition of $\overline{g}^{(\tau-1)}$. Upon summing up (4.30) from $\tau = 1$ to $\tau = t$ and using

Lemma 4.4 and $F = \frac{1}{n}\sum_{i=1}^{n} f_i + h$, we obtain

$$\begin{aligned}
&\sum_{\tau=1}^{t} a_\tau \left(F(y^{(\tau)}) - F(x^*)\right) \\
&\leq \frac{1}{n}\sum_{\tau=1}^{t}\sum_{i=1}^{n} a_\tau \Big(\frac{L}{2}\|y^{(\tau)} - x_i^{(\tau-1)}\|^2 - \frac{\mu}{2}\|x_i^{(\tau-1)} - x^*\|^2\Big) \\
&\quad + \frac{\mu}{2}\sum_{\tau=1}^{t} a_\tau \left(\|\overline{x}^{(\tau-1)} - x^*\|^2 - \|\overline{x}^{(\tau-1)} - y^{(\tau)}\|^2\right) \\
&\quad - \frac{1}{2}\sum_{\tau=1}^{t}(1 + \mu A_{\tau-1})\|y^{(\tau)} - y^{(\tau-1)}\|^2 + d(x^*).
\end{aligned}$$

Using the definition of $\Delta\mathbf{y}^{(\tau-1)}$ and the fact

$$\|\overline{x}^{(\tau-1)} - x^*\|^2 = \left\|\frac{1}{n}\sum_{i=1}^{n} x_i^{(\tau-1)} - x^*\right\|^2 \leq \frac{1}{n}\sum_{i=1}^{n}\|x_i^{(\tau-1)} - x^*\|^2,$$

the above inequality can be simplified to

$$\begin{aligned}
&\sum_{\tau=1}^{t} a_\tau \left(F(y^{(\tau)}) - F(x^*)\right) \\
&\leq \frac{1}{n}\sum_{\tau=1}^{t}\sum_{i=1}^{n} a_\tau \left(\frac{L}{2}\|y^{(\tau)} - x_i^{(\tau-1)}\|^2 - \frac{\mu}{2}\|y^{\tau} - \overline{x}^{(\tau-1)}\|^2\right) \\
&\quad - \frac{1}{n}\sum_{\tau=1}^{t}\frac{1 + \mu A_{\tau-1}}{2}\|\Delta\mathbf{y}^{(\tau-1)}\|^2 + d(x^*).
\end{aligned} \tag{4.31}$$

By the definition of $\overline{x}^{(\tau-1)}$, $\mathbf{x}^{(\tau)}$, and $\mathbf{y}^{(\tau)}$, one can verify

$$\begin{aligned}
&\sum_{i=1}^{n}\|y^{(\tau)}-\overline{x}^{(\tau-1)}\|^2\\
&=\sum_{i=1}^{n}\left(\|\overline{x}^{(\tau-1)}\|^2-\|x_i^{(\tau-1)}\|^2+\|y^{(\tau)}-x_i^{(\tau-1)}\|^2\right)\\
&=\sum_{i=1}^{n}\left(\|\overline{x}^{(\tau-1)}-y^{(\tau-1)}\|^2-\|x_i^{(\tau-1)}-y^{(\tau-1)}\|^2\right)+\sum_{i=1}^{n}\|y^{(\tau)}-x_i^{(\tau-1)}\|^2\\
&\geq\sum_{i=1}^{n}\left(\|y^{(\tau)}-x_i^{(\tau-1)}\|^2-\|x_i^{(\tau-1)}-y^{(\tau-1)}\|^2\right)\\
&=\|\mathbf{y}^{(\tau)}-\mathbf{x}^{(\tau-1)}\|^2-\|\mathbf{y}^{(\tau-1)}-\mathbf{x}^{(\tau-1)}\|^2.
\end{aligned}$$

Besides, recall that F is convex , $\tilde{y}^{(t)}=A_t^{-1}\sum_{\tau=1}^{t}a_\tau y^{(\tau)}$, and $A_t=\sum_{\tau=1}^{t}a_\tau$. These, together with (4.31), yields

$$\begin{aligned}
&A_t\left(F(\tilde{y}^{(t)})-F(x^*)\right)\\
&\leq\frac{1}{n}\sum_{\tau=1}^{t}a_\tau\left(\frac{L-\mu}{2}\|\mathbf{y}^{(\tau)}-\mathbf{x}^{(\tau-1)}\|^2+\frac{\mu}{2}\|\mathbf{x}^{(\tau-1)}-\mathbf{y}^{(\tau-1)}\|^2\right)+d(x^*)\\
&\quad-\frac{1}{n}\sum_{\tau=1}^{t}\frac{1+\mu A_{\tau-1}}{2}\|\Delta\mathbf{y}^{(\tau-1)}\|^2.
\end{aligned}$$

Upon using the inequality

$$\|\mathbf{y}^{(\tau)}-\mathbf{x}^{(\tau-1)}\|^2\leq 2\|\Delta\mathbf{y}^{(\tau-1)}\|^2+2\|\mathbf{y}^{(\tau-1)}-\mathbf{x}^{(\tau-1)}\|^2,$$

we further obtain

$$\begin{aligned}
&A_t\left(F(\tilde{y}^{(t)})-F(x^*)\right)\\
&\leq\frac{1}{n}\sum_{\tau=1}^{t}a_\tau\left(L-\mu-\frac{1+\mu A_{\tau-1}}{2a_\tau}\right)\|\Delta\mathbf{y}^{(\tau-1)}\|^2\\
&\quad+\frac{2L-\mu}{2n}\sum_{\tau=1}^{t}a_\tau\|\mathbf{x}^{(\tau-1)}-\mathbf{y}^{(\tau-1)}\|^2+d(x^*)\\
&=\frac{2L-\mu-\frac{1}{a}}{2n}\sum_{\tau=1}^{t}a_\tau\|\Delta\mathbf{y}^{(\tau-1)}\|^2+\frac{2L-\mu}{2n}\sum_{\tau=1}^{t}a_\tau\|\mathbf{x}^{(\tau-1)}-\mathbf{y}^{(\tau-1)}\|^2+d(x^*),
\end{aligned}$$

where the equality follows from the identity

$$\frac{1+\mu A_{\tau-1}}{a_\tau} = \frac{1-a\mu}{a},$$

which holds due to the update rule of $\{a_t\}_{t\geq 0}$ and $\{A_t\}_{t\geq 0}$. Upon taking expectation on both sides of the above inequality and using Lemma 4.3, one has

$$\begin{aligned} &A_t\left(\mathbb{E}[F(\tilde{y}^{(t)})] - F(x^*)\right) + \frac{\gamma}{2n}\sum_{\tau=1}^{t} a_\tau \mathbb{E}[\|\Delta \mathbf{y}^{(\tau-1)}\|^2] \\ &\leq d(x^*) + \frac{(2L-\mu)a\sigma^2}{n\theta(L+\mu)^2} = C, \end{aligned} \tag{4.32}$$

where $\gamma > 0$ is defined in (4.19). This implies (6.25) as desired. Moreover, it follows from (4.32) and $A_t\left(\mathbb{E}[F(\tilde{y}^{(t)})] - F(x^*)\right) \geq 0$ that

$$\sum_{\tau=1}^{t} a_\tau \mathbb{E}[\|\Delta \mathbf{y}^{(\tau-1)}\|^2] \leq \frac{2nC}{\gamma}.$$

This, together with the convexity of $\|\cdot\|^2$, Jensen's Inequality, $a_t \leq a_{t+1}$ for all $t \geq 0$, and Lemma 4.3, yields

$$\begin{aligned} &A_t \mathbb{E}[\|\tilde{\mathbf{x}}^{(t)} - \tilde{\mathbf{y}}^{(t)}\|^2] \leq \sum_{\tau=1}^{t} a_\tau \mathbb{E}[\|\mathbf{x}^{(\tau)} - \mathbf{y}^{(\tau)}\|^2] \leq \sum_{\tau=0}^{t} a_{\tau+1} \mathbb{E}[\|\mathbf{x}^{(\tau)} - \mathbf{y}^{(\tau)}\|^2] \\ &\leq \frac{2}{\eta}\sum_{\tau=0}^{t-1} a_{\tau+1}\mathbb{E}[\|\Delta \mathbf{y}^{(\tau)}\|^2] + \frac{2a\sigma^2}{\theta(L+\mu)^2} \\ &\leq \frac{4nC}{\eta\gamma} + \frac{2a\sigma^2}{\theta(L+\mu)^2} = D, \end{aligned}$$

which implies (4.21) as desired. □

Next, we provide the proof of Corollary 4.1.

Proof of Corollary 4.1. Since $\mu > 0$, we obtain from the update of A_t in Algorithm 3 that

$$\frac{1}{A_t} = \frac{\mu}{\left(\frac{1}{1-a\mu}\right)^t - 1} \leq \frac{(1-a\mu)^t}{a}.$$

Besides, upon using the fact that F is strongly convex with modulus μ, one has that

for all $t \geq 0$ and $i = 1, \ldots, n$,

$$\begin{aligned}\|x_i^{(t)} - x^*\|^2 &\leq 2\|x_i^{(t)} - y^{(t)}\|^2 + 2\|y^{(t)} - x^*\|^2 \\ &\leq 2\|x_i^{(t)} - y^{(t)}\|^2 + \frac{1}{\mu}\left(F(y^{(t)}) - F(x^*)\right).\end{aligned}$$

These, together with (6.25) and (4.21), yields (4.23). □

Proof of Corollary 4.2. The upper bounds in (4.24) and (4.25) directly follow from the results in Theorem 4.2 and

$$\frac{1}{A_t} = \frac{1}{at}.$$

For the special case $h(x) = 0$ and $d(x) = \frac{1}{2}\|x\|^2$, we consider

$$\begin{aligned}&F(x_i^{(\tau)}) - F(y^{(\tau)}) = f(x_i^{(\tau)}) - f(y^{(\tau)}) \\ &\leq \frac{1}{n}\sum_{j=1}^{n}\left(f_j(x_i^{(\tau)}) - \langle \nabla f_j(x_j^{(\tau)}), y^{(\tau)} - x_j^{(\tau)}\rangle - f_j(x_j^{(\tau)})\right) \\ &\leq \frac{1}{n}\sum_{j=1}^{n}\left(\langle \nabla f_j(x_j^{(\tau)}), x_i^{(\tau)} - x_j^{(\tau)}\rangle - \langle \nabla f_j(x_j^{(\tau)}), y^{(\tau)} - x_j^{(\tau)}\rangle + \frac{L}{2}\|x_i^{(\tau)} - y^{(\tau)} + y^{(\tau)} - x_j^{(\tau)}\|^2\right) \\ &= \frac{1}{n}\sum_{j=1}^{n}\left(\langle \nabla f_j(x_j^{(\tau)}), x_i^{(\tau)} - y^{(\tau)}\rangle + L\|x_i^{(\tau)} - y^{(\tau)}\|^2 + L\|y^{(\tau)} - x_j^{(\tau)}\|^2\right) \\ &= \left\langle \overline{g}^{(\tau)}, x_i^{(\tau)} - y^{(\tau)}\right\rangle + L\|x_i^{(\tau)} - y^{(\tau)}\|^2 + \frac{L}{n}\|\mathbf{y}^{(\tau)} - \mathbf{x}^{(\tau)}\|^2,\end{aligned} \tag{4.33}$$

where the two inequalities follow from (2.1) and (2.3), respectively. The closed-form solutions for (4.14) and (4.18) can be derived as

$$x_i^{(\tau)} = -\frac{z_i^{(\tau)}}{1 + \mu A_\tau}, \quad y^{(\tau)} = -\frac{\overline{z}^{(\tau)}}{1 + \mu A_\tau}.$$

Therefore $y^{(\tau)} = \overline{x}^{(\tau)}$. We sum up (4.33) from $i = 1$ to $i = n$ to get

$$\sum_{i=1}^{n}\left(F(x_i^{(\tau)}) - F(y^{(\tau)})\right) \leq 2L\|\mathbf{y}^{(\tau)} - \mathbf{x}^{(\tau)}\|^2. \tag{4.34}$$

Upon summing up (4.34) from $\tau = 1$ to $\tau = t$ and using the convexity of F, we obtain

$$t\sum_{i=1}^{n}\Big(F(\tilde{x}_i^{(t)}) - F(y^{(\tau)})\Big) \le \sum_{\tau=1}^{t}\sum_{i=1}^{n}\Big(F(x_i^{(\tau)}) - F(y^{(\tau)})\Big) \le 2L\sum_{\tau=1}^{t}\|\mathbf{y}^{(\tau)} - \mathbf{x}^{(\tau)}\|^2, \tag{4.35}$$

where $\tilde{x}_i^{(t)} = \frac{1}{t}\sum_{\tau=1}^{t} x_i^{(\tau)}$. After taking expectation on both sides of the above inequality and using Lemma 4.3 with $a_\tau = a$, we get

$$\begin{aligned} &t\sum_{i=1}^{n}\mathbb{E}\Big[F(\tilde{x}_i^{(t)}) - F(y^{(t)})\Big] \\ &\le \frac{4L}{\eta}\sum_{\tau=1}^{t}\mathbb{E}[\|\Delta\mathbf{y}^{(\tau-1)}\|^2] + \frac{4L\sigma^2}{\theta(L+\mu)^2} = \frac{4L}{(1-\nu)^2}\sum_{\tau=1}^{t}\mathbb{E}[\|\Delta\mathbf{y}^{(\tau-1)}\|^2] + \frac{4\sigma^2}{L(1-\nu^2)}. \end{aligned} \tag{4.36}$$

By setting $\mu = 0$ and $d(x^*) = \frac{1}{2}\|x^*\|^2$ in (4.32), we have

$$\begin{aligned} &at\left(\mathbb{E}[F(\tilde{y}^{(t)})] - F(x^*)\right) \\ &\le -\left(\frac{1}{2a} - L - \frac{2L}{(1-\nu^2)}\right)\frac{a}{n}\sum_{\tau=1}^{t}\mathbb{E}[\|\Delta\mathbf{y}^{(\tau-1)}\|^2] + \frac{\|x^*\|^2}{2} + \frac{2a\sigma^2}{nL(1-\nu^2)}. \end{aligned} \tag{4.37}$$

Also, by multiplying $n/a > 0$ on both sides of the above inequality and adding the resultant inequality to (4.36), we obtain

$$\begin{aligned} &t\Big(\mathbb{E}[F(\tilde{x}_i^{(t)})] - F(x^*)\Big) \le t\sum_{i=1}^{n}\Big(\mathbb{E}[F(\tilde{x}_i^{(t)})] - F(x^*)\Big) \\ &\le -\Big(\frac{1}{2a} - L - \frac{6L}{(1-\nu)^2}\Big)\sum_{\tau=1}^{t}\mathbb{E}[\|\Delta\mathbf{y}^{(\tau-1)}\|^2] + \frac{n}{2a}\|x^*\|^2 + \frac{6\sigma^2}{L\big(1-\nu^2\big)}. \end{aligned} \tag{4.38}$$

Now, using the condition in (4.26), we arrive at (4.27). □

4.6 Proofs of Supporting Lemmas for Theorem 4.2

4.6.1 Proof of Lemma 4.2

Proof of Lemma 4.2. We first show that ν monotonically increases with a if $a \in (0, 1/\mu)$. Recall that $\mathbf{M}$ is defined in (4.15). Then, the characteristic polynomial of $\mathbf{M}$, denoted by $p(\lambda)$, is a quadratic function:

$$\begin{aligned} p(\lambda) :&= \det(\lambda I - \mathbf{M}) = (\lambda - M_{11})(\lambda - M_{22}) - M_{12}M_{21} \\ &= \lambda^2 - \frac{\beta(2+aL)}{1-a\mu}\lambda + \frac{\beta^2}{1-a\mu} - \frac{a\beta(L+\mu)}{(1-a\mu)^2}. \end{aligned} \tag{4.39}$$

Using this, we obtain that $\mathbf{M}$ has two real eigenvalues $\lambda_1 = (\xi_1 + \xi_2)/2$ and $\lambda_2 = (\xi_1 - \xi_2)/2$, where

$$\xi_1 = \frac{\beta(2+aL)}{1-a\mu}, \qquad \xi_2 = \frac{\sqrt{a^2\beta^2L^2 + 4a\beta(\beta+1)(L+\mu)}}{1-a\mu}. \tag{4.40}$$

Notice that $\xi_1 > 0$ and $\xi_2 > 0$ for any $a \in (0, 1/\mu)$. Thus, we have $\lambda_1 > 0$ and $|\lambda_1| > |\lambda_2|$ for any $a \in (0, 1/\mu)$. It then follows that $\rho(\mathbf{M}) = \lambda_1$ and

$$\nu(a) = \rho(\mathbf{M})\sqrt{1-a\mu} = \lambda_1\sqrt{1-a\mu} = \frac{\beta(2+aL)}{2\sqrt{1-a\mu}} + \frac{\sqrt{a^2\beta^2L^2 + 4a\beta(\beta+1)(L+\mu)}}{2\sqrt{1-a\mu}}. \tag{4.41}$$

By routine calculation, one can verify that $\nu'(a) > 0$ if $a \in (0, \mu^{-1})$. Therefore, the value of ν monotonically increases with a if $a \in (0, \mu^{-1})$.

Next, we show that $\nu < 1$ if (4.17) is satisfied. Note that (4.17) implies that $0 < a < \mu^{-1}$ and

$$\frac{\beta(2L+3\mu)}{(1-\beta)^2} < \frac{1}{a} - \mu = \frac{1-a\mu}{a}.$$

It then follows that $1 - a\mu \in (0, 1]$ and hence

$$0 < (1-\beta)^2 - \frac{a\beta(2L+3\mu)}{1-a\mu} = 1 + \beta^2 - \left(\beta + \frac{\beta + a\beta(L+\mu)}{1-a\mu}\right) - \frac{a\beta(L+\mu)}{1-a\mu}.$$

Upon dividing both sides of the above inequality by $1-a\mu$, we obtain

$$\begin{aligned} 0 &< \frac{1}{1-a\mu} + \frac{\beta^2}{1-a\mu} - \frac{1}{1-a\mu}\big(\beta + \frac{\beta + a\beta(L+\mu)}{1-a\mu}\big) - \frac{a\beta(L+\mu)}{(1-a\mu)^2} \\ &\leq \frac{1}{1-a\mu} + \frac{\beta^2}{1-a\mu} - \frac{1}{\sqrt{1-a\mu}}\big(\beta + \frac{\beta + a\beta(L+\mu)}{1-a\mu}\big) - \frac{a\beta(L+\mu)}{(1-a\mu)^2} \\ &= p\left(\frac{1}{\sqrt{1-a\mu}}\right), \end{aligned} \tag{4.42}$$

where the second inequality is due to $1-a\mu \in (0,1]$ and the equality follows from (4.39). Besides, using the definition of characteristic polynomial, one further has

$$0 < p\left(\frac{1}{\sqrt{1-a\mu}}\right) = \left(\frac{1}{\sqrt{1-a\mu}} - M_{11}\right)\left(\frac{1}{\sqrt{1-a\mu}} - M_{22}\right) - M_{12}M_{21}.$$

By (4.15), $\beta \in (0,1)$, and $1-a\mu \in (0,1]$, we have that $M_{12} > 0$, $M_{21} > 0$, and $1/\sqrt{1-a\mu} > M_{11}$. It then follows that $1/\sqrt{1-a\mu} > M_{22}$ and hence

$$p'\left(\frac{1}{\sqrt{1-a\mu}}\right) = \frac{2}{\sqrt{1-a\mu}} - M_{11} - M_{22} > 0.$$

This, together with the fact that q is a quadratic function, implies that $q(\lambda)$ is monotonically increasing on $[1/\sqrt{1-a\mu}, \infty)$. It then follows from (4.42) that

$$\frac{1}{\sqrt{1-a\mu}} > \lambda_1 = \rho(\mathbf{M}),$$

which implies that $\nu < 1$. □

4.6.2 Proof of Lemma 4.3

In this subsection, we first present three technical lemmas, and then provide the proof of Lemma 4.3. Before proceeding, we introduce the following notation:

$$\mathbf{x}^{(t)} = \begin{bmatrix} x_1^{(t)} \\ \vdots \\ x_n^{(t)} \end{bmatrix}, \quad \mathbf{s}^{(t)} = \begin{bmatrix} s_1^{(t)} \\ \vdots \\ s_n^{(t)} \end{bmatrix}, \quad \mathbf{z}^{(t)} = \begin{bmatrix} z_1^{(t)} \\ \vdots \\ z_n^{(t)} \end{bmatrix}, \quad \nabla^{(t)} = \begin{bmatrix} \nabla f_1(x_1^{(t)}) \\ \vdots \\ \nabla f_n(x_n^{(t)}) \end{bmatrix}, \quad \mathbf{y}^{(t)} = \begin{bmatrix} y^{(t)} \\ \vdots \\ y^{(t)} \end{bmatrix}, \tag{4.43}$$

$$\overline{x}^{(t)} = \frac{1}{n}\sum_{i=1}^{n} x_i^{(t)}, \quad \overline{g}^{(t)} = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(x_i^{(t)}), \quad \overline{s}^{(t)} = \frac{1}{n}\sum_{i=1}^{n} s_i^{(t)}, \quad \overline{z}^{(t)} = \frac{1}{n}\sum_{i=1}^{n} z_i^{(t)}, \tag{4.44}$$

$$\tilde{\mathbf{s}}^{(t)} = \mathbf{s}^{(t)} - \mathbf{1} \otimes \overline{s}^{(t)}, \quad \tilde{\mathbf{z}}^{(t)} = \mathbf{z}^{(t)} - \mathbf{1} \otimes \overline{z}^{(t)}, \quad \Delta\overline{x}^{(t-1)} = \overline{x}^{(t)} - \overline{x}^{(t-1)}, \tag{4.45}$$

$$\Delta\mathbf{x}^{(t-1)} = \mathbf{x}^{(t)} - \mathbf{x}^{(t-1)}, \quad \Delta\mathbf{y}^{(t-1)} = \mathbf{y}^{(t)} - \mathbf{y}^{(t-1)}, \quad \Delta y^{(t-1)} = y^{(t)} - y^{(t-1)}, \tag{4.46}$$

where $\mathbf{1}$ is an all-one column vector of dimension n. We remark that bold lowercase letters represent a vector of dimension $m \times n$, while normal lowercase letters represent a vector of dimension m. Equipped with these notation, we can re-write the update rule (4.13) in the following compact form:

$$\mathbf{z}^{(t)} = \mathbf{P}^{(t-1)}\Big(\mathbf{z}^{(t-1)} + a_t \mathbf{s}^{(t-1)}\Big), \tag{4.47a}$$

$$\mathbf{s}^{(t)} = \mathbf{P}^{(t-1)}\mathbf{s}^{(t-1)} + \nabla^{(t)} - \nabla^{(t-1)} - \mu\Delta\mathbf{x}^{(t-1)}, \tag{4.47b}$$

where $\mathbf{P}^{(t)} = P^{(t)} \otimes I$ with I being an identity matrix of size $n \times n$.

For a real-valued random vector x, we define

$$\|x\|_{\mathbb{E}} = \sqrt{\mathbb{E}[\|x\|^2]}. \tag{4.48}$$

Accordingly, for a square random matrix W, we define $\|W\|_{\mathbb{E}} = \sup_{\|x\|_{\mathbb{E}}=1} \|Wx\|_{\mathbb{E}}$. Given two real-valued random vectors x, y, the Minkowski inequality [22] states that

$$\|x + y\|_{\mathbb{E}} \leq \|x\|_{\mathbb{E}} + \|y\|_{\mathbb{E}}. \tag{4.49}$$

Now, we are ready to present three technical lemmas.

Lemma 4.5. *For the sequences $\{\overline{s}^{(t)}\}_{t\geq 0}$ and $\{\overline{z}^{(t)}\}_{t\geq 0}$ defined in (4.44), one has that for any $t \geq 0$,*

$$\overline{s}^{(t)} = \overline{g}^{(t)} - \mu\overline{x}^{(t)}, \qquad \overline{z}^{(t)} = \sum_{\tau=0}^{t-1} a_{\tau+1}\overline{s}^{(\tau)}. \tag{4.50}$$

Proof. We prove by an induction argument. Since $s_i^{(0)} = \nabla f_i(x^{(0)}) - \mu x^{(0)}$, $z_i^{(0)} = 0$ and $x_i^{(0)} = x^{(0)}$ for all i, we readily have that (4.50) holds when $t = 0$. Now, suppose that (4.50) holds for $t - 1$. From (4.43) and (4.44), we observe that the following

identities hold for any $\tau \geq 0$:

$$\overline{x}^{(\tau)} = \frac{1}{n}(\mathbf{1}^T \otimes I)\mathbf{x}^{(\tau)}, \quad \overline{g}^{(\tau)} = \frac{1}{n}(\mathbf{1}^T \otimes I)\nabla^{(\tau)}, \quad \overline{s}^{(\tau)} = \frac{1}{n}(\mathbf{1}^T \otimes I)\mathbf{s}^{(\tau)}, \quad \overline{z}^{(\tau)} = \frac{1}{n}(\mathbf{1}^T \otimes I)\mathbf{z}^{(\tau)}. \tag{4.51}$$

It then follows this and (4.47b) that

$$\begin{aligned}
&\overline{s}^{(t)} = \frac{1}{n}(\mathbf{1}^T \otimes I)\mathbf{s}^{(t)} \\
=&\frac{1}{n}(\mathbf{1}^T \otimes I)(P^{(t-1)} \otimes I)\mathbf{s}^{(t-1)} + \frac{1}{n}(\mathbf{1}^T \otimes I)\nabla^{(t)} - \frac{1}{n}(\mathbf{1}^T \otimes I)\nabla^{(t-1)} - \frac{\mu}{n}(\mathbf{1}^T \otimes I)\Delta\mathbf{x}^{(t-1)} \\
=&\frac{1}{n}(\mathbf{1}^T P^{(t-1)} \otimes I)\mathbf{s}^{(t-1)} + \overline{g}^{(t)} - \overline{g}^{(t-1)} - \mu\overline{x}^{(t)} + \mu\overline{x}^{(t-1)} \\
=&\frac{1}{n}(\mathbf{1}^T \otimes I)\mathbf{s}^{(t-1)} + \overline{g}^{(t)} - \overline{g}^{(t-1)} - \mu\overline{x}^{(t)} + \mu\overline{x}^{(t-1)} \\
=&\overline{s}^{(t-1)} + \overline{g}^{(t)} - \overline{g}^{(t-1)} - \mu\overline{x}^{(t)} + \mu\overline{x}^{(t-1)} \\
=&\overline{g}^{(t)} - \mu\overline{x}^{(t)},
\end{aligned}$$

where the second equality is due to (4.47b), the third equality uses the fact that $(A \otimes B)(C \otimes D) = (AC \otimes BD)$, the fourth equality follows from the fact that $P^{(t-1)}$ is doubly stochastic, and the last equality is due to the assumption that (4.50) holds for $t-1$. Similarly, by (4.47a) and (4.51), we obtain

$$\begin{aligned}
&\overline{z}^{(t)} = \frac{1}{n}(\mathbf{1}^T \otimes I)\mathbf{z}^{(t)} \\
=&\frac{1}{n}(\mathbf{1}^T \otimes I)(P^{(t-1)} \otimes I)\left(\mathbf{z}^{(t-1)} + a_t\mathbf{s}^{(t-1)}\right) = \frac{1}{n}(\mathbf{1}^T \otimes I)\left(\mathbf{z}^{(t-1)} + a_t\mathbf{s}^{(t-1)}\right) \\
=&\overline{z}^{(t-1)} + a_t\overline{s}^{(t-1)} = \sum_{\tau=0}^{t-2} a_{\tau+1}\overline{s}^{(\tau)} + a_t\overline{s}^{(t-1)} = \sum_{\tau=0}^{t-1} a_{\tau+1}\overline{s}^{(\tau)}.
\end{aligned}$$

Therefore, (4.50) holds for t and the induction argument is completed. □

Lemma 4.6. *For the sequence $\{x_i^{(t)} : i = 1, \ldots, n\}_{t\geq 0}$ generated by Algorithm 3 and the auxiliary sequence $\{y^{(t)}\}_{t\geq 0}$ defined in (4.18), one has that for all $t \geq 0$ and $i = 1, \ldots, n$,*

$$\|x_i^{(t)} - y^{(t)}\| \leq \frac{1}{1 + \mu A_t}\left\|z_i^{(t)} - \overline{z}^{(t)}\right\|, \tag{4.52}$$

where $\overline{z}^{(t)}$ is defined in (4.44).

Proof of Lemma 4.6. It is easy to see that (4.52) holds when $t = 0$ because both sides

of (4.52) equal 0. Now, suppose that $t \geq 1$. Recall that d is strongly convex with modulus 1. Let the mapping $R : \mathbb{R}^m \to \mathbb{R}^m$ be defined as

$$R(\omega) := \operatorname*{argmin}_{x \in \mathbb{R}^m} \{\langle \omega, x \rangle + \phi(x)\},$$

where $\phi(x) = A_t(\mu \|x\|^2/2 + h(x)) + d(x)$ is strongly convex with modulus $1 + \mu A_t$. Then, by (4.14) and (4.18), we have

$$y^{(t)} = R(\overline{z}^{(t)}), \qquad x_i^{(t)} = R(z_i^{(t)}), \quad \forall i = 1, \ldots, n.$$

Moreover, the mapping R is Lipschitz continuous with Lipschitz constant $(1+\mu A_t)^{-1}$; see, e.g., Proposition 4.9 in [29]. This immediately implies (4.52) as desired. □

Next, we recall a lemma from [111]. For completeness, we provide a proof here.

Lemma 4.7. *Suppose that $\{q^{(t)}\}_{t\geq 0}$ and $\{p^{(t)}\}_{t\geq 0}$ are two sequences of positive scalars such that for all $t \geq 0$,*

$$q^{(t)} \leq \nu^t q^{(0)} + \sum_{\tau=0}^{t-1} \nu^{t-\tau-1} p^{(\tau)}$$

where $\nu \in (0, 1)$. Then, the following holds for all $t \geq 0$:

$$\sum_{\tau=1}^{t} (q^{(\tau)})^2 \leq \frac{2}{(1-\nu)^2} \sum_{\tau=0}^{t-1} (p^{(\tau)})^2 + \frac{2}{1-\nu^2} (q^{(0)})^2.$$

Proof of Lemma 4.7. For any $\tau \geq 0$, we have

$$\begin{aligned}
(q^{(\tau)})^2 &\leq 2\nu^{2\tau}(q^{(0)})^2 + 2\left(\sum_{l=0}^{\tau-1} \nu^{\tau-l-1} p^{(l)}\right)^2 \\
&\leq 2\nu^{2\tau}(q^{(0)})^2 + 2\left(\sum_{l=0}^{\tau-1} (\nu^{\frac{\tau-l-1}{2}})^2\right)\left(\sum_{l=0}^{\tau-1} (\nu^{\frac{\tau-l-1}{2}} p^{(l)})^2\right) \\
&\leq 2\nu^{2\tau}(q^{(0)})^2 + \frac{2}{1-\nu} \sum_{l=0}^{\tau-1} \nu^{\tau-l-1} (p^{(l)})^2,
\end{aligned}$$

where the first inequality is due to $(a+b)^2 \leq 2a^2 + 2b^2$ for any $a, b \in \mathbb{R}$ and the second one uses Cauchy-Schwartz inequality. Upon summing up the above inequality from

$\tau = 1$ to $\tau = t$, we obtain

$$\begin{aligned}\sum_{\tau=1}^{t}(q^{(\tau)})^2 &\le \sum_{\tau=1}^{t}\left(2\nu^{2k}(q^{(0)})^2 + \frac{2}{1-\nu}\sum_{l=0}^{\tau-1}\nu^{\tau-l-1}(p^{(l)})^2\right)\\ &\le \frac{2}{1-\nu^2}(q^{(0)})^2 + \frac{2}{1-\nu}\sum_{\tau=1}^{t}\sum_{l=0}^{\tau-1}\nu^{\tau-l-1}(p^{(l)})^2\\ &= \frac{2}{1-\nu^2}(q^{(0)})^2 + \frac{2}{1-\nu}\sum_{l=0}^{t-1}\sum_{\tau=0}^{t-l-1}\nu^{\tau}(p^{(l)})^2\\ &\le \frac{2}{1-\nu^2}(q^{(0)})^2 + \frac{2}{(1-\nu)^2}\sum_{l=0}^{t-1}(p^{(l)})^2,\end{aligned}$$

where the equality follows from exchanging the order of the summation. □

Now we are ready to prove Lemma 4.3.

Proof of Lemma 4.3. From Lemma 4.5, we have

$$\overline{z}^{(\tau)} = \overline{z}^{(\tau-1)} + a_\tau \overline{s}^{(\tau-1)}.$$

This, together with (4.47a) and the definition of $\tilde{\mathbf{z}}^{(\tau)}$ in (4.45), yields

$$\tilde{\mathbf{z}}^{(\tau)} = \mathbf{P}^{(\tau-1)}\mathbf{z}^{(\tau-1)} - \mathbf{1}\otimes\overline{z}^{(\tau-1)} + a_\tau\left(\mathbf{P}^{(\tau-1)}\mathbf{s}^{(\tau-1)} - \mathbf{1}\otimes\overline{s}^{(\tau-1)}\right). \tag{4.53}$$

It then follows from (4.49) that

$$\|\tilde{\mathbf{z}}^{(\tau)}\|_{\mathbb{E}} \le \left\|\mathbf{P}^{(\tau-1)}\mathbf{z}^{(\tau-1)} - \mathbf{1}\otimes\overline{z}^{(\tau-1)}\right\|_{\mathbb{E}} + a_\tau\left\|\mathbf{P}^{(\tau-1)}\mathbf{s}^{(\tau-1)} - \mathbf{1}\otimes\overline{s}^{(\tau-1)}\right\|_{\mathbb{E}}. \tag{4.54}$$

Note that $\mathbf{1}\otimes\bar{z}^{(\tau-1)} = (\mathbf{1}\otimes I)\bar{z}^{(\tau-1)}$, which, together with (4.51) and the identity $(A\otimes B)(C\otimes D) = (AC\otimes BD)$, yields

$$\mathbf{1}\otimes\bar{z}^{(\tau-1)} = \frac{1}{n}(\mathbf{1}\otimes I)(\mathbf{1}^T\otimes I)\mathbf{z}^{(\tau-1)} = \left(\frac{\mathbf{1}\mathbf{1}^T}{n}\otimes I\right)\mathbf{z}^{(\tau-1)}.$$

Using this and $\mathbf{P}^{(\tau-1)} = P^{(\tau-1)} \otimes I$, we obtain

$$\begin{aligned}
&\mathbf{P}^{(\tau-1)}\mathbf{z}^{(\tau-1)} - \mathbf{1} \otimes \overline{z}^{(\tau-1)} \\
&= (P^{(\tau-1)} \otimes I)\mathbf{z}^{(\tau-1)} - \left(\frac{\mathbf{1}\mathbf{1}^T}{n} \otimes I\right) \mathbf{z}^{(\tau-1)} \\
&= \left(\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right) \otimes I\right) \mathbf{z}^{(\tau-1)}. \\
&= \left(\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right) \otimes I\right) \left(\tilde{\mathbf{z}}^{(\tau-1)} + (\mathbf{1} \otimes I)\bar{z}^{(\tau-1)}\right) \\
&= \left(\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right) \otimes I\right) \tilde{\mathbf{z}}^{(\tau-1)} + \left(\left(P^{(\tau-1)}\mathbf{1} - \mathbf{1}\right) \otimes I\right) \bar{z}^{(\tau-1)} \\
&= \left(\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right) \otimes I\right) \tilde{\mathbf{z}}^{(\tau-1)},
\end{aligned} \tag{4.55}$$

where the third equality uses (4.45) and $\mathbf{1} \otimes \bar{z}^{(\tau-1)} = (\mathbf{1} \otimes I)\bar{z}^{(\tau-1)}$, the fourth equality follows from the identity $(A \otimes B)(C \otimes D) = (AC \otimes BD)$, and the last one is due to the fact that $P^{(\tau-1)}$ is doubly stochastic. Then, by (4.48) and Assumption 4.2, one has

$$\begin{aligned}
&\left\|\mathbf{P}^{(\tau-1)}\mathbf{z}^{(\tau-1)} - (\mathbf{1} \otimes I)\overline{z}^{(\tau-1)}\right\|_{\mathbb{E}} = \left\|\left(\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right) \otimes I\right) \tilde{\mathbf{z}}^{(\tau-1)}\right\|_{\mathbb{E}} \\
&\overset{(i)}{=} \sqrt{\mathbb{E}\left[\tilde{\mathbf{z}}^{(\tau-1)^T}\left(\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)^T \otimes I^T\right)\left(\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right) \otimes I\right) \tilde{\mathbf{z}}^{(\tau-1)}\right]} \\
&\overset{(ii)}{=} \sqrt{\mathbb{E}\left[\tilde{\mathbf{z}}^{(\tau-1)T}\mathbb{E}\left[\left(\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)^T\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)\right) \otimes I \,\middle|\, \tilde{\mathbf{z}}^{(\tau-1)}\right] \tilde{\mathbf{z}}^{(\tau-1)}\right]} \\
&\overset{(iii)}{\leq} \|\tilde{\mathbf{z}}^{(\tau-1)}\|_{\mathbb{E}} \sqrt{\rho\left(\mathbb{E}\left[\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)^T\left(P^{(\tau-1)} - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)\right]\right)} \\
&= \|\tilde{\mathbf{z}}^{(\tau-1)}\|_{\mathbb{E}} \sqrt{\rho\left(\mathbb{E}\left[P^{(\tau-1)^T}P^{(\tau-1)}\right] - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)} \\
&\leq \beta\|\tilde{\mathbf{z}}^{(\tau-1)}\|_{\mathbb{E}}
\end{aligned}$$

where $(A \otimes B)^T = A^T \otimes B^T$ and $(A \otimes B)(C \otimes D) = AC \otimes BD$ are used to obtain (i) and (ii), respectively, and in (iii) we use that $P^{(\tau-1)}$ is independent of $\tilde{\mathbf{z}}^{(\tau-1)}$. Using

the same arguments as above, we have

$$\left\|\mathbf{P}^{(\tau-1)}\mathbf{s}^{(\tau-1)} - \mathbf{1}\otimes \overline{s}^{(\tau-1)}\ \right\|_{\mathbb{E}} \leq \beta\|\tilde{\mathbf{s}}^{(\tau-1)}\|_{\mathbb{E}}. \tag{4.56}$$

It then follows from (4.54) that

$$\|\tilde{\mathbf{z}}^{(\tau)}\|_{\mathbb{E}} \leq \beta\left(\|\tilde{\mathbf{z}}^{(\tau-1)}\|_{\mathbb{E}} + a_\tau\|\tilde{\mathbf{s}}^{(\tau-1)}\|_{\mathbb{E}}\right). \tag{4.57}$$

Similarly, from Lemma 4.5, (4.45), and (4.47b), we obtain

$$\begin{aligned}\|\tilde{\mathbf{s}}^{(\tau)}\|_{\mathbb{E}} = \Big\|&\mathbf{P}^{(\tau-1)}\mathbf{s}^{(\tau-1)} - (\mathbf{1}\otimes I)\,\overline{s}^{(\tau-1)} - (\mathbf{1}\otimes I)\left(\overline{g}^{(\tau)} - \overline{g}^{(\tau-1)} - \mu\Delta\overline{x}^{(\tau-1)}\right)\\ &+ \nabla^{(\tau)} - \nabla^{(\tau-1)} - \mu\Delta\mathbf{x}^{(\tau-1)}\Big\|_{\mathbb{E}}\\ \leq \Big\|&\mathbf{P}^{(\tau-1)}\mathbf{s}^{(\tau-1)} - (\mathbf{1}\otimes I)\,\overline{s}^{(\tau-1)}\Big\|_{\mathbb{E}} + \mu\left\|(\mathbf{1}\otimes I)\Delta\overline{x}^{(\tau-1)} - \Delta\mathbf{x}^{(\tau-1)}\right\|_{\mathbb{E}}\\ &+ \left\|\nabla^{(\tau)} - \nabla^{(\tau-1)} - (\mathbf{1}\otimes I)\,(\overline{g}^{(\tau)} - \overline{g}^{(\tau-1)})\right\|_{\mathbb{E}}.\end{aligned} \tag{4.58}$$

By (4.51), one can verify that

$$\begin{aligned}\nabla^{(\tau)} - \nabla^{(\tau-1)} - (\mathbf{1}\otimes I)(\overline{g}^{(\tau)} - \overline{g}^{(\tau-1)}) &= \left(\left(I - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)\otimes I\right)\left(\nabla^{(\tau)} - \nabla^{(\tau-1)}\right),\\ \Delta\mathbf{x}^{(\tau-1)} - (\mathbf{1}\otimes I)\,\Delta\overline{x}^{(\tau-1)} &= \left(\left(I - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)\otimes I\right)\Delta\mathbf{x}^{(\tau-1)},\end{aligned}$$

which respectively imply that

$$\begin{aligned}\|\nabla^{(\tau)} - \nabla^{(\tau-1)} - (\mathbf{1}\otimes I)(\overline{g}^{(\tau)} - \overline{g}^{(\tau-1)})\|_{\mathbb{E}} &\leq \left\|\left(I - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)\otimes I\right\|\|\nabla^{(\tau)} - \nabla^{(\tau-1)}\|_{\mathbb{E}}\\ &\leq \|\nabla^{(\tau)} - \nabla^{(\tau-1)}\|_{\mathbb{E}},\\ \|\Delta\mathbf{x}^{(\tau-1)} - (\mathbf{1}\otimes I)\,\Delta\overline{x}^{(\tau-1)}\|_{\mathbb{E}} &\leq \left\|\left(I - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)\otimes I\right\|\|\Delta\mathbf{x}^{(\tau-1)}\|_{\mathbb{E}}\\ &\leq \|\Delta\mathbf{x}^{(\tau-1)}\|_{\mathbb{E}}.\end{aligned}$$

Besides, it follows from (2.2) that $\|\nabla^{(\tau)} - \nabla^{(\tau-1)}\|_{\mathbb{E}} \leq L\|\Delta\mathbf{x}^{(k-1)}\|_{\mathbb{E}}$. Upon substituting these and (4.56) into (4.58), we obtain

$$\|\tilde{\mathbf{s}}_\tau\|_{\mathbb{E}} \leq \beta\|\tilde{\mathbf{s}}^{(\tau-1)}\|_{\mathbb{E}} + (L+\mu)\|\Delta\mathbf{x}^{(\tau-1)}\|_{\mathbb{E}}. \tag{4.59}$$

Multiplying the both sides of the above inequality by $a_{\tau+1} = a_\tau/(1-a\mu)$, we have

$$a_{\tau+1}\|\tilde{\mathbf{s}}^{(\tau)}\|_{\mathbb{E}} \leq \frac{1}{1-a\mu}\left(\beta a_\tau \|\tilde{\mathbf{s}}^{(\tau-1)}\|_{\mathbb{E}} + (L+\mu)a_\tau \|\Delta\mathbf{x}^{(\tau-1)}\|_{\mathbb{E}}\right). \tag{4.60}$$

Upon using Lemma 4.6 and

$$\|\Delta\mathbf{x}^{(\tau-1)}\|_{\mathbb{E}} \leq \|\mathbf{x}^{(\tau)} - \mathbf{y}^{(\tau)}\|_{\mathbb{E}} + \|\mathbf{x}^{(\tau-1)} - \mathbf{y}^{(\tau-1)}\|_{\mathbb{E}} + \|\Delta\mathbf{y}^{(\tau-1)}\|_{\mathbb{E}},$$

one has

$$\begin{aligned} a_\tau\|\Delta\mathbf{x}^{(\tau-1)}\|_{\mathbb{E}} &\leq \frac{a_\tau}{1+\mu A_\tau}\|\tilde{\mathbf{z}}^{(\tau)}\|_{\mathbb{E}} + \frac{a_\tau}{1+\mu A_{\tau-1}}\|\tilde{\mathbf{z}}^{(\tau-1)}\|_{\mathbb{E}} + a_\tau\|\Delta\mathbf{y}^{(\tau-1)}\|_{\mathbb{E}} \\ &= a\|\tilde{\mathbf{z}}^{(\tau)}\|_{\mathbb{E}} + \frac{a}{1-a\mu}\|\tilde{\mathbf{z}}^{(\tau-1)}\|_{\mathbb{E}} + a_\tau\|\Delta\mathbf{y}^{(\tau-1)}\|_{\mathbb{E}}, \end{aligned}$$

where the equality follows from (4.12). In light of (4.57), we have

$$a_\tau\|\Delta\mathbf{x}^{(\tau-1)}\|_{\mathbb{E}} \leq \left(a\beta + \frac{a}{1-a\mu}\right)\|\tilde{\mathbf{z}}^{(\tau-1)}\|_{\mathbb{E}} + a\beta a_\tau\|\tilde{\mathbf{s}}^{(\tau-1)}\|_{\mathbb{E}} + a_\tau\|\Delta\mathbf{y}^{(\tau-1)}\|_{\mathbb{E}}.$$

Therefore

$$\begin{aligned} a_{\tau+1}\|\tilde{\mathbf{s}}^{(\tau)}\|_{\mathbb{E}} \leq & \frac{\beta + a\beta(L+\mu)}{1-a\mu}a_\tau\|\tilde{\mathbf{s}}^{(\tau-1)}\|_{\mathbb{E}} + \frac{a(L+\mu)}{1-a\mu}\left(\beta + \frac{1}{1-a\mu}\right)\|\tilde{\mathbf{z}}^{(\tau-1)}\|_{\mathbb{E}} \\ & + \frac{L+\mu}{1-a\mu}a_\tau\|\Delta\mathbf{y}^{(\tau-1)}\|_{\mathbb{E}}. \end{aligned} \tag{4.61}$$

By combining (4.57) and (4.61), the following linear system inequality can be established:

$$\begin{bmatrix} \|\tilde{\mathbf{z}}^{(\tau)}\|_{\mathbb{E}} \\ a_{\tau+1}\|\tilde{\mathbf{s}}^{(\tau)}\|_{\mathbb{E}} \end{bmatrix} \leq \mathbf{M}\begin{bmatrix} \|\tilde{\mathbf{z}}^{(\tau-1)}\|_{\mathbb{E}} \\ a_\tau\|\tilde{\mathbf{s}}^{(\tau-1)}\|_{\mathbb{E}} \end{bmatrix} + \frac{L+\mu}{1-a\mu}\begin{bmatrix} 0 \\ a_\tau\|\Delta\mathbf{y}^{(\tau-1)}\|_{\mathbb{E}} \end{bmatrix}$$

where $\mathbf{M}$ is defined in (4.15). By iterating the preceding linear system inequality and using

$$\|\tilde{\mathbf{z}}^{(0)}\| = 0, \quad \|\tilde{\mathbf{s}}^{(0)}\| = \sqrt{\sum_{i=1}^{n}\left\|\nabla f_i(x^{(0)}) - \overline{g}^{(0)}\right\|^2} := \sigma,$$

we obtain

$$\begin{aligned}\begin{bmatrix}\|\tilde{\mathbf{z}}^{(t)}\|_{\mathbb{E}}\\ a_{t+1}\|\tilde{\mathbf{s}}^{(t)}\|_{\mathbb{E}}\end{bmatrix} \leq & \frac{L+\mu}{1-a\mu}\sum_{\tau=0}^{t-1}\mathbf{M}^{t-\tau-1}\sqrt{a_{\tau+1}}\begin{bmatrix}0\\ \sqrt{a_{\tau+1}}\|\Delta\mathbf{y}^{(\tau)}\|_{\mathbb{E}}\end{bmatrix}+\mathbf{M}^t\begin{bmatrix}0\\ a_1\sigma\end{bmatrix}\\ = & \sqrt{a_t}\frac{L+\mu}{1-a\mu}\sum_{\tau=0}^{t-1}\left(\mathbf{M}\sqrt{1-a\mu}\right)^{t-\tau-1}\begin{bmatrix}0\\ \sqrt{a_{\tau+1}}\|\Delta\mathbf{y}^{(\tau)}\|_{\mathbb{E}}\end{bmatrix}+\mathbf{M}^t\begin{bmatrix}0\\ a_1\sigma\end{bmatrix}.\end{aligned}$$

Recall the eigenvalues for matrix $\mathbf{M}$ are $\lambda_1=(\xi_1+\xi_2)/2$ and $\lambda_2=(\xi_1-\xi_2)/2$, where

$$\xi_1=\frac{\beta(2+aL)}{1-a\mu},\quad \xi_2=\frac{a(L+\mu)}{1-a\mu}\sqrt{\frac{\beta^2L^2}{(L+\mu)^2}+\frac{4\beta(\beta+1)}{a(L+\mu)}}.$$

Thus, the analytical form for the nth power of $\mathbf{M}$ is (see, e.g., [103])

$$\mathbf{M}^n=\lambda_1^n\left(\frac{\mathbf{M}-\lambda_2 I}{\lambda_1-\lambda_2}\right)+\lambda_2^n\left(\frac{\mathbf{M}-\lambda_1 I}{\lambda_2-\lambda_1}\right)=\lambda_1^n\left(\frac{\mathbf{M}-\lambda_2 I}{\lambda_1-\lambda_2}\right)-\lambda_2^n\left(\frac{\mathbf{M}-\lambda_1 I}{\lambda_1-\lambda_2}\right).$$

It then follows that

$$(\mathbf{M}^n)_{12}=\frac{\mathbf{M}_{12}(\lambda_1^n-\lambda_2^n)}{\lambda_1-\lambda_2}=\frac{\beta(\lambda_1^n-\lambda_2^n)}{\lambda_1-\lambda_2}\leq\frac{2\beta(\rho(\mathbf{M}))^n}{\xi_2},$$

where $\rho(\mathbf{M})$ is the spectral radius of $\mathbf{M}$. Due to our assumption that

$$\frac{1}{a}>\frac{\beta(2L+3\mu)}{(1-\beta)^2}+\mu>\frac{2\beta(L+\mu)}{(1-\beta)^2}$$

and $\beta\in(0,1)$, we have

$$\xi_2>\frac{2a\beta(L+\mu)}{1-a\mu}.$$

Therefore,

$$\begin{aligned}\|\tilde{\mathbf{z}}^{(t)}\|_{\mathbb{E}} \leq & \frac{2\beta\frac{L+\mu}{1-a\mu}\sqrt{a_t}}{\xi_2}\sum_{\tau=0}^{t-1}\nu^{t-\tau-1}\sqrt{a_{\tau+1}}\|\Delta\mathbf{y}^{(\tau)}\|_{\mathbb{E}}+\frac{2\beta}{\xi_2}\left(\rho(\mathbf{M})\right)^t a_1\sigma\\ \leq & \frac{\sqrt{a_t}}{a}\sum_{\tau=0}^{t-1}\nu^{t-\tau-1}\sqrt{a_{\tau+1}}\|\Delta\mathbf{y}^{(\tau)}\|_{\mathbb{E}}+\frac{1-a\mu}{a(L+\mu)}\left(\rho(\mathbf{M})\right)^t a_1\sigma.\end{aligned}\tag{4.62}$$

This bound, together with Lemma 4.6, yields

$$\begin{aligned}\sqrt{a_{t+1}}\|\mathbf{x}^{(t)}-\mathbf{y}^{(t)}\|_{\mathbb{E}} \leq & \frac{\sqrt{a_{t+1}}}{1+\mu A_t}\left(\frac{\sqrt{a_t}}{a}\sum_{\tau=0}^{t-1}\nu^{t-\tau-1}\sqrt{a_{\tau+1}}\|\Delta\mathbf{y}^{(\tau)}\|_{\mathbb{E}}+\frac{\left(\rho(\mathbf{M})\right)^t}{L+\mu}\sigma\right)\\ \leq & \frac{\sum_{\tau=0}^{t-1}\nu^{t-\tau-1}\sqrt{a_{\tau+1}}\|\Delta\mathbf{y}^{(\tau)}\|_{\mathbb{E}}}{\sqrt{1-a\mu}}+\frac{\sqrt{\frac{a}{1-a\mu}}\nu^t\sigma}{L+\mu}.\end{aligned} \tag{4.63}$$

The desired inequality (4.28) then follows from this and Lemma 4.7. □

4.6.3 Proof of Lemma 4.4

Proof of Lemma 4.4. Define

$$m_t(x) := \sum_{\tau=0}^{t-1} a_{\tau+1}\left(\langle\overline{g}^{(\tau)},x\rangle+\frac{\mu}{2}\|x-\overline{x}^{(\tau)}\|^2+h(x)\right)+d(x)$$

where $m_0(x)=d(x)$. Due to $\overline{z}^{(t)}=\sum_{\tau=0}^{t-1}a_{\tau+1}(\overline{g}^{(\tau)}-\mu\overline{x}^{(\tau)})$ in Lemma 4.5, we can equivalently express (4.18) as

$$y^{(t)}=\operatorname*{argmin}_{x\in\mathbb{R}^m} m_t(x).$$

Since $m_{\tau-1}(x)$ is strongly convex with modulus $1+\mu A_{\tau-1}$, we have

$$m_{\tau-1}(x)-m_{\tau-1}(y^{(\tau-1)})\geq\frac{1}{2}(1+\mu A_{\tau-1})\|x-y^{(\tau-1)}\|^2,\forall x\in\mathrm{dom}(h).$$

Further, by noticing

$$m_\tau(x)=m_{\tau-1}(x)+a_\tau\left(\langle\overline{g}^{(\tau-1)},x\rangle+\frac{\mu}{2}\|x-\overline{x}^{(\tau-1)}\|^2+h(x)\right),$$

we have

$$\begin{aligned}0\leq & m_{\tau-1}(y^{(\tau)})-m_{\tau-1}(y^{(\tau-1)})-\frac{1}{2}\left(1+\mu A_{\tau-1}\right)\|y^{(\tau)}-y^{(\tau-1)}\|^2\\ = & m_\tau(y^{(\tau)})-a_\tau\left(\langle\overline{g}^{(\tau-1)},y^{(\tau)}\rangle+\frac{\mu}{2}\|y^{(\tau)}-\overline{x}^{(\tau-1)}\|^2+h(y^{(\tau)})\right)-m_{\tau-1}(y^{(\tau-1)})\\ & -\frac{1}{2}\left(1+\mu A_{\tau-1}\right)\|y^{(\tau)}-y^{(\tau-1)}\|^2,\end{aligned}$$

which is equivalent to

$$
\begin{aligned}
&a_\tau \left(\langle \overline{g}^{(\tau-1)}, y^{(\tau)}\rangle + h(y^{(\tau)})\right) \\
&\leq m_\tau(y^{(\tau)}) - m_{\tau-1}(y^{(\tau-1)}) - \frac{1}{2}\Big(1+\mu A_{\tau-1}\Big)\|y^{(\tau)} - y^{(\tau-1)}\|^2 - \frac{\mu}{2}a_\tau \|y^{(\tau)} - \overline{x}^{(\tau-1)}\|^2.
\end{aligned}
$$

Summing up the above inequality from $\tau = 1$ to $\tau = t$ leads to

$$
\begin{aligned}
&\sum_{\tau=1}^{t} a_\tau \left(\langle \overline{g}^{(\tau-1)}, y^{(\tau)}\rangle + h(y^{(\tau)})\right) \\
&\leq m_t(y^{(t)}) - m_0(y^{(0)}) - \sum_{\tau=1}^{t} \frac{1}{2}\left((1+\mu A_{\tau-1})\|\Delta y^{(\tau-1)}\|^2 + \mu a_\tau \|y^{(\tau)} - \overline{x}^{(\tau-1)}\|^2\right) \\
&= m_t(y^{(t)}) - \sum_{\tau=1}^{t} \frac{1}{2}\left((1+\mu A_{\tau-1})\|y^{(\tau)} - y^{(\tau-1)}\|^2 + \mu a_\tau \|y^{(\tau)} - \overline{x}^{(\tau-1)}\|^2\right).
\end{aligned} \tag{4.64}
$$

where the equality is due to $y^{(0)} = x^{(0)}$ and (5.2). By a similar argument to (4.11), we obtain

$$
\sum_{\tau=1}^{t} \langle a_\tau \overline{g}^{(\tau-1)}, -x^*\rangle \leq -m_t(y^{(t)}) + d(x^*) + \sum_{\tau=1}^{t} a_\tau \left(\frac{\mu}{2}\|x^* - \overline{x}^{(\tau-1)}\|^2 + h(x^*)\right),
$$

which in conjunction with (4.64) leads to the inequality in (4.29). □

4.7 Experiments

For the experiments, we consider the decentralized sparse logistic regression problem [1] and the decentralized LASSO problem [96]. We present numerical results of Algorithm 3 (named as DDA below), and compare it with the following algorithms:

i) PG-EXTRA in [84]:

$$
\begin{aligned}
\mathbf{z}^{(t)} =& \mathbf{z}^{(t-1)} - \mathbf{x}^{(t-1)} + \tilde{\mathbf{P}}(2\mathbf{x}^{(t-1)} - \mathbf{x}^{(t-2)}) - a(\nabla^{(t-1)} - \nabla^{(t-2)}) \\
\mathbf{x}^{(t)} =& \mathrm{Prox}^a_{\mathbf{h}}(\mathbf{z}^{(t)}),
\end{aligned}
$$

where $\tilde{\mathbf{P}} = \frac{(I+P)\otimes I}{2}$, $\mathbf{h}(\mathbf{x}) = \sum_{i=1}^n h(x_i)$, and

$$\mathrm{Prox}_{\mathbf{h}}^a(\mathbf{z}) := \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^{mn}} \left\{ \mathbf{h}(\mathbf{x}) + \frac{1}{2a}\|\mathbf{x}-\mathbf{z}\|^2 \right\}.$$

ii) P2D2 in [1]:

$$\begin{aligned}\mathbf{z}^{(t)} =& (I-\alpha\mathbf{B})\,\mathbf{z}^{(t-1)} + (I-\mathbf{B})\,(\mathbf{x}^{(t-1)} - \mathbf{x}^{(t-2)}) - a(\nabla^{(t-1)} - \nabla^{(t-2)})\\ \mathbf{x}^{(t)} =& \mathrm{Prox}_{\mathbf{h}}^a(\mathbf{z}^{(t)}),\end{aligned}$$

where $\mathbf{B} = \frac{(I-P)\otimes I}{2}$.

iii) DSM in [55]:

$$\mathbf{x}^{(t)} = \mathbf{P}^{(t-1)}\mathbf{x}^{(t-1)} - a_{t-1}\mathbf{r}^{(t-1)},$$

where $\mathbf{r}^{(t)} \in \partial\mathbf{F}(\mathbf{x}^{(t)})$ and $\mathbf{F}(\mathbf{x}) = \sum_{i=1}^n F(x_i)$.

iv) Conventional DDA (named as C-DDA below) in [18]:

$$\begin{aligned}\mathbf{z}^{(t)} =& \mathbf{P}^{(t-1)}\mathbf{z}^{(t-1)} + \mathbf{r}^{(t-1)}\\ \mathbf{x}^{(t)} =& \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^{mn}} \left\{ a_{t-1}\langle \mathbf{z}^{(t)}, \mathbf{x}\rangle + \mathbf{d}(\mathbf{x}) \right\},\end{aligned}$$

where $\mathbf{d}(\mathbf{x}) = \sum_{i=1}^n d(x_i)$.

We note that when applied to solve Problem (4.1) in stochastic networks, PG-EXTRA and P2D2 have no convergence guarantees and DSM and C-DDA have sublinear convergence in theory.

4.7.1 Decentralized Logistic Regression

The aforementioned algorithms are applied to the following problem:

$$\min_{x\in\mathbb{R}^m} \frac{1}{n}\sum_{i=1}^n f_i(x) + \phi\|x\|_1, \qquad f_i(x) = \frac{1}{m_i}\sum_{j=1}^{m_i} \ln(1+\exp(-y_j^i M_j^{i\mathrm{T}} x)) + \frac{\mu}{2}\|x\|^2, \quad (4.65)$$

where $\{M_j^i, y_j^i\}_{j=1}^{m_i}$ are data samples private to agent i. In our experiment, we set $\phi = 0.001$, $\mu = 0.02$, and use *Spambase* data set in the UCI Machine Learning

Repository [17] to generate our problem instance. In particular, we extract 3000 out of the total 4601 samples in the original data set and evenly distribute them to the $n = 30$ agents, i.e., $m_i = 100$ for all i.

We consider two common configurations of stochastic communication networks. The first one is *Bernoulli networks* [31], where a fixed graph is first generated and at any time t, each edge of the fixed graph is sampled with probability $\iota \in (0, 1)$, which results in a random sub-graph of the fixed graph. In our experiment, we generate two fixed graphs in the same way as [85], where the sparsity parameter ξ, i.e., the ratio between the number of edges in the generated fixed graph and the number of edges in the complete graph, is chosen to be 0.2 and 0.4, respectively. Based on each fixed graph, we generate two Bernoulli networks with ι set to be 0.1 and 0.2, respectively. The second one is *randomized gossip networks* [4], where only a single edge of a fixed graph is sampled at any time t. In particular, the probability to sample the link (i, j) is set as $\frac{1}{n(|\mathcal{N}_i|+1)}$ with $|\mathcal{N}_i|$ representing the number of neighbors of i in the supergraph at every time t. In our experiment, we consider cycle graph, 2D grid, and complete graph as the fixed graphs for generating randomized gossip networks.

For all the tested algorithms, we evaluate their performance in terms of the relative square error (RSE) defined by $\text{RSE} = \frac{\sum_{i=1}^n \|x_i^{(t)} - x^*\|^2}{\sum_{i=1}^n \|x_i^{(0)} - x^*\|^2}$, where x^* is identified by applying the centralized proximal gradient method [72] to Problem (4.65) such that the norm of the difference of two consecutive iterates is less than 10^{-14}. All the algorithms are initialized with $x_i^{(0)} = 0$ for all agents i. The parameters of each algorithm are chosen properly to reflect their performance. For DDA and C-DDA, we simply choose $d(x) = \|x\|^2/2$. We choose $\alpha = 0.5$ in P2D2 and set $a_t = 1/\sqrt{t+1}$ for C-DDA and DSM. For the first group of Bernoulli networks, i.e., those sampled from a supergraph with sparsity parameter 0.2 (first row of Figure 4.1), we set the same $a = 0.1$ for DDA, P2D2, and PG-EXTRA. For the second group of Bernoulli networks, i.e., those sampled from a supergraph with sparsity parameter 0.4 (second row of Figure 4.1), we use the same $a = 0.2$ for DDA, P2D2, and PG-EXTRA. For randomized gossip, we use $a = 0.1$ for DDA, and set 10^{-4} for the step sizes in P2D2 and PG-EXTRA. We note that choosing a smaller step size in P2D2 and PG-EXTRA generally makes them more stabilizing. In fact, a larger step size will result in even worse behaviour of these two methods in randomized gossip networks.

The simulation results are plotted in Figure 4.1. Specifically, the first two rows of Figure 4.1 present the performance on Bernoulli networks and the last row shows the performance on randomized gossip networks. We note that the rightmost two

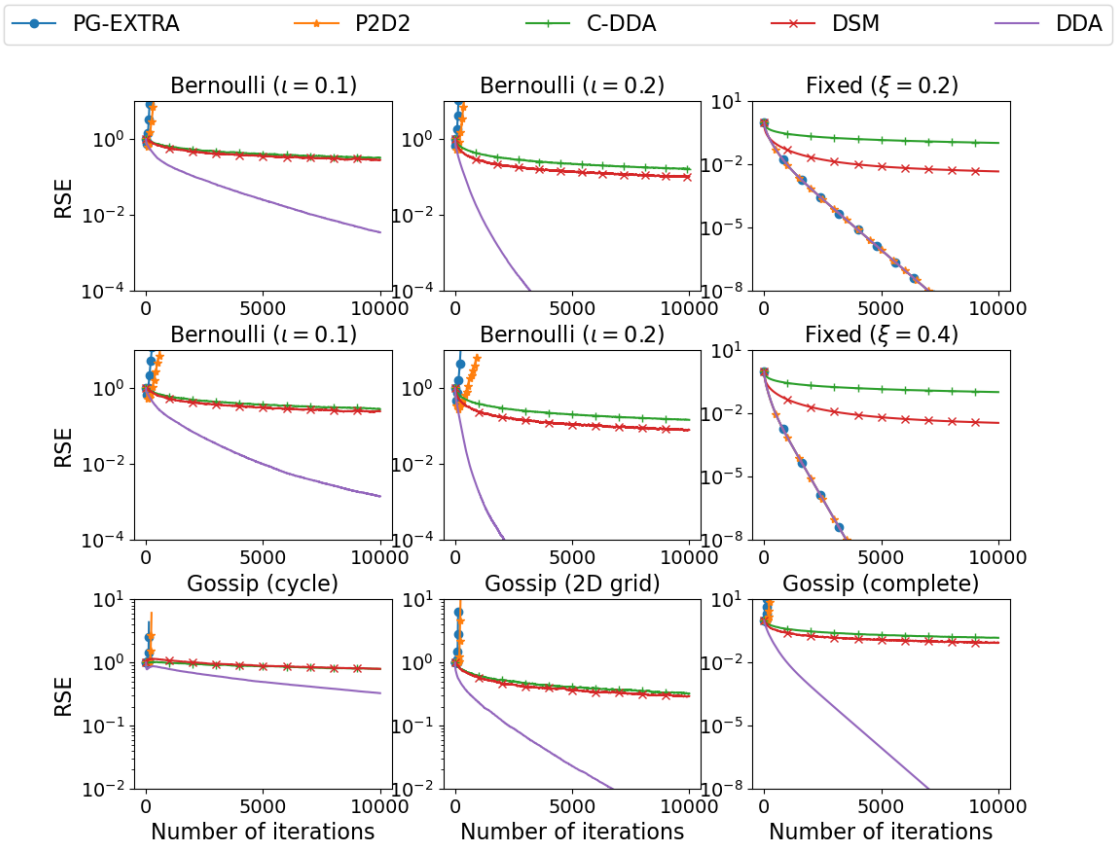


Figure 4.1: Comparison results for decentralized logistic regression in different network configurations.

plots of the first two rows are for time-invariant networks, which are the fixed graphs for generating the Bernoulli networks in the first two rows, respectively. One can observe that our DDA converges linearly and is substantially faster than DSM and C-DDA in all the network settings, which supports our theoretical development. In addition, while P2D2 and PG-EXTRA perform very similar to DDA on time-invariant networks, they both diverge when applied to stochastic networks. This suggests that decentralized algorithms that are designed for time-invariant networks may not work effectively in stochastic networks.

4.7.2 Decentralized LASSO

The following decentralized LASSO problem is considered

$$\min_{x\in\mathbb{R}^m} \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2}\|b_i - C_i x\|^2, \quad \text{s.t. } \|x\|_1 \leq R,$$

where $R > 0$ is a constant, and (C_i, b_i) represents the data tuple available to agent i with $C_i \in \mathbb{R}^{60\times 50}$ and $b_i \in \mathbb{R}^{60}$. The data is randomly generated according to the setting by [45]. Firstly, a sparse signal $x^\sharp \in \mathbb{R}^{50}$ is randomly generated, where the probability for each element being nonzero is 0.25. Then, each C_i is randomly generated and then normalized such that Assumption 4.1 holds with $L = 1$ and $\mu = 0.5$. Set $R = 1.1\|x^\sharp\|_1$. produced based on $b_i = C_i x^\sharp + \epsilon_i$, where ϵ_i is a random noise vector.

Two types of stochastic communication networks are considered. For Bernoulli networks, we generate two fixed graphs with the sparsity parameter $\xi = 0.1$ and 0.2. Based on each fixed graph, we construct two Bernoulli networks by setting $\iota = 0.05$ and $\iota = 0.1$, respectively. In the second setting, we also consider cycle graph, 2D grid, and complete graph as the fixed graphs for generating randomized gossip networks. We identify x^* by using the centralized proximal gradient method, where the stopping criterion is set as the norm of the difference of two consecutive iterates smaller than 10^{-14}. The performance of all the tested algorithms is evaluated in terms of RSE $= \frac{\sum_{i=1}^n \|x_i^{(t)} - x^*\|^2}{\sum_{i=1}^n \|x_i^{(0)} - x^*\|^2}$. The algorithm by [13] is used to perform projection onto l_1-norm ball. All the algorithms are initialized with $x_i^{(0)} = 0$ for all i. The parameters for each algorithm are chosen in the following way. For DDA and C-DDA, we employ $d(x) = \|x\|^2/2$. We choose $\alpha = 0.5$ in P2D2 and set $a_t = 1/\sqrt{t+1}$ for C-DDA. For the two groups of Bernoulli networks, we set the a in DDA to be 0.1 and set 0.1 for

the step sizes in P2D2 and PG-EXTRA. For randomized gossip, we use $a = 0.1$ for DDA, and set 10^{-4} for the step sizes in P2D2 and PG-EXTRA. Since DSM can not be applied to constrained problems, it is not considered in this setting.

The simulation results are plotted in Figure 4.2. In particular, the performance on Bernoulli networks and randomized gossip networks is presented in the first two rows and the last row of Figure 4.2, respectively. In the first two rows, the rightmost two plots demonstrate the performance in time-invariant networks that are used for generating Bernoulli networks. Although P2D2 and PG-EXTRA demonstrate a similar performance with DDA on time-invariant networks, they do not converge to the minimizer when applied to stochastic networks. In line with our theoretical results, DDA linearly converges and outperforms C-DDA in all the network configurations.

To summarize, the simulation results confirm our theoretical findings and demonstrate the superior performance of the proposed Algorithm 3 on both time-invariant and stochastic networks.

4.8 Conclusion

In this chapter, we have proposed a new decentralized algorithm for solving Problem (4.1) in stochastic networks. The proposed algorithm, based on the framework of dual averaging method, is facilitated by designing a novel dynamic averaging consensus protocol. To the best of our knowledge, this is the first linearly convergent DDA-type decentralized algorithm and also the first algorithm that attains global linear convergence for solving Problem (4.1) in stochastic networks.

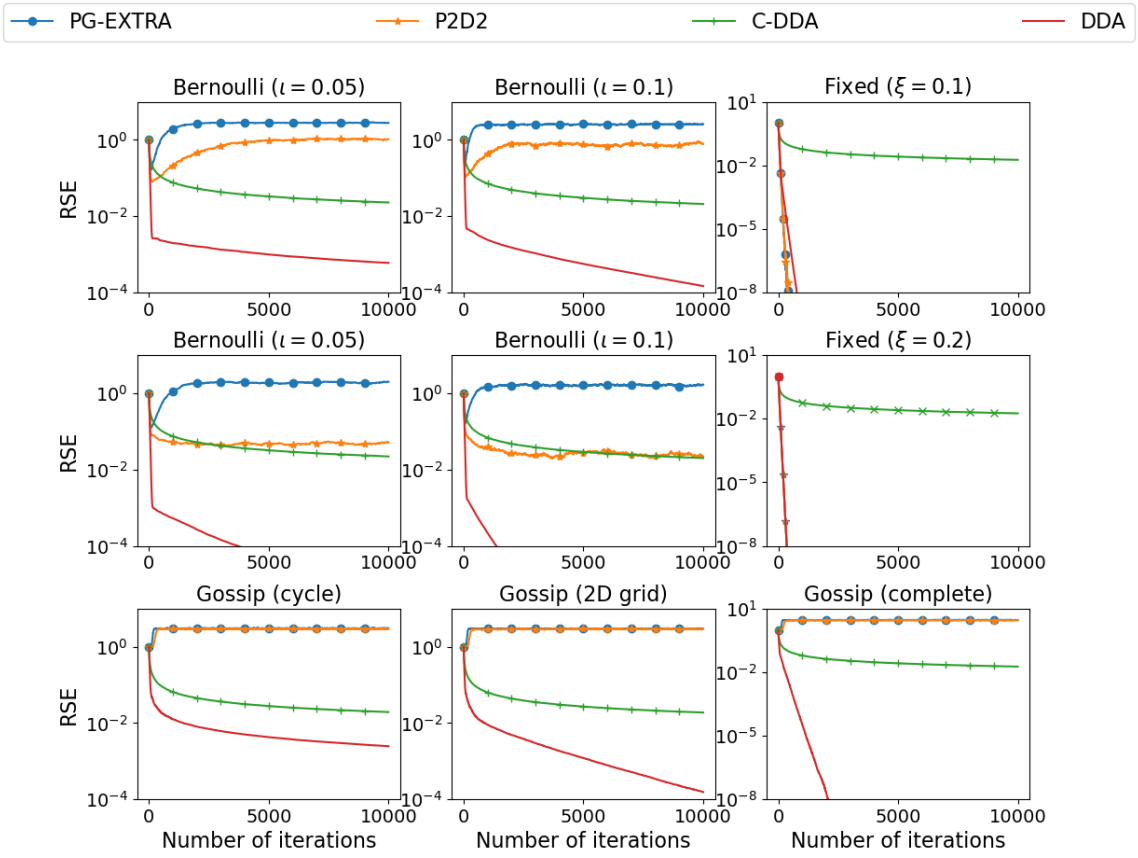


Figure 4.2: Comparison results for decentralized LASSO in different network configurations.

Chapter 5

Accelerated Decentralized Dual Averaging Method

5.1 Introduction

Consider an MAS consisting of n agents. They are connected via a communication network in order to collaboratively solve the following optimization problem:

$$\min_{x\in\mathcal{X}} \left\{ f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x) \right\} \tag{5.1}$$

where f_i represents the local smooth objective function of agent i and $\mathcal{X} \subseteq \mathbb{R}^m$ denotes the constraint set shared by all the agents.

Over the last decade, several accelerated decentralized optimization algorithms have been proposed for solving Problem (5.1). For *unconstrained* problems, i.e., $\mathcal{X} = \mathbb{R}^m$, the authors in [74] developed a decentralized Nesterov gradient descent, where the rate of convergence is accelerated to $\mathcal{O}(1/t^{1.4-\epsilon})$ for any $\epsilon \in (0, 1.4)$ at the expense of exchanging an additional variable among agents at each time instant. In [26], the authors proposed an accelerated decentralized algorithm with multiple consensus rounds at each time instant, and proved that after t local iterations and $\mathcal{O}(t \log t)$ communication rounds the objective error is bounded by $\mathcal{O}(1/t^2)$. By modeling Problem (5.1) as a linearly constrained optimization problem, centralized primal-dual paradigms such as the augmented Lagrangian method (ALM), the alternating direction method of multipliers (ADMM) and the dual ascent can also be used to design decentralized algorithms [24,85,91]. In particular, the authors in [91] developed

a modified dual formulation whose solution solves the original problem with prescribed accuracy for non-strongly convex and smooth objectives. In doing so, the primal objective becomes strongly convex, which leads to better theoretical results. Based on the primal-dual reformulation, an accelerated decentralized primal-dual method was developed in [109]. The rate of convergence is improved to $\mathcal{O}(1)\left(\frac{L}{t^2}+\frac{1}{t\sqrt{\eta}t}\right)$, where L denotes the smoothness parameter of each objective function and $\eta=\lambda_2(\mathcal{L})/\lambda_m(\mathcal{L})$ is the eigengap of the graph Laplacian $\mathcal{L}$. Notably, the authors in [109] established a lower bound for a class of decentralized primal-dual methods, suggesting that the developed algorithm therein is optimal in terms of gradient computations. The authors in [77] considered the Lagrangian dual formulation of the decentralized optimization problem and developed two algorithms based on dual accelerated methods. The algorithms are proved to be linearly convergent for strongly convex and smooth problems. Note that such a framework requires computing the convex conjugate of the objective at each iteration. For *constrained* problems, the authors in [40] proposed an accelerated decentralized penalty method (APM), where the constraint can be handled by incorporating a non-smooth indicator to the objective function.

For Problem (5.1), we provided a DDA method in Chapter 4 that has an $\mathcal{O}(1/t)$ rate of convergence under the smoothness assumption. Considering this, a question naturally arises: Is it possible to further accelerate the convergence rate of DDA? We provide affirmative answer to this question in this chapter. The main results and contributions are summarized in the following:

i) We propose an accelerated DDA (ADDA) algorithm. Different from DDA in Chapter 4 , each agent employs a first-order dynamic average consensus protocol to estimate the mean of local gradients and accumulates the estimate over time to generate a local dual variable. By solving the convex conjugate of a 1-strongly convex function over this local dual variable, each agent produces a primal variable and uses it to construct another two sequences of primal variables in an iterative manner based on the extrapolation technique in [11] and the average consensus protocol. The rate of convergence is proved to be $\mathcal{O}(1)\left(\frac{1}{t^2}+\frac{1}{t(1-\beta)^2}\right)$, where β denotes the second largest singular value of the mixing matrix. Notably, the condition for the algorithmic parameter to ensure convergence does not rely on the mixing matrix. Establishing such a condition that is independent on the mixing matrix offers the appealing advantage of convenient verification in practical applications.

ii) The proposed algorithms are tested and compared with a few methods in the literature on decentralized LASSO problems characterized by synthetic and real datasets. The comparison results demonstrate the efficiency of the proposed methods.

5.2 Problem Setup and Preliminaries

5.2.1 Problem Setup

We consider the finite-sum optimization problem (5.1), in which f_i satisfies the following assumptions for all $i = 1, \ldots, n$:

Assumption 5.1. *i) f_i is continuously differentiable on $\mathcal{X}$;*

ii) f_i is convex with on $\mathcal{X}$;

iii) ∇f_i is Lipschitz continuous on $\mathcal{X}$ with constant $L > 0$.

Throughout the paper, we denote by x^* an optimal solution of Problem (5.1). We consider solving Problem (5.1) in a decentralized fashion, that is, a pair of agents can exchange information only if they are connected in the communication network. We use a fixed undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ and a mixing matrix $P = [p_{ij}]$ to describe the network topology. The following standard assumption is made for P.

Assumption 5.2. *i) $P\mathbf{1} = \mathbf{1}$ and $\mathbf{1}^{\mathrm{T}} P = \mathbf{1}^{\mathrm{T}}$;*

ii) P has a strictly positive diagonal.

5.2.2 Centralized Accelerated Dual Averaging

Our algorithm is based on the centralized accelerated dual averaging method [11] that is applicable to solving Problem (5.1) in a centralized manner. Let d be a strongly convex and differentiable function with modulus 1 on $\mathcal{X}$ such that

$$x^{(0)} = \operatorname*{argmin}_{x \in \mathcal{X}} d(x) \quad \text{and} \quad d(x^{(0)}) = 0. \tag{5.2}$$

Starting with $u^{(1)} = w^{(0)} = x^{(0)}$, $v^{(1)} = w^{(1)}$, and

$$v^{(1)} = w^{(1)} = \operatorname*{argmin}_{x \in \mathcal{X}} \left\{ 2a \left\langle \nabla f(u^{(1)}), x \right\rangle + d(x) \right\},$$

the variables $\{u^{(t)}\}_{t\geq 1}$, $\{v^{(t)}\}_{t\geq 1}$, and $\{w^{(t)}\}_{t\geq 1}$ are generated iteratively according to

$$u^{(t)} = \frac{A_{t-1}}{A_t} v^{(t-1)} + \frac{a_t}{A_t} w^{(t-1)} \tag{5.3a}$$

$$v^{(t)} = \frac{A_{t-1}}{A_t} v^{(t-1)} + \frac{a_t}{A_t} w^{(t)}, \tag{5.3b}$$

where $a_t = a(t+1)$ for some $a > 0$, $A_t = \sum_{\tau=1}^{t} a_\tau$ and

$$w^{(t)} = \underset{x\in\mathcal{X}}{\operatorname{argmin}} \left\{ \sum_{\tau=1}^{t} a_\tau \left\langle \nabla f(u^{(\tau)}), x \right\rangle + d(x) \right\} \tag{5.4}$$

For convex and smooth objective functions, it is proved that $f(v^{(t)}) - f(x^*) \leq \mathcal{O}(1/t^2)$ [11].

5.3 Algorithm and Convergence Result

To solve Problem (5.1) over networks, we develop a decentralized variant of the accelerated dual averaging method in (5.3) and (5.4). In particular, we consider building consensus among variables $\{v_i^{(t)}, i = 1, \cdots, n\}$ and propose the following iteration rule:

$$u_i^{(t)} = \frac{A_{t-1}}{A_t} \sum_{j=1}^{n} p_{ij} v_j^{(t-1)} + \frac{a_t}{A_t} w_i^{(t-1)} \tag{5.5a}$$

$$v_i^{(t)} = \frac{A_{t-1}}{A_t} \sum_{j=1}^{n} p_{ij} v_j^{(t-1)} + \frac{a_t}{A_t} w_i^{(t)}, \tag{5.5b}$$

where

$$w_i^{(t)} = \underset{x\in\mathcal{X}}{\operatorname{argmin}} \left\{ \sum_{\tau=1}^{t} a_\tau \left\langle q_i^{(\tau)}, x \right\rangle + d(x) \right\}, \tag{5.6}$$

and $\{q_i^{(\tau)}, i = 1, \cdots, n\}_{t\geq 1}$ is updated via the following dynamic average consensus protocol

$$q_i^{(t)} = \sum_{j=1}^{n} p_{ij} q_j^{(t-1)} + \nabla f_i(u_i^{(t)}) - \nabla f_i(u_i^{(t-1)}). \tag{5.7}$$

The overall algorithm is summarized in Algorithm 4.

Assumption 5.3. *For the problem in* (5.1)*, the constraint set $\mathcal{X}$ is bounded with the*

Algorithm 4 Accelerated Decentralized Dual Averaging

Input: $a > 0$, $x^{(0)} \in \mathcal{X}$ and a strongly convex function d with modulus 1 such that (5.2) holds

Initialize: $A_1 = a_1 = 2a$, $u_i^{(1)} = w_i^{(0)} = x^{(0)}$, $q_i^{(1)} = \nabla f_i(x^{(0)})$, and $v_i^{(1)} = w_i^{(1)}$ for all $i = 1, \ldots, n$

for $t = 2, 3, \cdots$ **do**

set $a_t = a_{t-1} + a$ and $A_t = A_{t-1} + a_t$

In parallel (task for agent i, $i = 1, \ldots, n$)

collect $v_j^{(t-1)}$ and $q_j^{(t-1)}$ from all agents $j \in \mathcal{N}_i$

update $u_i^{(t)}$ by (5.5a)

update $q_i^{(t)}$ by (5.7)

compute $w_i^{(t)}$ by (5.6)

update $v_i^{(t)}$ by (5.5b)

broadcast $v_i^{(t)}$ and $q_i^{(t)}$ to all agents $j \in \mathcal{N}_i$

end for

following diameter:

$$G = \max_{x,y \in \mathcal{X}} \|x - y\|.$$

Theorem 5.1. *For Algorithm 4, if Assumptions 5.1, 5.2, and 5.3 are satisfied, and*

$$a \leq \frac{1}{6L}, \tag{5.8}$$

then, for all $t \geq 1$, it holds that

$$f(\overline{v}^{(t)}) - f(x^*) \leq \frac{d(x^*)}{A_t} + \frac{t}{A_t}\left(\frac{2G(LC_p + C_g)}{\sqrt{n}} + \frac{6LC_p^2}{n}\right), \tag{5.9}$$

where

$$C_p := \lceil \frac{3}{1-\beta} \rceil \sqrt{n} G$$

and

$$C_g := 2L \lceil \frac{3}{1-\beta} \rceil \frac{\sqrt{n}G + C_p}{1-\beta}.$$

In addition, for all $t \geq 1$ and $i = 1, \cdots, n$, we have

$$\left\| v_i^{(t)} - \overline{v}^{(t)} \right\|^2 \leq \frac{2aC_p}{A_t}. \tag{5.10}$$

Proof. The proof is postponed to Appendix C. □

For Algorithm 4 and Theorem 5.1, the following remarks are in order.

i) Comparison with existing accelerated algorithms. Accelerated methods for decentralized constrained optimization are rarely reported in the literature. Recently, the authors in [39] developed the APM algorithm, where the iteration rule reads

$$y_i^{(t)} = x_i^{(t)} + \frac{\theta_t(1-\theta_{t-1})}{\theta_{t-1}}\left(x_i^{(t)} - x_i^{(t-1)}\right) \tag{5.11a}$$

$$s_i^{(t)} = \nabla f_i(y_i^{(t)}) + \frac{\beta_0}{\theta_t}\sum_{i=1}^{n} p_{ij}\left(y_i^{(t)} - y_j^{(t)}\right) \tag{5.11b}$$

$$x_i^{(t+1)} = \operatorname*{argmin}_{x\in\mathcal{X}} \left\| x - y_i^{(t)} + \frac{s_i^{(t)}}{L+\beta_0/\theta_t}\right\|^2 \tag{5.11c}$$

where $\beta_0 = L/\sqrt{1-\lambda_2(P)}$ and θ_t is a decreasing parameter satisfying

$$\theta_t = \frac{\theta_{t-1}}{1+\theta_{t-1}}$$

with $\theta_0 = 1$. Letting $\hat{s}_i^{(t)} = \theta_k s_i^{(t)}$, we can equivalently rewrite (5.11b) and (5.11c) as

$$\begin{aligned}
\hat{s}_i^{(t)} &= \theta_t \nabla f_i(y_i^{(t)}) + \beta_0 \sum_{i=1}^{n} p_{ij}\left(y_i^{(t)} - y_j^{(t)}\right) \\
x_i^{(t+1)} &= \operatorname*{argmin}_{x\in\mathcal{X}} \left\| x - y_i^{(t)} + \frac{\hat{s}_i^{(t)}}{L\theta_t+\beta_0}\right\|^2,
\end{aligned}$$

from which we can see that new gradients are assigned with decreasing weights, whereas increasing weights are used for ADDA in (5.6). The reason for such different choices of parameters may be two-fold. First, parameter choices in (centralized) primal gradient descent and dual averaging methods are intrinsically different. Second, APM gradually increases the penalty parameter $1/\theta_t$ in order to enforce consensus, which essentially dilutes the weight for gradients, as shown above. We will show in simulation that decreasing weights over time slows down convergence.

There are also a few other accelerated decentralized methods such as [74, 109], however they do not apply to constrained problems.

ii) Discussion about optimality. For ADDA, the rate of convergence is proved

to be

$$\mathcal{O}(1)\left(\frac{1}{t^2}+\frac{1}{t(1-\beta)^2}\right).$$

In light of the lower bound in [109], it is not optimal in terms of the dependence on β. In particular, the dominant term of the error in $\mathcal{O}(1/(t(1-\beta)^2))$ becomes larger as β grows, i.e., the network becomes more sparsely connected. This is mainly because we consider a one-consensus-one-gradient update in the algorithm. However, extending the algorithm in [109] to handle constraints may require further investigation. In the simulation section, we demonstrate the superiority of ADDA over existing decentralized constrained optimization algorithms.

5.4 Proof of Convergence Result

5.4.1 Notations and Supporting Lemmas

For Algorithm 4, we define

$$\mathbf{u}^{(t)}=\begin{bmatrix}u_1^{(t)}\\ \vdots \\ u_n^{(t)}\end{bmatrix},\quad \mathbf{v}^{(t)}=\begin{bmatrix}v_1^{(t)}\\ \vdots \\ v_n^{(t)}\end{bmatrix},\quad \mathbf{w}^{(t)}=\begin{bmatrix}w_1^{(t)}\\ \vdots \\ w_n^{(t)}\end{bmatrix},\quad \mathbf{q}^{(t)}=\begin{bmatrix}q_1^{(t)}\\ \vdots \\ q_n^{(t)}\end{bmatrix},\ \hat{\nabla}^{(t)}=\begin{bmatrix}\nabla f_1(u_1^{(t)})\\ \vdots \\ \nabla f_n(u_n^{(t)})\end{bmatrix},$$

$$\overline{u}^{(t)}=\frac{1}{n}\sum_{i=1}^n u_i^{(t)},\ \overline{v}^{(t)}=\frac{1}{n}\sum_{i=1}^n v_i^{(t)},\ \overline{w}^{(t)}=\frac{1}{n}\sum_{i=1}^n w_i^{(t)},$$

$$\overline{\hat{g}}^{(t)}=\frac{1}{n}\sum_{i=1}^n \nabla f_i(u_i^{(t)}),\ \overline{q}_t=\frac{1}{n}\sum_{i=1}^n q_i^{(t)},\ \tilde{\mathbf{w}}^{(t)}=\mathbf{w}^{(t)}-\mathbf{1}\otimes\overline{w}^{(t)},$$

$$\tilde{\mathbf{u}}^{(t)}=\mathbf{u}^{(t)}-\mathbf{1}\otimes\overline{u}^{(t)},\ \tilde{\mathbf{v}}^{(t)}=\mathbf{v}^{(t)}-\mathbf{1}\otimes\overline{v}^{(t)},\ \tilde{\mathbf{q}}^{(t)}=\mathbf{q}^{(t)}-\mathbf{1}\otimes\overline{q}^{(t)}.$$

Based on these notation, we present the steps in (5.5) and (5.7) in the following compact form

$$\mathbf{u}^{(t)}=\frac{A_{t-1}}{A_t}\left(\mathbf{P}\mathbf{v}^{(t-1)}\right)+\frac{a_t}{A_t}\mathbf{w}^{(t-1)}, \tag{5.12a}$$

$$\mathbf{v}^{(t)}=\frac{A_{t-1}}{A_t}\left(\mathbf{P}\mathbf{v}^{(t-1)}\right)+\frac{a_t}{A_t}\mathbf{w}^{(t)}, \tag{5.12b}$$

$$\mathbf{q}^{(t)}=\mathbf{P}\mathbf{q}^{(t-1)}+\hat{\nabla}^{(t)}-\hat{\nabla}^{(t-1)}, \tag{5.12c}$$

where $\mathbf{P} = P \otimes I$. According to (5.5), we have

$$\overline{u}^{(t)} = \frac{A_{t-1}}{A_t}\overline{v}^{(t-1)} + \frac{a_t}{A_t}\overline{w}^{(t-1)}, \tag{5.13a}$$

$$\overline{v}^{(t)} = \frac{A_{t-1}}{A_t}\overline{v}^{(t-1)} + \frac{a_t}{A_t}\overline{w}^{(t)}. \tag{5.13b}$$

Before proving Theorem 5.1, we present Lemma 5.1 that establishes decreasing upper bounds for consensus error vectors $\tilde{\mathbf{u}}^{(t)}$ and $\tilde{\mathbf{v}}^{(t)}$.

Lemma 5.1. *For Algorithm 4, if Assumptions 5.1, 5.2, and 5.3 are satisfied, then*

$$\|\tilde{\mathbf{v}}^{(t)}\| \leq \frac{a_t}{A_t}C_p, \quad \|\tilde{\mathbf{u}}^{(t)}\| \leq \frac{a_t}{A_t}C_p \tag{5.14}$$

for all $t \geq 1$, where $C_p = \lceil \frac{3}{1-\beta} \rceil \sqrt{n}G$, and $\beta = \sigma_2(P)$.

Proof of Lemma 5.1. Since both $u_i^{(t)}, v_i^{(t)}, i = 1, \cdots, n$, $\overline{u}^{(t)}$ and $\overline{v}^{(t)}$ are within the constraint set, we readily have

$$\|\mathbf{u}^{(t)} - \mathbf{1} \otimes \overline{u}^{(t)}\| \leq \sqrt{n}G$$
$$\|\mathbf{v}^{(t)} - \mathbf{1} \otimes \overline{v}^{(t)}\| \leq \sqrt{n}G$$

by Assumption 5.3. Upon using

$$\frac{a_t}{A_t} = \frac{2(t+1)}{t(t+3)} \geq \frac{1}{t}, \quad \forall t \geq 1$$

and the definition of $C_p = \lceil \frac{3}{1-\beta} \rceil \sqrt{n}G$, we have that (5.14) holds for

$$1 \leq t < \lceil \frac{3}{1-\beta} \rceil.$$

When

$$t \geq \lceil \frac{3}{1-\beta} \rceil,$$

we prove by an induction argument. Suppose that (5.14) holds for some $t \geq \lceil \frac{3}{1-\beta} \rceil$. Next, we examine the upper bounds for $\|\tilde{\mathbf{v}}^{(t+1)}\|$ and $\|\tilde{\mathbf{u}}^{(t+1)}\|$, respectively.

i) Upper bound for $\|\tilde{\mathbf{v}}^{(t+1)}\|$. Using

$$\mathbf{P}\mathbf{v}^{(t)} - \mathbf{1} \otimes \overline{v}^{(t)} = \left(\left(P - \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}\right) \otimes I\right) \tilde{\mathbf{v}}^{(t)},$$

(5.12b) and (5.13b), we obtain

$$\tilde{\mathbf{v}}^{(t+1)} = \frac{A_t}{A_{t+1}} \left(\left(P - \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n} \right) \otimes I \right) \tilde{\mathbf{v}}^{(t)} + \frac{a_{t+1}}{A_{t+1}} \tilde{\mathbf{w}}^{(t+1)}.$$

Calculating the norm of both sides of the above equality yields

$$\begin{aligned} \|\tilde{\mathbf{v}}^{t+1}\| =& \left\| \frac{A_t}{A_{t+1}} \left(\left(P - \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n} \right) \otimes I \right) \tilde{\mathbf{v}}^{(t)} + \frac{a_{t+1}}{A_{t+1}} \tilde{\mathbf{w}}^{(t+1)} \right\| \\ \leq& \beta \|\tilde{\mathbf{v}}^{(t)}\| + \frac{a_{t+1}}{A_{t+1}} \left\| \tilde{\mathbf{w}}^{(t+1)} \right\| \\ \leq& \frac{a_t}{A_t} \beta C_p + \frac{a_{t+1}}{A_{t+1}} \sqrt{n} C_p, \end{aligned}$$

where the last inequality follows from the hypothesis that $\|\tilde{\mathbf{v}}^{(t)}\| \leq a_t C_p / A_t$ and Assumption 5.3. Since a_t/A_t monotonically decreases with t, we have

$$\|\tilde{\mathbf{v}}^{t+1}\| \leq \frac{a_t}{A_t} \left(\beta C_p + \sqrt{n} G \right) \leq \frac{a_t}{A_t} C_p \left(\beta + \frac{1}{\lceil \frac{3}{1-\beta} \rceil} \right),$$

where the last inequality is due to $\sqrt{n} G = \frac{C_p}{\lceil \frac{3}{1-\beta} \rceil}$. It then remains to prove

$$\left(\beta + \frac{1}{\lceil \frac{3}{1-\beta} \rceil} \right) \leq \frac{A_t}{a_t} \cdot \frac{a_{t+1}}{A_{t+1}}, \quad \forall t \geq \lceil \frac{3}{1-\beta} \rceil \tag{5.15}$$

to obtain the bound for $\|\tilde{\mathbf{v}}^{(t+1)}\|$ as desired. To prove (5.15), we let

$$t_0 = \lceil \frac{3}{1-\beta} \rceil,$$

which implies

$$\frac{3}{t_0} \leq 1 - \beta.$$

Based on the above relation, we further obtain

$$\beta + \frac{1}{t_0} \leq \frac{t_0 - 2}{t_0} \leq \frac{t_0 + 2}{t_0 + 4}. \tag{5.16}$$

This in conjunction with

$$\frac{t(t+3)}{(t+1)(t+1)} \geq 1, \quad \forall t \geq 1$$

and the definitions of a_t and A_t yields

$$\beta + \frac{1}{t_0} \leq \frac{t_0+2}{t_0+4} \cdot \frac{t_0(t_0+3)}{(t_0+1)(t_0+1)} = \frac{A_{t_0}}{a_{t_0}} \cdot \frac{a_{t_0+1}}{A_{t_0+1}}.$$

Since $\frac{A_t a_{t+1}}{a_t A_{t+1}}$ monotonically increases with t, we have (5.15) satisfied.

ii) Upper bound for $\|\tilde{\mathbf{u}}^{(t+1)}\|$. Using the same arguments as above, we have

$$\begin{aligned}
\|\tilde{\mathbf{u}}^{t+1}\| =& \left\| \frac{A_t}{A_{t+1}} \left(\left(P - \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n} \right) \otimes I \right) \tilde{\mathbf{v}}^{(t)} + \frac{a_{t+1}}{A_{t+1}} \tilde{\mathbf{w}}^{(t)} \right\| \\
\leq & \beta \|\tilde{\mathbf{v}}^{(t)}\| + \frac{a_{t+1}}{A_{t+1}} \left\| \tilde{\mathbf{w}}^{(t)} \right\| \\
\leq & \frac{a_t}{A_t} \beta C_p + \frac{a_{t+1}}{A_{t+1}} \sqrt{n} C_p.
\end{aligned}$$

By following the same line of reasoning as in the first part, we are able to obtain

$$\|\tilde{\mathbf{u}}^{(t+1)}\| \leq \frac{a_{t+1}}{A_{t+1}} C_p.$$

Summarizing the above bounds, the proof is completed. □

Lemma 5.2 proves the upper bound for the consensus vector $\tilde{\mathbf{q}}^{(t)}$.

Lemma 5.2. *Suppose Assumptions 5.1, 5.2, and 5.3 are satisfied. For Algorithm 4, we have*

$$\overline{q}^{(t)} = \overline{\hat{g}}^{(t)} \tag{5.17}$$

and

$$\|\tilde{\mathbf{q}}^{(t)}\| \leq \frac{a_t}{A_t} C_g \tag{5.18}$$

for all $t \geq 1$, *where* $C_g = \lceil \frac{3}{1-\beta} \rceil L \Big(2\sqrt{n} G + 2C_p \Big) / (1-\beta)$, *and* $\beta = \sigma_2(P)$.

Proof of Lemma 5.2. The proof of (5.17) directly follows from the proof of Lemma 4.5, and is omitted here for brevity.

For (5.18), we subtract $\mathbf{1} \otimes \overline{q}^{(t)}$ from both sides of (5.12c) to get

$$\begin{aligned}
\mathbf{q}^{(t)} - \mathbf{1} \otimes \overline{q}^{(t)} =& \mathbf{P}\mathbf{q}^{(t-1)} - \mathbf{1} \otimes \overline{q}^{(t-1)} \\
& + \hat{\nabla}^{(t)} - \hat{\nabla}^{(t-1)} - \mathbf{1} \otimes (\overline{q}^{(t)} - \overline{q}^{(t-1)}).
\end{aligned} \tag{5.19}$$

Using the same procedure in (4.55) leads to

$$\|\tilde{\mathbf{q}}^{(t)}\| \leq \beta\|\tilde{\mathbf{q}}^{(t-1)}\| + \|\hat{\nabla}^{(t)} - \hat{\nabla}^{(t-1)} - \mathbf{1} \otimes (\overline{q}^{(t)} - \overline{q}^{(t-1)})\|. \tag{5.20}$$

Since the objective is smooth, we obtain

$$\begin{aligned}
&\left\|\hat{\nabla}^{(t)} - \hat{\nabla}^{(t-1)} - \mathbf{1} \otimes \left(\overline{q}^{(t)} - \overline{q}^{(t-1)}\right)\right\| \\
=&\left\|\hat{\nabla}^{(t)} - \hat{\nabla}^{(t-1)} - \mathbf{1} \otimes \left(\overline{\hat{g}}^{(t)} - \overline{\hat{g}}^{(t-1)}\right)\right\| \\
=&\left\|\hat{\nabla}^{(t)} - \hat{\nabla}^{(t-1)} - \left(\frac{\mathbf{1}\mathbf{1}^T}{n} \otimes I\right)\left(\hat{\nabla}^{(t)} - \hat{\nabla}^{(t-1)}\right)\right\| \\
\leq&\left\|\hat{\nabla}^{(t)} - \hat{\nabla}^{(t-1)}\right\| \leq L\left\|\mathbf{u}^{(t)} - \mathbf{u}^{(t-1)}\right\|.
\end{aligned}$$

To bound $\left\|\mathbf{u}^{(t)} - \mathbf{u}^{(t-1)}\right\|$, we consider

$$\begin{aligned}
&\mathbf{u}^{(t)} - \mathbf{u}^{(t-1)} \\
=&\frac{A_{t-1}}{A_t}\mathbf{P}\mathbf{v}^{(t-1)} + \frac{a_t}{A_t}\mathbf{w}^{(t-1)} - \mathbf{u}^{(t-1)} \\
=&\frac{A_{t-1}}{A_t}\mathbf{P}\left(\mathbf{v}^{(t-1)} - \mathbf{u}^{(t-1)}\right) + \frac{A_{t-1}}{A_t}\left(\mathbf{P} - I \otimes I\right)\mathbf{u}^{(t-1)} \\
&+ \frac{a_t}{A_t}\left(\mathbf{w}^{(t-1)} - \mathbf{u}^{(t-1)}\right)
\end{aligned}$$

where the first equality is due to (5.12a). From (5.12a) and (5.12b), we have

$$\mathbf{v}^{(t-1)} - \mathbf{u}^{(t-1)} = \frac{a_{t-1}}{A_{t-1}}\left(\mathbf{w}^{(t-1)} - \mathbf{w}^{(t-2)}\right).$$

In addition, we have

$$\left(\mathbf{P} - I \otimes I\right)\mathbf{u}^{(t-1)} = \left(\mathbf{P} - I \otimes I\right)\left(\mathbf{u}^{(t-1)} - \mathbf{1} \otimes \overline{u}^{(t-1)}\right).$$

Therefore, it holds that

$$\begin{aligned}
&\left\|\mathbf{u}^{(t)}-\mathbf{u}^{(t-1)}\right\| \\
\leq&\frac{A_{t-1}}{A_t}\frac{a_{t-1}}{A_{t-1}}\left\|\mathbf{w}^{(t-1)}-\mathbf{w}^{(t-2)}\right\|+\frac{2A_{t-1}}{A_t}\left\|\tilde{\mathbf{u}}^{(t-1)}\right\|+\frac{a_t}{A_t}\left\|\mathbf{w}^{(t-1)}-\mathbf{u}^{(t-1)}\right\| \\
\leq&\frac{a_t}{A_t}\sqrt{n}G+\frac{2a_t}{A_t}C_p+\frac{a_t}{A_t}\sqrt{n}G \\
=&\frac{a_t}{A_t}\left(2\sqrt{n}G+2C_p\right)
\end{aligned} \tag{5.21}$$

where Lemma 5.1 and Assumption 5.3 are used to get the second inequality. By substituting (5.21) into (5.20), we obtain

$$\|\tilde{\mathbf{q}}^{(t)}\|\leq\beta\|\tilde{\mathbf{q}}^{(t-1)}\|+\frac{a_t}{A_t}L\left(2\sqrt{n}G+2C_p\right). \tag{5.22}$$

By initialization, we have $\tilde{\mathbf{q}}^{(0)}=0$ and therefore

$$\left\|\tilde{\mathbf{q}}^{(t_0)}\right\|\leq L\left(2\sqrt{n}G+2C_p\right)\sum_{\tau=1}^{t_0}\beta^{t_0-\tau}\frac{a_\tau}{A_\tau}\leq\frac{L\left(2\sqrt{n}G+2C_p\right)}{1-\beta},$$

implying that (5.18) is valid for $1\leq t<\lceil\frac{3}{1-\beta}\rceil$. Next, we prove that (5.18) also holds for $t\geq\lceil\frac{3}{1-\beta}\rceil$ by mathematical induction. Suppose that (5.18) holds true for some $t\geq\lceil\frac{3}{1-\beta}\rceil$. Using this hypothesis and (5.22), we obtain

$$\|\tilde{\mathbf{q}}^{(t+1)}\|\leq\frac{a_t}{A_t}\beta C_g+\frac{a_{t+1}}{A_{t+1}}L\left(2\sqrt{n}G+2C_p\right)\leq\frac{a_t}{A_t}C_g\left(\beta+\frac{1}{\lceil\frac{3}{1-\beta}\rceil}\right).$$

Finally, using the same argument with (5.15) and (5.16) in the proof of Lemma 5.1, we arrive at (5.18) as desired. □

Here, a variant of Lemma 4.4 is presented. For completeness, a proof is given.

Lemma 5.3. *Given a sequence of variables $\{\zeta^{(t)}\}_{t\geq0}$ and a positive sequence $\{a_t\}_{t\geq0}$, for $\{\nu^{(t)}\}_{t\geq0}$ generated by*

$$\nu^{(t)}=\operatorname*{argmin}_{x\in\mathcal{X}}\left\{\sum_{\tau=1}^{t}a_\tau\left\langle\zeta^{(\tau)},x\right\rangle+d(x)\right\},$$

where $\nu^{(0)} = x^{(0)}$ *in* (5.2)*, it holds*

$$\sum_{\tau=1}^{t} a_\tau \left\langle \zeta^{(\tau)}, \nu^{(\tau)} - x^* \right\rangle \leq d(x^*) - \sum_{\tau=1}^{t} \frac{1}{2}\|\nu^{(\tau)} - \nu^{(\tau-1)}\|^2. \tag{5.23}$$

Proof of Lemma 5.3. Define

$$m_t(x) = \sum_{\tau=1}^{t} a_\tau \left\langle \zeta^{(\tau)}, x \right\rangle + d(x)$$

where $m_0(x) = d(x)$. Since

$$\nu^{(\tau-1)} = \operatorname*{argmin}_{x\in\mathcal{X}} m_{\tau-1}(x)$$

and $m_{\tau-1}(x)$ is strongly convex with modulus 1, we have

$$m_{\tau-1}(x) - m_{\tau-1}(\nu^{(\tau-1)}) \geq \frac{1}{2}\|x - \nu^{(\tau-1)}\|^2, \forall x \in \mathcal{X}.$$

Upon taking $x = \nu^{(\tau)}$ in the above inequality and using

$$m_\tau(x) = m_{\tau-1}(x) + a_\tau \left\langle \zeta^{(\tau)}, x \right\rangle,$$

we obtain

$$\begin{aligned} 0 \leq & m_{\tau-1}(\nu^{(\tau)}) - m_{\tau-1}(\nu^{(\tau-1)}) - \frac{1}{2}\|\nu^{(\tau)} - \nu^{(\tau-1)}\|^2 \\ = & m_\tau(\nu^{(\tau)}) - a_\tau \left\langle \zeta^{(\tau)}, \nu^{(\tau)} \right\rangle - m_{\tau-1}(\nu^{(\tau-1)}) - \frac{1}{2}\|\nu^{(\tau)} - \nu^{(\tau-1)}\|^2, \end{aligned}$$

which is equivalent to

$$a_\tau \left\langle \zeta^{(\tau)}, \nu^{(\tau)} \right\rangle \leq m_\tau(\nu^{(\tau)}) - m_{\tau-1}(\nu^{(\tau-1)}) - \frac{1}{2}\|\nu^{(\tau)} - \nu^{(\tau-1)}\|^2.$$

Iterating the above equation from $\tau = 1$ to $\tau = t$ yields

$$\begin{aligned} \sum_{\tau=1}^{t} a_\tau \left\langle \zeta^{(\tau)}, \nu^{(\tau)} \right\rangle \leq & m_t(\nu^{(t)}) - m_0(\nu^{(0)}) - \sum_{\tau=1}^{t} \frac{1}{2}\|\nu^{(\tau)} - \nu^{(\tau-1)}\|^2 \\ = & m_t(\nu^{(t)}) - \sum_{\tau=1}^{t} \frac{1}{2}\|\nu^{(\tau)} - \nu^{(\tau-1)}\|^2 \end{aligned} \tag{5.24}$$

We turn to consider

$$\begin{aligned}\sum_{\tau=1}^{t} a_\tau \left\langle \zeta^{(\tau)}, -x^* \right\rangle &\leq \max_{x\in\mathcal{X}} \left\{ \sum_{\tau=1}^{t} a_\tau \left\langle \zeta^{(\tau)}, -x \right\rangle - d(x) \right\} + d(x^*) \\ &= -\min_{x\in\mathcal{X}} \left\{ \sum_{\tau=1}^{t} a_\tau \left\langle \zeta^{(\tau)}, x \right\rangle + d(x) \right\} + d(x^*) \\ &= -m_t(\nu^{(t)}) + d(x^*),\end{aligned}$$

which together with (5.24) leads to the inequality in (5.23), thereby concluding the proof. □

5.4.2 Proof of Theorem 5.1

Proof of Theorem 5.1. Using $A_{\tau-1} = A_\tau - a_\tau$, we have

$$\begin{aligned}&A_t\Big(f(\overline{v}^{(t)}) - f(x^*)\Big) \\ =&\sum_{\tau=1}^{t} \left(A_\tau f(\overline{v}^{(\tau)}) - A_{\tau-1} f(\overline{v}^{(\tau-1)})\right) - \sum_{\tau=1}^{t} a_\tau f(x^*) \\ =&\sum_{\tau=1}^{t} \Big(A_\tau \left(f(\overline{v}^{(\tau)}) - f(\overline{u}^{(\tau)})\right) + a_\tau \left(f(\overline{u}^{(\tau)}) - f(x^*)\right) + A_{\tau-1} \left(f(\overline{u}^{(\tau)}) - f(\overline{v}^{(\tau-1)})\right) \Big)\end{aligned}$$

Upon using the convexity of f, we obtain

$$\begin{aligned}A_t \left(f(\overline{v}^{(t)}) - f(x^*)\right) \leq \sum_{\tau=1}^{t} \Big(&A_\tau \left(f(\overline{v}^{(\tau)}) - f(\overline{u}^{(\tau)})\right) + a_\tau \left\langle \nabla f(\overline{u}^{(\tau)}), \overline{u}^{(\tau)} - x^* \right\rangle \\ &+ A_{\tau-1} \left\langle \nabla f(\overline{u}^{(\tau)}), \overline{u}^{(\tau)} - \overline{v}^{(\tau-1)} \right\rangle \Big)\end{aligned}$$

By (5.13b), we obtain

$$\begin{aligned}
&A_t\Big(f(\overline{v}^{(t)}) - f(x^*)\Big) \\
\leq &\sum_{\tau=1}^{t} A_\tau\Big(f(\overline{v}^{(\tau)}) - f(\overline{u}^{(\tau)}) + \left\langle \nabla f(\overline{u}^{(\tau)}), \overline{u}^{(\tau)} - \overline{v}^{(\tau)} \right\rangle\Big) \\
&+ \sum_{\tau=1}^{t} a_\tau \left\langle \nabla f(\overline{u}^{(\tau)}), \overline{w}^{(\tau)} - x^* \right\rangle \\
\leq &\sum_{\tau=1}^{t} \frac{A_\tau L}{2} \underbrace{\|\overline{u}^{(\tau)} - \overline{v}^{(\tau)}\|^2}_{(I)} + \frac{1}{n}\sum_{i=1}^{n} \underbrace{\sum_{\tau=1}^{t} a_\tau \left\langle q_i^{(\tau)}, w_i^{(\tau)} - x^* \right\rangle}_{(II)} \\
&+ \frac{1}{n}\sum_{\tau=1}^{t} \underbrace{\sum_{i=1}^{n} a_\tau \left\langle \nabla f(\overline{u}^{(\tau)}) - q_i^{(\tau)}, w_i^{(\tau)} - x^* \right\rangle}_{(III)}
\end{aligned} \tag{5.25}$$

where the last inequality is due to the smoothness of f. To bound (I), we consider

$$\begin{aligned}
&\|\overline{u}^{(\tau)} - \overline{v}^{(\tau)}\|^2 \\
= &\frac{1}{n}\sum_{i=1}^{n} \left\| \overline{u}^{(\tau)} - u_i^{(\tau)} + u_i^{(\tau)} - v_i^{(\tau)} + v_i^{(\tau)} - \overline{v}^{(\tau)} \right\|^2 \\
\leq &\frac{1}{n}\sum_{i=1}^{n} \left(3\left\| \overline{u}^{(\tau)} - u_i^{(\tau)} \right\|^2 + 3\left\| u_i^{(\tau)} - v_i^{(\tau)} \right\|^2 + 3\left\| v_i^{(\tau)} - \overline{v}^{(\tau)} \right\|^2 \right) \\
\leq &\left(\frac{a_\tau}{A_\tau}\right)^2 \frac{6C_p^2 + 3\left\|\mathbf{w}^{(\tau)} - \mathbf{w}^{(\tau-1)}\right\|^2}{n}
\end{aligned} \tag{5.26}$$

where the first inequality follows from

$$\|x + y + z\|^2 \leq 3\|x\|^2 + 3\|y\|^2 + 3\|z\|^2,$$

and the last inequality is due to Lemma 5.1 and (5.13). For (II), by letting $\zeta^{(\tau)} = q_i^{(\tau)}$ and $\nu^{(\tau)} = w_i^{(\tau)}$ in Lemma 5.3, we have

$$\sum_{\tau=1}^{t} a_\tau \left\langle q_i^{(\tau)}, w_i^{(\tau)} - x^* \right\rangle \leq d(x^*) - \sum_{\tau=1}^{t} \frac{1}{2}\|w_i^{(\tau)} - w_i^{(\tau-1)}\|^2. \tag{5.27}$$

To bound (III), we use (5.17) to get

$$\begin{aligned}
&a_\tau \left\langle \nabla f(\overline{u}^{(\tau)}) - q_i^{(\tau)}, w_i^{(\tau)} - x^* \right\rangle \\
\leq & G a_\tau \left\| \nabla f(\overline{u}^{(\tau)}) - \overline{\hat{g}}^{(\tau)} + \overline{q}^{(\tau)} - q_i^{(\tau)} \right\| \\
\leq & G a_\tau \left(\left\| \nabla f(\overline{u}^{(\tau)}) - \overline{\hat{g}}^{(\tau)} \right\| + \left\| \overline{q}^{(\tau)} - q_i^{(\tau)} \right\| \right).
\end{aligned}$$

Upon using Lemma 5.1, we obtain

$$\begin{aligned}
&\left\| \nabla f(\overline{u}^{(\tau)}) - \overline{\hat{g}}^{(\tau)} \right\| \leq \frac{1}{n} \sum_{i=1}^{n} \left\| \nabla f_i(\overline{u}^{(\tau)}) - \nabla f_i(u_i^{(\tau)}) \right\| \\
\leq & \frac{L}{n} \sum_{i=1}^{n} \left\| \overline{u}^{(\tau)} - u_i^{(\tau)} \right\| \leq L \sqrt{\frac{\|\tilde{\mathbf{u}}\|^2}{n}} \\
\leq & \left(\frac{a_\tau}{A_\tau} \right) \frac{LC_p}{\sqrt{n}}.
\end{aligned}$$

Recall Lemma 5.2 that $\left\| \overline{q}^{(\tau)} - q_i^{(\tau)} \right\| \leq C_g a_\tau / (\sqrt{n} A_\tau)$. Therefore

$$\sum_{i=1}^{n} a_\tau \left\langle \nabla f(\overline{u}^{(\tau)}) - q_i^{(\tau)}, w_i^{(\tau)} - x^* \right\rangle \leq \left(\frac{a_\tau^2}{A_\tau} \right) \sqrt{n} G (LC_p + C_g). \tag{5.28}$$

Finally, by collectively substituting (5.26), (5.27), and (5.28) into (5.25), we get

$$\begin{aligned}
&A_t \left(f(\overline{v}^{(t)}) - f(x^*) \right) \\
\leq & \left(\frac{G(LC_p + C_g)}{\sqrt{n}} + \frac{3LC_p^2}{n} \right) \sum_{\tau=1}^{t} \frac{a_\tau^2}{A_\tau} + d(x^*) + \frac{1}{2n} \sum_{\tau=1}^{t} \left(\frac{3La_\tau^2}{A_\tau} - 1 \right) \left\| \mathbf{w}^{(\tau)} - \mathbf{w}^{(\tau-1)} \right\|^2.
\end{aligned}$$

Based on the condition in (5.8) and the fact that $a_\tau^2 / A_\tau \leq 2a$, we obtain (5.9) as desired.

The inequality in (5.10) directly follows from Lemma 5.1. $\square$

5.5 Experiments

In this section, we verify the proposed methods by applying them to solve the following constrained LASSO problems:

$$\min_{x\in\mathbb{R}^m}\left\{f(x)=\frac{1}{2n}\sum_{i=1}^{n}\|M_ix-c_i\|^2\right\},\quad \text{s.t. } \|x\|_1\leq R$$

where $M_i\in\mathbb{R}^{p_i\times m}$, $c_i\in\mathbb{R}^{p_i}$, and R is a constant parameter that defines the constraint. In the simulation, each agent i has access to a local data tuple (y_i, A_i) and R. Two different problem instances characterized by both real and synthetic datasets are considered.

5.5.1 Case I: Real Dataset

In this setting, we use *sparco7* [93, 96] to define the LASSO problem, and consider a cycle graph and a complete graph of $n = 50$ nodes. The corresponding weight matrix P is determined by following the Metropolis-Hastings rule [105]. Each local measurement matrix $M_i\in\mathbb{R}^{12\times 2560}$, and the local corrupted measurement $c_i\in\mathbb{R}^{12}$. The constraint parameter is set as $R = 1.1\cdot\|x_g\|_1$, where x_g with $\|x_g\|_0 = 20$ denotes the unknown variable to be recovered via solving LASSO. In this case, the simulation experiments were performed using MATLAB R2020b.

For comparison, the PG-EXTRA method in [84] and the APM method in [52] are simulated. For their algorithmic parameters, the step size for PG-EXTRA is set as 10^{-4}, and the parameters for APM are set as $L = 250$ and $\beta_0 = L/\sqrt{1-\lambda_2(P)}$. For DDA in Chapter 4 and ADDA in this chapter, we use $a = 5\cdot10^{-4}$ and $a_t = (t+1)\cdot10^{-4}$, respectively, and $\|x\|^2/2$ as the prox-function. The projection onto an l_1 ball is carried out via the algorithm in [13]. All the methods are initialized with $x_i^{(0)} = 0, \forall i\in\mathcal{V}$.

The performance of four algorithms is displayed in Figure 5.1. Particularly, the performance is evaluated in terms of the objective error $f(\frac{1}{n}\sum_{i=1}^{n}x_i^{(t)})-f(x^*)$, where x^* is identified using CVX [21]. It demonstrates that the DDA method outperforms other methods when the graph is a cycle. As the graph becomes denser, i.e, complete graph, the convergence of all algorithms becomes faster. Among them, the ADDA method demonstrates the most significant improvement. This is in line with Theorem 5.1, where the network connectivity impacts the convergence error in $\mathcal{O}(1/t)$ as opposed to $\mathcal{O}(1/t^2)$.

5.5.2 Case II: Synthetic Dataset

For the synthetic dataset, the parameters are set as $n = 8$, $m = 30000$, $p_i = 2000, \forall i \in \mathcal{V}$, and the data is generated in the following way. First, each local measurement matrix M_i is randomly generated where each entry follows the normal distribution $\mathcal{N}(0, 1)$. Next, each entry of the sparse vector x_g to be recovered via LASSO is randomly generated from the normal distribution $\mathcal{N}(0, 1)$ with $\|x_g\|_0 = 1500$. Then the corrupted measurement c_i is produced based on

$$c_i = M_i x_g + b_i,$$

where b_i represents the Gaussian noise with zero mean and variance 0.01. The constraint parameter is set as $R = 1.1 \cdot \|x_g\|_1$. For this setting, we employed the message passing interface (MPI) in Python 3.7.3 to simulate a network of 8 nodes, where each node i is connected to a subset of nodes $\{1+i \bmod 8, 1+(i+3) \bmod 8, 1+(i+6) \bmod 8\}$. For comparison, the DDA in Chapter 4 and ADDA in this chapter are compared to their centralized counterparts. The parameters for dual averaging and accelerated dual averaging are set as $a = 1/(3 \cdot 10^5)$ and $a_t = a(t+1)$, respectively. Similarly, the function $\|x\|^2/2$ is used as a prox-function, and the algorithms are initialized with $x_i^{(0)} = 0, \forall i \in \mathcal{V}$.

The performance of the developed algorithm and its centralized counterpart, i.e., is illustrated in Figure 5.2. In particular, the performance is evaluated in terms of objective function value versus iteration number and computing time. It demonstrates that ADDA outperforms the centralized method in the sense that ADDA consumes less computing time to reach the same degree of accuracy than the centralized method.

5.6 Conclusion

In this chapter, we have designed an accelerated DDA algorithm for solving decentralized constrained optimization problems. In this algorithm, each agent retains the conventional first-order dynamic average consensus method to estimate the average of local gradients. Alternatively, the extrapolation technique together with the average consensus protocol is used to achieve acceleration over a decentralized network.

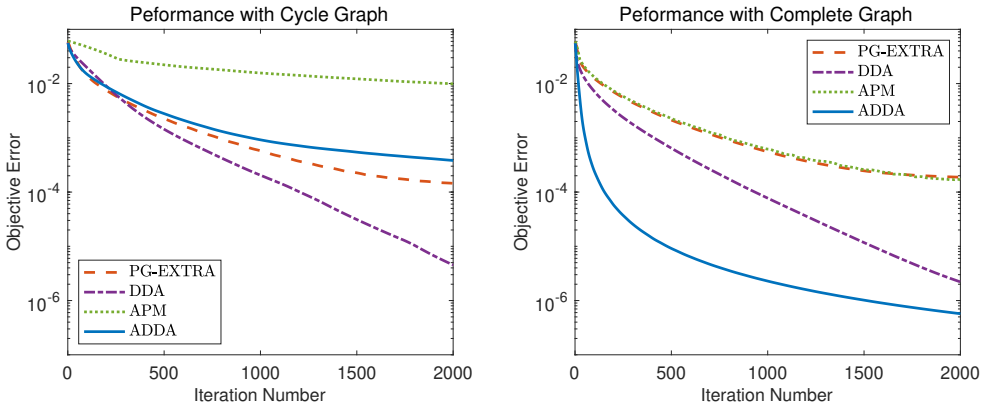


Figure 5.1: Comparison of objective error in Case I.

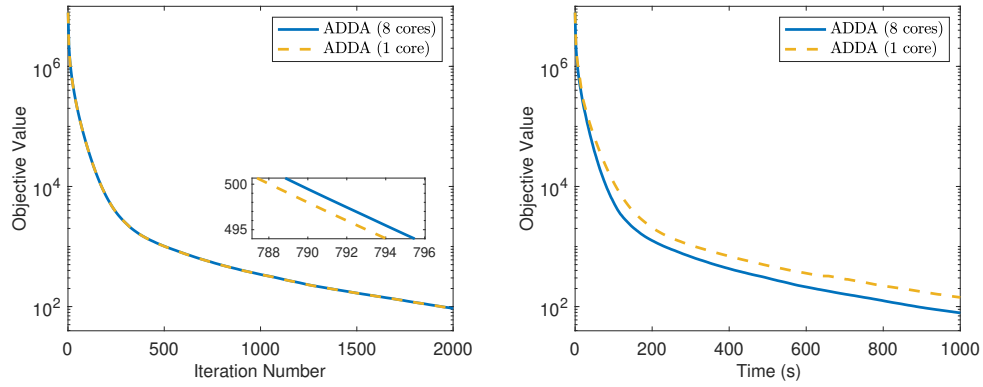


Figure 5.2: Comparison of objective value in Case II.

Chapter 6

Communication-Efficient Decentralized Primal-Dual Algorithms

6.1 Introduction

In previous chapters, the decentralized optimization algorithms require all the agent to synchronously communicate with their neighbors at every iteration. For cyber-physical systems operated in a communication-limited environment, these algorithms may become undesirable as they may consume more communication resources than they necessarily need. Event-triggered scheduling of network transmissions is a promising solution to this problem, and has been exploited to design communication-efficient controllers lately [2, 88, 98]. The idea is to generate network transmission only when the information conveyed by the message is deemed innovative to the system, and whether or not it is innovative is determined via an event-triggered function that takes the deviation between the actual system state and the state just broadcast as an argument.

Inspired by this attractive feature, event-triggered communication has been incorporated into decentralized optimization algorithms [10, 23, 30, 41, 51, 54]. Recently, the work in [54] presented an event-triggered decentralized ADMM that only requires each agent to broadcast the local primal variable to its neighbors, and proved the convergence of the algorithm when the objective function is convex. Convergence rates are analyzed for special strongly convex and smooth objectives. In [54], each

agent at every generic time instant is required to exactly solve a subproblem, which may be not practical in most cases. Considering this, two questions naturally arise: i) For general convex functions, is it possible to devise an event-triggered decentralized optimization algorithm that enjoys the same order of convergence with periodic algorithms even in the presence of variable errors due to event-triggered communication? ii) If the objective functions exhibit some desired properties, e.g., smooth or/and strongly convex, is it possible to simplify the subproblem-solving process to simple algebraic operations without sacrificing the rate of convergence?

We give affirmative answers to these questions in this chapter. First, the primal-dual methodology is used to tackle the decentralized optimization problem. More specifically, the linearized augmented Lagrangian method (LALM) in [114] with a specific pre-conditioning strategy is used to design a periodic decentralized algorithm. Then, each agent employs an event-triggered broadcasting strategy to communicate with its neighbors to avoid unnecessary network utilization. Compared with the state-of-the-art, the developed event-triggered method features the following. i) It ensures exact minimization with individual constant step sizes to improve the speed that is usually determined by the slowest agent in existing methods. This is made possible by adjusting the diagonal entries in the weight matrix that approximates the curvature of the objective. ii) It provides simplified (algebraic) local iteration rules for composite (smooth) problems to ease computational burden. iii) Convergence rates for different types of objective functions are proved for the first time, that is, an $\mathcal{O}(1/t)$ rate of convergence for non-smooth objective functions and linear convergence when the objective functions are strongly convex and smooths. To achieve this, a significantly different analysis from LALM is carried out since triggering schedulers inject errors into each iteration. In particular, we establish a new upper bound for the effect of errors on the primal-dual residual. Based on this, the same convergence rate $\mathcal{O}(1/t)$ with the standard primal-dual algorithm can be guaranteed for non-smooth convex problems.

6.2 Problem Setup and Preliminaries

6.2.1 Basic Setup

Consider an MAS consisting of n agents connected via a bidirectional network. They aim to solving the following cost-coupled optimization problem

$$\min_{x\in\mathbb{R}^m}\left\{\sum_{i=1}^{n} F_i(x) := f_i(x) + h_i(x)\right\} \tag{6.1}$$

where f_i and h_i represent the local smooth objective function and non-smooth regularization term of agent i, respectively.

The communication network among agents is characterized by a fixed undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. We make the following assumption for the communication graph.

Assumption 6.1. *$\mathcal{G}$ is fixed and connected.*

Our goal is to design an event-triggered decentralized first-order algorithm with individual constant step sizes for the cost-coupled optimization problem in (6.1) to save communication resources. In this framework, simplified local implementations will be used for composite and smooth problems to ease computational load. Furthermore, we will rigorously analyze the effect of triggering behavior on the iterates, and prove the rates of convergence for the algorithm under different settings.

6.2.2 Primal-Dual Formulation

Define $\mathbf{x} = [x_1^{\mathrm{T}}, \cdots, x_n^{\mathrm{T}}]^{\mathrm{T}}$, $\mathbf{F}(\mathbf{x}) = \sum_{i=1}^n F_i(x_i)$, $\mathbf{f}(\mathbf{x}) = \sum_{i=1}^n f_i(x_i)$, and $\mathbf{h}(\mathbf{x}) = \sum_{i=1}^n h_i(x_i)$. Following [80], the problem in (6.1) can be equivalently written as the following linearly constrained optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^{mn}} \mathbf{F}(\mathbf{x}) \quad \text{s.t.} \quad \big(\sqrt{\mathcal{L}}\otimes I\big)\mathbf{x} = 0 \tag{6.2}$$

The augmented Lagrangian for (6.2) is written as

$$\mathbf{F}(\mathbf{x}) + \Big\langle \mathbf{y}, \big(\sqrt{\mathcal{L}}\otimes I\big)\mathbf{x}\Big\rangle + \frac{\beta}{2}\big\|\mathbf{x}\big\|^2_{\mathcal{L}\otimes I}$$

where $\mathbf{y} = [y_1^{\mathrm{T}}, \cdots, y_n^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{mn}$ denotes the dual variable and $\beta > 0$ a designable parameter. The KKT conditions can be identified as

$$0 \in \partial \mathbf{F}(\mathbf{x}^*) + \big(\sqrt{\mathcal{L}} \otimes I\big)\mathbf{y}^* \tag{6.3a}$$
$$0 = \big(\sqrt{\mathcal{L}} \otimes I\big)\mathbf{x}^* \tag{6.3b}$$

where $(\mathbf{x}^*, \mathbf{y}^*)$ is an optimal primal-dual pair and $\partial \mathbf{F}(\mathbf{x}^*)$ the set of all subgradients of $\mathbf{F}$ over $\mathbf{x}^*$.

6.3 Algorithm and Convergence Results

6.3.1 Algorithm Development

Based on the above primal-dual formulation, we recruit the LALM [114] to solve the decentralized composite optimization problem in (6.1):

$$\begin{aligned} \mathbf{x}^{(t)} &= \operatorname*{argmin}_{\mathbf{x}} \left\{ \mathbf{R}(\mathbf{x}; \mathbf{x}^{(t-1)}) + \mathbf{h}(\mathbf{x}) + \left\langle \left(\sqrt{\mathcal{L}} \otimes I\right) \mathbf{y}^{(t-1)}, \mathbf{x} \right\rangle + \frac{\beta}{2} \big\|\mathbf{x}\big\|^2_{\mathcal{L}\otimes I} \right\} \\ \mathbf{y}^{(t)} &= \mathbf{y}^{(t-1)} + \beta(\sqrt{\mathcal{L}} \otimes I)\mathbf{x}^{(t)}, \end{aligned} \tag{6.4}$$

where

$$\mathbf{R}(\mathbf{x}; \mathbf{x}^{(t-1)}) = \mathbf{f}(\mathbf{x}^{(t-1)}) + \left\langle \nabla \mathbf{f}(\mathbf{x}^{(t-1)}), \mathbf{x} \right\rangle + \frac{1}{2} \big\|\mathbf{x} - \mathbf{x}^{(t-1)}\big\|^2_{(H-\beta\mathcal{L})\otimes I}$$

is a quadratic approximation of $\mathbf{f}$, and $H = \operatorname{diag}\{\eta_i\}_{i=1}^n \succ 0$ is a diagonal matrix. Further by letting $\mathbf{z} = (\sqrt{\mathcal{L}} \otimes I)\mathbf{y}$ [37, 91, 114], the iteration rule becomes

$$\begin{aligned} \mathbf{x}^{(t)} =& \operatorname*{argmin}_{\mathbf{x}} \left\{ \mathbf{R}(\mathbf{x}; \mathbf{x}^{(t-1)}) + \mathbf{h}(\mathbf{x}) + \left\langle \mathbf{z}^{(t-1)}, \mathbf{x} \right\rangle + \frac{\beta}{2} \big\|\mathbf{x}\big\|^2_{\mathcal{L}\otimes I} \right\} \\ \mathbf{z}^{(t)} =& \mathbf{z}^{(t-1)} + \beta(\mathcal{L} \otimes I)\mathbf{x}^{(t)}. \end{aligned}$$

and therefore

$$\begin{aligned} \mathbf{x}^{(t)} =& \operatorname*{argmin}_{\mathbf{x}} \left\{ \mathbf{R}'(\mathbf{x}; \mathbf{x}^{(t-1)}) + \mathbf{h}(\mathbf{x}) + \left\langle \mathbf{z}^{(t-1)} + \beta(\mathcal{L} \otimes I)\mathbf{x}^{(t-1)}, \mathbf{x} \right\rangle \right\} \\ \mathbf{z}^{(t)} =& \mathbf{z}^{(t-1)} + \beta(\mathcal{L} \otimes I)\mathbf{x}^{(t)}. \end{aligned}$$

where

$$\mathbf{R}'(\mathbf{x}; \mathbf{x}^{(t-1)}) = \mathbf{f}(\mathbf{x}^{(t-1)}) + \left\langle \nabla \mathbf{f}(\mathbf{x}^{(t-1)}), \mathbf{x} \right\rangle + \frac{1}{2} \big\|\mathbf{x} - \mathbf{x}^{(t-1)}\big\|^2_{H\otimes I}.$$

Element-wisely,

$$\begin{aligned} x_i^{(t)} =& \operatorname*{argmin}_x \left\{ R_i'(x; x_i^{(t-1)}) + h_i(x) + \left\langle z_i^{(t-1)} + \beta \sum_{j\in\mathcal{N}_i} \left(x_i^{(t-1)} - x_j^{(t-1)} \right), x \right\rangle \right\} \\ z_i^{(t)} =& z_i^{(t-1)} + \beta \sum_{j\in\mathcal{N}_i} \left(x_i^{(t)} - x_j^{(t)} \right) \end{aligned} \tag{6.5}$$

where

$$R_i'(x; x_i^{(t-1)}) = f_i(x_i^{(t-1)}) + \left\langle \nabla f_i(x_i^{(t-1)}), x \right\rangle + \frac{\eta_i}{2} \left\| x - x_i^{(t-1)} \right\|^2.$$

Denote the set of generic time instants by $\kappa = \{t | t \in \mathbb{N}\}$. It serves a global clock that synchronizes all the agents. At each time t, we define the true and broadcast primal variables $x_i^{(t)}$ and $\tilde{x}_i^{(t)}$ for agent i. Note that the variable $\tilde{x}_i^{(t)}$ is the same across all $j \in \mathcal{N}_i$. Let $\kappa_i = \{t_i^{[l]} | l \in \mathbb{N}\} \subseteq \kappa$ be the set of triggering time instants for agent i, where

$$t_i^{[l+1]} = \min \left\{ t \in \kappa | t > t_i^{[l]}, \left\| x_i^{(t)} - \tilde{x}_i^{(t-1)} \right\| > E_i^{(t)} \right\}, \tag{6.6}$$

$t \geq 1$, and $E_i^{(t)} \geq 0$ represents the triggering threshold. The broadcast variable $\tilde{x}_i^{(t)}$ is defined as

$$\tilde{x}_i^{(t)} = \begin{cases} x_i^{(t)}, & t \in \kappa_i \\ \tilde{x}_i^{(t-1)}, & \text{otherwise} \end{cases} \tag{6.7}$$

It can be verified from the definition that the deviation between $x_i^{(t)}$ and $\tilde{x}_i^{(t)}$ is always bounded from above by $E_i^{(t)}$, that is,

$$\left\| x_i^{(t)} - \tilde{x}_i^{(t)} \right\| \leq E_i^{(t)}.$$

For the triggering threshold, we make the following assumption.

Assumption 6.2. *Let $E^{(t)} = \max_{i\in\mathbb{N}_{[1,n]}} E_i^{(t)}$ for all $t \in \mathbb{N}$. $E^{(t)}$ is non-increasing and summable, i.e., $\sum_{t=0}^{\infty} E^{(t)} < \infty$.*

Based on the above communication pattern, we modify the iteration rule in (6.5) to

$$x_i^{(t)} = \operatorname*{argmin}_x \left\{ R_i'(x; x_i^{(t-1)}) + h_i(x) + \left\langle z_i^{(t-1)} + \beta \sum_{j\in\mathcal{N}_i} \left(\tilde{x}_i^{(t-1)} - \tilde{x}_j^{(t-1)} \right), x \right\rangle \right\} \tag{6.8a}$$

$$z_i^{(t)} = z_i^{(t-1)} + \beta \sum_{j\in\mathcal{N}_i} \left(\tilde{x}_i^{(t)} - \tilde{x}_j^{(t)} \right). \tag{6.8b}$$

The overall algorithm is summarized in Algorithm 5.

Algorithm 5 Event-triggered Decentralized Optimization

Input: $\eta_i > 0$ for all $i = 1, \ldots, n$
Initialize: $z_i^{(0)} = 0$ for all $i = 1, \ldots, n$, each agent i broadcasts $x_i^{(0)}$ to its neighbors
for $t = 1, 2, \cdots$ **do**
  In parallel (task for agent i, $i = 1, \ldots, n$)
  update $x_i^{(t)}$ by (6.8a)
  test the event condition in (6.6)
  if triggered **then**
    broadcast $x_i^{(t)}$ to its neighbors $j \in \mathcal{N}_i$
  end if
  construct $\tilde{x}_j^{(t)}, j \in \mathcal{N}_i$ according to (6.7)
  update $z_i^{(t)}$ by (6.8b)
end for

6.3.2 Convergence Results

Assumption 6.3. *For $i = 1, \ldots, n$, f_i is convex and l_{f_i}-smooth, and g_i is a proper closed convex function.*

By Assumption 6.3 and the Cauchy-Schwartz inequality, we have that the gradient of $\mathbf{f}(\mathbf{x})$ satisfies

$$\langle \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \leq \|\mathbf{x} - \mathbf{y}\|^2_{L_f \otimes I}, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^{mn},$$

where $L_f = \operatorname{diag}\{l_{f_i}\}_{i=1}^n$. Then, we examine the convergence rate for Algorithm 5 with Assumptions 6.1, 6.2, and 6.3 satisfied.

Theorem 6.1. *If Assumptions 6.1, 6.2, and 6.3 hold and*

$$P - L_f \succ 0,$$

then

$$\left\| (\sqrt{\mathcal{L}} \otimes I) \hat{\mathbf{x}}^{(t)} \right\| \leq \frac{\left(\left\| \mathbf{x}^{(0)} - \mathbf{x}^* \right\|_{P \otimes I} + \rho \left\| \mathcal{L} (\beta \mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})^{-1} \right\| + \sqrt{2b} A_t \right)^2}{2t \left(\rho - \|\mathbf{y}^*\| \right)}, \tag{6.9}$$

and

$$\begin{aligned} &-\frac{\|\mathbf{y}^*\|\left(\left\|\mathbf{x}^{(0)}-\mathbf{x}^*\right\|_{P\otimes I}+\rho\left\|\mathcal{L}(\beta(\mathcal{L}\otimes I)+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})^{-1}\right\|+\sqrt{2b}A_t\right)^2}{2t\left(\rho-\|\mathbf{y}^*\|\right)} \\ \leq&\mathbf{F}(\hat{\mathbf{x}}^{(t)})-\mathbf{F}(\mathbf{x}^*)\leq\frac{\left(\left\|\mathbf{x}^{(0)}-\mathbf{x}^*\right\|_{P\otimes I}+\rho\left\|\mathcal{L}(\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})^{-1}\right\|+\sqrt{2b}A_t\right)^2}{2t}, \end{aligned} \tag{6.10}$$

where $P=H-\beta\mathcal{L}$, $\hat{\mathbf{x}}^{(t)}=t^{-1}\sum_{\tau=1}^{t}\mathbf{x}^{(\tau)}$, $\left\|\mathbf{y}^*\right\|<\rho<\infty$, $A_t=\frac{2a\sqrt{n}}{b}\sum_{\tau=1}^{t}E^{(\tau-1)}$, $a=\max\left\{2\beta\overline{\lambda}(\mathcal{L}),1\right\}$, *and* $b=\min\left\{\underline{\lambda}(L_f),1/\overline{\lambda}(\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\right\}$.

Proof. Please refer to Appendix A. □

Remark 6.1. *In this chapter, the triggering scheduler imposes conditions on the information that is outdated but deemed effective, that is, the error between it and the real-time information should be decreasing fast enough (summable). A rigorous analysis that heavily exploits this property is then carried out. In particular, the effect of triggering behavior on the primal-dual residual is proved bounded when the triggering threshold is summable over time.*

The results in Theorem 6.1 are explained as follows.

i) *Comparison of sufficient conditions with [54]:* Theorem 6.1 states that both the consensus error $\left\|(\sqrt{\mathcal{L}}\otimes I)\hat{\mathbf{x}}^{(t)}\right\|$ and the objective error $\mathbf{F}(\hat{\mathbf{x}}^{(t)})-\mathbf{F}(\mathbf{x}^*)$ converge to zero at an ergodic convergence rate of $\mathcal{O}(1/t)$ if some reasonable assumptions hold true. Note that the result remains valid for smooth objective functions, i.e., $\mathbf{h}(\mathbf{x})=0$. For completely non-smooth objective functions, i.e., $\mathbf{f}(\mathbf{x})=0$, the condition for step size to ensure the same convergence rate is relaxed to

$$H-\beta\mathcal{L}\succ 0.$$

Note that the diagonal matrix H allows the use of different step sizes for agents, depending on the local Lipschitz modulus. By setting $H=\eta I$, the sufficient condition reduces to

$$\eta>\beta\overline{\lambda}(\mathcal{L}).$$

When the free parameter β approaches zero, this condition becomes equivalent to that in [54] for convergence.

ii) *Choices of β, H and $E^{(t)}$:* Theorem 6.1 indicates that, given a graph Laplacian, a larger β necessitates an H with larger diagonal entries that is used in the quadratic approximation. If H over-approximates the curvature of $\mathbf{f}$ in (6.4), the convergence of primal variables $\mathbf{x}^{(t)}$ will be slow. However, if η_i is too small and under-approximates the curvature, the primal iterate may oscillate quickly. In practice, the designable matrix H and parameter β should be carefully tuned to achieve a reasonable convergence rate. An appropriate choice would be to set $\beta = 1/(\overline{\lambda}(\mathcal{L}) + 1)$ and $H = I + L_f$. Theorem 6.1 also suggests that the threshold sequence $E^{(t)}$ will affect convergence. In particular, a more slowly decreasing $E^{(t)}$ satisfying Assumption 6.2 will result in a larger A_t and therefore a larger base in convergence constants. For composite optimization, one can set a threshold sequence that converges slightly faster than the guaranteed rate $\mathcal{O}(1/t)$, e.g., $E_i^{(0)}/t^2$. In addition, a base constant $E_i^{(0)}$ that is sufficiently smaller than the magnitude of $z_i^{(0)} + \nabla f_i(x_i^{(0)}) + \beta \sum_{j\in\mathcal{N}_i}(x_i^{(0)} - x_j^{(0)})$ is suggested to prevent oscillation in the beginning.

Next, we consider strongly convex and smooth objective functions, for which stronger convergence results can be stated. Formally, the following assumption is made for the objective functions.

Assumption 6.4. *For $i = 1, \ldots, n$, $h_i(\theta) \equiv 0$ and f_i is μ_{f_i}-strongly convex.*

As a direct consequence, the gradient of $\mathbf{f}$ satisfies

$$\langle \nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \|\mathbf{x} - \mathbf{y}\|^2_{M\otimes I}, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^{mn},$$

where $M = \mathrm{diag}\{\mu_{f_i}\}_{i=1}^n$.

Theorem 6.2. *If Assumptions 6.1-6.4 hold and*

$$P - L_f^2/k_1 \succ 0 \tag{6.11}$$

for some $0 < k_1 < 2\underline{\lambda}(M)$, then there exists some positive σ such that

$$\begin{aligned} &\frac{1}{2}\left\|\mathbf{x}^{(t)} - \mathbf{x}^*\right\|^2_{(P+k_4Q)\otimes I} + \frac{1}{2}\left\|\mathbf{z}^{(t)} - \mathbf{z}^*\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} + nC(E^{(t)})^2 \\ &\geq \frac{\sigma+1}{2}\left(\left\|\mathbf{x}^{(t+1)} - \mathbf{x}^*\right\|^2_{(P+k_4Q)\otimes I} + \left\|\mathbf{z}^{(t+1)} - \mathbf{z}^*\right\|^2_{\left(\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}\right)^{-1}}\right) \end{aligned} \tag{6.12}$$

where $P = H - \beta\mathcal{L}, Q = 2M - k_1 I, 0 < k_2$, $2 < k_3$, $0 < k_4 < 1$, $0 < k_5$, *and*

$$C = \left(\frac{2\big(k_3 + 1/k_2 - 1\big)(\sigma + k_5)\overline{\lambda}(\beta^2\mathcal{L}^2)}{(1 - 2/k_3)\underline{\lambda}(\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})} + \frac{\overline{\lambda}(\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})}{2k_5} + \frac{2\overline{\lambda}(\beta^2\mathcal{L}^2)}{k_5\underline{\lambda}(P + k_4 Q)} \right).$$

Proof. The proof is postponed to Appendix B. □

Some remarks on the results in Theorem 6.2 are in order.

i) *Linear convergence:* It is stated in Theorem 6.2 that if the objective function is further assumed to be strongly convex and the step size satisfies a relatively stricter condition then a much faster convergence rate can be obtained. In particular, if $E^{(t)}$ linearly converges then we obtain a linear convergence rate for the primal-dual residual. And if the rate constant for a linearly convergent $E^{(t)}$ is smaller than $\sqrt{1/(1+\sigma)}$ then the convergence of the primal-dual residual is linear with constant $1/(1+\sigma)$ as in a periodic algorithm.

ii) *Impact of free parameters:* In Theorem 6.2, several free parameters such as k_1, k_2, k_3, k_4, k_5 are used to describe convergence results. How the specific values of them affect the result is explained in the following.

   – k_1 is used in (6.30) and should be selected in set $(0, 2\underline{\lambda}(M))$. It directly affects the choices of H and β, as suggested by the sufficient condition in (6.11) for convergence.

   – k_2 and k_3 are used in (6.20) to separate triggering errors from the primal-dual residual, and should be chosen in $(0, \infty)$ and $(2, \infty)$, respectively. A smaller k_2 and a larger k_3 give a conservative constant C in (6.12), but allow us to get a larger σ and therefore faster convergence.

   – k_4 should be in $(0, 1)$. Its role can be observed in (6.35) and (6.36), where the relation between $\left\|(P \otimes I)\left(\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}\right)\right\|^2$ and the primal-dual residual is established. A larger k_4 renders the weight on the primal residual in (6.12) heavier, but makes σ smaller to get (6.35) satisfied.

   – The proof shows that the key for linear convergence is the satisfaction of (6.35) with a sufficiently small $\sigma + k_5$. This implies that $\sigma + k_5$ can only take values in

$$(0, \min\{R_1, R_2, R_3\}), \tag{6.13}$$

where

$$R_1 := \frac{\underline{\lambda}\big(k_4 Q (L_f^{-1})^2\big)(1-2/k_3)\underline{\lambda}(\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})}{k_2+k_3-1}$$
$$R_2 := \underline{\lambda}\Big(P^{-1}-L_f(P^{-1})^2/k_1\Big)\underline{\lambda}(\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})(1-2/k_3)$$
$$R_3 := (1-k_4)Q(P+k_4Q)^{-1},$$

to ensure (6.35). Therefore setting a larger k_5 in (6.13) leads to a smaller σ and slower convergence. In particular, given k_5 and a linearly decreasing threshold $E^{(t)} = E^{(0)}\rho^t$, the convergence rate becomes

$$\mathcal{O}\left(\max\left\{\rho^2, \frac{1}{\min\{R_1,R_2,R_3\}-k_5}\right\}^t\right).$$

iii) *Choices of β, H and $E^{(t)}$:* Selecting a larger η_i and a smaller β generally leads to heavier weights, i.e., $P+k_4Q$ and $\big(\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}\big)^{-1}$, on the primal and dual residuals in (6.12). However, a larger spectral radius of $H-\beta\mathcal{L}$ also results in a smaller $\min(R_1,R_2,R_3)$ in (6.13) and therefore slower convergence. For the reasonable choices of these parameters, one can set $\beta = 1/(\overline{\lambda}(\mathcal{L})+1)$ and $H = I + L_f^2/k_1$. Since the slower one in $E^{(t)}$ and $1/\sqrt{(1+\sigma)^t}$ will dominate the convergence, it is then always preferable to choose an exponentially decreasing sequence for the triggering threshold in decentralized strongly convex optimization. For the base constant, $E_i^{(0)} = 0.1|z_i^{(0)} + \nabla f_i(x_i^{(0)}) + \beta\sum_{j\in\mathcal{N}_i}(x_i^{(0)}-x_j^{(0)})|$ is an appropriate choice.

6.4 Proofs of Convergence Results

6.4.1 Proof of Theorem 6.1

Before developing the proof for Theorem 6.1, several useful technical lemmas are presented.

Lemma 6.1. *[114] Given a positive semidefinite matrix $W \in \mathbb{R}^{m\times m}$, it holds*

$$2\langle Wu, v\rangle = \|u\|_W^2 + \|v\|_W^2 - \|u-v\|_W^2, \forall u, v \in \mathbb{R}^m. \tag{6.14}$$

Lemma 6.2. *[112] If Assumption 6.1 holds, then for each $\mathbf{y} \in span^{\perp}(\mathbf{1}\otimes I)$, there*

exists a unique $\mathbf{y}' \in span^{\perp}(\mathbf{1}\otimes I)$ *such that* $\mathbf{y} = (\mathcal{L}\otimes I)\mathbf{y}'$ *and vice versa.*

Lemma 6.3. *If all the conditions in Theorem 6.1 hold, then, for any* $\mathbf{x} \in null(\mathcal{L}\otimes I)$ *and* $\mathbf{z} \in span^{\perp}(\mathbf{1}\otimes I)$*,*

$$\begin{aligned}
&\mathbf{F}(\mathbf{x}^{(\tau+1)}) - \mathbf{F}(\mathbf{x}) + \left\langle \mathbf{z}, \mathbf{x}^{(\tau+1)} \right\rangle \\
\leq& -\frac{1}{2}\left\|\mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)}\right\|^2_{(P-L_f)\otimes I} - \frac{1}{2}\left(\left\|\mathbf{x}^{(\tau+1)} - \mathbf{x}\right\|^2_{P\otimes I} - \left\|\mathbf{x}^{(\tau)} - \mathbf{x}\right\|^2_{P\otimes I}\right) \\
&- \left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, \beta(\mathcal{L}\otimes I)\left(\mathbf{e}^{(\tau)} - \mathbf{e}^{(\tau+1)}\right)\right\rangle + \left\langle \mathbf{e}^{(\tau+1)}, \mathbf{z}^{(\tau+1)} - \mathbf{z}\right\rangle \\
&+ \frac{1}{2}\left(\left\|\mathbf{z} - \mathbf{z}^{(\tau)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} - \left\|\mathbf{z} - \mathbf{z}^{(\tau+1)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}\right) \\
&- \frac{1}{2}\left\|\mathbf{z}^{(\tau+1)} - \mathbf{z}^{(\tau)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}
\end{aligned} \tag{6.15}$$

where P *is defined in Theorem 6.1 and* $\mathbf{e}^{(\tau)} = \tilde{\mathbf{x}}^{(\tau)} - \mathbf{x}^{(\tau)}$*.*

Proof of Lemma 6.3. By the smoothness of $\mathbf{f}$, we have

$$\mathbf{f}(\mathbf{x}^{(\tau+1)}) \leq \mathbf{f}(\mathbf{x}^{(\tau)}) + \left\langle \nabla\mathbf{f}(\mathbf{x}^{(\tau)}), \mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)}\right\rangle + \frac{1}{2}\left\|\mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)}\right\|^2_{L_f\otimes I}.$$

It follows from the convexity of $\mathbf{f}$

$$\mathbf{f}(\mathbf{x}^{(\tau)}) + \left\langle \nabla\mathbf{f}(\mathbf{x}^{(\tau)}), \mathbf{x} - \mathbf{x}^{(\tau)}\right\rangle \leq \mathbf{f}(\mathbf{x})$$

and $\mathbf{h}$

$$\mathbf{h}(\mathbf{x}^{(\tau+1)}) - \mathbf{h}(\mathbf{x}) \leq \left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, \tilde{\nabla}\mathbf{h}(\mathbf{x}^{(\tau+1)})\right\rangle$$

that

$$\mathbf{F}(\mathbf{x}^{(\tau+1)}) - \mathbf{F}(\mathbf{x}) \leq \left\langle \nabla\mathbf{f}(\mathbf{x}^{(\tau)}) + \tilde{\nabla}\mathbf{h}(\mathbf{x}^{(\tau+1)}), \mathbf{x}^{(\tau+1)} - \mathbf{x}\right\rangle + \frac{1}{2}\left\|\mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)}\right\|^2_{L_f\otimes I} \tag{6.16}$$

where $\tilde{\nabla}\mathbf{h}(\mathbf{x}^{(\tau+1)})$ is a subgradient of $\mathbf{h}$ evaluated at $\mathbf{x}^{(\tau+1)}$. From the iteration rule, we have

$$\begin{aligned}
0 =& \nabla\mathbf{f}(\mathbf{x}^{(\tau)}) + \mathbf{z}^{(\tau)} + \tilde{\nabla}\mathbf{h}(\mathbf{x}^{(\tau+1)}) - (H\otimes I)(\mathbf{x}^{(\tau)} - \mathbf{x}^{(\tau+1)}) + \beta(\mathcal{L}\otimes I)\tilde{\mathbf{x}}^{(\tau)} \\
0 =& \mathbf{z}^{(\tau+1)} - \mathbf{z}^{(\tau)} - \beta(\mathcal{L}\otimes I)\tilde{\mathbf{x}}^{(\tau+1)}.
\end{aligned}$$

This implies

$$0 = \nabla \mathbf{f}(\mathbf{x}^{(\tau)}) + \tilde{\nabla}\mathbf{h}(\mathbf{x}^{(\tau+1)}) + \mathbf{z}^{(\tau+1)} + (P \otimes I)(\mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)}) + \beta(\mathcal{L} \otimes I)(\mathbf{e}^{(\tau)} - \mathbf{e}^{(\tau+1)}). \tag{6.17}$$

Calculating the inner products of $\mathbf{x}^{(\tau+1)} - \mathbf{x}$ with both sides of the above equation leads to

$$\begin{aligned} &\left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, \nabla \mathbf{f}(\mathbf{x}^{(\tau)}) + \tilde{\nabla}\mathbf{h}(\mathbf{x}^{(\tau+1)}) \right\rangle \\ =& - \left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, \mathbf{z}^{(\tau+1)} - \mathbf{z} \right\rangle - \left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, (P \otimes I)(\mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)}) \right\rangle \\ &- \left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, \beta(\mathcal{L} \otimes I)\left(\mathbf{e}^{(\tau)} - \mathbf{e}^{(\tau+1)}\right) \right\rangle - \left\langle \mathbf{x}^{(\tau+1)}, \mathbf{z} \right\rangle \end{aligned} \tag{6.18}$$

for any $\mathbf{x} \in null(\mathcal{L} \otimes I)$ and $\mathbf{z} \in span^{\perp}(\mathbf{1} \otimes I)$. From Lemma 6.2 and the fact that

$$\mathbf{z}^{(\tau+1)} = \beta(\mathcal{L} \otimes I) \sum_{\iota=0}^{\tau+1} \tilde{\mathbf{x}}^{(\iota)},$$

we obtain

$$\begin{aligned} &\left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, \mathbf{z}^{(\tau+1)} - \mathbf{z} \right\rangle = \left\langle \tilde{\mathbf{x}}^{(\tau+1)}, \mathbf{z}^{(\tau+1)} - \mathbf{z} \right\rangle - \left\langle \mathbf{e}^{(\tau+1)}, \mathbf{z}^{(\tau+1)} - \mathbf{z} \right\rangle \\ &= \left\langle \beta(\mathcal{L} \otimes I)\tilde{\mathbf{x}}^{(\tau+1)}, \mathbf{z}'^{(\tau+1)} - \mathbf{z}' \right\rangle - \left\langle \mathbf{e}^{(\tau+1)}, \mathbf{z}^{(\tau+1)} - \mathbf{z} \right\rangle \\ &= \left\langle \beta(\mathcal{L} \otimes I)\left(\mathbf{z}'^{(\tau+1)} - \mathbf{z}'^{(\tau)}\right), \mathbf{z}'^{(\tau+1)} - \mathbf{z}' \right\rangle - \left\langle \mathbf{e}^{(\tau+1)}, \mathbf{z}^{(\tau+1)} - \mathbf{z} \right\rangle. \end{aligned} \tag{6.19}$$

It follows

$$\begin{aligned} &\mathbf{F}(\mathbf{x}^{(\tau+1)}) - \mathbf{F}(\mathbf{x}) + \left\langle \mathbf{z}, \mathbf{x}^{(\tau+1)} \right\rangle \\ \overset{i}{\leq} &\frac{1}{2}\left\|\mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)}\right\|^2_{L_f \otimes I} - \left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, (P \otimes I)\left(\mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)}\right) \right\rangle + \left\langle \mathbf{e}^{(\tau+1)}, \mathbf{z}^{(\tau+1)} - \mathbf{z} \right\rangle \\ &- \left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, \beta(\mathcal{L} \otimes I)\left(\mathbf{e}^{(\tau)} - \mathbf{e}^{(\tau+1)}\right) \right\rangle - \left\langle \beta(\mathcal{L} \otimes I)\left(\mathbf{z}'^{(\tau+1)} - \mathbf{z}'^{(\tau)}\right), \mathbf{z}'^{(\tau+1)} - \mathbf{z}' \right\rangle \\ \overset{ii}{=} &-\frac{1}{2}\left\|\mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)}\right\|^2_{(P - L_f) \otimes I} - \frac{1}{2}\left(\left\|\mathbf{x}^{(\tau+1)} - \mathbf{x}\right\|^2_{P \otimes I} - \left\|\mathbf{x}^{(\tau)} - \mathbf{x}\right\|^2_{P \otimes I}\right) \\ &- \left\langle \mathbf{x}^{(\tau+1)} - \mathbf{x}, \beta(\mathcal{L} \otimes I)\left(\mathbf{e}^{(\tau)} - \mathbf{e}^{(\tau+1)}\right) \right\rangle + \left\langle \mathbf{e}^{(\tau+1)}, \mathbf{z}^{(\tau+1)} - \mathbf{z} \right\rangle \\ &+ \frac{1}{2}\left(\left\|\mathbf{z}' - \mathbf{z}'^{(\tau)}\right\|^2_{\beta(\mathcal{L} \otimes I)} - \left\|\mathbf{z}'^{(\tau+1)} - \mathbf{z}'\right\|^2_{\beta(\mathcal{L} \otimes I)}\right) - \frac{1}{2}\left\|\mathbf{z}'^{(\tau+1)} - \mathbf{z}'^{(\tau)}\right\|^2_{\beta(\mathcal{L} \otimes I)}, \end{aligned} \tag{6.20}$$

where we plug (6.18) and (6.19) into (6.16) to get "i" and use Lemma 6.1 and

$$P \succ L_f \succeq 0,\ \beta\mathcal{L} \succeq 0$$

to get "ii". Due to $\mathbf{z}, \mathbf{z}', \mathbf{z}'^{(k)} \in span^{\perp}(\mathbf{1} \otimes I)$, we have

$$\mathbf{z} - \mathbf{z}^{(\tau)} = \beta(\mathcal{L} \otimes I)\left(\mathbf{z}' - \mathbf{z}'^{(\tau)}\right) = \left(\beta(\mathcal{L} \otimes I) + \frac{(\mathbf{1}\mathbf{1}^{\mathrm{T}}) \otimes I}{n}\right)\left(\mathbf{z}' - \mathbf{z}'^{(\tau)}\right).$$

and therefore $\mathbf{z}' - \mathbf{z}'^{(\tau)} = \left((\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}) \otimes I\right)^{-1}\left(\mathbf{z} - \mathbf{z}^{(\tau)}\right)$. Then we consider

$$\left\|\mathbf{z}' - \mathbf{z}'^{(\tau)}\right\|^2_{\beta(\mathcal{L}\otimes I)} = \left\langle \mathbf{z} - \mathbf{z}^{(\tau)}, \mathbf{z}' - \mathbf{z}'^{(\tau)} \right\rangle = \left\|\mathbf{z} - \mathbf{z}^{(\tau)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}},$$

which together with (6.20) gives the desired inequality. □

Lemma 6.4. *If all the conditions in Theorem 6.1 hold, then, for $\tau \leq t$,*

$$\left\|\mathbf{x}^{(\tau)} - \mathbf{x}^*\right\| + \left\|\mathbf{z}^* - \mathbf{z}^{(\tau)}\right\| \leq 2A_t + \sqrt{2/b}\left(\left\|\mathbf{x}^{(0)} - \mathbf{x}^*\right\|_{P\otimes I} + \left\|\mathbf{z}^{(0)} - \mathbf{z}^*\right\|_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}\right)$$

where P, b and A_t are defined in Theorem 6.1.

Proof of Lemma 6.4. First, we use the convexity of $\mathbf{F}$ and the KKT conditions (6.3) to obtain

$$\begin{aligned} &\mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{x}^*) + \left\langle \mathbf{y}^*, \left(\sqrt{\mathcal{L}} \otimes I\right)\mathbf{x} \right\rangle \\ \geq &\left\langle \tilde{\nabla}\mathbf{F}(\mathbf{x}^*), \mathbf{x} - \mathbf{x}^* \right\rangle + \left\langle \mathbf{y}^*, \left(\sqrt{\mathcal{L}} \otimes I\right)(\mathbf{x} - \mathbf{x}^*) \right\rangle \\ = &\left\langle \tilde{\nabla}\mathbf{F}(\mathbf{x}^*) + \left(\sqrt{\mathcal{L}} \otimes I\right)\mathbf{y}^*, \mathbf{x} - \mathbf{x}^* \right\rangle = 0, \forall \mathbf{x} \end{aligned} \tag{6.21}$$

where $\tilde{\nabla}\mathbf{F}(\mathbf{x}^*) \in \partial\mathbf{F}(\mathbf{x}^*)$. Then, we let $\mathbf{x} = \mathbf{x}^*$ and $\mathbf{z} = \mathbf{z}^* = (\sqrt{\mathcal{L}} \otimes I)\mathbf{y}^*$ in (6.15), and sum the resultant inequality from $k = 0$ to $k = t - 1$ to get

$$\begin{aligned}
0 \leq & \sum_{\tau=0}^{t-1} \left(\mathbf{F}(\mathbf{x}^{(\tau+1)}) - \mathbf{F}(\mathbf{x}^*) + \left\langle \mathbf{y}^*, (\sqrt{\mathcal{L}} \otimes I)\mathbf{x}^{(\tau+1)} \right\rangle \right) \\
\leq & -\frac{1}{2} \left(\sum_{\tau=0}^{t-1} \left\| \mathbf{x}^{(\tau+1)} - \mathbf{x}^{(\tau)} \right\|^2_{(P-L_f)\otimes I} + \left\| \mathbf{x}^{(t)} - \mathbf{x}^* \right\|^2_{P\otimes I} - \left\| \mathbf{x}^{(0)} - \mathbf{x}^* \right\|^2_{P\otimes I} \right) \\
& - \frac{1}{2} \left\| \mathbf{z}^* - \mathbf{z}^{(t)} \right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} + \frac{1}{2} \left\| \mathbf{z}^* - \mathbf{z}^{(0)} \right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} \\
& + \sum_{\tau=0}^{t-1} \left(\left\langle \mathbf{x}^* - \mathbf{x}^{(\tau+1)}, \beta(\mathcal{L} \otimes I) \left(\mathbf{e}^{(\tau)} - \mathbf{e}^{(\tau+1)} \right) \right\rangle + \left\langle \mathbf{z}^{(\tau+1)} - \mathbf{z}^*, \mathbf{e}^{(\tau+1)} \right\rangle \right) \\
& - \frac{1}{2} \sum_{\tau=0}^{t-1} \left\| \mathbf{z}^{(\tau+1)} - \mathbf{z}^{(\tau)} \right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}.
\end{aligned} \tag{6.22}$$

Since $P - L_f \succ 0$ and $(\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})^{-1} \succ 0$, it holds

$$\begin{aligned}
& \frac{1}{2} \left\| \mathbf{x}^{(t)} - \mathbf{x}^* \right\|^2_{P\otimes I} + \frac{1}{2} \left\| \mathbf{z}^* - \mathbf{z}^{(t)} \right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} \\
\leq & \frac{1}{2} \left\| \mathbf{x}^{(0)} - \mathbf{x}^* \right\|^2_{P\otimes I} + \frac{1}{2} \left\| \mathbf{z}^{(0)} - \mathbf{z}^* \right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} \\
& + \sum_{\tau=1}^{t} \left(\left\langle \mathbf{x}^* - \mathbf{x}^{(\tau)}, \beta\mathcal{L} \left(\mathbf{e}^{(\tau-1)} - \mathbf{e}^{(\tau)} \right) \right\rangle + \left\langle \mathbf{z}^{(\tau)} - \mathbf{z}^*, \mathbf{e}^{(\tau)} \right\rangle \right).
\end{aligned}$$

By the monotonicity of $E^{(t)}$ and the Cauchy-Schwarz inequality, we further have

$$\begin{aligned}
& \frac{1}{4} \min \left\{ \underline{\lambda}(L_f), \frac{1}{\overline{\lambda}(\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})} \right\} \left(\left\| \mathbf{x}^{(t)} - \mathbf{x}^* \right\| + \left\| \mathbf{z}^* - \mathbf{z}^{(t)} \right\| \right)^2 \\
\leq & \frac{1}{2} \left\| \mathbf{x}^{(t)} - \mathbf{x}^* \right\|^2_{P\otimes I} + \frac{1}{2} \left\| \mathbf{z}^* - \mathbf{z}^{(t)} \right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} \\
& + \sum_{\tau=1}^{t} \left(\left\langle \mathbf{x}^* - \mathbf{x}^{(\tau)}, \beta\mathcal{L} \left(\mathbf{e}^{(\tau-1)} - \mathbf{e}^{(\tau)} \right) \right\rangle + \left\langle \mathbf{z}^{(\tau)} - \mathbf{z}^*, \mathbf{e}^{(\tau)} \right\rangle \right) \\
\leq & \frac{1}{2} \left\| \mathbf{x}^{(0)} - \mathbf{x}^* \right\|^2_{P\otimes I} + \frac{1}{2} \left\| \mathbf{z}^{(0)} - \mathbf{z}^* \right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} \\
& + \sum_{\tau=1}^{t} \max \left\{ 2\beta\overline{\lambda}(\mathcal{L}), 1 \right\} \sqrt{n} E^{(\tau-1)} \left(\left\| \mathbf{x}^* - \mathbf{x}^{(\tau)} \right\| + \left\| \mathbf{z}^{(\tau)} - \mathbf{z}^* \right\| \right).
\end{aligned} \tag{6.23}$$

Upon using Lemma 1 in [79], we obtain

$$\begin{aligned}&\left\|\mathbf{x}^{(t)}-\mathbf{x}^*\right\|+\left\|\mathbf{z}^*-\mathbf{z}^{(t)}\right\|\\&\leq A_t+\sqrt{\frac{2\left\|\mathbf{x}^{(0)}-\mathbf{x}^*\right\|_{P\otimes I}+2\left\|\mathbf{z}^{(0)}-\mathbf{z}^*\right\|_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}}{b}+A_t^2}\end{aligned}$$

where b and A_t are defined in Theorem 6.1. By the monotonicity and positivity of A_t, the desired result follows. □

We are now in a position to present the proof for Theorem 6.1.

Proof of Theorem 6.1. Manipulating (6.24) and using the similar procedure as in (6.23) allow us to get

$$\begin{aligned}&\frac{1}{2}\sum_{\tau=0}^{t-1}\left(\left\|\mathbf{x}^{(\tau+1)}-\mathbf{x}^{(\tau)}\right\|_{(P-L_f)\otimes I}^2+\left\|\mathbf{z}^{(\tau+1)}-\mathbf{z}^{(\tau)}\right\|_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}^2\right)\\\leq&\frac{1}{2}\left\|\mathbf{x}^{(0)}-\mathbf{x}^*\right\|_{P\otimes I}^2+\frac{1}{2}\left\|\mathbf{z}^*-\mathbf{z}^{(0)}\right\|_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}^2\\&+\left(\left\|\mathbf{x}^*-\mathbf{x}^{(t)}\right\|+\left\|\mathbf{z}^*-\mathbf{z}^{(t)}\right\|\right)\sum_{\tau=1}^{t}a\sqrt{n}E^{(\tau-1)}\end{aligned}\tag{6.24}$$

where a is defined in Theorem 6.1. In light of Lemma 6.4, we have that if $E^{(\tau)}$ is summable, then

$$\sum_{\tau=0}^{\infty}\left(\left\|\mathbf{x}^{(\tau+1)}-\mathbf{x}^{(\tau)}\right\|_{(P-L_f)\otimes I}^2+\left\|\mathbf{z}^{(\tau)}-\mathbf{z}^{(\tau+1)}\right\|_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}^2\right)<\infty.$$

Since $P-L_f\succ 0,\left(\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}\right)^{-1}\succ 0$, we further have

$$\lim_{\tau\to\infty}\left(\mathbf{x}^{(t+1)},\mathbf{z}^{(t+1)}\right)-\left(\mathbf{x}^{(t)},\mathbf{z}^{(t)}\right)=0.$$

Denote the limit point of $\left\{\left(\mathbf{x}^{(t)},\mathbf{z}^{(t)}\right)\right\}_{t\geq 1}$ by $(\mathbf{x}_\infty,\mathbf{z}_\infty)$. Note that $\lim\limits_{t\to\infty}E^{(t)}=0$ by assumptions. From

$$\beta(\mathcal{L}\otimes I)\left(\mathbf{e}^{(\tau+1)}+\mathbf{x}^{(\tau+1)}\right)=\mathbf{z}^{(\tau+1)}-\mathbf{z}^{(\tau)},$$

and

$$0 = \tilde{\nabla}\mathbf{F}(\mathbf{x}^{(\tau+1)}) + \mathbf{z}^{(\tau)} - (H \otimes I)\mathbf{x}^{(\tau)} + \beta(\mathcal{L} \otimes I)\left(\mathbf{e}^{(\tau)} + \mathbf{x}^{(\tau)}\right) + (H \otimes I)\mathbf{x}^{(\tau+1)}$$

where $\tilde{\nabla}\mathbf{F}(\mathbf{x}^{(\tau+1)})$ is a subgradient of $\mathbf{F}$ over $\mathbf{x}^{(\tau+1)}$, we obtain $(\mathcal{L} \otimes I)\mathbf{x}_\infty = 0$ and $\tilde{\nabla}\mathbf{F}(\mathbf{x}_\infty) + \mathbf{z}_\infty = 0$, respectively. This implies that $(\mathbf{x}_\infty, \mathbf{y}_\infty)$ is a KKT point, where $\mathbf{z}_\infty = (\sqrt{\mathcal{L}} \otimes I)\mathbf{y}_\infty$. Again, from (6.24), we have

$$\begin{aligned}
&\sum_{\tau=0}^{t-1}\left(\mathbf{F}(\mathbf{x}^{(\tau+1)}) - \mathbf{F}(\mathbf{x}^*) + \left\langle \mathbf{y}^*, (\sqrt{\mathcal{L}} \otimes I)\mathbf{x}^{(\tau+1)} \right\rangle\right) \\
\leq &\frac{1}{2}\left\|\mathbf{x}^{(0)} - \mathbf{x}^*\right\|^2_{P\otimes I} + \frac{1}{2}\left\|\mathbf{z}^{(0)} - \mathbf{z}^*\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} \\
&+ \frac{b}{2}A_t\left(2A_t + \sqrt{2/b}\left(\left\|\mathbf{x}^{(0)} - \mathbf{x}^*\right\|_{P\otimes I} + \left\|\mathbf{z}^{(0)} - \mathbf{z}^*\right\|_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}\right)\right) \\
\leq &\left(\frac{\left\|\mathbf{x}^{(0)} - \mathbf{x}^*\right\|_{P\otimes I} + \left\|\mathbf{z}^{(0)} - \mathbf{z}^*\right\|_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}}{\sqrt{2}} + \sqrt{b}A_t\right)^2,
\end{aligned} \tag{6.25}$$

which in conjunction with

$$\begin{aligned}
&t\left(\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*) + \left\langle \mathbf{y}^*, (\sqrt{\mathcal{L}} \otimes I)\hat{\mathbf{x}}^{(t)} \right\rangle\right) \\
&\leq \sum_{\tau=0}^{t-1}\left(\mathbf{F}(\mathbf{x}^{(\tau+1)}) - \mathbf{F}(\mathbf{x}^*) + \left\langle \mathbf{y}^*, (\sqrt{\mathcal{L}} \otimes I)\mathbf{x}^{(\tau+1)} \right\rangle\right)
\end{aligned}$$

gives

$$\begin{aligned}
&\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*) + \left\langle \mathbf{y}^*, (\sqrt{\mathcal{L}} \otimes I)\hat{\mathbf{x}}^{(t)} \right\rangle \\
\leq &\frac{1}{2t}\left(\left\|\mathbf{x}^{(0)} - \mathbf{x}^*\right\|_{P\otimes I} + \left\|\mathbf{z}^{(0)} - \mathbf{z}^*\right\|_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} + \sqrt{2b}A_t\right)^2 \\
= &\frac{1}{2t}\left(\left\|\mathbf{x}^{(0)} - \mathbf{x}^*\right\|_{P\otimes I} + \left\|(\sqrt{\mathcal{L}} \otimes I)\mathbf{y}^*\right\|_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} + \sqrt{2b}A_t\right)^2
\end{aligned}$$

where the initialization step $\mathbf{z}^{(0)} = 0$ is used to get the last equality. Finally, we

consider

$$\begin{aligned}
&\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*) \leq \mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*) + \rho \left\| (\sqrt{\mathcal{L}} \otimes I)\hat{\mathbf{x}}^{(t)} \right\| \\
\leq &\sup_{\|\mathbf{y}^*\| \leq \rho} \frac{\left(\left\| \mathbf{x}^{(0)} - \mathbf{x}^* \right\|_{P \otimes I} + \left\| (\sqrt{\mathcal{L}} \otimes I)\mathbf{y}^* \right\|_{\left((\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}) \otimes I\right)^{-1}} + \sqrt{2b} A_t \right)^2}{2t} \\
\leq &\frac{\left(\left\| \mathbf{x}^{(0)} - \mathbf{x}^* \right\|_{P \otimes I} + \rho \left\| \mathcal{L}(\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})^{-1} \right\| + \sqrt{2b} A_t \right)^2}{2t}.
\end{aligned} \tag{6.26}$$

By (6.21), it holds that

$$\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*) \geq - \|\mathbf{y}^*\| \left\| (\sqrt{\mathcal{L}} \otimes I)\hat{\mathbf{x}}^{(t)} \right\|. \tag{6.27}$$

By combining (6.27) with (6.26), one gets

$$(\rho - \|\mathbf{y}^*\|) \left\| (\sqrt{\mathcal{L}} \otimes I)\hat{\mathbf{x}}^{(t)} \right\| \leq \frac{\left(\left\| \mathbf{x}^{(0)} - \mathbf{x}^* \right\|_{P \otimes I} + \rho \left\| \mathcal{L}(\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})^{-1} \right\| + \sqrt{2b} A_t \right)^2}{2t}.$$

Therefore the bound for $\left\| (\sqrt{\mathcal{L}} \otimes I)\hat{\mathbf{x}}^{(t)} \right\|$ in (6.9) holds. Using (6.27) again allows us to obtain the lower bound for $\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*)$ in (6.10). This completes the proof. □

6.4.2 Proof of Theorem 6.2

Proof of Theorem 6.2. By setting $\tilde{\nabla}\mathbf{h}(\mathbf{x}^{(t+1)}) = 0$ in (6.17) and $\mathbf{z}^* = (\sqrt{\mathcal{L}} \otimes I)\mathbf{y}^*$ in (6.3a), we have

$$\begin{aligned}
0 = &\nabla\mathbf{f}(\mathbf{x}^{(t)}) - \nabla\mathbf{f}(\mathbf{x}^*) + \mathbf{z}^{(t+1)} - \mathbf{z}^* + (P \otimes I)\left(\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}\right) \\
&+ \beta(\mathcal{L} \otimes I)\left(\mathbf{e}^{(t)} - \mathbf{e}^{(t+1)}\right).
\end{aligned} \tag{6.28}$$

As in the proof of Lemma 6.3, we consider the inner products of $\mathbf{x}^{(t+1)}-\mathbf{x}^*$ with both sides of the above equality

$$\begin{aligned}&\underbrace{\left\langle \mathbf{x}^{(t+1)}-\mathbf{x}^*, \nabla\mathbf{f}(\mathbf{x}^{(t)})-\nabla\mathbf{f}(\mathbf{x}^*)\right\rangle}_{i}+\underbrace{\left\langle \mathbf{x}^{(t+1)}-\mathbf{x}^*, \mathbf{z}^{(t+1)}-\mathbf{z}^*\right\rangle}_{ii}\\&+\underbrace{\left\langle \mathbf{x}^{(t+1)}-\mathbf{x}^*, (P\otimes I)\left(\mathbf{x}^{(t+1)}-\mathbf{x}^{(t)}\right)\right\rangle}_{iii}+\left\langle \mathbf{x}^{(t+1)}-\mathbf{x}^*, \beta(\mathcal{L}\otimes I)\left(\mathbf{e}^{(t)}-\mathbf{e}^{(t+1)}\right)\right\rangle=0.\end{aligned} \tag{6.29}$$

For "i", we consider $\nabla\mathbf{f}(\mathbf{x}^{(t)})-\nabla\mathbf{f}(\mathbf{x}^*)=\nabla\mathbf{f}(\mathbf{x}^{(t)})-\nabla\mathbf{f}(\mathbf{x}^{(t+1)})+\nabla\mathbf{f}(\mathbf{x}^{(t+1)})-\nabla\mathbf{f}(\mathbf{x}^*)$ and get from the strong convexity and smoothness of $\mathbf{f}$ that

$$\begin{aligned}i\geq&\left\|\mathbf{x}^{(t+1)}-\mathbf{x}^*\right\|_{M\otimes I}^2-\frac{1}{2k_1}\left\|\nabla\mathbf{f}(\mathbf{x}^{(t)})-\nabla\mathbf{f}(\mathbf{x}^{(t+1)})\right\|^2-\frac{k_1}{2}\left\|\mathbf{x}^{(t+1)}-\mathbf{x}^*\right\|^2\\\geq&\frac{1}{2}\left\|\mathbf{x}^{(t+1)}-\mathbf{x}^*\right\|_{Q\otimes I}^2-\frac{1}{2k_1}\left\|\mathbf{x}^{(t)}-\mathbf{x}^{(t+1)}\right\|_{L_f^2\otimes I}^2.\end{aligned} \tag{6.30}$$

Using the same reasoning as in (6.19), we have

$$\begin{aligned}ii=&\frac{1}{2}\left\|\mathbf{z}^*-\mathbf{z}^{(t+1)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}+\frac{1}{2}\left\|\mathbf{z}^{(t)}-\mathbf{z}^{(t+1)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}\\&-\frac{1}{2}\left\|\mathbf{z}^{(t)}-\mathbf{z}^*\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}-\left\langle \mathbf{e}^{(t+1)}, \mathbf{z}^{(t+1)}-\mathbf{z}^*\right\rangle.\end{aligned} \tag{6.31}$$

Using Lemma 6.1 allows us to obtain

$$iii=\frac{1}{2}\left(\left\|\mathbf{x}^{(t+1)}-\mathbf{x}^*\right\|_{P\otimes I}^2+\left\|\mathbf{x}^{(t+1)}-\mathbf{x}^{(t)}\right\|_{P\otimes I}^2-\left\|\mathbf{x}^{(t)}-\mathbf{x}^*\right\|_{P\otimes I}^2\right). \tag{6.32}$$

Combing equations. (6.29)-(6.32) yields

$$\begin{aligned}&\frac{1}{2}\left\|\mathbf{x}^{(t)}-\mathbf{x}^*\right\|_{P\otimes I}^2+\frac{1}{2}\left\|\mathbf{z}^{(t)}-\mathbf{z}^*\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}\\&+\left\langle \mathbf{e}^{(t+1)}, \mathbf{z}^{(t+1)}-\mathbf{z}^*\right\rangle-\left\langle \mathbf{x}^{(t+1)}-\mathbf{x}^*, \beta(\mathcal{L}\otimes I)\left(\mathbf{e}^{(t)}-\mathbf{e}^{(t+1)}\right)\right\rangle\\\geq&\frac{1}{2}\left\|\mathbf{x}^{(t+1)}-\mathbf{x}^*\right\|^2_{(P+Q)\otimes I}+\frac{1}{2}\left\|\mathbf{x}^{(t+1)}-\mathbf{x}^{(t)}\right\|^2_{(P-L_f^2/k_1)\otimes I}\\&+\frac{1}{2}\left\|\mathbf{z}^*-\mathbf{z}^{(t+1)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}+\frac{1}{2}\left\|\mathbf{z}^{(t)}-\mathbf{z}^{(t+1)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}.\end{aligned} \tag{6.33}$$

In order to obtain linear convergence from (6.33), we establish a relation between $\left\|\mathbf{x}^{(t+1)}-\mathbf{x}^{(t)}\right\|^2$ and the primal-dual residual in the following. By (6.28) and the

inequality

$$2\langle \mathbf{u}, \mathbf{v}\rangle \geq -w\|\mathbf{u}\|^2 - \frac{1}{w}\|\mathbf{v}\|^2, \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^{mn}, w > 0,$$

it holds

$$\begin{aligned}
&\left\|(P\otimes I)\left(\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}\right)\right\|^2 \\
=&\left\|\nabla \mathbf{f}(\mathbf{x}^{(t)}) - \nabla \mathbf{f}(\mathbf{x}^*) + \mathbf{z}^{(t+1)} - \mathbf{z}^* + \beta(\mathcal{L}\otimes I)\left(\mathbf{e}^{(t)} - \mathbf{e}^{(t+1)}\right)\right\|^2 \\
\geq&(1-k_2-k_3)\left\|\mathbf{x}^{(t)} - \mathbf{x}^*\right\|^2_{L_f^2\otimes I} + (1-2/k_3)\left\|\mathbf{z}^{(t+1)} - \mathbf{z}^*\right\|^2 \\
&+(1-1/k_2-k_3)\left\|\beta(\mathcal{L}\otimes I)\left(\mathbf{e}^{(t)} - \mathbf{e}^{(t+1)}\right)\right\|^2
\end{aligned} \tag{6.34}$$

for any $k_2 > 0$ and $k_3 > 2$. If $\sigma + k_5$ is sufficiently small such that

$$\frac{(k_2+k_3-1)(\sigma+k_5)L_f^2}{(1-2/k_3)\underline{\lambda}(\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})} \preceq k_4 Q \tag{6.35a}$$

$$\frac{(\sigma+k_5)P^2}{(1-2/k_3)\underline{\lambda}(\beta\mathcal{L} + \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})} \preceq P - L_f^2/k_1 \tag{6.35b}$$

$$(\sigma+k_5)(P+k_4Q) \preceq (1-k_4)Q \tag{6.35c}$$

for some $0 < k_4 < 1$, then we can get from (6.34) that

$$\begin{aligned}
&\frac{1}{2}\left\|\mathbf{x}^{(t+1)} - \mathbf{x}^*\right\|^2_{(1-k_4)Q\otimes I} + \frac{1}{2}\left\|\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}\right\|^2_{(P-L_f^2/k_1)\otimes I} + \frac{1}{2}\left\|\mathbf{x}^{(t)} - \mathbf{x}^*\right\|^2_{k_4Q\otimes I} \\
&+\frac{1}{2}\left\|\mathbf{z}^{(t)} - \mathbf{z}^{(t+1)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} + \frac{(k_3+1/k_2-1)(\sigma+k_5)}{2(1-2/k_3)\underline{\lambda}(\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})}\left\|\beta(\mathcal{L}\otimes I)\left(\mathbf{e}^{(t)} - \mathbf{e}^{(t+1)}\right)\right\|^2 \\
&\geq \frac{\sigma+k_5}{2}\left(\left\|\mathbf{x}^{(t+1)} - \mathbf{x}^*\right\|^2_{(P+k_4Q)\otimes I} + \left\|\mathbf{z}^{(t+1)} - \mathbf{z}^*\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}\right).
\end{aligned} \tag{6.36}$$

Combining (6.36) and (6.33) leads to

$$\begin{aligned}
&\frac{1}{2}\left\|\mathbf{x}^{(t)} - \mathbf{x}^*\right\|^2_{(P+k_4Q)\otimes I} + \frac{1}{2}\left\|\mathbf{z}^{(t)} - \mathbf{z}^*\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}} + \left\langle \mathbf{e}^{(t+1)}, \mathbf{z}^{(t+1)} - \mathbf{z}^*\right\rangle \\
&+\frac{(k_3+1/k_2-1)(\sigma+k_5)}{2(1-2/k_3)\underline{\lambda}(\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})}\left\|\beta(\mathcal{L}\otimes I)\left(\mathbf{e}^{(t)} - \mathbf{e}^{(t+1)}\right)\right\|^2 \\
&-\left\langle \mathbf{x}^{(k+1)} - \mathbf{x}^*, \beta(\mathcal{L}\otimes I)\left(\mathbf{e}^{(t)} - \mathbf{e}^{(t+1)}\right)\right\rangle \\
\geq&\frac{\sigma+k_5+1}{2}\left(\left\|\mathbf{x}^{(t+1)} - \mathbf{x}^*\right\|^2_{(P+k_4Q)\otimes I} + \left\|\mathbf{z}^* - \mathbf{z}^{(t+1)}\right\|^2_{\left((\beta\mathcal{L}+\frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n})\otimes I\right)^{-1}}\right).
\end{aligned}$$

By monotonicity of $E^{(t)}$ and the inequality

$$\langle \mathbf{u}, \mathbf{v} \rangle \leq \frac{k_5}{2} \|\mathbf{u}\|_O^2 + \frac{1}{2k_5} \|\mathbf{v}\|_{O^{-1}}^2, \forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^{mn}, O \succ 0,$$

we are able to separate triggering errors from the primal-dual residual and arrive at (6.12). This completes the proof. □

6.5 Experiments

In this section, we test the proposed algorithm and compare it with some recent event-triggered decentralized optimization algorithms in the literature.

6.5.1 Decentralized l_1-l_2 Minimization

Consider the decentralized l_1-l_2 minimization problem:

$$\min_x \sum_{i=1}^{n} \left\{ \frac{1}{2} \|b_i - A_i x\|^2 + \tau_i \|x\|_1 \right\},$$

where data $A_i \in \mathbb{R}^{p_i \times m}$, $b_i \in \mathbb{R}^{p_i}$ and regularization parameter $\tau_i > 0$ are private to agent i. The two component functions for each agent are $f_i(x) = \|b_i - A_i x\|^2/2$ that is convex with Lipschitz continuous gradient, and $h_i(x) = \tau_i \|x\|_1$ that is convex but non-differentiable. The parameters are chosen as $p_i = 3$, $m = 50$, and $n = 100$; the data A_i and b_i are randomly generated with normalization.

In the simulation, a network of $n = 100$ agents is randomly chosen with connectivity ratio $r = 0.4$ [101], where r is defined as the number of links divided by the number of all possible links $n(n-1)/2$. We compare the performance of the proposed methods with the ADMM-based algorithm [85] and its event-triggered variant (COCA) [54]. For [54, 85], the projected scaled subgradient method available as a Matlab function $L1General2\_PSSgb$ in [78] is used to solve the subproblems with an accuracy of 10^{-10} in terms of the l_∞ norm of the subgradient. Communication strategies in which each agent triggers network transmission every two iterations or four iterations are also simulated. The parameters for these algorithms are manually tuned in periodic setting to achieve the best performance: $H = 0.6I, \beta = 0.0025$ and $c = 0.0025$ are considered for the proposed method and [54, 85], respectively. For event-triggered methods, the triggering thresholds for agents are set as $E_i^{(t)} = 20/t^{1.2}$.

The primal and dual iterates of all the methods are initialized with 0. The performance is evaluated in terms of the objective error $|\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*)|$ over the number of iteration steps and broadcasting times of the first agent.

The results are plotted in Figure 6.1. We observe that event-triggered LALM and COCA, while demonstrating comparable performance with their periodic counterparts, achieve significant communication reductions. Note that ADMM-based approaches outperform LALM-based ones, because the former used the original augmented Lagrangian while the latter used a linearized one to ease the computational burden of solving subproblems. As a consequence, ADMM and COCA consume much more computational resources than the proposed methods at each iteration. For this specific example, the time spent per iteration for COCA is 0.2431s on average and the time for the proposed method is 0.0068s. In practice, a trade-off between network utilization and computational resource consumption should be made. The periodic scheme of 2 periods halves the number of communication rounds for each agent. However, when the number of periods increases to 4, the iterates diverge. Compared to the periodic scheme of 2 periods, the proposed algorithm consumes less communication cost and is guaranteed to converge.

Then, a sparser random network with $r = 0.04$ is considered. The parameters are tuned as $H = 0.6I, \beta = 0.01$ to achieve the best performance. The results are presented in Figure 6.2. They indicate that the denser configuration ($r = 0.4$) leads to faster convergence, and each agent broadcasts less in denser networks to achieve a given accuracy. This is primarily because that a denser network has a more balanced set of weights for agents, and more information from neighbors can be used in each iteration/communication round.

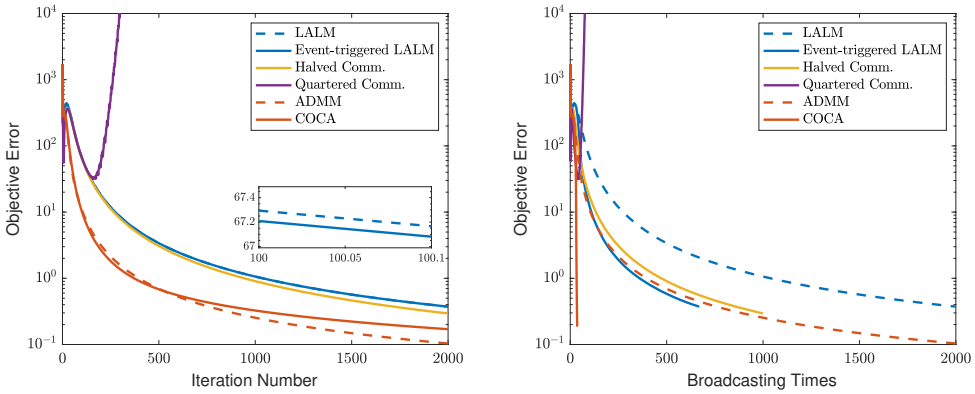


Figure 6.1: Objective error $|\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*)|$ versus iteration number and broadcasting times when $r = 0.4$.

Algorithms	Time spent per iteration (sec)
COCA	0.2431
Event-triggered LALM	0.0068

Table 6.1: The time spent per iteration for COCA and event-triggered LALM

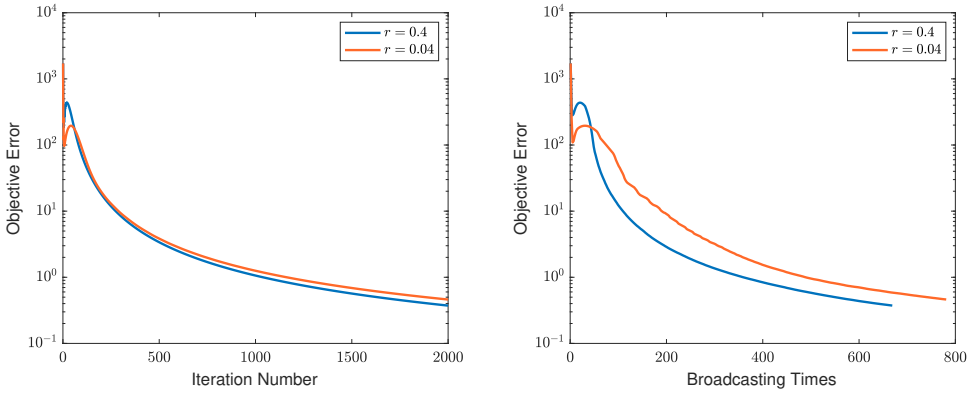


Figure 6.2: Objective error $|\mathbf{F}(\hat{\mathbf{x}}^{(t)}) - \mathbf{F}(\mathbf{x}^*)|$ versus iteration number and broadcasting times in different random networks.

6.5.2 Decentralized Logistic Regression

Consider the following decentralized logistic regression problem:

$$\min_x \sum_{i=1}^{n} \Big\{ \sum_{j=1}^{m_i} \ln\Big(1+\exp\big(-y_j^i(M_j^{i\mathrm{T}}x)\big)\Big)\Big\}.$$

where the input features $M_j^i \in \mathbb{R}^m$ and the class labels $y_j^i \in \{-1,1\}$ with $j = 1,\cdots,m_i$ are private to each agent i. Note that we set the last element of the feature vector $M_j^i \in \mathbb{R}^m$ to 1 as in standard logistic regression, then the last element of the decision variable x becomes the adjustable bias of the logistic regression model. The number of samples for each agent i is $m_i = 8$, and the dimension for decision variable is $m = 10$. In the simulation, all the 400 samples are generated randomly. A network of $n = 100$ with $r = 0.04$ is considered.

The linearized ADMM-based algorithm (DLM) in [49] and the gradient-tracking method in [73], and their event-triggered variants [44] (COLA) and [23] are simulated for comparison. Their parameters are manually tuned in periodic setting to achieve the best performance: $H = 55I_n, \beta = 1$ for the proposed method, $c = 1, \rho = 50$ for [44,49], and $\eta = 0.06$ for [23,73]. The mixing matrix in [23,73] is selected with the Metropolis rule [105]. The primal and dual iterates of all the methods are initialized with 0. The triggering threshold for exchanging primal variables is set as $E^{(t)} = 0.9^{0.1t}$ for all the event-triggered methods. For [23], another triggering threshold is selected as $0.3^{0.1t}$ for the event-triggered dynamic average consensus scheme used to track the gradient. We evaluate the performance by considering the relative square error (RSE) defined by $\frac{\|\mathbf{x}^{(t)}-\mathbf{x}^*\|_F}{\|\mathbf{x}^{(0)}-\mathbf{x}^*\|_F}$ over the number of local iteration iteration number and communication times of the first agent.

The results are reported in Figure 6.3. All the methods exactly converge. However, the gradient-tracking method converges at a much slower rate than other two types of methods. This is primarily because this algorithm only allows one parameter to be tuned while other methods have two. The results also show that generally event-triggered methods converge at slower rates and present more oscillatory trajectories than their periodic counterparts, mainly due to the variable errors caused by event-triggered communication. However, significant reductions in network utilization are observed in event-triggered methods. In particular, the proposed method and COLA save $\sim \frac{1}{2}$ communication cost to achieve an accuracy of 10^{-4}. The gradient-tracking method consumes much heavier communication cost since both the estimated gradient

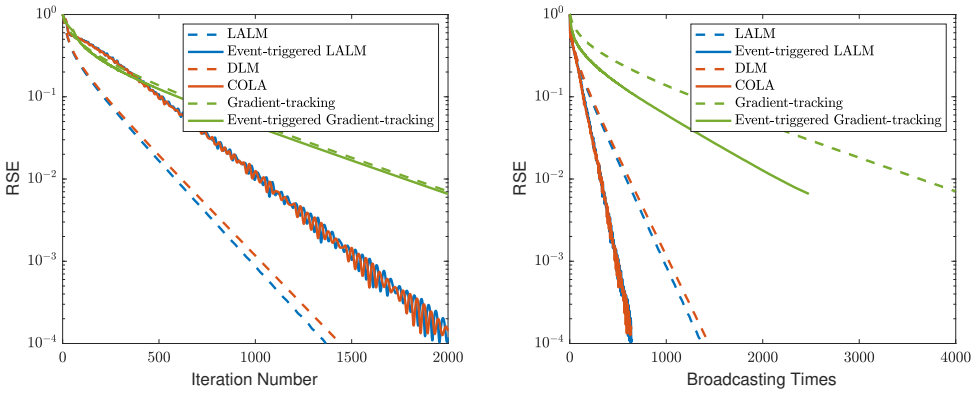


Figure 6.3: RSE versus iteration number and broadcasting times when $r = 0.04$.

and the local decision variable have to be exchanged.

6.6 Conclusion

This chapter has investigated the communication-efficient decentralized optimization problem. Based on the primal-dual formulation and LALM, we have designed a new event-triggered decentralized optimization algorithm, where each agent is allowed to communicate with its neighbors sporadically. We have proved the rates of convergence for the proposed algorithm under different problem settings. Numerical experiments have demonstrated the capability of the proposed method in reducing the utilization of network resources.

Chapter 7

Conclusion and Future Directions

This dissertation considers four concrete settings in decentralized optimization, that is, constraint-coupled and cost-coupled decentralized optimization in fixed networks (Chapters 3 and 5), composite cost-coupled decentralized optimization in stochastic networks (Chapter 4), and communication-efficient decentralized optimization (Chapter 6). We design four new algorithms for solving them, respectively, and rigorously analyze their rates of convergence.

7.1 Conclusions

Chapter 3 has addressed the non-smooth constraint-coupled decentralized optimization problem. We have leveraged Lagrangian relaxation to transform the coupling in constraints into that in objective function of the dual problem. For dual Lagrangian problems, most decentralized optimization algorithms cannot generate a convergent sequence of dual iterates and therefore are not directly applicable. To solve this issue, we have proposed the DSA_2 algorithm that guarantees the convergence of the local last iterate. We have proved that the dual objective error and the quadratic penalty for the violation of coupled constraints converge at rate $\mathcal{O}(1/\sqrt{t})$, and the primal objective error asymptotically vanishes.

Chapter 4 has investigated the decentralized composite optimization problem in stochastic networks. Most existing approaches cannot exploit the composite structure when the communication network is stochastic and thus converge only sublinearly. To tackle this challenging problem, we have designed a novel dynamic consensus protocol and a new DDA algorithm. Under a rather mild condition on the stochastic network,

our algorithm enjoys an $\mathcal{O}(1/t)$ rate of convergence in the general case and a global linear rate of convergence if each local objective function is strongly convex. To the best of our knowledge, this is the first algorithm that attains linear convergence for solving decentralized composite optimization in stochastic networks. Numerical results have been presented to support our design and analysis.

Chapter 5 has studied the accelerated decentralized constrained optimization problem. We have developed the ADDA algorithm, where the extrapolation technique together with the average consensus protocol is used to achieve acceleration over a decentralized network. Particularly, i) each agent uses the conventional first-order dynamic average consensus method to estimate the average of local gradients. ii) After deriving a local dual variable based on the estimates, each agent further generates a primal variable via solving the convex conjugate of a 1-strongly convex function over this dual variable. iii) Taking such a primal variable as an input, two additional sequences of primal variables are constructed based on the average consensus protocol. Let β be the second largest singular value of the mixing matrix, we have proved an $\mathcal{O}\left(\frac{1}{t^2}+\frac{1}{t(1-\beta)^2}\right)$ rate of convergence for ADDA, provided that each objective function is smooth. Numerical results have been presented to demonstrate the efficiency of the proposed methods.

Chapter 6 has tackled the communication-efficient decentralized optimization problem. For general composite objectives, we have designed an event-triggered decentralized primal-dual algorithm that only requires peer-to-peer communication at sporadic triggering time instants. The event-triggered broadcasting strategy is implemented by locally comparing the difference between true and broadcast variables with time-varying triggering thresholds. We have proved an $\mathcal{O}(1/t)$ rate of convergence in the general case provided that the threshold is summable over time, and a linear rate of convergence if the objective function is strongly convex and smooth, and the triggering threshold geometrically decreases. Numerical comparison results have been reported to highlight its performance and superiority in exploiting communication resources.

7.2 Future Work

7.2.1 Privacy-Preserving and Resilient Decentralized Optimization

The distributed nature of multi-agent optimization renders the system vulnerable to various network-induced issues such as eavesdropping and malicious cyber attacks. However, most algorithms assume the agents and the communication channels between agents to be completely trustworthy. This is rarely the case in practice. For example, an attacker can intrude a sub-system operated by the agents, and deliberately edit the message to be shared, i.e., deception attacks. This may result in an unstable system with possible damages to hardware and the system overall. To tackle this practical issue systematically, the techniques from robust statistics [90] and graph augmentation [99] may be incorporated into the decentralized dual averaging framework in Chapter 4.

7.2.2 Dual Averaging Methods for Decentralized Online Optimization

In various areas, e.g., scheduling of energy systems, the environment is highly dynamic and difficult to model. Therefore, the cost function to be minimized changes with time, and its value is observed only in hindsight. This is referred to as the online optimization problem in the literature [71], where the goal is to minimize the following *regret* function

$$\text{Regret}_T(u) := \sum_{t=1}^{T} f^{(t)}(x^{(t)}) - \sum_{t=1}^{T} f^{(t)}(u)$$

with respect to any u. For this type of problems, the algorithms in this dissertation cannot be directly applied. Interestingly, the online version dual averaging method, that is, follow-the-regularized-leader, is a powerful and generic algorithm to do online convex optimization. Motivated by this, future works will be devoted to the extensions of the algorithms in Chapters 4 and 5 to online optimization [120].

7.2.3 Rate Analysis of DDA Methods Under Error Bound Conditions

Lately, some researchers discerned the linear convergence of a class of first-order algorithms, e.g., the proximal gradient method (PGM) and the randomized block coordinate PGM, for convex [28] and nonconvex [97] optimization problems based on error bound conditions. Note that the strong convexity of f_i assumed in Chapter 4 is stronger than the metric subregularity of the subdifferential ∂f_i. An interesting direction of future work is whether or not the linear convergence of dual averaging methods can be established for the following two general classes of problems under similar conditions: i) Each $f_i : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a function of the form

$$f_i(x) = q_i(Ax) + \langle s_i, x \rangle$$

where A is some $l \times m$ matrix, s_i is some vector in $\mathbb{R}^m$, and $q_i : \mathbb{R}^l \to \mathbb{R} \cup \{+\infty\}$ is strongly convex and smooth on any convex compact subset of $\mathrm{dom}(q_i)$. ii) Each f_i is nonconvex.

Appendix A

Publications

- **Journal Papers**

  1. **C. Liu**, H. Li, and Y. Shi. Resource-aware exact decentralized optimization using event-triggered broadcasting. *IEEE Transactions on Automatic Control*, 66(7): 2961-2974, 2021. (Full Paper)
     (This work is presented in Chapter 6.)
  2. **C. Liu**, H. Li, and Y. Shi. A unitary distributed subgradient method for multi-agent optimization with different coupling sources. *Automatica*, 114, Paper ID: 108834, 2020. (Regular Paper)
     (This work is presented in Chapter 3.)
  3. **C. Liu**, H. Li, Y. Shi, and D. Xu. Distributed event-triggered gradient method for constrained convex minimization. *IEEE Transactions on Automatic Control*, 65(2): 778-785, 2020. (Technical Note)
  4. D. Ji, J. Ren, **C. Liu**, and Y. Shi. Stabilizing terminal constraint-free nonlinear MPC via sliding mode-based terminal cost. *Automatica*, 2021. (Accepted as Regular Paper)

- **Journal Paper Under Review**

  1. **C. Liu**, Y. Shi, H. Li, and W. Du. Accelerated dual averaging methods for decentralized constrained optimization. Submitted.
     (This work is presented in Chapter 5.)

- **Conference Papers**

  1. **C. Liu**, H. Li, and Y. Shi. Towards an $\mathcal{O}(1/t)$ convergence rate for distributed dual averaging, in *Proceedings of the 21st IFAC World Congress*, Berlin, Germany, July 12-17, 2020.
  2. **C. Liu**, H. Li, Y. Shi, and D. Xu. Event-triggered broadcasting for distributed smooth optimization, in *Proceedings of the 58th IEEE Conference on Decision and Control*, Nice, France, December 11-13, 2019.
  3. **C. Liu**, H. Li, and Y. Shi. Distributed dual subgradient method with double averaging: Application to QoS optimization in wireless networks, in *Proceedings of the 28th IEEE International Symposium on Industrial Electronics*, Vancouver, Canada, June 12-14, 2019.
  4. K. Zhang, **C. Liu**, and Y. Shi. Computationally efficient adaptive model predictive control for constrained linear system with parametric uncertainties, in *Proceedings of the 28th IEEE International Symposium on Industrial Electronics*, Vancouver, Canada, June 12-14, 2019.

- **Conference Paper Under Review**

  1. **C. Liu**, Z. Zhou, J. Pei, Y. Zhang, and Y. Shi. Decentralized composite optimization in stochastic networks: A dual averaging approach with linear convergence. Submitted.
     (This work is presented in Chapter 4.)

Bibliography

[1] 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.

[2] Karl Johan Astrom and Bo M Bernhardsson. Comparison of riemann and lebesgue sampling for first order stochastic systems. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, volume 2, pages 2011–2016. IEEE, 2002.

[3] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[4] Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52(6):2508–2530, 2006.

[5] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Machine Learning*, 3(1):1–122, 2010.

[6] Tsung-Hui Chang, Mingyi Hong, Wei-Cheng Liao, and Xiangfeng Wang. Asynchronous distributed admm for large-scale optimization—part i: Algorithm and convergence analysis. *IEEE Transactions on Signal Processing*, 64(12):3118–3130, 2016.

[7] Nikolaos Chatzipanagiotis, Darinka Dentcheva, and Michael M Zavlanos. An augmented lagrangian method for distributed optimization. *Mathematical Programming*, 152(1-2):405–434, 2015.

[8] Nikolaos Chatzipanagiotis and Michael M Zavlanos. On the convergence of a distributed augmented lagrangian method for nonconvex optimization. *IEEE Transactions on Automatic Control*, 62(9):4405–4420, 2017.

[9] Tianyi Chen, Georgios Giannakis, Tao Sun, and Wotao Yin. Lag: Lazily aggregated gradient for communication-efficient distributed learning. In *Advances in Neural Information Processing Systems*, pages 5050–5060, 2018.

[10] Weisheng Chen and Wei Ren. Event-triggered zero-gradient-sum distributed consensus optimization over directed networks. *Automatica*, 65:90–97, 2016.

[11] Michael Cohen, Jelena Diakonikolas, and Lorenzo Orecchia. On acceleration with noise-corrupted gradients. In *International Conference on Machine Learning*, pages 1019–1028, 2018.

[12] Igor Colin, Aurelien Bellet, Joseph Salmon, and Stéphan Clémençon. Gossip dual averaging for decentralized optimization of pairwise functions. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1388–1396, New York, New York, USA, 20–22 Jun 2016. PMLR.

[13] Laurent Condat. Fast projection onto the simplex and the l1 ball. *Mathematical Programming*, 158(1-2):575–585, 2016.

[14] Antonio J Conejo, Enrique Castillo, Roberto Minguez, and Raquel Garcia-Bertrand. *Decomposition techniques in mathematical programming: engineering and science applications*. Springer Science & Business Media, 2006.

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

[16] Thinh T Doan, Siva Theja Maguluri, and Justin Romberg. Convergence rates of distributed gradient methods under random quantization: A stochastic approximation approach. *IEEE Transactions on Automatic Control*, 2020.

[17] Dheeru Dua and Casey Graff. UCI Machine Learning Repository, 2017.

[18] John C Duchi, Alekh Agarwal, and Martin J Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic control*, 57(3):592–606, 2011.

[19] Alessandro Falsone, Kostas Margellos, Simone Garatti, and Maria Prandini. Dual decomposition for multi-agent distributed optimization with coupling constraints. *Automatica*, 84:149–158, 2017.

[20] Alessandro Falsone, Ivano Notarnicola, Giuseppe Notarstefano, and Maria Prandini. Tracking-admm for distributed constraint-coupled optimization. *Automatica*, 117:108962, 2020.

[21] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1, 2014.

[22] Allan Gut. *Probability: A Graduate Course*, volume 75. Springer Science & Business Media, 2013.

[23] Naoki Hayashi, Tomohiro Sugiura, Yuichi Kajiyama, and Shigemasa Takai. Event-triggered consensus-based optimization algorithm for smooth and strongly convex cost functions. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 2120–2125. IEEE, 2018.

[24] Dušan Jakovetić. A unification and generalization of exact distributed first-order methods. *IEEE Transactions on Signal and Information Processing over Networks*, 5(1):31–46, 2018.

[25] Dušan Jakovetić, Dragana Bajović, Nataša Krejić, and Nataša Krklec Jerinkić. Distributed gradient methods with variable number of working nodes. *IEEE Transactions on Signal Processing*, 64(15):4080–4095, 2016.

[26] Dušan Jakovetić, Joao Xavier, and José MF Moura. Fast distributed gradient methods. *IEEE Transactions on Automatic Control*, 59(5):1131–1146, 2014.

[27] Dušan Jakovetić, Joao Manuel Freitas Xavier, and José MF Moura. Convergence rates of distributed nesterov-like gradient methods on random networks. *IEEE Transactions on Signal Processing*, 62(4):868–882, 2013.

[28] J Ye Jane, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. Variational analysis perspective on linear convergence of some first order methods for nonsmooth convex optimization problems. *Set-Valued and Variational Analysis*, pages 1–35, 2021.

[29] Anatoli Juditsky, Joon Kwon, and Éric Moulines. Unifying mirror descent and dual averaging. *arXiv preprint arXiv:1910.13742*, 2019.

[30] Yuichi Kajiyama, Naoki Hayashi, and Shigemasa Takai. Distributed subgradient method with edge-based event-triggered communication. *IEEE Transactions on Automatic Control*, 63(7):2248–2255, 2018.

[31] Soummya Kar and José MF Moura. Sensor networks with random links: Topology design for distributed consensus. *IEEE Transactions on Signal Processing*, 56(7):3315–3326, 2008.

[32] Soummya Kar and José MF Moura. Distributed consensus algorithms in sensor networks: Quantized data and random link failures. *IEEE Transactions on Signal Processing*, 58(3):1383–1400, 2009.

[33] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized sgd with changing topology and local updates. In *International Conference on Machine Learning*, pages 5381–5393. PMLR, 2020.

[34] 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.

[35] Guanghui Lan, Soomin Lee, and Yi Zhou. Communication-efficient algorithms for decentralized and stochastic optimization. *Mathematical Programming*, pages 1–48, 2018.

[36] Puya Latafat, Nikolaos M Freris, and Panagiotis Patrinos. A new randomized block-coordinate primal-dual proximal algorithm for distributed optimization. *IEEE Transactions on Automatic Control*, 64(10):4050–4065, 2019.

[37] Puya Latafat, Lorenzo Stella, and Panagiotis Patrinos. New primal-dual proximal algorithm for distributed optimization. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 1959–1964. IEEE, 2016.

[38] Soomin Lee, Angelia Nedić, and Maxim Raginsky. Coordinate dual averaging for decentralized online optimization with nonseparable global objectives. *IEEE Transactions on Control of Network Systems*, 5(1):34–44, 2016.

[39] Huan Li, Cong Fang, Wotao Yin, and Zhouchen Lin. A sharp convergence rate analysis for distributed accelerated gradient methods. *arXiv preprint arXiv:1810.01053*, 2018.

[40] Huan Li, Cong Fang, Wotao Yin, and Zhouchen Lin. Decentralized accelerated gradient methods with increasing penalty parameters. *IEEE Transactions on Signal Processing*, 68:4855–4870, 2020.

[41] Huaqing Li, Shuai Liu, Yeng Chai Soh, and Lihua Xie. Event-triggered communication and data rate constraint for distributed optimization of multiagent systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(11):1908–1919, 2017.

[42] 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 *11th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 14)*, pages 583–598, 2014.

[43] Tao Li, Minyue Fu, Lihua Xie, and Ji-Feng Zhang. Distributed consensus with limited communication data rate. *IEEE Transactions on Automatic Control*, 56(2):279–292, 2010.

[44] Weiyu Li, Yaohua Liu, Zhi Tian, and Qing Ling. Communication-censored linearized admm for decentralized consensus optimization. *IEEE Transactions on Signal and Information Processing over Networks*, 6:18–34, 2019.

[45] 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.

[46] 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.

[47] Shu Liang, George Yin, et al. Distributed smooth convex optimization with coupled constraints. *IEEE Transactions on Automatic Control*, 65(1):347–353, 2019.

[48] Shu Liang, George Yin, et al. Dual averaging push for distributed convex optimization over time-varying directed graph. *IEEE Transactions on Automatic Control*, 65(4):1785–1791, 2019.

[49] 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.

[50] Changxin Liu, Huiping Li, and Yang Shi. Towards an $\mathcal{O}(1/t)$ convergence rate for distributed dual averaging. *IFAC-PapersOnLine*, 53(2):3254–3259, 2020.

[51] Changxin Liu, Huiping Li, Yang Shi, and Demin Xu. Distributed event-triggered gradient method for constrained convex minimization. *IEEE Transactions on Automatic Control*, 65(2):778–785, 2019.

[52] Sijia Liu, Pin-Yu Chen, and Alfred O Hero. Accelerated distributed dual averaging over evolving networks of growing connectivity. *IEEE Transactions on Signal Processing*, 66(7):1845–1859, 2018.

[53] Xiaorui Liu, Yao Li, Rongrong Wang, Jiliang Tang, and Ming Yan. Linear convergent decentralized optimization with compression. *arXiv preprint arXiv:2007.00232*, 2020.

[54] Yaohua Liu, Wei Xu, Gang Wu, Zhi Tian, and Qing Ling. Communication-censored admm for decentralized consensus optimization. *IEEE Transactions on Signal Processing*, 67(10):2565–2579, 2019.

[55] Ilan Lobel and Asuman Ozdaglar. Distributed subgradient methods for convex optimization over random networks. *IEEE Transactions on Automatic Control*, 56(6):1291–1306, 2010.

[56] Haihao Lu, Robert M Freund, and Yurii Nesterov. Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018.

[57] 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.

[58] Xianghui Mao, Kun Yuan, Yubin Hu, Yuantao Gu, Ali H Sayed, and Wotao Yin. Walkman: A communication-efficient random-walk algorithm for decentralized optimization. *IEEE Transactions on Signal Processing*, 68:2513–2528, 2020.

[59] David Mateos-Núnez and Jorge Cortés. Distributed saddle-point subgradient algorithms with Laplacian averaging. *IEEE Transactions on Automatic Control*, 62(6):2720–2735, 2016.

[60] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

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

[62] Angelia Nedić, Alex Olshevsky, and Michael G Rabbat. Network topology and communication-computation tradeoffs in decentralized optimization. *Proceedings of the IEEE*, 106(5):953–976, 2018.

[63] 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.

[64] Angelia Nedić and Asuman Ozdaglar. Approximate primal solutions and rate analysis for dual subgradient methods. *SIAM Journal on Optimization*, 19(4):1757–1780, 2009.

[65] Angelia Nedic, Asuman Ozdaglar, and Pablo A Parrilo. Constrained consensus and optimization in multi-agent networks. *IEEE Transactions on Automatic Control*, 55(4):922–938, 2010.

[66] Arkadii Semenovich Nemirovsky and David Borisovich Yudin. Problem complexity and method efficiency in optimization. 1983.

[67] Yu Nesterov and Vladimir Shikhman. Quasi-monotone subgradient methods for nonsmooth convex minimization. *Journal of Optimization Theory and Applications*, 165(3):917–940, 2015.

[68] Yu Nesterov and Vladimir Shikhman. Dual subgradient method with averaging for optimal resource allocation. *European Journal of Operational Research*, 270(3):907–916, 2018.

[69] Yurii Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical programming*, 120(1):221–259, 2009.

[70] Ivano Notarnicola and Giuseppe Notarstefano. Constraint-coupled distributed optimization: a relaxation and duality approach. *IEEE Transactions on Control of Network Systems*, 7(1):483–492, 2019.

[71] Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

[72] Neal Parikh and Stephen Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):127–239, 2014.

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

[74] Guannan Qu and Na Li. Accelerated distributed nesterov gradient descent. *IEEE Transactions on Automatic Control*, 65(6):2566–2581, 2019.

[75] Michael Rabbat and Robert Nowak. Distributed optimization in sensor networks. In *Proceedings of the 3rd international symposium on Information processing in sensor networks*, pages 20–27, 2004.

[76] Robin L Raffard, Claire J Tomlin, and Stephen P Boyd. Distributed optimization for cooperative agents: Application to formation flight. In *2004 43rd IEEE Conference on Decision and Control (CDC)*, volume 3, pages 2453–2459. IEEE, 2004.

[77] Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *International Conference on Machine Learning*, pages 3027–3036. PMLR, 2017.

[78] Mark Schmidt, Glenn Fung, and Rmer Rosales. Fast optimization methods for l1 regularization: A comparative study and two new approaches. In *European Conference on Machine Learning*, pages 286–297. Springer, 2007.

[79] Mark Schmidt, Nicolas Le Roux, and Francis Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, pages 1458–1466, 2011.

[80] Jacob H Seidman, Mahyar Fazlyab, George J Pappas, and Victor M Preciado. A chebyshev-accelerated primal-dual method for distributed optimization. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 1775–1781. IEEE, 2018.

[81] Shahin Shahrampour and Ali Jadbabaie. Distributed online optimization in dynamic environments using mirror descent. *IEEE Transactions on Automatic Control*, 63(3):714–725, 2017.

[82] Thomas Sherson, Richard Heusdens, and W Bastiaan Kleijn. On the duality of globally constrained separable problems and its application to distributed signal processing. In *2016 24th European Signal Processing Conference (EUSIPCO)*, pages 1083–1087. IEEE, 2016.

[83] Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.

[84] 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.

[85] Wei Shi, Qing Ling, Kun Yuan, Gang Wu, and Wotao Yin. On the linear convergence of the admm in decentralized consensus optimization. *IEEE Transactions on Signal Processing*, 62(7):1750–1761, 2014.

[86] Andrea Simonetto and Hadi Jamali-Rad. Primal recovery from consensus-based dual decomposition for distributed convex optimization. *Journal of Optimization Theory and Applications*, 168(1):172–197, 2016.

[87] Navjot Singh, Deepesh Data, Jemin George, and Suhas Diggavi. Sparq-sgd: Event-triggered and compressed communication in decentralized optimization. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 3449–3456. IEEE, 2020.

[88] Paulo Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685, 2007.

[89] Konstantinos I Tsianos, Sean Lawlor, and Michael G Rabbat. Push-sum distributed dual averaging for convex optimization. In *2012 51st IEEE Conference on Decision and Control (CDC)*, pages 5453–5458. IEEE, 2012.

[90] Berkay Turan, Cesar Uribe, Hoi-To Wai, and Mahnoosh Alizadeh. Resilient primal-dual optimization algorithms for distributed resource allocation. *IEEE Transactions on Control of Network Systems*, 2020. doi: 10.1109/TCNS.2020.3024485.

[91] César A Uribe, Soomin Lee, Alexander Gasnikov, and Angelia Nedić. A dual approach for optimal algorithms in distributed optimization over networks. *Optimization Methods and Software*, pages 1–40, 2020.

[92] César A Uribe, Hoi-To Wai, and Mahnoosh Alizadeh. Resilient distributed optimization algorithms for resource allocation. In *2019 58th IEEE Conference on Decision and Control (CDC)*, pages 8341–8346. IEEE, 2019.

[93] Ewout van den Berg, MP Friedlander, G Hennenfent, F Herrmann, R Saab, and O Yılmaz. Sparco: A testing framework for sparse reconstruction. *Dept. Comput. Sci., Univ. British Columbia, Vancouver, Tech. Rep. TR-2007-20,[Online]. Available: http://www. cs. ubc. ca/labs/scl/sparco*, 2007.

[94] Damiano Varagnolo, Filippo Zanella, Angelo Cenedese, Gianluigi Pillonetto, and Luca Schenato. Newton-raphson consensus for distributed convex optimization. *IEEE Transactions on Automatic Control*, 61(4):994–1009, 2015.

[95] Robin Vujanic, Peyman Mohajerin Esfahani, Paul J Goulart, Sébastien Mariéthoz, and Manfred Morari. A decomposition method for large scale milps, with performance guarantees and a power system application. *Automatica*, 67:144–156, 2016.

[96] Hoi-To Wai, Jean Lafond, Anna Scaglione, and Eric Moulines. Decentralized frank–wolfe algorithm for convex and nonconvex problems. *IEEE Transactions on Automatic Control*, 62(11):5522–5537, 2017.

[97] Xiangfeng Wang, J Ye Jane, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. Perturbation techniques for convergence analysis of proximal gradient method and other first-order algorithms via variational analysis. *Set-Valued and Variational Analysis*, pages 1–41, 2021.

[98] Xiaofeng Wang and Michael D Lemmon. Event-triggering in distributed networked control systems. *IEEE Transactions on Automatic Control*, 56(3):586–601, 2010.

[99] Yongqiang Wang. Privacy-preserving average consensus via state decomposition. *IEEE Transactions on Automatic Control*, 64(11):4711–4716, 2019.

[100] Zheming Wang and Chong-Jin Ong. Accelerated distributed mpc of linear discrete-time systems with coupled constraints. *IEEE Transactions on Automatic Control*, 63(11):3838–3849, 2018.

[101] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[102] Ermin Wei and Asuman Ozdaglar. On the o (1= k) convergence of asynchronous distributed alternating direction method of multipliers. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 551–554. IEEE, 2013.

[103] Kenneth S Williams. The n th power of a 2× 2 matrix. *Mathematics Magazine*, 65(5):336–336, 1992.

[104] Tianyu Wu, Kun Yuan, Qing Ling, Wotao Yin, and Ali H Sayed. Decentralized consensus optimization with asynchrony and delays. *IEEE Transactions on Signal and Information Processing over Networks*, 4(2):293–307, 2017.

[105] Lin Xiao and Stephen Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

[106] Lin Xiao, Stephen Boyd, and Seung-Jean Kim. Distributed average consensus with least-mean-square deviation. *Journal of parallel and distributed computing*, 67(1):33–46, 2007.

[107] 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.

[108] Cuixia Xu, Junlong Zhu, Youlin Shang, and Qingtao Wu. A distributed conjugate gradient online learning method over networks. *Complexity*, 2020, 2020.

[109] 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.

[110] Jinming Xu, Ye Tian, Ying Sun, and Gesualdo Scutari. A unified algorithmic framework for distributed composite optimization. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 2309–2316. IEEE, 2020.

[111] Jinming Xu, Shanying Zhu, Yeng Chai Soh, and Lihua Xie. Convergence of asynchronous distributed gradient methods over stochastic networks. *IEEE Transactions on Automatic Control*, 63(2):434–448, 2017.

[112] Jinming Xu, Shanying Zhu, Yeng Chai Soh, and Lihua Xie. A bregman splitting scheme for distributed optimization over networks. *IEEE Transactions on Automatic Control*, 63(11):3809–3824, 2018.

[113] Jinming Xu, Shanying Zhu, Yeng Chai Soh, and Lihua Xie. A dual splitting approach for distributed resource allocation with regularization. *IEEE Transactions on Control of Network Systems*, 6(1):403–414, 2018.

[114] Yangyang Xu. Accelerated first-order primal-dual proximal methods for linearly constrained composite convex programming. *SIAM Journal on Optimization*, 27(3):1459–1484, 2017.

[115] Peng Yi and Yiguang Hong. Quantized subgradient algorithm and data-rate analysis for distributed optimization. *IEEE Transactions on Control of Network Systems*, 1(4):380–392, 2014.

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

[117] Jiaojiao Zhang, Qing Ling, and Anthony Man-Cho So. A newton tracking algorithm with exact linear convergence for decentralized consensus optimization. *IEEE Transactions on Signal and Information Processing over Networks*, 2021.

[118] Jiaqi Zhang, Keyou You, and Tamer Başar. Distributed discrete-time optimization in multiagent networks using only sign of relative state. *IEEE Transactions on Automatic Control*, 64(6):2352–2367, 2018.

[119] Jiaqi Zhang, Keyou You, and Kai Cai. Distributed dual gradient tracking for resource allocation in unbalanced networks. *IEEE Transactions on Signal Processing*, 68:2186–2198, 2020.

[120] Yan Zhang, Robert J Ravier, Michael M Zavlanos, and Vahid Tarokh. A distributed online convex optimization algorithm with improved dynamic regret. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 2449–2454. IEEE, 2019.

[121] Yan Zhang and Michael M Zavlanos. A consensus-based distributed augmented lagrangian method. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 1763–1768. IEEE, 2018.

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