

**DISTRIBUTED CONSENSUS WITH
DUAL DECOMPOSITION METHODS
UNDER ADDITIVE AND BOUNDED
ERRORS**

By

Hansi Kavindika Abeynanda

MPhil

2022

**DISTRIBUTED CONSENSUS WITH DUAL
DECOMPOSITION METHODS UNDER ADDITIVE
AND BOUNDED ERRORS**

By

Hansi Kavindika Abeynanda

Thesis submitted to the University of Sri Jayewardenepura
for the award of the Degree of Master of Philosophy

Declaration of the Candidate

The work described in this thesis was carried out by me under the supervision of Dr. G. H. J. Lanel and Dr. Chaturanga Weeraddana and a report on this has not been submitted in whole or in part to any university or any other institution for another Degree/Diploma.



.....
Hansi Kavindika Abeynanda

Date: 17/02/2023

Certification of the Supervisors

We certify that the candidate has incorporated all corrections, additions, and amendments recommended by the examiners to this version of the MPhil thesis.

.....

Dr. G. H. J. Lanel

Senior Lecturer

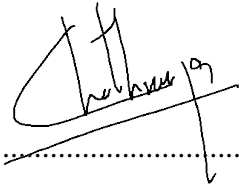
Department of Mathematics

Faculty of Applied Sciences

University of Sri Jayewardenepura

Sri Lanka

Date:



.....

Dr. Chaturanga Weeraddana

Senior Research Fellow

Center for Wireless Communications

University of Oulu

Finland

Date:18/02/2023.....

List of Contents

List of Contents	i
List of Figures	iii
Notation	vii
Acknowledgement	viii
Abstract	x
1 Introduction	1
1.1 Purpose of the Thesis	1
1.2 Motivation	4
1.3 Our Contribution	6
1.4 Outline of the Thesis	8
1.5 Related Theory	10
1.5.1 Mathematical Optimization	10
1.5.2 Convexity	13
1.5.3 Duality	17
1.6 Mathematical preliminaries	20
1.6.1 Basic Definitions	20
1.6.2 Basic Results	24
2 Literature Review	25
2.1 Distributed Optimization: The background	25
2.2 Distributed Optimization Methods	28
2.2.1 Decomposition Methods	28
2.2.2 The Subgradient Method	32

2.2.3	Alternating Direction Method of Multipliers (ADMM)	45
2.2.4	Proximal Gradient Method	50
2.2.5	Dual Averaging	52
2.2.6	Classification of Convergence Rates	53
2.3	Challenges	56
2.3.1	Distributed Optimization over Non-ideal Settings	57
3	Materials and Methods	61
3.1	Problem Formulation	61
3.2	Dual Decomposition Approach	62
3.3	Imperfect Coordination Between Subsystems	65
3.4	Distributed Algorithms over Non-ideal Settings	65
3.4.1	Partially Distributed Algorithm	66
3.4.2	Fully Distributed Algorithm	68
4	Results and Discussion	70
4.1	Analysis of the properties of the Dual Function	70
4.1.1	Dual Function as a Restriction of f^*	70
4.1.2	Lipschitzian Properties	71
4.1.3	Strong Convexity Properties	74
4.1.4	Bounding Properties for the Primal Error	76
4.2	Convergence Analysis: Global Consensus	78
4.2.1	Key Remarks, and Related Results	80
4.2.2	Convergence Analysis: CASE 1	82
4.2.3	Convergence Analysis: CASE 2	90
4.2.4	Feasible Points from Algorithm 5 and Algorithm 6	101
4.3	Convergence Analysis: General Consensus	106
4.3.1	Generalized Problem Formulation	107
4.3.2	Generalized Results	111

4.4	Numerical Results	114
4.4.1	CASE 1: Strongly Convex Local Objectives at Subsystems	115
4.4.2	CASE 2: Strongly convex and gradient Lipschitz Continuous local objectives at subsystems	125
5	Conclusions and Recommendations	133
5.1	Conclusion	133
5.2	Future Work	134

References

Appendix 1: List of Publications

List of Figures

1.1	An example of a power grid.	2
1.2	The probability $p = \exp(a^T x + b)/1 + \exp(a^T x + b)$ with $x \in \mathbb{R}$, $a = 2$, and $b = -5$	5
1.3	Graphical interpretation of convex and nonconvex sets. (a) The ellipse is a convex set. (b) The kidney shaped set is not convex, since the line segment between the given two points is not contained in the set.	13
1.4	Classification of convex functions: (a) Convex function: The line segment between any two points on the graph lies on or above the graph. (b) Strictly convex function: The line segment between any two points on the graph lies above the graph. (c) Strongly convex function: The graph is always lower bounded by a convex quadratic function drawn to the graph at any point in the domain.	15
2.1	Distributed networked system.	27
2.2	Standard gradient method with primal decomposition: Convergence of $y^{(k)}$ using different fixed step sizes.	40
2.3	Standard gradient method with primal decomposition: Convergence of $y^{(k)}$ using different dimensions of y	40
2.4	Standard gradient method with primal decomposition: Convergence of $S(y^{(k)})$ using different dimensions of y	41
2.5	The subgradient method with dual decomposition: The graph of the dual function $g(\lambda)$ corresponding to the primal problem (2.31).	44
2.6	The subgradient method with dual decomposition: Convergence of dual function values using different fixed step sizes.	45

2.7	The subgradient method with dual decomposition: Convergence of dual function values using constant, nonsummable diminishing, and square summable but not summable step size rules.	46
2.8	The subgradient method with dual decomposition: Convergence of dual variable iterates using different fixed step sizes.	47
2.9	The subgradient method with dual decomposition: Convergence of dual variable iterates using different step size rules.	48
2.10	The subgradient method with dual decomposition: Convergence of dual function values and primal function values evaluated at feasible points. Solid line demonstrates the optimal value f^* of the primal problem (2.31).	49
3.1	Decomposition Structure: There are m subsystems with the public variable \mathbf{z} . Functions associated with subsystems are $f_i(\mathbf{z}), i \in \{1, \dots, m\}$. . .	62
3.2	Graph of the Communication Structure: Partially Distributed Algorithm . . .	66
3.3	Graph of the Communication Structure: Fully Distributed Algorithm . . .	68
4.1	Graphs of $(k + p)^p$ and $p(k + 1)$ with $p = \log k/k$. Figure clearly shows $p(k + 1) > (k + p)^p$ for large k	98
4.2	Decomposition structure: There are five subsystems and three nets (i.e., $q = 3$) with the public variable $\mathbf{z} = [\mathbf{z}_1^T \ \mathbf{z}_2^T \ \mathbf{z}_3^T]^T$. Net variables are $\mathbf{z}_1, \mathbf{z}_2,$ and \mathbf{z}_3 . Functions associated with subsystems are $f_1(\mathbf{z}_1), f_2(\mathbf{z}_1, \mathbf{z}_2), f_3(\mathbf{z}_2, \mathbf{z}_3), f_4(\mathbf{z}_3),$ and $f_5(\mathbf{z}_3)$. The sets $\mathcal{M}_1 = \mathcal{M}_2 = \{1, 2\},$ and $\mathcal{M}_3 = \{1, 2, 3\}$	108
4.3	The quantization scheme in CASE 1 with $n = 2$ and $b = 2$. The constraint set \mathcal{Y} is a box of width 6 per dimension. The box is partitioned into identical 16 mini-boxes of width $t = 1.5$. The exact solution $\mathbf{y}_i^{(k)}$ of subsystem i is given in blue. The distorted vector $\hat{\mathbf{y}}_i^{(k)}$ which is given in red, is chosen to be the centroid of the respective mini-box.	116

4.4	CASE 1: Convergence of minimal norm gradients of the negative dual function h . The figure shows the effect of choice of p in the step size $\gamma_k = (1/L_h)/(k + 1)^p$ on the convergence, by fixing $b = 5$	117
4.5	CASE 1: Convergence of minimal norm gradients of the negative dual function h . The figure shows the effect of choice of b on the convergence, by fixing $\gamma_k = 1/L_h$	117
4.6	CASE 1: Convergence of minimal norm primal feasible points of the problem (3.2). The figure shows the effect of the choice of p in the step size $\gamma_k = (1/L_h)/(k + 1)^p$ on the convergence by fixing $b = 5$	118
4.7	CASE 1: Convergence of minimal norm primal feasible points of the problem (3.2). The figure shows the effect of the choice of b on the convergence by fixing $\gamma_k = 1/L_h$	118
4.8	CASE 1: The effect of dimension n of the local variables y_i s on SO using fixed step size rule $\gamma_k = 1/L_h$ for different bits.	119
4.9	CASE 1: The effect of number of users m on SO using fixed step size rule $\gamma_k = 1/L_h$ for different bits.	120
4.10	CASE 1: The effect of dimension n of the local variables y_i s on SO using nonsummable step size rule $\gamma_k = 1/(L_h k^{0.1})$ for different bits.	120
4.11	CASE 1: The effect of number of users m on SO using nonsummable step size rule $\gamma_k = 1/(L_h k^{0.1})$ for different bits.	121
4.12	CASE 1: The trade-offs between b and SO for different dimensions n using fixed step size rule $\gamma_k = 1/L_h$	122
4.13	CASE 1: The trade-offs between b and SO for different users m using fixed step size rule $\gamma_k = 1/L_h$	122
4.14	CASE 1: The trade-offs between b and SO for different dimensions n using nonsummable step size rule $\gamma_k = \gamma_0/k$, where γ_0 is chosen suitably.	123
4.15	CASE 1: The trade-offs between b and SO for different users m using nonsummable step size rule $\gamma_k = \gamma_0/k$, where γ_0 is chosen suitably.	123

4.16 CASE 2: Convergence of dual function iterates using constant and non-summable step sizes.	124
4.17 CASE 2: Effect of choice of ς on the convergence of dual function iterates.	125
4.18 CASE 2: Convergence of primal feasible points using constant and non-summable step sizes.	126
4.19 CASE 2: Effect of choice of ς on the convergence of primal feasible points.	126
4.20 CASE 2: The effect of dimension n of the local variables y_i s on SO using fixed step size rule $\gamma_k = 1/L_h$ for different ς	127
4.21 CASE 2: The effect of number of users m on SO using fixed step size rule $\gamma_k = 0.01$ for different ς	127
4.22 CASE 2: The effect of dimension n of the local variables y_i s on SO using nonsummable step size rule $\gamma_k = \gamma_0/k$ for different ς	128
4.23 CASE 2: The effect of number of users m on SO using nonsummable step size rule $\gamma_k = \gamma_0/k$ for different ς	128
4.24 CASE 2: The effect of dimension n of the local variables y_i s on SO using fixed step size rule $\gamma_k = 1/L_h$ for different ς	129
4.25 CASE 2: The effect of number of users m on SO using fixed step size rule $\gamma_k = 0.01$ for different bits.	130
4.26 CASE 2: The effect of dimension n of the local variables y_i s on SO using nonsummable step size rule $\gamma_k = \gamma_0/k$ for different ς	130
4.27 CASE 2: The effect of number of users m on SO using nonsummable step size rule $\gamma_k = \gamma_0/k$ for different ς	131

Notation

$\overline{\mathbb{R}}$	Set of extended real numbers
\mathbb{R}	Set of real numbers
\mathbb{R}^n	Set of real n -vectors
$\mathbb{R}^{m \times n}$	Set of real $m \times n$ matrices
\mathbb{R}^+	Set of positive real numbers
\mathbb{Z}_+	Set of positive integers
\mathbb{Z}_+^0	Set of nonnegative integers
∂f	Subdifferential of the function f
∇f	Gradient of the function f
$f : \mathbb{R}^n \rightarrow \mathbb{R}$	f is a real valued function defined on some subset of \mathbb{R}^n , specifically the domain of f , which we denote by $\text{dom } f$.
\exists	There exists
$ \cdot $	Absolute value
$\ \cdot\ $	ℓ_2 -norm
<i>s.t.</i>	Such that
<i>e.g.</i>	Example
<i>cf.</i>	Compare with
$\text{int } \mathcal{C}$	Interior of the set \mathcal{C}
$\text{relint } \mathcal{C}$	Relative interior of the set \mathcal{C}
\setminus	Set minus
\forall	For all
$[\mathbf{x}]_{\mathcal{X}}$	Projection of $\mathbf{x} \in \mathbb{R}^n$ on to the set $\mathcal{X} \subseteq \mathbb{R}^n$
\mathbf{A}_{ij}	The element corresponding to the i th row and j th column of matrix \mathbf{A}
\mathbb{S}_{++}^n	The set of symmetric positive definite $n \times n$ matrices
$\mathbf{1}_n$	$n \times n$ identity matrix
$\mathbf{1}_{n \times m}$	$n \times m$ matrix with all entries equal to one
$\mathbf{A} \otimes \mathbf{B}$	The Kronecker product of matrices \mathbf{A} and \mathbf{B}
$[a, b]$	Closed interval in real line
$\lambda_{\max}(\mathbf{A})$	Highest eigenvalue of \mathbf{A}
$\lambda_{\min}(\mathbf{A})$	Smallest non-negative eigenvalue of \mathbf{A}

Acknowledgements

Foremost, I would like to express my sincere gratitude to my supervisors Dr. G. H. J. Lanel, Department of Mathematics, Faculty of Applied Sciences, University of Sri Jayewardenepura, and Dr. Chathuranga Weeraddana, Center for Wireless Communications, University of Oulu, Finland, for their guidance, encouragement, and ceaseless support throughout my research study. I dedicate my special thanks to Dr. G. H. J. Lanel for his continual help and fruitful discussions during the weekly research progress meetings, which were truly beneficial for enriching my research work.

I owe my deepest thank to Dr. Chathuranga Weeraddana for his tremendous support and invaluable guidance throughout the time of my research. I am deeply grateful to you for your comments, advice, and encouragement in the preparation of the publications related to this research study. Without your kind support and guidance, I would not have come this far.

I also want to thank my colleagues Nuwan Jayarathne and Kasuni Welihinda for their comments during the time of the preparation of my MPhil thesis, and to the Faculty of Graduate Studies of the University of Sri Jayewardenepura and Dr. Kaushika De Silva for their support given during the process of thesis submission.

I want to express my heartfelt gratitude to my loving parents for their love and for always being with me by my side all the time in my life. I would like to thank my loving brothers Tharaka, Shanaka, and their families, and my loving parents-in-law and their family for the love and support they have given throughout these years.

Finally, I want to thank my loving husband Maduka, and my loving daughter Nathara for your love, patience, understanding, support, encouragement, and always being with me throughout my life during all the ups and downs. Without you, none of these would have been possible.

Distributed Consensus with Dual Decomposition Methods Under Additive and Bounded Errors

Hansi Kavindika Abeynanda

ABSTRACT

With the unprecedented growth of signal processing and machine learning application domains, there has been a tremendous expansion of interest in distributed optimization methods to cope with the underlying large-scale problems. The distributed optimization methods allow large-scale systems that consist of many subsystems to solve a global problem interactively via communication. In particular, many considerable advantages such as the high fault-tolerance, scalability, less communication cost, solution speed, and data privacy have invoked the application of distributed optimization methods in many large-scale optimization problems. Nonetheless, inevitable system-specific challenges such as limited computational power, limited communication, latency requirements, measurement errors, and noises in wireless channels impose restrictions on the exactness of the underlying distributed algorithms. Such restrictions have appealed to the exploration of algorithms' convergence behaviors under inexact settings. Thus, the main purpose of this thesis is to analyze the convergence properties of distributed optimization method under non-ideal settings. Moreover, we provide a systematic exposition on state-of-the-art distributed optimization methods that cope with large-scale distributed problems.

Our main focus in this research lies in the inexactness of *dual decomposition methods* for distributed optimization. However, if such an inexactness is modeled as if it stems from the dual-domain, investigating how it might evolve into the primal-domain and how it might influence the primal optimality, and more importantly, how it affects the optimality of a *feasible point* return by the underlying machinery is utmost important. Howbeit, it seems that the analysis of convergences of dual decomposition methods concerning primal optimality and primal feasibility, together with dual optimality is less investigated in the literature. Therefore, it is desirable to have an exposition that lays

out the consequences of inexactness on the convergence properties of the primal-domain, together with the convergence properties in the dual-domain. Motivated by this, here we provide a systematic exposition on the convergence of dual decomposition methods under inexact settings, for an important class of constrained global variable consensus optimization problems. Convergences and the rate of convergences of the algorithms are mathematically substantiated, not only from a dual-domain standpoint but also from a primal-domain standpoint. More importantly, we provide the convergence results under two cases, CASE 1 and CASE 2. The convergence results using strongly convex local objective functions are established under CASE 1, and the convergence results using both strongly convex and gradient Lipschitz continuous local objective functions are asserted under CASE 2. In particular, all the theoretical results under both scenarios are established using two step size rules, the constant step size rule, and the nonsummable step size rule. Our analytical results show that the algorithms get into a neighborhood of optimality in both dual and primal domains, the size of which depends on the level of underlying distortions. In further, an elaboration of a generalized problem formulation, which is known as a *general consensus problem* is also furnished, together with the related convergence properties of underlying algorithms. Finally, the theoretical derivations are verified by numerical experiments.

Keywords: Distributed optimization, Consensus problem, Dual decomposition, Imperfect coordination, Bounded errors, Primal feasibility

Chapter 1

Introduction

1.1 Purpose of the Thesis

Our world is a consequence of an interplay between a large number of networked systems with huge data volumes. The most common large-scaled networked systems in our world include the internet, wireless networks (e.g., cell phone networks and satellite communication networks), and power grids (See Figure 1.1). These important infrastructures consist of many subsystems which make local decisions and coordinate information to accomplish their tasks. During this process, distributed optimization plays an important role, which enables a system to solve a global optimization problem interactively (*cf.* Chapter 2.1).

Distributed optimization is commonly used in many application fields such as signal processing, machine learning, telecommunication networks, control systems, robotics, and network applications, among others, [1–10]. Nonetheless, these underlying systems face many system-specific challenges such as limited computational power, limited communication, latency requirements, measurement errors, and noises in wireless channels, which impose restrictions on the exactness of the underlying distributed optimization methods. Thus, the analysis of distributed algorithms over non-ideal settings has been an appealing area of study [11–28]. However, the analysis of dual decomposition methods over non-ideal settings, concerning the convergences of primal feasibility points is less investigated in the literature. Motivated by this, the main focus of this study lies around the analysis of dual decomposition methods under non-ideal settings together with a systematic exposition of state-of-the-art distributed optimization methods. More specifically, the main objectives of our study are as follows:

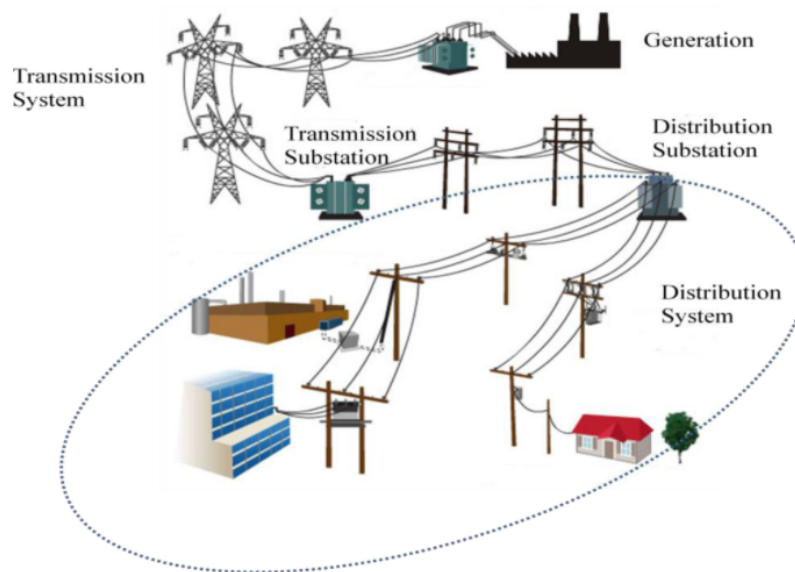


Figure 1.1: An example of a power grid.

1. Analyze the state-of-the-art distributed methods that are crucial in distributed optimization settings (*cf.* Chapter 2).
 - The contribution towards this objective is presented in Chapter 2. In particular, we analyse the subgradient methods, Alternating Direction Method of Multipliers, proximal gradient method, and dual averaging methods.
2. Investigate the fundamental theories associated with the distributed optimization techniques (*cf.* Chapter 1.5).
 - The basic theory related to convexity and duality, on which this thesis is built is presented in Chapter 1.5.
3. Design distributed algorithms based on dual decomposition methods under numerous non-ideal settings (*cf.* Chapter 3.4).
 - Two distributed algorithms (Algorithm 5 and Algorithm 6) modeled under non-ideal settings are proposed in Chapter 3.4. In particular, Algorithm 5 is a partially distributed algorithm while Algorithm 6 is being a fully distributed algorithm.
4. Analyze the convergence properties of designed distributed algorithms (*cf.* Chapter 4.1, Chapter 4.2, and Chapter 4.3.).

- Our main contribution towards this thesis lies around this objective. Related findings are presented in Chapter 4.1, Chapter 4.2, and Chapter 4.3. More specifically, an analysis on the properties of the dual function is presented in Chapter 4.1, Convergence properties of proposed algorithms based on the global consensus problem [*cf.* problem (3.1)] is presented in Chapter 4.2, and related convergence results based on the general consensus problem (*cf.* problem (4.133)) are presented in Chapter 4.3.

5. Test the validity of theoretical results by simulations (*cf.* Chapter 4.4).

- Theoretical assertions presented in Chapter 4.2 based on the global consensus problem are numerically tested in Chapter 4.4.

Our primary focus of this study is drawn to address a global consensus optimization problem (*cf.* Chapter 3.1), which is frequently applied in many large-scale networked systems that require distributed solution techniques [29].

A global consensus problem has the form

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^m f_i(\mathbf{x}) \\ & \text{subject to} && \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{1.1}$$

with the variable $\mathbf{x} \in \mathbb{R}^n$ and $\mathcal{X} \subseteq \mathbb{R}^n$, which is considered as a common constraint set. It is worth emphasizing that a general formulation to the problem (1.1) arises when the functions f_i s depend only on a part of the variable \mathbf{x} . This particular general problem structure is known as the general consensus problem [*cf.* problem (4.133)]. We also address this general problem formulation in Chapter 4.3 and related theoretical results are discussed.

Our proposed distributed algorithms are based on dual decomposition techniques and are modeled to capture a wide range of distortions. Convergences of the algorithms are extensively analyzed in both primal and dual domains together with their rates of convergences. More importantly, convergence properties of primal feasible points are also

theoretically substantiated which is of utmost importance in both analytical and practical perspectives.

1.2 Motivation

The global consensus optimization problems of the form (1.1) play a crucial role in many application domains. Some important examples include distributed averaging [30, 31], power system control [32, 33], decentralized decision making and computation [34] and distributed machine learning [35, 36]. Among these, machine learning has placed greater importance in many applications such as medical diagnosis, image recognition, speech recognition, social media, transportation, and many others. Here we present an important example in medical diagnosis. Identifying whether an individual in a certain population is having particular cancer is an important application in machine learning under medical diagnosis. For this purpose, we use previous data from a fair amount of suitably chosen training samples to model the probability p , that a randomly selected individual has cancer. The probability p is modeled using the *logistic model* which has the form (cf. [37, Section 7])

$$p = \frac{\exp(\mathbf{a}^T \mathbf{x} + b)}{1 + \exp(\mathbf{a}^T \mathbf{x} + b)}, \quad (1.2)$$

where $\mathbf{x} \in \mathbb{R}^n$ is called the explanatory variable, which represents a medically relevant variable (see Figure 1.2). The variables $\mathbf{a} \in \mathbb{R}^n$ and $b \in \mathbb{R}$ are called the model parameters. In particular, the explanatory variable \mathbf{x} characterizes the factors affecting cancer (for example, \mathbf{x} can represent expression levels of an individual's genes). Suppose that the historical data for the variable x from different individuals are available from m randomly chosen hospitals. Let $u \in \{0, 1\}$ represents a random variable, where $u = 1$ denotes that an individual in a particular hospital is having cancer and $u = 0$ denotes that an individual is not having cancer. Moreover, suppose that the explanatory variables $\mathbf{x}_{l1}, \dots, \mathbf{x}_{lq_l} \in \mathbb{R}^n$ are available from a set of q_l individuals in the l th hospital, where $l = 1, \dots, m$, along with the corresponding outcomes $u_{l1}, \dots, u_{lq_l} \in \{0, 1\}$. Moreover, we assume that there

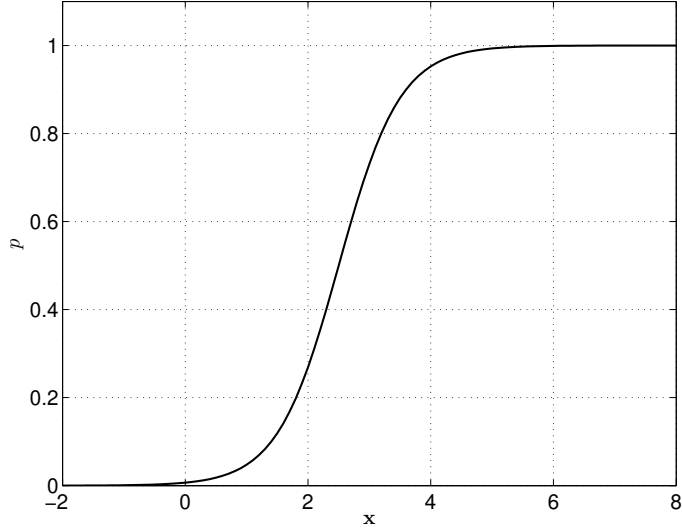


Figure 1.2: The probability $p = \exp(a^T x + b) / (1 + \exp(a^T x + b))$ with $x \in \mathbb{R}$, $a = 2$, and $b = -5$.

are r_l individuals in the l th hospital who have cancer, where $r_l \leq q_l$. Without loss of generality, we consider that $\mathbf{x}_{l1}, \dots, \mathbf{x}_{lr_l}$ are the explanatory variables that belong to the individuals who have cancer. Thus, we have $u_{li} = 1$ for all $i = 1, \dots, r_l$, and $u_{li} = 0$ for all $i = r_l + 1, \dots, q_l$. Then, we can find a maximum likelihood estimate of the model parameters a and b . Let p_{li} denote the probability determined by the logistic model (1.2) using $\mathbf{x}_{li} \in \mathbb{R}^n$ that belongs to the i th individual of the l th hospital, where $i = 1, \dots, q_l$ and $l = 1, \dots, m$. Then the likelihood function used only with the l th hospital has the form

$$g_l(\mathbf{a}, b) = \prod_{i=1}^{r_l} p_{li} \prod_{i=r_l+1}^{q_l} (1 - p_{li}). \quad (1.3)$$

The log likelihood function then given by

$$f_l(\mathbf{a}, b) = \log(g_l(\mathbf{a}, b)) = \sum_{i=1}^{r_l} \log p_{li} + \sum_{i=r_l+1}^{q_l} \log(1 - p_{li}) \quad (1.4)$$

$$= \sum_{i=1}^{r_l} (a^T \mathbf{x}_{li} + b) - \sum_{i=1}^{q_l} \log(1 + \exp(a^T \mathbf{x}_{li} + b)), \quad (1.5)$$

where (1.4) follows directly using (1.3) and (1.5) is immediate using simple calculation.

Then the log likelihood function using m hospitals is given by

$$\sum_{i=1}^m f_i(\mathbf{a}, b) = \sum_{i=1}^m \left(\sum_{l=1}^{r_l} (\mathbf{a}^T \mathbf{x}_{li} + b) - \sum_{l=1}^{q_l} \log(1 + \exp(\mathbf{a}^T \mathbf{x}_{li} + b)) \right) \quad (1.6)$$

Finally, the maximum likelihood estimate is any optimal point to the problem

$$\underset{(\mathbf{a}, b) \in \mathbb{R}^n \times \mathbb{R}}{\text{maximize}} f(\mathbf{a}, b) = \sum_{i=1}^m f_i(\mathbf{a}, b) \quad (1.7)$$

It can easily be observed that the problem (1.7) takes the form of a global consensus problem [cf. (1.1)]. Then the maximum likelihood estimate of the parameters a and b can be found by solving the problem (1.7) using a suitable distributed method.

1.3 Our Contribution

In this thesis, we analyse the convergences of dual decomposition methods under non-ideal settings together with a systematic analysis on state-of-the-art distributed optimization methods. In particular, we consider a problem of minimizing a global convex objective function, which is a sum of local convex objective functions under general convex constraints [cf. Chapter 3.1], a formulation common to many types of large-scale signal processing and machine learning applications. The problem is commonly known as the global consensus problem [cf. problem (3.1)]. In further, a general formulation to the global consensus problem, which is known as the general consensus problem [cf. problem (4.133)] is also considered and related theoretical assertions are derived. More specifically, the main contributions of this study are as follows:

1. *State-of-the-art distributed optimization methods:* We analyse state-of-the-art distributed optimization methods that cope with large-scaled optimization problems (cf. Chapter 2). In particular, we provide a systematic exposition on subgradient methods with numerical implementations, on which our study is mainly centered around (cf. Chapter 2.2.2).

2. *Distributed optimization algorithms:* Two distributed algorithms are proposed based on dual decomposition techniques over non-ideal settings (*cf.* Chapter 3.4.1 and Chapter 3.4.2), where a set of subproblems coordinate towards achieving the global objective. Our modeling captures a wide range of distortions, including quantization errors, approximation errors, errors due to subproblem solver accuracy, noise in wireless settings, and measurement errors, among others, as long as they are *additive* and *bounded* (*cf.* Chapter 3.4: Remark 11).
3. *Properties of the dual function:* Important properties of the dual function, that rely on certain characteristics of the underlying primal problem are explicitly identified. Related Lipschitzian properties and strong convexity properties are analytically substantiated (*cf.* Proposition 1 and Proposition 2). Moreover, the bounding properties for the primal error, in terms of the dual error are also analysed (*cf.* Lemma 6).
4. *Convergence analysis in the dual-domain:* Under mild conditions, convergences of the algorithms in the dual-domain are analytically substantiated under both *fixed* step size and *nonsummable* step size rules (*cf.* Chapter 4.2.2: Corollary 2 and *cf.* Chapter 4.2.3: Corollary 3, Corollary 4). We show that the algorithms get into a neighborhood of optimality, the size of which depends on the level of underlying distortions. Convergence rates are also derived.
5. *Convergence analysis in the primal domain:* Under mild conditions, convergences of the algorithms in the primal-domain are analytically substantiated under both *fixed* step size and *nonsummable* step size rules (*cf.* Chapter 4.2.2: Proposition 4.2.2 and *cf.* Chapter 4.2.3: Proposition 4.2.3). Despite *primal infeasibility*, we show that the algorithms get into a neighborhood of optimality, the size of which depends on the level of underlying distortions. Convergence rates are also derived.
6. *Constructing primal feasible points and their optimality:* Constructing a feasible solution by using current infeasible primal variables is highlighted in Chapter 4.2.4 (*cf.* Remark 16). Under mild conditions, convergences of the algorithms in primal-

domain, while maintaining feasibility, are analytically substantiated (*cf.* Chapter 4.2.4: Proposition 4.2.5, Proposition 4.2.6, and Proposition 4.2.7). Convergence rates are also derived.

7. *Generalized problem formulation (general consensus problem) and related results:*

A generalization to the original problem formulation is furnished in Chapter 4.3.1. More importantly, we highlight how the theoretical assertions presented in chapter 4.1 (dual function properties) and Chapter 4.2 (convergence results in the dual domain and the primal domain) can be deduced with the generalized problem formulation (*cf.* Corollary 5, Corollary 6, and Corollary 7).

8. *Numerical Examples:* Theoretical assertions presented in this study are empirically evaluated in Chapter 4.4. The effect of quantization on the convergence in the dual domain and the primal domain are empirically tested in Chapter 4.4.1. The effect of measurement errors on the related convergences are empirically evaluated in Chapter 4.4.2.

Moreover, the major part of this thesis is mainly built on the contents presented in our manuscripts [38,39] and the conference paper [40]. The manuscript [38] is published in the Journal of Mathematics (Hindawi), and the manuscript [39] is accepted for publication in the IEEE Transactions on Signal Processing. The conference paper [40] is published in the Proceedings of the SLIIT International Conference on Advancements in Sciences and Humanities 2022.

1.4 Outline of the Thesis

In this section, we describe the outline of the thesis.

In Chapter 1.5, we present the background material, such as the related theoretical concepts, basic notations, and definitions on which this thesis is built. In particular, first, we provide an exposition of mathematical optimization problems. Further, we present basic theories related to convexity and duality together with their important consequences.

A review of the related literature is thoroughly presented in Chapter 2. The background of distributed optimization is discussed first. More importantly, the state-of-the-art distributed optimization methods together with related basic theories are discussed in more detail. A thorough exposition of the subgradient method with numerical examples is provided as our study is mainly based on subgradient methods together with dual decomposition. Further, different classes of convergence rates of algorithmic sequences are introduced. Finally, the challenges that arise in distributed optimization are discussed. Especially, distributed optimization over non-ideal settings has been discussed in more detail.

Chapter 3 introduces the main problem that we consider in this study and discuss the related distributed solution methods based on dual decomposition. More specifically, the effect of imperfect coordination between subsystems is considered. Further, two distributed algorithms over non-ideal settings are proposed based on dual decomposition.

Chapter 4 contains the main results of this research study. An extensive analysis of the properties of the dual function associated with the considered primal problem has been provided in Chapter 4.1. More importantly, the Lipschitzian and strong convexity properties of the dual function are analyzed and related theoretical results are explicitly derived. Further, useful relations among dual and primal variables are also presented, which are used to analyze the convergence properties of the proposed algorithms in the primal domain. Chapter 4.2 presents the convergence analysis of our proposed algorithms over non-ideal settings. Especially, the convergence properties are discussed under two main cases, CASE 1 and CASE 2. The convergence results in both dual and primal domains are established together with related rates of convergences. Moreover, convergences in primal feasible points are also discussed and related theoretical assertions are established. The generalized problem formulation is presented in Chapter 4.3. The extensions to all the theoretical assertions presented in Chapter 4.1 and Chapter 4.2 based on the global consensus problem are presented in this chapter. The theoretical assertions presented in this study are numerically evaluated in Chapter 4.4. More importantly, the effect of quan-

tization and measurement errors on the convergence in the dual domain and the primal domain are empirically tested.

Finally, we summarize the thesis and discuss the results in Chapter 5. We also discuss the possible future research directions to continue the work started with this thesis.

1.5 Related Theory

In this section, first, we present a brief introduction to mathematical optimization. Next, we present basic theory related to convexity and duality, on which this thesis is built. In particular, we present basic notations, definitions, and important consequences of convexity and duality. We refer the readers [37, 41] for thorough exposition.

1.5.1 Mathematical Optimization

Mathematical optimization is a technology that can be used to determine the best possible solution corresponding to the optimum performance of a quantitatively well defined system. Related technology is invoked by many systems that are employed in a variety of contexts, such as machine learning, automatic control, estimation and signal processing, communications and networks, electronic circuit design, data analysis and modeling, statistics, finance, and many others [42–44]. The process of reaching the best possible decision requires a phase of constructing a suitable mathematical model for a given solid problem, followed by a suitable solution method. Primarily, a well defined optimization model requires a quantitative objective criterion, in which our goal is to maximize (e.g. profit) or minimize (e.g. cost). Further, it requires specification of suitable constraints that representing the limitations of different resources that are equipped with the problem structure to be optimized. The best design of a well posed optimization model is a one which produce the best possible objective value, together with satisfying all problem constraints.

1.5.1.1 Optimization Problems

In this section, we introduce the general form of a mathematical optimization problem and discuss the basic terminology used in an optimization problem.

General form

Formally, a mathematical optimization problem has the form

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, p \\ & && h_i(\mathbf{x}) = 0, \quad i = 1, \dots, q, \end{aligned} \tag{1.8}$$

where $\mathbf{x} \in \mathbb{R}^n$ is called the *decision variable* or optimization variable and $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is called the *objective function*. The inequalities $g_i(\mathbf{x}) \leq 0$ are known as inequality constraints and the equations $h_i(\mathbf{x}) = 0$ are called equality constraints. The functions $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ and $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$ are known as inequality constraint functions and equality constraint functions, respectively. The set

$$\mathcal{S} = \text{dom } f \cap \left(\bigcap_{i=1}^p \text{dom } g_i \right) \cap \left(\bigcap_{i=1}^q \text{dom } h_i \right) \tag{1.9}$$

is called the domain of the optimization problem (1.8).

In general, optimization problem (1.8) is called a constrained optimization problem. In the absence of constraints, it is considered as an unconstrained problem.

It is worth noting that, the constraints given in (1.8) can be described abstractly by embedding all the constraints in to a single set. The corresponding formulation has the form

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{1.10}$$

where $\mathbf{x} \in \mathbb{R}^n$ and $\mathcal{X} \subseteq \mathbb{R}^n$.

Feasibility

A point $\mathbf{x} \in \mathcal{S}$ is called a *feasible point* of the problem (1.8) if it satisfies the constraints $g_i(\mathbf{x}) \leq 0, i = 1, \dots, p$, and $h_i(\mathbf{x}) = 0, i = 1, \dots, q$. The set of all feasible points is called the feasible set. If the feasible set is nonempty, we say that the optimization problem (1.8) is feasible.

Optimality

The *optimal value* of the problem (1.8) is denoted by f^* , and is defined by

$$f^* = \inf\{f(\mathbf{x}), \mid \mathbf{x} \in \mathcal{S}, g_i(\mathbf{x}) \leq 0, i = 1, \dots, p, h_i(\mathbf{x}) = 0, i = 1, \dots, q\}. \quad (1.11)$$

We say \mathbf{x}^* is an *optimal point* or optimal solution to the problem (1.8) if it is feasible and $f(\mathbf{x}^*) = f^*$.

Sub-optimality

A feasible point $\mathbf{x} \in \mathcal{S}$ with $f(\mathbf{x}) \leq f^* + \varepsilon$, where $\varepsilon > 0$ is called a ε -*suboptimal point*.

Local and global optimal points

A feasible point $x \in \mathcal{S}$ is called a *local optimal point* for the problem (1.8) if there exists an $r \in \mathbb{R}^+$ s.t.

$$f(\mathbf{x}) = \inf\{f(\mathbf{y}) \mid \mathbf{y} \in \mathcal{S}, g_i(\mathbf{y}) \leq 0, i = 1, \dots, p, h_i(\mathbf{y}) = 0, i = 1, \dots, q, \|\mathbf{y} - \mathbf{x}\| \leq r\}. \quad (1.12)$$

In other words, this means the local minimizer \mathbf{x} minimizes f only over a neighborhood of \mathbf{x} in the feasible set. The *global optimal point* is simply an optimal point \mathbf{x}^* to the problem (1.8).

1.5.2 Convexity

1.5.2.1 Convex Sets

Let $\mathcal{X} \subseteq \mathbb{R}^n$ and $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n$ are any two points in \mathcal{X} . Then, \mathcal{X} is said to be convex if

$$\alpha \mathbf{x}_1 + (1 - \alpha) \mathbf{x}_2 \in \mathcal{X},$$

where $\alpha \in \mathbb{R}$ and $0 \leq \alpha \leq 1$. Intuitively, a set \mathcal{X} is said to be convex if it contains the line segment between any two points in it (see Figure 1.3).

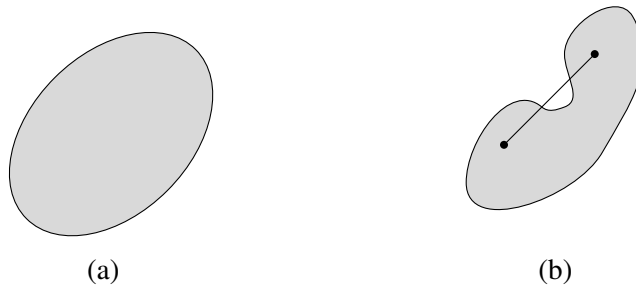


Figure 1.3: Graphical interpretation of convex and nonconvex sets. (a) The ellipse is a convex set. (b) The kidney shaped set is not convex, since the line segment between the given two points is not contained in the set.

1.5.2.2 Basics of Convex Functions

In this section, we discuss about convex functions which play an important role in convex optimization. In particular, we define convex, strictly convex, and strongly convex functions, and discuss important consequences.

Convex functions

Definition 1 (Convex function). *A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is called convex if $\text{dom } f$ is a convex set and if for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$ and for all $t \in \mathbb{R}$ s.t. $0 \leq t \leq 1$, it holds that*

$$f(t\mathbf{x} + (1 - t)\mathbf{y}) \leq tf(\mathbf{x}) + (1 - t)f(\mathbf{y}). \quad (1.13)$$

Geometrically, a function is convex if the chord from \mathbf{x} to \mathbf{y} , i.e. the line segment between the points $(\mathbf{x}, f(\mathbf{x}))$ and $(\mathbf{y}, f(\mathbf{y}))$ lies on or above the graph of f [see Figure 1.4 (a) for an illustration]. We say that f is concave if $-f$ is convex.

Next, we mention the following important theorem, which states an alternative way to characterize the convexity of a function.

Theorem 1. *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a differentiable function and suppose that $\text{dom } f$ is convex. Then, f is convex if and only if*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}), \text{ for all } \mathbf{x}, \mathbf{y} \in \text{dom } f. \quad (1.14)$$

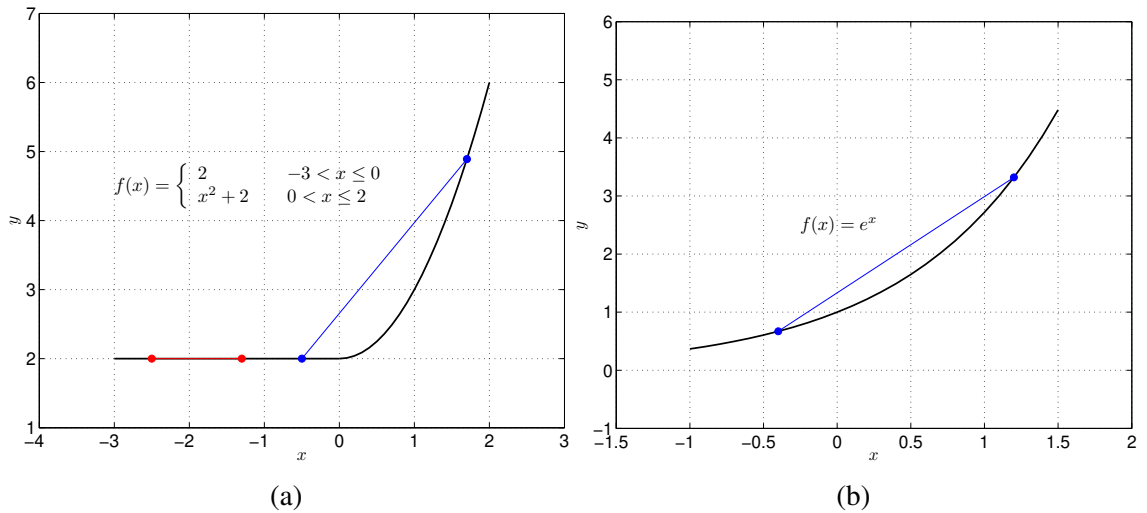
Inequality (1.14) of Theorem 1 has an interesting geometric interpretation, that is the first order Taylor expansion at any point in the domain is a global under estimator of the function f . Roughly speaking, the graph of f is bounded below by any tangent hyperplane of f drawn at any point on the domain.

Strictly convex functions

Definition 2 (Strictly convex function). *A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is called strictly convex if for all $\mathbf{x}, \mathbf{y} \in \text{dom } f$ with $\mathbf{x} \neq \mathbf{y}$ and for all $t \in \mathbb{R}$ with $0 < t < 1$, it holds that*

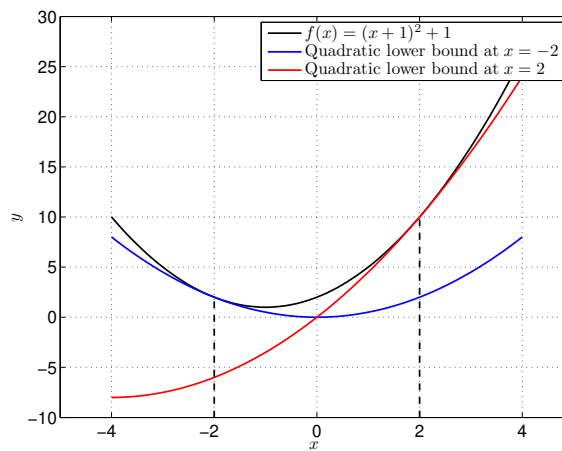
$$f(t\mathbf{x} + (1 - t)\mathbf{y}) < tf(\mathbf{x}) + (1 - t)f(\mathbf{y}). \quad (1.15)$$

This means, a function is strictly convex if the inequality (1.13) holds strictly whenever $\mathbf{x} \neq \mathbf{y}$ and $0 < t < 1$ [See Figure 1.4 (b)]. Obviously, a strictly convex function is always convex.



(a)

(b)



(c)

Figure 1.4: Classification of convex functions: (a) Convex function: The line segment between any two points on the graph lies on or above the graph. (b) Strictly convex function: The line segment between any two points on the graph lies above the graph. (c) Strongly convex function: The graph is always lower bounded by a convex quadratic function drawn to the graph at any point in the domain.

Strongly convex functions

Definition 3 (Strongly convex function). A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is strongly convex on $\mathcal{C} \subseteq \text{dom } f$, if $\exists l > 0$, s.t

$$f(tx + (1-t)y) \leq tf(\mathbf{x}) + (1-t)f(\mathbf{y}) - \frac{1}{2}lt(1-t)\|\mathbf{x} - \mathbf{y}\|_2^2, \forall \mathbf{x}, \mathbf{y} \in \mathcal{C}, \text{ when } 0 < t < 1, \quad (1.16)$$

where l is called the strong convexity constant of f .

Next, we present some important consequences of strongly convex functions.

Theorem 2 ([45], Exercise 12.59). *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a given function and $l \in \mathbb{R}^+$. Then the following properties are equivalent.*

- a) ∂f is strongly monotone with constant l .
- b) f is strongly convex with constant l .
- c) $f - \frac{1}{2}l\|\mathbf{x}\|_2^2$ is convex.

Theorem 3. *A differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is strongly convex with constant l if and only if*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) + \frac{l}{2}\|\mathbf{y} - \mathbf{x}\|_2^2, \text{ for all } \mathbf{x}, \mathbf{y} \in \text{dom}f. \quad (1.17)$$

Remark 1 (Quadratic lower bound). *Theorem 3 indicates that a differentiable strongly convex function is always lower bounded by a convex quadratic function $f_0(\mathbf{x}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) + \frac{l}{2}\|\mathbf{y} - \mathbf{x}\|_2^2$ at any point $\mathbf{x} \in \text{dom}f$ [see Figure 1.4 (c)].*

1.5.2.3 Convex Optimization Problems

We restate the problem (1.8) considered in Section 1.5.1.1 for clarity:

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, p \\ & && h_i(\mathbf{x}) = 0, \quad i = 1, \dots, q, \end{aligned} \quad (1.18)$$

where $\mathbf{x} \in \mathbb{R}^n$. Then the problem (1.18) is called convex if

- the objective function f is convex,
- the inequality constraint functions g_i s are convex, and
- the equality constraint functions h_i s are affine.

If the problem (1.18) has linear equality constraints, i.e., $h_i(\mathbf{x}) = \mathbf{a}_i^T \mathbf{x} - b_i$, where $\mathbf{a}_i \in \mathbb{R}^n$ and $b_i \in \mathbb{R}$ for all $i = 1, \dots, q$, then the set of equality constraints can express more compactly in the matrix form $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{q \times n}$ and $\mathbf{b} \in \mathbb{R}^q$. Then the related convex optimization problem takes the form

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, p \\ & && \mathbf{Ax} = \mathbf{b}. \end{aligned} \tag{1.19}$$

More importantly, convex optimization problems can be solved very reliably and efficiently using interior-point methods and first order methods, and most of the theories related to convex optimization have been already developed. Therefore, recognizing or formulating a problem as a convex optimization problem gives us a great advantage [37,46]. For example, if we consider a non-convex constrained optimization problem (a minimization problem), the associated dual problem (which is a maximization problem) is always concave. Thus the equivalent minimization problem with the negative dual function is always convex. Hence, under certain conditions, the original problem can be solved using the dual problem which provides an easy working environment due to the convexity of the negative dual function.

1.5.3 Duality

1.5.3.1 The Lagrange Dual Function

Let us first consider the general form (1.8) of an optimization problem. In particular, the optimization problem (1.8) is called the primal problem. Suppose that the set of optimal solutions to the problem (1.8) is nonempty. Let f^* denote the optimal value. The problem (1.8) is not necessarily to be convex. Then the Lagrangian $L : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \rightarrow \mathbb{R}$

associated with the problem (1.8) is given by

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\eta}) = f(\mathbf{x}) + \sum_{i=1}^p \lambda_i g_i(\mathbf{x}) + \sum_{i=1}^q \eta_i h_i(\mathbf{x}), \quad (1.20)$$

where $\lambda_i \in \mathbb{R}$ and $\eta_i \in \mathbb{R}$ represent the *Lagrange multipliers* associated with the i th inequality constraint $g_i(\mathbf{x}) \leq 0$ and the i th equality constraint $h_i(\mathbf{x}) = 0$, respectively. Moreover, $\boldsymbol{\lambda} = [\lambda_1 \dots \lambda_p]^T \in \mathbb{R}^p$ and $\boldsymbol{\eta} = [\eta_1 \dots \eta_q]^T \in \mathbb{R}^q$ are called the *dual variables* or *Lagrange multiplier vectors* associated with (1.8).

Finally, The *Lagrange dual function* $g : \mathbb{R}^p \times \mathbb{R}^q \rightarrow \mathbb{R}$ is given by

$$g(\boldsymbol{\lambda}, \boldsymbol{\eta}) = \inf_{\mathbf{x} \in \mathcal{S}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\eta}) = \inf_{\mathbf{x} \in \mathcal{S}} \left(f(\mathbf{x}) + \sum_{i=1}^p \lambda_i g_i(\mathbf{x}) + \sum_{i=1}^q \eta_i h_i(\mathbf{x}) \right). \quad (1.21)$$

Important properties of the dual function associated with the primal problem (1.8) are summarized below.

Remark 2 (Concavity). *The dual function is always concave as it is the pointwise infimum of a family of affine functions of $\boldsymbol{\lambda}$ and $\boldsymbol{\eta}$.*

Remark 3 (Lower bounds on optimal value). *For any $\boldsymbol{\lambda} \geq \mathbf{0}$ and for any $\boldsymbol{\eta} \in \mathbb{R}^q$, the dual function (1.21) yields lower bounds on f^* of the primal problem (1.8), i.e.,*

$$g(\boldsymbol{\lambda}, \boldsymbol{\eta}) \leq f^*. \quad (1.22)$$

1.5.3.2 The Lagrange Dual Problem

It is worth noting that the Lagrange dual function $g(\boldsymbol{\lambda}, \boldsymbol{\eta})$ [cf. (1.21)] produces lower bounds on the optimal value f^* of the problem (1.8) for any $\boldsymbol{\lambda} \geq \mathbf{0}$ and $\boldsymbol{\eta} \in \mathbb{R}^q$ [cf. Remark 3]. Thus, it is interesting to find the best lower bound on f^* that is produced by $g(\boldsymbol{\lambda}, \boldsymbol{\eta})$. The procedure to find the best lower bound leads to solving the optimization problem

$$\begin{aligned} & \text{maximize} && g(\boldsymbol{\lambda}, \boldsymbol{\eta}) \\ & \text{subject to} && \boldsymbol{\lambda} \geq \mathbf{0}. \end{aligned} \quad (1.23)$$

This problem (1.23) is called the *Lagrange dual problem* associated with the primal problem (1.8). It is worth emphasizing that the equivalent minimization problem of the Lagrange dual problem (1.23) can be posed as

$$\begin{aligned} & \text{minimize} && -g(\boldsymbol{\lambda}, \boldsymbol{\eta}) \\ & \text{subject to} && \boldsymbol{\lambda} \geq \mathbf{0}. \end{aligned} \tag{1.24}$$

The optimization problem (1.24) is always convex, as $-g(\boldsymbol{\lambda}, \boldsymbol{\eta})$ is convex (*cf.* Remark 2) and the constraint is convex. Thus, the Lagrange dual problem can always be viewed as a convex optimization problem.

1.5.3.3 Weak and Strong Duality

Weak Duality: Let d^* be the optimal value of the Lagrange dual problem (1.23). Thus from (1.22) we have $d^* \leq f^*$. This property is called *weak duality*.

Strong Duality: If it holds $d^* = f^*$, we say that *strong duality* holds.

Following result gives a certificate for holding the strong duality.

Theorem 4 ([37], Section 5.2.3). *Suppose that the primal problem (1.8) is convex and the constraints satisfy the Slater's condition: $\exists \mathbf{x} \in \text{relint } \mathcal{S}$ s.t.,*

$$g_i(\mathbf{x}) < 0, \quad i = 1, \dots, p, \quad \text{and} \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \tag{1.25}$$

where $\mathbf{A} \in \mathbb{R}^{q \times n}$ and $\mathbf{b} \in \mathbb{R}^q$. Then the strong duality holds.

1.5.3.4 KKT Optimality Conditions

Consider the primal problem (1.8) and its associated dual problem (1.23). Then the *Karush-Kuhn-Tucker* (KKT) conditions associated with the problem (1.8) are given by

1. Stationary condition: $\mathbf{0} \in \partial (f(\mathbf{x}) + \sum_{i=1}^p \lambda_i g_i(\mathbf{x}) + \sum_{i=1}^q \eta_i h_i(\mathbf{x}))$
2. Primal feasibility: $g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, p$

$$h_i(\mathbf{x}) = 0, \quad i = 1, \dots, q$$

3. Dual feasibility: $\lambda_i \geq 0, \quad i = 1, \dots, p$

4. Complementary slackness: $\lambda_i g_i(\mathbf{x}) = 0, \quad i = 1, \dots, p,$

where $\mathbf{x} \in \mathcal{S}$.

KKT conditions are usually used to determine primal and dual optimal values. Here we provide a well-known result, which allows us to determine whether a given pair of primal and dual optimal points produce optimal values.

Theorem 5. *Suppose that the strong duality holds for a given optimization problem. Then, \mathbf{x}^* and the pair $(\boldsymbol{\lambda}^*, \boldsymbol{\eta}^*)$ are the primal and dual optimal values respectively, if and only if \mathbf{x}^* and the pair $(\boldsymbol{\lambda}^*, \boldsymbol{\eta}^*)$ satisfy the KKT conditions.*

1.6 Mathematical preliminaries

In this section, we present important definitions we use in most of the proofs of convergence properties discussed in this thesis. Moreover, some well-known theoretical results are also presented for completeness.

1.6.1 Basic Definitions

Definition 4 (Open set). *A set $\mathcal{C} \subseteq \mathbb{R}^n$ is called open if $\text{int } \mathcal{C} = \mathcal{C}$.*

Definition 5 (Closed set). *A set $\mathcal{C} \subseteq \mathbb{R}^n$ is called closed if its complement $\mathbb{R}^n \setminus \mathcal{C}$ is open.*

Definition 6 (Epi graph). *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$. The set $\text{epi } f = \{(\mathbf{x}, t) \mid \mathbf{x} \in \text{dom } f, t \in \mathbb{R}, t \geq f(\mathbf{x})\}$ is called the epigraph of f .*

Note that the $\text{epi } f$ is a subset of \mathbb{R}^{n+1} . An equivalent definition for convex functions can also be given using the epigraph of a function as highlighted in the following remark.

Remark 4. *A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex if and only if its epigraph is a convex set in \mathbb{R}^{n+1} .*

Definition 7 (Closed function). A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is said to be closed if $\text{epi } f$ is closed.

Definition 8 (ℓ_2 -norm). The ℓ_2 -norm or the Euclidean norm of a vector $\mathbf{x} \in \mathbb{R}^n$ is defined as,

$$\|\mathbf{x}\|_2 = (\mathbf{x}^T \mathbf{x})^{1/2} = \left(\sum_{i=1}^n x_i^2 \right)^{1/2}, \quad (1.26)$$

where $\mathbf{x} = [x_1 \dots x_n]^T$ and $x_i \in \mathbb{R}, \forall i = 1, \dots, n$.

Definition 9 (Matrix 2-norm). The matrix norm induced by the Euclidean vector norm is given by

$$\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{x}\|_2, \quad (1.27)$$

and is known as the matrix 2-norm. The matrix 2-norm is often called the spectral norm.

Definition 10 (Spectrum of a matrix). The set of distinct eigenvalues of an $n \times n$ matrix \mathbf{A} is called the spectrum of \mathbf{A} and is denoted by $\sigma(\mathbf{A})$.

Definition 11 (Spectral radius). Let λ be an eigenvalue of a square matrix $\mathbf{A}^{n \times n}$. Then,

$$\rho(\mathbf{A}) = \sup_{\lambda \in \sigma(\mathbf{A})} |\lambda| \quad (1.28)$$

is called the spectral radius of \mathbf{A} .

Remark 5 (Properties of spectral norm).

- i. $\|\mathbf{A}\| = \sqrt{\rho(\mathbf{A}^T \mathbf{A})}$
- ii. $\rho(\mathbf{A}) \leq \|\mathbf{A}\|$ (This property is true for any matrix norm).
- iii. When \mathbf{A} is symmetric, $\|\mathbf{A}\| = \rho(\mathbf{A})$.
- iv. $\|\mathbf{A}\mathbf{x}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|$, where $\mathbf{x} \in \mathbb{R}^n$ (This property is true for any matrix norm with its underlying vector norm).

Definition 12 (Kronecker Product). *The Kronecker product of two matrices $\mathbf{A}^{m \times n}$ and $\mathbf{B}^{p \times q}$ is defined to be the $mp \times nq$ matrix*

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \cdots & a_{2n}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & a_{m2}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix}. \quad (1.29)$$

Remark 6. *Let λ_i , $i \in \{1, \dots, n\}$ and μ_i , $i \in \{1, \dots, m\}$ be eigenvalues of matrices $\mathbf{A}^{n \times n}$ and $\mathbf{B}^{m \times m}$, respectively. Then the eigenvalues of $\mathbf{A} \otimes \mathbf{B}$ are the mn values $\lambda_i \mu_j$, where $i \in \{1, \dots, n\}$ and $j \in \{1, \dots, m\}$.*

Definition 13 (Strong monotonicity). *A mapping $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is called strongly monotone, if $\exists \kappa > 0$, s.t*

$$(\mathbf{y}_1 - \mathbf{y}_0)^T (\mathbf{x}_1 - \mathbf{x}_0) \geq \kappa \|\mathbf{x}_1 - \mathbf{x}_0\|_2^2, \text{ whenever } \mathbf{y}_0 \in f(\mathbf{x}_0) \text{ and } \mathbf{y}_1 \in f(\mathbf{x}_1),$$

$$\text{where } \mathbf{x}_0, \mathbf{x}_1 \in \mathbb{R}^n. \quad (1.30)$$

Definition 14 (Lipschitz continuity). *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ with $\text{dom } f = \mathcal{X}$. Then, f is Lipschitz continuous on $\mathcal{C} \subseteq \mathcal{X}$, if $\exists L > 0$, s.t*

$$|f(\mathbf{x}) - f(\mathbf{y})| \leq L \|\mathbf{x} - \mathbf{y}\|_2, \forall \mathbf{x}, \mathbf{y} \in \mathcal{C}. \quad (1.31)$$

Here L is called the Lipschitz constant for f on \mathcal{C} .

Remark 7. *A differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is said to have a Lipschitz continuous gradient on $\mathcal{C} \subseteq \text{dom } f$, if $\exists L > 0$ s.t.*

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2 \leq L \|\mathbf{x} - \mathbf{y}\|_2, \forall \mathbf{x}, \mathbf{y} \in \mathcal{C}, \quad (1.32)$$

where L is called the gradient Lipschitz continuous constant of f .

Definition 15 (Conjugate function). Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Then the function $f^* : \mathbb{R}^n \rightarrow \mathbb{R}$ defined by

$$f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \text{dom } f} (\mathbf{y}^T \mathbf{x} - f(\mathbf{x})) \quad (1.33)$$

is called the conjugate function of f .

Definition 16 (Indicator function). Let $\mathcal{C} \subseteq \mathbb{R}^n$ be any given set. Then the function $\delta_{\mathcal{C}}$ defined by

$$\delta_{\mathcal{C}}(\mathbf{x}) = \begin{cases} 0 & ; \mathbf{x} \in \mathcal{C} \\ \infty & ; \mathbf{x} \notin \mathcal{C}, \end{cases} \quad (1.34)$$

is called the indicator function of the set \mathcal{C} .

Definition 17 (Subgradient). Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a real valued function. Then, a vector $\mathbf{d} \in \mathbb{R}^n$ is called a subgradient of f at $\mathbf{x} \in \text{dom } f$, if

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{d}^T(\mathbf{y} - \mathbf{x}), \text{ for all } \mathbf{y} \in \text{dom } f. \quad (1.35)$$

The set of all subgradients of f at $\mathbf{x} \in \text{dom } f$ is called the subdifferential of f at \mathbf{x} and denoted by $\partial f(\mathbf{x})$. If f is convex and differentiable at \mathbf{x} , then its subgradient at \mathbf{x} is unique and it is the gradient of f at \mathbf{x} , i.e., $\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}$.

Remark 8. If $f(\mathbf{y}) < f(\mathbf{x}) + \mathbf{d}^T(\mathbf{y} - \mathbf{x})$, for all $\mathbf{y} \in \text{dom } f$, then the vector $\mathbf{d} \in \mathbb{R}^n$ is called a supergradient of f at $\mathbf{x} \in \text{dom } f$ [cf. (1.35)].

Definition 18 (Limit superior). The limit superior of a sequence x_n is defined by

$$\limsup_{n \rightarrow \infty} x_n = \lim_{n \rightarrow \infty} \left(\sup_{m \geq n} x_m \right), \quad (1.36)$$

where $\sup_{m \geq n} x_m = \sup\{x_m : m \geq n\}$.

Definition 19 (“Small oh” notation). If the sequences $u^{(k)} \in \mathbb{R}^n$, $v^{(k)} \in \mathbb{R}^m$, where $k \in \mathbb{Z}_+^0$, are such that $\|v^{(k)}\|/\|u^{(k)}\| \rightarrow 0$ as $k \rightarrow \infty$, then $v^{(k)} = o(u^{(k)})$. The notation $o(\cdot)$ is called the “small oh” notation.

Definition 20 (“Big oh” notation). *If for sequences $u^{(k)} \in \mathbb{R}^n$, $v^{(k)} \in \mathbb{R}^m$, where $k \in \mathbb{Z}_0^+$, there exist $\alpha > 0$ and $k_0 \in \mathbb{Z}_0^+$, such that $\|v^{(k)}\| \leq \alpha \|u^{(k)}\|$ for all $k \geq k_0$, then $v^{(k)} = O(u^{(k)})$. The notation $O(\cdot)$ is called the “big Oh” notation.*

1.6.2 Basic Results

Lemma 1 ([41], Section 12). *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a continuously differentiable function and the gradient ∇f is Lipschitz continuous on $\mathcal{C} \subseteq \mathbb{R}^n$ with constant $L > 0$ (cf. Remark 7). Then it holds*

$$|f(\mathbf{x}) - f(\mathbf{y}) - \nabla f(\mathbf{y})^T(\mathbf{x} - \mathbf{y})| \leq \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|_2^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{C}. \quad (1.37)$$

Next, an important consequence of Lemma 1 is highlighted in the following remark.

Remark 9 (Quadratic upper bound). *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a continuously differentiable function and is with a Lipschitz continuous gradient on $\mathcal{C} \subseteq \mathbb{R}^n$. Then, at any point $\mathbf{y} \in \mathcal{C}$, the function $f(\mathbf{x})$ can be upper bounded by a strongly convex quadratic function $f_{upper} = f(\mathbf{y}) + \nabla f(\mathbf{y})^T(\mathbf{x} - \mathbf{y}) + (L/2)\|\mathbf{x} - \mathbf{y}\|_2^2$.*

Lemma 2 ([41], Section 12). *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a continuously differentiable and convex function on \mathbb{R}^n . If the gradient ∇f is Lipschitz continuous on \mathbb{R}^n with constant $L > 0$, then*

$$0 \leq f(\mathbf{x}) - f(\mathbf{y}) - \nabla f(\mathbf{y})^T(\mathbf{x} - \mathbf{y}) \leq \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|_2^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n. \quad (1.38)$$

Theorem 6 ([47], Section 1.2.2). *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a convex function. Suppose that f is differentiable at $\mathbf{x}^* \in \mathbb{R}^n$ and $\nabla f(\mathbf{x}^*) = \mathbf{0}$. Then \mathbf{x}^* is a global minimum point of $f(\mathbf{x})$ on \mathbb{R}^n .*

Theorem 7 ([47], Section 1.3.2). *A minimum point of a strictly convex function is (globally) unique.*

Chapter 2

Literature Review

This chapter is organized as follows. First, we discuss the importance of distributed optimization. Next, we discuss the state-of-the-art distributed optimization methods together with basic theories. Here, we provide a systematic exposition of decomposition methods which are general approaches to solving an optimization problem in a distributed manner. More importantly, we provide a thorough exposition of the subgradient method with numerical examples as our study is mainly based on subgradient methods together with dual decomposition. Further, we discuss different classes of convergences rates of iterative/algorithmic solution sequences. Finally, the challenges that arise in distributed optimization are discussed.

2.1 Distributed Optimization: The background

Historically, centralized methods have been the primary tool for solving optimization problems in means of many application fields. In centralized optimization, the underlying system is considered as one whole system which is operated under one central controller. However, the application of centralized methods was not suitable with the increase in problem dimensionality and large data sets in modern systems. Consequently, there has been a tremendous expansion of interest in distributed optimization methods to cope with the underlying large-scale problems.

A distributed optimization setting consists of many subsystems (usually call as users or agents) to solve a global problem interactively via communication among neighboring subsystems. Rather than performing a central calculation, solving a problem in a distributed manner afford many considerable advantages. The entire process in a centralized

system may fail if the central controller is corrupted. However, the failures in several subsystems in a distributed system may not harm the operation of a distributed setting. Moreover, subsystems of a distributed setting share information only with required neighboring subsystems. This can improve cybersecurity and reduce communication costs. Further, distributed methods are computationally superior in terms of the solution speed compared to centralized methods with the ability to perform parallel computations in distributed algorithms [48]. Finally, another major drawback in a centralized setting appears in respect of the privacy of data. However, distributed methods have the potential to secure the privacy of data among respective agents with the existing distributed computing structure in the underlying distributed system. With all these benefits, there has been much recent interest in the study of distributed optimization techniques over large-scale data intensive problems.

Distributed optimization techniques are widely used in many application fields, including machine learning, signal processing, communications [49–52], electricity grid [53,54], smart grids, wireless sensor networks [55], and statistical learning [56]. In many of these applications, the main goal is to optimize a global objective using several subsystems through local computations and local communications among the neighboring subsystems in the underlying networked system. In general, a distributed optimization problem consists of the following components (see Figure 2.1):

- The optimization problem (objective function and related constraints), that the agents in the network need to solve collaboratively.
- The local information structure, which determines what information is locally available for each agent in the network.
- The communication structure, which specifies the connectivity of the underlying networked system.

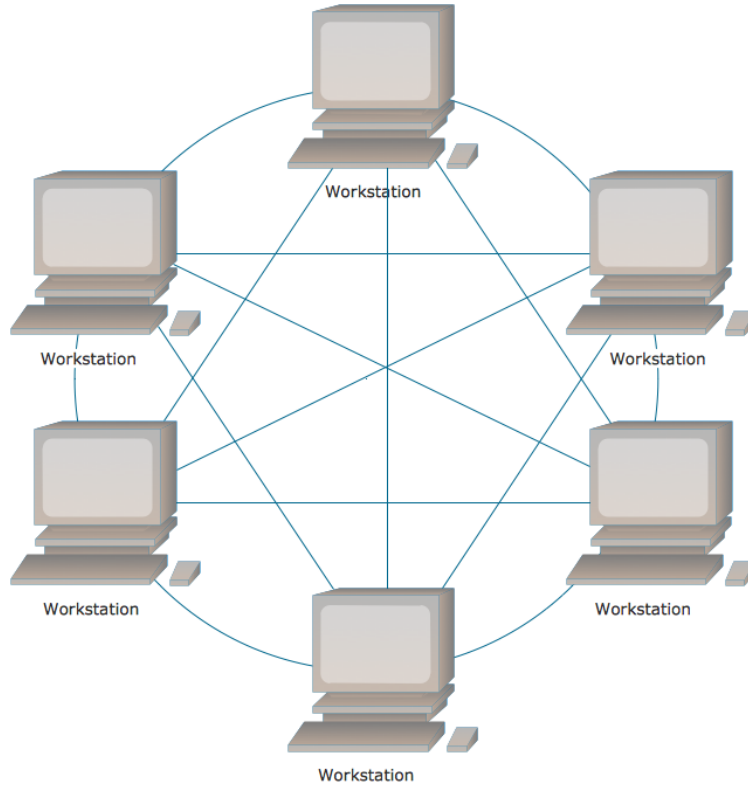


Figure 2.1: Distributed networked system.

In general, a distributed optimization problem has the form

$$\begin{aligned}
 &\text{minimize} && f(\mathbf{x}) = \sum_{i=1}^m f_i(\mathbf{x}) \\
 &\text{subject to} && \mathbf{x} \in \mathcal{X},
 \end{aligned} \tag{2.1}$$

where $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is usually convex and considered as the local objective function of agent i , where $i = 1, \dots, m$, and $\mathcal{X} \subseteq \mathbb{R}^n$ is called the common constraint set which is closed and convex. Here, the function f_i is only known to agent i and $\mathbf{x} \in \mathcal{X}$ represents the global decision vector which wants to estimate collectively by agents in the system using local information. The problem 2.1 is usually called the global consensus problem (*cf.* Chapter 1.1).

2.2 Distributed Optimization Methods

A well posed optimization problem requires a suitable optimization method to determine the best possible solution. However, in reality, it may be difficult or not possible to find closed form solutions to optimization problems associated with many real world applications. Thus, the approximate methods that provide approximate solutions are heavily used to solve optimization problems with a required degree of accuracy. Roughly speaking, with the existing large data sets available in modern systems, iterative methods are indeed of great importance in producing distributed optimization methods.

In particular, distributed optimization methods enable an optimization system to solve a global problem interactively with many subsystems. With challenges, such as huge problem dimensionality, large data volumes, and the geographical distribution of data, the most commonly used distributed optimization methods are the first-order methods.

In general, first order methods (e.g., subgradient methods and alternating direction method of multipliers) are the techniques that only use function values and first order information, i.e, the information on gradients/subgradients of functions comprising in an underlying optimization model. Compared to second order methods (e.g., Newton's method and interior-point methods) [57], the first order methods require low computational cost as they do not require any computation on second order information or the Hessian. Thus, it requires low iteration cost as well as low memory storage. Consequently, it creates a revived interest in using first-order methods in many large-scale optimization problems. In this respect, the currently existing state-of-the-art first order methods are the subgradient methods [58], alternating direction method of multipliers (ADMM) [56], proximal gradient method [59], and dual averaging [60].

2.2.1 Decomposition Methods

Many distributed optimization algorithms are built on decomposition methods. In particular, a decomposition is an interesting approach to solving an optimization problem

by breaking it up into smaller subproblems and solving each of them separately. These subproblems get solved either in parallel or sequentially [46, 61–64]. Decomposition in the field of optimization appears in early work on large-scale linear programs from the 1960s [65]. The simplest decomposition structure is available in block separable problems. An example of a block separable problem is

$$\begin{aligned} & \text{minimize} && f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2) \\ & \text{subject to} && \mathbf{x}_1 \in \mathcal{X}_1, \mathbf{x}_2 \in \mathcal{X}_2, \end{aligned} \tag{2.2}$$

where $\mathcal{X}_1 \subseteq \mathbb{R}^n$ and $\mathcal{X}_2 \subseteq \mathbb{R}^m$. The problem formulation given in 2.2 allows us to minimize the functions $f_1(\mathbf{x}_1)$ and $f_2(\mathbf{x}_2)$ separately and in parallel. However, this problem formulation is trivial and not interesting too, as many real life working problems appear in a more complex form than this [61]. The problem 2.2 will appear in a more complicated form and it will create more interest when the variables \mathbf{x}_1 and \mathbf{x}_2 are coupled. Then the functions $f_1(\mathbf{x}_1)$ and $f_2(\mathbf{x}_2)$ cannot be solved separately. Thus, the techniques that handle such situations are of utmost importance. In this respect, the most well-known currently available decomposition methods are *primal decomposition* and *dual decomposition*.

2.2.1.1 Primal decomposition

We consider a constrained minimization problem which is jointly solved by m subsystems (usually called as users), where $m \in \mathbb{Z}^+$. The respective problem takes the form

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) = \sum_{i=1}^m f_i(\mathbf{x}_i, \mathbf{y}) \\ & \text{subject to} && \mathbf{x}_i \in \mathcal{C}_i, i = 1, 2, \dots, m, \mathbf{y} \in \mathcal{Y}, \end{aligned} \tag{2.3}$$

where $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m, \mathbf{y})$, $\mathcal{C}_i \subseteq \mathbb{R}^n$, and $\mathcal{Y} \subseteq \mathbb{R}^n$. The functions $f_i : \mathbb{R}^{2n} \rightarrow \mathbb{R}$ represent real valued local objective functions of individual subsystems. Here, it can easily be observed that the variable \mathbf{y} has coupled the local objective functions of individual subsystems. Thus, we call the variable \mathbf{y} the *complicating variable*. When the variable \mathbf{y} is fixed the problem (2.3) becomes separable, [cf. (2.2)] and it decomposes in to m smaller

subproblems

$$S_i(\mathbf{y}) = \underset{\mathbf{x}_i \in \mathcal{C}_i}{\text{minimize}} f_i(\mathbf{x}_i, \mathbf{y}).$$

Then the original problem (2.3) is equivalent to the problem

$$\underset{\mathbf{y} \in \mathcal{Y}}{\text{minimize}} S(\mathbf{y}) = \sum_{i=1}^m S_i(\mathbf{y}). \quad (2.4)$$

The problem (2.4) is called the master problem in primal decomposition [61]. Next, the original problem (2.3) is usually can be solved by solving the master problem (2.4) using a suitable distributed algorithm.

2.2.1.2 Dual decomposition

Here we consider the same problem (2.3) discussed under primal decomposition only with two users for clarity. Then, the related minimization problem takes the form

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}) = f_1(\mathbf{x}_1, \mathbf{y}) + f_2(\mathbf{x}_2, \mathbf{y}) \\ \text{subject to} \quad & \mathbf{x}_1 \in \mathcal{X}_1, \mathbf{x}_2 \in \mathcal{X}_2, \mathbf{y} \in \mathcal{Y}, \end{aligned} \quad (2.5)$$

where $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{y})$, $\mathbf{x}_1 \in \mathbb{R}^n$, $\mathbf{x}_2 \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^n$, $\mathcal{X}_1 \subseteq \mathbb{R}^n$, $\mathcal{X}_2 \subseteq \mathbb{R}^n$, and $\mathcal{Y} \subseteq \mathbb{R}^n$. As usual, f_1 and f_2 represent local objective functions of subsystem 1 and subsystem 2, respectively. Next, we rearrange the problem (2.5) as

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}_1, \mathbf{y}_2) = f_1(\mathbf{x}_1, \mathbf{y}_1) + f_2(\mathbf{x}_2, \mathbf{y}_2) \\ \text{subject to} \quad & \mathbf{y}_1 = \mathbf{y}_2, \\ & \mathbf{x}_1 \in \mathcal{X}_1, \mathbf{x}_2 \in \mathcal{X}_2, \mathbf{y}_1, \mathbf{y}_2 \in \mathcal{Y} \end{aligned} \quad (2.6)$$

by introducing new variables \mathbf{y}_1 , \mathbf{y}_2 , and an equality constraint [61]. According to this new formulation, the objective function f is now separable. Next we consider the dual problem formulation of (2.6). The Lagrangian associated with (2.6) is given by (*cf.* Sec-

tion 1.5.3.1)

$$L(\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}_1, \mathbf{y}_2, \boldsymbol{\lambda}) = f_1(\mathbf{x}_1, \mathbf{y}_1) + f_2(\mathbf{x}_2, \mathbf{y}_2) + \boldsymbol{\lambda}^T(\mathbf{y}_1 - \mathbf{y}_2), \quad (2.7)$$

where $\boldsymbol{\lambda} \in \mathbb{R}^n$. Next, the related dual function (cf. Section 1.5.3.1) is given by

$$g(\boldsymbol{\lambda}) = \inf_{\substack{\mathbf{x}_1 \in \mathcal{X}_1, \mathbf{x}_2 \in \mathcal{X}_2 \\ \mathbf{y}_1, \mathbf{y}_2 \in \mathcal{Y}}} L(\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}_1, \mathbf{y}_2) \quad (2.8)$$

$$= \inf_{\substack{\mathbf{x}_1 \in \mathcal{X}_1, \mathbf{x}_2 \in \mathcal{X}_2 \\ \mathbf{y}_1, \mathbf{y}_2 \in \mathcal{Y}}} [(f_1(\mathbf{x}_1, \mathbf{y}_1) + \boldsymbol{\lambda}^T \mathbf{y}_1) + (f_2(\mathbf{x}_2, \mathbf{y}_2) - \boldsymbol{\lambda}^T \mathbf{y}_2)]. \quad (2.9)$$

We can note that the problem (2.9) is separable. Thus the dual function $g(\boldsymbol{\lambda})$ can be obtained by solving the subproblems

$$\text{Subproblem 1: } g_1(\boldsymbol{\lambda}) = \inf_{\mathbf{x}_1 \in \mathcal{X}_1, \mathbf{y}_1 \in \mathcal{Y}} f_1(\mathbf{x}_1, \mathbf{y}_1) - \boldsymbol{\lambda}^T \mathbf{y}_1 \quad (2.10)$$

$$\text{Subproblem 2: } g_2(\boldsymbol{\lambda}) = \inf_{\mathbf{x}_2 \in \mathcal{X}_2, \mathbf{y}_2 \in \mathcal{Y}} f_2(\mathbf{x}_2, \mathbf{y}_2) - \boldsymbol{\lambda}^T \mathbf{y}_2 \quad (2.11)$$

Then the associated dual problem is given by

$$\underset{\boldsymbol{\lambda} \in \mathbb{R}^n}{\text{maximize}} \quad g(\boldsymbol{\lambda}) = g_1(\boldsymbol{\lambda}) + g_2(\boldsymbol{\lambda}). \quad (2.12)$$

This is called the master problem in dual decomposition. This problem can be solved using an iterative method such as the subgradient method. It is worth noting that although we are able to solve the dual problem and find dual optimal measures, we still cannot guarantee that we can find primal optimal measures without introducing some acceptable conditions on the local objective functions f_1 and f_2 . For example, if f_1 and f_2 are strictly convex, then the primal variables $\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}_1$, and \mathbf{y}_2 found by solving two subproblems g_1 and g_2 , are guaranteed to converge towards the optimal solution of the primal problem (2.5) [61].

2.2.2 The Subgradient Method

The subgradient method has been commonly used in the literature as it is simple, amenable to implementation, and easily generalizable. In particular, the subgradient method can be easily combined with primal or dual decomposition techniques (*cf.* Section 2.2.1.1 and Section 2.2.1.2) and produce simple distributed algorithms for a given problem. The subgradient method is usually used to minimize nondifferentiable convex problems. This method can be considered as an extended version of the gradient method, or on the other hand, one can view the gradient method as a special case of the subgradient method when the underlying objective function is nondifferentiable.

Consider an unconstrained optimization problem of the form

$$\text{minimize } f(\mathbf{x}), \quad (2.13)$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex and $\mathbf{x} \in \mathbb{R}^n$. Then the subgradient method to solve the optimization problems of the form (2.13) is given by

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k \mathbf{g}^k, \quad (2.14)$$

where k represents the iteration index, \mathbf{x}^k is an approximate solution to the problem (2.13) at k th iteration, \mathbf{g}^k is any subgradient of f at \mathbf{x}^k (*cf.* Definition 17), and $\gamma_k > 0$ is a step size selection at k th iteration. When the function f is differentiable, the subgradient method (2.14) is simply reduced to the standard gradient method

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k \nabla f(\mathbf{x}^k). \quad (2.15)$$

Step size selection plays a major role in obtaining the convergence properties of the method (2.14). In general, commonly used step sizes are given below [58].

1. Constant step size: $\gamma_k = \gamma \forall k$.

2. Square summable but not summable: The step sizes satisfy

$$\gamma_k \geq 0, \quad \sum_{k=1}^{\infty} \gamma_k^2 < \infty, \quad \text{and} \quad \sum_{k=1}^{\infty} \gamma_k = \infty.$$

Ex: $\gamma_k = 1/k$.

3. Nonsummable diminishing: The step sizes satisfy

$$\gamma_k \geq 0, \quad \lim_{k \rightarrow \infty} \gamma_k = 0, \quad \text{and} \quad \sum_{k=1}^{\infty} \gamma_k = \infty.$$

Ex: $\gamma_k = 1/\sqrt{k}$.

The convergence of the method (2.14) is guaranteed under various consideration using above step size rules. Some basic convergence results are outlined below.

Theorem 8 ([58], Section 3). *Let \mathcal{X}^* , the set of minimizers of the problem is nonempty, $\|\mathbf{g}^k\|$ is bounded, and $\|\mathbf{x}^0 - \mathbf{x}^*\|$ is bounded, where $\mathbf{x}^* \in \mathbf{X}^*$ and \mathbf{x}^0 is an initial point of the algorithm (2.14). Then in method/algorithm (2.14) it holds*

$$f_{best}^k - f^* \leq \frac{R^2 + G^2 \sum_{i=1}^k \gamma_i^2}{2 \sum_{i=1}^k \gamma_i}, \quad (2.16)$$

where R is s.t $\|\mathbf{x}^0 - \mathbf{x}^*\| \leq R$, G is s.t $\|\mathbf{g}^k\| \leq G$ for all k , and $f_{best}^k = \min_{i \in \{1, \dots, k\}} f(\mathbf{x}^i)$.

Moreover,

1. with constant step size rule, f_{best}^k converges to within $G^2\gamma/2$ of optimal,
2. with square summable but not summable step size rule, $f_{best}^k \rightarrow f^*$, and
3. with nonsummable diminishing step size rule, $f_{best}^k \rightarrow f^*$.

Theorem 9 ([47], Theorem 2, Section 1.4). *Suppose $f(\mathbf{x})$ be differentiable on \mathbb{R}^n , ∇f is Lipschitz continuous with constant L , and $f(\mathbf{x})$ is strongly convex with constant μ . Let $\gamma_k = \gamma \forall k \in \mathbb{Z}_+^0$. Then for $0 < \gamma < 2/L$, the method (2.14) holds*

$$\|\mathbf{x}^k - \mathbf{x}^*\| \leq cq^k, \quad (2.17)$$

where $0 \leq q < 1$.

Theorem 9 indicates that the subgradient method (2.14) can converge to a unique global minimum point \mathbf{x}^* with the rate of geometric progression (*cf.* Section 2.2.6), when the function f is strongly convex and is with Lipschitz continuous gradients.

2.2.2.1 The projected subgradient method

The projected subgradient method is usually used in constrained optimization. Consider the optimization problem

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{2.18}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and the constraint set \mathcal{X} are convex. Then the projected subgradient method to solve the optimization problems of the form (2.18) is given by

$$\mathbf{x}^{k+1} = [\mathbf{x}^k - \alpha_k \mathbf{g}^k]_{\mathcal{X}}, \tag{2.19}$$

where \mathbf{g}^k is any subgradient of f at \mathbf{x}^k . Similar step size rules used under the subgradient method can also be used here with similar convergence results [58]. We note that the projected subgradient method is a one variation of the subgradient method (2.14). When $\mathcal{X} = \mathbb{R}^n$, the projected subgradient method is simply reduced to the basic subgradient method.

2.2.2.2 The Stochastic Subgradient Method

The stochastic subgradient method is usually used in stochastic optimization processes, where the random variables appear in the formulation of the optimization problem. Consider the unconstrained optimization problem (2.13). Then the stochastic subgradient method to solve (2.13) is given by

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \gamma_k \tilde{\mathbf{g}}^{(k)}, \tag{2.20}$$

where $\tilde{\mathbf{g}}^{(k)}$ is a noisy subgradient (*cf.* [66, Section 1]) of f at $\mathbf{x}^{(k)}$. Convergence results for the stochastic subgradient method (2.20) can be found in [66, Section 3].

Next, we will discuss two commonly used distributed methods in the literature, which are based on the subgradient methods. The first method is based on dual decomposition methods [49–52, 67–70] and the second method is an approach coalescing consensus algorithms with subgradient methods [3, 4, 71]. Dual decomposition methods are used when the problem (2.1) is separable while the consensus algorithms are used when the problem is not separable [1, Section 10].

2.2.2.3 Dual Decomposition Algorithms

Distributed algorithms using dual decomposition with subgradient methods to solve utility based resource allocation problems (*network utility maximization* (NUM) problems) were presented in [1, Section 10] and [49, 50, 70] (See also [51, 52, 67, 69]). An elegant review on NUM problems with applications can be found in [68]. In general, a NUM problem has the form (*cf.* [1, Section 10.2.3])

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) = \sum_{i=1}^S u_i(\mathbf{x}_i) \\ & \text{subject to} && \sum_{i \in \mathcal{S}_l} \mathbf{x}_i \leq c_l \quad \text{for all } l \in \mathcal{L} \\ & && \mathbf{x}_i \in I_i \quad \text{for all } i \in \mathcal{S}, \end{aligned} \tag{2.21}$$

where $\mathcal{S} = \{1, \dots, S\}$ represents the set of sources in the considered network, $\mathcal{L} = \{1, \dots, L\}$ denotes the set of undirected links, $x_i \in \mathbb{R}_+^0$ is the source rate, where $i \in \mathcal{S}$, the capacity that the link l has is denoted by c_l , where $l \in \mathcal{L}$, and $u_i : \mathbb{R}_+^0 \rightarrow \mathbb{R}_+^0$ is concave and an increasing utility function of source i , where $i \in \mathcal{S}$. Moreover, the set of sources that use the link l is denoted by $\mathcal{S}_l = \{i \in \mathcal{S} \mid l \in \mathcal{L}_i\}$, where $\mathcal{L}_i \subset \mathcal{L}$ denotes the set of links used by source i . Then, the dual function associated with the NUM problem (2.21) is given by

$$g(\boldsymbol{\lambda}) = \sum_{i=1}^S \max_{\mathbf{x}_i \in I_i} \{u_i(\mathbf{x}_i) - x_i \boldsymbol{\lambda}_i\} + \sum_{l=1}^L \boldsymbol{\lambda}_l c_l \tag{2.22}$$

, where $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_1 \dots \boldsymbol{\lambda}_L]^T$ and $\boldsymbol{\lambda}_i = \sum_{l \in \mathcal{L}_i} \boldsymbol{\lambda}_l$, $i \in \mathcal{S}$, where $\boldsymbol{\lambda}_l$ is the Lagrange multiplier corresponding to the link l .

The related dual problem is give by

$$\begin{aligned} & \text{minimize} && g(\boldsymbol{\lambda}) \\ & \text{subject to} && \boldsymbol{\lambda} \in \mathbb{R}_+^L. \end{aligned} \tag{2.23}$$

Then, the distributed subgradient method to solve the dual problem (2.23) yields the following steps, which provides price updates (Lagrange multiplier updates) and rate updates performed by links and sources respectively:

$$\text{Link Price Update: } \boldsymbol{\lambda}_l(k+1) = [\boldsymbol{\lambda}_l(k) + \gamma \mathbf{d}_l(k)]^+.$$

$$\text{Source Rate Update: } \mathbf{x}_i(k+1) = \mathbf{x}_i \in I_i \text{ argmax } \{u_i(\mathbf{x}_i) - \mathbf{x}_i \boldsymbol{\lambda}_i\}.$$

where $\mathbf{d}_l(k) = \sum_{i \in \mathcal{S}_l} \mathbf{x}_i(k) - c_l$ and k represents the iteration index.

The utility functions of NUM problems used in [49] and [50] are considered as increasing and strictly concave. Such strict concavity assumptions are accepted as the law of diminishing returns applies in practice [68]. Nonetheless, in most practically relevant examples, the utility functions can be considered as strongly concave [70]. Existing studies which have explored their work over NUM problems only using concave utility functions suffer from a slow rate of convergence $O(1/\sqrt{k})$ in subgradient methods. However, the authors in [70] have provided an improved rate of convergence $O(1/k)$ in primal variables to the optimal solution of the NUM problem using strongly concave utility functions. They have used the fast gradient method (*cf.* [72]) to solve the dual of the NUM problem, and the convergences in the primal variables are shown using strong duality assumptions.

2.2.2.4 Consensus Algorithms

In general, consensus algorithms are used to minimize the sum of non-separable convex functions corresponding to multiple agents connected over a network. The dual decom-

position algorithms based on the NUM framework (*cf.* Section 2.2.2.3) are limited to applications where the utility function of an individual agent depends only on the resource allocated to that agent. However, there are many applications where the agents' utility can depend on the entire resource allocation vector (*cf.* Section 1.2). Consensus algorithms are commonly used in such applications, where the related problem structures are usually centered around consensus type problems. In particular, the considered problem is [1, Section 10.3]

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^m f_i(\mathbf{x}) \\ & \text{subject to} && \mathbf{x} \in \mathbb{R}^n, \end{aligned} \tag{2.24}$$

where $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex and represents the local objective function (cost function) of agent i and $\mathbf{x} \in \mathbb{R}^n$ is a decision vector. The global consensus problem (1.1), which we have considered in Section 1.1, is equivalent to the preceding problem (2.24) when $\mathcal{X} = \mathbb{R}^n$. Let f^* and X^* denote the optimal value and the set of optimal solutions to the problem (2.24), respectively. Then, the distributed algorithm to solve the problem (2.24) is given by

$$\mathbf{x}_i(k+1) = \sum_{j=1}^m w_{ij}(k) \mathbf{x}_j(k) - \gamma \mathbf{d}_i(k); \quad i = 1, \dots, m, \tag{2.25}$$

where $\gamma > 0$ is a step size, and w_{ij} represents the weight that agent i assigns to the estimate \mathbf{x}^j receives from a neighboring agent j . The vector $\mathbf{d}_i(k)$ is a subgradient of the agent i 's objective function $f_i(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}_i(k)$. The convergence of the algorithm (2.25) was discussed in [71] (see also [1, Section 10.3]) by analyzing the behavior of a matrix given by

$$\phi(k, s) = A(k)A(k-1) \dots A(s+1)A(s), \tag{2.26}$$

where $A(k)$ is a matrix with the vector $a_i = [a_{i1}(k) \ a_{i2}(k) \ \dots \ a_{im}(k)]^T$ in its i th column and $k \geq s$. The matrix $\phi(k, s)$ is called the *transition matrix*.

2.2.2.5 Numerical Examples

We present two examples with numerical illustrations to provide a methodical exposition on the convergence of the gradient/subgradient method. Example 1 is based on the primal decomposition and the Dual decomposition approach is presented in Example 2.

Example 1 (Standard Gradient Method: Primal Decomposition Approach). *Consider an unconstrained minimization problem with two users;*

$$\text{minimize } f(\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}) = f_1(\mathbf{x}_1, \mathbf{y}) + f_2(\mathbf{x}_2, \mathbf{y}), \quad (2.27)$$

where $\mathbf{x}_1 \in \mathbb{R}^{n_1}$, $\mathbf{x} \in \mathbb{R}^{n_1}$, $\mathbf{x} \in \mathbb{R}^{n_2}$, and $f_1 : \mathbb{R}^{n_1} \rightarrow \mathbb{R}$ and $f_2 : \mathbb{R}^{n_2} \rightarrow \mathbb{R}$ are quadratic with the form $f_1(\mathbf{x}_1, \mathbf{y}) = [\mathbf{x}_1^T \ \mathbf{y}^T] \mathbf{A}_1 [\mathbf{x}_1^T \ \mathbf{y}^T]^T$ and $f_2(\mathbf{x}_2, \mathbf{y}) = [\mathbf{x}_2^T \ \mathbf{y}^T] \mathbf{A}_2 [\mathbf{x}_2^T \ \mathbf{y}^T]^T$. Here $\mathbf{A}_1 \in \mathbb{R}^{(n_1+n_2) \times (n_1+n_2)}$ and $\mathbf{A}_2 \in \mathbb{R}^{(n_1+n_2) \times (n_1+n_2)}$ are positive definite matrices. Let $(\mathbf{x}_1^*, \mathbf{x}_2^*, \mathbf{y}^*)$ and f^* denote the optimal solution and the optimal value of (2.27) respectively.

We use the gradient method (2.15) with primal decomposition (cf. Section 2.2.1.1) to solve (2.27). The subproblems associated with (2.27) are given by:

$$\text{Subproblem 1 : } S_1(\mathbf{y}) = \underset{\mathbf{x}_1}{\text{minimize}} [\mathbf{x}_1^T \ \mathbf{y}^T] \mathbf{A}_1 [\mathbf{x}_1^T \ \mathbf{y}^T]^T. \quad (2.28)$$

$$\text{Subproblem 2 : } S_2(\mathbf{y}) = \underset{\mathbf{x}_2}{\text{minimize}} [\mathbf{x}_2^T \ \mathbf{y}^T] \mathbf{A}_2 [\mathbf{x}_2^T \ \mathbf{y}^T]^T. \quad (2.29)$$

Then the master problem in primal decomposition is given by

$$\underset{\mathbf{y}}{\text{minimize}} S(\mathbf{y}) = S_1(\mathbf{y}) + S_2(\mathbf{y}). \quad (2.30)$$

Subproblem (2.28) and subproblem (2.29) can be solved analytically by simple calculations. Related optimal functions are given by

$$S_i(\mathbf{y}) = \mathbf{y}^T \mathbf{A}_{i4} \mathbf{y} + [(\mathbf{x}_i^*)^T \mathbf{A}_{i3}^T + (\mathbf{x}_i^*)^T \mathbf{A}_{i2}] \mathbf{y} + (\mathbf{x}_i^*)^T \mathbf{A}_{i1} \mathbf{x}_i^*; \quad i = 1, 2,$$

where $\mathbf{x}_i^* = \underset{\mathbf{x}_i}{\operatorname{argmin}} [\mathbf{x}_i^T \mathbf{y}^T] \mathbf{A}_i [\mathbf{x}_i^T \mathbf{y}^T]^T$ and $\mathbf{A}_i = \begin{bmatrix} \mathbf{A}_{i1} & \mathbf{A}_{i2} \\ \mathbf{A}_{i3} & \mathbf{A}_{i4} \end{bmatrix}$ with $\mathbf{A}_{i1} \in \mathbb{R}^{n_1 \times n_1}$, $\mathbf{A}_{i2} \in \mathbb{R}^{n_1 \times n_2}$, $\mathbf{A}_{i3} \in \mathbb{R}^{n_2 \times n_1}$, and $\mathbf{A}_{i4} \in \mathbb{R}^{n_2 \times n_2}$ for all $i = 1, 2$.

It can easily be observed that $S(y)$ [cf. equation (2.30)] is quadratic as $S_1(y)$ and $S_2(y)$ are quadratic. Clearly, the objective function $S(y)$ of the master problem is differentiable. Moreover, $S_1(y)$ and $S_2(y)$ are strongly convex since \mathbf{A}_{14} and \mathbf{A}_{24} are positive definite. Thus, $S(y)$ is also strongly convex with Lipschitz continuous gradients. Therefore, Theorem 9 ensures that the gradient method with a constant step size rule converges to the optimal point of the problem (2.30). We illustrate the convergence results numerically. The Related algorithm (cf. Algorithm 1) is presented below.

Algorithm 1 Standard gradient method: Primal decomposition

Require: $\mathbf{y}^0 \in \mathbb{R}^{n_2}$

- 1: $k = 0$.
 - 2: **repeat**
 - 3: Solve subproblems (2.28) and (2.28) in parallel with $\mathbf{y} = \mathbf{y}^{(k)}$ to yield $\mathbf{x}_1^{(k)}$ and $\mathbf{x}_2^{(k)}$. The solutions are;

$$\mathbf{x}_1^{(k)} = \underset{\mathbf{x}_1}{\operatorname{argmin}} [\mathbf{x}_1^T (\mathbf{y}^{(k)})^T] \mathbf{A}_1 [\mathbf{x}_1^T (\mathbf{y}^{(k)})^T]^T.$$

$$\mathbf{x}_2^{(k)} = \underset{\mathbf{x}_2}{\operatorname{argmin}} [\mathbf{x}_2^T (\mathbf{y}^{(k)})^T] \mathbf{A}_2 [\mathbf{x}_2^T (\mathbf{y}^{(k)})^T]^T.$$
 - 4: Compute: $\nabla S(\mathbf{y}^{(k)}) = \mathbf{A}_{11}^T \mathbf{x}_1^{(k)} + \mathbf{A}_{13} \mathbf{x}_1^{(k)} + 2\mathbf{A}_{14} \mathbf{y}^{(k)} + \mathbf{A}_{21}^T \mathbf{x}_2^{(k)} + \mathbf{A}_{23} \mathbf{x}_2^{(k)} + 2\mathbf{A}_{24} \mathbf{y}^{(k)}$
 - 5: \mathbf{y} variable update: $\mathbf{y}^{k+1} = \mathbf{y}^k + \gamma \nabla S(\mathbf{y}^k)$
 - 6: $k := k + 1$.
 - 7: **until** a stopping criterion true
-

In Algorithm 1, the step 4 follows because the gradient update is given by $\nabla S(\mathbf{y}^{(k)}) = \nabla S_1(\mathbf{y}^{(k)}) + \nabla S_2(\mathbf{y}^{(k)})$, where $\nabla S_1(\mathbf{y}^{(k)}) = \mathbf{A}_{12}^T \mathbf{x}_1^{(k)} + \mathbf{A}_{13} \mathbf{x}_1^{(k)} + 2\mathbf{A}_{14} \mathbf{y}^{(k)}$ and $\nabla S_2(\mathbf{y}^{(k)}) = \mathbf{A}_{22}^T \mathbf{x}_2^{(k)} + \mathbf{A}_{23} \mathbf{x}_2^{(k)} + 2\mathbf{A}_{24} \mathbf{y}^{(k)}$ [cf. equation (2.30)].

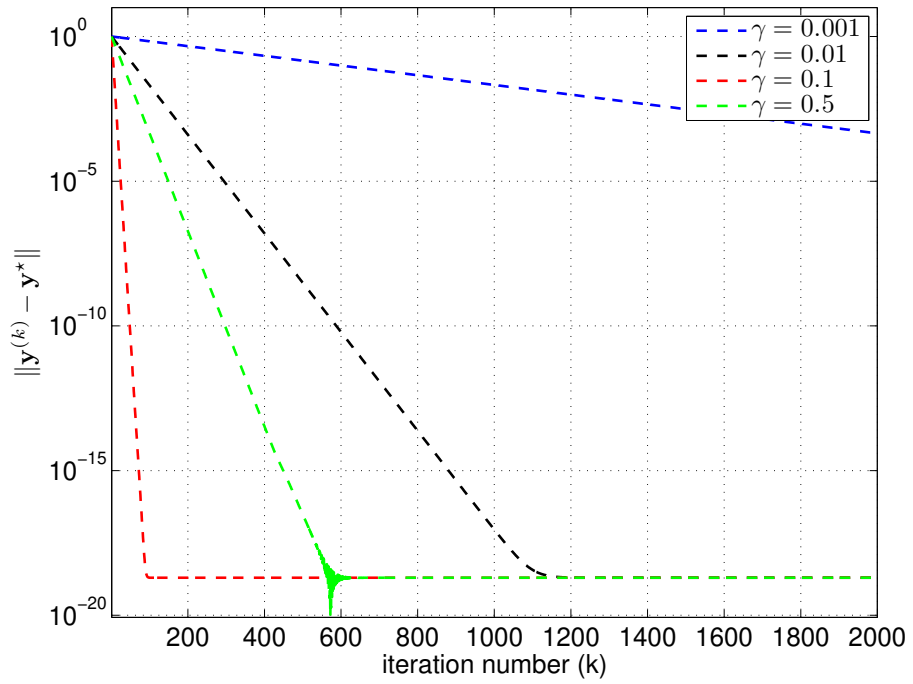


Figure 2.2: Standard gradient method with primal decomposition: Convergence of $y^{(k)}$ using different fixed step sizes.

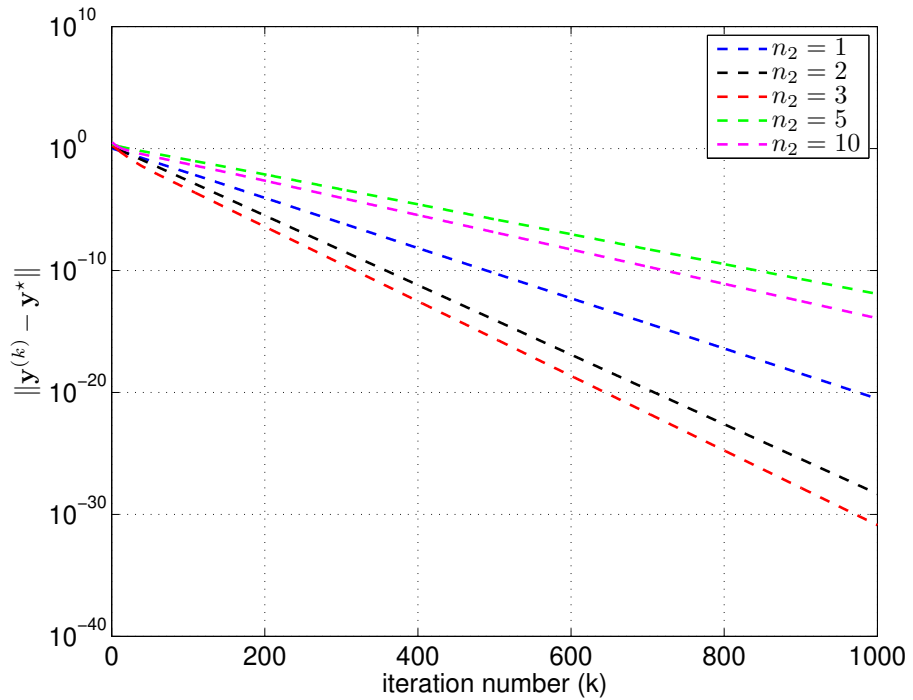


Figure 2.3: Standard gradient method with primal decomposition: Convergence of $y^{(k)}$ using different dimensions of y .

Figure 2.2 depicts the convergence of the complicating variable y^k using $n_1 = n_2 = 1$ for different fixed step sizes. The figure clearly shows linear convergence rates for

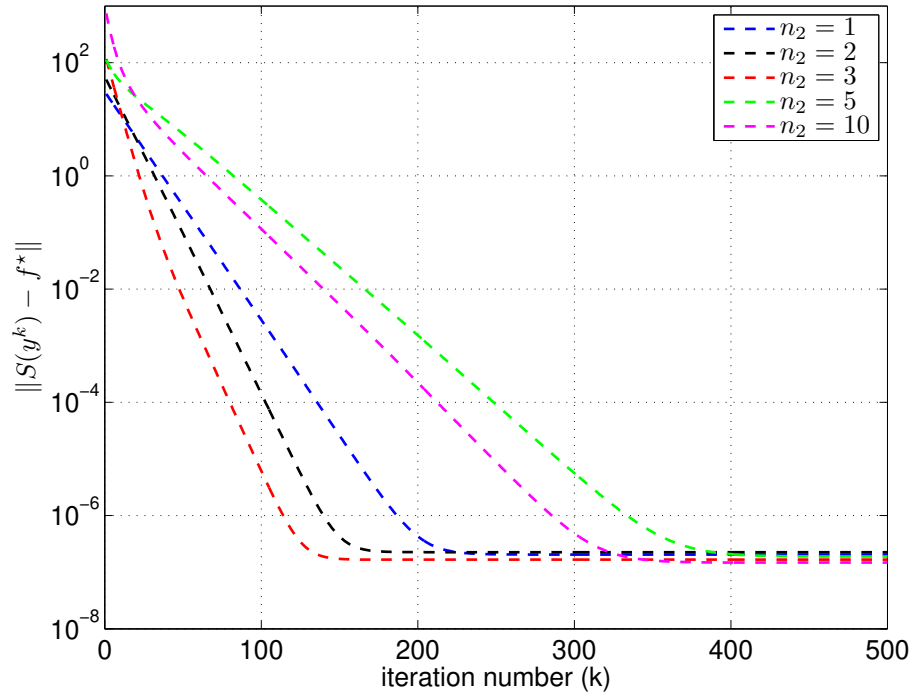


Figure 2.4: Standard gradient method with primal decomposition: Convergence of $S(y^{(k)})$ using different dimensions of y .

each step size. Moreover, results show slow convergence for relatively small step sizes. Figure 2.3 shows the convergence of $\mathbf{y}^{(k)}$ for different dimensions of the complicating variable \mathbf{y} using $n_1 = 10$, and with the step size $\gamma = 0.001$. Results show that the linear convergence of $\mathbf{y}^{(k)}$ to \mathbf{y}^* is guaranteed regardless of the dimension of \mathbf{y} . Finally, the convergence of $S(\mathbf{y}^k)$ [cf. equation (2.30)] for different dimensions of \mathbf{y} is depicted in Figure 2.4, again using $n_1 = 10$, and with the step size $\gamma = 0.001$. The figure shows $S(\mathbf{y}^k)$ converges to f^* linearly for each dimension.

Example 2 (The Basic Subgradient Method: Dual Decomposition Approach). *Consider a constrained minimization problem with two users:*

$$\begin{aligned}
 & \text{minimize} && f(\mathbf{x}_1, \mathbf{y}_1, \mathbf{x}_2, \mathbf{y}_2) = f_1(\mathbf{x}_1, \mathbf{y}_1) + f_2(\mathbf{x}_2, \mathbf{y}_2) \\
 & \text{subject to} && \mathbf{y}_1 = \mathbf{y}_2, \\
 & && \mathbf{x}_1 \in \mathbf{X}_1, \mathbf{x}_2 \in \mathcal{X}_2, y_1, y_2 \in \mathcal{Y},
 \end{aligned} \tag{2.31}$$

where $f_1(\mathbf{x}_1, \mathbf{y}_1) = \cosh(\mathbf{a}_1^T \mathbf{x}) + \mathbf{a}_2^T \mathbf{x}$ and $f_2(\mathbf{x}_2, \mathbf{y}_2) = \cosh(\mathbf{b}_1^T \mathbf{y}) + \mathbf{b}_2^T \mathbf{y}$ with $\mathbf{x} =$

$(\mathbf{x}_1, \mathbf{y}_1), \mathbf{y} = (\mathbf{x}_2, \mathbf{y}_2), \mathcal{X}_1 \subseteq \mathbb{R}^{n_1}, \mathcal{X}_2 \subseteq \mathbb{R}^{n_1}, \mathcal{Y} \subseteq \mathbb{R}^{n_2}$, and $\mathbf{a}_1, \mathbf{a}_2, \mathbf{b}_1, \mathbf{b}_2 \in \mathbb{R}^{(n_1+n_2)}$.

Here we use the subgradient method (2.14) with dual decomposition (*cf.* Section 2.2.1.2) to solve (2.31). The dual function corresponding to the primal problem (2.31) is given by [*cf.* (2.9)]

$$g(\boldsymbol{\lambda}) = \inf_{\substack{\mathbf{x}_1 \in \mathcal{X}_1, \mathbf{x}_2 \in \mathcal{X}_2 \\ \mathbf{y}_1, \mathbf{y}_2 \in \mathcal{Y}}} (f_1(\mathbf{x}_1, \mathbf{y}_1) + f_2(\mathbf{x}_2, \mathbf{y}_2) + \boldsymbol{\lambda}^T(\mathbf{y}_1 - \mathbf{y}_2)). \quad (2.32)$$

The subproblems associated with (2.32) are given by:

$$\text{Subproblem 1 : } g_1(\boldsymbol{\lambda}) = \inf_{\mathbf{x}_1 \in \mathcal{X}_1, \mathbf{y}_1 \in \mathcal{Y}} f_1(\mathbf{x}_1, \mathbf{y}_1) + \boldsymbol{\lambda}^T \mathbf{y}_1. \quad (2.33)$$

$$\text{Subproblem 2 : } g_2(\boldsymbol{\lambda}) = \inf_{\mathbf{x}_2 \in \mathcal{X}_2, \mathbf{y}_2 \in \mathcal{Y}} f_2(\mathbf{x}_2, \mathbf{y}_2) - \boldsymbol{\lambda}^T \mathbf{y}_2. \quad (2.34)$$

Then the dual problem associated with the primal problem (2.31) is given by

$$\text{maximize}_{\boldsymbol{\lambda} \in \mathbb{R}^n} g(\boldsymbol{\lambda}) = g_1(\boldsymbol{\lambda}) + g_2(\boldsymbol{\lambda}). \quad (2.35)$$

Clearly, the optimal solutions and the optimal value of the dual problem (2.35) can be determined by minimizing the negative dual function $-g(\boldsymbol{\lambda})$. We use $h(\boldsymbol{\lambda}) = -g(\boldsymbol{\lambda})$ for notational convenience. The equivalent minimization problem is given by

$$\text{maximize}_{\boldsymbol{\lambda} \in \mathbb{R}^n} h(\boldsymbol{\lambda}) = -g_1(\boldsymbol{\lambda}) - g_2(\boldsymbol{\lambda}). \quad (2.36)$$

We determine some characteristics of the dual function $g(\boldsymbol{\lambda})$ for clarity. We consider that $\mathcal{X}_1 = [-1, 0]$, $\mathcal{X}_2 = [1, 2]$, $\mathcal{Y} = [-2, 2]$, $\mathbf{a}_1 = [1 \ 1]^T$, $\mathbf{a}_2 = [3 \ -2]^T$, $\mathbf{b}_1 = [1 \ 1]^T$, $\mathbf{b}_2 = [-2 \ 5]^T$, and the variables $\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}_1$, and \mathbf{y}_2 are with $n_1 = n_2 = 1$. Figure 2.5 presents the graph of $g(\boldsymbol{\lambda})$ and it confirms the concavity of $g(\boldsymbol{\lambda})$ / convexity of $-g(\boldsymbol{\lambda})$ (*cf.* Remark 2). Moreover, it too confirms the nondifferentiability of $g(\boldsymbol{\lambda})$. Thus we use the subgradient method (2.14) to solve (2.36). The related subgradient algorithm is given below (*cf.* Algorithm 2). Note that in Algorithm 2, the difference $\mathbf{y}_2^k - \mathbf{y}_1^k$ used in the dual

variable update (*cf.* step 4 of Algorithm 2) represents a subgradient \mathbf{g}^k of $-g(\boldsymbol{\lambda})$ at $\boldsymbol{\lambda}^k$.

Algorithm 2 Subgradient method: Dual decomposition

Require: $\boldsymbol{\lambda}^0 \in \mathbb{R}^{n_2}$

1: $k = 0$.

2: **repeat**

3: Solve subproblems (2.33) and (2.33) in parallel with $\boldsymbol{\lambda} = \boldsymbol{\lambda}^{(k)}$ to yield $(\mathbf{x}_1^{(k)}, \mathbf{y}_1^{(k)})$ and $(\mathbf{x}_2^{(k)}, \mathbf{y}_2^{(k)})$. The solutions are;

$$(\mathbf{x}_1^{(k)}, \mathbf{y}_1^{(k)}) = \underset{\mathbf{x}_1 \in \mathcal{X}_1, \mathbf{y}_1 \in \mathcal{Y}}{\operatorname{argmin}} f_1(\mathbf{x}_1, \mathbf{y}_1) + \boldsymbol{\lambda}^{(k)} \mathbf{y}_1$$

$$(\mathbf{x}_2^{(k)}, \mathbf{y}_2^{(k)}) = \underset{\mathbf{x}_2 \in \mathcal{X}_2, \mathbf{y}_2 \in \mathcal{Y}}{\operatorname{argmin}} f_2(\mathbf{x}_2, \mathbf{y}_2) - \boldsymbol{\lambda}^{(k)} \mathbf{y}_2$$

4: Compute: $\mathbf{g}^{(k)} = \mathbf{y}_1^k - \mathbf{y}_2^k$.

5: Dual variable update: $\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \gamma_k \mathbf{g}^{(k)}$.

6: $k := k + 1$.

7: **until** a stopping criterion true

Figure 2.5 clearly shows that there exists an optimal point $\boldsymbol{\lambda}^*$ to the dual function $g(\boldsymbol{\lambda})$. Thus the set of optimal solutions \mathcal{X}^* to the problem (2.35) is nonempty. Moreover $\|\mathbf{g}^{(k)}\|$ is bounded because, $\|\mathbf{g}^{(k)}\| = \|\mathbf{y}_2^k - \mathbf{y}_1^k\| \leq 4$ as $\mathbf{y}_1, \mathbf{y}_2 \in \mathcal{Y} = [-2, 2]$. In further, we use the initialization $\boldsymbol{\lambda}^{(0)} = 1$, and the cvx solver produces that $\boldsymbol{\lambda}^* \approx 5.14$. It turns out that the distance to the dual optimal solution $\|\boldsymbol{\lambda}^{(0)} - \boldsymbol{\lambda}^*\| \approx 4.14$ and thus it is bounded. Therefore, all the assumptions required by the Theorem 8 are satisfied. Thus the convergence of the subgradient method (2.14) to the optimal value of the problem (2.36) is assured by Theorem 8. We illustrate the convergence results numerically.

Figure 2.6 depicts the convergence of negative dual function values $h(\boldsymbol{\lambda}^{(k)})$ for different fixed step sizes $\gamma_k = \gamma$. The figure shows higher the value of the step length, the higher the rate of convergence. Figure 2.7 illustrates the convergence of $h(\boldsymbol{\lambda}^{(k)})$ using $\gamma_k = 0.1$, $\gamma_k = 0.1/k$, and $\gamma_k = 0.1/\sqrt{k}$. Results demonstrate a slower rate of convergence for $\gamma_k = 0.1/k$ (square summable but not summable) and $\gamma_k = 0.1/\sqrt{k}$ (nonsummable di-

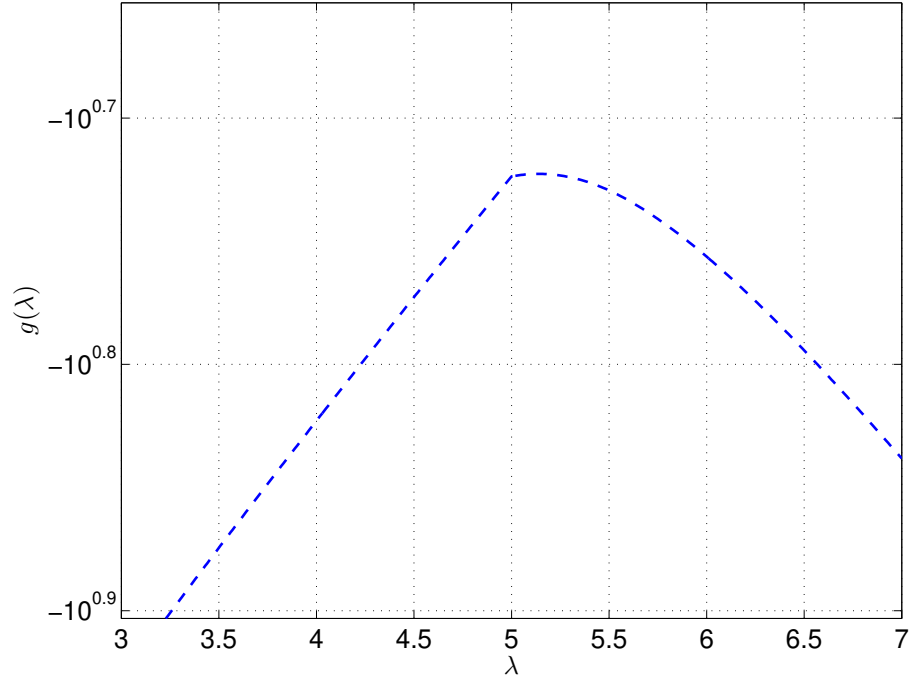


Figure 2.5: The subgradient method with dual decomposition: The graph of the dual function $g(\lambda)$ corresponding to the primal problem (2.31).

minishing) than that of using the constant step size $\gamma_k = \gamma$. Figure 2.8 and Figure 2.9 show convergences of corresponding dual variable iterates. Results demonstrate similar behaviors as that of Figure 2.6 and Figure 2.7.

Figures 2.6 - 2.9 only show the convergence in the dual domain. However, the convergence in the primal domain is of utmost importance as our focus is to solve the primal problem (2.31). In general, the iterates $\mathbf{y}_1^{(k)}$ and $\mathbf{y}_2^{(k)}$ are not feasible (i.e., $\mathbf{y}_1^{(k)} \neq \mathbf{y}_2^{(k)}$). Thus, at each iteration in the Algorithm 2, a feasible point $\bar{\mathbf{y}}$ is obtained by averaging the solutions of the subproblems (2.33) and (2.34), i.e., $\bar{\mathbf{y}}^{(k)} = (\mathbf{y}_1^{(k)} + \mathbf{y}_2^{(k)})/2$ [61, Section 2]. Then at each iteration, the function value f is calculated at $(\mathbf{x}_1^{(k)}, \mathbf{x}_2^{(k)}, \bar{\mathbf{y}}^{(k)})$ [cf. (2.31)]. Figure 2.10 shows the convergence of $f(\mathbf{x}_1^{(k)}, \mathbf{x}_2^{(k)}, \bar{\mathbf{y}}^{(k)}) = f_1(\mathbf{x}_1^{(k)}, \bar{\mathbf{y}}^{(k)}) + f_2(\mathbf{x}_2^{(k)}, \bar{\mathbf{y}}^{(k)})$ and $g(\boldsymbol{\lambda}^{(k)})$ using $\gamma_k = 0.1$. Moreover, the optimal value f^* of the primal problem (2.31) is also presented in the same figure. Results clearly show that both $f(\mathbf{x}_1^{(k)}, \mathbf{x}_2^{(k)}, \bar{\mathbf{y}}^{(k)})$ and $g(\boldsymbol{\lambda}^{(k)})$ converge to f^* . Thus the feasible points obtained by solving the subproblems (2.33) and (2.34) in the dual decomposition are guaranteed to converge towards the optimal solution of the primal problem (2.31).

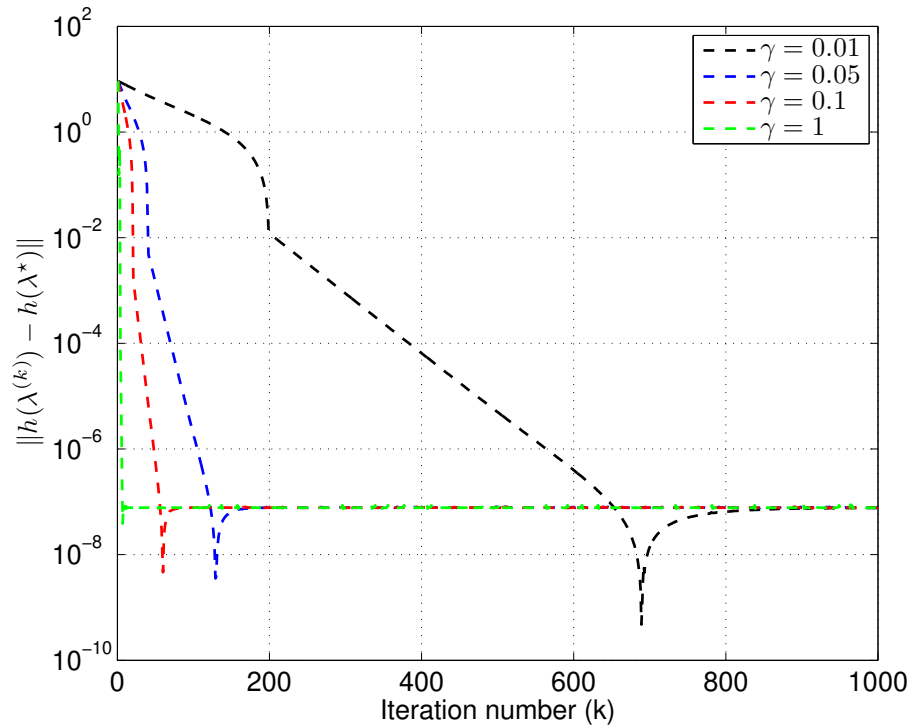


Figure 2.6: The subgradient method with dual decomposition: Convergence of dual function values using different fixed step sizes.

2.2.3 Alternating Direction Method of Multipliers (ADMM)

ADMM is a simple but powerful method that is used in distributed convex optimization [56]. It can be viewed as a variant of augmented Lagrangian and method of multipliers with the blend of dual decomposition. Consider the equality constrained optimization problem

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{Ax} = \mathbf{b}, \end{aligned} \tag{2.37}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex. The Lagrangian for (2.37) is given by

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T(\mathbf{Ax} - \mathbf{b}), \tag{2.38}$$

where $\boldsymbol{\lambda}$ denotes the dual variable. Then, the problem (2.37) usually can be solved using the dual subgradient method (i.e., the subgradient method implemented on the dual

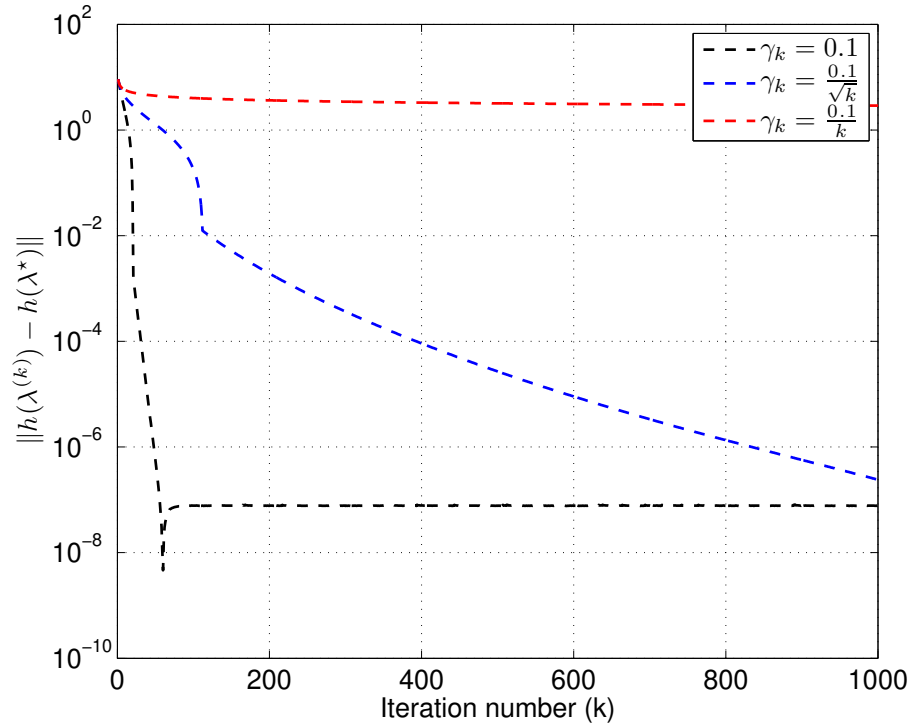


Figure 2.7: The subgradient method with dual decomposition: Convergence of dual function values using constant, nonsummable diminishing, and square summable but not summable step size rules.

domain) [cf. equation (2.14)] with steps

$$\mathbf{x}^{(k+1)} = \underset{\mathbf{x} \in \mathbb{R}^n}{\operatorname{argmin}} L(\mathbf{x}, \boldsymbol{\lambda}^{(k)}) \quad (2.39)$$

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \gamma_k (\mathbf{A}\mathbf{x}^{(k+1)} - \mathbf{b}), \quad (2.40)$$

where $\gamma_k > 0$ is a suitably chosen step size (cf. Theorem 8).

The augmented Lagrangian for the problem (2.37) is given by

$$L_p(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T (\mathbf{A}\mathbf{x} - \mathbf{b}) + (p/2) \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2, \quad (2.41)$$

where $p > 0$ is called the penalty parameter [56, Section 2.3]. Clearly, the augmented

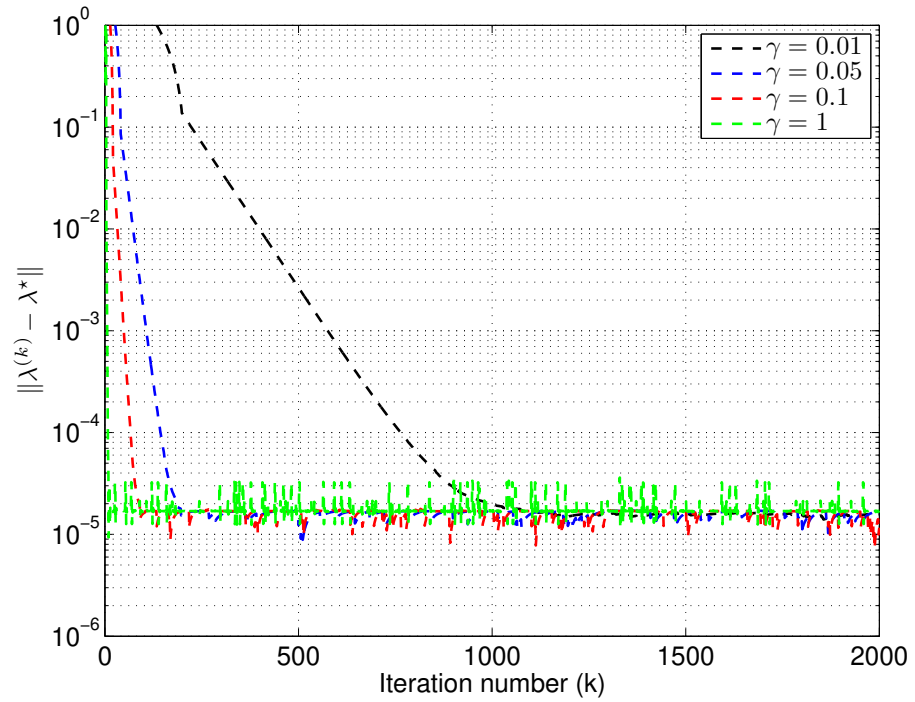


Figure 2.8: The subgradient method with dual decomposition: Convergence of dual variable iterates using different fixed step sizes.

Lagrangian (2.41) is the Lagrangian associated with the problem

$$\begin{aligned}
 &\text{minimize} && f(\mathbf{x}) + (p/2)\|\mathbf{Ax} - \mathbf{b}\|^2 \\
 &\text{subject to} && \mathbf{Ax} = \mathbf{b},
 \end{aligned} \tag{2.42}$$

and this problem is equivalent to the original problem (2.37). The associated dual function is

$$g_p(\boldsymbol{\lambda}) = \inf_{\mathbf{x}} L_p(\mathbf{x}, \boldsymbol{\lambda}). \tag{2.43}$$

Then the related dual problem is

$$\begin{aligned}
 &\text{maximize} && g_p(\boldsymbol{\lambda}) \\
 &\text{subject to} && \boldsymbol{\lambda} \in \mathbb{R}^n.
 \end{aligned} \tag{2.44}$$

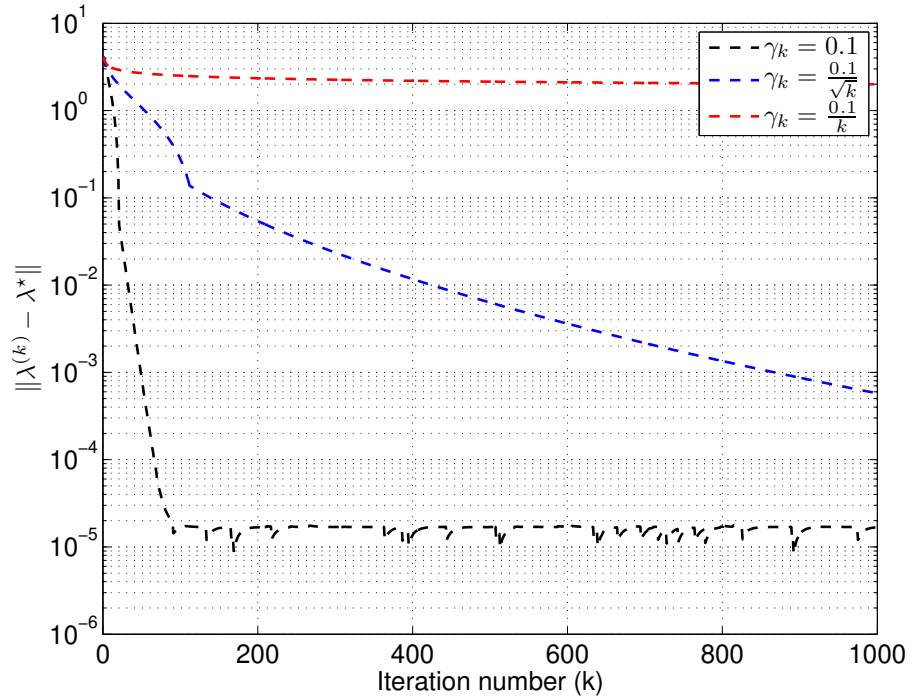


Figure 2.9: The subgradient method with dual decomposition: Convergence of dual variable iterates using different step size rules.

The problem (2.44) can solve using the subgradient method with steps

$$\mathbf{x}^{(k+1)} = \underset{\mathbf{x} \in \mathbb{R}^n}{\operatorname{argmin}} L_p(\mathbf{x}, \boldsymbol{\lambda}^{(k)}) \quad (2.45)$$

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + p(\mathbf{A}\mathbf{x}^{(k+1)} - \mathbf{b}). \quad (2.46)$$

The steps (2.45) and (2.46) are known as the method of multipliers for solving the problem (2.37). The only difference in the method of multipliers [cf. steps (2.45) and (2.46)] compared with the dual subgradient method [cf. steps (2.39) and (2.39)] is, the method of multipliers uses the augmented Lagrangian in the x minimization step [cf. step (2.45)] and the penalty parameter p is used as the step size instead γ_k . However, the method of multipliers is considered as a method of robusting the dual subgradient method as it converges under more general conditions compared to the dual subgradient method [56]. howbeit the method of multipliers cannot be used for decomposition even if the primal function $f(\mathbf{x})$ [cf. (2.37)] is separable, because, the augmented Lagrangian is not separable. To address this issue, The ADMM has been introduced [73], a method that

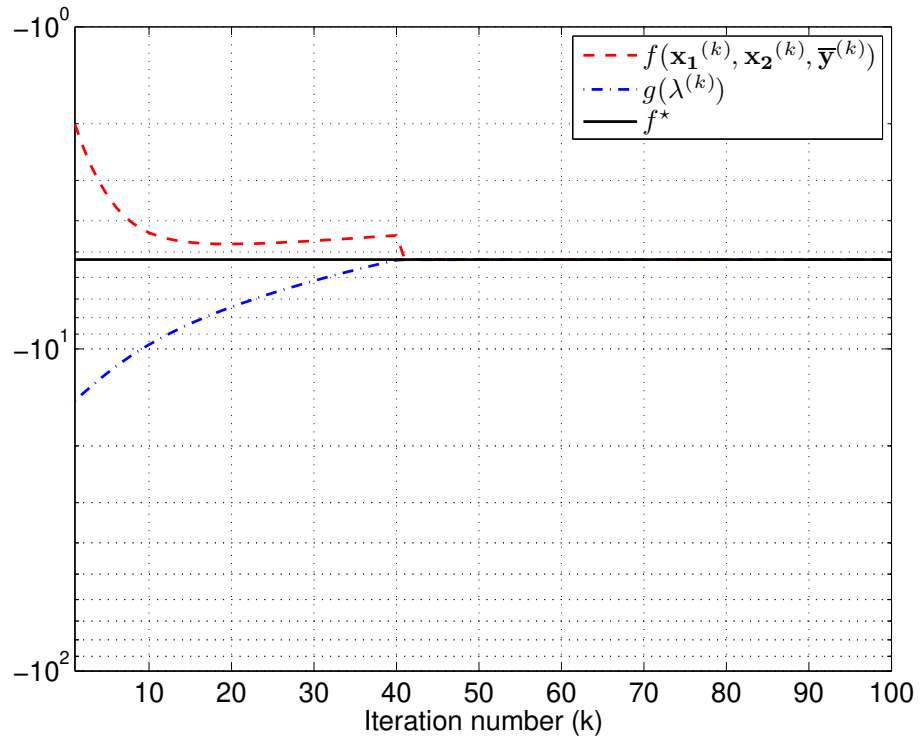


Figure 2.10: The subgradient method with dual decomposition: Convergence of dual function values and primal function values evaluated at feasible points. Solid line demonstrates the optimal value f^* of the primal problem (2.31).

can be viewed as a variant of the so called method of multipliers with the blend of dual decomposition.

In general, the problem in ADMM takes the form

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) + g(\mathbf{y}) \\ & \text{subject to} && \mathbf{Ax} + \mathbf{By} = \mathbf{c}, \end{aligned} \tag{2.47}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$, $\mathbf{A} \in \mathbb{R}^{q \times n}$, $\mathbf{B} \in \mathbb{R}^{q \times m}$, and $\mathbf{c} \in \mathbb{R}^q$. Moreover $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $g : \mathbb{R}^m \rightarrow \mathbb{R}$ are convex functions. The problem (2.47) is a variant of the problem (2.37), where the variable x in (2.37) is split into two parts x and y in (2.47).

The augmented Lagrangian for (2.47) is given by

$$L_p(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T(\mathbf{Ax} + \mathbf{By} - \mathbf{c}) + (p/2)\|\mathbf{Ax} + \mathbf{By} - \mathbf{c}\|^2, \tag{2.48}$$

where $p > 0$ is the penalty parameter.

Then the ADMM Algorithm to solve (2.47) is given below (*cf.* Algorithm 3).

Algorithm 3 Alternating Direction Method of Multipliers (ADMM)

Require: $\mathbf{y}^{(0)} \in \mathbb{R}^m$ and $\boldsymbol{\lambda}^{(0)} \in \mathbb{R}^q$

1: $k = 0$.

2: **repeat**

3: \mathbf{x} minimization step:

$$\mathbf{x}^{(k+1)} = \underset{\mathbf{x}}{\operatorname{argmin}} L_p(\mathbf{x}, \mathbf{y}^k, \boldsymbol{\lambda}^k).$$

4: \mathbf{y} minimization step:

$$\mathbf{y}^{(k+1)} = \underset{\mathbf{y}}{\operatorname{argmin}} L_p(\mathbf{x}^*, \mathbf{y}, \boldsymbol{\lambda}^k)$$

5: Dual variable update:

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + p(\mathbf{A}\mathbf{x}^{(k+1)} + \mathbf{B}\mathbf{y}^{(k+1)} - \mathbf{c}). \quad (2.49)$$

6: $k := k + 1$.

7: **until** a stopping criterion true

In general, ADMM can produce slow convergence to achieve high accuracy. Albeit, ADMM is often useful practically when modest accuracy is sufficient [56]. In particular, this is indeed the case in many kinds of large-scale problems we consider in a variety of real life applications. We refer the readers [56] to get a thorough exposition including the convergence properties, extensions, and variations of ADMM. Further, many other properties of ADMM can be found in [74–76].

2.2.4 Proximal Gradient Method

The *proximal gradient method* is a proximal algorithm [59] that is especially well-suited for large-scale distributed convex problems. The base point in proximal algorithms is

evaluating a *proximal operator* of a function.

Definition 21 (Proximal operator). *Let $f : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ be a closed proper convex function (cf. Definition 7). Then the operator $\text{prox}_f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ s.t.,*

$$\text{prox}_f(\mathbf{y}) = \underset{\mathbf{x}}{\text{argmin}} \left(f(\mathbf{x}) - \frac{1}{2} \|\mathbf{x} - \mathbf{y}\|^2 \right)$$

is called the proximal operator of f .

The proximal operator of the scaled function γf , where $\gamma > 0$, is given by

$$\text{prox}_{\gamma f}(\mathbf{y}) = \underset{\mathbf{x}}{\text{argmin}} \left(f(\mathbf{x}) - \frac{1}{2\gamma} \|\mathbf{x} - \mathbf{y}\|^2 \right).$$

This is usually called the proximal operator of f with parameter γ .

Consider the problem

$$\text{minimize } f(\mathbf{x}) + g(\mathbf{x}), \tag{2.50}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $g : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ are closed, proper, and convex. Moreover, f is differentiable. Then the proximal gradient method to solve the problems of the form (2.50) is

$$\mathbf{x}^{(k+1)} = \text{prox}_{\gamma_k g}(\mathbf{x}^{(k)} - \gamma_k \nabla f(\mathbf{x}^{(k)})), \tag{2.51}$$

where k denotes the iteration index and γ_k represents a step size. We note that the proximal gradient method reduces to the standard gradient method when $g = 0$ [cf. (2.15)].

The proximal gradient method is beneficial in many aspects.

1. The algorithm smoothly works when the underlying objective functions are nondifferentiable.
2. Well-suited for solving large-scale distributed optimization problems.
3. Challenging problems can be solved efficiently if the proximal operators for underlying functions are quickly evaluable.

When the function f is with Lipschitz continuous gradients, the proximal gradient method (2.51) with constant step size $\gamma_k = \gamma \in (0, L]$ can converge with a rate $O(1/k)$, where L denotes the gradient Lipschitz constant [59, Section 4.2].

2.2.5 Dual Averaging

Consider an optimization problem of the form (cf. [60])

$$\begin{aligned} & \text{minimize} && \frac{1}{m} \sum_{i=1}^m f_i(\mathbf{x}) \\ & \text{subject to} && \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{2.52}$$

where $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex and Lipschitz continuous (cf. Definition 14) on \mathcal{X} , for all $i = 1, \dots, m$. The set \mathcal{X} is closed and convex. The optimization problem (2.52) is based on functions that are distributed over a network. Let $G(V, E)$ be an undirected graph over a set of nodes $\mathcal{V} = \{1, \dots, m\}$ and a set of edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. The local objective function f_i is only known to agent i associated with the node i , and each agent i can communicate only with its immediate neighbors $j \in \mathcal{N}(i) = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}$. Without loss of generality, assume that $\mathbf{0} \in \mathcal{X}$. Then, the *dual averaging* scheme for solving an optimization problem of the form (2.52) is based on a proximal function $\phi : \mathcal{X} \rightarrow \mathbb{R}$, which is assumed to be strongly convex with constant l (cf. Definition 3). Moreover, suppose that $\phi(\mathbf{x}) \geq 0, \forall \mathbf{x} \in \mathcal{X}$, and $\phi(\mathbf{0}) = 0$. Then, the standard dual averaging algorithm to solve the problem (2.52) is with updates

$$\mathbf{z}^{(k+1)} = \mathbf{z}^{(k)} + \mathbf{g}^{(k)} \tag{2.53}$$

$$\mathbf{x}^{(k+1)} = \arg \min_{\mathbf{x} \in \mathcal{X}} \left\{ (\mathbf{z}^{(k+1)})^T \mathbf{x} + \frac{1}{\alpha_k} \phi(\mathbf{x}) \right\}, \tag{2.54}$$

where the sequence of iterates $\{\mathbf{x}^{(k)}, \mathbf{z}^{(k)}\}_{k=0}^{\infty}$ contained within $\mathcal{X} \times \mathbb{R}^n$, $\mathbf{g} \in \partial f(\mathbf{x}^{(k)})$, and α_k is a non-increasing step size.

Next, to obtain a distributed solution method, at each iteration k , where $k \in \mathbb{Z}_+$, each

node i in the algorithm maintains a pair of vectors $(\mathbf{x}^{(k)}, \mathbf{z}^{(k)}) \in \mathcal{X} \times \mathbb{R}^n$ and computes a subgradient $\mathbf{g}_i \in \partial f_i(\mathbf{x}^{(k)})$ while receiving information $z_j^{(k)}$ from its neighboring nodes $j \in \mathcal{N}(i)$. Then, the node i 's update of the currently estimated solution $x_i^{(k+1)}$ is computed using a weighted average of $z_j^{(k)}$, where $j \in \mathcal{N}(i)$. Let $\mathbf{A} \in \mathbb{R}^{m \times m}$ be a matrix of non-negative weights that represent the structure of graph G . Moreover, assume that the matrix \mathbf{A} is a doubly stochastic matrix, s.t.,

$$\begin{aligned} \sum_{j=1}^m \mathbf{A}_{ij} &= \sum_{j \in \mathcal{N}(i)} \mathbf{A}_{ij} = 1 \quad \forall i \in \mathcal{V}, \quad \text{and} \\ \sum_{i=1}^m \mathbf{A}_{ij} &= \sum_{i \in \mathcal{N}(j)} \mathbf{A}_{ij} = 1 \quad \forall j \in \mathcal{V}. \end{aligned}$$

Then the *distributed dual averaging method* consists of the following updates:

$$\begin{aligned} \mathbf{z}_i^{(k+1)} &= \sum_{j \in \mathcal{N}(i)} \mathbf{A}_{ji} \mathbf{z}_j^{(k)} + \mathbf{g}_i^{(k)} \\ \mathbf{x}_i^{(k+1)} &= \arg \min_{\mathbf{x} \in \mathcal{X}} \left\{ (\mathbf{z}_i^{(k+1)})^\top \mathbf{x} + \frac{1}{\alpha_k} \phi(\mathbf{x}) \right\}. \end{aligned}$$

The convergence properties of the local variable \mathbf{x}_i are analyzed in [60] using the running local average $\hat{\mathbf{x}}_i(T) = (1/T) \sum_{k=1}^T \mathbf{x}_i^{(k)}$, where T is the number of iterations.

2.2.6 Classification of Convergence Rates

The value of any practically relevant optimization method (iterative methods) relies on its convergence properties, which determine the convergence behavior of the underlying algorithm towards an optimal solution to the considered optimization problem. Usually, to classify among different iterative methods, their rates of convergences are of utmost importance. In general, convergence rates are expressed with respect to some convenient error functions. Let us illustrate this with an unconstrained optimization problem of the

form

$$\begin{aligned} & \text{minimize} && f(\mathbf{x}) \\ & \text{subject to} && \mathbf{x} \in \mathbb{R}^n, \end{aligned} \tag{2.55}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex and differentiable. Suppose \mathcal{X}^* and f^* denote the set of optimal solutions and the optimal value of the problem (2.55) respectively. Then, the most common error functions used in the literature to determine the rates of convergences are

$$u(\mathbf{x}^{(k)}) = \|f(\mathbf{x}^{(k)}) - f(\mathbf{x}^*)\|_2 \tag{2.56}$$

$$u(\mathbf{x}^{(k)}) = \|\mathbf{x}^{(k)} - \mathbf{x}^*\|_2 \tag{2.57}$$

$$u(\mathbf{x}^{(k)}) = \|\nabla f(\mathbf{x}^{(k)})\|_2. \tag{2.58}$$

In some cases, the rates of convergences are established by tracking the minimum values of error functions (2.56), (2.57), and (2.58) over iterations (See Lemma 8). Moreover, the error functions (2.56) and (2.58) (or their minimum values) are preferred over (2.57) in the absence of strong convexity of the objective function (*cf.* Lemma 8).

Remark 10. *It is worth noting that in Lemma 8, the objective function associated with the error function is the negative dual function h . Although the primal objective function is strongly convex (*cf.* Assumption 4.1.1 and Lemma 4), the associated negative dual function h is only with Lipschitz continuous gradients (*cf.* Proposition 1).*

In general, with specific conditions, the error function $u(\mathbf{x}^{(k)}) \rightarrow 0$ as $k \rightarrow \infty$, under ideal settings, when $\mathbf{x}^{(k)} \rightarrow \mathbf{x}^*$, where $\mathbf{x}^* \in \mathcal{X}^*$ (See Theorem 8: 2), 3), and Theorem 9). However, either with some specific conditions or under nonideal settings, $u(\mathbf{x}^{(k)})$ can converge to a neighborhood around 0. In such cases, the sequences $\{f(\mathbf{x}^{(k)})\}$, $\{\mathbf{x}^{(k)}\}$, or $\{\|\nabla f(\mathbf{x}^{(k)})\|\}$ converge to a neighborhood around the optimal value [*cf.* (2.56)], optimal solution [*cf.* (2.57)], or 0 [*cf.* (2.58)].

Next, we will classify the different classes of rates of convergences of the error sequence $\{u(\mathbf{x}^{(k)})\}$. In each class, we assume that $u(\mathbf{x}^{(k)}) \rightarrow 0$ for clarity. However, the

classification is still valid as it is, when $u(\mathbf{x}^{(k)}) \not\rightarrow 0$. The difference is, that only a positive quantity is added to the right hand side of the respective inequality. Albeit, the rate of convergence of the respective algorithm remained unchanged [see Corollary 2: *cf.* equation (4.130), Corollary 3, and Corollary 4: *cf.* equation (4.93)].

2.2.6.1 Linear Convergence Rate

A sequence $\{u(\mathbf{x}^{(k)})\}$ converges to zero linearly, if $\exists q \in (0, 1)$ and $a \in \mathbb{R}_+$ s.t.,

$$u(\mathbf{x}^{(k)}) \leq aq^k, \quad \forall k \in \mathbb{Z}_+^0. \quad (2.59)$$

Here, the constant q is called the convergence ratio and it is the principal factor that determines the linear convergence rate (*cf.* Corollary 3). The linear convergence is also known as the geometric convergence.

2.2.6.2 Sublinear Convergence Rate

In general, the sequences which do not converge linearly are known to be convergent sublinearly. In particular, Sublinearly convergent sequences include

$$u(\mathbf{x}^{(k)}) \leq \frac{a}{k^p}, \quad \forall k \in \mathbb{Z}_+^0, \text{ where } a \in \mathbb{R}_+ \text{ and } p \in (0, \infty). \quad (2.60)$$

See Corollary 2: 2) and Corollary 4: 2) for sublinear convergences.

2.2.6.3 Superlinear Convergence Rate

A sequence $\{u(\mathbf{x}^{(k)})\}$ is called converges superlinearly to zero, if it converges linearly with any convergence ratio $q \in (0, 1)$.

2.2.6.4 Quadratic Convergence Rate

A sequence $\{u(\mathbf{x}^{(k)})\}$ is called converges quadratically to zero if $\exists a \in \mathbb{R}_+$ s.t.,

$$u(\mathbf{x}^{(k+1)}) \leq au(\mathbf{x}^{(k)})^2 \quad \forall k \in \mathbb{Z}_+^0. \quad (2.61)$$

2.3 Challenges

Distributed optimization methods have received much recent interest in many application fields due to their potential advantages in scalability, cybersecurity, flexibility, privacy, and robustness compared to centralized methods. However, inevitable system-specific challenges such as limited computational power, limited communication, latency requirements, measurement errors, and noises in wireless channels impose restrictions on the applicability of underlying distributed algorithms in pure form.

In particular, distributed optimization techniques that are used to solve large-scaled distributed problems heavily depend on the exchange of information among various agents (or subsystems) [77]. Then, the challenge in distributed operations arises from the communication structure used in such networks. Usually, underlying communication networks have limited bandwidths in practice and thus, the perfect communication between subsystems is not possible ¹. Further, distributed optimization platforms require more communication infrastructure as the underlying large-scaled systems consist of a large number of agents. More importantly, the communication scheme is directly involved in the performance of distributed algorithms which determines the convergence guarantees of underlying distributed algorithms [25, 27, 28].

However, in addition to imperfect communication, computational errors also exist when individual subsystems solve their local subproblems in distributed optimization [24, 26]. More specifically, the solutions to local subproblems usually deviate from the exact optimal solutions, depending on the error tolerances of solvers and the type of the

¹Typically, the subsystems in the communication network exchange quantized information among subsystems

problem. In general, the accumulation of subproblem errors in iterative algorithms may cause to build a considerable error, which directly affects the convergence properties of underlying algorithms. Moreover, approximation errors [20,21], noise induced in wireless settings [2], and measurement errors are some other challenges in distributed optimization that impact the exactness of underlying distributed algorithms.

2.3.1 Distributed Optimization over Non-ideal Settings

Distributed methods have been extensively analyzed in many works under ideal settings [78–81]. Moreover, convergence properties of exact algorithms are thoroughly analyzed under both constant and nonsummable step size rules in the literature (*cf.* Chapter 2.2). However, as we discussed in the preceding section, it restricts the application of exact algorithms in many real world applications. Thus, the analysis of distributed algorithms under non-ideal settings has been an appealing area of study [11–28]. An elegant discussion on the influence of noise in subgradient methods can be found in [11] under both differentiable and nondifferentiable settings (see [11, section 4, and section 5.5]). More importantly, [11] provides a repository of techniques that can serve as building blocks that are indispensable when analysing algorithms with imperfections. Algorithms based on combining consensus algorithms with subgradient methods (*cf.* section 2.2.2.4) under nonideal settings have been discussed in [12–14]. The key idea of this type of algorithms is to align the primal variables of each subsystem with its neighbors, followed by a local update of the variables at each subsystem aiming to minimize its own cost function. When the subsystems are communicating their variables to neighbors for aligning the iterates, either a deterministic or dynamic quantization of primal variables is considered. Under assumptions such as uniform boundedness of subgradients, among others, error bounds for suboptimality are derived in [12]. The boundedness assumption might restrict the range of applicability of the methods. For example, in many applied fields, it is now commonplace to form the objective function with a quadratic regularization term, where the bounded assumption is no longer affirmative. The quantization mechanism consid-

ered in [13] incorporates a zooming-in strategy that underlies an asymptotic decay of the quantization errors. Consequently, the convergence of the algorithm to the optimal value is guaranteed, where assumptions such as, uniform boundedness of subgradients have again been used. A similar dynamic quantization mechanism is adopted in [14], where the resulting quantization errors are again diminishing. Together with stronger assumptions, authors claim, not only the optimality but also better convergence rates. Although a diminishing error is favorable from a standpoint of establishing desirable convergences, it cannot capture common scenarios where the error is persistent throughout the iterations of the algorithm, e.g., measurement errors.

Inexact gradient methods are analysed in [15–23]. These methods are directly related to dual decomposition methods. It is usually the case that subgradient type algorithms are used to solve dual problems in a dual decomposition setting. The effect of noise in subgradient type methods has been discussed in [15, 16] with *compact* constraint sets. From a distributed optimization standpoint with dual decomposition, compactness is a restriction, because constraint sets appearing in dual-domain usually turn out to be noncompact. Authors in [17] have discussed the convergence properties of gradient methods with inexact gradients using a general continuously differentiable function (possibly nonconvex) with Lipschitz continuous gradients. The errors are assumed to be diminishing and dependent on the current step size and the exact gradient. However, such impositions seem like a restriction in practice as we have already pointed out in the preceding discussion. The influence of deterministic and bounded errors in gradients is considered in [18], where the feasible set is assumed to be compact. Recall that compactness assumption is often too restrictive for dual decomposition. The exposition given in [19] includes an inexact first-order oracle, based on what the behavior of several first-order methods under nonideal conditions has been analyzed. Despite its generality, an approximate subgradient is not necessarily be fleshed from the accompanying structure of the inexact oracle, unless the underlying constrain set is bounded. Subgradient errors, from a machine learning context, are considered in [20]. Errors are modeled from a stochastic standpoint and are assumed

to be biased and consistent. Biasedness can directly be linked with bounded errors from a deterministic point of view. Roughly speaking, the notion of consistency is dictated by a choice of a stochastic limit that decreases exponentially with the sample size. The latter assumption appears to be meaningful, especially in machine learning settings. However, in a distributed optimization setting with dual decomposition, one has to be cautious. Unlike a learning setting, in which regulating errors by an adequate choice of the sample size is at our disposal, in the dual decomposition setting the underlying errors are not necessarily controllable. Authors in [21] studied a class of problems where the classic stochastic optimization is a special case. Their assertions rely on assumptions such as a bounded second moment of the approximate subgradients. Errors in dual decomposition settings may not fit there, because the assumption can directly impose a requirement on the norm of the true subgradients. References [22, 23] discuss means of modeling inexactness of subgradients again from stochastic/deterministic points of view. They appear to be readily applied in a distributed optimization setting with dual decomposition.

Inexact gradient methods within the dual machinery are discussed in [24–28]. Rather than exploiting duality for enabling distributed optimization, authors in [24] uses it in a centralized setting. Only a set of inequality constraints are dualized, assuming that the other constraints are efficiently handled. The related Lagrangian minimization is considered to be inexact, which in turn leads to an inexact subgradient method for the dual variable update. Convergence results associated with *primal feasible* points are asserted for a very special case of a model predictive control problem, but not in general. Problems of the form of sharing are considered in [25, 26]. Dual decomposition is applied to decouple the problem considered in [25], where the related dual function turns out to be defined on scalars. The dual gradient, which is a scalar, is quantized, where an additional assumption on the compactness of the dual-domain is artificially imposed for tractability. The authors yield the *feasibility* of the primal points returned by their algorithms by restricting the updates of the dual variable to lie in a region with positive dual gradients. In a more general sharing problem, for example, a dual function defined on vectors, one has

to be cautious, however, because the technique adopted in [25] does not apply directly. A more general formulation is considered in [26], despite impositions on the structure of problem data that enable the dual decomposition. An averaging of primal variables is introduced that appears to minimize the primal feasibility violation. However, unlike [25], the primal feasibility of the iterates is not guaranteed in [26]. References [27, 28] discuss gradient methods, together with their applications to distributed optimization under limited communication settings. The inexactness of the algorithms is solely due to the quantization errors of gradients. The quantization considered in [27] is based on a finite set of points on a unit sphere. Roughly speaking, the quantized vectors are such that the angle between any normalized gradient and some quantized vector is always acute. This indirectly imposes conditions on the inexactness or the underlying error. The use of normalized gradients, together with the preceding quantization form a sort of *zooming-in and quantized* policy that imposes errors to diminish as the iteration number increases, fostering the convergence of iterates to optimality. A related *zooming-in and quantized* policy are discussed also in [28]. Similar to [27], the modeling assumptions impose conditions on errors to diminish as the iteration number increases. We note that, unlike a quantization setting, in which one is allowed to control the error, in a general setting, it is not necessarily the case that controlling errors is at one's disposal.

Chapter 3

Materials and Methods

In this chapter, we introduce the main problem that we consider in this study and discuss the related distributed solution methods based on dual decomposition. Chapter 3.1 introduces the main problem. The dual decomposition approach is discussed in Chapter 3.2 and the imperfect coordination between subsystems is considered in Chapter 3.3. Finally, related distributed algorithms over non-ideal settings are presented in Chapter 3.4.

3.1 Problem Formulation

We consider a problem of minimizing a global convex objective function, which is a sum of local convex objective functions under general convex constraints, a formulation common to many types of large-scale signal processing and machine learning applications. In particular, a collection of m subsystems, where $m \in \mathbb{Z}_+$ is considered, who jointly solve the optimization problem

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^m f_i(\mathbf{z}) \\ & \text{subject to} && \mathbf{z} \in \mathcal{Y}, \end{aligned} \tag{3.1}$$

where the variable is $\mathbf{z} \in \mathbb{R}^n$ and $\mathcal{Y} \subseteq \mathbb{R}^n$ is considered as a common constraint set. Each $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is a strictly convex function associated with subsystem i , for all $i \in \{1, \dots, m\}$. Without loss of generality, we let $\text{dom } f_i = \mathbb{R}^n$ for all $i \in \{1, \dots, m\}$ ¹. Here \mathbf{z} is called the *public* variable. This problem is commonly known as the *global consensus problem*, a key formulation prevalent in statistical and machine learning application domains. Other real-world applications of (3.1) include networked vehicles, smart power grids, control of UAV/multiple robots, and TCP control systems, [82, 83]. We

¹Otherwise, we can encode respective domain information into each f_i by redefining it as $f_i := f_i + \delta_{\text{dom } f_i}$ (cf. Definition 16). Still, all the mathematical substantiations remain intact.

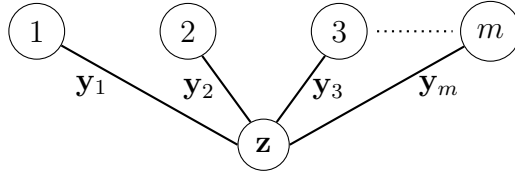


Figure 3.1: Decomposition Structure: There are m subsystems with the public variable \mathbf{z} . Functions associated with subsystems are $f_i(\mathbf{z})$, $i \in \{1, \dots, m\}$.

made the following assumption on the constraint set \mathcal{Y} and each local objective function f_i , $i = 1, \dots, m$.

Assumption 1 (Closedness). *The set \mathcal{Y} and the functions f_i s are closed.*

Here it is important to highlight that all our derivations and results can easily be extended to a more general formulation, where f_i s depend only on a part of the variable \mathbf{z} . The related generalized problem is called the general consensus problem. Related results are presented in Chapter 4.3.

Problem (3.1) can also be considered in centralized settings, where a certain central authority has the accessibility to all local objective functions. However, in practice, the unprecedented growth of the size of modern datasets, decentralized collection of datasets, and the underlying high-dimensional decision spaces, prevent the applicability of centralized methods such as interior-point algorithms [57]. In fact, they entail the development of scalable distributed solution methods for problems of the form (3.1), [56, 59]. A commonly used technique to yield distributed solution methods is based on the dual decomposition [61], where the decomposition structure of the underlying problem places a crucial role (see Figure 3.1).

3.2 Dual Decomposition Approach

First, associated with each subsystem i , a *private variable* \mathbf{y}_i is introduced instead of the global variable \mathbf{z} , together with necessary constraints to ensure their consistency $\mathbf{z} = \mathbf{y}_i$, for all $i = 1, \dots, m$. Thus, the problem (3.1) is equivalently reformulated as follows:

$$\begin{aligned}
& \text{minimize} && f(\mathbf{y}) = \sum_{i=1}^m f_i(\mathbf{y}_i) \\
& \text{subject to} && \mathbf{y}_i \in \mathcal{Y}, \quad i = 1, \dots, m \\
& && \mathbf{y}_i = \mathbf{y}_{i+1}, \quad i = 1, \dots, m-1,
\end{aligned} \tag{3.2}$$

where $\mathbf{y}_i \in \mathbb{R}^n$, $i = 1, \dots, m$, are newly introduced local versions of the public variable \mathbf{z} and $\mathbf{y} = [\mathbf{y}_1^T \dots \mathbf{y}_m^T]^T$. It can easily be observed that the objective is now separable. Let $\boldsymbol{\lambda}_i \in \mathbb{R}^n$ denote the Lagrange multiplier associated with the consistency constraint $\mathbf{y}_i = \mathbf{y}_{i+1}$, $i = 1, \dots, m-1$ and $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_1^T \dots \boldsymbol{\lambda}_{m-1}^T]^T$ for clarity. Then, the dual function $g : \mathbb{R}^{n(m-1)} \rightarrow \overline{\mathbb{R}}$ corresponding to (3.2) is given by

$$\begin{aligned}
g(\boldsymbol{\lambda}) &= \inf_{\mathbf{y}_i \in \mathcal{Y}, i=1, \dots, m} \left[\sum_{i=1}^m f_i(\mathbf{y}_i) + \sum_{i=1}^{m-1} \boldsymbol{\lambda}_i^T (\mathbf{y}_i - \mathbf{y}_{i+1}) \right] \\
&= \inf_{\mathbf{y}_1 \in \mathcal{Y}} [f_1(\mathbf{y}_1) + \boldsymbol{\lambda}_1^T \mathbf{y}_1] + \sum_{i=2}^{m-1} \inf_{\mathbf{y}_i \in \mathcal{Y}} [f_i(\mathbf{y}_i) + (\boldsymbol{\lambda}_i - \boldsymbol{\lambda}_{i-1})^T \mathbf{y}_i] \\
&\quad + \inf_{\mathbf{y}_m \in \mathcal{Y}} [f_m(\mathbf{y}_m) - \boldsymbol{\lambda}_{m-1}^T \mathbf{y}_m].
\end{aligned} \tag{3.3}$$

$$\tag{3.4}$$

Here, the last equality (3.4) follows because, for fixed $\boldsymbol{\lambda}$, the infimization can be performed in parallel by each subsystem. Thus, associated with each subsystem, there is a subproblem that can be handled locally. The related subproblems are given as follows:

$$\text{Subproblem 1 : } \inf_{\mathbf{y}_1 \in \mathcal{Y}} [f_1(\mathbf{y}_1) + \boldsymbol{\lambda}_1^T \mathbf{y}_1] \tag{3.5}$$

$$\text{Subproblem } i : \inf_{\mathbf{y}_i \in \mathcal{Y}} [f_i(\mathbf{y}_i) + (\boldsymbol{\lambda}_i - \boldsymbol{\lambda}_{i-1})^T \mathbf{y}_i], \quad i = 2, \dots, m-1 \tag{3.6}$$

$$\text{Subproblem } m : \inf_{\mathbf{y}_m \in \mathcal{Y}} [f_m(\mathbf{y}_m) - \boldsymbol{\lambda}_{m-1}^T \mathbf{y}_m] \tag{3.7}$$

Then, the dual problem is given by

$$\text{maximize}_{\boldsymbol{\lambda} \in \mathbb{R}^{n(m-1)}} g(\boldsymbol{\lambda}). \tag{3.8}$$

Next, the dual problem (3.8) can be solved using an iterative algorithm such as the

classical (or basic) subgradient method. The respective dual variable update ($\boldsymbol{\lambda}$ update) is given by

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \gamma_k \mathbf{d}^{(k)}, \quad (3.9)$$

where $\gamma_k > 0$ is the step size and $\mathbf{d}^{(k)}$ is a supergradient (*cf.* Remark 8) of g at $\boldsymbol{\lambda}^{(k)} \in \mathbb{R}^n$, and k indicates the iteration index. Equation (3.4) clearly shows that the subproblem coordination is solely required to jointly construct the supergradient $\mathbf{d}^{(k)}$ in (3.9) at iterate k . Related distributed algorithm (dual decomposition algorithm, *cf.* Algorithm 4) is presented below.

Algorithm 4 Dual Decomposition Algorithm

Require: $\boldsymbol{\lambda}^{(0)} \in \mathbb{R}^{n(m-1)}$.

- 1: $k = 0$.
 - 2: **repeat**
 - 3: Solve subproblems in parallel with $\boldsymbol{\lambda} = \boldsymbol{\lambda}^{(k)}$ to yield $\mathbf{y}^{(k)} \in \mathbb{R}^{nm}$.
▷ Local computations, *cf.* (3.5), (3.6), and (3.7)
 - 4: Compute $\mathbf{d}^{(k)} = [(\mathbf{y}_1^{(k)} - \mathbf{y}_2^{(k)})^\top \dots (\mathbf{y}_{m-1}^{(k)} - \mathbf{y}_m^{(k)})^\top]^\top$. ▷ Subproblem coordination
 - 5: $\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \gamma_k \mathbf{d}^{(k)}$. ▷ Dual variable update
 - 6: $k := k + 1$.
 - 7: **until** a stopping criterion true
-

Under mild technical conditions such as the gradient Lipschitz continuity or strong convexity of the global objective function f , the convergence of $\boldsymbol{\lambda}^{(k)}$ to the optimal solution $\boldsymbol{\lambda}^*$ of (3.8) can be ensured, i.e., $\boldsymbol{\lambda}^{(k)} \rightarrow \boldsymbol{\lambda}^*$ [11, Section 1.4.2]. Thus, together with additional assumptions such as the strong duality between (3.2) and (3.8), at the termination of the algorithm, a reasonable guess for the solution \mathbf{y}^* of (3.2) is obtained by averaging the solutions of the subproblems, i.e., $(1/m) \sum_{i=1}^m \mathbf{y}_i^{(k)}$ [84, Section 5.5.5].

3.3 Imperfect Coordination Between Subsystems

It is worth emphasizing that, in practice, the subproblem coordination cannot be performed in pure form due to system-specific restrictions imposed on real world systems. Thus, we consider the case where the subproblem coordination in each iteration k is not perfect (cf. line 4 of Algorithm 4). In particular, instead of the exact $\mathbf{y}_i^{(k)}$, a distorted vector $\hat{\mathbf{y}}_i^{(k)}$ is used when computing $\mathbf{d}^{(k)}$ in (3.9). As a result, instead of the exact $\mathbf{d}^{(k)}$, a distorted vector $\hat{\mathbf{d}}^{(k)}$ given by

$$\hat{\mathbf{d}}^{(k)} = [(\hat{\mathbf{y}}_1^{(k)} - \hat{\mathbf{y}}_2^{(k)})^\top \dots (\hat{\mathbf{y}}_{m-1}^{(k)} - \hat{\mathbf{y}}_m^{(k)})^\top]^\top \quad (3.10)$$

is used in the dual variable update of Algorithm 4 (cf. line 5).

Next, we denote by $\mathbf{r}_i^{(k)} \in \mathbb{R}^n$, the distortion associated with $\mathbf{y}_i^{(k)}$, for all $i = 1, \dots, m$ and $k \in \mathbb{Z}_+^0$. Then, the distorted vector $\hat{\mathbf{y}}_i^{(k)}$ is simply given by

$$\hat{\mathbf{y}}_i^{(k)} = \mathbf{y}_i^{(k)} + \mathbf{r}_i^{(k)}. \quad (3.11)$$

However, if such a distortion is associated with an underlying system, then the analysis of how the related distributed algorithms might develop to model those imperfections is of utmost importance.

3.4 Distributed Algorithms over Non-ideal Settings

In this section, we propose two distributed algorithms to deploy over various non-ideal settings, which impose restrictions on the exactness of the underlying algorithms. It is worth noting that the distortion $\mathbf{r}_i^{(k)}$ can model numerous inexact settings as remarked below.

Remark 11. *The additive distortion $\mathbf{r}_i^{(k)}$ can model errors in many large-scale optimization problems, including quantization errors [25, 27, 28], approximation errors [20, 21],*

errors due to subproblem solver accuracy [24, 26], noise induced in wireless settings [2], and measurement errors, among others.

Despite the generality of $\mathbf{r}_i^{(k)}$, we refer to it as a distortion due to imperfect coordination, unless otherwise specified. When modeling the distortion $\mathbf{r}_i^{(k)}$, we assume nothing except the norm boundedness of the distortion. More specifically, we have the following assumption about the distortion $\mathbf{r}_i^{(k)}$.

Assumption 2 (Absolute Deterministic Distortion). *The distortion $\mathbf{r}_i^{(k)}$ associated with $\mathbf{y}_i^{(k)}$, $i = 1, \dots, m$, $k \in \mathbb{Z}_+^0$ is bounded by $\varepsilon_i \in \mathbb{R}$, i.e.,*

$$\|\mathbf{r}_i^{(k)}\| \leq \varepsilon_i, \quad i = 1, \dots, m, \quad k \in \mathbb{Z}_+^0. \quad (3.12)$$

That is the coordination of $\mathbf{y}_i^{(k)}$, $i = 1, \dots, m$, $k \in \mathbb{Z}_+^0$, always undergoes a given absolute error ε_i . Note that the distortion $\mathbf{r}_i^{(k)}$ need not be random. If it is random, it need not be stationary, uncorrelated, or even zero mean.

Our exposition of imperfect coordination is centered on two variants of Algorithm 4. The first algorithm is a partially distributed algorithm (cf. Algorithm 5), where a central node solely performs the subproblem coordination. The second one is fully distributed (cf. Algorithm 6), in the sense that there is no central authority. Any subsystem communicates at most with two other subsystems, during the subproblem coordination step. In the sequel, details of the two algorithms are outlined.

3.4.1 Partially Distributed Algorithm

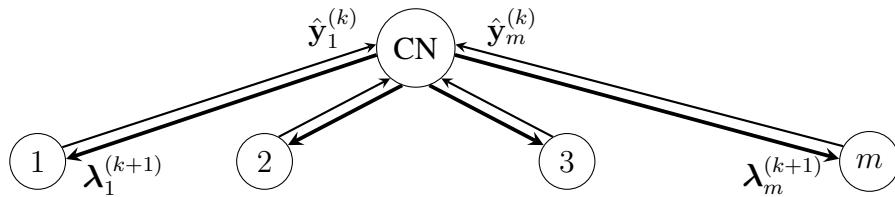


Figure 3.2: Graph of the Communication Structure: Partially Distributed Algorithm

In the proposed partially distributed algorithm, a central node (CN), who is involved during the subproblem coordination ² is available, in addition to the subsystems. The subproblem coordination is enabled with the presence of the following resources.

1. An error-free broadcast channel between CN and subsystems,
2. A communication channel between each subsystem and the CN, which conforms to the absolute error conditions specified in Assumption 2.

See Figure 3.2 for an illustration. The proposed algorithm is summarized below.

Algorithm 5 Partially Distributed Algorithm

Require: $\boldsymbol{\lambda}^{(0)} \in \mathbb{R}^{n(m-1)}$; $\boldsymbol{\lambda}_0^{(j)} = \boldsymbol{\lambda}_m^{(j)} = \mathbf{0} \in \mathbb{R}^n$, $j \in \mathbb{Z}_+^0$.

- 1: $k = 0$.
 - 2: CN broadcasts $\boldsymbol{\lambda}^{(0)}$ to subsystems. ▷ Initial dual variables
 - 3: **repeat**
 - 4: $\forall i$, subsystem i computes $\mathbf{y}_i^{(k)}$ by solving ▷ Local computations

$$\underset{\mathbf{y}_i \in \mathcal{Y}}{\text{minimize}} \quad f_i(\mathbf{y}_i) + (\boldsymbol{\lambda}_i^{(k)} - \boldsymbol{\lambda}_{i-1}^{(k)})^\top \mathbf{y}_i.$$
 - 5: $\forall i$, subsystem i transmits $\mathbf{y}_i^{(k)}$ to CN. ▷ Subproblem coordination: stage 1
 - 6: $\forall i$, CN receives $\hat{\mathbf{y}}_i^{(k)}$, cf. (3.11). ▷ Subproblem coordination: stage 2
 - 7: CN computes $\hat{\mathbf{d}}^{(k)} = [(\hat{\mathbf{y}}_1^{(k)} - \hat{\mathbf{y}}_2^{(k)})^\top \dots (\hat{\mathbf{y}}_{m-1}^{(k)} - \hat{\mathbf{y}}_m^{(k)})^\top]^\top$ ▷ Subproblem coordination: stage 3
 - 8: CN computes $\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \gamma_k \hat{\mathbf{d}}^{(k)}$ ▷ Dual variable update
 - 9: CN broadcasts $\boldsymbol{\lambda}^{(k+1)}$ to subsystems. ▷ Current dual variables
 - 10: $k := k + 1$
 - 11: **until** a stopping criterion true
-

It is worth noting that, the distortion $\mathbf{r}_i^{(k)}$ can model persistent measurement errors or inevitable approximation errors that occur as a result of attempts to reduce the computational complexity in large-scale machine learning algorithms [85, 86].

²The role of CN is virtual in the sense that, an arbitrarily chosen subsystem itself can act as the CN.

Each operation listed from lines 4 – 6 is conducted in parallel among $i \in \{1, \dots, m\}$. Note that the subproblem coordination is conducted via the intervention of CN (*cf.* steps 5-7 of Algorithm 5). Subproblem coordination stage 2 (see line 6) is the source of the imperfect coordination, where an absolute deterministic distortion is introduced [*cf.* Assumption 2].

3.4.2 Fully Distributed Algorithm

A fully distributed algorithm is presented here in which a central node is not available. In particular, it turns out that the decomposition structure (*cf.* Figure 3.1) considered when reformulating problem (3.2) suggests a subproblem coordination mechanism where only the communication between neighboring subsystems is necessary. The communication structure is depicted in Figure 3.3. Associated with each $i, i = 1, \dots, m$, the following resources are there to enable the subproblem coordination:

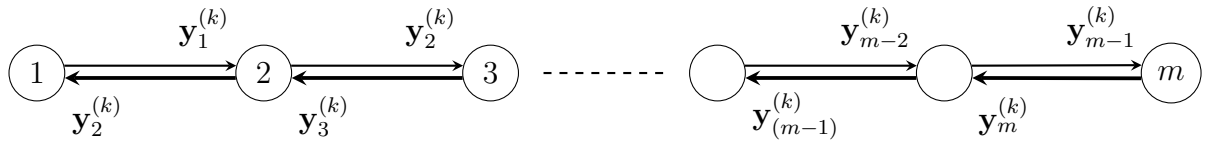


Figure 3.3: Graph of the Communication Structure: Fully Distributed Algorithm

1. An error-free communication channel from subsystem $i - 1$ to i ,
2. An error-free communication channel from subsystem $i + 1$ to i .

In this context, the distortion $\mathbf{r}_i^{(k)}$ is due to inevitable approximation errors that come about as a result of attempts to reduce communication overhead, see [25, 28].

The resulting algorithm is summarized below.

Algorithm 6 Fully Distributed Algorithm

Require: $\boldsymbol{\lambda}^{(0)} \in \mathbb{R}^{n(m-1)}$; $\boldsymbol{\lambda}_0^{(j)} = \boldsymbol{\lambda}_m^{(j)} = \mathbf{0} \in \mathbb{R}^n, j \in \mathbb{Z}_+$; $\mathbf{y}_0^{(j)} = \mathbf{y}_{m+1}^{(j)} = \mathbf{0} \in \mathbb{R}^n, j \in \mathbb{Z}_+$.

- 1: $k = 0$.

2: **repeat**

3: $\forall i$, subsystem (SS) i computes $\mathbf{y}_i^{(k)}$ by solving ▷ Local computations

$$\underset{\mathbf{y}_i \in \mathcal{Y}}{\text{minimize}} \quad f_i(\mathbf{y}_i) + (\boldsymbol{\lambda}_i^{(k)} - \boldsymbol{\lambda}_{i-1}^{(k)})^\top \mathbf{y}_i.$$

4: $\forall i$, SS i transmits $\hat{\mathbf{y}}_i^{(k)}$ to $i - 1$ and $i + 1$, *cf.* (3.11). ▷ Subproblem coordination:
stage 1

5: $\forall i$, SS i receives $\hat{\mathbf{y}}_{i-1}^{(k)}$ and $\hat{\mathbf{y}}_{i+1}^{(k)}$ from $i - 1$ and $i + 1$. ▷ Subproblem coordination:
stage 2

6: $\forall i$, SS i computes $\hat{\mathbf{d}}_i^{(k)} = [(\hat{\mathbf{y}}_{i-1}^{(k)} - \hat{\mathbf{y}}_i^{(k)})^\top (\hat{\mathbf{y}}_i^{(k)} - \hat{\mathbf{y}}_{i+1}^{(k)})^\top]^\top$. ▷ Subproblem
coordination: stage 3

7: $\forall i$, SS i computes $[(\boldsymbol{\lambda}_{i-1}^{(k+1)})^\top (\boldsymbol{\lambda}_i^{(k+1)})^\top]^\top = [(\boldsymbol{\lambda}_{i-1}^{(k)})^\top (\boldsymbol{\lambda}_i^{(k)})^\top]^\top + \gamma_k \hat{\mathbf{d}}_i^{(k)}$. ▷ Local
dual variable update

8: $k := k + 1$

9: **until** a stopping criterion true

Note that each operation listed from lines 3 – 7 is conducted in parallel among $i \in \{1, \dots, m\}$. Unlike the Algorithm 5, here the subproblem coordination is solely achieved by the subsystem's communication with its neighbors and local computations (*cf.* lines 4-6 of Algorithm 6). Subproblem coordination stage 1 (see line 4) is the source of the imperfect coordination, where an absolute deterministic distortion is introduced (*cf.* Assumption 2).

Chapter 4

Results and Discussion

4.1 Analysis of the properties of the Dual Function

The dual function g [cf. Equation (3.3)] associated with the primal problem (3.2) plays a major role when asserting convergence properties of Algorithm 5 and Algorithm 6. Thus, the focus of this chapter is to provide extensive analysis on the properties of the dual function g . In particular, we proceed toward hypothesizing some important characteristics of underlying primal functions and derive useful results, which in turn are used in asserting convergence properties of underlying algorithms.

4.1.1 Dual Function as a Restriction of f^*

First, we highlight an important relationship between g , and the conjugate function f^* (cf. Definition 15) of $f + \delta_{\bar{\mathcal{Y}}}$ ¹, where $\bar{\mathcal{Y}}$ denotes the m -fold Cartesian product of \mathcal{Y} , i.e.,

$$\bar{\mathcal{Y}} = \mathcal{Y}^m = \underbrace{\mathcal{Y} \times \mathcal{Y} \dots \times \mathcal{Y}}_{m \text{ times}}. \quad (4.1)$$

The result is outlined as follows.

Lemma 3. *Let $f^* : \mathbb{R}^{nm} \rightarrow \overline{\mathbb{R}}$ denote the conjugate function of $f + \delta_{\bar{\mathcal{Y}}}$. Then*

$$g(\boldsymbol{\lambda}) = -f^*(\mathbf{A}^T \boldsymbol{\lambda}), \quad (4.2)$$

¹Recall that f is the objective function of problem (3.2) and $\delta_{\bar{\mathcal{Y}}}$ is the indicator function of the set $\bar{\mathcal{Y}}$ (cf. Definition 16).

where \mathbf{A} is an $n(m-1) \times nm$ matrix with the special block structure given by:

$$\mathbf{A} = \begin{bmatrix} \mathbf{I}_n & -\mathbf{I}_n & \mathbf{0} & \cdots & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n & -\mathbf{I}_n & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_n & -\mathbf{I}_n & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \vdots & \ddots & \mathbf{0} & \mathbf{I}_n & -\mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{I}_n & -\mathbf{I}_n \end{bmatrix}. \quad (4.3)$$

Proof. It is straightforward to see that the equality constraint $\mathbf{y}_i = \mathbf{y}_{i+1}$, $i = 1, \dots, m-1$ of problem 3.2 is equivalent to $\mathbf{A}\mathbf{y} = \mathbf{0}$ [cf. the structure of matrix \mathbf{A} given in (4.3)]. Then,

$$g(\boldsymbol{\lambda}) = \inf_{\mathbf{y} \in \mathcal{Y}} (f(\mathbf{y}) - \boldsymbol{\lambda}^T \mathbf{A}\mathbf{y}) \quad (4.4)$$

$$= \inf_{\mathbf{y}} (f(\mathbf{y}) + \delta_{\mathcal{Y}}(\mathbf{y}) - \boldsymbol{\lambda}^T \mathbf{A}\mathbf{y}) \quad (4.5)$$

$$= -\sup_{\mathbf{y}} ((\mathbf{A}^T \boldsymbol{\lambda})^T \mathbf{y} - f(\mathbf{y}) - \delta_{\mathcal{Y}}(\mathbf{y})) \quad (4.6)$$

$$= -f^*(\mathbf{A}^T \boldsymbol{\lambda}), \quad (4.7)$$

where (4.4) directly follows from (3.3), (4.5) follows from the definition of $\delta_{\mathcal{Y}}$, (4.6) follows simply by replacing inf by sup, and finally, (4.7) follows from the definition of the conjugate function of $f + \delta_{\mathcal{Y}}$. \square

Lemma 3 indicates that the dual function g is a restriction of f^* to a linear space.

4.1.2 Lipschitzian Properties

In general, The Lipschitzian property of the gradient (gradient Lipschitz continuity) is a common characteristic that is satisfied by most underlying objective functions, when asserting convergence results. Thus, we now furnish a simple, but important result that verifies the Lipschitzian properties of the dual function g of the problem (3.2) (cf. Definition 14).

First, we made the following hypothesis that is satisfied by most standard utility functions considered in the literature.

Assumption 4.1.1 (Strongly convex local objectives at subsystems). *The local objective functions f_i s in problem (3.2) are strongly convex with constant $\mu_i > 0$, $i = 1, \dots, m$.*

Next, the strong convexity of the global objective function f of the problem (3.2) is verified by the following Lemma.

Lemma 4. *Suppose Assumption 4.1.1 holds. Then, the objective function f of problem (3.2) given by $f(\mathbf{y}) = \sum_{i=1}^m f_i(\mathbf{y}_i)$ is strongly convex with constant $\mu = \min_i \mu_i$.*

Proof. The condition that f_i is strongly convex with constant $\mu_i > 0$ is equivalent to the strong monotonicity condition of ∂f_i (cf. Definition 13 and Theorem 2), i.e.,

$$(\mathbf{v}_i - \bar{\mathbf{v}}_i)^\top (\mathbf{x}_i - \bar{\mathbf{x}}_i) \geq \mu_i \|\mathbf{x}_i - \bar{\mathbf{x}}_i\|^2, \quad \text{where } \mathbf{v}_i \in \partial f_i(\mathbf{x}_i), \bar{\mathbf{v}}_i \in \partial f_i(\bar{\mathbf{x}}_i). \quad (4.8)$$

We now let $\mathbf{v} = [\mathbf{v}_1^\top \dots \mathbf{v}_m^\top]^\top$, $\bar{\mathbf{v}} = [\bar{\mathbf{v}}_1^\top \dots \bar{\mathbf{v}}_m^\top]^\top$, $\mathbf{x} = [\mathbf{x}_1^\top \dots \mathbf{x}_m^\top]^\top$, $\bar{\mathbf{x}} = [\bar{\mathbf{x}}_1^\top \dots \bar{\mathbf{x}}_m^\top]^\top$, and put together the strong monotonicity property (4.8) for all $i \in \{1, \dots, m\}$. Then

$$(\mathbf{v} - \bar{\mathbf{v}})^\top (\mathbf{x} - \bar{\mathbf{x}}) = \sum_{i=1}^m (\mathbf{v}_i - \bar{\mathbf{v}}_i)^\top (\mathbf{x}_i - \bar{\mathbf{x}}_i) \quad (4.9)$$

$$\geq \sum_{i=1}^m \mu_i \|\mathbf{x}_i - \bar{\mathbf{x}}_i\|^2 \quad (4.10)$$

$$\geq \min_{j \in \{1, \dots, m\}} \mu_j \sum_{i=1}^m \|\mathbf{x}_i - \bar{\mathbf{x}}_i\|^2 \quad (4.11)$$

$$= \min_{j \in \{1, \dots, m\}} \mu_j \|\mathbf{x} - \bar{\mathbf{x}}\|^2, \quad (4.12)$$

where (4.9) follows from that $\mathbf{v} \in \partial f(\mathbf{x}), \bar{\mathbf{v}} \in \partial f(\bar{\mathbf{x}}) \iff \mathbf{v}_i \in \partial f_i(\mathbf{x}_i), \bar{\mathbf{v}}_i \in \partial f_i(\bar{\mathbf{x}}_i)$ for all $i \in \{1, \dots, m\}$ ², (4.10) is immediate from (4.8), (4.11) trivially follows since $\min_j \mu_j \leq \mu_i, \forall i$, and (4.12) follows from the definition of ℓ_2 -norm. Then, the strong convexity of f with constant $\mu = \min_i \mu_i$ is immediate from the equivalence between strong monotonicity of ∂f and strong convexity of f (cf. Theorem 2). \square

²In Particular, note that $\partial f(\mathbf{x}) = \partial f_1(\bar{\mathbf{x}}_1) \times \partial f_2(\bar{\mathbf{x}}_2) \times \dots \times \partial f_m(\bar{\mathbf{x}}_m)$.

Then, the strong convexity of $f + \delta_{\bar{\mathcal{Y}}}$ is an immediate consequence of Lemma 4. The result is outlined in the following remark.

Remark 12. *Suppose Assumption 4.1.1 holds. Then, the function $f + \delta_{\bar{\mathcal{Y}}}$ is strongly convex with constant $\mu = \min_i \mu_i$.*

Proof. This is immediate from Lemma 4 and the convexity of $\bar{\mathcal{Y}}$ [cf. equation (4.1)]. \square

Finally, the following result claims the Lipschitzian property of the gradient of the dual function g .

Proposition 1. *Suppose Assumption 1 and Assumption 4.1.1 hold. Then the dual function g is differentiable. Moreover, the gradient ∇g of g is Lipschitz continuous with constant $(1/\mu)(2 + 2 \cos(\pi/m))$, where $\mu = \min_i \mu_i$.*

Proof. Under Assumption 1, it is immediate that function $f + \delta_{\bar{\mathcal{Y}}}$ is closed (cf. Definition 7). This together with [45, Theorem 11.13], ensures the differentiability of f^* . Thus, the differentiability of g follows directly from Lemma 3.

To show the Lipschitz continuity of ∇g , let us first assert the property for ∇f^* . Recall that the function f^* is the conjugate function of $f + \delta_{\bar{\mathcal{Y}}}$. Assumption 1 ensures that the function $f + \delta_{\bar{\mathcal{Y}}}$ is closed and convex, and so is f^* by [45, Theorem 11.1]. Moreover, From Remark 12, we have that $f + \delta_{\bar{\mathcal{Y}}}$ is strongly convex with constant $\mu = \min_i \mu_i$. Thus, by invoking the equivalence of [45, Proposition 12.60: (a) and (b)], together with the biconjugate property $(f^*)^* = f$ the Lipschitz continuity of the gradients ∇f^* with constant $1/\mu$ is immediate. Then for any $\gamma, \delta \in \mathbb{R}^{n(m-1)}$,

$$\|\nabla g(\gamma) - \nabla g(\delta)\| = \|\nabla f^*(\mathbf{A}^T \gamma) - \nabla f^*(\mathbf{A}^T \delta)\| \quad (4.13)$$

$$= \|\mathbf{A} [\nabla f^*(\mathbf{A}^T \gamma) - \nabla f^*(\mathbf{A}^T \delta)]\| \quad (4.14)$$

$$\leq \|\mathbf{A}\|_2 \|\nabla f^*(\mathbf{A}^T \gamma) - \nabla f^*(\mathbf{A}^T \delta)\| \quad (4.15)$$

$$\leq \frac{\|\mathbf{A}\|}{\mu} \|\mathbf{A}^T \gamma - \mathbf{A}^T \delta\| \quad (4.16)$$

$$\leq \frac{\|\mathbf{A}\|^2}{\mu} \|\gamma - \delta\| \quad (4.17)$$

$$= \frac{\lambda_{\max}(\mathbf{A}\mathbf{A}^T)}{\mu} \|\boldsymbol{\gamma} - \boldsymbol{\delta}\| \quad (4.18)$$

$$= \frac{2 + 2 \cos(\pi/m)}{\mu} \|\boldsymbol{\gamma} - \boldsymbol{\delta}\|, \quad (4.19)$$

where (4.13) follows from Lemma 3, (4.14) is immediate from chain rule properties, (4.15) is immediate from spectral norm properties, (4.16) follows from the Lipschitzian properties of ∇f^* , (4.17) is again from spectral norm properties, (4.18) is from that $\|\mathbf{A}\|_2^2 = \lambda_{\max}(\mathbf{A}\mathbf{A}^T)$, and finally (4.19) is immediate from the block-tridiagonal structure of $\mathbf{A}\mathbf{A}^T$ [cf. (4.3) and [87, p. 565]]. \square

4.1.3 Strong Convexity Properties

The closedness of f (cf. Assumption 1), together with the Legendre-Fenchel transform [45, Theorem 11.1], allows a dual result of Proposition 1 to be worked out, again by using [45, Proposition 12.60: (a),(b)]. Let us first outline a natural assumption that emerges in this regard.

Assumption 4.1.2 (Gradient Lipschitz continuous local objectives at subsystems). *The set \mathcal{Y} in problem (3.2) equals \mathbb{R}^n . Moreover, f_i s are differentiable and the gradients ∇f_i s are Lipschitz continuous on \mathbb{R}^n with constant $L_i > 0$, $i = 1, \dots, m$.*

Lemma 5. *Suppose Assumption 4.1.2 holds. Then, the gradient ∇f of f is Lipschitz continuous on \mathbb{R}^n with constant $L = \max_i L_i$.*

Proof. Let $\mathbf{x} = [\mathbf{x}_1^T \dots \mathbf{x}_m^T]^T$, $\bar{\mathbf{x}} = [\bar{\mathbf{x}}_1^T \dots \bar{\mathbf{x}}_m^T]^T$. Then

$$\|\nabla f(\mathbf{x}) - \nabla f(\bar{\mathbf{x}})\| = \left\| \left[[\nabla f_1(\mathbf{x}_1) - \nabla f_1(\bar{\mathbf{x}}_1)]^T \dots [\nabla f_m(\mathbf{x}_m) - \nabla f_m(\bar{\mathbf{x}}_m)]^T \right]^T \right\| \quad (4.20)$$

$$= \sqrt{\sum_{i=1}^m \|\nabla f(\mathbf{x}_i) - \nabla f(\bar{\mathbf{x}}_i)\|^2} \quad (4.21)$$

$$\leq \sqrt{\sum_{i=1}^m L_i^2 \|\mathbf{x}_i - \bar{\mathbf{x}}_i\|^2} \quad (4.22)$$

$$\leq \max_{j \in \{1, \dots, m\}} L_j \sqrt{\sum_{i=1}^m \|\mathbf{x}_i - \bar{\mathbf{x}}_i\|^2} \quad (4.23)$$

$$= \max_{j \in \{1, \dots, m\}} L_j \|\mathbf{x} - \bar{\mathbf{x}}\|, \quad (4.24)$$

where (4.20) follows directly from the definition of the gradient, (4.21) follows simply from the definition of ℓ_2 -norm (cf. Definition 8), (4.22) is immediate from the Lipschitz continuity of ∇f_i s, and (4.23) and (4.24) follow trivially from simple manipulations. \square

Finally, the following lemma claims the strong convexity property of the negative dual function $-g$ under mild conditions.

Proposition 2. *Suppose Assumption 4.1.2 holds. Then the function $-g$ is strongly convex with constant $(1/L)(2 - 2\cos(\pi/m))$, where $L = \min_i L_i$.*

Proof. To show the result, we use the equivalence between the strong convexity and the strong monotonicity condition (cf. Definition 13 and Theorem 2).

Strong convexity of f^* is immediate by invoking Lemma 5 together with the equivalence of [45, Proposition 12.60: (a)-(b)], i.e.,

$$(\mathbf{y} - \bar{\mathbf{y}})^T(\boldsymbol{\nu} - \bar{\boldsymbol{\nu}}) \geq (1/L)\|\boldsymbol{\nu} - \bar{\boldsymbol{\nu}}\|_2^2, \quad \mathbf{y} \in \partial f^*(\boldsymbol{\nu}), \bar{\mathbf{y}} \in \partial f^*(\bar{\boldsymbol{\nu}}). \quad (4.25)$$

It remains to be shown the strong monotonicity of $\partial(-g)$. Let $h = -g$ for clarity. Moreover, for any $\boldsymbol{\gamma}, \boldsymbol{\delta} \in \mathbb{R}^{n(m-1)}$, let $\mathbf{x} \in \partial h(\boldsymbol{\gamma})$ and $\bar{\mathbf{x}} \in \partial h(\boldsymbol{\delta})$ which entails

$$\mathbf{x} \in \mathbf{A}\partial f^*(\mathbf{A}^T\boldsymbol{\gamma}) \text{ and } \bar{\mathbf{x}} \in \mathbf{A}\partial f^*(\mathbf{A}^T\boldsymbol{\delta}). \quad (4.26)$$

Then,

$$(\mathbf{x} - \bar{\mathbf{x}})^T(\boldsymbol{\gamma} - \boldsymbol{\delta}) = (\mathbf{A}\mathbf{y} - \mathbf{A}\bar{\mathbf{y}})^T(\boldsymbol{\gamma} - \boldsymbol{\delta}), \quad \mathbf{y} \in \partial f^*(\mathbf{A}^T\boldsymbol{\gamma}), \bar{\mathbf{y}} \in \partial f^*(\mathbf{A}^T\boldsymbol{\delta}) \quad (4.27)$$

$$= (\mathbf{y} - \bar{\mathbf{y}})^T(\mathbf{A}^T\boldsymbol{\gamma} - \mathbf{A}^T\boldsymbol{\delta}), \quad \mathbf{y} \in \partial f^*(\mathbf{A}^T\boldsymbol{\gamma}), \bar{\mathbf{y}} \in \partial f^*(\mathbf{A}^T\boldsymbol{\delta}) \quad (4.28)$$

$$\geq \frac{1}{L} \|\mathbf{A}^T\boldsymbol{\gamma} - \mathbf{A}^T\boldsymbol{\delta}\|_2^2, \quad \mathbf{x} \in \partial h(\boldsymbol{\gamma}), \bar{\mathbf{x}} \in \partial h(\boldsymbol{\delta}) \quad (4.29)$$

$$= \frac{1}{L} (\boldsymbol{\gamma} - \boldsymbol{\delta})^T \mathbf{A} \mathbf{A}^T (\boldsymbol{\gamma} - \boldsymbol{\delta}), \quad \mathbf{x} \in \partial h(\boldsymbol{\gamma}), \bar{\mathbf{x}} \in \partial h(\boldsymbol{\delta}) \quad (4.30)$$

$$\geq \frac{\lambda_{\min}(\mathbf{A} \mathbf{A}^T)}{L} \|\boldsymbol{\gamma} - \boldsymbol{\delta}\|_2^2, \quad \mathbf{x} \in \partial h(\boldsymbol{\gamma}), \bar{\mathbf{x}} \in \partial h(\boldsymbol{\delta}) \quad (4.31)$$

$$= \frac{2 - 2 \cos(\pi/m)}{L} \|\boldsymbol{\gamma} - \boldsymbol{\delta}\|_2^2, \quad \mathbf{x} \in \partial h(\boldsymbol{\gamma}), \bar{\mathbf{x}} \in \partial h(\boldsymbol{\delta}). \quad (4.32)$$

The first equality (4.27) follows from (4.26) since $\mathbf{x} \in \partial h(\boldsymbol{\gamma})$ and $\bar{\mathbf{x}} \in \partial h(\boldsymbol{\delta}) \implies \exists \mathbf{y} \in \partial f^*(\mathbf{A}^T \boldsymbol{\gamma})$ such that $\mathbf{x} = \mathbf{A} \mathbf{y}$ and $\exists \bar{\mathbf{y}} \in \partial f^*(\mathbf{A}^T \boldsymbol{\delta})$ such that $\bar{\mathbf{x}} = \mathbf{A} \bar{\mathbf{y}}$. The equality (4.28) is immediate from (4.27). The inequality (4.29) is obtained from (4.25) and (4.30) follows trivially from the expansion of $\|\cdot\|_2^2$. Finally, (4.31) and (4.32) follow from properties of eigenvalues [cf. equation (4.3) and [87, p. 565]]. \square

4.1.4 Bounding Properties for the Primal Error

In this study, the convergence properties of our proposed algorithms are analyzed using dual decomposition. However, if the algorithms are modeled as they originated from the dual-domain, then the analysis of how they might evolve into the primal-domain is of utmost importance. In this respect, the focus of this section is to build useful relations among dual and primal variables, which in turn are used in subsequent sections to analyze the convergence properties of algorithms in the primal domain. In particular, bounding relations in terms of distance to the dual optimal value $g(\boldsymbol{\lambda}^*) - g(\boldsymbol{\lambda})$, distance to the primal optimal value $\|f(\mathbf{y}(\boldsymbol{\lambda})) - f(\mathbf{y}^*)\|$, and distance to the primal optimal solution $\|\mathbf{y}(\boldsymbol{\lambda}) - \mathbf{y}^*\|$ are explicitly derived (cf. Lemma 6).

First, let us invoke the strong duality assumption (cf. section 1.5.3.3), one of the standard assumptions which hold in many practically relevant convex optimization problems.

Assumption 3 (Strong Duality). *The optimal values p^* and d^* of the problems (3.2) and (3.8), respectively, are attained. Moreover, strong duality between (3.2) and (3.8) holds, i.e.,*

$$p^* = f(\mathbf{y}^*) = g(\boldsymbol{\lambda}^*) = d^*, \quad (4.33)$$

for some $\mathbf{y}^* \in \{\mathbf{y} \in \mathbb{R}^{nm} \mid \forall i \mathbf{y}_i \in \mathcal{Y}, \mathbf{A} \mathbf{y} = \mathbf{0}\}$ and for some $\boldsymbol{\lambda}^* \in \mathbb{R}^{n(m-1)}$, where \mathbf{A}

is defined in (4.3).

Lemma 6. *Suppose Assumption 1, Assumption 4.1.1, and Assumption 3 hold. If the functions f_i , $i = 1, \dots, m$ are differentiable, then*

1. $g(\boldsymbol{\lambda}^*) - g(\boldsymbol{\lambda}) \geq \mu/2 \|y(\boldsymbol{\lambda}) - \mathbf{y}^*\|^2$ for all $\boldsymbol{\lambda} \in \mathbb{R}^{n(m-1)}$,
2. $g(\boldsymbol{\lambda}^*) - g(\boldsymbol{\lambda}) + S\|\boldsymbol{\lambda}\|\sqrt{g(\boldsymbol{\lambda}^*) - g(\boldsymbol{\lambda})} \geq \|f(\mathbf{y}(\boldsymbol{\lambda})) - f(\mathbf{y}^*)\|$,

where

$$y(\boldsymbol{\lambda}) = \operatorname{argmin}_{\mathbf{y} \in \bar{\mathcal{Y}}} (f(\mathbf{y}) + \boldsymbol{\lambda}^T \mathbf{A}\mathbf{y}), \quad (4.34)$$

the set $\bar{\mathcal{Y}}$ is given in (4.1), the matrix \mathbf{A} is given in (4.3), $\mu = \min_i \mu_i$, and $S = \sqrt{(4 + 4 \cos(\pi/m))/\mu}$.

Proof. Let us first define compactly the partial Lagrangian $L : \mathbb{R}^{m \times n(m-1)} \rightarrow \mathbb{R}$ associated with Problem (3.2) [cf. equations (3.3) and (4.3)], i.e.,

$$L(\mathbf{y}, \boldsymbol{\lambda}) = f(\mathbf{y}) + \boldsymbol{\lambda}^T \mathbf{A}\mathbf{y}. \quad (4.35)$$

Then,

$$g(\boldsymbol{\lambda}^*) - g(\boldsymbol{\lambda}) = \inf_{\mathbf{y} \in \bar{\mathcal{Y}}} L(\mathbf{y}, \boldsymbol{\lambda}^*) - \inf_{\mathbf{y} \in \bar{\mathcal{Y}}} L(\mathbf{y}, \boldsymbol{\lambda}) \quad (4.36)$$

$$= L(\mathbf{y}^*, \boldsymbol{\lambda}^*) - L(\mathbf{y}(\boldsymbol{\lambda}), \boldsymbol{\lambda}) \quad (4.37)$$

$$= L(\mathbf{y}^*, \boldsymbol{\lambda}) - L(\mathbf{y}(\boldsymbol{\lambda}), \boldsymbol{\lambda}) \quad (4.38)$$

$$\geq \frac{\mu}{2} \|\mathbf{y}(\boldsymbol{\lambda}) - \mathbf{y}^*\|^2, \quad (4.39)$$

where (4.36) follows from the definition of the dual function, (4.37) follows from Assumption 3, and (4.38) is immediate from that $\mathbf{A}\mathbf{y}^* = \mathbf{0}$. Finally, the inequality (4.39) follows from [11, page 11, equation (35)] since L is a strongly convex function of \mathbf{y} with constant μ for fixed $\boldsymbol{\lambda}$, and the supposition that f is differentiable. This concludes the proof of the first part.

For the second claim, we start with equation (4.37). Then we have

$$\|g(\boldsymbol{\lambda}^*) - g(\boldsymbol{\lambda})\| = f(\mathbf{y}^*) - f(\mathbf{y}(\boldsymbol{\lambda})) - \boldsymbol{\lambda}^\top \mathbf{A} \mathbf{y}(\boldsymbol{\lambda}) \quad (4.40)$$

$$\geq \|f(\mathbf{y}(\boldsymbol{\lambda})) - f(\mathbf{y}^*)\| - \|\boldsymbol{\lambda}\| \|\mathbf{A}\| \|\mathbf{y}(\boldsymbol{\lambda}) - \mathbf{y}^*\| \quad (4.41)$$

$$\begin{aligned} &= \|f(\mathbf{y}(\boldsymbol{\lambda})) - f(\mathbf{y}^*)\| \\ &\quad - \sqrt{2 + 2 \cos(\pi/m)} \|\boldsymbol{\lambda}\| \|\mathbf{y}(\boldsymbol{\lambda}) - \mathbf{y}^*\|, \end{aligned} \quad (4.42)$$

where (4.40) follows simply from Assumption 3 and the definition of the Lagrangian. The inequality (4.41) and the last equality (4.42) follow immediately by applying the triangular and Cauchy–Schwarz inequalities, together with the properties of the spectral norm of matrix \mathbf{A} (*cf.* Remark 5). Finally, by applying (4.39) in (4.42) and rearranging the terms, we obtain the intended result

$$g(\boldsymbol{\lambda}^*) - g(\boldsymbol{\lambda}) + S \|\boldsymbol{\lambda}\| \sqrt{g(\boldsymbol{\lambda}^*) - g(\boldsymbol{\lambda})} \geq \|f(\mathbf{y}(\boldsymbol{\lambda})) - f(\mathbf{y}^*)\|,$$

where $S = \sqrt{(4 + 4 \cos(\pi/m))/\mu}$. □

4.2 Convergence Analysis: Global Consensus

In this section, we analyze the convergence properties of Algorithm 5 and Algorithm 6, which are modeled based on the global consensus problem (3.1). In particular, we discuss the convergence properties under two main cases:

1. **CASE 1:** The local objective functions f_i s in problem (3.2) are strongly convex with constants $\mu_i > 0$, $i = 1, \dots, m$ (i.e., Assumption 4.1.1 holds).
2. **CASE 2:** The set \mathcal{Y} in problem (3.2) equals \mathbb{R}^n . Moreover, the local objective functions f_i s in problem (3.2) are differentiable, the gradients ∇f_i s are Lipschitz continuous on \mathbb{R}^n with constants $L_i > 0$, and f_i s are strongly convex with constants $\mu_i > 0$, $i = 1, \dots, m$ (i.e., Assumption 4.1.2 and Assumption 4.1.1 hold).

Roughly speaking, the CASE 1 is representing a scenario where the dual function g is with Lipschitz continuous gradients (*cf.* Proposition 1), and the CASE 2 is representing a scenario where the negative dual function is both strongly convex and with Lipschitz continuous gradients (*cf.* Proposition 1 and Proposition 2).

For each case above, it is useful to restate the Lagrange multiplier update performed by Algorithm 5 (see line 8 of Algorithm 5) or Algorithm 6 (line 7 of Algorithm 6), i.e.,

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \gamma_k \hat{\mathbf{d}}^{(k)}. \quad (4.43)$$

Moreover, recall that the primal solution computed by Algorithm 5 or Algorithm 6 in each iteration $k \in \mathbb{Z}_+^0$ is $\mathbf{y}^{(k)}$, where

$$\mathbf{y}^{(k)} = [(\mathbf{y}_1^{(k)})^T \dots (\mathbf{y}_m^{(k)})^T]^T = \arg \min_{\mathbf{y} \in \bar{\mathcal{Y}}} f(\mathbf{y}) + (\boldsymbol{\lambda}^{(k)})^T \mathbf{A}\mathbf{y}. \quad (4.44)$$

It is worth noting that, for each case, first we derive convergence results of the sequences of Lagrange multipliers $\{\boldsymbol{\lambda}^{(k)}\}$, which represent convergences in the dual domain. Secondly, the convergences of the primal solutions $\{\mathbf{y}^{(k)}\}$ and the primal objective function values $\{f(\mathbf{y}^{(k)})\}$ are derived. Finally, in the latter part of this chapter, we discuss how a feasible point $\tilde{\mathbf{y}}^{(k)}$ to problem (3.2) is computed by using $\mathbf{y}^{(k)}$. More importantly, convergences of the sequences $\{\tilde{\mathbf{y}}^{(k)}\}$ and $\{f(\tilde{\mathbf{y}}^{(k)})\}$ are also mathematically substantiated.

It is worth emphasizing that, we derive all the convergence results under two step size rules. In particular, we consider

- 1) constant step size rule: i.e., $\gamma_k = \gamma, \forall k$,
- 2) nonsummable step size rule: i.e.,

$$\sum_{k=0}^{\infty} \gamma_k = \infty. \quad (4.45)$$

Remark 13. *The constant step size rule $\gamma_k = \gamma, \forall k$, is a particular case of the non-summable step size rule (4.45).*

Here, it is important to note that, many works in the literature which have analyzed convergence results using the nonsummable step size rule (4.45), are considered with some other additional conditions, such as the square summability of step sizes γ_k or diminishing condition of γ_k [17, 18]. These particular step size rules are usually known as the square summable but not summable step size rule and nonsummable diminishing step size rule, respectively (*cf.* section 2.2.2). However, in our study, we only consider the nonsummable condition (4.45) for our derivations.

4.2.1 Key Remarks, and Related Results

Before establishing convergence results, we outline some useful results in this section. First, a consequence of Assumption 2, which is an upper bound on the overall distortion due to imperfect subproblem coordination is outlined below.

Remark 14. *Assumption 2 entails an absolute deterministic distortion of $\mathbf{d}^{(k)}$ [*cf.* (3.9) and (3.10)]. In particular, we have,*

$$\|\hat{\mathbf{d}}^{(k)} - \mathbf{d}^{(k)}\| \leq \epsilon \quad (4.46)$$

for both Algorithm 5 and Algorithm 6, where

$$\epsilon = \sqrt{\sum_{i=1}^{m-1} (\varepsilon_i + \varepsilon_{i+1})^2}. \quad (4.47)$$

We let $h = -g$, that is the negative dual function, for clarity. It is worth noting that the function h is differentiable as remarked below.

Remark 15. *Assumption 1 together with Assumption 4.1.1, entail the differentiability of h on $\mathbb{R}^{n(m-1)}$.*

Finally, we record a lemma highlighting a recursive inequality that is useful when asserting convergences of both cases, i.e., CASE 1 and CASE 2.

Lemma 7. *Suppose Assumption 1, Assumptions 2, and Assumption 4.1.1 hold. Let γ_i satisfy the condition $0 < \gamma_i \leq 1/L_h$ for all $i \in \mathbb{Z}_+^0$. Then, the function h evaluated at Lagrange multipliers computed in consecutive iterations k , and $k + 1$ of Algorithm 5 conforms to*

$$h(\boldsymbol{\lambda}^{(k+1)}) \leq h(\boldsymbol{\lambda}^{(k)}) - \frac{\gamma_k}{2} \|\nabla h(\boldsymbol{\lambda}^{(k)})\|^2 + \frac{\gamma_k}{2} \epsilon^2, \quad (4.48)$$

where $L_h = (1/\mu)(2 + 2\cos(\pi/m))$, with $\mu = \min_i \mu_i$. The same holds for Algorithm 6.

Proof. Assumption 1 and Assumption 4.1.1 entail that Proposition 1 holds. Thus, by using descent lemma [88, Section 5.1.1] (see also Lemma 2), we have

$$h(\boldsymbol{\gamma}) \leq h(\boldsymbol{\delta}) + \nabla h(\boldsymbol{\delta})^T(\boldsymbol{\gamma} - \boldsymbol{\delta}) + \frac{L_h}{2} \|\boldsymbol{\gamma} - \boldsymbol{\delta}\|^2, \quad \forall \boldsymbol{\gamma}, \boldsymbol{\delta} \in \mathbb{R}^{n(m-1)}. \quad (4.49)$$

Now let $\boldsymbol{\gamma} = \boldsymbol{\lambda}^{(k+1)}$ and $\boldsymbol{\delta} = \boldsymbol{\lambda}^{(k)}$. Thus we have $\boldsymbol{\gamma} - \boldsymbol{\delta} = \gamma_k \hat{\mathbf{d}}^{(k)}$ [cf. (4.43)]. Moreover, note that $\mathbf{r}^{(k)} = \hat{\mathbf{d}}^{(k)} - \mathbf{d}^{(k)}$ and $\mathbf{d}^{(k)} = -\nabla h(\boldsymbol{\lambda}^{(k)})$ (cf. Remark 15). Then, starting from (4.49) we have

$$h(\boldsymbol{\lambda}^{(k+1)}) \leq h(\boldsymbol{\lambda}^{(k)}) + \gamma_k \nabla h(\boldsymbol{\lambda}^{(k)})^T \hat{\mathbf{d}}^{(k)} + \frac{\gamma_k^2 L_h}{2} \|\hat{\mathbf{d}}^{(k)}\|^2 \quad (4.50)$$

$$\begin{aligned} &= h(\boldsymbol{\lambda}^{(k)}) + \gamma_k \nabla h(\boldsymbol{\lambda}^{(k)})^T (\mathbf{r}^{(k)} - \nabla h(\boldsymbol{\lambda}^{(k)})) \\ &\quad + \frac{\gamma_k^2 L_h}{2} (\mathbf{r}^{(k)} - \nabla h(\boldsymbol{\lambda}^{(k)}))^T (\mathbf{r}^{(k)} - \nabla h(\boldsymbol{\lambda}^{(k)})) \end{aligned} \quad (4.51)$$

$$\begin{aligned} &= h(\boldsymbol{\lambda}^{(k)}) - \gamma_k \|\nabla h(\boldsymbol{\lambda}^{(k)})\|^2 + \gamma_k \nabla h(\boldsymbol{\lambda}^{(k)})^T \mathbf{r}^{(k)} \\ &\quad + \frac{\gamma_k^2 L_h}{2} (\|\nabla h(\boldsymbol{\lambda}^{(k)})\|^2 - 2\nabla h(\boldsymbol{\lambda}^{(k)})^T \mathbf{r}^{(k)} + \|\mathbf{r}^{(k)}\|^2) \end{aligned} \quad (4.52)$$

$$\leq h(\boldsymbol{\lambda}^{(k)}) - \frac{\gamma_k}{2} \|\nabla h(\boldsymbol{\lambda}^{(k)})\|^2 + \frac{\gamma_k}{2} \|\mathbf{r}^{(k)}\|^2 \quad (4.53)$$

$$\leq h(\boldsymbol{\lambda}^{(k)}) - \frac{\gamma_k}{2} \|\nabla h(\boldsymbol{\lambda}^{(k)})\|^2 + \frac{\gamma_k}{2} \epsilon^2, \quad (4.54)$$

where (4.50), (4.51) and (4.52) are straightforward by simple calculations and that $\hat{\mathbf{d}}^{(k)} = \mathbf{r}^{(k)} - \nabla h(\boldsymbol{\lambda}^{(k)})$. The inequality (4.53) follows from that $0 < \gamma_k \leq 1/L_h$ for all $k \in \mathbb{Z}_+^0$,

and (4.54) follows from Remark 14. □

4.2.2 Convergence Analysis: CASE 1

This section presents the convergence properties of both Algorithm 5 and Algorithm 6 under CASE 1, along with their rates of convergences. First, a result that is useful when asserting convergences using both constant and nonsummable step size rules is presented below.

Lemma 8. *Suppose Assumption 1, Assumptions 2, and Assumption 4.1.1 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6 with the step size γ_i satisfying the condition $0 < \gamma_i \leq 1/L_h$ for all $i \in \mathbb{Z}_+^0$. Then*

$$\min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\| \leq \sqrt{\frac{2 \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right)}{\sum_{i=0}^k \gamma_i}} + \epsilon, \quad (4.55)$$

where $L_h = (1/\mu) (2 + 2 \cos(\pi/m))$, with $\mu = \min_i \mu_i$.

Proof. By using the recursive application of (4.48) [cf. Lemma 7], we get

$$h(\boldsymbol{\lambda}^{(k+1)}) \leq h(\boldsymbol{\lambda}^{(0)}) - \frac{1}{2} \sum_{i=0}^k \gamma_i \|\nabla h(\boldsymbol{\lambda}^{(i)})\|^2 + \frac{\epsilon^2}{2} \sum_{i=0}^k \gamma_i. \quad (4.56)$$

Rearranging terms of (4.56), yields

$$\begin{aligned} \sum_{i=0}^k \gamma_i \|\nabla h(\boldsymbol{\lambda}^{(i)})\|^2 &\leq 2(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^{(k+1)})) + \epsilon^2 \sum_{i=0}^k \gamma_i \\ &\leq 2 \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right) + \epsilon^2 \sum_{i=0}^k \gamma_i, \end{aligned} \quad (4.57)$$

where (4.57) is immediate from that $h(\boldsymbol{\lambda}^{(k)}) \geq h(\boldsymbol{\lambda}^*)$, for all $k \in \mathbb{Z}_+^0$. Here $\boldsymbol{\lambda}^*$ is a dual solution [cf. Assumption 3]. Because $\min_j \|\nabla h(\boldsymbol{\lambda}^{(j)})\|^2 \leq \|\nabla h(\boldsymbol{\lambda}^{(i)})\|^2$ for all

$i \in \{0, \dots, k\}$, (4.57) implies

$$\min_{k \in \{1, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(j)})\|^2 \leq \frac{2 \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right)}{\sum_{i=0}^k \gamma_i} + \epsilon^2. \quad (4.58)$$

Since $\sqrt{\min_j \|\cdot\|^2} = \min_j \sqrt{\|\cdot\|^2}$, (4.58) ensures that

$$\begin{aligned} \min_j \|\nabla h(\boldsymbol{\lambda}^{(j)})\| &\leq \sqrt{\frac{2 \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right)}{\sum_{i=0}^k \gamma_i} + \epsilon^2} \\ &\leq \sqrt{\frac{2 \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right)}{\sum_{i=0}^k \gamma_i}} + \epsilon, \end{aligned} \quad (4.59)$$

where (4.59) immediately follows from that $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$ for all $x, y \geq 0$. \square

4.2.2.1 Constant Step Size Rule

In this section, we derive convergence results in the dual domain and those related to the primal domain (primal optimality violations) using the constant step size rule under CASE 1, along with their rates of convergences.

First, the convergence results in the dual domain are presented below.

Corollary 1. *Suppose Assumption 1, Assumptions 2, and Assumption 4.1.1 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6. Moreover, suppose $\gamma_k = \gamma$, $\forall k \in \mathbb{Z}_+^0$ with $0 < \gamma \leq 1/L_h$. Then*

1. $\lim_k \min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\| \leq \epsilon$.
2. $\min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\| = O\left(\frac{1}{\sqrt{k}}\right) + \epsilon$.

Proof. Let $\gamma_i = \gamma$, $\forall i \in \mathbb{Z}_+^0$ in Lemma 8, where $0 < \gamma \leq 1/L_h$. Then, first part of Corollary 1 follows immediately from Lemma 8, because, $\lim_{k \rightarrow \infty} \sum_{i=0}^k \gamma_i = \lim_{k \rightarrow \infty} (k + 1)\gamma = \infty$.

To claim the second part, we start with (4.55) in Lemma 8 with $\gamma_i = \gamma$, $\forall i \in \mathbb{Z}_+^0$.

$$\begin{aligned} \min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\| &\leq \sqrt{\frac{2 \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right)}{\sum_{i=0}^k \gamma}} + \epsilon \\ &= \sqrt{\frac{2 \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right)}{(k+1)\gamma}} + \epsilon \end{aligned} \quad (4.60)$$

$$\leq \sqrt{\frac{2 \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right)}{\gamma}} \frac{1}{\sqrt{k}} + \epsilon \quad (4.61)$$

where (4.60) is immediate by simple calculation, (4.61) follows because $k < k+1$, $\forall k \in \mathbb{Z}_+^0$. Finally, the result $\min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\| = O\left(1/\sqrt{k}\right) + \epsilon$ follows using asymptotic notation “ O ” (cf. Definition 20). \square

Corollary 1 indicates that, the minimal norm gradients $\min_i \|\nabla h(\boldsymbol{\lambda}^{(i)})\|$ can converge to a neighborhood around 0 at a rate of order $O(1/\sqrt{k})$, where the size of the neighborhood depends on the level of the distortion ϵ [cf. (4.46)]. It is straightforward to see that the fastest rate corresponding to the fixed step size rule is attained with the largest step size in the range $0 < \gamma \leq 1/L_h$, i.e., with $\gamma = 1/L_h$ [cf. (4.61)].

Next, the convergences of the primal variables and the primal objective values are established in the following proposition.

Proposition 4.2.1. *Suppose Assumption 1, Assumptions 2, Assumption 3, and Assumption 4.1.1 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6, and $\{\mathbf{y}^{(k)}\}$ be the corresponding sequence of primal variables. Moreover, suppose that the functions f_i , $i = 1, \dots, m$ are differentiable. Let $\gamma_k = \gamma$, $\forall k \in \mathbb{Z}_+^0$ with $0 < \gamma \leq 1/L_h$. If the distance from $\boldsymbol{\lambda}^{(k)}$ to the dual optimal solution $\boldsymbol{\lambda}^*$, i.e., $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|$ is uniformly bounded by some scalar D , then*

1. $\lim_k \min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\| \leq \sqrt{2D\epsilon/\mu}$.
2. $\lim_k \min_{i \in \{0, \dots, k\}} (f(\mathbf{y}^{(i)}) - f(\mathbf{y}^*)) \leq D\epsilon + \sqrt{D}S(D + \|\boldsymbol{\lambda}^*\|)\sqrt{\epsilon}$, where the positive scalar $S = \sqrt{(4 + 4 \cos(\pi/m))/\mu}$.

3. $\min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\| = O\left(\frac{1}{\sqrt[4]{k}}\right) + \sqrt{2D\epsilon/\mu}.$
4. $\min_{i \in \{0, \dots, k\}} \|f(\mathbf{y}^{(i)}) - f(\mathbf{y}^*)\| = O\left(\frac{1}{\sqrt[4]{k}}\right) + D\epsilon + \sqrt{D}S(D + \|\boldsymbol{\lambda}^*\|)\sqrt{\epsilon}$

Proof. Part 1: Recall that the undistorted local version of the public variable \mathbf{z} is $\mathbf{y}_i \in \mathbb{R}^n$, $i = 1, \dots, m$. Moreover, subsystems solve in parallel the problem

$$\underset{\mathbf{y} \in \mathcal{Y}}{\text{minimize}} \quad f(\mathbf{y}) + (\boldsymbol{\lambda}^{(k)})^T \mathbf{A} \mathbf{y} \quad (4.62)$$

to yield the solution $\mathbf{y}^{(k)} = [\mathbf{y}_1^{(k)}, \dots, \mathbf{y}_m^{(k)}]^T$ (cf. line 4 of Algorithm 5 and lines 3 of Algorithm 6, respectively). Then

$$\|\mathbf{y}^{(k)} - \mathbf{y}^*\|^2 \leq (2/\mu) \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right) \quad (4.63)$$

$$\leq (2/\mu) \nabla h(\boldsymbol{\lambda}^{(k)})^T (\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*) \quad (4.64)$$

$$\leq (2/\mu) \|\nabla h(\boldsymbol{\lambda}^{(k)})\| \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\| \quad (4.65)$$

$$\leq (2D/\mu) \|\nabla h(\boldsymbol{\lambda}^{(k)})\|, \quad (4.66)$$

where (4.63) follows from the part 1 of Lemma 6, (4.64) follows immediately due to the convexity and differentiability of h (cf. Theorem 1), (4.65) follows from Cauchy–Schwarz inequality, and finally (4.66) follows directly using the uniform boundedness of $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|$ by D . Then from (4.66) it is straightforward that

$$\min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\|^2 \leq (2D/\mu) \min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\|. \quad (4.67)$$

Next, because $\sqrt{\min_j \|\cdot\|^2} = \min_j \sqrt{\|\cdot\|^2}$, (4.67) yields

$$\min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\| \leq \sqrt{(2D/\mu)} \sqrt{\min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\|}. \quad (4.68)$$

Finally, equation (4.68) together with part 1 of Corollary 1 yields the first claim of Propo-

sition 4.2.1, i.e.,

$$\lim_k \min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\| \leq \sqrt{2D\epsilon/\mu}. \quad (4.69)$$

Part 2: By using the part 2 of Lemma 6, we have

$$\begin{aligned} f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*) &\leq (h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*)) + S\|\boldsymbol{\lambda}^{(k)}\| \sqrt{h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*)} \\ &\leq D\|\nabla h(\boldsymbol{\lambda}^{(k)})\| + \sqrt{DS}\|\boldsymbol{\lambda}^{(k)}\| \sqrt{\|\nabla h(\boldsymbol{\lambda}^{(k)})\|} \end{aligned} \quad (4.70)$$

$$\leq D\|\nabla h(\boldsymbol{\lambda}^{(k)})\| + \sqrt{DS}(D + \|\boldsymbol{\lambda}^*\|) \sqrt{\|\nabla h(\boldsymbol{\lambda}^{(k)})\|}, \quad (4.71)$$

where (4.70) follows directly using (4.66), and (4.71) follows from the supposition $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\| \leq D$, $\forall k \in \mathbb{Z}_+^0$ together with that $\|\boldsymbol{\lambda}^{(k)}\| - \|\boldsymbol{\lambda}^*\| \leq \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|$. Thus, the second part of the proposition follows from part 1 of Corollary 1 and (4.71).

Part 3: To prove the third claim, we start with equation (4.68). Then, the inequality (4.68) together with part 2 of Corollary 1 yields

$$\begin{aligned} \min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\| &\leq \sqrt{(2D/\mu)} \sqrt{O\left(\frac{1}{\sqrt{k}}\right) + \epsilon} \\ &\leq \sqrt{(2D/\mu)} \left(O\left(\frac{1}{\sqrt[4]{k}}\right) + \sqrt{\epsilon} \right) \end{aligned} \quad (4.72)$$

where (4.72) follows from that $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$ for all $x, y \geq 0$. Thus we have $\min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\| = O\left(1/\sqrt[4]{k}\right) + \sqrt{2D\epsilon/\mu}$, the intended result (cf. Definition 20).

Part 4: The inequality (4.71) yields that

$$\begin{aligned}
\min_{i \in \{0, \dots, k\}} \|f(\mathbf{y}^{(i)}) - f(\mathbf{y}^*)\| &\leq D \min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\| \\
&\quad + \sqrt{DS}(D + \|\boldsymbol{\lambda}^*\|) \sqrt{\min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\|} \\
&\leq D \left(O\left(\frac{1}{\sqrt{k}}\right) + \epsilon \right) \\
&\quad + \sqrt{DS}(D + \|\boldsymbol{\lambda}^*\|) \sqrt{\left(O\left(\frac{1}{\sqrt{k}}\right) + \epsilon \right)} \quad (4.73)
\end{aligned}$$

$$\begin{aligned}
&\leq D \left(O\left(\frac{1}{\sqrt{k}}\right) + \epsilon \right) \\
&\quad + \sqrt{DS}(D + \|\boldsymbol{\lambda}^*\|) \left(O\left(\frac{1}{\sqrt[4]{k}}\right) + \sqrt{\epsilon} \right), \quad (4.74)
\end{aligned}$$

where (4.73) follows using part 2 of Corollary 1 and (4.74) follows again from that $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$ for all $x, y \geq 0$. Thus, the fourth part of the proposition immediately follows from (4.74) together with the asymptotic notation “ O ” (cf. Definition 20). \square

4.2.2.2 Nonsummable Step size Rule

Convergence properties in both dual and primal domains using the nonsummable step size rule under CASE 1 are presented in this section.

First, the following result establishes the convergences in the dual domain.

Corollary 2. *Suppose Assumption 1, Assumptions 2, and Assumption 4.1.1 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6. Moreover, suppose γ_k satisfies the nonsummable step size rule given in (4.45) with $0 < \gamma_k \leq 1/L_h$. Then*

1. $\lim_k \min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\| \leq \epsilon.$

2. for $\gamma_k = \gamma/(k+1)^p$, where $0 < \gamma \leq 1/L_h$ and $0 \leq p \leq 1$,

$$\min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\| = \begin{cases} O\left(\frac{1}{\sqrt{k^{1-p}}}\right) + \epsilon & p \in [0, 1) \\ O\left(\frac{1}{\sqrt{\ln k}}\right) + \epsilon & p = 1. \end{cases} \quad (4.75)$$

Proof. The first part of the Corollary 2 follows immediately from Lemma 8 and from that

$$\lim_{k \rightarrow \infty} \sum_{i=0}^k \gamma_i = \infty \text{ [cf. (4.45)].}$$

Next, to claim the second part, we start with the summation $\sum_{i=0}^k (1/(i+1)^p)$. Then, clearly we have

$$\sum_{i=0}^k \frac{1}{(i+1)^p} \geq \int_0^{k+1} \frac{1}{(x+1)^p} dx \quad (4.76)$$

$$= \begin{cases} \frac{(k+2)^{1-p} - 1}{1-p} & ; p \in [0, 1) \\ \ln(k+2) & ; p = 1. \end{cases} \quad (4.77)$$

Next, using equation (4.77) together with Lemma 8 [cf. equation (4.55)] yield

$$\min_{i \in \{0, \dots, k\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\| \leq \begin{cases} \sqrt{\frac{2(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*)) (1-p)}{\gamma((k+2)^{1-p} - 1)}} + \epsilon & ; p \in [0, 1) \\ \sqrt{\frac{2(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*))}{\gamma \ln(k+2)}} + \epsilon & ; p = 1 \end{cases} \\ \leq \begin{cases} \frac{K_1}{\sqrt{(k^{1-p} - 1)}} + \epsilon & ; p \in [0, 1) \\ \frac{K_2}{\sqrt{\ln k}} + \epsilon & ; p = 1, \end{cases} \quad (4.78)$$

where $K_1 = 2(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*)) (1-p)/\gamma$ and $K_2 = 2(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*)) / \gamma$. The last inequality (4.78) follows because, $k < k+2 \forall k \in \mathbb{Z}_+^0$. Finally, the result holds with the asymptotic representation of the right-hand side of the inequality (4.78). \square

Corollary 2 indicates that, the minimal norm gradients $\min_i \|\nabla(h(\boldsymbol{\lambda}^{(i)}))\|$ can converge

to a neighborhood around 0 at a rate of order $O(1/\sqrt{k^{1-p}})$ and $O(1/\sqrt{\ln k})$, when $p \in [0, 1)$ and $p = 1$ respectively, where the size of the neighborhood depends on ϵ [cf. (4.46)].

It is worth noting that, the Corollary 2 directly reduces to Corollary 1 when $p = 0$. Moreover, it is straightforward to see that the fastest rate is achieved when $p = 0$, which corresponds to the fixed step size rule with $\gamma_k = 1/L_h$ for all $k \in \mathbb{Z}_+^0$, and it is of the order $O(1/\sqrt{k})$.

Convergences of the primal variables and the primal objective values are established in the following proposition.

Proposition 4.2.2. *Suppose Assumption 1, Assumptions 2, Assumption 3, and Assumption 4.1.1 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6 and $\{\mathbf{y}^{(k)}\}$ be the corresponding sequence of primal variables. Moreover, suppose that the functions f_i , $i = 1, \dots, m$ are differentiable. Let γ_k satisfies the nonsummable step size rule given in (4.45) with $0 < \gamma_k \leq 1/L_h$. If the distance from $\boldsymbol{\lambda}^{(k)}$ to the dual optimal solution $\boldsymbol{\lambda}^*$, i.e., $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|$ is uniformly bounded by some scalar D , then*

1. $\lim_k \min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\| \leq \sqrt{2D\epsilon/\mu}$.
2. $\lim_k \min_{i \in \{0, \dots, k\}} (f(\mathbf{y}^{(i)}) - f(\mathbf{y}^*)) \leq D\epsilon + \sqrt{DS}(D + \|\boldsymbol{\lambda}^*\|)\sqrt{\epsilon}$, where the positive scalar $S = \sqrt{(4 + 4 \cos(\pi/m))/\mu}$.
3. for $\gamma_k = \gamma/(k+1)^p$, where $0 < \gamma \leq 1/L_h$ and $0 \leq p \leq 1$,

$$\min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\| = \begin{cases} O\left(\frac{1}{\sqrt[4]{k^{1-p}}}\right) + \sqrt{2D\epsilon/\mu}; & p \in [0, 1) \\ O\left(\frac{1}{\sqrt[4]{\ln k}}\right) + \sqrt{2D\epsilon/\mu}; & p = 1. \end{cases} \quad (4.79)$$

4. for $\gamma_k = \gamma/(k+1)^p$, where $0 < \gamma \leq 1/L_h$ and $0 \leq p \leq 1$,

$$\min_{i \in \{0, \dots, k\}} \|f(\mathbf{y}^{(i)}) - f(\mathbf{y}^*)\| = \begin{cases} O\left(\frac{1}{\sqrt[4]{k^{1-p}}}\right) + D\epsilon + \sqrt{DS}(D + \|\boldsymbol{\lambda}^*\|)\sqrt{\epsilon}; & p \in [0, 1) \\ O\left(\frac{1}{\sqrt[4]{\ln k}}\right) + D\epsilon + \sqrt{DS}(D + \|\boldsymbol{\lambda}^*\|)\sqrt{\epsilon}; & p = 1. \end{cases} \quad (4.80)$$

Proof. The proof of Proposition 4.2.2 is similar to that presented in the proof of Proposition 4.2.1. Thus the proof is omitted. \square

It is worth pointing out that, Proposition 4.2.1 and Proposition 4.2.2 for primal domain convergences rely on few additional hypotheses, unlike Corollary 1 and Corollary 2 in which the convergences are established in the dual-domain.

4.2.3 Convergence Analysis: CASE 2

In this section, we present the convergence properties of both Algorithm 5 and Algorithm 6 under CASE 2, along with their rates of convergences.

A useful result that is employed when deriving convergence results using both constant and nonsummable step size rules is presented below.

Lemma 9. *Suppose Assumption 1, Assumptions 2, Assumption 4.1.1, and Assumption 4.1.2 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6 with the step size γ_i satisfying the condition $0 < \gamma_i \leq 1/L_h$ for all $i \in \mathbb{Z}_+^0$.*

Then

$$h(\boldsymbol{\lambda}^{(k+1)}) - h(\boldsymbol{\lambda}^*) \leq (1 - \gamma_k \mu_h) \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right) + \frac{\gamma_k}{2} \epsilon^2, \quad (4.81)$$

where $L_h = (1/\mu)(2 + 2\cos(\pi/m))$, $\mu = \min_i \mu_i$, and $\mu_h = (1/L)(2 - 2\cos(\pi/m))$, with $L = \max_i L_i$.

Proof. Since Assumption 4.1.2 holds, $h(\boldsymbol{\lambda})$ is strongly convex with constant μ_h (cf. Proposition 2). Hence we have that $\|\nabla h(\boldsymbol{\lambda}^{(k)})\|^2 \geq 2\mu_h \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right)$ [11, page 24]. This together with Lemma 7 yields

$$h(\boldsymbol{\lambda}^{(k+1)}) - h(\boldsymbol{\lambda}^*) \leq (1 - \gamma_k \mu_h) \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right) + \frac{\gamma_k}{2} \epsilon^2,$$

the intended result. □

It is worth noting that $0 \leq 1 - \gamma_k \mu_h < 1$ in Lemma 9, because $0 < \gamma_k \leq 1/L_h$ and $\mu_h \leq L_h$.

4.2.3.1 Constant Step Size Rule

The convergence results for the constant step size rule are presented in this section. The following result is immediate from Lemma 9.

Corollary 3. *Suppose Assumption 1, Assumptions 2, Assumption 4.1.1 and Assumption 4.1.2 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6 with the step size $\gamma_k = \gamma$ for all $k \in \mathbb{Z}_+^0$. Then for $0 < \gamma \leq 1/L_h$*

$$h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \leq (1 - \gamma \mu_h)^k \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right) + \frac{\epsilon^2}{2\mu_h}. \quad (4.82)$$

Moreover,

$$\limsup_k \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right) \leq \frac{\epsilon^2}{2\mu_h}. \quad (4.83)$$

Proof. Using the recursive application of (4.81) with $\gamma_k = \gamma$ yields

$$\begin{aligned} h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) &\leq (1 - \gamma\mu_h)^k \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right) + \frac{\gamma\epsilon^2}{2} \sum_{i=0}^{k-1} (1 - \gamma\mu_h)^i \\ &\leq (1 - \gamma\mu_h)^k \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right) + \frac{\gamma\epsilon^2}{2} \sum_{i=0}^{\infty} (1 - \gamma\mu_h)^i \end{aligned} \quad (4.84)$$

$$= (1 - \gamma\mu_h)^k \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right) + \frac{\epsilon^2}{2\mu_h}, \quad (4.85)$$

where (4.84) follows using that the infinite sum $\sum_{i=0}^{\infty} (1 - \gamma\mu_h)^i$ is larger than the finite sum $\sum_{i=0}^{k-1} (1 - \gamma\mu_h)^i$, and (4.85) is straightforward using that the geometric sum $\sum_{i=0}^{\infty} (1 - \gamma\mu_h)^i = 1/\gamma\mu_h$.

The second part of the corollary follows trivially because $\limsup_k (1 - \gamma\mu_h)^k = 0$ (remind that $0 \leq 1 - \gamma\mu_h < 1$ and $h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) < \infty$). \square

According to Corollary 3, with constant step size, the least upper bound of $h(\boldsymbol{\lambda}^{(k)})$ converges into a neighborhood of the optimal value $h(\boldsymbol{\lambda}^*)$ with the rate of geometric progression. In this case, the size of the neighborhood depends on both ϵ [cf. (4.46)] and the constant μ_h that characterizes the strong convexity of h (cf. Proposition 2).

The convergences of the primal variables and the primal objective function values are asserted in the following proposition.

Proposition 4.2.3. *Suppose Assumption 1, Assumptions 2, Assumption 3, Assumption 4.1.1, and Assumption 4.1.2 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6 and $\{\mathbf{y}^{(k)}\}$ be the corresponding sequence of primal variables. Moreover, suppose that the functions f_i , $i = 1, \dots, m$ are differentiable. Let the step size $\gamma_k = \gamma$ for all $k \in \mathbb{Z}_+^0$. Then for $0 < \gamma \leq 1/L_h$*

1. $\limsup_k \|\mathbf{y}^{(k)} - \mathbf{y}^*\| \leq \epsilon \sqrt{1/(\mu\mu_h)}$.
2. $\limsup_k \|f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*)\| \leq (1 + S\sqrt{2/\mu_h}) \frac{\epsilon^2}{2\mu_h} + \frac{S\|\boldsymbol{\lambda}^*\|\epsilon}{\sqrt{2\mu_h}}$, where the positive scalar $S = \sqrt{(4 + 4\cos(\pi/m))/\mu}$.

3. the least upper bound of $\|\mathbf{y}^{(k)} - \mathbf{y}^*\|$ converges into a neighbourhood of 0 with the rate of geometric progression.
4. the least upper bound of $\|f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*)\|$ converges into a neighbourhood of 0 with the rate of geometric progression.

Proof. The steps of the proof are similar to that of Proposition 4.2.1. Let $\mathbf{y}^{(k)}$ be the solution to the problem (4.62). Then, the norm-squared error in the primal solution is

$$\|\mathbf{y}^{(k)} - \mathbf{y}^*\|^2 \leq (2/\mu) \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right) \quad (4.86)$$

$$\leq (2/\mu) (1 - \gamma_k \mu_h)^k \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right) + \epsilon^2 / (\mu \mu_h), \quad (4.87)$$

where (4.86) follows from part 1 of Lemma 6 and (4.87) follows from Corollary 3. Thus the claims 1 and 3 of the proposition are immediate from (4.87).

Next, to prove the second and fourth claims, we start with part 2 of Lemma 6.

$$\begin{aligned} f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*) &\leq (h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*)) + S \|\boldsymbol{\lambda}^{(k)}\| \sqrt{h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*)} \\ &\leq (h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*)) + S \left(\|\boldsymbol{\lambda}^*\| + \sqrt{(2/\mu_h)(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*))} \right) \times \\ &\quad \sqrt{h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*)} \end{aligned} \quad (4.88)$$

$$= (1 + S\sqrt{2/\mu_h})(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*)) + S\|\boldsymbol{\lambda}^*\| \sqrt{h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*)}, \quad (4.89)$$

where $S = \sqrt{(4 + 4 \cos(\pi/m))/\mu}$. The inequality (4.88) follows from the strong convexity of h (cf. Proposition 2), and (4.89) follows using simple calculations. In particular, from [11, page 11, Equation (35)] we have

$$\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|^2 \leq \frac{2}{\mu_h} \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right),$$

which in turn ensures that $\|\boldsymbol{\lambda}^{(k)}\| \leq \|\boldsymbol{\lambda}^*\| + \sqrt{(2/\mu_h)(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*))}$ since $\|\boldsymbol{\lambda}^{(k)}\| -$

$\|\boldsymbol{\lambda}^*\| \leq \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|$. Thus, (4.89) together with (4.82) yields

$$\begin{aligned} f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*) &\leq (1 + S\sqrt{2/\mu_h}) \left((1 - \gamma\mu_h)^k \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right) + \frac{\epsilon^2}{2\mu_h} \right) \\ &\quad + S\|\boldsymbol{\lambda}^*\| \sqrt{(1 - \gamma\mu_h)^k \left(h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) \right) + \frac{\epsilon^2}{2\mu_h}} \end{aligned} \quad (4.90)$$

Finally, claims 2 and 4 of the proposition are straightforward using (4.90). \square

Unlike Propositions 4.2.1 and 4.2.2, in which the distance to the dual optimal solution $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|$ is assumed to be uniformly bounded by some scalar D , in Proposition 4.2.3, no such assumptions are made. This is a consequence of the strong convexity of g .

4.2.3.2 Nonsummable Step Size Rule

This section presents convergence results using the nonsummable step size rule under CASE 2. First, we present the following lemma, which plays a key role in asserting related convergence results.

Lemma 10 ([11], Section 2.2, Lemma 3). *Let $u^{(k)}$ be a sequence such that*

$$u^{(k+1)} \leq p^{(k)}u^{(k)} + \alpha^{(k)}, \quad (4.91)$$

where $0 \leq p^{(k)} < 1$ and $\alpha^{(k)} \geq 0$ with

$$\sum_{k=0}^{\infty} (1 - p^{(k)}) = \infty, \quad \text{and} \quad \lim_{k \rightarrow \infty} \frac{\alpha^{(k)}}{(1 - p^{(k)})} = 0. \quad (4.92)$$

Then, $\limsup_k u^{(k)} \leq 0$. If $u^{(k)} \geq 0$, then $u^{(k)} \rightarrow 0$.

Corollary 4. *Suppose Assumption 1, Assumptions 2, Assumption 4.1.1, and Assumption 4.1.2 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6. Moreover, suppose γ_k satisfies the nonsummable step size rule given in (4.45) with $0 < \gamma_k \leq 1/L_h$. Then*

$$1. \limsup_k h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \leq \frac{\epsilon^2}{2\mu_h}.$$

2. for $\gamma_k = (c/\mu_h)/(k+1)^p$, where $0 < p \leq 1$ and $0 < c \leq \mu_h/L_h$,

$$h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) = O\left(\frac{1}{k^{c/p}}\right) + \frac{\epsilon^2}{2\mu_h}. \quad (4.93)$$

Proof. By subtracting the term $(1 - \gamma_k\mu_h)\epsilon^2/2\mu_h$ on both sides of the inequality (4.81)

(cf. Lemma 9) yields

$$\begin{aligned} h(\boldsymbol{\lambda}^{(k+1)}) - h(\boldsymbol{\lambda}^*) - (1 - \gamma_k\mu_h)\frac{\epsilon^2}{2\mu_h} &\leq (1 - \gamma_k\mu_h) \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right) \\ &\quad + \frac{\gamma_k}{2}\epsilon^2 - (1 - \gamma_k\mu_h)\frac{\epsilon^2}{2\mu_h}. \end{aligned} \quad (4.94)$$

By rearranging the terms, (4.94) yields

$$h(\boldsymbol{\lambda}^{(k+1)}) - h(\boldsymbol{\lambda}^*) - \frac{\epsilon^2}{2\mu_h} \leq (1 - \gamma_k\mu_h) \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) - \frac{\epsilon^2}{2\mu_h} \right). \quad (4.95)$$

Here, (4.95) is in the same form of (4.91) (cf. Lemma 10) with $p^{(k)} = 1 - \gamma_k\mu_h$, $u^{(k)} = h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) - \epsilon^2/2\mu_h$, and $\alpha^{(k)} = 0$. Then, $\sum_{i=0}^{\infty} (1 - p^{(k)}) = \infty$, because γ_k satisfies the nonsummable step size rule given in (4.45). Moreover, it is easily seen that $\alpha^{(k)}/(1 - p^{(k)}) \rightarrow 0$ as $k \rightarrow \infty$. Hence, Lemma 10 ensures that $\limsup_k h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) - (\epsilon^2/2\mu_h) \leq 0$. Thus we have $\limsup_{k \rightarrow \infty} h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \leq \epsilon^2/2\mu_h$ which completes the proof of part 1.

Next, to prove the second part of the Corollary 4, we let $u^{(k)} = h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) - \epsilon^2/2\mu_h$ and $v^{(k)} = (k+1)^{c/p}u^{(k)}$. Then,

$$v^{(k+1)} = (k+2)^{c/p}u^{(k+1)} \quad (4.96)$$

$$\leq (k+1)^{c/p} \left(1 + \frac{1}{k+1}\right)^{c/p} (1 - \gamma_k\mu_h)u^{(k)} \quad (4.97)$$

$$= \left(1 + \frac{1}{k+1}\right)^{c/p} \left(1 - \frac{c}{(k+1)^p}\right)v^{(k)} \quad (4.98)$$

$$\leq \left(1 + \frac{c}{p(k+1)} + \frac{c^2}{2p^2(k+1)^2} + o\left(\frac{1}{(k+1)^2}\right)\right) \left(1 - \frac{c}{(k+1)^p}\right) v^{(k)} \quad (4.99)$$

$$\leq \left(1 + \frac{c}{(k+1)^p} + \frac{c^2}{2p^2(k+1)^2} + o\left(\frac{1}{(k+1)^2}\right)\right) \left(1 - \frac{c}{(k+1)^p}\right) v^{(k)},$$

for all $k \geq \lceil e^{\frac{\ln p}{(p-1)}} - 1 \rceil$

$$(4.100)$$

$$= \left(1 - \frac{c^2}{(k+1)^{2p}} + \frac{c^2}{2p^2(k+1)^2} + o\left(\frac{1}{(k+1)^2}\right)\right) v^{(k)}, \text{ for all } k \geq \lceil e^{\frac{\ln p}{(p-1)}} - 1 \rceil$$

$$(4.101)$$

$$\leq \left(1 - \frac{c^2}{(k+1)^{2p}} + \frac{c^2}{2(k+1)^{2p}} + o\left(\frac{1}{(k+1)^2}\right)\right) v^{(k)}, \text{ for all } k \geq \lceil e^{\frac{\ln p}{(p-1)}} - 1 \rceil$$

$$(4.102)$$

$$= \left(1 - \frac{c^2}{2(k+1)^{2p}} + o\left(\frac{1}{(k+1)^2}\right)\right) v^{(k)}, \text{ for all } k \geq \lceil e^{\frac{\ln p}{(p-1)}} - 1 \rceil$$

$$(4.103)$$

$$\leq v^{(k)}, \text{ for sufficiently large } k, \quad (4.104)$$

where (4.96) follows using the definition of $v^{(k)}$, (4.97) follows by simple calculations and using (4.95), (4.98) follows by replacing γ_k and $u^{(k)}$ with their definitions, and (4.99) follows simply by using the binomial expansion. When deriving (4.100), we use that $(x+1)^p \leq p(x+1)$ for sufficiently large x , when $p \in (0, 1]$. The equality (4.101) is immediate from simple calculations. The inequality (4.102) follows again from that $(x+1)^{2p} \leq p^2(x+1)^2$ for all sufficiently large x , when $p \in (0, 1]$. The equality (4.103) and the last inequality (4.104) are immediate from simple calculations. Then, summing over k , we get the boundedness of $v^{(k)}$, i.e.,

$$v^{(k)} \leq v^{(0)}. \quad (4.105)$$

Thus, with the definition of $v^{(k)}$, (4.105) yields

$$(k+1)^{c/p} u^{(k)} \leq h(\boldsymbol{\lambda}^{(0)}) - h(\boldsymbol{\lambda}^*) - \epsilon^2/2\mu_h. \quad (4.106)$$

Finally, the inequality (4.106), together with the definition of $u^{(k)}$ yields the result $h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) = O(1/k^{c/p}) + \epsilon^2/2\mu_h$. \square

Corollary 4 indicates that the least upper bound of $h(\boldsymbol{\lambda}^{(k)})$ converges into a neighborhood of the optimal value $h(\boldsymbol{\lambda}^*)$ at a rate of $O(1/k^{c/p})$, where the size of the neighborhood explicitly depends on ϵ and the strong convexity constant μ_h of h . Note that the rate of convergence depends on the ratio c/p . It can easily be observed that, for a given c value, the rate of convergence increases when the value of p decreases, where $0 < c \leq \mu_h/L_h$ and $0 < p \leq 1$.

Note that the case $p = 0$ in the nonsummable step size rule $\gamma_k = (c/\mu_h)/(k+1)^p$ corresponds to a constant step size rule. This suggests, as in the constant step size rule (cf. Corollary 3), that when $p \rightarrow 0$, (4.93) should be a good resemblance of (4.82). According to the proof of Corollary 4, p is to be chosen in such a manner that $(k+1)^p \leq p(k+1)$ for sufficiently large k . One such choice is $p = \log k/k$ (see Figure 4.1). With this choice, we have that $0 < p < 1$ and $p \rightarrow 0$ as $k \rightarrow \infty$. Thus, clearly, for sufficiently large k , the nonsummable step size rule $\gamma_k = (c/\mu_h)/(k+1)^p$ with $p = \log k/k$ corresponds to a constant step size rule $\gamma_k = c/\mu_h$. In particular, The following Lemma will show that $h(\boldsymbol{\lambda}^{(k)})$ converges into a neighborhood of the optimal value $h(\boldsymbol{\lambda}^*)$ at a rate of geometric progression with the above choice of p (See Figure 4.16 for numerical illustrations).

Lemma 11. *Let $s(k; p) = O(1/k^{(c/p)})$ (cf. Definition 20), where $c > 0$ and $0 < p < 1$. Then $s(k; p)$ with $p = \log k/k$ converges to 0 with the rate of geometric progression.*

Proof. Since $s(k; p) = O(1/k^{(c/p)})$ (cf. Definition 20), there exists $\alpha > 0$ and $k_0 \in \mathbb{Z}_0^+$ such that

$$\begin{aligned} s(k; p) &\leq \alpha \left(\frac{1}{k^{c/p}} \right) && \text{for all } k \geq k_0 \\ &= \alpha k^{(-kc/\log k)} && \text{for all } k \geq k_0, \end{aligned} \quad (4.107)$$

where (4.107) follows using $p = \log k/k$.

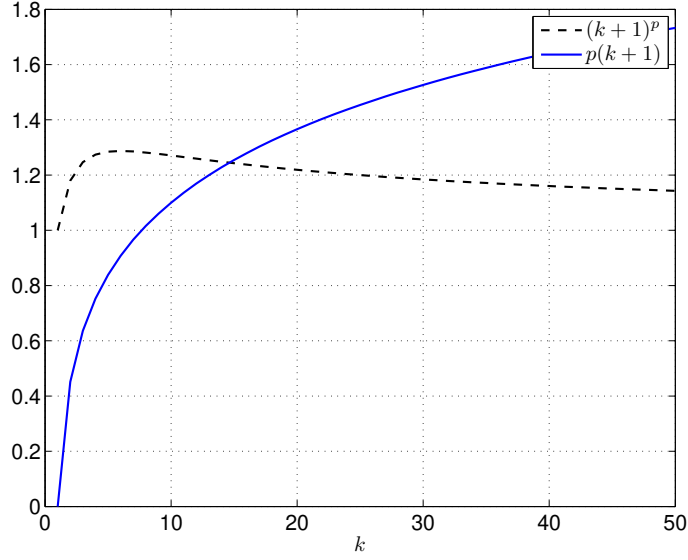


Figure 4.1: Graphs of $(k + p)^p$ and $p(k + 1)$ with $p = \log k/k$. Figure clearly shows $p(k + 1) > (k + p)^p$ for large k .

Next, using logarithms in the inequality (4.107) yields

$$\log(s(k; p)) \leq \log \alpha - kc \quad \text{for all } k \geq k_0. \quad (4.108)$$

Then by simple calculations (4.108) yields

$$s(k; p) \leq R \left(\frac{1}{e^c} \right)^k \quad \text{for all } k \geq k_0, \quad (4.109)$$

where $R = \exp(\log \alpha)$ is a constant. Thus, clearly $s(k; p)$ with $p = \log k/k$ converges to 0 with the rate of geometric progression as $0 < 1/e^c < 1$. \square

Lemma 11 indicates that $h(\boldsymbol{\lambda}^{(k)})$ converges into a neighborhood of the optimal value $h(\boldsymbol{\lambda}^*)$ at a rate of geometric progression with $p = \log k/k$ in $\gamma_k = (c/\mu_h)/(k + 1)^p$ (cf. Equation (4.93) in Corollary 4).

By using Corollary 4, convergence assertions similar to Proposition 4.2.3 for the sequences $\{\mathbf{y}^{(k)}\}$ and $\{f(\mathbf{y}^{(k)})\}$ can be derived analogously, which corresponds to γ_k being nonsummable. The related result is given below.

Proposition 4.2.4. *Suppose Assumption 1, Assumptions 2, Assumption 3, Assumption 4.1.1,*

and Assumption 4.1.2 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6 and $\{\mathbf{y}^{(k)}\}$ be the corresponding sequence of primal variables. Moreover, suppose that the functions f_i , $i = 1, \dots, m$ are differentiable. Let γ_k satisfies the nonsummable step size rule given in (4.45) with $0 < \gamma_k \leq 1/L_h$. Then,

1. $\limsup_k \|\mathbf{y}^{(k)} - \mathbf{y}^*\| \leq \frac{\epsilon}{\sqrt{\mu\mu_h}}$.
2. $\limsup_k \|f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*)\| \leq \left(1 + S\sqrt{\frac{2}{\mu_h}}\right) \frac{\epsilon^2}{2\mu_h} + \frac{S\|\boldsymbol{\lambda}^*\|\epsilon}{\sqrt{2\mu_h}}$, where the positive scalar $S = \sqrt{(4 + 4\cos(\pi/m))/\mu}$.
3. for $\gamma_k = (c/\mu_h)/(k+1)^p$, where $0 < p \leq 1$ and $0 < c \leq \mu_h/L_h$,

$$\|\mathbf{y}^{(k)} - \mathbf{y}^*\| = O\left(\frac{1}{k^{c/2p}}\right) + \frac{\epsilon}{\mu_h}. \quad (4.110)$$

4. for $\gamma_k = (c/\mu_h)/(k+1)^p$, where $0 < p \leq 1$ and $0 < c \leq \mu_h/L_h$,

$$f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*) = O\left(\frac{1}{k^{c/2p}}\right) + \frac{S\epsilon^2}{\mu_h\sqrt{2\mu_h}} + \frac{S\|\boldsymbol{\lambda}^*\|\epsilon}{\sqrt{2\mu_h}} + \frac{\epsilon^2}{2\mu_h}. \quad (4.111)$$

Proof. Part 1: Using part 1 of Lemma 6 we have,

$$\|\mathbf{y}^{(k)} - \mathbf{y}^*\|^2 \leq (2/\mu) \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right). \quad (4.112)$$

Thus the part 1 of the proposition is immediate using (4.112) together with part 1 of corollary 4.

Part 2: The proof of part 2 of the proposition is similar to the proof of part 2 of Proposition 4.2.3. Thus we start with (4.89) of the proof of part 2 of Proposition 4.2.3, which is a consequence of part 2 of Lemma 6 and the strong convexity property of h .

$$f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*) \leq \left(1 + S\sqrt{\frac{2}{\mu_h}}\right) \left(h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*) \right) + S\|\boldsymbol{\lambda}^*\|\sqrt{h(\boldsymbol{\lambda}^{(k)}) - h(\boldsymbol{\lambda}^*)} \quad (4.113)$$

Then, inequality (4.113) together with part 1 of Corollary 4 yields the result

$$\limsup_k \|f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*)\| \leq (1 + S\sqrt{2/\mu_h}) \frac{\epsilon^2}{2\mu_h} + \frac{S\|\boldsymbol{\lambda}^*\|\epsilon}{\sqrt{2\mu_h}}. \quad (4.114)$$

Part 3: The inequality (4.86), which follows from part 1 of Lemma 6 together with part 2 of Corollary 4 yields

$$\begin{aligned} \|\mathbf{y}^{(k)} - \mathbf{y}^*\| &\leq \sqrt{\frac{2}{\mu}} \sqrt{O\left(\frac{1}{k^{c/p}}\right) + \frac{\epsilon^2}{2\mu_h}} \\ &\leq \sqrt{\frac{2}{\mu}} \left(O\left(\frac{1}{k^{c/2p}}\right) + \frac{\epsilon}{\sqrt{2\mu_h}} \right), \end{aligned} \quad (4.115)$$

where (4.115) follows using $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$ for all $x, y \geq 0$ together with the asymptotic notation “ O ” (cf. Definition 20). Thus we have the result $\|\mathbf{y}^{(k)} - \mathbf{y}^*\| = O(1/k^{c/2p}) + \epsilon/\mu_h$ (cf. Definition 20).

Part 4: The inequality (4.113) together with part 2 of Corollary 4 yields

$$\begin{aligned} f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*) &\leq \left(1 + S\sqrt{\frac{2}{\mu_h}}\right) \left(O\left(\frac{1}{k^{c/p}}\right) + \frac{\epsilon^2}{2\mu_h}\right) + S\|\boldsymbol{\lambda}^*\| \sqrt{O\left(\frac{1}{k^{c/p}}\right) + \frac{\epsilon^2}{2\mu_h}} \\ &\leq \left(1 + S\sqrt{\frac{2}{\mu_h}}\right) \left(O\left(\frac{1}{k^{c/p}}\right) + \frac{\epsilon^2}{2\mu_h}\right) + S\|\boldsymbol{\lambda}^*\| \left(O\left(\frac{1}{k^{c/2p}}\right) + \frac{\epsilon}{\sqrt{2\mu_h}}\right) \end{aligned} \quad (4.116)$$

$$\begin{aligned} &= S\|\boldsymbol{\lambda}^*\| O\left(\frac{1}{k^{c/2p}}\right) + \left(1 + S\sqrt{\frac{2}{\mu_h}}\right) O\left(\frac{1}{k^{c/p}}\right) + \frac{S\epsilon^2}{\mu_h\sqrt{2\mu_h}} + \frac{S\|\boldsymbol{\lambda}^*\|\epsilon}{\sqrt{2\mu_h}} \\ &\quad + \frac{\epsilon^2}{2\mu_h}, \end{aligned} \quad (4.117)$$

where (4.116) follows from that $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$ for all $x, y \geq 0$ together with the asymptotic notation “ O ” (cf. Definition 20), and (4.117) follows using simple calculations.

Thus, (4.117) yields the intended result

$$f(\mathbf{y}^{(k)}) - f(\mathbf{y}^*) = O\left(\frac{1}{k^{c/2p}}\right) + \frac{S\epsilon^2}{\mu_h\sqrt{2\mu_h}} + \frac{S\|\boldsymbol{\lambda}^*\|\epsilon}{\sqrt{2\mu_h}} + \frac{\epsilon^2}{2\mu_h},$$

where $S = \sqrt{(4 + 4 \cos(\pi/m))/\mu}$. □

4.2.4 Feasible Points from Algorithm 5 and Algorithm 6

4.2.4.1 Computation of a Feasible Point and Related Results

We have derived convergences in both dual and primal domains under both cases CASE 1 and CASE 2 in the preceding section. More importantly, the Proposition 4.2.1, Proposition 4.2.2, Proposition 4.2.3, and Proposition 4.2.4 present convergences of sequences $\{\mathbf{y}^{(k)}\}$ and $\{f(\mathbf{y}^{(k)})\}$, the sequences of primal variable iterates and primal function value iterates, respectively. In particular, they characterize how far the locally computed solution $\mathbf{y}^{(k)}$ (*cf.* line 4 of Algorithm 5 and lines 3 of Algorithm 6, respectively) is located from the primal solution \mathbf{y}^* . More specifically, $\mathbf{y}^{(k)}$ is not necessarily feasible to problem (3.2), despite k being very large. That is, $\mathbf{A}\mathbf{y}^{(k)} \neq \mathbf{0}$, no matter how big the iteration index k is, where \mathbf{A} is defined in (4.3) [*cf.* the last equality constraint of problem (3.2)].

However, the computation of a *feasible point* and deriving related convergence results are of utmost importance from both analytical and practical perspectives. Thus, we next provide an exposition for computing a feasible point by using $\mathbf{y}^{(k)}$ s and quantifying the convergences of the related sequences of feasible points.

First, a simple criterion for computing a feasible point by using $\mathbf{y}^{(k)}$ s is presented below.

Remark 16. Let $\tilde{\mathbf{y}}^{(k)}$ be a point in \mathbb{R}^{nm} given by

$$\tilde{\mathbf{y}}^{(k)} = \frac{1}{m} (\mathbf{1}_{m \times m} \otimes \mathbf{I}_n) \mathbf{y}^{(k)}, \quad (4.118)$$

where $\mathbf{y}^{(k)}$ is given in (4.44). Then $\tilde{\mathbf{y}}^{(k)}$ is a feasible point of the problem (3.2).

Proof. It is straightforward to see that $\tilde{\mathbf{y}}^{(k)}$ is computed simply by averaging $\mathbf{y}_i^{(k)}$ s. In particular,

$$\tilde{\mathbf{y}}^{(k)} = [(\tilde{\mathbf{y}}_1^{(k)})^\top \dots (\tilde{\mathbf{y}}_m^{(k)})^\top]^\top, \quad (4.119)$$

where $\tilde{\mathbf{y}}_i^{(k)} = 1/m \sum_{i=1}^m \mathbf{y}_i^{(k)}$ for all $i \in \{1, \dots, m\}$. Thus, the last equality constraint of the problem (3.2) is satisfied with $\tilde{\mathbf{y}}_i^{(k)}$, $i \in \{1, \dots, m\}$. Moreover, the first constraint of the problem (3.2) is also satisfied with $\tilde{\mathbf{y}}_i^{(k)}$ for all $i \in \{1, \dots, m\}$, because it is a convex combination of $\mathbf{y}_1^{(k)}, \dots, \mathbf{y}_{m-1}^{(k)}$, and $\mathbf{y}_m^{(k)}$ with $\mathbf{y}_i^{(k)} \in \mathcal{Y}$, for all $i \in \{1, \dots, m\}$. Thus, $\tilde{\mathbf{y}}^{(k)}$ is a feasible point of the problem (3.2). \square

Next, we present some important results which are useful when deriving convergences of primal feasible points.

Lemma 12. *Let \mathbf{y} be a vector in $\bar{\mathcal{Y}}$ given in (4.1) and $\tilde{\mathbf{y}} = 1/m (\mathbf{1}_{m \times m} \otimes \mathbf{I}_n) \mathbf{y}$ (cf. Definition 12). Then,*

1. $\|\mathbf{y} - \mathbf{y}^*\| \geq \|\tilde{\mathbf{y}} - \mathbf{y}^*\|$.
2. $\|\tilde{\mathbf{y}} - \mathbf{y}^*\| \geq (1/\tilde{D})(f(\tilde{\mathbf{y}}) - f(\mathbf{y}^*))$, if $\tilde{D} < \infty$, where

$$\tilde{D} = \sup_{\substack{\hat{\mathbf{y}} \in \bar{\mathcal{Y}} \\ \mathbf{A}\hat{\mathbf{y}} = \mathbf{0}}} \{\|\boldsymbol{\nu}\| \mid \boldsymbol{\nu} \in \partial f(\hat{\mathbf{y}})\}, \quad (4.120)$$

and \mathbf{A} is defined in (4.3).

Proof. Let $\mathbf{B} = (1/m)(\mathbf{1}_{m \times m} \otimes \mathbf{I}_n)$ for clarity. We note that $\sigma(\mathbf{1}_{m \times m}) = \{0, m\}$ and $\sigma(\mathbf{I}_n) = \{1\}$. Then, it is straightforward to see that

$$\begin{aligned} \sigma(\mathbf{1}_{m \times m} \otimes \mathbf{I}_n) &= \{\lambda\mu \mid \lambda \in \sigma(\mathbf{1}_{m \times m}) \text{ and } \mu \in \sigma(\mathbf{I}_n)\}, \text{ (see Remark 6)} \\ &= \{m\}. \end{aligned} \quad (4.121)$$

Thus we have

$$\|\mathbf{B}\|_2 = \max_{\lambda \in \sigma(\mathbf{B})} |\lambda| \quad (4.122)$$

$$= (1/m) \max_{\lambda \in \sigma(\mathbf{1}_{m \times m} \otimes \mathbf{I}_n)} |\lambda| \quad (4.123)$$

$$= 1, \quad (4.124)$$

where (4.122) follows because \mathbf{B} is symmetric (*cf.* Part 3 of Remark 5), (4.123) follows by simple calculations, and (4.124) is straightforward using (4.121). Then

$$\|\tilde{\mathbf{y}} - \mathbf{y}^*\| = \|\mathbf{B}\mathbf{y} - \mathbf{B}\mathbf{y}^*\| \quad (4.125)$$

$$\leq \|\mathbf{B}\|_2 \|\mathbf{y} - \mathbf{y}^*\| \quad (4.126)$$

$$= \|\mathbf{y} - \mathbf{y}^*\|, \quad (4.127)$$

where (4.125) follows using that $\tilde{\mathbf{y}} = \mathbf{B}\mathbf{y}$ (*cf.* Remark 16) together with that $\mathbf{y}^* = \mathbf{B}\mathbf{y}^*$, because the local solutions \mathbf{y}_i s satisfy the consistency constraint $\mathbf{y}_i = \mathbf{y}_{i+1}$, $i = 1, \dots, m - 1$ in problem (3.2) at the optimal point, and (4.126) follows by part 4 of Remark 5. Finally (4.127) follows because, $\|\mathbf{B}\|_2 = 1$ [*cf.* (4.124)].

The second assertion is essentially based on the convexity of f and (4.120). In particular, we have

$$f(\tilde{\mathbf{y}}) - f(\mathbf{y}^*) \leq \|\tilde{\boldsymbol{\nu}}\| \|\tilde{\mathbf{y}} - \mathbf{y}^*\|, \quad \forall \tilde{\mathbf{y}} \in \mathcal{Y}_{\text{feas}}, \forall \tilde{\boldsymbol{\nu}} \in \partial f(\tilde{\mathbf{y}}) \quad (4.128)$$

$$\leq \tilde{D} \|\tilde{\mathbf{y}} - \mathbf{y}^*\|, \quad \forall \tilde{\mathbf{y}} \in \mathcal{Y}_{\text{feas}}, \quad (4.129)$$

where $\mathcal{Y}_{\text{feas}} = \{\mathbf{y} \in \text{dom } f \mid \mathbf{y} \in \bar{\mathcal{Y}}, \mathbf{A}\mathbf{y} = \mathbf{0}\}$. □

It is worth noting that part 2 of Lemma 12 relies on certain Lipschitzian properties of the primal function f [*cf.* Assumption 4.1.2].

4.2.4.2 Convergence Properties Using Feasible Points Under CASE 1

The convergence properties of primal feasible points $\{\tilde{\mathbf{y}}^{(k)}\}$ under CASE I using the non-summable step size rule are established below. Here we do not provide a separate convergence proof for the constant step size rule as it is a particular case of the nonsummable step size rule (*cf.* Remark 13). Thus, the convergence properties using the constant step size rule $\gamma_k = \gamma$ are also characterized using the following proposition.

Proposition 4.2.5. *Suppose Assumption 1, Assumptions 2, Assumption 3, and Assumption 4.1.1 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6, $\{\mathbf{y}^{(k)}\}$ be the corresponding sequence of primal variables, and $\{\tilde{\mathbf{y}}^{(k)}\}$ be the resulting sequence of primal feasible points, where $\tilde{\mathbf{y}}^{(k)} = 1/m (\mathbf{1}_{m \times m} \otimes \mathbf{I}_n) \mathbf{y}^{(k)}$. Moreover, suppose that the functions f_i , $i = 1, \dots, m$ are differentiable and the distance from $\boldsymbol{\lambda}^{(k)}$ to the dual optimal solution $\boldsymbol{\lambda}^*$, i.e., $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|$ is uniformly bounded by some scalar D . Let γ_k satisfies the nonsummable step size rule given in (4.45) with $0 < \gamma_k \leq 1/L_h$. Then*

1. $\lim_k \min_{i \in \{0, \dots, k\}} \|\tilde{\mathbf{y}}^{(i)} - \mathbf{y}^*\| \leq \sqrt{2D\epsilon/\mu}$.
2. for $\gamma_k = \gamma/(k+1)^p$, where $0 < \gamma \leq 1/L_h$ and $0 \leq p \leq 1$,

$$\min_{i \in \{0, \dots, k\}} \|\tilde{\mathbf{y}}^{(i)} - \mathbf{y}^*\| = \begin{cases} O\left(\frac{1}{\sqrt[4]{k^{1-p}}}\right) + \sqrt{2D\epsilon/\mu} & p \in [0, 1) \\ O\left(\frac{1}{\sqrt[4]{\ln k}}\right) + \sqrt{2D\epsilon/\mu} & p = 1, \end{cases} \quad (4.130)$$

and the best convergence rate is of the order $O(1/\sqrt[4]{k})$, which is achieved when $p = 0$.

Proof. Part 1 of Lemma 12 yields that

$$\min_{i \in \{0, \dots, k\}} \|\tilde{\mathbf{y}}^{(i)} - \mathbf{y}^*\| \leq \min_{i \in \{0, \dots, k\}} \|\mathbf{y}^{(i)} - \mathbf{y}^*\|. \quad (4.131)$$

Thus the part 1 of Proposition 4.2.5 is straightforward using part 1 of Proposition 4.2.2.

The equation (4.131) together with part 3 of Proposition (4.2.2) claims the second part of the proposition. Moreover, using (4.130), it is straightforward to see that the best convergence rate is of order $O(1/\sqrt[4]{k})$, which is achieved when $p = 0$, i.e., the constant step size rule $\gamma_k = \gamma$. \square

It is important to note that the hypotheses of Proposition 4.2.5 do not provide any means for quantifying a bound on the error of primal objective values evaluated at feasi-

ble points $\{\tilde{\mathbf{y}}^{(k)}\}$, i.e., a bound on $\|f(\tilde{\mathbf{y}}^{(k)}) - f(\mathbf{y}^*)\|$. However, a quantification is possible with both strong convexity and gradient Lipschitz continuity properties of f (cf. Assumption 4.1.1 and Assumption 4.1.2), i.e., under CASE 2.

4.2.4.3 Convergence Properties Using Feasible Points Under CASE 2

The convergence properties of primal feasible points $\{\tilde{\mathbf{y}}^{(k)}\}$ and primal objective values $\{f(\tilde{\mathbf{y}}^{(k)})\}$ under CASE 2 using constant step size rule are asserted below.

Proposition 4.2.6. *Suppose Assumption 1, Assumptions 2, Assumption 3, Assumption 4.1.1, and Assumption 4.1.2 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6, $\{\mathbf{y}^{(k)}\}$ be the corresponding sequence of primal variables, and $\{\tilde{\mathbf{y}}^{(k)}\}$ be the resulting sequence of primal feasible points, where $\tilde{\mathbf{y}}^{(k)} = 1/m (\mathbf{1}_{m \times m} \otimes \mathbf{I}_n) \mathbf{y}^{(k)}$. Moreover, suppose that the functions f_i , $i = 1, \dots, m$ are differentiable. Let the step size $\gamma_k = \gamma$ for all $k \in \mathbb{Z}_+^0$. Then for $0 < \gamma \leq 1/L_h$,*

1. $\limsup_k \|\tilde{\mathbf{y}}^{(k)} - \mathbf{y}^*\| \leq \frac{\epsilon}{\sqrt{\mu\mu_h}}$.
2. $\limsup_k \|f(\tilde{\mathbf{y}}^{(i)}) - f(\mathbf{y}^*)\| \leq \frac{\tilde{D}\epsilon}{\sqrt{\mu\mu_h}}$, where \tilde{D} is defined in (4.120).
3. *the least upper bound of $\min_{i \in \{0, \dots, k\}} \|\tilde{\mathbf{y}}^{(i)} - \mathbf{y}^*\|$ converges into a neighbourhood of 0 with the rate of geometric progression.*
4. *the least upper bound of $\min_{i \in \{0, \dots, k\}} \|f(\tilde{\mathbf{y}}^{(i)}) - f(\mathbf{y}^*)\|$ converges into a neighbourhood of 0 with the rate of geometric progression.*

Proof. The proofs of all the parts of the proposition are straightforward using Proposition 4.2.3, combined with Lemma 12. More specifically, Lemma 12 provides upper-bounds for $\|\tilde{\mathbf{y}}^{(k)} - \mathbf{y}^*\|$ and $f(\tilde{\mathbf{y}}^{(k)}) - f(\mathbf{y}^*)$ using $\|\mathbf{y}^{(k)} - \mathbf{y}^*\|$. Thus the proof of the proposition is straightforward using Proposition 4.2.3, which asserts convergence results using primal variables $\mathbf{y}^{(k)}$.

□

Proposition 4.2.7. *Suppose Assumption 1, Assumptions 2, Assumption 3, Assumption 4.1.1, and Assumption 4.1.2 hold. Let $\{\boldsymbol{\lambda}^{(k)}\}$ be the sequence of Lagrange multipliers generated by either Algorithm 5 or Algorithm 6, $\{\mathbf{y}^{(k)}\}$ be the corresponding sequence of primal variables, and $\{\tilde{\mathbf{y}}^{(k)}\}$ be the resulting sequence of primal feasible points, where $\tilde{\mathbf{y}}^{(k)} = 1/m (\mathbf{1}_{m \times m} \otimes \mathbf{I}_n) \mathbf{y}^{(k)}$. Moreover, suppose that the functions f_i , $i = 1, \dots, m$ are differentiable. Let γ_k satisfies the nonsummable step size rule given in (4.45) with $0 < \gamma_k \leq 1/L_h$. Then*

1. $\limsup_k \|\tilde{\mathbf{y}}^{(k)} - \mathbf{y}^*\| \leq \epsilon \sqrt{1/(\mu\mu_h)}$.
2. $\limsup_k \|f(\tilde{\mathbf{y}}^{(k)}) - f(\mathbf{y}^*)\| \leq \tilde{D}\epsilon \sqrt{1/(\mu\mu_h)}$, where \tilde{D} is defined in (4.120).
3. for $\gamma_k = (c/\mu_h)/(k+1)^p$, where $0 < p \leq 1$ and $0 < c \leq \mu_h/L_h$,

$$\|\tilde{\mathbf{y}}^{(k)} - \mathbf{y}^*\| = O\left(\frac{1}{k^{c/2p}}\right) + \frac{\epsilon}{\mu_h}.$$

4. for $\gamma_k = (c/\mu_h)/(k+1)^p$, where $0 < p \leq 1$ and $0 < c \leq \mu_h/L_h$,

$$f(\tilde{\mathbf{y}}^{(k)}) - f(\mathbf{y}^*) = O\left(\frac{1}{k^{c/2p}}\right) + \frac{\tilde{D}\epsilon}{\mu_h}.$$

Proof. The proofs of all the parts of the proposition are straightforward using Proposition 4.2.4, combined with Lemma 12, which provides upperbounds for $\|\tilde{\mathbf{y}}^{(k)} - \mathbf{y}^*\|$ and $f(\tilde{\mathbf{y}}^{(k)}) - f(\mathbf{y}^*)$ using $\|\mathbf{y}^{(k)} - \mathbf{y}^*\|$. \square

4.3 Convergence Analysis: General Consensus

In Chapter 3, we have introduced the main problem [cf. problem (3.1)] that is considered in our study. In particular, the problem was known as the global consensus problem, as all the local functions f_i s depend on the same global variable \mathbf{z} . However, the global consensus problem (3.1) can easily be extended to a more generalized setting, where the local functions f_i s depend only on a part of the variable \mathbf{z} . Thus, the focus of this chapter is

to introduce the generalized problem formulation, and discuss how all our derivations and results presented in Chapter 4.1 and Chapter 4.2 can be extended under this generalized setting.

4.3.1 Generalized Problem Formulation

We consider a network consisting of m subsystems, where $m \in \mathbb{Z}_+$ with associated functions f_i s, $i \in \{1, \dots, m\}$. Unlike the global consensus problem (3.1) (cf. Chapter 3.1), f_i s of the general case need not depend on the whole vector $\mathbf{z} \in \mathbb{R}^n$. Instead, we consider that f_i s depend on different parts of the variable \mathbf{z} . Without loss of generality, this can be modeled by partitioning \mathbf{z} into q subvectors, \mathbf{z}_j , $j \in \mathcal{Q} = \{1, \dots, q\}$, each of which can be an argument of f_i s. In particular, we refer to partitions as *nets* and \mathbf{z}_j as the *net variable* associated with net $j \in \mathcal{Q}$ (cf. Figure 4.2). We consider that associated with j th net, there is a set $\mathcal{Y}_j \subseteq \mathbb{R}^{n_j}$ such that $\mathbf{z}_j \in \mathcal{Y}_j$. Thus, the generalized problem formulation is

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^m f_i(\mathbf{E}_i^T \mathbf{z}) \\ & \text{subject to} && \mathbf{z} \in \mathcal{Y}_1 \times \dots \times \mathcal{Y}_q, \end{aligned} \tag{4.132}$$

where each f_i , $i \in \{1, \dots, m\}$ is strictly convex, and the matrix \mathbf{E}_i encodes the selection of net variables of subsystem i . We refer the problem (4.132) as the *general consensus problem*.

We made the following assumption on each \mathcal{Y}_j , $j \in \mathcal{Q}$ and f_i , $i \in \{1, \dots, m\}$ (cf. Assumption 1)

Assumption 4.3.1 (Closedness). *The sets \mathcal{Y}_j s and the functions f_i s are closed.*

To equivalently transform the problem (4.132) into a form of (3.2) (especially to obtain a distributed solution method), we need to introduce local versions of the net variables \mathbf{z}_j . To this end, first we let m_j denote the number of subsystems whose objective function depends on \mathbf{z}_j and $\mathcal{M}_j = \{1, \dots, m_j\}$, $j \in \mathcal{Q}$. Moreover, for notational convenience, let us enumerate the local versions of the net variables as \mathbf{y}_{kj} , where $j \in \mathcal{Q}$ and $k \in$

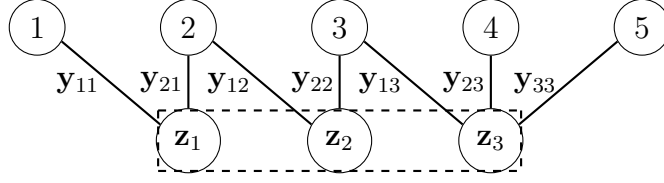


Figure 4.2: Decomposition structure: There are five subsystems and three nets (i.e., $q = 3$) with the public variable $\mathbf{z} = [\mathbf{z}_1^T \ \mathbf{z}_2^T \ \mathbf{z}_3^T]^T$. Net variables are \mathbf{z}_1 , \mathbf{z}_2 , and \mathbf{z}_3 . Functions associated with subsystems are $f_1(\mathbf{z}_1)$, $f_2(\mathbf{z}_1, \mathbf{z}_2)$, $f_3(\mathbf{z}_2, \mathbf{z}_3)$, $f_4(\mathbf{z}_3)$, and $f_5(\mathbf{z}_3)$. The sets $\mathcal{M}_1 = \mathcal{M}_2 = \{1, 2\}$, and $\mathcal{M}_3 = \{1, 2, 3\}$.

\mathcal{M}_j (cf. Figure 4.2). Finally, to associate \mathbf{y}_{kj} s to respective subsystems, we denote by $[\mathbf{y}]_i$ the vector of local versions of the net variables owned by the subsystem i , where $\mathbf{y} = [\mathbf{y}_{11}^T \ \dots \ \mathbf{y}_{m_q q}^T]^T$. For example, $[\mathbf{y}]_1 = \mathbf{y}_{11}$, $[\mathbf{y}]_2 = [\mathbf{y}_{21}^T \ \mathbf{y}_{12}^T]^T$, $[\mathbf{y}]_3 = [\mathbf{y}_{22}^T \ \mathbf{y}_{13}^T]^T$, $[\mathbf{y}]_4 = \mathbf{y}_{23}$, and $[\mathbf{y}]_5 = \mathbf{y}_{33}$ in Figure 4.2. Thus, the problem is equivalently reformulated as

$$\begin{aligned}
& \text{minimize} && f_{\text{gen}}(\mathbf{y}) = \sum_{i=1}^m f_i([\mathbf{y}]_i) \\
& \text{subject to} && \mathbf{y}_{kj} \in \mathcal{Y}_j, \quad j \in \mathcal{Q}, k \in \mathcal{M}_j \\
& && \mathbf{y}_{kj} = \mathbf{y}_{(k+1)j}, \quad j \in \mathcal{Q}, k \in \mathcal{M}_j \setminus \{m_j\},
\end{aligned} \tag{4.133}$$

where the variable is $\mathbf{y} \in \mathbb{R}^{\sum_{j=1}^q n_j m_j}$. The equality constraint $\mathbf{y}_{kj} = \mathbf{y}_{(k+1)j}$, $k \in \mathcal{M}_j \setminus \{m_j\}$ ensures the consistency of the local variables associated with j th net. It is not difficult to verify that the problem (4.133) decouples among subsystems, leading to distributed algorithms.

The equality constraint of problem (4.133) is equivalent to $\mathbf{A}_j \mathbf{y}_j = \mathbf{0}$, where $\mathbf{y}_j = [\mathbf{y}_{1j}^T \ \dots \ \mathbf{y}_{m_j j}^T]^T$ and $\mathbf{A}_j \in \mathbb{R}^{n_j(m_j-1) \times n_j m_j}$, a matrix similar to (4.3) with the structure given by:

$$\mathbf{A}_j = \begin{bmatrix} \mathbf{I}_{n_j} & -\mathbf{I}_{n_j} & \mathbf{0} & \dots & \dots & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{n_j} & -\mathbf{I}_{n_j} & \mathbf{0} & \dots & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{n_j} & -\mathbf{I}_{n_j} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \vdots & \ddots & \mathbf{0} & \mathbf{I}_{n_j} & -\mathbf{I}_{n_j} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{I}_{n_j} & -\mathbf{I}_{n_j} \end{bmatrix}, \tag{4.134}$$

where $j \in \mathcal{Q}$. Thus, the dual function $g_{\text{gen}} : \mathbb{R}^{\sum_{j=1}^q n_j(m_j-1)} \rightarrow \overline{\mathbb{R}}$ associated with the

problem (4.133) is given by

$$g_{\text{gen}}(\boldsymbol{\lambda}) = \inf_{\mathbf{y}_j \in \mathcal{Y}_j^{m_j}, j \in \mathcal{Q}} \left[\sum_{i=1}^m f_i([\mathbf{y}]_i) + \sum_{j=1}^q \boldsymbol{\lambda}_j^T \mathbf{A}_j \mathbf{y}_j \right], \quad (4.135)$$

where ³ $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_1^T \dots \boldsymbol{\lambda}_q^T]^T$ and $\boldsymbol{\lambda}_j \in \mathbb{R}^{n_j(m_j-1)}$, $j \in \mathcal{Q}$. Let's consider an example in the general setting for clarity.

Example 3. We consider a network consisting of five subsystems with three nets. That is, $q = 3$ and $\mathcal{Q} = \{1, 2, 3\}$ (cf Figure 4.2). Then we discuss the decomposition structure associated with this network.

Suppose that the subsystems 1 and 2 are associated with net 1, Subsystems 2 and 3 are associated with net 2, and subsystems 3, 4, and 5 are associated with net 3. Thus we have $\mathcal{M}_1 = \mathcal{M}_2 = \{1, 2\}$, and $\mathcal{M}_3 = \{1, 2, 3\}$. Consequently, we have that the local versions of the net variables owned by subsystems are $[\mathbf{y}]_1 = \mathbf{y}_{11}$, $[\mathbf{y}]_2 = [\mathbf{y}_{21}^T \ \mathbf{y}_{12}^T]^T$, $[\mathbf{y}]_3 = [\mathbf{y}_{22}^T \ \mathbf{y}_{13}^T]^T$, $[\mathbf{y}]_4 = \mathbf{y}_{23}$, and $[\mathbf{y}]_5 = \mathbf{y}_{33}$ (cf. Figure 4.2). Suppose that the sets associated with nets are $\mathcal{Y}_1 = \mathcal{Y}_2 = \mathcal{Y}_3 = \mathbb{R}$, the set of real numbers, for clarity. Then the distributed problem associated with our network is given by [cf. the distributed problem (4.133)]

$$\begin{aligned} & \text{minimize} && f_{\text{gen}}(\mathbf{y}) = \sum_{i=1}^5 f_i([\mathbf{y}]_i) \\ & \text{subject to} && \mathbf{y}_{11} \in \mathbb{R}, \mathbf{y}_{21} \in \mathbb{R}, \mathbf{y}_{12} \in \mathbb{R}, \mathbf{y}_{22} \in \mathbb{R}, \mathbf{y}_{13} \in \mathbb{R}, \mathbf{y}_{23} \in \mathbb{R}, \mathbf{y}_{33} \in \mathbb{R} \\ & && \mathbf{y}_{11} = \mathbf{y}_{21} \\ & && \mathbf{y}_{12} = \mathbf{y}_{22} \\ & && \mathbf{y}_{13} = \mathbf{y}_{23} \\ & && \mathbf{y}_{23} = \mathbf{y}_{33}, \end{aligned} \quad (4.136)$$

where $\mathbf{y} = [\mathbf{y}_{11} \ \mathbf{y}_{21} \ \mathbf{y}_{12} \ \mathbf{y}_{22} \ \mathbf{y}_{13} \ \mathbf{y}_{23} \ \mathbf{y}_{33}]^T$.

It is worth noting that the first equality constraint of (4.136) is associated with net 1

³We use the same notation $\boldsymbol{\lambda}$ as in (3.3) for notational simplicity.

and is equivalent to $\mathbf{A}_1 \mathbf{y}_1 = \mathbf{0}$, where

$$\mathbf{y}_1 = [\mathbf{y}_{11} \ \mathbf{y}_{21}]^T, \quad \text{and} \quad \mathbf{A}_1 = [1 \ -1]. \quad (4.137)$$

Next, the second equality constraint of (4.136) is associated with net 2 and is equivalent to $\mathbf{A}_2 \mathbf{y}_2 = \mathbf{0}$, where

$$\mathbf{y}_2 = [\mathbf{y}_{12} \ \mathbf{y}_{22}]^T, \quad \text{and} \quad \mathbf{A}_2 = [1 \ -1]. \quad (4.138)$$

Finally, the third and fourth equality constraints of (4.136) are associated with net 3 and are equivalent to $\mathbf{A}_3 \mathbf{y}_3 = \mathbf{0}$, where

$$\mathbf{y}_3 = [\mathbf{y}_{13} \ \mathbf{y}_{23} \ \mathbf{y}_{33}]^T, \quad \text{and} \quad \mathbf{A}_3 = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}. \quad (4.139)$$

Then the dual function $g_{\text{gen}} : \mathbb{R}^4 \rightarrow \overline{\mathbb{R}}$ associated with the generalized problem (4.136) is given by [cf. the dual problem (4.135)]

$$\begin{aligned} g_{\text{gen}}(\boldsymbol{\lambda}) &= \inf_{\mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^2, \mathbf{y}_3 \in \mathbb{R}^3} \left[\sum_{i=1}^5 f_i([\mathbf{y}]_i) + \sum_{j=1}^3 \boldsymbol{\lambda}_j^T \mathbf{A}_j \mathbf{y}_j \right] \\ &= \inf_{\mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^2, \mathbf{y}_3 \in \mathbb{R}^3} \left[f_1(\mathbf{y}_{11}) + f_2(\mathbf{y}_{21}, \mathbf{y}_{12}) + f_3(\mathbf{y}_{22}, \mathbf{y}_{13}) + f_4(\mathbf{y}_{23}) + f_5(\mathbf{y}_{33}) \right. \\ &\quad \left. + \boldsymbol{\lambda}_{11}(\mathbf{y}_{11} - \mathbf{y}_{21}) + \boldsymbol{\lambda}_{12}(\mathbf{y}_{12} - \mathbf{y}_{22}) + \boldsymbol{\lambda}_{13}(\mathbf{y}_{13} - \mathbf{y}_{23}) \right. \\ &\quad \left. + \boldsymbol{\lambda}_{23}(\mathbf{y}_{23} - \mathbf{y}_{33}) \right] \end{aligned} \quad (4.140)$$

$$\begin{aligned} &= \left[\inf_{\mathbf{y}_{11} \in \mathbb{R}} (f_1(\mathbf{y}_{11}) + \boldsymbol{\lambda}_{11} \mathbf{y}_{11}) \right] + \left[\inf_{\mathbf{y}_{21}, \mathbf{y}_{12} \in \mathbb{R}} (f_2(\mathbf{y}_{21}, \mathbf{y}_{12}) + \boldsymbol{\lambda}_{12} \mathbf{y}_{12} - \boldsymbol{\lambda}_{11} \mathbf{y}_{21}) \right] \\ &\quad + \left[\inf_{\mathbf{y}_{22}, \mathbf{y}_{13} \in \mathbb{R}} (f_3(\mathbf{y}_{22}, \mathbf{y}_{13}) + \boldsymbol{\lambda}_{13} \mathbf{y}_{13} - \boldsymbol{\lambda}_{12} \mathbf{y}_{22}) \right] \\ &\quad + \left[\inf_{\mathbf{y}_{23} \in \mathbb{R}} (f_4(\mathbf{y}_{23}) + (\boldsymbol{\lambda}_{23} - \boldsymbol{\lambda}_{13}) \mathbf{y}_{23}) \right] + \left[\inf_{\mathbf{y}_{33} \in \mathbb{R}} (f_5(\mathbf{y}_{33}) - \boldsymbol{\lambda}_{23} \mathbf{y}_{33}) \right] \end{aligned} \quad (4.141)$$

where $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_{11} \ \boldsymbol{\lambda}_{12} \ \boldsymbol{\lambda}_{13} \ \boldsymbol{\lambda}_{23}]^T$ and $\boldsymbol{\lambda}_{11}$, $\boldsymbol{\lambda}_{12}$, $\boldsymbol{\lambda}_{13}$, and $\boldsymbol{\lambda}_{23}$ are Lagrange multipliers as-

sociated with the equality constraints $\mathbf{y}_{11} = \mathbf{y}_{21}$, $\mathbf{y}_{12} = \mathbf{y}_{22}$, $\mathbf{y}_{13} = \mathbf{y}_{23}$, and $\mathbf{y}_{23} = \mathbf{y}_{33}$ respectively. Here (4.140) follows using $[\mathbf{y}]_i$ s, the local versions of the net variables owned by the subsystems, and using related \mathbf{A}_j and \mathbf{y}_j for each net j (cf. (4.137), (4.137), and (4.137)). The last equality (4.138) follows because, for fixed $\boldsymbol{\lambda}$, the infimization can be performed in parallel by each subsystem. Thus, the related subproblems that can be solved locally by subsystems are given by

$$\text{Subproblem 1 : } \left[\inf_{\mathbf{y}_{11} \in \mathbb{R}} (f_1(\mathbf{y}_{11}) + \boldsymbol{\lambda}_{11} \mathbf{y}_{11}) \right] \quad (4.142)$$

$$\text{Subproblem 2 : } \left[\inf_{\mathbf{y}_{21}, \mathbf{y}_{12} \in \mathbb{R}} (f_2(\mathbf{y}_{21}, \mathbf{y}_{12}) + \boldsymbol{\lambda}_{12} \mathbf{y}_{12} - \boldsymbol{\lambda}_{11} \mathbf{y}_{21}) \right] \quad (4.143)$$

$$\text{Subproblem 3 : } \left[\inf_{\mathbf{y}_{22}, \mathbf{y}_{13} \in \mathbb{R}} (f_3(\mathbf{y}_{22}, \mathbf{y}_{13}) + \boldsymbol{\lambda}_{13} \mathbf{y}_{13} - \boldsymbol{\lambda}_{12} \mathbf{y}_{22}) \right] \quad (4.144)$$

$$\text{Subproblem 4 : } \left[\inf_{\mathbf{y}_{23} \in \mathbb{R}} (f_4(\mathbf{y}_{23}) + (\boldsymbol{\lambda}_{23} - \boldsymbol{\lambda}_{13}) \mathbf{y}_{23}) \right] \quad (4.145)$$

$$\text{Subproblem 5 : } \left[\inf_{\mathbf{y}_{33} \in \mathbb{R}} (f_5(\mathbf{y}_{33}) - \boldsymbol{\lambda}_{23} \mathbf{y}_{33}) \right] \quad (4.146)$$

The related dual problem is given by

$$\text{maximize}_{\boldsymbol{\lambda} \in \mathbb{R}^4} g_{\text{gen}}(\boldsymbol{\lambda}). \quad (4.147)$$

Then, the dual problem (4.147) can be solved either in a partially or fully distributed manner using related distributed subgradient algorithms (cf. Algorithm 5 and Algorithm 6).

4.3.2 Generalized Results

In this section, we will present how all the theoretical derivations established in this study using the global consensus problem (3.1) (cf. Chapter 4.1 and Chapter 4.2) can be generalized using the general consensus problem (4.132). Specifically, we will rely on some useful assumptions, which are in turn used to generalize the results.

Let $\mathbf{r}_{l_j}^{(k)}$ be the distortion associated with $\mathbf{y}_{l_j}^{(k)}$, where $j \in \mathcal{Q}$, $l \in \mathcal{M}_j$, and $k \in \mathbb{Z}_+^0$

during subproblem coordination, cf. (3.11), (For example, see the subproblems associated with Example 3, cf. (4.142), (4.143), (4.144), (4.145), and (4.146)). Then, we made the following assumption on $\mathbf{r}_{lj}^{(k)}$, where $j \in \mathcal{Q}$, $l \in \mathcal{M}_j$, and $k \in \mathbb{Z}_+^0$.

Assumption 4.3.2 (Absolute Deterministic Distortion). *The distortion $\mathbf{r}_{lj}^{(k)}$ associated with $\mathbf{y}_{lj}^{(k)}$ is bounded by ε_{lj} , where $j \in \mathcal{Q}$, $l \in \mathcal{M}_j$, and $k \in \mathbb{Z}_+^0$, i.e.,*

$$\|\mathbf{r}_{lj}^{(k)}\| \leq \varepsilon_{lj}, \quad j \in \mathcal{Q}, \quad l \in \mathcal{M}_j, \quad k \in \mathbb{Z}_+^0. \quad (4.148)$$

Next, we made the following two hypotheses that is satisfied by most standard local objective functions considered in the literature. However, we note that these two assumptions are explicitly invoked only when necessary.

Assumption 4.3.3 (Strongly Convex local Objectives at Subsystems). *The local objective functions f_i s in problem (4.133) are strongly convex with constant $\mu_i^{\text{gen}} > 0$, $i = 1, \dots, m$.*

Assumption 4.3.4 (Gradient Lipschitz Continuous local objectives at subsystems). *The sets \mathcal{Y}_j , $j \in \mathcal{Q}$ in problem (4.133) equal \mathbb{R}^n . Moreover, f_i s are differentiable and the gradients ∇f_i s are Lipschitz continuous with constant $L_i^{\text{gen}} > 0$, $i = 1, \dots, m$.*

Finally, we rely on the strong duality assumption.

Assumption 4.3.5 (Strong Duality). *The optimal values p_{gen}^* and d_{gen}^* of the problems (4.133) and (3.8), respectively, are attained. Moreover, strong duality between (4.133) and (3.8) holds, i.e.,*

$$p_{\text{gen}}^* = f(\mathbf{y}_{\text{gen}}^*) = g_{\text{gen}}(\boldsymbol{\lambda}^*) = d_{\text{gen}}^*, \quad (4.149)$$

for some $\mathbf{y}_{\text{gen}}^* \in \{\mathbf{y}_{\text{gen}} \in \mathbb{R}^{nm} \mid \forall i \mathbf{y}_i \in \mathcal{Y}, \mathbf{A}\mathbf{y} = \mathbf{0}\}$ and for some $\boldsymbol{\lambda}^* \in \mathbb{R}^{n(m-1)}$, where \mathbf{A} is defined in (4.3).

Then, with the key changes pointed out in the following results, all the remaining theoretical assertions presented in Section 4.1.4 and Chapter 4.2 can be generalized in a straightforward manner.

First, we highlight the relationship between the dual function g_{gen} [cf. (4.135)] and the conjugate function f_{gen}^* [cf. Lemma 3 corresponding to the problem (3.2)] of $f_{\text{gen}} + \delta_{\bar{\mathcal{Y}}_{\text{gen}}}$, where $\bar{\mathcal{Y}}_{\text{gen}}$ denotes the Cartesian product of $\mathcal{Y}_j^{m_j}$ s, $j \in \mathcal{Q}$, i.e.,

$$\bar{\mathcal{Y}}_{\text{gen}} = \mathcal{Y}_1^{m_1} \times \dots \times \mathcal{Y}_q^{m_q},$$

where $\mathcal{Y}_j^{m_j}$, $j \in \mathcal{Q}$ denotes the m_j -fold Cartesian product of \mathcal{Y} .

Corollary 5 (cf. Lemma 3). *Let $\mathbf{A}_{\text{gen}} = \text{diag}(\mathbf{A}_1, \dots, \mathbf{A}_q)$. i.e.,*

$$\mathbf{A}_{\text{gen}} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_q \end{bmatrix}, \quad (4.150)$$

where \mathbf{A}_j , $j \in \mathcal{Q}$ has the similar form given in (4.134). Then $g_{\text{gen}}(\boldsymbol{\lambda}) = -f_{\text{gen}}^*(\mathbf{A}_{\text{gen}}^T \boldsymbol{\lambda})$, where $f_{\text{gen}}^* : \mathbb{R}^{\sum_{j=1}^q n_j m_j} \rightarrow \bar{\mathbb{R}}$ denotes the conjugate function of $f_{\text{gen}} + \delta_{\bar{\mathcal{Y}}_{\text{gen}}}$.

Proof. The equality constraint $\mathbf{A}_j \mathbf{y}_j = \mathbf{0}$, $j \in \mathcal{Q}$ of the problem (4.133) is equivalent to $\mathbf{A}_{\text{gen}} \mathbf{Y} = \mathbf{0}$, where $\mathbf{Y} = [\mathbf{y}_1^T \dots \mathbf{y}_q^T]^T$ and $\mathbf{y}_j = [\mathbf{y}_{1j}^T \dots \mathbf{y}_{m_j j}^T]^T$ for all $j \in \mathcal{Q}$. Hence, the result $g_{\text{gen}}(\boldsymbol{\lambda}) = -f^*(\mathbf{A}_{\text{gen}} \boldsymbol{\lambda})$ is straightforward by using the similar approach used in the proof of Lemma 3. \square

Corollary 6 (cf. Proposition 1). *Suppose Assumption 4.3.1 and Assumption 4.3.3 hold. Then the dual function g_{gen} is differentiable and ∇g_{gen} is Lipschitz continuous with constant $(1/\mu_{\text{gen}}) \max_{j \in \mathcal{Q}} \lambda_{\max}(\mathbf{A}_j \mathbf{A}_j^T)$, where μ_{gen} represents the strong convexity constant of f_{gen} in (4.133) (cf Lemma 4).*

Proof. The proof of Corollary 6 is similar to that presented in the proof of Proposition 1. In particular, replace g and \mathbf{A} in the proof of Proposition 1 [cf. Chapter 4.1.2] with g_{gen} and \mathbf{A}_{gen} , respectively. Then it immediately follows that ∇g_{gen} is Lipschitz continuous with the constant $(1/\mu_{\text{gen}}) \lambda_{\max}(\mathbf{A}_{\text{gen}} \mathbf{A}_{\text{gen}}^T)$ [cf. (4.18)]. Here $\mathbf{A}_{\text{gen}} \mathbf{A}_{\text{gen}}^T \in$

$\mathbb{R}^{\sum_{j=1}^q n_j(m_j-1) \times \sum_{j=1}^q n_j(m_j-1)}$ is a block diagonal matrix with the structure

$$\mathbf{A}_{\text{gen}} \mathbf{A}_{\text{gen}}^T = \begin{bmatrix} \mathbf{A}_1 \mathbf{A}_1^T & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 \mathbf{A}_2^T & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_q \mathbf{A}_q^T \end{bmatrix}, \quad (4.151)$$

where $\mathbf{A}_j \mathbf{A}_j^T \in \mathbb{R}^{n_j(m_j-1) \times n_j(m_j-1)}$ for all $j \in \mathcal{Q}$. Thus, the result yields because $\lambda_{\max}(\mathbf{A}_{\text{gen}} \mathbf{A}_{\text{gen}}^T) = \max_{j \in \mathcal{Q}} \lambda_{\max}(\mathbf{A}_j \mathbf{A}_j^T)$. \square

Corollary 7 (cf. Proposition 2). *Let Assumption 4.3.4 holds. Then the function $-g_{\text{gen}}$ is strongly convex with constant $(1/L_{\text{gen}}) \min_{j \in \mathcal{Q}} \lambda_{\min}(\mathbf{A}_j \mathbf{A}_j^T)$, where L_{gen} represents the gradient Lipschitz continuous constant of f_{gen} in (4.133) (cf Lemma 5).*

Proof. Replace g and \mathbf{A} in the proof of Proposition 2 [cf. Chapter 4.1.3] with g_{gen} [cf. (4.135)] and \mathbf{A}_{gen} [cf. (4.150)], respectively. Then we have $-g_{\text{gen}}$ is strongly convex with the constant $(1/L_{\text{gen}}) \lambda_{\min}(\mathbf{A}_{\text{gen}} \mathbf{A}_{\text{gen}}^T)$ [cf. (4.31)], where $\mathbf{A}_{\text{gen}} \mathbf{A}_{\text{gen}}^T$ has the same structure as given in (4.151). Thus, $\lambda_{\min}(\mathbf{A}_{\text{gen}} \mathbf{A}_{\text{gen}}^T) = \min_{j \in \mathcal{Q}} \lambda_{\min}(\mathbf{A}_j \mathbf{A}_j^T)$ and the result holds. \square

4.4 Numerical Results

In this section, we test empirically the theoretical assertions presented in this study. To this end, problem (3.2) is considered with quadratic f_i s, i.e.,

$$f_i(\mathbf{y}_i) = \mathbf{y}_i^T \mathbf{A}_i \mathbf{y}_i + \mathbf{q}_i^T \mathbf{y}_i, \quad \mathbf{A}_i \in \mathbb{S}_{++}^n, \quad \mathbf{q}_i \in \mathbb{R}^n, \quad (4.152)$$

where \mathbf{A}_i and \mathbf{q}_i are arbitrarily chosen. Let l_i and ν_i denote the minimum and maximum eigenvalues of the matrix A_i respectively, where $i \in \{1, \dots, m\}$. Since A_i s are positive definite, each f_i is strongly convex with constant l_i , $i \in \{1, \dots, m\}$. Thus, Assumptions 4.1.1 holds. Moreover, Assumption 3 and closedness of f_i s [cf. Assumption 1] also hold throughout the rest of the chapter. More specifically, we consider a system consist-

ing of five subsystems (i.e., $m = 5$) with scalar valued local variables y_i (i.e., $n = 1$). In particular, simulation results demonstrate convergence properties of both cases CASE 1 and CASE 2 (cf. Chapter 4.2).

4.4.1 CASE 1: Strongly Convex Local Objectives at Subsystems

We consider that the constraint set $\mathcal{Y} = \{\mathbf{u} \in \mathbb{R}^n \mid -a\mathbf{1}_{n \times 1} \leq \mathbf{u} \leq a\mathbf{1}_{n \times 1}\}$, where $a \in \mathbb{R}_+$ (cf. Problem (3.2)). Then \mathcal{Y} is a box of width $2a$ per-dimension. Thus, \mathcal{Y} is not only closed [cf. Assumption 1], but also compact. Note that the CASE 1 represents a scenario where the dual function g is with Lipschitz continuous gradients (cf. Proposition 1). We consider that the distorted vector $\hat{\mathbf{d}}^{(k)}$ [cf. (3.10)] is a consequence of a naive quantization scheme implemented in step 4 of Algorithm 6. In particular, the box \mathcal{Y} is partitioned into identical mini-boxes of width $t = 2a/2^b$ per-dimension, where $b \in \mathbb{Z}_+$ represents the number of bits transmitted per-dimension (note that one side of the box \mathcal{Y} is then split in to 2^b parts). The indexing of the mini-boxes is common to all subsystems. In step 4 of the Algorithm 6, the subsystem i first chooses $\hat{\mathbf{y}}_i^{(k)}$ to be the centroid of the mini-box in which $\mathbf{y}_i^{(k)}$ lies (see Figure 4.3 for clarity). Then the index of the chosen mini-box is transmitted, which is simply an nb -bit word. As a result, the norm of the distortion $\mathbf{r}_i^{(k)}$ associated with $\mathbf{y}_i^{(k)}$ is bounded as remarked below, conforming to Assumption 2.

Remark 17. *the distortion $\mathbf{r}_i^{(k)}$ of $\mathbf{y}_i^{(k)}$ is bounded, i.e., $\|\mathbf{r}_i^{(k)}\| \leq \varepsilon_i = \sqrt{n} a/2^b$, where $i \in \{1, \dots, m\}$.*

Proof. We have that the distortion $\mathbf{r}_i^{(k)} = \hat{\mathbf{y}}_i^{(k)} - \mathbf{y}_i^{(k)}$ [cf. (3.11)]. Then

$$\|\mathbf{r}_i^{(k)}\| \leq \sqrt{\sum_{i=1}^n \left(\frac{t}{2}\right)^2} \quad (4.153)$$

$$= \frac{t}{2} \sqrt{n}, \quad (4.154)$$

$$= \frac{\sqrt{n} a}{2^b}, \quad (4.155)$$

where (4.153) follows because the distortion $\mathbf{r}_i^{(k)}$ does not exceed half of a diagonal of

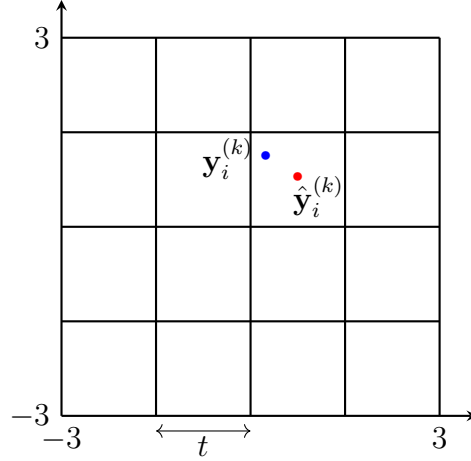


Figure 4.3: The quantization scheme in CASE 1 with $n = 2$ and $b = 2$. The constraint set \mathcal{Y} is a box of width 6 per dimension. The box is partitioned into identical 16 mini-boxes of width $t = 1.5$. The exact solution $\mathbf{y}_i^{(k)}$ of subsystem i is given in blue. The distorted vector $\hat{\mathbf{y}}_i^{(k)}$ which is given in red, is chosen to be the centroid of the respective mini-box.

a mini-box of width t , (4.154) is straightforward using simple calculation, and (4.155) follows using $t = 2a/2^b$, the width of a mini-box. \square

Next, the absolute deterministic distortion $\|\mathbf{r}^{(k)}\|$ of the subgradient $\mathbf{d}^{(k)}$, where $\mathbf{r}^{(k)} = \hat{\mathbf{d}}^{(k)} - \mathbf{d}^{(k)}$ [cf. (4.46)] is bounded as stated below.

Remark 18. The norm of the total error vector $\mathbf{r}^{(k)} = \hat{\mathbf{d}}^{(k)} - \mathbf{d}^{(k)}$ of the subgradient $\mathbf{d}^{(k)}$ is bounded, i.e., $\|\mathbf{r}^{(k)}\| \leq \epsilon = a\sqrt{n(m-1)}/2^{b-1}$ (cf. Corollary 2).

Proof. Using equation (4.46) (cf. Remark 14 of Chapter 4.2 and (4.47)) we have

$$\begin{aligned} \|\mathbf{r}^{(k)}\| &\leq \sqrt{\sum_{i=1}^{m-1} (\varepsilon_i + \varepsilon_{i+1})^2} \\ &= \sqrt{\sum_{i=1}^{m-1} \left(\frac{2\sqrt{na}}{2^b}\right)^2} \end{aligned} \quad (4.156)$$

$$= \frac{a\sqrt{n(m-1)}}{2^{b-1}}, \quad (4.157)$$

where (4.156) follows using Remark 17 and (4.157) follows by simple calculation. \square

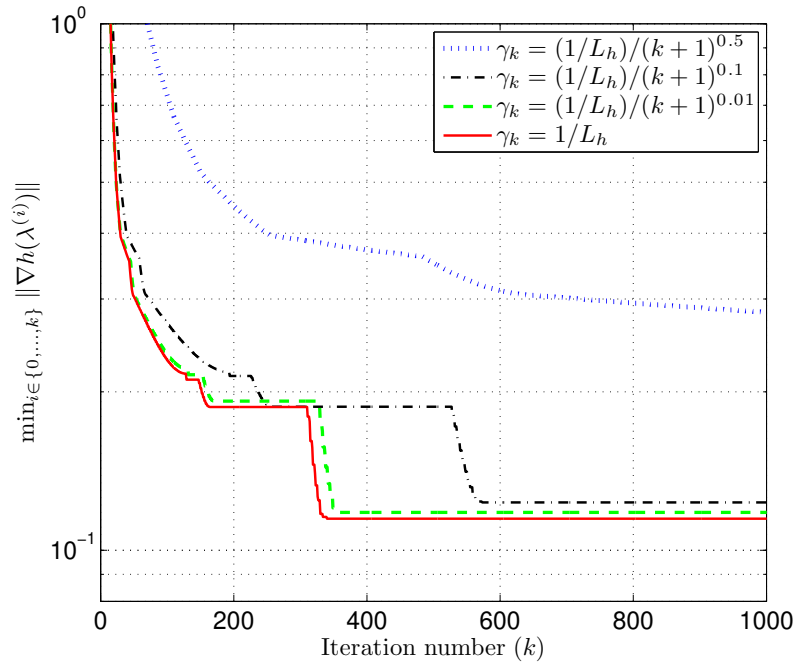


Figure 4.4: CASE 1: Convergence of minimal norm gradients of the negative dual function h . The figure shows the effect of choice of p in the step size $\gamma_k = (1/L_h)/(k+1)^p$ on the convergence, by fixing $b = 5$.

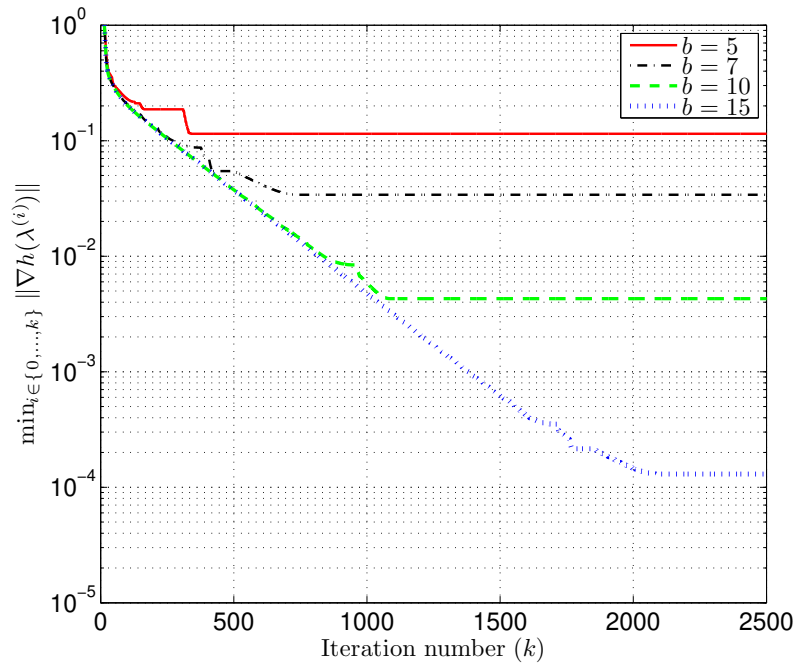


Figure 4.5: CASE 1: Convergence of minimal norm gradients of the negative dual function h . The figure shows the effect of choice of b on the convergence, by fixing $\gamma_k = 1/L_h$.

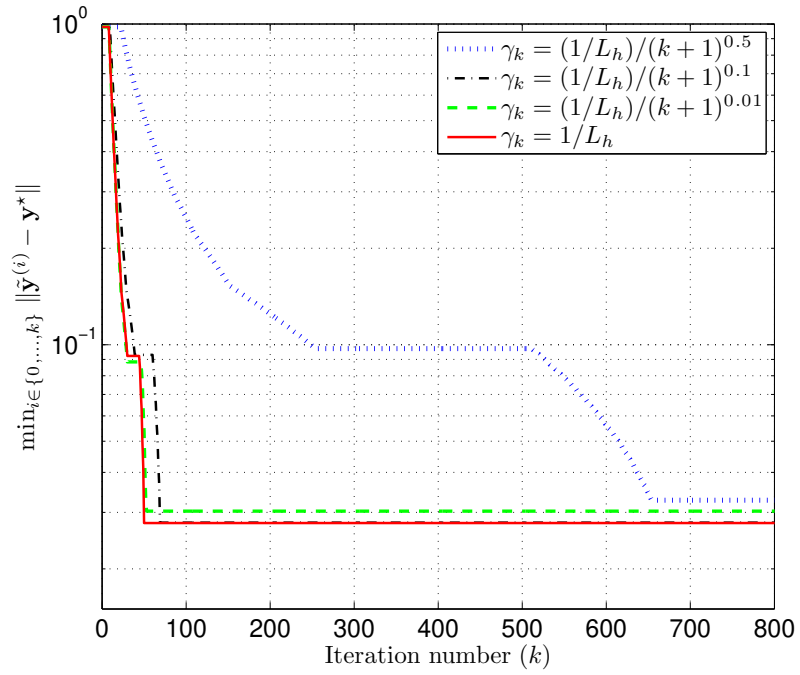


Figure 4.6: CASE 1: Convergence of minimal norm primal feasible points of the problem (3.2). The figure shows the effect of the choice of p in the step size $\gamma_k = (1/L_h)/(k+1)^p$ on the convergence by fixing $b = 5$.

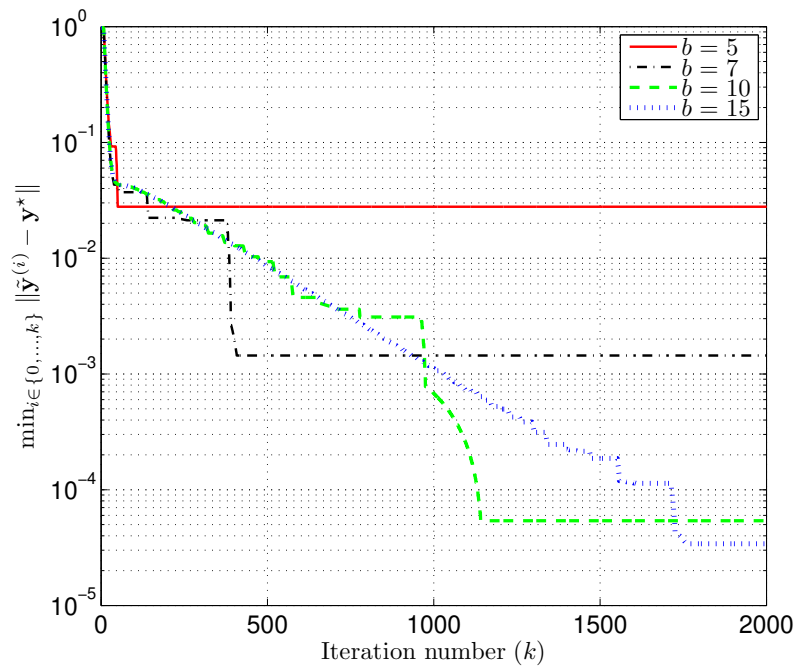


Figure 4.7: CASE 1: Convergence of minimal norm primal feasible points of the problem (3.2). The figure shows the effect of the choice of b on the convergence by fixing $\gamma_k = 1/L_h$.

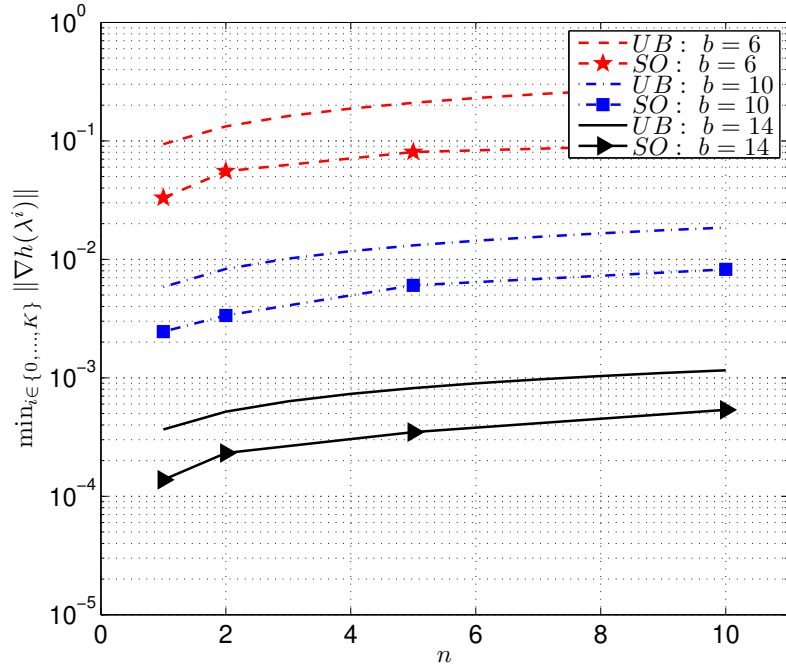


Figure 4.8: CASE 1: The effect of dimension n of the local variables y_i s on SO using fixed step size rule $\gamma_k = 1/L_h$ for different bits.

We illustrate the convergence properties of CASE 1 using \mathcal{Y} with $a = 3$. Figure 4.4 and Figure 4.5 show the convergence of minimal norm gradients of the negative dual function h , i.e., the convergence results established in Corollary 2 (note that the Corollary 1 directly reduces to Corollary 2 when $p = 0$). In particular, Figure 4.4 depicts the effect of the choice of p in the step size $\gamma_k = (1/L_h)/(k+1)^p$ by fixing $b = 5$. Results show that the smaller the value of p , the higher the rate of convergence, as claimed in Corollary 2-(2). Moreover, the figure demonstrates that the best rate is achieved when $p = 0$ which corresponds to the fixed step size rule. Figure 4.5 shows the effect of the choice of b on the convergence of minimal norm gradients of h , by fixing $\gamma_k = 1/L_h$ (i.e., the fixed step size rule). Results show that when the number of bits b increases the size of the neighborhood around 0 to which $\min_i \|\nabla h(\lambda^{(i)})\|$ converges decreases. This is readily expected from Corollary 2-(1), together with Remark 18, because ϵ , the neighborhood, is inversely proportional to 2^b .

Figure 4.6 and Figure 4.7 show the convergence of corresponding primal feasible points, the related results derived in Proposition 4.2.5). Results show similar behavior

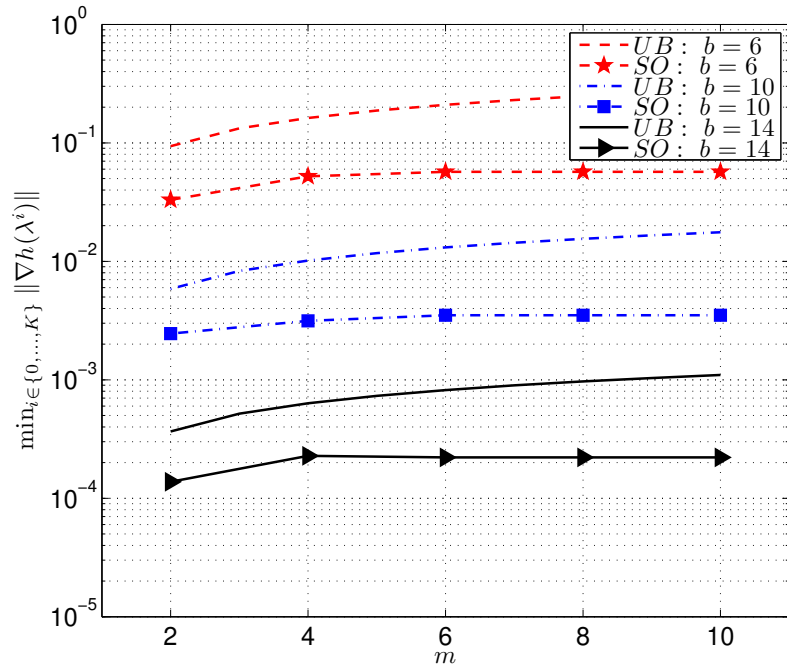


Figure 4.9: CASE 1: The effect of number of users m on SO using fixed step size rule $\gamma_k = 1/L_h$ for different bits.

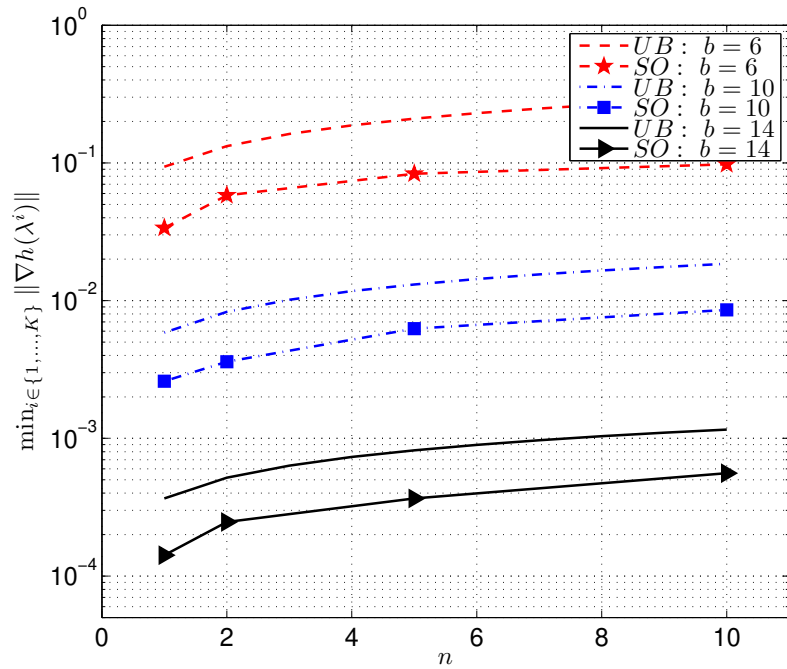


Figure 4.10: CASE 1: The effect of dimension n of the local variables y_i s on SO using nonsummable step size rule $\gamma_k = 1/(L_h k^{0.1})$ for different bits.

as that of Figures 4.4 and 4.5 with respect to the rate of convergence and the size of the converging neighborhood respectively. Figure 4.6 shows that the number of iterations re-

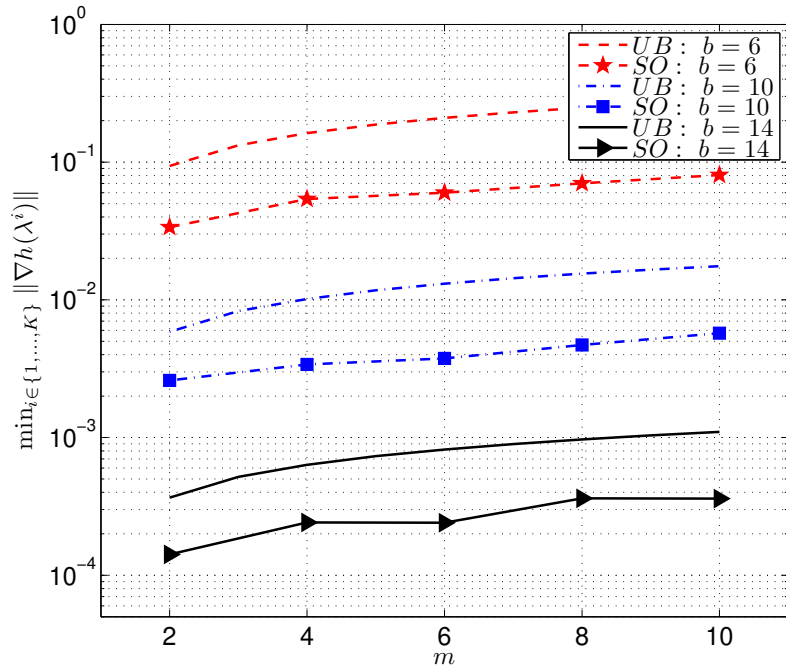


Figure 4.11: CASE 1: The effect of number of users m on SO using nonsummable step size rule $\gamma_k = 1/(L_h k^{0.1})$ for different bits.

quired for reaching the neighborhood in the primal domain appears to be relatively smaller than that in the dual domain, especially for smaller p values. This behavior is typical for many methods in general because a good feasible point can usually be computed, even with a relatively smart heuristic method. This means even though a heuristic method need not find an optimal point, for many instances, it does quickly find a feasible point that is not too far from the optimal point.

Figure 4.8 and the Figure 4.9 depict the effect of dimension n of the local variables y_i s, $i \in \{1, \dots, m\}$ and the number of users m on the suboptimality (SO), respectively, using a fixed step size rule $\gamma_k = 1/L_h$ for different bits ($b = 6, 10,$ and 14). The suboptimality is measured in terms of $\min_{i \in \{0, \dots, K\}} \|\nabla h(\boldsymbol{\lambda}^{(i)})\|$, where K is the iteration index at the algorithm termination. The Figure 4.8 demonstrates the convergence results using two users, i.e., $m = 2$. The effect of the number of users m on SO presented in Figure 4.9 is illustrated using $n = 1$. Each curve in both figures demonstrates averaged values obtained for SO using 100 variations of the problem (3.2). Variations are obtained by randomizing the matrix \mathbf{A}_i and the vector \mathbf{q}_i in (4.152). Figures exhibit that SO in-

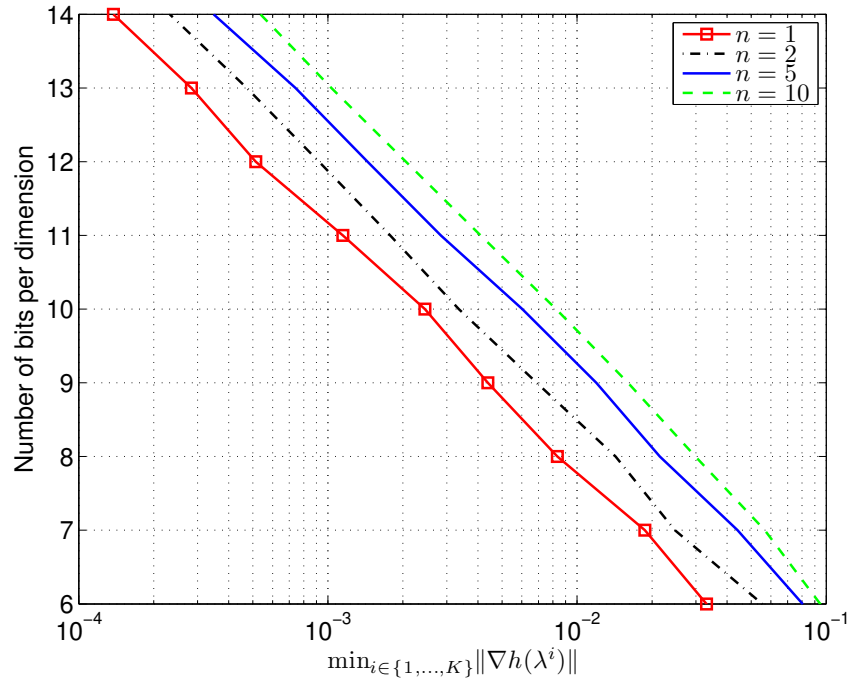


Figure 4.12: CASE 1: The trade-offs between b and SO for different dimensions n using fixed step size rule $\gamma_k = 1/L_h$.

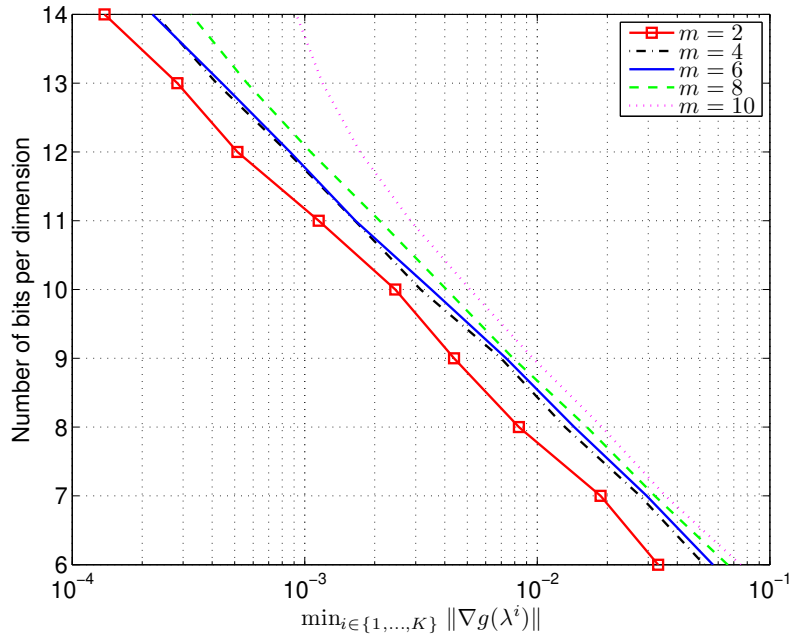


Figure 4.13: CASE 1: The trade-offs between b and SO for different users m using fixed step size rule $\gamma_k = 1/L_h$.

creases as n and m increase. This is expected as claimed in Lemma 8 because, the upper bound ϵ on convergence, [cf. (4.55)] directly depends on n and m [See Remark 18]. Moreover, the upper bound ϵ for each curve is demonstrated within the graphs (curves without

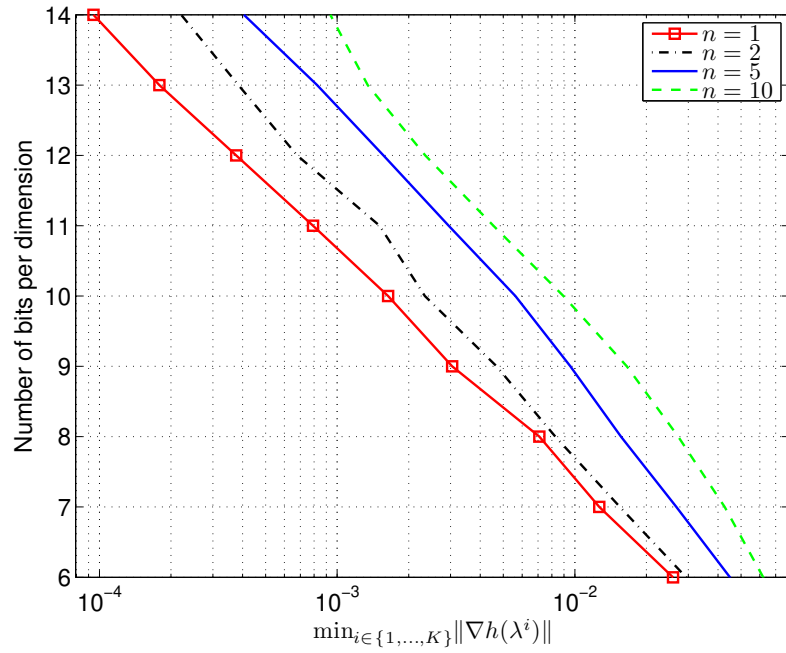


Figure 4.14: CASE 1: The trade-offs between b and SO for different dimensions n using nonsummable step size rule $\gamma_k = \gamma_0/k$, where γ_0 is chosen suitably.

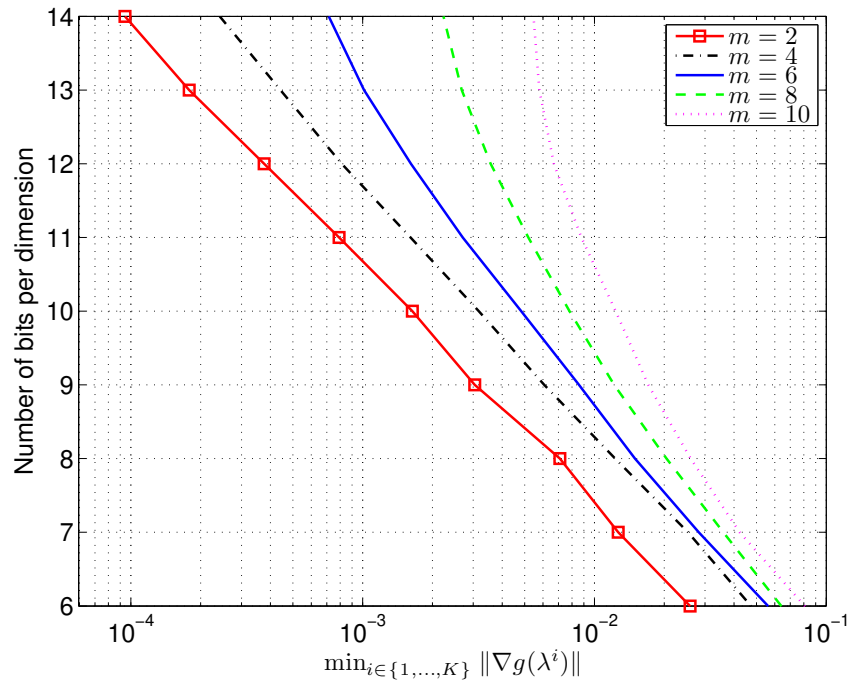


Figure 4.15: CASE 1: The trade-offs between b and SO for different users m using nonsummable step size rule $\gamma_k = \gamma_0/k$, where γ_0 is chosen suitably.

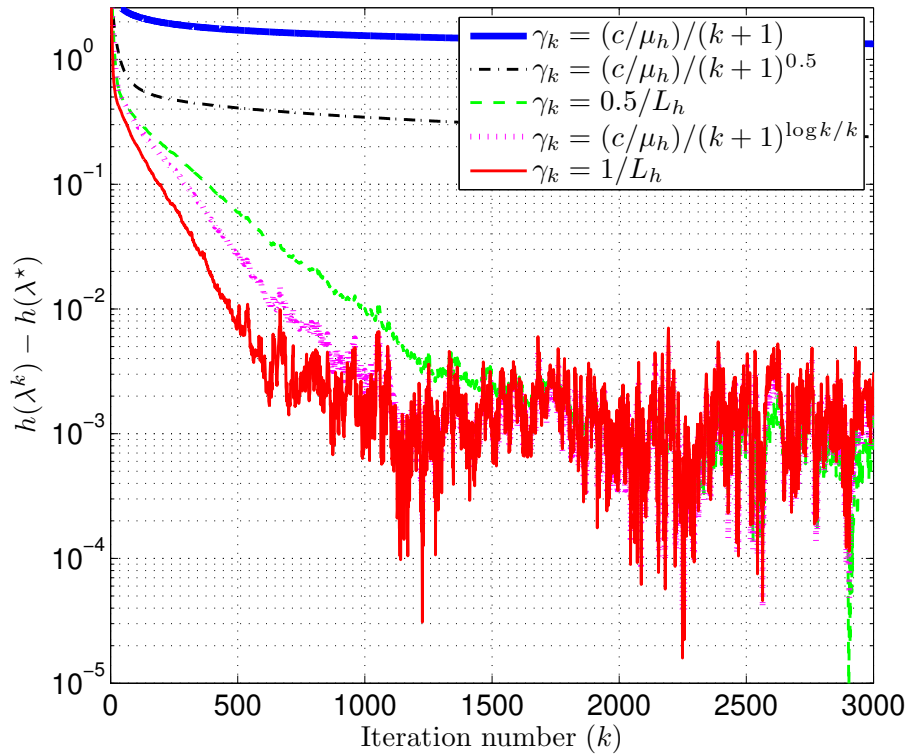


Figure 4.16: CASE 2: Convergence of dual function iterates using constant and non-summable step sizes.

marker symbols). Figures 4.10 and 4.11 show the corresponding convergences using the nonsummable step size rule $\gamma_k = 1/(L_h k^{0.1})$. The figures show similar behaviors as that obtained in Figures 4.8 and 4.9.

Figure 4.12 and Figure 4.13 show trade-offs between b (bits per-dimension) and SO for different dimensions n and different users m , respectively, using a fixed step size rule $\gamma_k = 1/L_h$. Figure 4.12 shows the convergence results using $m = 2$ for different n , and Figure 4.13 depicts the related results using $n = 1$ for different m . In particular, each curve in both figures demonstrates averaged values obtained for SO using 100 variations of the problem (3.2). The figures show that for fixed b , SO increases as n or m increases. Figures 4.14 and 4.15 show the corresponding convergences using the nonsummable step size rule $\gamma_k = \gamma_0/k$, where $\gamma_0 \in \mathbb{R}_+$ is chosen suitably. The figures show similar behaviors as that obtained in Figures 4.12 and 4.13.

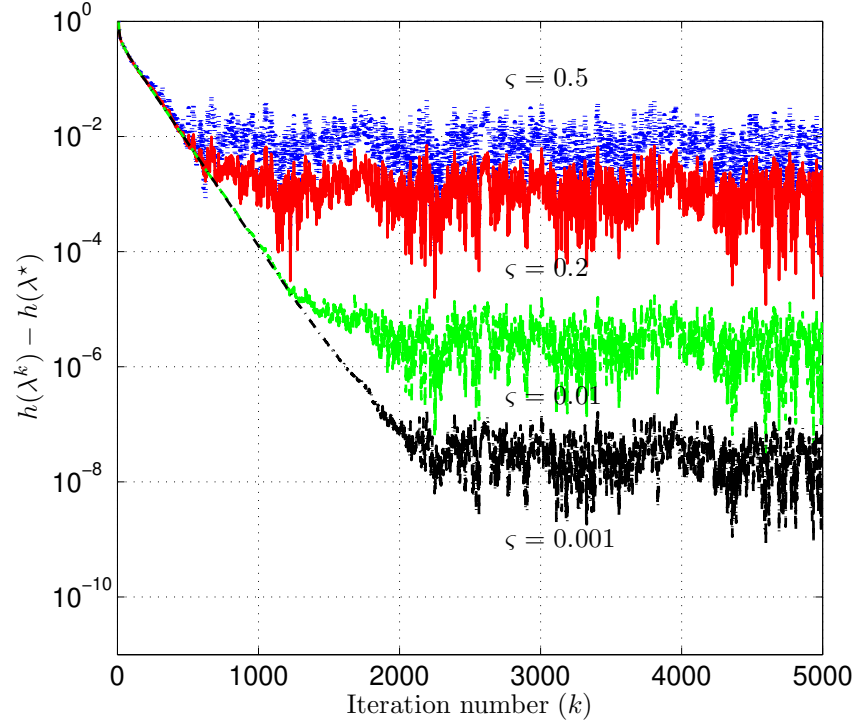


Figure 4.17: CASE 2: Effect of choice of ς on the convergence of dual function iterates.

4.4.2 CASE 2: Strongly convex and gradient Lipschitz Continuous local objectives at subsystems

Let $\mathcal{Y} = \mathbb{R}^n$, i.e., \mathcal{Y} is closed [cf. Assumption 1]. In this setting, each f_i is strongly convex with constant l_i (cf. Assumption 4.1.1) and with a Lipschitz continuous gradient with constant ν_i (cf. Assumption 4.1.2), $i \in \{1, \dots, m\}$ (since \mathbf{A}_i s are positive definite). Thus, the dual function g is with a Lipschitz continuous gradient and is strongly concave (cf. Proposition 1 and Proposition 2).

We consider that the distorted vector $\hat{\mathbf{d}}^{(k)}$ [cf. (3.10)] is a consequence of measurement errors at CN in step 6 of Algorithm 5. The magnitudes of measurement errors are bounded from above by some $\varsigma > 0$ per dimension. As a result, the distortion $\mathbf{r}_i^{(k)}$ of $\mathbf{y}_i^{(k)}$ is bounded s.t., $\|\mathbf{r}_i^{(k)}\| \leq \varepsilon_i = \sqrt{n} \varsigma$, conforming to Assumption 2. Then the absolute deterministic distortion $\|\mathbf{r}^{(k)}\|$ of the subgradient $\mathbf{d}^{(k)}$ [cf. (4.46)] is bounded as follows.

Remark 19. The norm of the total error vector $\mathbf{r}^{(k)} = \hat{\mathbf{d}}^{(k)} - \mathbf{d}^{(k)}$ of the subgradient $\mathbf{d}^{(k)}$ is bounded, i.e., $\|\mathbf{r}^{(k)}\| \leq \epsilon = 2\sqrt{n(m-1)} \varsigma$, cf. Corollary 3 and Corollary 4.

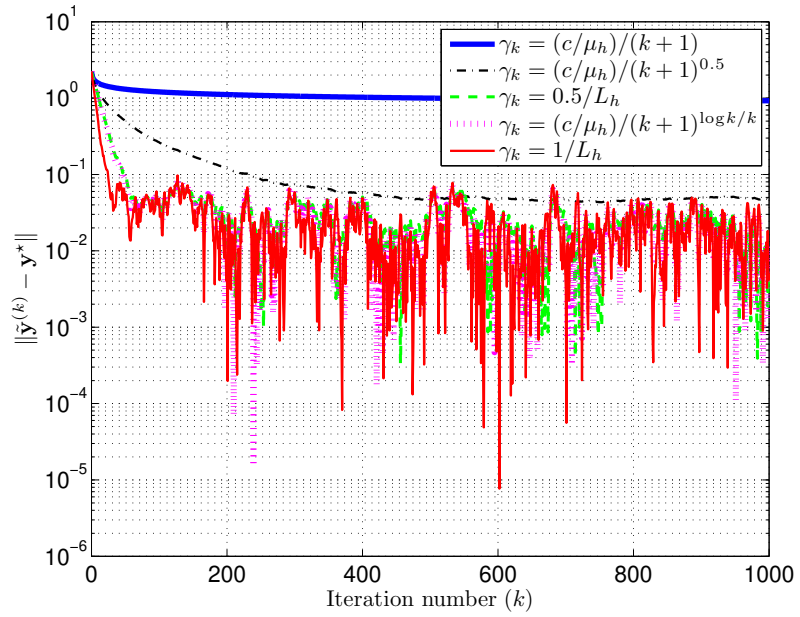


Figure 4.18: CASE 2: Convergence of primal feasible points using constant and non-summable step sizes.

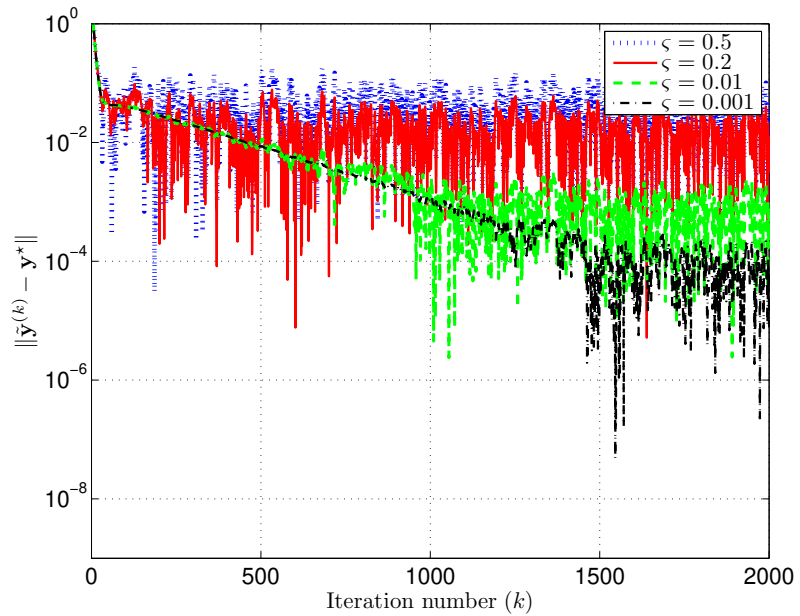


Figure 4.19: CASE 2: Effect of choice of ζ on the convergence of primal feasible points.

Proof. The proof of Remark 19 is straightforward using equation (4.47) of Remark 14. □

Figure 4.16 shows the convergence of dual function iterates for both fixed step size rule (i.e., Corollary 3) and for nonsummable step size rule (i.e., Corollary 4). In particular,

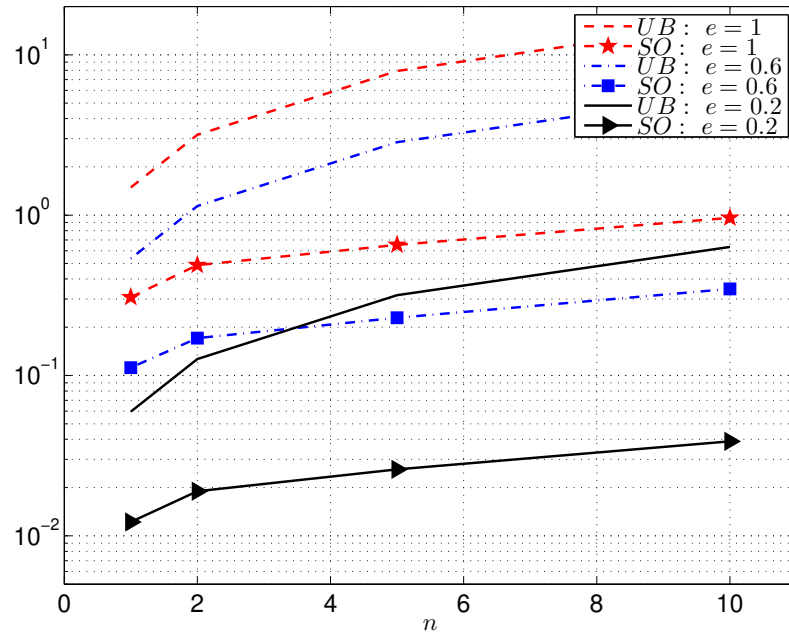


Figure 4.20: CASE 2: The effect of dimension n of the local variables y_i s on SO using fixed step size rule $\gamma_k = 1/L_h$ for different ς .

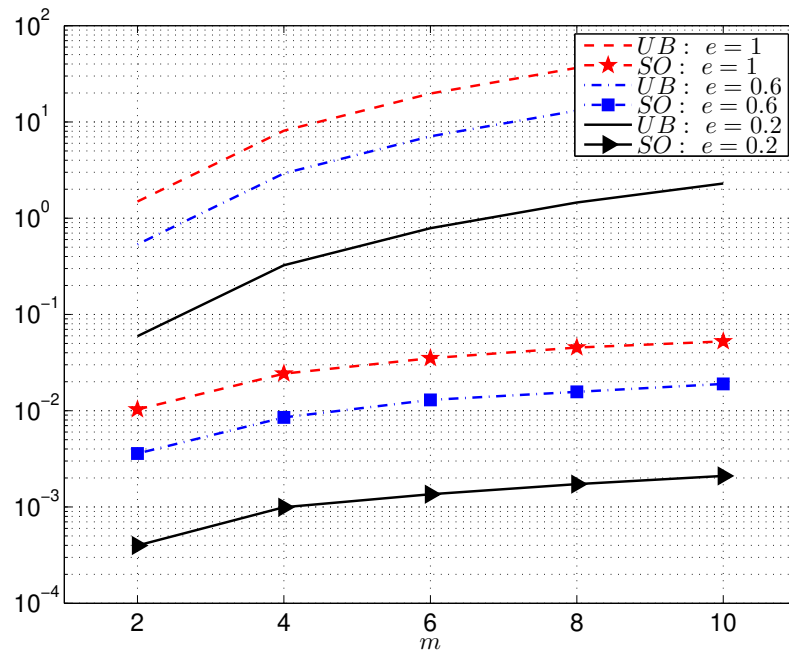


Figure 4.21: CASE 2: The effect of number of users m on SO using fixed step size rule $\gamma_k = 0.01$ for different ς .

the figure shows that linear convergence is guaranteed with fixed step sizes, while $\gamma_k = 1/L_h$ is the best choice. This clearly agrees with the assertions claimed in Corollary 3 [cf. (4.82)]. Moreover, results demonstrate the effect of the choice of p in the step size

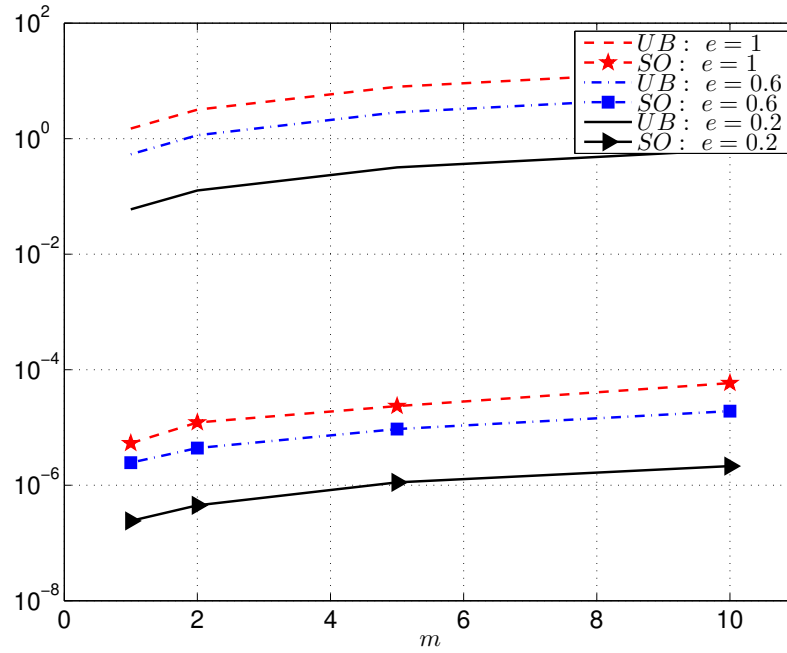


Figure 4.22: CASE 2: The effect of dimension n of the local variables y_i s on SO using nonsummable step size rule $\gamma_k = \gamma_0/k$ for different ς .

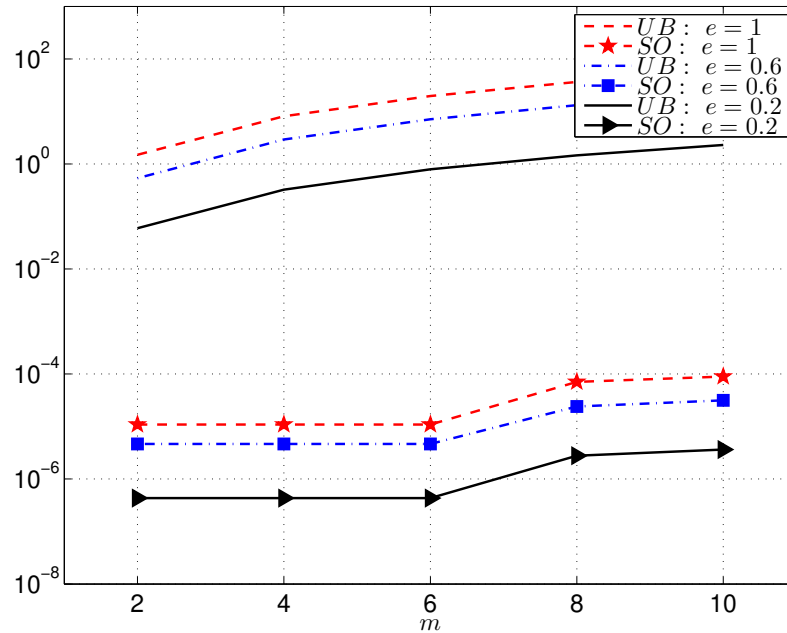


Figure 4.23: CASE 2: The effect of number of users m on SO using nonsummable step size rule $\gamma_k = \gamma_0/k$ for different ς .

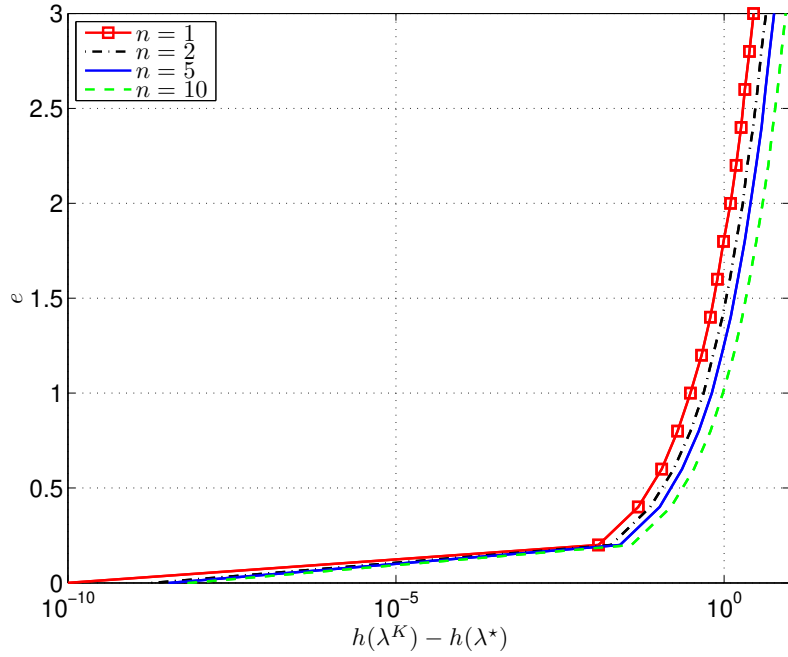


Figure 4.24: CASE 2: The effect of dimension n of the local variables y_i s on SO using fixed step size rule $\gamma_k = 1/L_h$ for different ς .

$\gamma_k = (c/\mu_h)/(k+1)^p$ by fixing $\varsigma = 0.2$ and $c = 0.004$. Note that c is carefully chosen so that it lies inside the prescribed limits $0 < c \leq \mu_h/L_h$ imposed by Corollary 4. Results show that the smaller the value of p , the higher the rate of convergence, as claimed in Corollary 4-(2). For comparisons, we have also included the convergence of dual function values for $\gamma_k = (c/\mu_h)/(k+1)^p$ with $p = \log k/k$ which can be interpreted as a limiting case of $\gamma_k = (c/\mu_h)/(k+1)^p$ as $p \rightarrow 0$ (cf. Remark 11). Results clearly demonstrate a linear convergence as claimed in Remark 11.

Figure 4.17 shows the effect of the choice of ς by fixing $\gamma_k = 1/L_h$. Results show that when ς decreases, so is the size of the neighborhood around $h(\lambda^*)$ to which $h(\lambda^{(k)})$ converges. This behavior is expected from Corollary 3, together with Remark 19, because ϵ , the neighborhood, is linearly related to ς .

Figure 4.18 and Figure 4.19 show the convergence of corresponding primal feasible points, i.e. the related results presented in Proposition 4.2.6 and Proposition 4.2.7. Both figures demonstrate a similar behavior as that of Figure 4.16 and Figure 4.17 with respect to the rate of convergence and the size of the converging neighborhood, respectively.

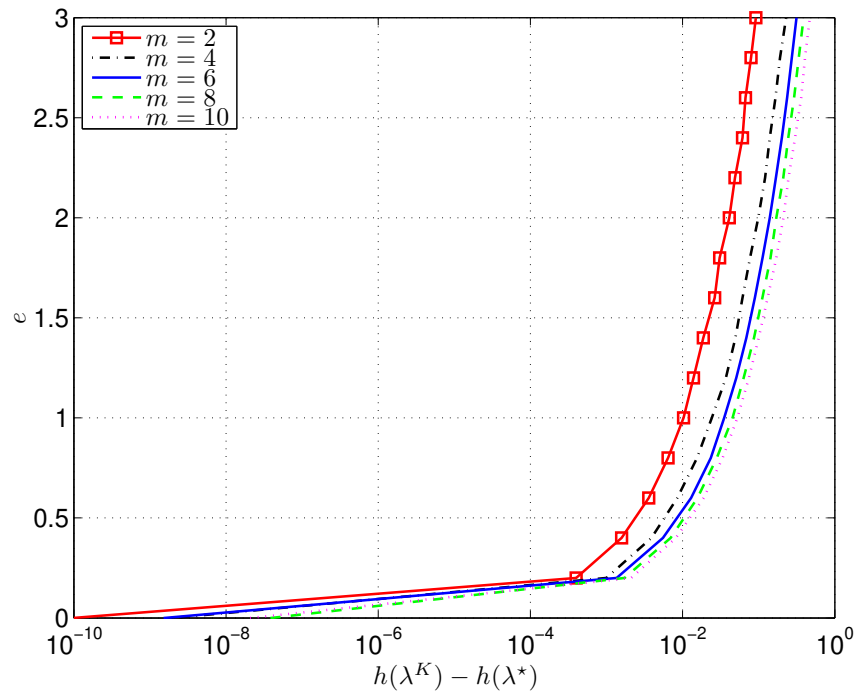


Figure 4.25: CASE 2: The effect of number of users m on SO using fixed step size rule $\gamma_k = 0.01$ for different bits.

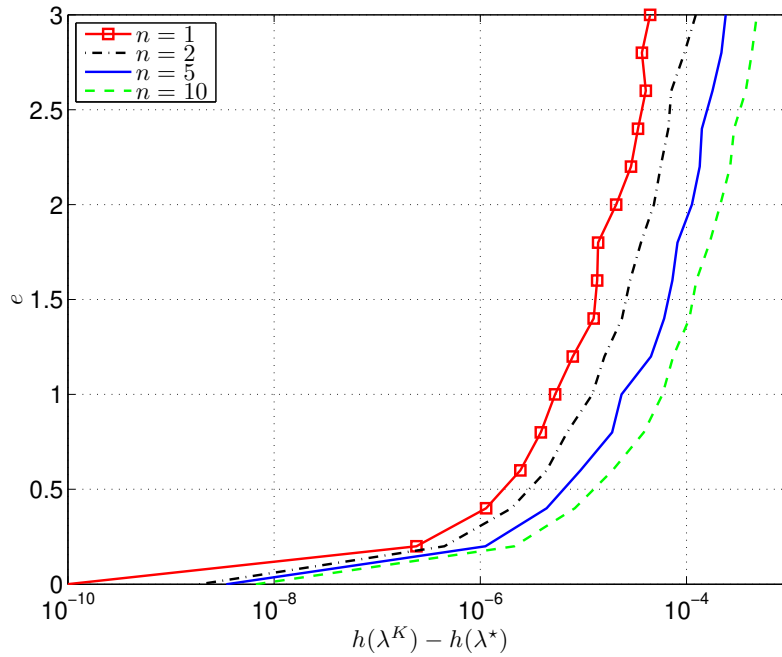


Figure 4.26: CASE 2: The effect of dimension n of the local variables y_i s on SO using nonsummable step size rule $\gamma_k = \gamma_0/k$ for different ς .

Figure 4.20 and Figure 4.21 depict the effect of dimension n of the local variables y_i s, $i \in \{1, \dots, m\}$ and the number of users m on SO, respectively, for different mea-

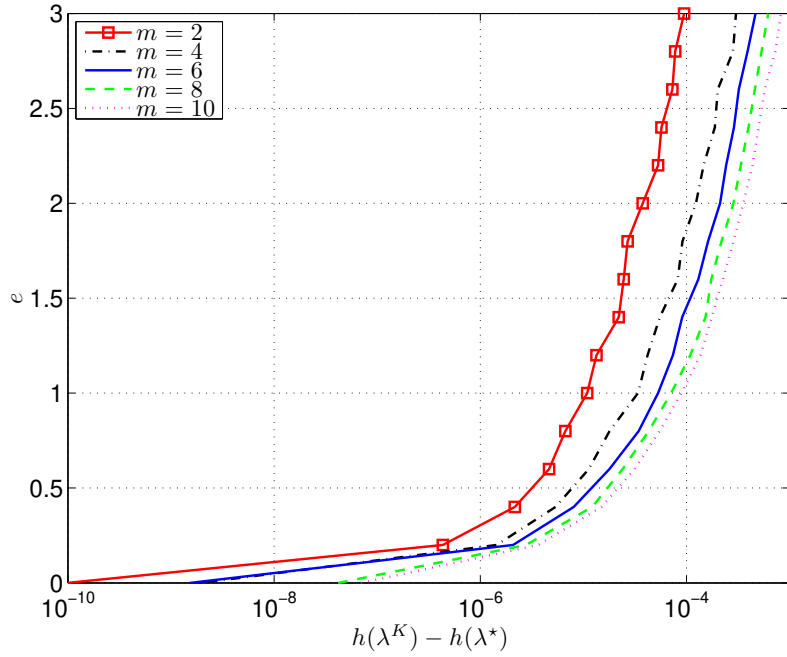


Figure 4.27: CASE 2: The effect of number of users m on SO using nonsummable step size rule $\gamma_k = \gamma_0/k$ for different ς .

surement errors ς ($\varsigma = 0.2, 0.6, \text{ and } 1$). The suboptimality is measured in terms of $\|h(\boldsymbol{\lambda}^{(K)}) - h(\boldsymbol{\lambda}^*)\|$, where K is the iteration index at the algorithm termination. Figure 4.20 demonstrates the convergence results using two users, i.e., $m = 2$ with fixed step size rule $\gamma_k = 1/L_h$. The effect of the number of users m on SO is presented in Figure 4.21 and is illustrated using $n = 1$ with fixed step size $\gamma_k = 0.01$. The figures exhibit that SO increases as n and m increase, similarly to the results that we obtained in CASE 1 (*cf.* Figure 4.8 and Figure 4.9). Moreover, the upper bound $\epsilon^2/2\mu_h$ [*cf.* Corollary 3, equation (4.82)] for each curve is demonstrated within the graphs (curves without marker symbols). Figures 4.22 and 4.23 show the corresponding convergences using the nonsummable step size rule $\gamma_k = \gamma_0/k$, where $\gamma_0 \in \mathbb{R}_+$ is chosen suitably. The figures show similar results as those obtained in Figures 4.20 and 4.21.

Figure 4.24 and Figure 4.25 show trade-offs between ς (the measurement error per dimension) and SO for different dimensions n and different users m , respectively. The Figure 4.24 shows the convergence results using $m = 2$ for different n with $\gamma_k = 1/L_h$, and Figure 4.25 depicts the related results using $n = 1$ for different m with $\gamma_k = 0.01$.

The figures show that for fixed ς , SO increases as n or m increases. Figures 4.26 and 4.27 show the corresponding convergences using the nonsummable step size rule $\gamma_k = \gamma_0/k$, where $\gamma_0 \in \mathbb{R}_+$ is chosen suitably. The figures show similar behaviors as that obtained in Figures 4.24 and 4.25.

Chapter 5

Conclusions and Recommendations

5.1 Conclusion

In this thesis, we have studied the inexactness of dual decomposition methods for solving global variable consensus optimization problems that are commonly used in many types of large-scale signal processing and machine learning application domains. Moreover, we have provided a systematic exposition on state-of-the-art distributed optimization methods that cope with large-scale distributed problems. In general, the currently existing state-of-the-art distributed methods are the subgradient methods, Alternating Direction Method of Multipliers (ADMM), proximal gradient method, and dual averaging. In particular, two commonly used distributed methods that are based on the subgradient methods are the dual decomposition methods and approaches coalescing consensus algorithms with subgradient methods. In this thesis, we primarily use dual decomposition with subgradient methods to establish our convergence results. The decomposition methods are the general approaches to solving optimization problems in a distributed manner. Decomposition methods are interesting approaches to solving optimization problems by breaking them up into smaller subproblems and solving each of them separately. Those subproblems are solved by using an appropriate optimization method such as the subgradient method.

We have developed two distributed algorithms, a partially distributed algorithm and a fully distributed algorithm that can be deployed over numerous non-ideal settings. More importantly, our proposed algorithms can model errors in many large-scale optimization problems, including quantization errors, approximation errors, errors due to subproblem solver accuracy, noise induced in wireless settings, and measurement errors, among others, as long as they are additive and bounded. The convergence properties of proposed

algorithms were extensively analyzed in both dual and primal domains. More specifically, convergences of dual variable iterates, primal variable iterates, and primal feasible iterates are theoretically substantiated along with their rates of convergences. All the convergence results are established under two main cases, where the CASE 1 is representing a scenario that the dual function associated with the primal problem is with Lipschitz continuous gradients, and the CASE 2 is representing a scenario that the negative dual function is both strongly convex and with Lipschitz continuous gradients. Moreover, all the theoretical results under both cases are derived using both constant and nonsummable step size rules. Our analytical assertions showed that the feasible points computed by the proposed algorithms converge into a neighborhood of optimality. The size of the neighborhood was explicitly quantified in terms of the underlying inexactness. Further, all the preceding convergence assertions were extended to a general consensus formulation. Finally, numerical experiments were conducted to verify the theoretical results.

5.2 Future Work

In general, the subgradient methods are commonly used to solve many distributed problems due to their simplicity. However, it would be interesting to analyze other methods such as ADMM or higher order methods to see whether faster convergences can be achieved. Further, different types of errors can be explored other than the additive and bounded errors to see whether it would be possible to get more closer to optimality. In this thesis, we have discussed the inexactness of dual decomposition methods with a deterministic error. However, analyzing the methods with stochastic errors, then which leads to stochastic subgradient methods would be more convenient with many realistic applications. Further, the subproblem coordination in our proposed algorithms were considered with error-free broadcast channels between subsystems. Nevertheless, it would be more interesting to seek how the results can be extended with more realistic networks. Moreover, another path of research is extending the results with nondifferentiable settings.

References

- [1] D. P. Palomar and Y. C. Eldar, *Convex Optimization in Signal Processing and Communications*, Cambridge University Press, New York, 2010.
- [2] H. Hellström, J. M. B. da Silva Jr, V. Fodor, and C. Fischione, “Wireless for machine learning,” 2020. [Online]. Available: <https://arxiv.org/abs/2008.13492>
- [3] A. Nedić, *Convergence Rate of Distributed Averaging Dynamics and Optimization in Networks*. Foundations and Trends in Systems and Control, 2015.
- [4] T. Yang, X. Yi, J. Wu, Y. Yuan, D. Wu, Z. Meng, Y. Hong, H. Wang, Z. Lin, and K. H. Johansson, “A survey of distributed optimization,” *Annual Reviews in Control*, vol. 47, pp. 278–305, 2019.
- [5] A. Nedić and J. Liu, “Distributed optimization for control,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. 77–103, 2018.
- [6] L. Xiao, M. Johansson, and S. Boyd, “Simultaneous routing and resource allocation via dual decomposition,” *IEEE Transactions on Communications*, vol. 52, no. 7, pp. 1136–1144, 2004.
- [7] R. Madan and S. Lall, “Distributed algorithms for maximum lifetime routing in wireless sensor networks,” *IEEE Transactions on Wireless Communications*, vol. 5, no. 8, pp. 2185–2193, Aug 2006.
- [8] A. Nedić and A. Ozdaglar, “Distributed subgradient methods for multi-agent optimization,” *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.
- [9] Q. Zhang, K. Dehghanpour, Z. Wang, F. Qiu, and D. Zhao, “Multi-agent safe policy learning for power management of networked microgrids,” *IEEE Transactions on Smart Grid*, vol. 12, no. 2, pp. 1048–1062, 2021.

- [10] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, “A survey of distributed optimization and control algorithms for electric power systems,” *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2941–2962, 2017.
- [11] B. T. Polyak, *Introduction to Optimization*. NY: Optimization Software, Inc., Publications Division, 1987.
- [12] A. Nedić, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, “Distributed subgradient methods and quantization effects,” in *2008 47th IEEE Conference on Decision and Control*, 2008, pp. 4177–4184.
- [13] P. Yi and Y. Hong, “Quantized subgradient algorithm and data-rate analysis for distributed optimization,” *IEEE Transactions on Control of Network Systems*, vol. 1, no. 4, pp. 380–392, 2014.
- [14] C.-S. Lee, N. Michelusi, and G. Scutari, “Finite rate quantized distributed optimization with geometric convergence,” in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, 2018, pp. 1876–1880.
- [15] M. V. Solodov and S. K. Zavriev, “Error stability properties of generalized gradient-type algorithms,” *Journal of Optimization Theory and Applications*, vol. 98, no. 3, pp. 663–680, 1998.
- [16] M. Rabbat and R. Nowak, “Quantized incremental algorithms for distributed optimization,” *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 798–808, 2005.
- [17] D. P. Bertsekas and J. N. Tsitsiklis, “Gradient convergence in gradient methods with errors,” *SIAM Journal on Optimization*, vol. 10, no. 3, pp. 627–642, 1999.
- [18] A. Nedić and D. P. Bertsekas, “The effect of deterministic noise in subgradient methods,” *Mathematical Programming*, vol. 125, no. 1, p. 75–99, 2010.

- [19] O. Devolder, F. Glineur, and Y. Nesterov, "First-order methods of smooth convex optimization with inexact oracle," *Mathematical Programming*, vol. 146, pp. 37–75, 2014.
- [20] J. Chen and R. Luss, "Stochastic gradient descent with biased but consistent gradient estimators," *CoRR*, vol. abs/1807.11880, 2018. [Online]. Available: <http://arxiv.org/abs/1807.11880>
- [21] Y. Hu, S. Zhang, X. Chen, and N. He, "Biased stochastic gradient descent for conditional stochastic optimization," 2020. [Online]. Available: <http://arxiv.org/abs/2002.10790>
- [22] A. Ajalloeian and S. U. Stich, "On the convergence of sgd with biased gradients," 2021. [Online]. Available: <https://arxiv.org/abs/2008.00051>
- [23] S. Khirirat, S. Magnússon, and M. Johansson, "Compressed gradient methods with hessian-aided error compensation," *IEEE Transactions on Signal Processing*, vol. 69, pp. 998–1011, 2021.
- [24] I. Necoara and V. Nedelcu, "Rate analysis of inexact dual first-order methods application to dual decomposition," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1232–1243, 2014.
- [25] S. Magnússon, C. Enyioha, N. Li, C. Fischione, and V. Tarokh, "Communication complexity of dual decomposition methods for distributed resource allocation optimization," *IEEE Journal on Selected Topics in Signal Processing*, vol. 12, no. 4, p. 717–732, 2018.
- [26] Y. Su, Z. Wang, M. Cao, M. Jia, and F. Liu, "Convergence analysis of dual decomposition algorithm in distributed optimization: Asynchrony and inexactness," 2021. [Online]. Available: <https://arxiv.org/abs/2103.02784>

- [27] S. Magnússon, C. Enyioha, N. Li, C. Fischione, and V. Tarokh, “Convergence of limited communication gradient methods,” *IEEE Transactions on Automatic Control*, vol. 63, no. 5, pp. 1356–1371, 2018.
- [28] S. Magnússon, H. Shokri-Ghadikolaei, and N. Li, “On maintaining linear convergence of distributed learning and optimization under limited communication,” *IEEE Transactions on Signal Processing*, vol. 68, pp. 6101–6116, 2020.
- [29] A. Nedić, A. Olshevsky, and W. Shi, *Decentralized Consensus Optimization and Resource Allocation*. Cham: Springer International Publishing, 2018, pp. 247–287.
- [30] J. C. Duchi, A. Agarwal, and M. J. Wainwright, “Dual averaging for distributed optimization: Convergence analysis and network scaling,” *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 592–606, 2012.
- [31] K. Cai and H. Ishii, “Average consensus on arbitrary strongly connected digraphs with time-varying topologies,” *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 1066–1071, 2014.
- [32] C. D. Persis, E. Weitenberg, and F. Dörfler, “A power consensus algorithm for dc microgrids,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 10 009–10 014, 2017.
- [33] Q. Li, D. W. Gao, H. Zhang, Z. Wu, and F.-y. Wang, “Consensus-based distributed economic dispatch control method in power systems,” *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 941–954, 2019.
- [34] J. N. Tsitsiklis, “Problems in decentralized decision making and computation,” Ph.D. dissertation, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 1984.
- [35] L. Georgopoulos and M. Hasler, “Distributed machine learning in networks by consensus,” *Neurocomputing*, vol. 124, pp. 2–12, 2014.

- [36] P. A. Forero, A. Cano, and G. B. Giannakis, “Consensus-based distributed support vector machines,” *Journal of Machine Learning Research*, vol. 11, pp. 1663–1707, 2010.
- [37] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, USA, 2004.
- [38] H. K. Abeynanda and G. H. J. Lanel, “A study on distributed optimization over large-scale networked systems,” *Journal of Mathematics*, vol. 2021, 2021.
- [39] H. Abeynanda, C. Weeraddana, G. H. J. Lanel, and C. Fischione, “On the primal feasibility in dual decomposition methods under additive and bounded errors,” 2021. [Online]. Available: <https://arxiv.org/abs/2111.02525>
- [40] H. Abeynanda and G. H. J. Lanel, “Convergence of gradient methods with deterministic and bounded noise,” in *Proceedings of SLIIT International Conference on Advancements in Sciences and Humanities, 2022*, pp. 189 – 195.
- [41] G. Calafiore and L. E. Ghaoui, *Optimization Models*, Cambridge University Press, UK, 2014.
- [42] M. Zibulevsky and M. Elad, “L1-L2 Optimization in signal and image processing,” *IEEE Signal Processing Magazine*, vol. 27, no. 3, pp. 76–88, 2010.
- [43] Z. Luo and W. Yu, “An introduction to convex optimization for communications and signal processing,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1426–1438, 2006.
- [44] S. Dedu and F. Şerban, “Multiobjective mean-risk models for optimization in finance and insurance,” *Procedia Economics and Finance*, vol. 32, pp. 973–980, 2015.
- [45] R. T. Rockafellar and R. J. B. Wets, *Variational Analysis*. Berlin: Springer-Verlag Berlin Heidelberg, 2009.

- [46] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed., Athena Scientific, Belmont, MA, 1999, 3rd printing (2008).
- [47] B. T. Polyak, *Introduction to Optimization*, Optimization Software, Inc., Publications Division, 4 Park Avenue, New York, 1987.
- [48] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, “A survey of distributed optimization and control algorithms for electric power systems,” *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2941–2962, 2017.
- [49] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, “Rate control for communication networks: Shadow prices, proportional fairness and stability,” *Journal of the Operational Research Society*, vol. 49, pp. 237–252, 1998.
- [50] S. H. Low and D. E. Lapsley, “Optimization flow control—I: basic algorithm and convergence,” *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 861–874, Dec 1999.
- [51] D. P. Palomar and M. Chiang, “A tutorial on decomposition methods for network utility maximization,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1439–1451, aug 2006.
- [52] D. Palomar and M. Chiang, “Alternative distributed algorithms for network utility maximization: Framework and applications,” *IEEE Transactions on Automatic Control*, vol. 52, no. 12, pp. 2254–2269, Dec 2007.
- [53] N. Li, L. Chen, and S. H. Low, “Optimal demand response based on utility maximization in power networks,” in *2011 IEEE Power and Energy Society General Meeting*, 2011, pp. 1–8.

- [54] L. Chen, N. Li, S. H. Low, and J. C. Doyle, "Two market models for demand response in power networks," in *IEEE International Conference on Smart Grid Communications*, 2010, pp. 397–402.
- [55] R. Madan and S. Lall, "Distributed algorithms for maximum lifetime routing in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 5, no. 8, pp. 2185–2193, Aug 2006.
- [56] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.
- [57] S. Boyd and L. Vandenberghe, "Interior-point methods." [Online]. Available: <https://web.stanford.edu/class/ee364a/lectures/barrier.pdf>
- [58] S. Boyd, "Subgradient methods," 2007, [Online]. Available: http://stanford.edu/class/ee364b/lectures/subgrad_method_notes.pdf.
- [59] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, p. 127–239, January 2014.
- [60] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 592–606, 2012.
- [61] S. Boyd, L. Xiao, A. Mutapcic, and J. Mattingley, "Notes on decomposition methods," 2007. [Online]. Available: http://stanford.edu/class/ee364b/lectures/decomposition_notes.pdf
- [62] J. N. Tsitsiklis, "Problems in decentralized decision making and computation," Ph.D. dissertation, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 1984.

- [63] J. Tsitsiklis, D. Bertsekas, and M. Athans, “Distributed asynchronous deterministic and stochastic gradient optimization algorithms,” *IEEE Transactions on Automatic Control*, vol. 31, no. 9, pp. 803–812, 1986.
- [64] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, 2nd ed., Athena Scientific, Belmont, Massachusetts, USA, 1997.
- [65] G. B. Dantzig and P. Wolfe, “Decomposition principle for linear programs,” *Operations Research*, vol. 8, no. 1, pp. 101–111, 1960.
- [66] S. Boyd and A. M. J. Duchi, “Stochastic subgradient methods,” 2018, [Online]. Available: https://web.stanford.edu/class/ee364b/lectures/stoch_subgrad_notes.pdf.
- [67] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, “Layering as optimization decomposition: A mathematical theory of network architectures,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, 2007.
- [68] S. Shakkottai and R. Srikant, “Network optimization and control,” *Foundations and Trends in Networking*, vol. 2, no. 3, pp. 271–379, 2007.
- [69] R. Srikant, *The Mathematics of Internet Congestion Control (Systems and Control: Foundations and Applications)*. SpringerVerlag, 2004.
- [70] A. Beck, A. Nedić, A. Ozdaglar, and M. Teboulle, “An $o(1/k)$ gradient method for network resource allocation problems,” *IEEE Transactions on Control of Network Systems*, vol. 1, no. 1, pp. 64–73, 2014.
- [71] A. Nedić and A. Ozdaglar, “Distributed subgradient methods for multi-agent optimization,” *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.
- [72] Y. Nesterov, *Lectures on Convex Optimization*. Gewerbestrasse 11, 6330 Cham, Switzerland: Springer International Publishing, 2018.

- [73] D. Gabay and B. Mercier, “A dual algorithm for the solution of nonlinear variational problems via finite element approximation,” *Computers & Mathematics with Applications*, vol. 2, pp. 17–40, 1976.
- [74] J. Eckstein and D. P. Bertsekas, “On the douglas—rachford splitting method and the proximal point algorithm for maximal monotone operators,” *Mathematical Programming*, vol. 55, pp. 293–318, 1992.
- [75] J. Eckstein and M. C. Ferris, “Operator-splitting methods for monotone affine variational inequalities, with a parallel application to optimal control,” *INFORMS Journal on Computing*, vol. 10, no. 2, pp. 218–235, 1998.
- [76] G. Chen and M. Teboulle, “A proximal-based decomposition method for convex minimization problems,” *Mathematical Programming*, vol. 64, pp. 81–101, 1994.
- [77] N. Patari, V. Venkataramanan, A. Srivastava, D. K. Molzahn, N. Li, and A. Anaswamy, “Distributed optimization in distribution systems: Use cases, limitations, and research needs,” *IEEE Transactions on Power Systems*, 2021.
- [78] S. Pu, W. Shi, J. Xu, and A. Nedić, “Push-pull gradient methods for distributed optimization in networks,” *IEEE Transactions on Automatic Control*, vol. 66, no. 1, pp. 1–16, 2021.
- [79] A. Nedić and A. Ozdaglar, “On the rate of convergence of distributed subgradient methods for multi-agent optimization,” in *2007 46th IEEE Conference on Decision and Control*, 2007, pp. 4711–4716.
- [80] L. Romao, K. Margellos, G. Notarstefano, and A. Papachristodoulou, “Convergence rate analysis of a subgradient averaging algorithm for distributed optimisation with different constraint sets,” in *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2019, pp. 7448–7453.

- [81] X. Ren, D. Li, Y. Xi, and H. Shao, “Distributed subgradient algorithm for multi-agent optimization with dynamic stepsize,” *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 8, pp. 1451–1464, 2021.
- [82] T. Halsted, O. Shorinwa, J. Yu, and M. Schwager, “A survey of distributed optimization methods for multi-robot systems,” 2021. [Online]. Available: <https://arxiv.org/abs/2103.12840>
- [83] J. Alonso-Mora, E. Montijano, M. Schwager, and D. Rus, “Distributed multi-robot formation control among obstacles: A geometric and optimization approach with consensus,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 5356–5363.
- [84] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, USA, 2004.
- [85] J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer, “A survey on distributed machine learning,” 2019. [Online]. Available: <http://arxiv.org/abs/1912.09789>
- [86] G. Li, S. K. S. Hari, M. Sullivan, T. Tsai, K. Pattabiraman, J. Emer, and S. W. Keckler, “Understanding error propagation in deep learning neural network (dnn) accelerators and applications,” in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2017.
- [87] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, SIAM, 2000.
- [88] A. Beck, *First-Order Methods in Optimization*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2017.

Appendix 1: List of Publications

1. **Hansi K. Abeynanda**, and G. H. J. Lanel, “A Study on Distributed Optimization over Large-Scale Networked Systems,” *Journal of Mathematics*, vol. 2021, Article ID 5540262, 19 pages, 2021, <https://doi.org/10.1155/2021/5540262>.
2. **H. Abeynanda**, C. Weeraddana, G. H. J. Lanel, and C. Fischione, “On the primal feasibility in dual decomposition methods under additive and bounded errors,” 2021. [Online]. Available: <https://arxiv.org/abs/2111.02525>. (Accepted for publication in the IEEE Transactions on Signal Processing).
3. **Hansi Abeynanda**, G. H. Jayantha Lanel. (2022). Convergence of Gradient Methods with Deterministic and Bounded Noise. Proceedings of SLIIT International Conference on Advancements in Sciences and Humanities, (11) October, Colombo, 189 - 195.