The Dissertation Committee for Abolfazl Hashemi
certifies that this is the approved version of the following dissertation:

Efficient Algorithms for Structured Inference and Collaborative Learning

Committee:

Haris Vikalo, Supervisor

Gustavo de Veciana

Alex Dimakis

Qiang Liu

Sujay Sanghavi

Efficient Algorithms for Structured Inference and Collaborative Learning

by

Abolfazl Hashemi

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

August 2020

Efficient Algorithms for Structured Inference and Collaborative Learning

Publication No. _____

Abolfazl Hashemi, Ph.D.
The University of Texas at Austin, 2020

Supervisor: Haris Vikalo

Massive amounts of data collected by modern information systems give rise to new challenges in the fields of signal processing, machine learning, and data analysis. In contemporary large-scale datasets, there are often hidden low-dimensional structures either in the form of parsimonious representations that best fit the data or the desired unknown information itself. Identifying parsimonious representations and exploiting underlying structural constraints lead to improved inference. Furthermore, these large-scale datasets are distributed among a network of resource-constrained systems capable of exchanging information. Hence, designing accelerated and communication efficient learning and inference algorithms is of critical importance. In the first part of this dissertation, we first study the setting where the unknown parameter of interest has hidden sparsity structures. The task of reconstructing

the sparse parameter can be formulated as an ℓ_0-constrained least square problem. Motivated by the need for fast and accurate sparse recovery in large-scale setting, we propose two efficient sparse reconstruction and support selection algorithms and analyze their reconstruction performance in a variety of settings. Next, we consider applications of the proposed algorithms in structured data clustering problems where the high-dimensional data is a collection of points lying on a union of low-dimensional and evolving subspaces. By exploiting sparsity to model the low-dimensional union-of-subspaces structure of the data as well as its underlying evolutionary structure, we propose a novel evolutionary subspace clustering framework and demonstrate its successful deployment in computer vision and oceanography applications. In the second part of this dissertation, we consider observation selection and information gathering algorithms in communication-constrained networked systems where we study structural properties of observation selection criteria, design efficient greedy algorithms, and analyze their performance by leveraging the framework of weak submodular optimization. In the final part of this dissertation, we study the task of learning parameters of a machine learning model in a collaborative manner over a communication-constrained network, and design an efficient communication compressing optimization algorithm that reduces the amount of communication in the network while achieving a near optimal converge rate for general nonconvex learning tasks.

Table of Contents

Abstract **iv**

List of Tables **x**

List of Figures **xi**

Chapter 1. Introduction **1**
1.1 Motivation 1
1.2 Research Topics 2
1.3 Thesis Contributions 6
1.4 Thesis Outline 9

Chapter 2. Methodological Background **12**
2.1 Notation 12
2.2 Sparse Reconstruction and Support Selection 13
2.3 Weak Submodular Optimization 15
2.4 Decentralized Optimization 21

Chapter 3. Greedy Algorithms for Sparse Reconstruction **26**
3.1 Introduction 27
3.2 Accelerated Orthogonal Least Squares Algorithm 31
3.2.1 Analysis of sample complexity 38
3.2.2 Numerical experiments 42
3.3 Progressive Stochastic Greedy Algorithm 50
3.3.1 Analysis of sample complexity 52
3.3.2 Importance of progression to m evaluations 56
3.3.3 Numerical experiments 58
3.4 Applications 60
3.4.1 Sparse subspace clustering 60

3.4.2 Column subset selection 64
3.5 Conclusion 69

Chapter 4. Evolutionary Subspace Clustering 71
4.1 Introduction 72
4.1.1 Connection to subspace clustering 74
4.1.2 Connection to evolutionary clustering 76
4.2 Evolutionary Subspace Clustering 81
4.2.1 Convex evolutionary self-expressive model 83
4.3 Alternating Minimization Algorithms for Evolutionary Subspace Clustering 85
4.3.1 Finding parameters of the CESM model 85
4.3.2 Complexity analysis 87
4.4 Practical Extensions 88
4.4.1 Tracking the evolution of clusters 88
4.4.2 Adding and removing data points over time 89
4.4.3 Accelerated representation learning 89
4.4.4 Dealing with outliers and missing entries 92
4.5 Numerical Experiments 93
4.5.1 Synthetic data 94
4.5.2 Real-time motion segmentation 99
4.5.3 Ocean water mass clustering 103
4.6 Conclusion 107

Chapter 5. Submodular Observation Selection in Networks 109
5.1 Introduction 110
5.2 Observation Selection in Linear Model 116
5.2.1 System model 116
5.2.2 Weak submodular linear sensor selection 119
5.2.3 Numerical experiments 126
5.3 Observation Selection in Quadratic Model 127
5.3.1 System model 127
5.3.2 Proposed formulation 131

5.3.3 Greedy selection of range observations 134
5.3.4 Numerical experiments 137
5.4 Randomized Greedy Observation Selection 139
5.4.1 Proposed scheme 139
5.4.2 Performance analysis of the proposed scheme 141
5.4.3 Numerical Experiments 149
5.4.3.1 Kalman filtering in random sensor networks . . 150
5.4.3.2 State estimation in large-scale networks 153
5.4.3.3 Accelerated multi-object tracking 154
5.5 Submodular Information-Exchange Communication Protocol . 156
5.5.1 System model 157
5.5.2 Proposed formulation 161
5.5.3 Numerical experiments 167
5.6 Conclusion 169

Chapter 6. Compressed Decentralized Optimization via Multiple Gossip Steps 171
6.1 Introduction 172
6.1.1 Significance and Related Work 174
6.2 Multi-step Gossip Decentralized Gradient Descent 177
6.3 Convergence Analysis 179
6.4 Numerical Experiments 184
6.5 Conclusion 188

Chapter 7. Conclusion and Future Work 189
7.1 Conclusions 189
7.2 Future Work 190

Appendices 193

Appendix A. Missing Proofs from Chapter 3 **194**
A.1 Useful Lemmas 194
A.2 Proof of Theorem 3.2.2 197
A.3 Proof of Theorem 3.2.3 200
A.4 Proof of Theorem 3.3.1 204
A.5 Proof of Theorem 3.3.2 213
A.6 Proof of Theorem 3.3.3 219

Appendix B. Missing Proofs from Chapter 5 **223**
B.1 Proof of Proposition 5.2.1 223
B.2 Proof of Lemma 5.4.1 224

Appendix C. Missing Proofs from Chapter 6 **227**
C.1 Proof of Theorem 6.3.1 228
C.1.1 Useful lemmas 229
C.1.2 Proof of the main theorem 235
C.2 Proof of Theorem 6.3.2 239
C.2.1 Useful lemmas 239
C.2.2 Proof of the main theorem 244

Bibliography **247**

List of Tables

4.1 Performance comparison of static and various evolutionary subspace clustering algorithms on real-time motion segmentation dataset. The best results for each row are in boldface fonts. For the CESM framework, the top results in each row correspond to the case of using a constant smoothing factor with the lowest average error while the bottom results in each row are achieved by using the proposed alternating minimization schemes to learn the smoothing parameter at each time step. . 100

4.2 Average salinity and temperature of four different types of water masses at 1000 dbar near the coast of south Africa identified by CESM framework employing AOLS-based representation learning strategy with $L = 3$ at different time steps. The results in top, middle and bottom for each cluster correspond to $t = 2, 4, 6$, respectively. 105

5.1 Running time comparison of the randomized greedy, greedy, and SDP relaxation sensor selection schemes ($m = 50$, $n = 400$, $K = 55$, $\epsilon = 0.001$). 151

6.1 Comparison of convergence rates of different decentralized optimization algorithms under smoothness. In the table, $\rho, \rho_1, \rho_2, \rho_3 \in (0, 1)$ denote the rate of linear convergence. α_1, α, and C depend on network and function properties. Further, $\alpha < 1$, $Q > 1$ is the number of rounds of consensus, and C depends on compression rate. In the table, SC stands for strong convexity. 174

List of Figures

3.1 Number of noiseless measurements required for sparse reconstruction with $\beta^2 = 0.05$ when $m = 1024$. The regression line is $n = 2.0109\ k \log(\frac{m}{k\sqrt[3]{\beta}})$ with the coefficient of determination $R^2 = 0.9888$. 42

3.2 A comparison of the theoretical probability of exact recovery provided by Theorem 3.2.2 with the empirical one, where $m = 1024$ and the non-zero elements of $\mathbf{x}$ are drawn independently from a normal distribution. 43

3.3 A comparison of the theoretical probability of exact recovery provided by Theorem 3.2.3 with the empirical one, where $m = 1024$ and non-zero elements $\mathbf{x}$ are set to $(1 + \delta + 20)\|\mathbf{e}\|_2$. . . . 44

3.4 Exact recovery rate comparison of AOLS, MOLS, OMP, MMP-DP, MMP-BP, and LASSO for $n = 512$, $m = 1024$, and k non-zero components of $\mathbf{x}$ uniformly drawn from $\mathcal{N}(0, 1)$ distribution. 45

3.5 Partial recovery rate comparison of AOLS, MOLS, OMP, MMP-DP, MMP-BP, and LASSO for $n = 512$, $m = 1024$, and k non-zero components of $\mathbf{x}$ uniformly drawn from $\mathcal{N}(0, 1)$ distribution. 46

3.6 A comparison of AOLS, MOLS, OMP, MMP-DP, MMP-BP, and LASSO for $n = 512$, $m = 1024$, and k non-zero components of $\mathbf{x}$ uniformly drawn from the $\mathcal{N}(0, 1)$ distribution. 47

3.7 Empirical exact recovery rate of PSG for various values of β. . 58

3.8 Empirical evaluation of the theoretical bounds established by Theorem Theorem 3.3.3. 59

3.9 Performance comparison of ASSC, SSC-OMP [1, 2], and SSC-BP [3,4] on synthetic data with no perturbation. The points are drawn from 5 subspaces of dimension 6 in ambient dimension 9. Each subspace contains the same number of points and the overall number of points is varied from 250 to 5000. 61

3.10 Performance comparison of ASSC, SSC-OMP [1, 2], and SSC-BP [3,4] on synthetic data with perturbation terms $Q \sim \mathcal{U}(0, 1)$. The points are drawn from 5 subspaces of dimension 6 in ambient dimension 9. Each subspace contains the same number of points and the overall number of points is varied from 250 to 5000. 62

3.11 Performance comparison of various CSS schemes and the top-k SVD lower bound on a synthetic data. 66

3.12 Face clustering: given images of multiple subjects, the goal is to find images that belong to the same subject (Examples from the EYaleB dataset [5]). 67

3.13 Performance comparison of various CSS schemes on EYaleB dataset. 68

4.1 Comparison of clustering accuracy of static and various evolutionary subspace clustering schemes employing OMP-based representation learning strategy on a simulated data containing 500 points that belong to a union of 10 rotating random subspaces in $\mathbb{R}^{10}$, each of dimension 6. The proposed CESM framework significantly improves the clustering accuracy and is superior to the AFFECT strategy. Moreover, CESM framework adapts to subspace changes at times $t = 6, 13$ as shown in the right-most plots. 95

4.2 Comparison of the smoothing parameter α for various evolutionary subspace clustering schemes employing OMP-based representation learning strategy on a simulated data containing 500 points lying on a union of 10 rotating random subspaces in $\mathbb{R}^{10}$, each of dimension 6. AFFECT's smoothing parameter remains approximately constant regardless of the underlying evolutionary behavior while the smoothing parameter for the CESM framework dynamically reflects the structure and reacts to cluster changes. 96

4.3 Example frames from the videos in the Hopkins 155 dataset [6]. 97

4.4 Clustering results of four different types of water masses at 1000 dbar near the coast of south Africa (colored with blue). using static and various evolutionary subspace clustering schemes employing AOLS-based representation learning strategy with $L = 3$. The static subspace clustering scheme and AFFECT fail to keep track of the orange water mass at time $t = 6$ and $t = 4$, respectively. However, our proposed CESM framework detects homogeneous water masses across all time steps. . . . 104

5.1 Evaluation of theoretical results in Theorem 5.2.1 for a sensor network with $m = 3$ and $n = 12$. 126

5.2 Comparison of MSEs for random, linearized, and quadratic observation selection schemes in the multi-target tracking application. 137

5.3 MSE comparison of randomized greedy, greedy, and SDP relaxation sensor selection schemes employed in Kalman filtering. . 151
5.4 Comparison of randomized greedy, greedy, and SDP relaxation schemes as the number of selected sensors increases. 152
5.5 Histogram of MSE values for 100 independent realization of a sensor scheduling task for a sensor network with $m = 50$, $K = 60$, and $n = 400$. 153
5.6 A comparison of the randomized greedy and greedy algorithms for varied network size. 154
5.7 Multi-object tracking via a swarm of UAVs. The UAVs can communicate with each other and are equipped with GPS and radar systems. The objective is to select a small subset of range and angular measurements gathered by the UAVs to communicate to the control unit. 155
5.8 A comparison of the randomized greedy and greedy algorithms for a multi-object tracking application. 155
5.9 A fully connected network of units with sensing, communication, and processing capabilities; the communication between the units is constrained. Also shown is a scheduler R that organizes exchange of observations $\mathcal{O}_{i,j}$ and $\mathcal{O}_{i,k}$ to node i from nodes j and k, respectively. 157
5.10 Comparison of sum of pairwise node-level MSE distances . . . 167

6.1 Effect of varying Q under different consensus learning rates γ. MNIST setting (top row): $n = 9$, top(0.05); SYN-1 setting (mid row): $n = 16$, qsgd_2; SYN-2 (bottom row): $n = 16$, qsgd_2, ℓ_2-regularization value = 0.001. We used $\eta = 0.2$ and torus topology for all datasets. 186
6.2 Effect of network topology settings on the convergence rates; showing dependence on the number of nodes n and mixing steps Q. MNIST setting(top row): top(0.05), $\gamma = 0.5$, $\eta = 0.2$. SYN-1 setting (bottom row): qsgd_2, $\gamma = 0.1$; We used $\eta = 0.2$ for $n = 9, 16$ and $\eta = 0.15$ for $n = 25$, respectively. SYN2 setting (bottom row): qsgd_2, $\eta = 0.2$, ℓ_2-regularization parameter = 0.001; We used $\gamma = 0.1$ for $n = 9, 16$ and $\gamma = 0.15$ for $n = 25$. 187
6.3 Comparison of various compression operators over torus topology. MNIST (Left): $n = 9, \eta = 0.2$; For qsgd_2, top(0.05), rand(0.05), we used $Q = \{10, 15, 15\}$ and $\gamma = \{0.05, 0.1, 0.05\}$, respectively. SYN-1 (Middle): $n = 16, \eta = 0.2$, $Q = 5$; For qsgd_2, top(0.05), and rand(0.05), we used $\gamma = 0.2$, 0.2 and 0.05, respectively. SYN2 (Right): $n = 16, \eta = 0.2$, $Q = 5$, ℓ_2-regularization parameter = 0.001; For qsgd_2, top(0.05), and rand(0.05) we used $\gamma = 0.2$, $\gamma = 0.2$, and $\gamma = 0.05$, respectively. 188

Chapter 1

Introduction

1.1 Motivation

In recent years, data-driven approaches have become immensely popular in science and engineering due to their successes in a wide variety of applications. The enabling factor for success of modern data-driven systems is the ability to learn efficient algorithms from massive amounts of large-scale data. Such large-scale and high-dimensional datasets have certain structures and properties, including: (i) hidden low-dimensional structures in the form of parsimonious representations that best fit the data, and (ii) being distributed across a network of resource-constrained systems capable of exchanging information. Identifying and exploiting these underlying relational and structural constraints will not only enable us to improve performance of current data-driven approaches, but also will help us achieve deeper understanding of how and when the designed learning schemes succeed. In this dissertation, we aim to contribute to solving these fundamental challenges by improving the computational efficiency of finding solutions to several large-scale inference and learning problems.

While structured and distributed inference and learning tasks are well-

studied in a wide range of scenarios, large-scale problems dealing with massive amounts of high-dimensional data still present numerous challenges. For instance, for sparse regression and support selection tasks where the data often contains hidden low-dimensional structures, existing algorithms entail a computational cost that is linear in the number of variables representing the features of the underlying problem. Furthermore, ever-increasing size of the models in distributed and collaborative learning tasks poses communication burdens on existing schemes that rely on exact (full) communication among the participating agents in the network. Motivated by these observations and challenges, in this dissertation we focus on the design and analysis of efficient algorithms for structured regression and clustering tasks as well as communication-efficient optimization and information-sharing schemes for collaborative learning problems.

1.2 Research Topics

The first part of this dissertation presents our work on the development and analysis of efficient algorithms for identifying low-dimensional structures for improved performance in regression and clustering. First, we consider the task of large-scale sparse reconstruction and support selection where the goal is the recover an unknown sparse signal from few linear measurements of its coordinates. Specifically, this involves solving a least-squares regression problem where the number of non-zero entries in the unknown vector is constrained by a given upper bound. This task is encountered in a number of settings in

machine learning and signal processing including sparse linear regression [7], compressed sensing [8], image processing [9], subspace clustering [3], column subset selection [10], and group testing [11]. In each of these applications, exploiting the information about sparsity of the unknown variable enables finding a near optimal solution in a faster and more accurate fashion. In this part of the dissertation, we present two efficient greedy algorithms and theoretical analyze their reconstruction performance. In doing so, we show that the proposed schemes are able to reconstruct the unknown sparse vector exactly with high probability from a few random linear measurements. We further show that the proposed schemes achieve the information theoretic lower bound on sample complexity of this problem while requiring lower computational costs compared to existing schemes. The proposed schemes have been applied to a structured clustering and dimensionality reduction tasks in signal processing and machine learning.

In the next part of this dissertation, our focus is on development of efficient low-dimensional representation learning schemes for the problem of structured clustering of temporally evolving datasets. Specifically, we consider problems where the data can be thought of as being a collection of points lying on a union of low-dimensional and evolving subspaces. Such tasks are encountered in many applications including motion segmentation and face clustering in computer vision [12, 13], image representation and compression in image clustering [14, 15], robust principal component analysis (PCA), and robust subspace recovery and tracking [16–20]. Exploiting the temporal behavior of

the data as well as the underlying low-dimensional subspace structure provides more informative description and enables improved clustering accuracy. To this end, we provide a mathematical formulation of evolutionary subspace clustering and introduce the convex evolutionary self-expressive model, an optimization framework that exploits the self-expressiveness property of data and learns sparse representations while taking into account prior representations. We demonstrate that learning parameters of the proposed framework requires finding solutions to a nonconvex optimization problem which we solve approximately by relying on the alternating minimization ideas. In the process of learning data representation, we automatically tune a smoothing parameter which is reflective of the rate of evolution of the data and signifies the amount of temporal changes in consecutive data snapshots. We demonstrate that the proposed framework significantly improves the performance and shortens runtimes of state-of-the-art clustering algorithms through testing the performance of the proposed scheme in two real-world applications, namely, real-time motion segmentation and a study of evolution of ocean water masses.

In the next contribution of this dissertation, we consider the task of observation selection in resource-constrained networks. Sensor networks deploy a large number of nodes that either exchange their noisy and possibly processed observations of a random process or forward those observations to a data fusion center. Due to constraints on computation, power and communication resources, instead of estimating the process using information collected by the entire network, the fusion center typically queries a relatively small sub-

set of the available sensors. The problem of selecting the sensors that would acquire the most informative observations arises in a number of applications in control and signal processing systems including sensor selection for Kalman filtering [21–23], batch state and stochastic process estimation [24, 25], minimal actuator placement [26, 27], voltage control and meter placement in power networks [28–30], sensor scheduling in wireless sensor networks [21, 31], and subset selection in machine learning [32–34]. For a variety of performance criteria, finding an optimal subset of sensors requires solving a computationally challenging combinatorial optimization problem. Additionally, non-linearity of the observation model as well as the large number of gathered measurements by contemporary networked systems calls for efficient observation selection and information gathering schemes with guaranteed performance. To address these challenges, we examine conditions under which the mean-square error behaves similar to a submodular function. We further propose a randomized greedy algorithm for observation selection and establish performance guarantees on its achievable mean-square error. Finally, we propose a novel submodular information-exchange protocol to reduce the amount of communication in a network of sensing units operating under communication constraints. The proposed schemes have been tested in multi-target tracking applications via a swarm of UAVs.

In the final part of this dissertation, we study the problem of decentralized learning over communication-constrained networks where the goal of participating clients is to collaboratively optimize a global objective. This

tasks arises in many important distributed machine learning, signal processing and control tasks, such as federated learning and multi-agent systems [35–37]. Solving such distributed tasks is often facilitated by communication of agents' local model parameters over a network that that limits the amount of information they can exchange. Compared to a centralized optimization framework, distributed optimization enables locality of data storage and model updates which in turn offers computational advantages by delegating computation to multiple clients, and further promotes preservation of privacy of user information [35]. As the size of ML models grows, exchanging information across the network becomes a major challenge in distributed optimization [37]. It is therefore imperative to design communication-efficient strategies which reduce the amount of communicated data by performing compressed communication while at the same time, despite the use of compressed communication, achieve convergence that is on par with the performance of centralized and distributed methods utilizing uncompressed information [37–39]. To this end, we propose an iterative decentralized optimization algorithm with compressed communication and establish theoretical guarantees on its achievable convergence rate for a variety of decentralized learning scenarios. We further demonstrate efficacy of the proposed scheme in the context of distributed regression and classification.

1.3 Thesis Contributions

The contributions made in this dissertation are formally stated below.

Greedy Algorithms for Sparse Reconstruction

- I developed the accelerated orthogonal least-squares and the progressive stochastic greedy algorithms for the problem of sparse reconstruction and support selection, and derived a theoretical lower bound for their probability of exact reconstruction.

- I showed that the proposed schemes achieve the information-theoretic lower bound on sample complexity while incurring lower computational costs compared to existing schemes.

- I demonstrated efficacy of the developed algorithms in practical settings, including applications to sparse subspace clustering and column subset selection.

Evolutionary Subspace Clustering

- I proposed evolutionary subspace clustering, a method whose objective is to cluster a collection of evolving data points that lie on a union of low-dimensional evolving subspaces. The proposed framework learns a parsimonious representation of the data points at each time step by establishing a non-convex optimization framework that exploits the self-expressiveness property of the evolving data while taking into account representation from the preceding time step.

- I developed a scheme based on alternating minimization that both learns the parsimonious representation as well as adaptively tunes and infers a smoothing parameter reflective of the rate of data evolution.

- I successfully employed the proposed algorithm in a motion segmentation application as well as an application in oceanography.

Submodular Observation Selection in Networks

- I established conditions under which the mean-square objective for observation selections in linear sensor network is weak submodular.

- I proposed new weak submodular observation selection criteria for sensor networks following a quadratic model.

- I proposed a new randomized greedy algorithm for weak submodular observation selection in sensor networks and established a lower bound on its worst-case achievable performance.

- I proposed a greedy information-exchange scheme to reduce the amount of communication in a network of sensing units operating under communication constraints. The proposed scheme aims to minimize the network-wise estimation error while promoting a balanced performance among the participating units.

Compressed Decentralized Optimization via Multiple Gossip Steps

- I proposed an iterative decentralized algorithm with arbitrary communication compression (both biased and unbiased compression operators) that performs multiple gossip steps in each iteration to enable fast convergence.

- I showed that the proposed scheme achieves a near-optimal convergence rate for convex and nonconvex collaborative learning tasks that satisfy the Polyak-Lojasiewicz condition.

- I demonstrated that the proposed scheme compares favorably to centralized and decentralized schemes without communication compression in a variety of convex and nonconvex learning tasks.

1.4 Thesis Outline

The rest of this dissertation is organized as follows.

To make the thesis self-contained, in Chapter 2 we provide an overview of the existing mathematical background relevant to the material in subsequent chapters. This chapter aims to facilitate comprehension of the methodologies and concepts discussed in Chapters 3, 4, 5, and 6, namely, sparse reconstruction, weak submodular optimization, and decentralized optimization methods.

Chapter 3 presents the novel greedy sparse reconstruction and support selection algorithms and discusses techniques that we adopt in the theoretical analysis. Proofs of the theoretical results are provided in Appendix A. The chapter concludes with a demonstration of the efficacy of the method when ap-

plied to sparse subspace clustering and column subset selection, two important applications in machine learning and computer vision.

Chapter 4 presents the evolutionary subspace clustering problem that deals with clustering of evolving data that lies on a union of low-dimensional subspaces. The chapter further presents an approximate solution based on alternating minimization, namely the convex evolutionary self-expressive model (CESM) algorithm and discusses potential practical issues and challenges that may come up in applications, and demonstrate how the proposed framework can be extended to handle such cases. The chapter concludes with extensive numerical experiments with applications in real-time motion segmentation and study of ocean water masses.

Chapter 5 presents the contributions of this dissertation in design of observation selection and information gathering methods for networked systems. First, we discuss conditions under which the mean-square error in linear systems is weak submodular. Then, the chapter continues to define new weak submodular performance criteria for quadratic observation models. The chapter further presents novel greedy information-exchange and observation selection algorithms along with their theoretical analysis. The proofs of the theoretical claims are provided in Appendix B.

Chapter 6 presents a decentralized communication-efficient algorithm for collaborative learning using a resource-constrained network. Detailed proofs of the theoretical claims regarding performance of the proposed scheme are provided in Appendix C.

Chapter 7 concludes the dissertation with a summary of the presented works and outlines possible avenues of future research. The

Chapter 2

Methodological Background

This chapter presents the concepts and background materials pertinent to the methods developed in subsequent chapters. We start this chapter by defining the notation that will be used throughout this dissertation.

2.1 Notation

Italic letters represent scalars and numerical constants, e.g., α, c, and C. We use calligraphic letters to denote sets, e.g., $\mathcal{S}$. Bold capital letters denote matrices, e.g., $\mathbf{A}$, while bold lowercase letters represent column vectors, e.g., $\mathbf{a}$. Matrix or vector transpose is represented by the superscript $\top$, e.g., $\mathbf{A}^\top$. We denote the j^{th} column of $\mathbf{A}$ by $\mathbf{a}_j$, and use $\mathbf{A}_\mathcal{S}$ to denote the submatrix of $\mathbf{A}$ that consists of the columns of $\mathbf{A}$ indexed by the set $\mathcal{S}$. Identity matrices of size n are represented by $\mathbf{I}_n$. Vectors and matrices of all zeros and ones are denoted by $\mathbf{0}$ and $\mathbf{1}$, respectively. We further denote the set $\{1, 2, \ldots, n\}$ by $[n]$. Further, $\|\mathbf{a}\|_2$ and $\|\mathbf{a}\|_0$ denote the Euclidean norm and the number of nonzero entries of $\mathbf{a}$, respectively. Finally, $\|\mathbf{A}\|$ denotes the Frobenius norm of matrix $\mathbf{A}$.

2.2 Sparse Reconstruction and Support Selection

In this section, we overview the problem of sparse reconstruction and discuss greedy solutions studied later in Chapters 3 and 4 in the context of performing structured clustering tasks.

The goal of sparse reconstruction, or sparse support selection, is to reconstruct a sparse vector from a relatively small number of its linear measurements. In particular, we are given a linear measurement model

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\nu}, \tag{2.1}$$

where $\mathbf{x} \in \mathbb{R}^m$ is a k-sparse unknown vector, i.e., a vector with at most k non-zero components, $\mathbf{y} \in \mathbb{R}^n$ denotes the vector of measurements, $\mathbf{A} \in \mathbb{R}^{n \times m}$ is the coefficient matrix assumed to be full rank, and $\boldsymbol{\nu} \in \mathcal{R}^n$ denotes the additive measurement noise vector. For simplicity, we here focus on the case $\boldsymbol{\nu} = \mathbf{0}$ and $\mathbf{A} \sim \mathcal{N}\left(0, \frac{1}{n}\right)$. The search for a sparse approximation of $\mathbf{x}$ leads to the NP-hard cardinality-constrained least-squares problem

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 \\ \text{subject to} \quad & \|\mathbf{x}\|_0 \leq k. \end{aligned} \tag{2.2}$$

One can readily reformulate (2.2) as a subset selection task according to the following procedure. For a fixed subset $\mathcal{S} \subset [m]$ where $|\mathcal{S}| \leq n$, we can find an approximation to $\mathbf{x}$ via the least-squares solution $\mathbf{x}_{LS} = \mathbf{A}_{\mathcal{S}}^{\dagger}\mathbf{y}$, where $\mathbf{A}_{\mathcal{S}}^{\dagger} = \left(\mathbf{A}_{\mathcal{S}}^{\top}\mathbf{A}_{\mathcal{S}}\right)^{-1}\mathbf{A}_{\mathcal{S}}^{\top}$ denotes the Moore-Penrose pseudo-inverse of $\mathbf{A}_{\mathcal{S}}$. Finding the optimal k-sparse vector $\mathbf{x}^{\star}$ is equivalent to identifying the support of $\mathbf{x}^{\star}$, i.e.,

determining the set of nonzero entries of $\mathbf{x}^\star$ which we denote by $\mathcal{S}^\star$. More formally, (2.2) is recast as

$$\begin{aligned} \underset{\mathcal{S}}{\text{minimize}} \quad & \|\mathbf{y} - \mathbf{P}(\mathcal{S})\mathbf{y}\|_2^2 \\ \text{subject to} \quad & |\mathcal{S}| \leq k, \end{aligned} \tag{2.3}$$

where $\mathbf{P}(\mathcal{S}) = \mathbf{A}_{\mathcal{S}}\mathbf{A}_{\mathcal{S}}^{\dagger}$ is the projection operator onto the subspace spanned by the columns of $\mathbf{A}_{\mathcal{S}}$. Since $\|\mathbf{y}\|_2^2 = \|\mathbf{y} - \mathbf{P}(\mathcal{S})\mathbf{y}\|_2^2 + \|\mathbf{P}(\mathcal{S})\mathbf{y}\|_2^2$, (2.3) can equivalently be written as

$$\begin{aligned} \underset{\mathcal{S}}{\text{maximize}} \quad & g(\mathcal{S}) := \|\mathbf{P}(\mathcal{S})\mathbf{y}\|_2^2 \\ \text{subject to} \quad & |\mathcal{S}| \leq k, \end{aligned} \tag{2.4}$$

which we denote by $\mathcal{P}(m, k)$. Note that since $\mathbf{A}$ is full rank, it can be shown that (2.4) has a unique solution.

Sparse reconstruction schemes can be broadly categorized as those pursuing ℓ_1 minimization and greedy schemes. Although both groups are shown to achieve the optimal sampling complexity, schemes in the latter category are typically characterized by lower computational complexity. In general, greedy schemes identify one or more non-zero components of a sparse vector in an iterative manner according to a specific selection criterion.

The vanilla greedy selection scheme for solving (2.4) is in signal processing community known as orthogonal least-squares (OLS) [40]. Orthogonal Matching Pursuit (OMP) [41] is another well-known greedy algorithm for solving the same task and can be thought of as an efficient approximation of OLS. In particular, OLS uses the criteria

$$\max_j g_j(\mathcal{S}) := \max_j \frac{\mathbf{y}^\top \left(\mathbf{I}_n - \mathbf{P}(\mathcal{S})\right) \mathbf{a}_j}{\|\mathbf{a}_j\|_2 - \|\mathbf{P}(\mathcal{S})\mathbf{a}_j\|_2}, \tag{2.5}$$

while OMP performs greedy selection according to

$$\max_j \tilde{g}_j(\mathcal{S}) := \max_j \frac{\mathbf{y}^\top \left(\mathbf{I}_n - \mathbf{P}(\mathcal{S})\right) \mathbf{a}_j}{\|\mathbf{a}_j\|_2}. \tag{2.6}$$

For $\mathbf{A} \sim \mathcal{N}\left(0, \frac{1}{n}\right)$, the performance of OMP and OLS is nearly identical since both $\|\mathbf{a}_j\|_2$ and $\|\mathbf{a}_j\|_2 - \|\mathbf{P}(\mathcal{S})\mathbf{a}_j\|_2$ are concentrated around 1 [42].

2.3 Weak Submodular Optimization

In this section, we provide an overview of weak submodular optimization that is the pillar of the results in Chapter 5.

Definition 2.3.1. *Set function $f : 2^{\mathcal{X}} \to \mathbb{R}$ is monotone non-decreasing if $f(\mathcal{S}) \leq f(\mathcal{T})$ for all $\mathcal{S} \subseteq \mathcal{T} \subseteq \mathcal{X}$.*

Definition 2.3.2. *Set function $f : 2^{\mathcal{X}} \to \mathbb{R}$ is submodular if*

$$f(\mathcal{S} \cup \{j\}) - f(\mathcal{S}) \geq f(\mathcal{T} \cup \{j\}) - f(\mathcal{T}) \tag{2.7}$$

for all subsets $\mathcal{S} \subseteq \mathcal{T} \subset \mathcal{X}$ and $j \in \mathcal{X} \backslash \mathcal{T}$. The term $f_j(\mathcal{S}) = f(\mathcal{S} \cup \{j\}) - f(\mathcal{S})$ is the marginal value of adding element j to set $\mathcal{S}$.

Definition 2.3.3. *The multiplicative curvature of a monotone non-decreasing function f is defined as*

$$c_f = \max_{(\mathcal{S},\mathcal{T},i) \in \tilde{\mathcal{X}}} f_i(\mathcal{T}) / f_i(\mathcal{S}), \tag{2.8}$$

where $\tilde{\mathcal{X}} = \{(\mathcal{S}, \mathcal{T}, i) | \mathcal{S} \subseteq \mathcal{T} \subset \mathcal{X}, i \in \mathcal{X} \backslash \mathcal{T}\}$.

The multiplicative curvature [43,44] is a closely related concept to submodularity and essentially quantifies how close the set function is to being submodular. A set function with bounded curvature is called weak submodular. It is worth noting that a set function $f(\mathcal{S})$ is submodular if and only if its multiplicative curvature satisfies $c_f \leq 1$ [45–47].

A similar notion of weak submodularity is the additive curvature defined below [43].

Definition 2.3.4. *The additive curvature of a monotone non-decreasing function f is defined as*

$$\epsilon_f = \max_{(\mathcal{S},\mathcal{T},i)\in\tilde{\mathcal{X}}} f_i(\mathcal{T}) - f_i(\mathcal{S}), \tag{2.9}$$

where $\tilde{\mathcal{X}} = \{(\mathcal{S},\mathcal{T},i)|\mathcal{S}\subseteq\mathcal{T}\subset\mathcal{X}, i\in\mathcal{X}\backslash\mathcal{T}\}$.

Note that when $f(\mathcal{S})$ is submodular, its additive curvature satisfies $\epsilon_f \leq 0$. Multiplicative and additive curvatures are closely related to submodularity ratio [45].

For the above additive and multiplicative curvatures, we have the following proposition [22, 43, 48–50].

Proposition 2.3.1. *Let c_f and ϵ_f be the multiplicative and additive curvatures of $f(\mathcal{S})$, a monotone non-decreasing function with $f(\emptyset) = 0$. Let $\mathcal{S}$ and $\mathcal{T}$ be any subsets such that $\mathcal{S}\subset\mathcal{T}\subseteq\mathcal{X}$ with $|\mathcal{T}\backslash\mathcal{S}| = r$. Then, it holds that*

$$f(\mathcal{T}) - f(\mathcal{S}) \leq \frac{1}{r}\left(1 + (r-1)c_f\right)\sum_{j\in\mathcal{T}\backslash\mathcal{S}} f_j(\mathcal{S}), \tag{2.10}$$

and

$$f(\mathcal{T}) - f(\mathcal{S}) \leq (r-1)\epsilon_f + \sum_{j \in \mathcal{T}\backslash\mathcal{S}} f_j(\mathcal{S}). \tag{2.11}$$

Proof. First note that we can define c_f equivalently as $c_f = \max_{l=1}^{n-1} C_l$ where

$$C_l = \max_{(\mathcal{S},\mathcal{T},i)\in\mathcal{X}_l} f_i(\mathcal{T})/f_i(\mathcal{S}), \tag{2.12}$$

and $\mathcal{X}_l = \{(\mathcal{S},\mathcal{T},i) | \mathcal{S} \subseteq \mathcal{T} \subset \mathcal{X}, i \in \mathcal{X}\backslash\mathcal{T}, |\mathcal{T}\backslash\mathcal{S}| = l\}$. Now, let $\mathcal{S} \subset \mathcal{T}$ and $\mathcal{T}\backslash\mathcal{S} = \{j_1, \ldots, j_r\}$. Then,

$$\begin{aligned} f(\mathcal{T}) - f(\mathcal{S}) &= f(\mathcal{S} \cup \{j_1, \ldots, j_r\}) - f(\mathcal{S}) \\ &= f_{j_1}(\mathcal{S}) + f_{j_2}(\mathcal{S} \cup \{j_1\}) + \ldots \\ &\quad + f_{j_r}(\mathcal{S} \cup \{j_1, \ldots, j_{r-1}\}). \end{aligned} \tag{2.13}$$

Applying (2.12) yields

$$\begin{aligned} f(\mathcal{T}) - f(\mathcal{S}) &\leq f_{j_1}(\mathcal{S}) + \mathcal{C}_1 f_{j_2}(\mathcal{S}) + \cdots + \mathcal{C}_{r-1} f_{j_r}(\mathcal{S}) \\ &= f_{j_1}(\mathcal{S}) + \sum_{l=1}^{r-1} C_l f_{j_t}(\mathcal{S}). \end{aligned} \tag{2.14}$$

Note that (2.14) is invariant to the ordering of elements in $\mathcal{T}\backslash\mathcal{S}$. In fact, it is straightforward to see that given ordering $\{j_1, \ldots, j_r\}$, one can choose a set $\mathcal{Q} = \{\mathcal{P}_1, \ldots, \mathcal{P}_r\}$ with r permutations – e.g., by defining the right circular-shift operator $\mathcal{P}_t(\{j_1, \ldots, j_r\}) = \{j_{r-t+1}, \ldots, j_1, \ldots\}$ for $1 \leq t \leq r$ – such that $\mathcal{P}_p(j) \neq \mathcal{P}_q(j)$ for $p \neq q$ and $\forall j \in \mathcal{T}\backslash\mathcal{S}$. Hence, (2.14) holds for r such

permutations. Summing all of these r inequalities we obtain

$$\begin{aligned} f(\mathcal{T}) - f(\mathcal{S}) &\leq \frac{1}{r}\left(1 + \sum_{l=1}^{r-1} C_l\right) \sum_{j \in \mathcal{T}\backslash\mathcal{S}} f_j(\mathcal{S}) \\ &\leq \frac{1}{r}\left(1 + (r-1)c_f\right) \sum_{j \in \mathcal{T}\backslash\mathcal{S}} f_j(\mathcal{S}). \end{aligned} \tag{2.15}$$

Next, we prove the second inequality. Note that we can define $\epsilon_f = \max_{l=1}^{n-1} \epsilon_l$ where $\epsilon_l = \max_{(\mathcal{S},\mathcal{T},i)\in\mathcal{X}_l} f_i(\mathcal{T}) - f_i(\mathcal{S})$. Using a similar argument as the one that we used for c_f, for any $\mathcal{S} \subset \mathcal{T}$ and $\mathcal{T}\backslash\mathcal{S} = \{j_1, \ldots, j_r\}$, it holds that

$$\begin{aligned} f(\mathcal{T}) - f(\mathcal{S}) &\leq \sum_{l=1}^{r-1} \epsilon_l + \sum_{j \in \mathcal{T}\backslash\mathcal{S}} f_j(\mathcal{S}) \\ &\leq (r-1)\epsilon_f + \sum_{j \in \mathcal{T}\backslash\mathcal{S}} f_j(\mathcal{S}), \end{aligned} \tag{2.16}$$

which completes the proof. ■

Definition 2.3.5. *Let $\mathcal{X}$ be a finite set and let $\mathcal{I}$ denote a collection of subsets of $\mathcal{X}$. The pair $\mathcal{M} = (\mathcal{X}, \mathcal{I})$ is a matroid if the following two statements hold:*

- *Hereditary property. If $\mathcal{T} \in \mathcal{I}$, then $\mathcal{S} \in \mathcal{I}$ for all $\mathcal{S} \subseteq \mathcal{T}$.*
- *Augmentation property. If $\mathcal{S}, \mathcal{T} \in \mathcal{I}$ and $|\mathcal{S}| < |\mathcal{T}|$, then there exists $e \in \mathcal{T}\backslash\mathcal{S}$ such that $\mathcal{S} \cup \{e\} \in \mathcal{I}$.*

The collection $\mathcal{I}$ is called the set of independent sets of the matroid $\mathcal{M}$. A maximal independent set is a basis. It is easy to show that all the bases of a matroid have the same cardinality.

Algorithm 1 Greedy subset Selection

Input: Utility function $f(\mathcal{S})$, set of all observations $\mathcal{X}$, number of selected observations K.
Output: Subset $\mathcal{S}^g \subseteq \mathcal{X}$ with $|\mathcal{S}^g| = K$.
Initialize $\mathcal{S}^g = \emptyset$
for $i = 0, \ldots, K-1$ **do**
    $j_s = \text{argmax}_{j \in \mathcal{X} \setminus \mathcal{S}^g} f_j(\mathcal{S}^g)$
    $\mathcal{S}^g \leftarrow \mathcal{S}^g \cup \{j_s\}$
end for
return $\mathcal{S}^g$.

Given a monotone non-decreasing set function $f : 2^{\mathcal{X}} \to \mathbb{R}$ with $f(\emptyset) = 0$, and a uniform matroid $\mathcal{M} = (X, \mathcal{I})$, we are interested in solving the combinatorial problem

$$\max_{\mathcal{S} \in \mathcal{I}} f(\mathcal{S}). \tag{2.17}$$

It has been shown that finding an optimal solution to (2.17) is generally NP-hard [51]. To this end, efficient heuristic approaches that rely on a simple greedy search (see Algorithm 1) are developed. If the set function $f(\mathcal{S})$ is monotone, Algorithm 1 has a guaranteed lower bound on its achievable performance as stated in Proposition 2.3.2 [22, 43, 48–50].

Proposition 2.3.2. *Let c_f and ϵ_f be the multiplicative and additive curvatures of $f(\mathcal{S})$, a monotone non-decreasing function with $f(\emptyset) = 0$. Let $\mathcal{S}^g \subseteq \mathcal{X}$ with $|\mathcal{S}^g| \leq K$ be the subset selected when maximizing $f(\mathcal{S})$ subject to a cardinality constraint via the greedy observation selection scheme, and let $\mathcal{S}^\star$ denote the optimal subset. Then*

$$f(\mathcal{S}^g) \geq \left(1 - e^{-\frac{1}{c}}\right) f(\mathcal{S}^\star), \tag{2.18}$$

where $c = \max\{c_f, 1\}$ *and*

$$f(\mathcal{S}^g) \geq \left(1 - \frac{1}{e}\right)(f(\mathcal{S}^\star) - (k-1)\epsilon_f). \tag{2.19}$$

Proof. The proof follows the classical proof of greedy maximization of submodular functions given in [48]. We first prove the performance bound stated in terms of c_f. Consider $\mathcal{S}_i$, the set generated at the end of the i^{th} iteration of the greedy algorithm and assume $|\mathcal{S}^\star \backslash S_t^{(i)}| = r \leq k$. Employing Proposition 2.3.1 with $\mathcal{S} = \mathcal{S}_i$ and $\mathcal{T} = \mathcal{S}^\star \cup \mathcal{S}_i$, and using monotonicity of f yields

$$\begin{aligned} \frac{f(\mathcal{S}^\star) - f(\mathcal{S}_i)}{\frac{1}{r}(1 + (r-1)c_f)} &\leq \frac{f(\mathcal{S}^\star \cup \mathcal{S}_i) - f(\mathcal{S}_i)}{\frac{1}{r}(1 + (r-1)c_f)} \\ &\leq \sum_{j \in \mathcal{S}^\star \backslash \mathcal{S}_i} f_j(\mathcal{S}_i) \\ &\leq r(f(\mathcal{S}_{i+1}) - f(\mathcal{S}_i)), \end{aligned} \tag{2.20}$$

where we use the fact that the greedy algorithm selects the element with the maximum marginal gain in each iteration. It is easy to verify, e.g., by taking the derivative, that $\frac{1}{r}(1 + (r-1)c_f)$ is decreasing (increasing) with respect to r if $c_f < 1$ ($c_f > 1$). Let $c = \max\{c_f, 1\}$. Then $\frac{1}{r}(1 + (r-1)\mathcal{C}_{\max}) \leq c$. Therefore, using the fact that $r \leq k$ we get

$$f(\mathcal{S}^\star) - f(\mathcal{S}_i) \leq ck(f(\mathcal{S}_{i+1}) - f(\mathcal{S}_i)). \tag{2.21}$$

By induction and due to the fact that $f(\emptyset) = 0$ we obtain

$$f(\mathcal{S}_g) \geq \left(1 - \left(1 - \frac{1}{kc}\right)^k\right) f(\mathcal{S}^\star) \geq \left(1 - e^{-\frac{1}{c}}\right) f(\mathcal{S}^\star), \tag{2.22}$$

where we use the fact that $(1+x)^y \leq e^{xy}$ for $y > 0$. The proof of second inequality is almost identical except we employ the second result of Proposition 2.3.1 to begin the proof. ■

In Chapter 5, we resort to (2.17) to propose novel observation selection schemes in networked systems.

2.4 Decentralized Optimization

One of the major bottleneck of centralized optimization algorithms is the high communication cost, especially in large-scale settings. Compared to a centralized optimization framework, decentralized optimization enables locality of data storage and model updates which in turn offers computational advantages by delegating computations to multiple clients, and further promotes preservation of privacy of user information [35].

In the standard decentralized optimization setup [52], n clients, each having a local function $f_i(.)$, aim to collaboratively reach $\mathbf{x}^* \in \mathcal{X}^* \subset \mathbb{R}^d$, an optimizer of the following optimization problem

$$\min_{\mathbf{x}} \left[f(\mathbf{x}) := \sum_{i=1}^{n} f_i(\mathbf{x}) \right]. \tag{2.23}$$

Problem (2.23) can be written equivalently as [39, 52–54]

$$\min_{\mathbf{x}_1=\cdots=\mathbf{x}_n} \left[F(\mathbf{X}) := \sum_{i=1}^{n} f_i(\mathbf{x}_i) \right], \tag{2.24}$$

where $\mathbf{x}_i \in \mathbb{R}^d$ is the vector collecting the local parameters of client i, and $\mathbf{X} \in \mathbb{R}^{d\times n}$ is a matrix having $\mathbf{x}_i$ as its i^{th} column. Therefore, the goal of the agents in the network is to achieve consensus such that $\mathbf{x}_i = \mathbf{x}^*$ for some $\mathbf{x}^* \in \mathcal{X}^*$; in matrix notation, $\mathbf{X} = \mathbf{X}^*$, where all the columns of $\mathbf{X}^*$ are equal to $\mathbf{x}^*$, i.e. $\mathbf{X}^* = \mathbf{x}^*\mathbf{1}^\top$.

To solve (2.24), each client can communicate only with its neighbors, where the communication in the network is modeled by a graph. Specifically, we assume each node i associates a non-negative weight w_{ij} to any node j in the network, and $w_{ij} > 0$ if and only if node j can communicate with node i, and $w_{ii} > 0$ for all i. Let $\mathbf{W} = [w_{ij}] \in [0,1]^{n\times n}$ be the matrix that collects these weights. We call $\mathbf{W}$ the mixing or gossip matrix and state some its properties (following [55]) below.

Assumption 2.4.1. (**Mixing Matrix**) *The gossip matrix* $\mathbf{W} = [w_{ij}] \in [0,1]^{n\times n}$ *associated with a connected graph is non-negative, symmetric and doubly stochastic, i.e.*

$$\mathbf{W} = \mathbf{W}^\top, \qquad \mathbf{W}\mathbf{1} = \mathbf{1}. \tag{2.25}$$

Under this condition, eigenvalues of $\mathbf{W}$ *can be shown to satisfy* $1 = |\lambda_1(\mathbf{W})| > |\lambda_2(\mathbf{W})| \geq \cdots \geq |\lambda_n(\mathbf{W})|$ *[55]. Furthermore,* $\delta := 1 - |\lambda_2(\mathbf{W})| \in (0,1]$ *is the so-called spectral gap of* $\mathbf{W}$.

A frequently used choice for $\mathbf{W}$ is to set $w_{ij} = 1/\max\{\deg(i), \deg(j)\}$ [55]. A large spectral gap implies a faster convergence rate of decentralized algorithms. When the graph is fully connected and $\deg(i) = n$, with $\mathbf{W} = \mathbf{1}\mathbf{1}^\top/n$, it holds that $\delta = 1$ which in turn implies consensus can be achieved exactly after one iteration of message passing.

Designing the communication network and its associated mixing matrix $\mathbf{W}$ with a large spectral gap is an important task and an active area of research in multi-agent systems [52,55,56] which is beyond the scope of this dissertation.

Here, we make the common assumption that $\mathbf{W}$ and its spectral gap δ are known and can be used as inputs of our proposed algorithm.

We now define some commonly assumed properties of the objective function.

Assumption 2.4.2. (**Smoothness**) *A function f_i is L_i-smooth, if*

$$f_i(\mathbf{x}) \leq f_i(\mathbf{y}) + (\mathbf{x}-\mathbf{y})^\top \nabla f_i(\mathbf{y}) + \frac{L_i}{2}\|\mathbf{x}-\mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d. \tag{2.26}$$

Assumption 2.4.3. (**strong convexity**) A function f_i is μ_i-strongly convex if

$$f_i(\mathbf{x}) \geq f_i(\mathbf{y}) + (\mathbf{x}-\mathbf{y})^\top \nabla f_i(\mathbf{y}) + \frac{\mu_i}{2}\|\mathbf{x}-\mathbf{y}\|^2, \qquad \mu_i > 0. \tag{2.27}$$

We will find it useful to define $L := \sum_i L_i/n$, $\hat{L} := \max_i L_i$, $\mu := \sum_i \mu_i/n$ and $\hat{\mu} := \max_i \mu_i$.

Assumption 2.4.4. (**Polyak-Lojasiewicz Condition**) *A function f satisfies the Polyak-Lojasiewicz condition (PLC) with parameter μ, if*

$$\|\nabla f(\mathbf{x})\|^2 \geq 2\mu(f(\mathbf{x}) - f^*), \quad \mu > 0, \quad f^* = \min_{\mathbf{x}} f(\mathbf{x}), \quad \forall \mathbf{x} \in \mathbb{R}^d. \tag{2.28}$$

The Polyak-Lojasiewicz condition implies that when multiple global optima exist, each stationary point of the objective function is a global optimum [57,58]. This setting enables studies of modern large-scale ML tasks that are generally nonconvex. It is worth noting that μ-strongly convex functions satisfy PLC with parameter μ – thus, PLC is a weaker assumption than strong convexity.

Convergence of centralized gradient descent under PLC follows a very simple analysis [57, 58]. However, in decentralized settings with compression, analysis of the existing algorithms, e.g. [36, 39, 53], relies on co-coercivity of strongly convex objectives (see Theorem 2.1.11 in [59]). Unfortunately, the results of such analysis do not generalize to PLC settings. In Chapter 6, by performing a novel and simple convergence analysis, we establish convergence of DeLi-CoCo for decentralized nonconvex problems with compressed communication under PLC.

Finally, we characterize the compression operator $\mathcal{C}$ that we use in our algorithm. The following assumption is standard and has been previously made by [39, 60, 61].

Assumption 2.4.5. (**Contraction Compression**) *The compression operator $\mathcal{C}$ satisfies*

$$\mathbb{E}_{\mathcal{C}}\left[\|\mathcal{C}(\mathbf{x})-\mathbf{x}\|^2 \mid \mathbf{x}\right] \leq (1-\omega)\|\mathbf{x}\|^2, \quad \forall \mathbf{x} \in \mathbb{R}^d, \quad 0<\omega \leq 1, \tag{2.29}$$

where the expectation is over the internal randomness of $\mathcal{C}$.

Note that $\mathcal{C}$ can be a biased or an unbiased compression operator including:

- Random selection of k out of d coordinates or k coordinates with the largest magnitudes. In this case $\omega = k/d$ [60]. We denote these two by rand(ω) and top(ω), respectively.

- Setting $\mathcal{C}(\mathbf{x}) = \mathbf{x}$ with probability p and $\mathcal{C}(\mathbf{x}) = 0$ otherwise. In this case $\omega = p$ [39]. We denote this by rand2(ω).

- b-bit random quantization (i.e., the number of quantization levels is 2^b)

$$\text{qsgd}_b(\mathbf{x}) = \frac{\text{sign}(\mathbf{x})\|\mathbf{x}\|}{2^b w} \left\lfloor 2^b \frac{|\mathbf{x}|}{\|\mathbf{x}\|} + \mathbf{u} \right\rfloor, \quad \text{qsgd}_b(\mathbf{0}) = \mathbf{0} \tag{2.30}$$

  where $w = 1 + \min\{\sqrt{d}/2^b, d/2^{2b}\}$ and $\mathbf{u} \sim [0,1]^d$. In this case, $\omega = 1/w$ [62].

Chapter 3

Greedy Algorithms for Sparse Reconstruction

Sparse reconstruction and sparse support selection, i.e., the tasks of inferring an arbitrary m-dimensional sparse vector $\mathbf{x}$ having k nonzero entries from n measurements of linear combinations of its components, are often encountered in machine learning, computer vision, and signal processing. In this chapter, we study large-scale sparse reconstruction problems where we aim to design efficient schemes that entail lower computational costs compared to existing methods while at the same time maintain a favorable and provable reconstruction performance. To this end, swe propose two new iterative schemes, namely accelerated orthogonal least-squares (AOLS) and progressive stochastic greedy (PSG). AOLS aims to decrease the cost of reconstruction via incurring fewer number of iterations while PSG aims to decrease the cost each iteration compared to existing greedy schemes, e.g. OLS and OMP. We further consider application of the proposed methods in clustering high-dimensional data lying on the union of low-dimensional subspaces and demonstrate its superiority over existing methods. The content of this chapter was published in [42, 63].[1]

[1]This chapter is based on existing publication: [Hashemi, Abolfazl, and Haris Vikalo. Accelerated orthogonal least-squares for large-scale sparse reconstruction. Digital Signal

3.1 Introduction

The task of sparse reconstruction and support selection that we introduced in Chapter 2 is encountered in a number of settings in machine learning and signal processing including sparse linear regression [7, 64, 65], compressed sensing [8], image processing [9], subspace clustering [3], column subset selection [10], group testing [11], and graph signal processing [66]. The common goal in such problems is to identify the support (i.e., the collection of nonzero components) of a high-dimensional data vector such as an image or a signal that has sparse representation in a certain domain. The identification relies on noisy measurements of random linear combinations of the sparse vector's components.

Finding the optimal solution to sparse reconstruction or support selection is in general an NP-hard problem; this in turn motivates the design of efficient approximation algorithms and studying the conditions under which exact identification of the optimal subset is possible. In a series of prominent papers, Candes et al. [67–69] show that in order to find the support of a k-sparse m-dimensional vector with overwhelming probability, the information-theoretic lower bound on *sample complexity*, i.e., the minimum number of measurements, is $\mathcal{O}(k \log \frac{m}{k})$. Additionally, they develop an approximation algorithm based on linear programming and ℓ_1 minimization known as basis pursuit (BP) – which shares similarities with LASSO [70] – that achieves the

Processing 82 (2018): 91-105.] The author of this dissertation is the primary contributor. Prof. Vikalo aided in editing the paper and supervised the work.

optimal sample complexity. BP however incurs a computational complexity which is often prohibitive in settings where one deals with high-dimensional and large-scale data.

The OMP and OLS algorithms [40,41,71] that we introduced in Chapter 2 are more efficient alternatives to BP and LASSO.

Recently, necessary and sufficient conditions for exact reconstruction of sparse signals using OMP have been established. Examples of such results include analysis under Restricted Isometry Property (RIP) [72–74], and recovery conditions based on Mutual Incoherence Property (MIP) and Exact Recovery Condition (ERC) [75–77]. For the case of random measurements, performance of OMP was analyzed in [78, 79]. Tropp et al. in [78] showed that in the noise-free scenario, $\mathcal{O}(k \log m)$ measurements is adequate to recover k-sparse m-dimensional signals with high probability. In [80], this result was extended to the asymptotic setting of noisy measurements in high signal-to-noise ratio (SNR) under the assumption that the entries of $\mathbf{A}$ are i.i.d Gaussian and that the length of the unknown vector approaches infinity. Recently, the asymptotic sampling complexity of OMP and GOMP is improved to $\mathcal{O}(k \log \frac{m}{k})$ in [81] and [82], respectively.

Recently, performance of OLS was analyzed in the sparse signal recovery settings with deterministic coefficient matrices. In [83], OLS was analyzed in the noise-free scenario under Exact Recovery Condition (ERC), first introduced in [75]. Herzet et al. [84] provided coherence-based conditions for sparse recovery of signals via OLS when the nonzero components of $\mathbf{x}$ obey certain de-

cay conditions. In [85], sufficient conditions for exact recovery are stated when a subset of true indices is available. In [86] an extension of OLS that employs the idea of [87, 88] and identifies multiple indices in each iteration is proposed and its performance is analyzed under RIP. However, all the existing analysis and performance guarantees for OLS pertain to non-random measurements and cannot directly be applied to random coefficient matrices. For instance, the main results in the notable work [81] relies on the assumption of having dictionaries with ℓ_2-norm normalized columns while this obviously does not hold in the scenarios where the coefficient matrix is composed of entries that are drawn from a Gaussian distribution.

Motivated by the need for fast and accurate sparse recovery in large-scale setting, in this chapter we propose two efficient sparse reconstruction and support selection algorithms. First, we propose the accelerated OLS algorithm that efficiently exploits recursive relation between components of the optimal solution to the original ℓ_0-constrained least-squares problem (2.2). AOLS, similar to GOMP [88] and MOLS [86] exploits the observation that columns having strong correlation with the current residual are likely to have strong correlation with residuals in subsequent iterations; this justifies selection of multiple columns in each iteration and formulation of an overdetermined system of linear equation having solution that is generally more accurate than the one found by OLS or OMP. However, compared to MOLS, our proposed algorithm is orders of magnitude faster and thus more suitable for high-dimensional data applications.

We theoretically analyze the performance of the proposed AOLS algorithm and, by doing so, establish conditions for the exact recovery of the sparse vector $\mathbf{x}$ from measurements $\mathbf{y}$ in (2.1) when the entries of the coefficient matrix $\mathbf{A}$ are drawn at random from a Gaussian distribution – the first such result under these assumptions for an OLS-based algorithm.

Next, we propose the progressive stochastic greedy (PSG) algorithm, the first greedy algorithm with quasilinear $\mathcal{O}(m \log^2 k)$ computational complexity that attains the information-theoretic lower bound $\mathcal{O}(k \log \frac{m}{k})$ on sample complexity. The proposed algorithm builds upon OMP to iteratively identify the optimal support with the following major difference: in each iteration the *search space* of the greedy approach is randomly restricted to significantly reduce the number of oracle calls. The size of these restricted random search spaces follows a strictly increasing sequence to ensure that the search is successful with an overwhelmingly high probability. We further argue the necessity of increasing search space, guaranteed worst-case performance of the proposed scheme, and present its application to the task of column subset selection.

Finally, to further demonstrate efficacy of the proposed techniques, we consider applications of the proposed algorithms to the task of sparse subspace clustering (SSC) and the problem of column subset selection that are often encountered in machine learning and computer vision.

3.2 Accelerated Orthogonal Least Squares Algorithm

In Chapter 2 we introduced OLS. It can be shown, see, e.g., [64, 83], that the index selection criterion (2.5) of OLS can alternatively be expressed as

$$j_s = \arg\max_{j \in \mathcal{I} \setminus \mathcal{S}^{(i-1)}} \left| \mathbf{r}_{i-1}^{\top} \frac{\mathbf{P}(\mathcal{S}^{(i-1)})^{\perp} \mathbf{a}_j}{\|\mathbf{P}(\mathcal{S}^{(i-1)})^{\perp} \mathbf{a}_j\|_2} \right|, \tag{3.1}$$

where $\mathbf{r}_{i-1}$ denotes the residual vector in the i^{th} iteration. Moreover, projection matrix needed for the subsequent iteration is related to the current projection matrix according to

$$\mathbf{P}_{i+1}^{\perp} = \mathbf{P}(\mathcal{S}^{(i)})^{\perp} - \frac{\mathbf{P}(\mathcal{S}^{(i)})^{\perp} \mathbf{a}_{j_s} \mathbf{a}_{j_s}^{\top} \mathbf{P}(\mathcal{S}^{(i)})^{\perp}}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp} \mathbf{a}_{j_s}\|_2^2}. \tag{3.2}$$

It should be noted that $\mathbf{r}_{i-1}$ in (3.1) can be replaced by $\mathbf{y}$ because of the idempotent property of the projection matrix,

$$\mathbf{P}(\mathcal{S}^{(i)})^{\perp} = \mathbf{P}(\mathcal{S}^{(i)})^{\perp^{\top}} = \mathbf{P}(\mathcal{S}^{(i)})^{\perp^2}. \tag{3.3}$$

This substitution reduces complexity of OLS although, when sparsity level k is unknown, the norm of $\mathbf{r}_i$ still needs to be computed since it is typically used when evaluating a stopping criterion.

The complexity of the OLS and its existing variants such as MOLS [86] is dominated by the so-called identification and update steps where the algorithm evaluates projections $\mathbf{P}(\mathcal{S}^{(i-1)})^{\perp}\mathbf{a}_j$ of not-yet-selected columns onto the space spanned by the selected ones and then computes the projection matrix $\mathbf{P}_i$ needed for the next iteration. This becomes practically infeasible in applications that involve dealing with high-dimensional data, including sparse

subspace clustering. To this end, in Theorem 3.2.1 below, we establish a set of recursions which significantly reduce the complexity of the identification and update steps without sacrificing the performance. AOLS then relies on these efficient recursions to identify the indices corresponding to nonzero entries of $\mathbf{x}$ with a significantly lower computational costs with respect to OLS and MOLS. This is further verified in our simulation studies.

Theorem 3.2.1. *Let $\mathbf{r}_i$ denote the residual vector in the i^{th} iteration of OLS with $\mathbf{r}_0 = \mathbf{y}$. The identification step (i.e., step 1 in Algorithm 1) in the $(i+1)^{st}$ iteration of OLS can be rephrased as*

$$j_s = \arg\max_{j \in \mathcal{I} \backslash \mathcal{S}^{(i)}} \left\| \mathbf{q}_j \right\|_2, \tag{3.4}$$

where

$$\mathbf{q}_j \triangleq \frac{\mathbf{a}_j^\top \mathbf{r}_i}{\mathbf{a}_j^\top \mathbf{t}_j^{(i)}} \mathbf{t}_j^{(i)}, \quad \mathbf{t}_j^{(i+1)} \triangleq \mathbf{a}_j - \sum_{l=1}^{i} \frac{\mathbf{a}_j^\top \mathbf{u}_l}{\left\| \mathbf{u}_l \right\|_2^2} \mathbf{u}_l = \mathbf{t}_j^{(i)} - \frac{{\mathbf{t}_j^{(i)}}^\top \mathbf{u}_i}{\left\| \mathbf{u}_i \right\|_2^2} \mathbf{u}_i, \tag{3.5}$$

where $\mathbf{t}_j^{(0)} = \mathbf{a}_j$ for all $j \in \mathcal{I}$. Furthermore, the residual vector $\mathbf{r}_{i+1}$ required for the next iteration is formed as

$$\mathbf{u}_{i+1} \triangleq \mathbf{q}_{j_s}, \quad \mathbf{r}_{i+1} = \mathbf{r}_i - \mathbf{u}_{i+1}. \tag{3.6}$$

Proof. Assume that column $\mathbf{a}_{j_s}$ is selected in the $(i+1)^{\text{st}}$ iteration of the algorithm. Define $\bar{\mathbf{q}}_j = \frac{\mathbf{P}_{(\mathcal{S}^{(i)})^\perp} \mathbf{a}_j}{\left\| \mathbf{P}_{(\mathcal{S}^{(i)})^\perp} \mathbf{a}_j \right\|_2^2} \mathbf{a}_j^\top \mathbf{r}_i$, $\forall j \in \mathcal{I} \backslash \mathcal{S}^{(i)}$. Therefore, by using the

definition of $\bar{\mathbf{q}}_j$,

$$\begin{aligned}\arg\max_{j\in\mathcal{I}\setminus\mathcal{S}^{(i)}} \|\bar{\mathbf{q}}_j\|_2 &= \arg\max_{j\in\mathcal{I}\setminus\mathcal{S}^{(i)}} \|\frac{\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_j}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_j\|_2^2}\mathbf{a}_j^{\top}\mathbf{r}_i\|_2 \\ &= \arg\max_{j\in\mathcal{I}\setminus\mathcal{S}^{(i)}} \frac{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_j\|_2}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_j\|_2^2}|\mathbf{a}_j^{\top}\mathbf{r}_i| \\ &= \arg\max_{j\in\mathcal{I}\setminus\mathcal{S}^{(i)}} \frac{|\mathbf{a}_j^{\top}\mathbf{r}_i|}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_j\|_2}.\end{aligned} \tag{3.7}$$

The idempotent property of $\mathbf{P}(\mathcal{S}^{(i)})^{\perp}$ and the fact that

$$\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{r}_i = \mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{y} = \mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{y} = \mathbf{r}_i \tag{3.8}$$

imply that the last line in (3.7) leads to the same index selection as the OLS rule (3.1). That is,

$$\frac{|\mathbf{a}_j^{\top}\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{r}_i|}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_j\|_2} = \frac{|\mathbf{a}_j^{\top}\mathbf{r}_i|}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_j\|_2}. \tag{3.9}$$

Therefore, $j_s = \arg\max_{j\in\mathcal{I}\setminus\mathcal{S}^{(i)}} \|\bar{\mathbf{q}}_{j_s}\|_2$. Let us post-multiply both sides of (3.2) with the observation vector $\mathbf{y}$, leading to

$$\mathbf{P}_{i+1}^{\perp}\mathbf{y} = \mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{y} - \frac{\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}\mathbf{a}_{j_s}^{\top}\mathbf{P}(\mathcal{S}^{(i)})^{\perp}}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}\|_2^2}\mathbf{y}. \tag{3.10}$$

Recall that $\mathbf{r}_i = \mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{y}$, implying that

$$\mathbf{r}_{i+1} = \mathbf{r}_i - \frac{\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}\|_2^2}\mathbf{a}_{j_s}^{\top}\mathbf{r}_i = \mathbf{r}_i - \bar{\mathbf{q}}_{j_s}. \tag{3.11}$$

Comparing the above expression with (3.6), to complete the proof one needs to show that $\mathbf{q}_{j_s} = \bar{\mathbf{q}}_{j_s}$; this, in turn, is equivalent to demonstrating $\frac{\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}\|_2^2} = \frac{1}{\mathbf{a}_{j_s}^{\top}\mathbf{t}_j^{(i)}}\mathbf{t}_j^{(i)}$. Since $\mathbf{A}$ is full rank, the selected columns are linearly independent.

Let $\{\widetilde{\mathbf{a}}_l\}_{l=1}^{i}$ denote the collection of columns selected in the first i iterations and let $\mathcal{L}_i = \{\widetilde{\mathbf{a}}_1, \ldots, \widetilde{\mathbf{a}}_i\}$ denote the subspace spanned by those columns. Consider the orthogonal projection of the selected column $\mathbf{a}_{j_s}$ onto $\mathcal{L}_i$, $\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}$. Clearly, $\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s} = \mathbf{a}_{j_s} - \mathbf{P}(\mathcal{S}^{(i)})\mathbf{a}_{j_s}$. Noting the idempotent property of $\mathbf{P}(\mathcal{S}^{(i)})^{\perp}$ and the fact that $\|\mathbf{a}_{j_S}\|_2^2 = \|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}\|_2^2 + \|\mathbf{P}(\mathcal{S}^{(i)})\mathbf{a}_{j_s}\|_2^2$, we obtain

$$\frac{\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}}{\|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_{j_s}\|_2^2} = \frac{\mathbf{a}_{j_s} - \mathbf{P}(\mathcal{S}^{(i)})\mathbf{a}_{j_s}}{\mathbf{a}_{j_s}^{\top}\left(\mathbf{a}_{j_s} - \mathbf{P}(\mathcal{S}^{(i)})\mathbf{a}_{j_s}\right)}. \tag{3.12}$$

Hence, in order to show step 1 of OLS algorithm can equivalently be replaced by (3.4)-(3.6), we need to demonstrate that $\mathbf{P}(\mathcal{S}^{(i)})\mathbf{a}_{j_s} = \sum_{l=1}^{i} \frac{\mathbf{a}_{j_s}^{\top}\mathbf{u}_l}{\|\mathbf{u}_l\|_2^2}\mathbf{u}_l$. That is, the collection of vectors $\{\mathbf{u}_l\}_{l=1}^{i}$ constructed by (3.5) and (3.6) is an orthogonal basis for $\mathcal{L}_i$. To this end, we employ an inductive argument. Consider $\mathbf{u}_1$ and $\mathbf{u}_2$ associated with the 1st and 2nd iterations. Using the relations and definitions given in (3.5) and (3.6),

$$\mathbf{u}_1 = \frac{\widetilde{\mathbf{a}}_1^{\top}\mathbf{r}_0}{\|\widetilde{\mathbf{a}}_i\|_2^2}\widetilde{\mathbf{a}}_1, \tag{3.13}$$

$$\mathbf{u}_2 = \frac{\widetilde{\mathbf{a}}_2^{\top}(\mathbf{r}_0 - \mathbf{u}_1)}{\widetilde{\mathbf{a}}_2^{\top}\left(\widetilde{\mathbf{a}}_2 - \frac{\widetilde{\mathbf{a}}_2^{\top}\mathbf{u}_1}{\|\mathbf{u}_1\|_2^2}\mathbf{u}_1\right)}\left(\widetilde{\mathbf{a}}_2 - \frac{\widetilde{\mathbf{a}}_2^{\top}\mathbf{u}_1}{\|\mathbf{u}_1\|_2^2}\mathbf{u}_1\right). \tag{3.14}$$

It is straightforward to see that $\widetilde{\mathbf{a}}_1^{\top}\left(\widetilde{\mathbf{a}}_2 - \frac{\widetilde{\mathbf{a}}_2^{\top}\mathbf{u}_1}{\|\mathbf{u}_1\|_2^2}\mathbf{u}_1\right) = 0$; therefore, $\mathbf{u}_1^{\top}\mathbf{u}_2 = 0$. Now, a collection of orthogonal columns $\{\mathbf{u}_l\}_{l=1}^{i-1}$ forms a basis for $\mathcal{L}_{i-1}$. It follows from (3.5) that

$$\mathbf{u}_i = \frac{\widetilde{\mathbf{a}}_i^{\top}(\mathbf{r}_{i-2} - \mathbf{u}_{i-1})}{\widetilde{\mathbf{a}}_i^{\top}\left(\widetilde{\mathbf{a}}_i - \sum_{l=1}^{i-1}\frac{\widetilde{\mathbf{a}}_i^{\top}\mathbf{u}_l}{\|\mathbf{u}_1\|_2^2}\mathbf{u}_l\right)}\left(\widetilde{\mathbf{a}}_i - \sum_{l=1}^{i-1}\frac{\widetilde{\mathbf{a}}_i^{\top}\mathbf{u}_l}{\|\mathbf{u}_1\|_2^2}\mathbf{u}_l\right). \tag{3.15}$$

Consider $\mathbf{u}_l^{\top}\mathbf{u}_i$ for any $l \in \{1, \ldots, i-1\}$. Since the collection $\{\mathbf{u}_l\}_{l=1}^{i-1}$ is orthogonal, $\mathbf{u}_l^{\top}\mathbf{u}_i$ is proportional to $\widetilde{\mathbf{a}}_l^{\top}\left(\widetilde{\mathbf{a}}_i - \frac{\widetilde{\mathbf{a}}_i^{\top}\mathbf{u}_l}{\|\mathbf{u}_1\|_2^2}\mathbf{u}_l\right)$, which is readily shown to be

zero. Consequently, $\{\mathbf{u}_l\}_{l=1}^{i}$ is an orthogonal basis for $\mathcal{L}_i$ and the orthogonal projection of $\mathbf{a}_{j_s}$ is formed as the Euclidean projection of $\mathbf{a}_{j_s}$ onto each of the orthogonal vectors $\mathbf{u}_l$. Therefore, $\mathbf{P}(\mathcal{S}^{(i)})\mathbf{a}_{j_s} = \sum_{l=1}^{i} \frac{\mathbf{a}_{j_s}^{\top}\mathbf{u}_l}{\|\mathbf{u}_l\|_2^2}\mathbf{u}_l$ and hence $\{\mathbf{u}_l\}_{l=1}^{i}$ is an orthogonal basis for $\mathcal{L}_i$. Using a similar inductive argument one can show that $\mathbf{t}_j^{(i+1)} = \mathbf{t}_j^{(i)} - \frac{\mathbf{t}_j^{(i)^{\top}}\mathbf{u}_i}{\|\mathbf{u}_i\|_2^2}\mathbf{u}_i$, hence demonstrating that step 1 of OLS is equivalent to (3.4)-(3.6); this completes the proof of the theorem. ■

The geometric interpretation of the recursive equations established in Theorem 3.2.1 is stated in Corollary 3.2.1.1. Intuitively, after orthogonalizing selected columns, a new column is identified and added it to the subset thus expanding the corresponding subspace.

Corollary 3.2.1.1. *Let $\{\widetilde{\mathbf{a}}_l\}_{l=1}^{i}$ denote the set of columns selected in the first i iterations of the OLS algorithm and let $\mathcal{L} = \{\widetilde{\mathbf{a}}_1, \ldots, \widetilde{\mathbf{a}}_i\}$ be the subspace spanned by these columns. Then $\{\mathbf{u}_l\}_{l=1}^{i}$ generated according to Theorem 3.2.1 forms an orthogonal basis for $\mathcal{L}_i$.*

Selecting multiple indices per iteration was first proposed in [87, 88] and shown to improve performance while reducing the number of OMP iterations. However, since selecting multiple indices increases computational cost of each iteration, relying on OMP/OLS identification criterion (as in, e.g., [86]) does not necessarily reduce the complexity and may in fact be prohibitive in practice, as we will demonstrate in our simulation results. Motivated by this observation, we rely on recursions derived in Theorem 3.2.1 to develop a novel, computationally efficient variant of OLS that we refer to as Accelerated OLS

(AOLS) and formalize it as Algorithm 2. The proposed AOLS algorithm starts with $\mathcal{S}_0 = \emptyset$ and, in each step, selects $1 \leq L \leq \lfloor \frac{n}{k} \rfloor$ columns of matrix $\mathbf{A}$ such that their normalized projections onto the orthogonal complement of the subspace spanned by the previously chosen columns have higher correlation with the residual vector than remaining non-selected columns. That is, in the i^{th} iteration, AOLS identifies L indices $\{s_1, \ldots, s_L\} \subset \mathcal{I} \backslash \mathcal{S}_{i-1}$ corresponding to the L largest terms $\|\mathbf{q}_j\|_2^2$. After such indices are identified, AOLS employs (3.6) to repeatedly update the residual vector required for consecutive iterations. Note that since in each iteration of AOLS we select L indices, we need to construct L linearly independent vectors $\{\mathbf{u}_{\ell_1}, \ldots, \mathbf{u}_{\ell_L}\}_{\ell=1}^{i}$ in i^{th} iteration. Similarly, to formula to update $\mathbf{t}_j$'s now contains L subtractions. The procedure continues until a stopping criterion (e.g., a predetermined threshold on the norm of the residual vector) is met, or a preset maximum number of iterations is reached.

Remark 3.2.1. We here analyze the worst case computational complexity of AOLS (Algorithm 2). Step 1 requires searching over at most m columns and entails computing inner-product of vectors to find $\|\mathbf{q}_j\|_2$. The overall cost of this step is $\mathcal{O}(mn)$. Step 2 and 3 are variable updates and have constant computational costs. Step 4 requires $\mathcal{O}(Ln)$ operations to update the residual vector. In step 5, we update the $\mathbf{t}_j$'s for $j = 1, \ldots, m$, for the overall cost of $\mathcal{O}(Lnm)$. Finally, in step 6, we solve a least-square problem using the MGS algorithm that costs $\mathcal{O}(L^2nk)$. If there are at most $\ell \leq k$ iterations, the total cost of algorithm 2 is $\mathcal{O}(mn\ell + Ln\ell + Lmn\ell + L^2nk\ell) = \mathcal{O}(Lmn\ell + L^2nk\ell)$. Note that, as confirmed by our simulation results, when the number

of measurements is large compared to the sparsity level, the total number of iterations is significantly lower than k and the overall cost is approximately $\mathcal{O}(Lmnk)$, i.e., it is linear in k. However, if k is relatively large, more iterations of AOLS are required and the complexity can be approximated by $\mathcal{O}(Lmnk + L^2nk^2)$, i.e., the complexity is quadratic in k.

Remark 3.2.2. As we show in our simulation results, performance of AOLS matches that of the MOLS algorithm. However, AOLS is much faster and more suitable for real-world applications involving high-dimensional signals. In particular, the worst case computational costs of Algorithm 1 and MOLS are $\mathcal{O}\left(mn^2k\right)$ and $\mathcal{O}\left(Lmn^2k + L^2nk^2\right)$, respectively; therefore, AOLS is significantly less complex than the conventional OLS and MOLS algorithms.

Algorithm 2 Accelerated Orthogonal Least-Squares (AOLS)

Input: $\mathbf{y}$, $\mathbf{A}$, sparsity level k, threshold ϵ, $1 \leq L \leq \lfloor \frac{n}{k} \rfloor$

Output: recovered support $\mathcal{S}_k$, estimated signal $\hat{\mathbf{x}}_k$

Initialize: $i = 0$, $\mathcal{S}^{(i)} = \emptyset$, $\mathbf{r}_i = 0$, $\mathbf{t}_j^{(i)} = \mathbf{a}_j$, $\mathbf{q}_j = \frac{\mathbf{a}_j^\top \mathbf{r}_i}{\mathbf{a}_j^\top \mathbf{t}_j^{(i)}} \mathbf{t}_j^{(i)}$ for all $j \in \mathcal{I}$.

while $\|\mathbf{r}_i\|_2 \geq \epsilon$ and $i < k$

1. Select $\{j_{s_1}, \ldots, j_{s_L}\}$ corresponding to L largest terms $\|\mathbf{q}_j\|_2$
2. $i \leftarrow i + 1$
3. $\mathcal{S}^{(i)} = \mathcal{S}^{(i-1)} \cup \{j_{s_1}, \ldots, j_{s_L}\}$
4. Perform (3.6) L times to update $\{\mathbf{u}_{\ell_1}, \ldots, \mathbf{u}_{\ell_L}\}_{\ell=1}^{i}$ and $\mathbf{r}_i$
5. $\mathbf{t}_j^{(i)} = \mathbf{t}_j^{(i-1)} - \sum_{l=1}^{L} \frac{\mathbf{t}_j^{(i-1)^\top} \mathbf{u}_{i_l}}{\|\mathbf{u}_{i_l}\|_2^2} \mathbf{u}_{i_l}$ for all $j \in \mathcal{I} \backslash \mathcal{S}^{(i)}$

end while

6. $\hat{\mathbf{x}} = \mathbf{A}_{\mathcal{S}^{(i)}}^{\dagger} \mathbf{y}$

3.2.1 Analysis of sample complexity

In this section, we first study performance of AOLS in the random measurements and noise-free scenario; specifically, we consider the linear model (2.1) where the elements of $\mathbf{A}$ are drawn from $\mathcal{N}\left(0, \frac{1}{n}\right)$ and $\mathbf{e} = \mathbf{0}$, and derive conditions for the exact recovery via AOLS. Then we generalize this result to the noisy scenario.

The following theorem establishes that when the coefficient matrix consists of entries drawn from $\mathcal{N}(0, 1/n)$ and the measurements are noise-free, AOLS with high probability recovers an unknown sparse vector from the linear combinations of its entries in at most k iterations.

Theorem 3.2.2. *Suppose $\mathbf{x} \in \mathbb{R}^m$ is an arbitrary sparse vector with $k < m$ non-zero entries. Let $\mathbf{A} \in \mathbb{R}^{n \times m}$ be a random matrix with entries drawn independently from $\mathcal{N}(0, 1/n)$. Let Σ denote an event wherein given noiseless measurements $\mathbf{y} = \mathbf{A}\mathbf{x}$, AOLS can recover $\mathbf{x}$ in at most k iterations. Then $\Pr\{\Sigma\} \geq p_1 p_2 p_3$, where*

$$\begin{aligned} p_1 &= \left(1 - 2e^{-(n-k+1)c_0(\epsilon)}\right)^2, \\ p_2 &= 1 - 2(\frac{12}{\delta})^k e^{-nc_0(\frac{\delta}{2})}, \text{ and} \\ p_3 &= \left(1 - \sum_{i=0}^{k-1} e^{-\frac{n}{k-i}\frac{1-\epsilon}{1+\epsilon}(1-\delta)^2}\right)^{m-k-L+1}, \end{aligned} \tag{3.16}$$

for any $0 < \epsilon < 1$ and $0 < \delta < 1$.

Proof. See Appendix A.2. ■

Using the result of Theorem 3.2.2, one can numerically show that AOLS successfully recovers k-sparse $\mathbf{x}$ if the number of measurements is linear in k (sparsity) and logarithmic in $\frac{m}{k+L-1}$.

Corollary 3.2.2.1. *Let $\mathbf{x} \in \mathbb{R}^m$ be an arbitrary k-sparse vector and let $\mathbf{A} \in \mathbb{R}^{n\times m}$ denote a matrix with entries that are drawn independently from $\mathcal{N}(0, 1/n)$; moreover, assume that $n \geq \max\{\frac{6}{C_1} k \log \frac{m}{(k+L-1)\sqrt[3]{\beta}}, \frac{C_2 k + \log \frac{8}{\beta^2}}{C_3}\}$, where $0 < \beta < 1$ and C_1, C_2, and C_3 are positive constants independent of β, n, m, and k. Given noiseless measurements $\mathbf{y} = \mathbf{Ax}$, AOLS can recover $\mathbf{x}$ in at most k iterations with probability of success exceeding $1 - \beta^2$.*

Proof. Let us first take a closer look at p_3. Note that $(1-x)^l \geq 1 - lx$ is valid for $x \leq 1$ and $l \geq 1$; since replacing $k - i$ with k in the expression for p_3 in (3.16) decreases p_3, $k(m-k-L+1) \leq \frac{1}{4}(\frac{m}{k+L-1})^6$ for $m > (k+L-1)^{3/2}$ and we obtain

$$p_3 \geq 1 - \frac{1}{4}(\frac{m}{k+L-1})^6 e^{-C_1 \frac{n}{k}}, \tag{3.17}$$

where $C_1 = \frac{1-\epsilon}{1+\epsilon}(1-\delta)^2 > 0$. Multiplying both sides of (3.17) with p_1 and p_2 and discarding positive higher order terms leads to

$$\Pr\{\Sigma\} \geq 1 - \frac{1}{4}(\frac{m}{k+L-1})^6 e^{-C_1 \frac{n}{k}} - 2e^{\log \frac{12}{\delta} k} e^{-nc_0(\frac{\delta}{2})} - 4e^{c_0(\epsilon)k} e^{-nc_0(\epsilon)}. \tag{3.18}$$

This inequality is readily simplified by defining positive constants

$$C_2 = \max_{0<\epsilon,\delta<1} \{\log \frac{12}{\delta}, c_0(\epsilon)\}, \quad C_3 = \min_{0<\epsilon,\delta<1} \{c_0(\frac{\delta}{2}), c_0(\epsilon)\} \tag{3.19}$$

to

$$\Pr\{\Sigma\} \geq 1 - \frac{1}{4}(\frac{m}{k+L-1})^6 e^{-C_1 \frac{n}{k}} - 6e^{C_2 k} e^{-nC_3}. \tag{3.20}$$

We need to show that $\Pr\{\Sigma\} \geq 1 - \beta^2$. To this end, it suffices to demonstrate that

$$\beta^2 \geq \frac{1}{4}(\frac{m}{k+L-1})^6 e^{-C_1 \frac{n}{k}} + 6e^{C_2 k} e^{-nC_3}. \tag{3.21}$$

Let $n \geq \frac{C_2 k + \log \frac{8}{\beta^2}}{C_3}$.[2] This ensures $6e^{C_2 k} e^{-nC_3} \leq \frac{3\beta^2}{4}$ and thus gives the desired result. Moreover,

$$n \geq \max\{\frac{6}{C_1} k \log \frac{m}{(k+L-1)\sqrt[3]{\beta}}, \frac{C_2 k + \log \frac{8}{\beta^2}}{C_3}\} \tag{3.22}$$

guarantees that $\Pr\{\Sigma\} \geq 1 - \beta^2$ with $0 < \beta < 1$. ■

Remark 3.2.3. Note that when $k \to \infty$ (and so do m and n), p_1, p_2, and p_3 are very close to 1. Therefore, one may assume very small ϵ and δ which implies $C_1 \approx 1$.

We now turn to the general case of noisy random measurements and study the conditions under which AOLS with high probability exactly recovers support of $\mathbf{x}$ in at most k iterations. Note that similar to the noiseless scenario, here the successful recovery is defined as exact support recovery.

Theorem 3.2.3. *Let $\mathbf{x} \in \mathbb{R}^m$ be an arbitrary k-sparse vector and let $\mathbf{A} \in \mathbb{R}^{n \times m}$ denote a matrix with entries that are drawn independently from $\mathcal{N}(0, 1/n)$. Given the noisy measurements $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$ where $\|\mathbf{e}\|_2 \leq \gamma_{\mathbf{e}}$, and $\mathbf{e}$ is independent of $\mathbf{A}$ and $\mathbf{x}$, if $\min_{\mathbf{x}_j \neq 0} |\mathbf{x}_j| \geq (1 + \delta + t)\gamma_{\mathbf{e}}$ for any $t > 0$, AOLS can recover $\mathbf{x}$ in at most k iterations with probability of success $\mathbf{P}\{\Sigma\} \geq p_1 p_2 p_3$*

[2] This implies $n \geq k$ for all m, n, and k.

where

$$
\begin{aligned}
p_1 &= \left(1 - 2e^{-(n-k+1)c_0(\gamma)}\right)^2 \\
p_2 &= 1 - 2(\frac{12}{\delta})^k e^{-nc_0(\frac{\delta}{2})}, \text{ and} \\
p_3 &= \left(1 - \sum_{i=0}^{k-1} e^{-\frac{n\frac{1-\gamma}{1+\gamma}(1-\delta)^4}{k\left[\frac{1}{(k-i)t^2}+(1+\delta)^2\right]}}\right)^{m-k-L+1}
\end{aligned}
\tag{3.23}
$$

for any $0 < \gamma < 1$, $0 < \delta < 1$.

Proof. See Appendix A.3. ■

Remark 3.2.4. If we define SNR $= \frac{\|\mathbf{Ax}\|_2^2}{\|\mathbf{e}\|_2^2}$, the condition $\min_{\mathbf{x}_j \neq 0} |\mathbf{x}_j| \geq (1 + \delta + t)\gamma_{\mathbf{e}}$ implies

$$\text{SNR} \approx k(1 + \delta + t)^2, \tag{3.24}$$

which suggests that for exact support recovery via OLS, SNR should scale linearly with sparsity level.

Corollary 3.2.3.1. *Let* $\mathbf{x} \in \mathbb{R}^m$ *be an arbitrary* k*-sparse vector and let* $\mathbf{A} \in \mathbb{R}^{n \times m}$ *denote a matrix with entries that are drawn independently from* $\mathcal{N}(0, 1/n)$*; moreover, assume that* $n \geq \max\{\frac{6}{C_1} k \log \frac{m}{(k+L-1)\sqrt[3]{\beta}}, \frac{C_2 k + \log \frac{8}{\beta^2}}{C_3}\}$ *where* $0 < \beta < 1$ *and* C_1*,* C_2*, and* C_3 *are positive constants that are independent of* β*,* n*,* m*, and* k*. Given the noisy measurements* $\mathbf{y} = \mathbf{Ax} + \mathbf{e}$ *where* $\mathbf{e} \sim \mathcal{N}(0, \sigma^2)$ *is independent of* $\mathbf{A}$ *and* $\mathbf{x}$*, if* $\min_{\mathbf{x}_j \neq 0} |\mathbf{x}_j| \geq C_4 \|\mathbf{e}\|_2$ *for some* $C_4 > 1$*, AOLS can recover* $\mathbf{x}$ *in at most* k *iterations with probability of success exceeding* $1 - \beta^2$*.*

Proof. The proof follows the steps of the proof to Corollary 3.2.2.1, leading us to constants $C_1 = \frac{1-\gamma}{1+\gamma}(1-\delta)^4(1+t^2(1+\delta)^2)^{-1}$, $C_2 = \max_{0<\gamma,\delta<1}\{\log \frac{12}{\delta}, c_0(\gamma)\} > 0$, $C_3 = \min_{0<\gamma,\delta<1}\{c_0(\frac{\delta}{2}), c_0(\gamma)\} > 0$, and $C_4 = (1 + \delta + t)$. ■

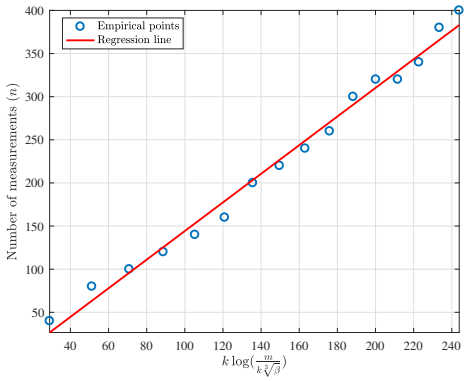


Figure 3.1: Number of noiseless measurements required for sparse reconstruction with $\beta^2 = 0.05$ when $m = 1024$. The regression line is $n = 2.0109\ k\log(\frac{m}{k\sqrt[3]{\beta}})$ with the coefficient of determination $R^2 = 0.9888$.

Remark 3.2.5. In general, for the case of noisy measurements C_1 is smaller than that of the noiseless setting, implying a more demanding sampling requirement for the former.

3.2.2 Numerical experiments

In this section, we verify our theoretical results by comparing them to the empirical ones obtained via Monte Carlo simulations.

First, we consider the results of Corollary 3.2.2.1 with $L = 1$. In each trial, we select locations of the nonzero elements of $\mathbf{x}$ uniformly at random and draw those elements from a normal distribution. Entries of the coefficient matrix $\mathbf{A}$ are also generated randomly from $\mathcal{N}(0, \frac{1}{n})$. Fig. 3.1 plots the number of noiseless measurement n needed to achieve at least 0.95 probability of perfect recovery (i.e., $\beta^2 = 0.05$) as a function of $k\log(\frac{m}{k\sqrt[3]{\beta}})$. The length of the unknown vector $\mathbf{x}$ here is set to $m = 1024$, and the results (shown as circles)

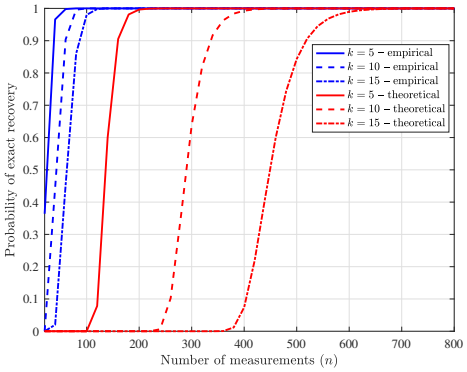


Figure 3.2: A comparison of the theoretical probability of exact recovery provided by Theorem 3.2.2 with the empirical one, where $m = 1024$ and the non-zero elements of $\mathbf{x}$ are drawn independently from a normal distribution.

are averaged over 1000 independent trials. The solid regression line in Fig. 3.1 implies linear relation between n and $k \log(\frac{m}{k\sqrt[3]{\beta}})$ as predicted by Corollary 3.2.2.1. Specifically, for the considered setting, $n \approx 2.0109\ k \log(\frac{m}{k\sqrt[3]{\beta}})$. Recall that, according to Remark 1, for a high-dimensional problem where the exact support recovery has the probability of success overwhelmingly close to 1, $C_1 \approx 1$; this implies $n \geq 6\ k \log(\frac{m}{k\sqrt[3]{\beta}})$ for all m and k. Therefore, Fig. 3.1 suggests that our theoretical result is somewhat conservative (which is due to approximations that we rely on in the proof of Theorem 3.2.2 and Corollary 3.2.2.1).

In Fig. 3.2, we compare the lower bound on probability of exact recovery from noiseless random measurements established in Theorem 3.2.2 with empirical results. In particular, we consider the setting where $L = 1$, $m = 1000$ and the non-zero elements of $\mathbf{x}$ are independent and identically distributed normal random variables. For three sparsity levels ($k = 5, 10, 15$) we vary the

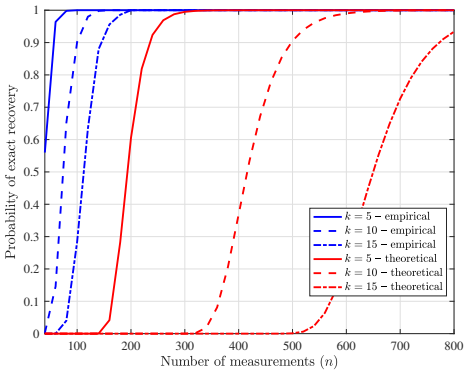
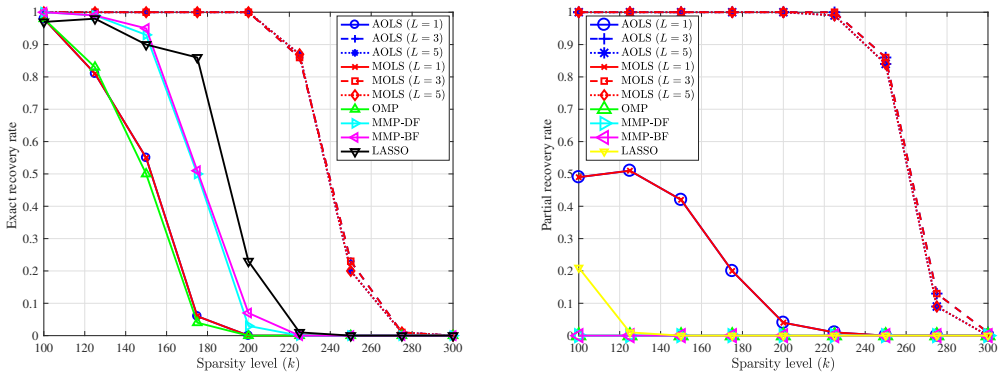


Figure 3.3: A comparison of the theoretical probability of exact recovery provided by Theorem 3.2.3 with the empirical one, where $m = 1024$ and non-zero elements $\mathbf{x}$ are set to $(1 + \delta + 20)\|\mathbf{e}\|_2$.

number of measurements and plot the empirical probability of exact recovery, averaged over 1000 independent instances. Fig. 3.2 illustrates that the theoretical lower bound established in (3.16) becomes more tight as the signal becomes more sparse.

Next, we compare the lower bound on probability of exact recovery from noiseless random measurements established in Theorem 3.2.3 with empirical results. More specifically, $L = 1$, $m = 1000$, $k = 5, 10, 15$, and the non-zero elements of $\mathbf{x}$ are set to $(1+\delta+20)\|\mathbf{e}\|_2$ to ensure that the condition of Theorem 3.2.3 imposed on the smallest nonzero element of $\mathbf{x}$ is satisfied. For this setting, in Fig. 3.3 the results of Theorem 3.2.3 are compared with the empirical ones (the latter are averaged over 1000 independent instances). As can be seen from the figure, the lower bound on probability of successful recovery becomes more accurate for lower k, similar to the results for the noiseless scenario illustrated in Fig. 3.2.

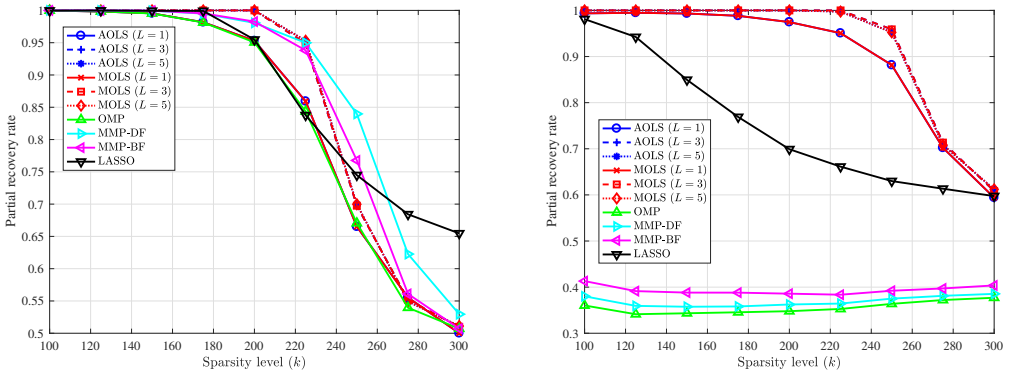


(a) $T = 0$ (b) $T = 0.5$

(c) $T = 1$ (d) $T = 10$

Figure 3.4: Exact recovery rate comparison of AOLS, MOLS, OMP, MMP-DP, MMP-BP, and LASSO for $n = 512$, $m = 1024$, and k non-zero components of $\mathbf{x}$ uniformly drawn from $\mathcal{N}(0, 1)$ distribution.

To evaluate performance of the AOLS algorithm, we compare it with five state-of-the-art sparse recovery algorithms for varied sparsity levels k. In particular, we considered OMP [41], Least Absolute Shrinkage and Selection Operator (LASSO) [67,70], MOLS [86] with $L = 1, 3, 5$, depth-first and breath-first multipath matching pursuit [89] (refered to as MMP-DF and MMP-BP, respectively). It is shown in [86,89] that MOLS, MMP-DF, and MMP-BP out-



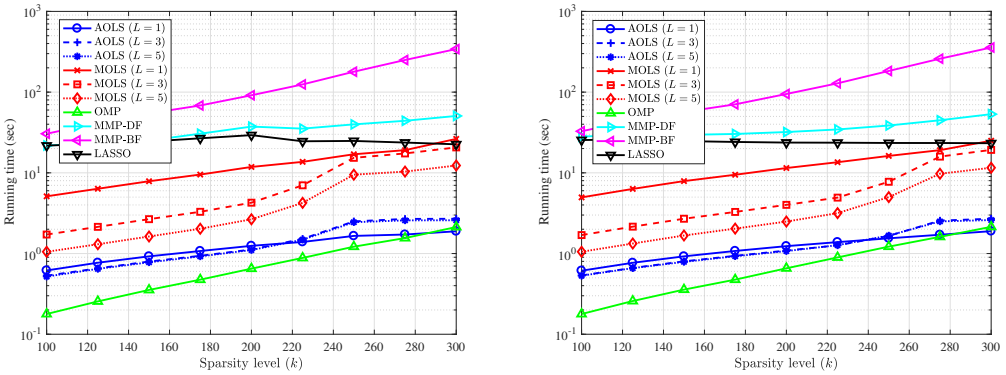
(a) $T = 0$ (b) $T = 0.5$

(c) $T = 1$ (d) $T = 10$

Figure 3.5: Partial recovery rate comparison of AOLS, MOLS, OMP, MMP-DP, MMP-BP, and LASSO for $n = 512$, $m = 1024$, and k non-zero components of $\mathbf{x}$ uniformly drawn from $\mathcal{N}(0, 1)$ distribution.

perform many of the sparse recovery algorithms, including OLS [40], OMP [41], GOMP [88], StOMP [87], and BP [67]. Therefore, to demonstrate performance of AOLS with respect to other sparse recovery methods, we compare it to these three schemes. We also include the performances of OMP and LASSO as baselines.

For MOLS, MMP-DF, and MMP-BF we used the MATLAB implemen-



(a) $T = 0$ (b) $T = 0.5$

(c) $T = 1$ (d) $T = 10$

Figure 3.6: A comparison of AOLS, MOLS, OMP, MMP-DP, MMP-BP, and LASSO for $n = 512$, $m = 1024$, and k non-zero components of $\mathbf{x}$ uniformly drawn from the $\mathcal{N}(0, 1)$ distribution.

tations provided by the authors of [86,89]. To solve the least-square problem in OMP, GOMP, MMP-DF, and MMP-BF we use the MGS algorithm which obtains the solution at low cost. As typically done in benchmarking tests [88,90], we used CVX [91, 92] to implement the LASSO algorithm. We explored various values of L (specifically, $L = 1, 3, 5$) to better understand its effect on the performance of AOLS. When k is known, we run k iterations of OMP and

OLS. In contrast to OMP and OLS, other algorithms considered in this section, AOLS, MMP-DF, MOLS, and LASSO, need a stopping criterion; we set the threshold to 10^{-13}. Note that MMP-BF, a breadth-first algorithm, does not use a stopping threshold.

We consider sparse recovery from random measurements in a large-scale setting to fully understand scalability of tested algorithms. To this end, we set $n = 512$ and $m = 1024$; k changes from 100 to 300. The non-zero elements of $\mathbf{x}$ – whose locations are chosen uniformly – are independent and identically distributed normal random variables. In order to construct $\mathbf{A}$, we consider the so-called hybrid scenario [83] to simulate both correlated and uncorrelated dictionaries. Specifically, we set $\mathbf{A}_j = \frac{\mathbf{b}_j + t_j \mathbf{1}}{\|\mathbf{b}_j + t_j \mathbf{1}\|_2}$ where $\mathbf{b}_j \sim \mathcal{N}(0, \frac{1}{n})$, $t_j \sim \mathcal{U}(0, T)$ with $T \geq 0$, and $\mathbf{1} \in \mathbb{R}^n$ is the all-ones vector. In addition, $\{\mathbf{b}_j\}_{j=1}^m$ and $\{t_j\}_{j=1}^m$ are statistically independent. Notice that as T increases, the so-called mutual coherence parameter of $\mathbf{A}$ increases, resulting in a more correlated coefficient matrix; $T = 0$ corresponds to an incoherent $\mathbf{A}$. For each scenario, we use Monte Carlo simulations with 100 independent instances. Performance of each algorithm is characterized by three metrics: (i) Exact Recovery Rate (ERR), defined as the percentage of instances where the support of $\mathbf{x}$ is recovered exactly, (ii) Partial Recovery Rate (PRR), measuring the fraction of support which is recovered correctly, and (iii) the running time of the algorithm in MATLAB found via *tic* and *toc* commands which are Mathwork's recommended choices for measuring runtimes of different functions.

The exact recovery rate, partial recovery rate, and running time com-

parisons are shown in Fig. 3.4, Fig. 3.5, and Fig. 3.6, respectively. As can be seen from Fig. 3.4, AOLS and MOLS with $L = 3, 5$ achieve the best exact recovery rate for various values of T. We also observe that as T increases, performance of all schemes, except for AOLS and MOLS, significantly deteriorates and they can never exactly recover the support for $T = 1$ and $T = 10$. Note that our theoretical results suggests that in the settings of this experiment, $k \leq 115$ is a sufficient condition for exact recovery with high probability. As for the partial recovery rate shown in Fig. 3.5, for $T = 0$ all methods perform similarly. However, AOLS and MOLS are robust to changes in T while other schemes perform poorly for larger values of T. Running time comparison results, depicted in Fig. 3.6, demonstrate that for all scenarios the AOLS algorithm is essentially as fast as OMP, while AOLS is significantly more accurate. We also observe from the figure that AOLS is significantly faster than other schemes. Specifically, for larger values of k, AOLS is around 15 times faster than MOLS, while they deliver essentially the same performance. Note that a larger L results in a lower running time for both AOLS and MOLS as these schemes find the support of the signal with fewer iterations than k. Since the cost of conventional OLS is relatively high, this gain in speed is more noticeable for MOLS than for AOLS. Moreover, as we discussed in Section 3, as k grows the recovery becomes harder and more iterations are needed. Specifically, we observe quadratic trends in the running times of AOLS and MOLS, where the complexity growth is more pronounced for MOLS and larger L.

Overall, the results depicted in Fig. 3.4, Fig. 3.5, and Fig. 3.6 suggest

that MOLS and AOLS provide the best recovery rate, even when the measurement matrix contains highly-correlated columns. However, AOLS enjoys a running time similar to the low-cost and popular OMP algorithm, and is significantly faster than MOLS.

3.3 Progressive Stochastic Greedy Algorithm

In this section, we propose the Progressive Stochastic Greedy (PSG) algorithm (formalized as Algorithm 3). The algorithm is built upon the idea that in order to identify the optimal support $\mathcal{S}^*$ exactly, the number of oracle calls in each iteration of OMP need not be equal. Specifically, in the early iterations, the search space can be drastically reduced to a small subset that with high probability contains at least one index from $\mathcal{S}^*$. However, in the subsequent iterations, assuming that the algorithm has been accurately identifying indices of the non-zero entries of a sparse vector, the search domain needs to be as large as $O(m)$ to allow the possibility of including an index from $\mathcal{S}^*$. That is, since the goal is to identify exactly all the elements of $\mathcal{S}^\star$, one should *progressively* increase the size of the search set thus improving the probability of success.

To this end, we propose a method which employs the intuitive progression of the search set size. Specifically, in the i^{th} iteration the proposed scheme samples $r_i = \frac{m}{k-i} \log \frac{1}{\epsilon}$ elements uniformly at random from $[m]$ to construct the search set $\mathcal{R}_{psg}^{(i)}$. Here ϵ, selected such that $e^{-k} \leq \epsilon \leq e^{-\frac{k}{m}}$, is a parameter that allows one to strike a desired balance between the perfor-

mance and complexity. It should be noted that in practice the sampling may be with or without replacement. Additionally, since it should hold that $r_i \leq m$ for all $i = 0, \ldots, k-1$, for any iteration i such that $i \geq k - \log \frac{1}{\epsilon}$ we set r_i to its maximum value, m.

Algorithm 3 Progressive Stochastic Greedy (PSG)

Input: Measurements y, sensing matrix $\mathbf{A}$, number of elements to be selected k, search space parameter ϵ.
Output: Subset $\mathcal{S}_{psg} \subseteq [m]$ with $|\mathcal{S}_{psg}| = k$.
Initialize $\mathcal{S}_{psg}^{(0)} = \emptyset$
for $i = 0, \ldots, k-1$
  if $i < k - \log \frac{1}{\epsilon}$ **then**
    $r_i = \frac{m}{k-i} \log \frac{1}{\epsilon}$
  else
    $r_i = m$
  end if
  Construct $\mathcal{R}_{psg}^{(i)}$ by sampling r_i elements uniformly at random from $[m]$.
  $j_s \in \text{argmax}_{j \in \mathcal{R}_{psg}^{(i)}} \frac{\mathbf{y}^\top \left(\mathbf{I}_n - \mathbf{P}(\mathcal{S}_{psg}^{(i)})\right) \mathbf{a}_j}{\|\mathbf{a}_j\|_2}$
  $\mathcal{S}_{psg}^{(i+1)} = \mathcal{S}_{psg}^{(i)} \cup \{j_s\}$
end for
Return $\mathcal{S}_{psg} = \mathcal{S}_{psg}^{(k)}$.

As shown later in this section, under this simple modification one can still provide optimal sample complexity guarantees. Before proceeding to the sample complexity analysis, we examine the computational gain that the proposed method offers.

The complexity of PSG is determined by the sum of the number of function evaluations throughout the iterations of the algorithm. Therefore,

for our choice for r_i, the computational complexity of PSG is

$$\begin{aligned}
&\mathcal{O}\left(\sum_{i=0}^{k-\log\frac{1}{\epsilon}} \frac{m}{k-i}\log\frac{1}{\epsilon} + m\log\frac{1}{\epsilon}\right) \\
&= \mathcal{O}\left(m\log\frac{1}{\epsilon}\left(1+\sum_{i=0}^{k-\log\frac{1}{\epsilon}}\frac{1}{k-i}\right)\right) \\
&= \mathcal{O}\left(m\log\frac{1}{\epsilon}\left(1+H_k-H_{\log\frac{1}{\epsilon}}\right)\right) \\
&= \mathcal{O}\left(m\log(\frac{1}{\epsilon})(\log k - \log(\log\frac{1}{\epsilon})\right),
\end{aligned} \tag{3.25}$$

where H_k denotes the Harmonic series and where we used the fact that $\log(k) < H_k < \log(k)+1$ to obtain the last equality. For instance, if $\epsilon = e^{-k}$ then PSG reduces to OMP with the computational complexity of $\mathcal{O}(mk)$. Further, if $\epsilon = c/k^\ell$ for some positive constants $c > 0$ and $\ell \geq 1$, then the complexity of PSG is $\mathcal{O}(m\log^2 k) = \tilde{\mathcal{O}}(m)$, i.e. quasilinear, as opposed to the $\mathcal{O}(mk)$ computational complexity of OMP. Next, we show that this value of ϵ is in fact sufficient to guarantee that PSG achieves the optimal sampling complexity.

3.3.1 Analysis of sample complexity

As mentioned in Chapter 2, the problem of sparse support selection in (2.3) is NP-hard. Typically, analysis of an approximation scheme for this problem is focused on either (i) establishing nontrivial worst-case approximation factor, or (ii) establishing lower bounds on probability of exact recovery. For sparse support selection, the latter has been the prevalent type of a guarantee, and is also the primary type of analysis in the remainder of this chapter.

Theorem 3.3.1 summarizes the main results of this section. It states that PSG successfully recovers k-sparse $\mathbf{x}$ with high probability as long as the number of measurements is linear in k (sparsity) and logarithmic in $\frac{m}{k}$, achieving the optimal sample complexity established by Candes and Tao [67].

Theorem 3.3.1. *Let $\mathbf{x} \in \mathbb{R}^m$ be an arbitrary sparse vector with k non-zero entries and let $\mathbf{A} \in \mathbb{R}^{n \times m}$ denote a random matrix with entries drawn independently from $\mathcal{N}(0, 1/n)$. Given noiseless measurements $\mathbf{y} = \mathbf{A}\mathbf{x}$, PSG with parameter $e^{-k} \leq \epsilon \leq e^{-\frac{k}{m}}$ finds a solution that satisfies*

$$\Pr\left(\mathcal{S}_{psg} = \mathcal{S}^\star\right) \geq (1-\epsilon)^{k - \log\frac{1}{\epsilon}} \left(1 - c_1 (\frac{m}{k})^{c_2} \exp(-c_3 \frac{n}{k})\right), \tag{3.26}$$

for some positive universal constants c_1, c_2, and c_3. Furtherore, assume that $m > k\sqrt{k}$ and

$$n \geq \max\left(\frac{6}{C_1} k \log \frac{m}{k\sqrt[6]{4\beta}} \ , \ C_2 k\right), \tag{3.27}$$

where $0 < \beta < 1$, and C_1 and C_2 are positive constants independent of β, n, m, and k. Then, PSG with parameter $\epsilon < \frac{\beta}{k}$ can exactly identify the optimal support subset $\mathcal{S}^\star$ with a probability of success exceeding $1 - 2\beta$.

Proof. See Appendix A.4. ■

Theorem 3.3.1 demonstrates that it is possible to achieve optimal sample complexity for the task of sparse support selection with only $\mathcal{O}(m \log^2 k) = \tilde{\mathcal{O}}(m)$ function evaluations; to our knowledge, PSG is the first algorithm capable of reconstructing an arbitrary sparse vector with $\tilde{\mathcal{O}}(m)$ computational

complexity. We note that OMP and its variants achieve the same order of sample complexity while requiring $\mathcal{O}(mk)$ function evaluations in general.

It is worth pointing out that although PSG, OMP, and the convex relaxation methods (i.e. BP and LASSO [8, 67, 70]) all achieve the optimal asymptotic sampling complexity, convex relaxation methods enjoy smaller constants in the sample complexity bound compared to greedy methods.[3] This in turn leads to superior performance of convex relaxation methods for problems with large support size. However, in such regimes convex relaxation methods incur prohibitive computational complexities which may render them impractical.

We extend our results on the noiseless and perfectly sparse signals scenario to the general case of noisy measurements where the goal is to identify the best k-sparse approximation of $\mathbf{x}$.

Theorem 3.3.2. *Let $\mathbf{x} \in \mathbb{R}^m$ be an arbitrary signal vector and denote by $\hat{\mathbf{x}}$ its best k-sparse approximation. Furthermore, let $\mathbf{A} \in \mathbb{R}^{n \times m}$ denote a random matrix with entries drawn independently from $\mathcal{N}(0, 1/n)$. Given noisy measurements $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$ where $\mathbf{e}$ is independent of $\mathbf{A}$, if*

$$\min_{j, \hat{x}_j \neq 0} \hat{x}_j \geq \left[(1 - \sqrt{\frac{k}{n}} - \delta)^{-1} + t \right] \epsilon_{\mathbf{n}} \tag{3.28}$$

for any $t, \delta > 0$, where

$$\epsilon_{\mathbf{n}} := \|\mathbf{e}\|_2 + \sigma_{\max}(\mathbf{A}) \|\mathbf{x} - \hat{\mathbf{x}}\|_2, \tag{3.29}$$

[3] The constant in the bound for PSG is higher than for OMP.

PSG with parameter $e^{-k} \leq \epsilon \leq e^{-\frac{k}{m}}$ *identifies the indices of nonzero entries in* $\hat{x}$ *exactly (i.e. the optimal subset* $\mathcal{S}^*$*), and the approximation error further satisfies*

$$\|\mathbf{x}_{psg} - \hat{\mathbf{x}}\|_2^2 \leq \frac{\epsilon_{\mathbf{n}}}{(1 - \sqrt{\frac{k}{n}} - \delta)^2}, \tag{3.30}$$

with probability exceeding

$$(1-\epsilon)^{k - \log \frac{1}{\epsilon}} \left(\left(1 - 2e^{-c_0(\gamma)n}\right)^m - 2e^{-\delta^2 \frac{n}{2}} \right) q, \tag{3.31}$$

where

$$\left(1 - \sum_{i=0}^{k-1} e^{-\frac{nc_1(\gamma)^2(1-\sqrt{\frac{k}{n}}-\delta)^4}{2k\left[\frac{1}{(k-i)t^2} + (1+\sqrt{\frac{k}{n}}+\delta)^2\right]}} \right)^{(m-k)} . \tag{3.32}$$

Furthermore, assume that $m > k\sqrt{k}$ *and*

$$n \geq \max\left(\frac{6}{C_1'} k \log \frac{m}{k\sqrt[6]{4\beta}} \ , \ C_2' k \right), \tag{3.33}$$

where $0 < \beta < 1$*, and* C_1' *and* C_2' *are positive constants independent of* β*,* n*,* m*, and* k*. Then, PSG with parameter* $\epsilon < \frac{\beta}{k}$ *satisfies* (3.30) *and identifies* $\mathcal{S}^*$ *exactly with probability exceeding* $1 - 2\beta$*.*

Proof. See Appendix A.5. ■

Remark 3.3.1. It is worth to take a closer look at the conditions in (3.28) and (3.30). These conditions impose a SNR constraint in the sense that the smallest entry of the k-sparse approximation of the signal should not be too small compared to the additive noise $\mathbf{e}$ and the the part of signal not present in the k-sparse approximation (i.e. $\mathbf{x} - \hat{\mathbf{x}}$). Note that if the signal is exactly

k-sparse and the measurements are noiseless, $\|\mathbf{x} - \hat{\mathbf{x}}\|_2 = 0$ and $\|\mathbf{e}\|_2 = 0$. Thus, $\epsilon_{\mathbf{n}}$ becomes zero and Theorem 3.3.2 reduces to Theorem thm:sss.

3.3.2 Importance of progression to m evaluations

Implications of sampling $r_i = \frac{m}{k-i} \log \frac{1}{\epsilon}$ elements in the i^{th} iteration of PSG are twofold: first, the search space expands as the algorithm proceeds through iterations, and second, the search space eventually grows to m thus essentially reducing PSG to OMP in the last $\log 1/\epsilon$ iterations of the selection procedure. In Theorem 3.3.3, we formally argue that the observed properties of the search space are required in order to establish sample complexity results and guarantee overwhelmingly high probability of the exact recovery.

Theorem 3.3.3. *Consider a sequence of optimization problems $\mathcal{P}(m, k)$ in (2.4) under an increasingly high-dimensional settings, i.e., where $m, k \to \infty$, $m > k$. Let ALG denote a variant of OMP with a restricted uniform search space $\mathcal{R} \subset [m]$ having cardinality r, i.e., r denotes the number of oracle calls in each iteration of ALG. The following claims hold:*

1. *If there exists $\alpha \in (0, 1)$ such that $r \leq k^{\alpha-1}m$, then the probability that ALG succeeds on $\mathcal{P}(m, k)$ goes to zero, i.e.,*

$$\lim sup_{m,k\to\infty} \Pr\left(\mathcal{S}_{alg}^{(k)} = \mathcal{S}^{\star}\right) = 0. \tag{3.34}$$

2. *If there exists $\alpha_1 \in (0, 1)$ such that $r \leq \alpha_1 m$, then the probability that ALG succeeds on $\mathcal{P}(m, k)$ satisfies*

$$\lim sup_{m,k\to\infty} \Pr\left(\mathcal{S}_{alg}^{(k)} = \mathcal{S}^{\star}\right) \in (\delta_1, \delta_2), \tag{3.35}$$

where δ_1 and δ_2 are positive constants that depend on α_1 such that $0 < \delta_1 < \delta_2 < 0.63$.[4]

Proof. See Appendix A.6. ■

Theorem Theorem 3.3.3 establishes upper bounds on the probability that a variant of OMP with a restricted search space constructed uniformly at random exactly identifies $\mathcal{S}^\star$ in two scenarios: (i) If the size of the search space remains fixed in each iteration of ALG and the algorithm makes $\mathcal{O}(mk^\alpha)$ oracle calls for some $\alpha \in (0, 1)$, then the probability of the exact identification approaches zero as the problem dimension grows. (ii) If the size of the search space remains fixed in each iteration of ALG and strictly less than $[m]$, and the algorithm makes $\mathcal{O}(mk)$ oracle calls, then although the probability of the exact identification does not approach zero, it is not asymptotically one either. An arbitrarily high success probability is a condition that is required to establish any nontrivial sample complexity results (even a suboptimal one). Therefore, the two parts of Theorem Theorem 3.3.3 collectively imply that having an increasing *schedule* of search spaces which ultimately reaches to m is a necessary condition to exactly identify the optimal support $\mathcal{S}^\star$ with high probability.

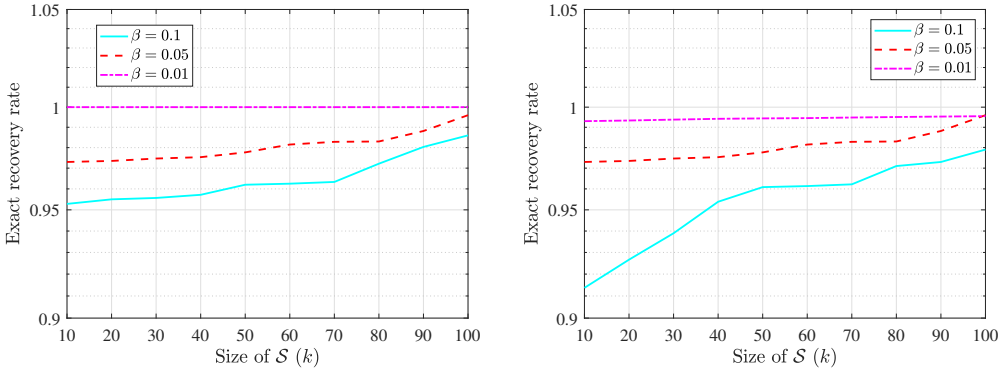
A schedule of search spaces with cardinality $r_i = \frac{m}{k-i} \log \frac{1}{\epsilon}$, explored by PSG, satisfies this necessary condition and hence the result of Theorem

[4]Note that $\mathcal{S}_{alg}^{(k)}$ and $\mathcal{S}^\star$ are quantities that depend on m.

Theorem 3.3.3 does not apply there. We further note that the choice of $r_i = \frac{m}{k-i} \log \frac{1}{\epsilon}$ with $\epsilon = \beta/k$ considered in Theorem 3.3.1 is further appealing since it simplifies the analysis of the lower bound on the sampling complexity.

Finally, note that Theorem Theorem 3.3.3 does not specify the smallest number of oracle calls needed to achieve the optimal sample complexity results established by [67]. However, the theorem paves the way to provide an answer to this important open problem. We note that $\mathcal{O}(m)$ is a trivial lower bound the smallest number of oracle calls. Therefore, PSG achieves this lower bound up to the logarithmic factors. Indeed, it remains of interest to show whether $\mathcal{O}(m \text{ poly}(\log k))$ is in fact the minimum computational cost required to achieve the optimal order of sample complexity.

3.3.3 Numerical experiments

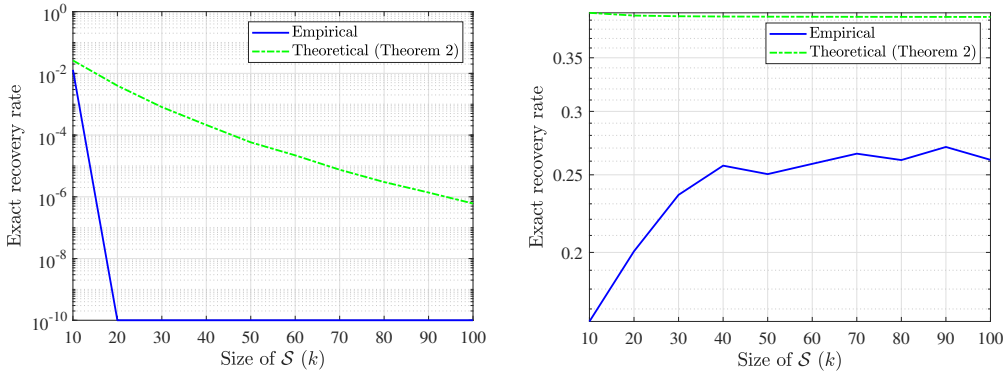


(a) noiseless (Theorem 3.3.1)

(b) noiseless (Theorem 3.3.2)

Figure 3.7: Empirical exact recovery rate of PSG for various values of β.

In this section, we verify our theoretical results by comparing them to the empirical ones obtained via Monte Carlo (MC) simulations. Specifically,



(a) $r = m/\sqrt{k}$ (b) $r = m/2$

Figure 3.8: Empirical evaluation of the theoretical bounds established by Theorem Theorem 3.3.3.

we consider the task of sparse support selection with increasing support size k (varied from 10 to 100). We set the dimension of the signal and the number of measurements to $m = 2k^{1.5}$ and $n = 6k\log(m/k\sqrt[6]{4\beta})$, respectively, for three different values of $\beta = 0.1, 0.05, 0.01$. In each trial, we select locations of the nonzero elements of $\mathbf{x}$ uniformly at random and draw those elements from a normal distribution. Entries of the coefficient matrix $\mathbf{A}$ are also generated randomly from $\mathcal{N}(0, \frac{1}{n})$. The results are averaged over 1000 MC trials.

First, we investigate the exact performance of PSG with the choice of $\epsilon = \beta/k$ for $\beta = 0.1, 0.05, 0.01$, and show the results in Fig. 3.7. As we can see from the figure, the empirical exact recovery rate of PSG is very close to one, which coincides with the theoretical lower bound of $1 - 2\beta$ established in Theorem 3.3.1 and 3.3.2 (i.e, $0.8, 0.9, 0.98$ for $\beta = 0.1, 0.05, 0.01$, respectively).

Next, we empirically verify the results of Theorem Theorem 3.3.3 wherein we established an upper bound on the success probability of a variant of OMP,

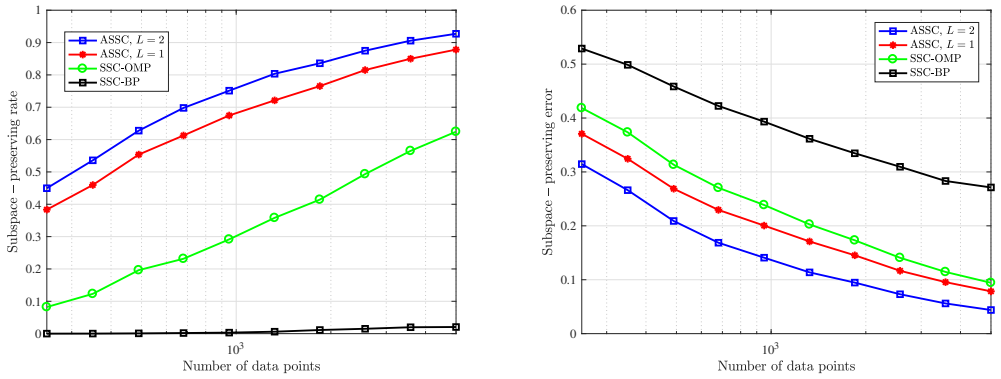
named ALG, with a restricted uniform search space. Fig. 3.8 compares this theoretical result with the empirical success rate for $r = m/\sqrt{k}$ and $r = m/2$, which correspond to instances of the two settings considered in Theorem Theorem 3.3.3. Fig. 3.8(a) shows that for $r = m/\sqrt{k}$ the success rate goes to zero as k increases, as predicted by the first part of Theorem Theorem 3.3.3. [5] In Fig. 3.8(b) we see that the success rate does not go to zero for $r = m/2$; however, it is always bounded by $1 - e^{-0.5} \approx 0.39$, as claimed by the second part of Theorem Theorem 3.3.3.

3.4 Applications

3.4.1 Sparse subspace clustering

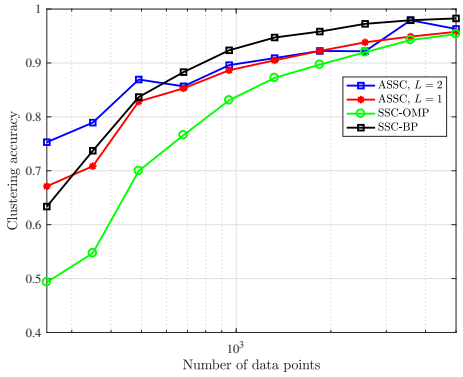
Sparse subspace clustering (SSC), which received considerable attention in recent years, relies on sparse signal reconstruction techniques to organize high-dimensional data known to have low-dimensional representation [3]. In particular, in SSC problems we are given matrix $\mathbf{A}$ which collects data points $\mathbf{a}_i$ drawn from a union of low-dimensional subspaces, and are interested in partitioning the data according to their subspace membership. State-of-the-art SSC schemes such as SSC-OMP [1, 2] and SSC-BP [3, 4] typically consist of two steps. In the first step, one finds a similarity matrix $\mathbf{W}$ characterizing relative affinity of data points by forming a representation matrix $\mathcal{C}$. Once $\mathbf{W} = |\mathcal{C}| + |\mathcal{C}|^\top$ is generated, the second step performs data segmentation by

[5] Note that in this setting, for $k \geq 20$ this variant of OMP failed in all of the trials; however, for illustration purposes (i.e., to be able to show the plot in the logarithmic scale) we set the success rate of ALG for $k \geq 20$ to 10^{-10}.

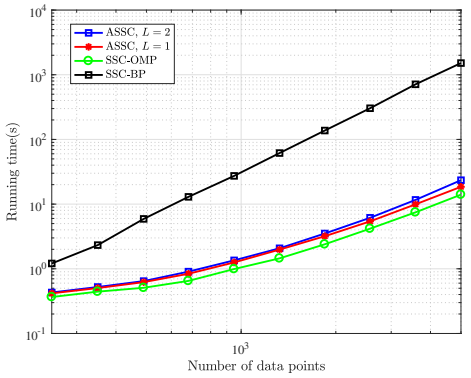


(a) Subspace preserving rate

(b) Subspace preserving error



(c) Clustering accuracy

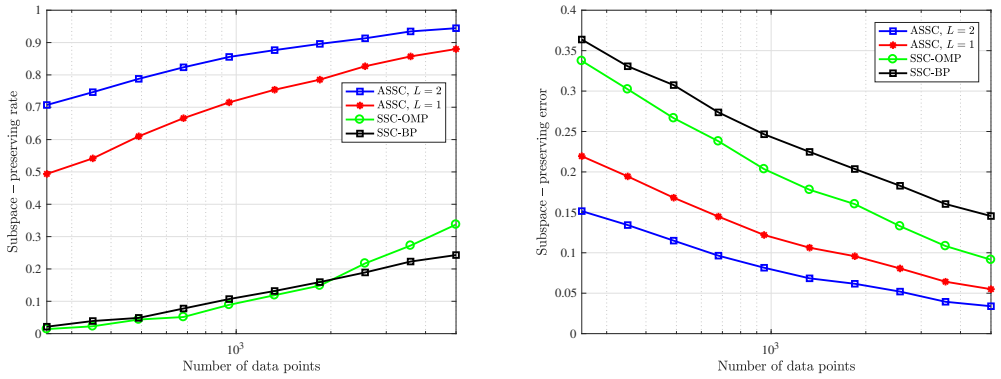


(d) Running time (sec)

Figure 3.9: Performance comparison of ASSC, SSC-OMP [1, 2], and SSC-BP [3, 4] on synthetic data with no perturbation. The points are drawn from 5 subspaces of dimension 6 in ambient dimension 9. Each subspace contains the same number of points and the overall number of points is varied from 250 to 5000.

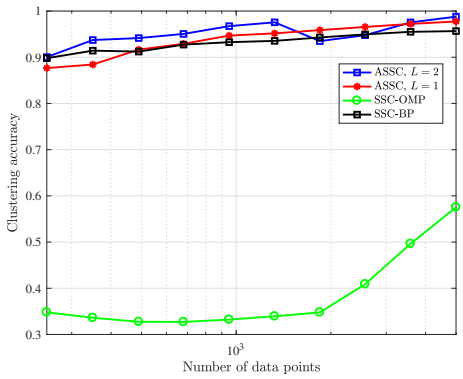
applying spectral clustering [93] on $\mathbf{W}$. Most of the SSC methods rely on the so-called self-expressiveness property of data belonging to a union of subspaces which states that each point in a union of subspaces can be written as a linear combination of other points in the union [3].

In this section, we employ the proposed AOLS algorithm to generate the subspace-preserving similarity matrix $\mathbf{W}$ and empirically compare the re-

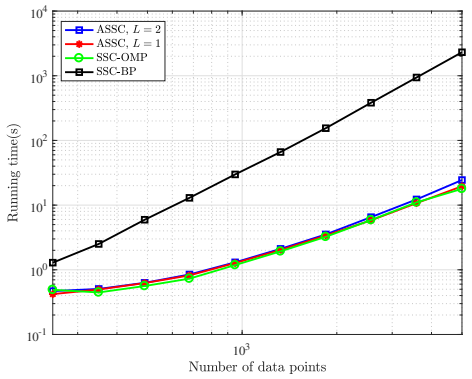


(a) Subspace preserving rate

(b) Subspace preserving error



(c) Clustering accuracy



(d) Running time (sec)

Figure 3.10: Performance comparison of ASSC, SSC-OMP [1, 2], and SSC-BP [3, 4] on synthetic data with perturbation terms $Q \sim \mathcal{U}(0, 1)$. The points are drawn from 5 subspaces of dimension 6 in ambient dimension 9. Each subspace contains the same number of points and the overall number of points is varied from 250 to 5000.

sulting SSC performance with that of SSC-OMP [1,2] and SSC-MP [3,4].[6] For SSC-BP, two implementations based on ADMM and interior point methods are available by the authors of [3, 4]. In our simulation studies we use the ADMM implementation of SSC-BP in [3, 4] as it is faster than the interior

[6]We refer to our proposed scheme for the SSC problem as Accelerated SSC (ASSC).

point method implementation. Our scheme is tested for $L = 1$ and $L = 2$. We consider the following two scenarios: (1) A random model where the subspaces are with high probability near-independent; and (2) The setting where we used hybrid dictionaries [83] to generate similar data points across different subspaces which in turn implies the independence assumption no longer holds. In both scenarios, we randomly generate $n = 5$ subspaces, each of dimension $d = 6$, in an ambient space of dimension $D = 9$. Each subspace contains N_i sample points where we vary N_i from 50 to 1000; therefore, the total number of data points, $N = \sum_{i=1}^{n} N_i$, is varied from 250 to 5000. The results are averaged over 20 independent instances. For scenario (1), we generate data points by uniformly sampling from the unit sphere. For the second scenario, after sampling a data point we add a perturbation term $Q\mathbf{1}_D$ where $Q \sim \mathcal{U}(0, 1)$.

In addition to comparing the algorithms in terms of their clustering accuracy and running time, we use the following metrics defined in [3, 4] that quantify the subspace preserving property of the representation matrix $\mathcal{C}$ returned by each algorithm:

- *Subspace preserving rate:* The fraction of points whose representations are subspace-preserving.

- *Subspace preserving error:* The fraction of ℓ_1 norms of the representation coefficients associated with points from other subspaces, i.e.,

$$\frac{1}{N} \sum_{j} \Big(\sum_{i \in \mathcal{O}} |\mathcal{C}_{ij}| / \|\mathbf{c}_j\|_1 \Big) \tag{3.36}$$

  where $\mathcal{O}$ represents the set of data points from other subspaces.

The results for the scenario (1) and (2) are illustrated in Fig. 3.9 and Fig. 3.10, respectively. As can be seen in Fig. 3.9, ASSC is nearly as fast as SSC-OMP and orders of magnitude faster than SSC-BP while ASSC achieves better subspace preserving rate, subspace preserving error, and clustering accuracy compared to competing schemes. In the second scenario, we observe that the performance of SSC-OMP is severely deteriorated while ASSC still outperforms both SSC-BP and SSC-OMP in terms of accuracy. Further, similar to the first scenario, running time of ASSC is similar to that of SSC-OMP while both methods are much faster that SSC-BP. As Fig. 3.9 and Fig. 3.10 suggest, the ASSC algorithm, especially with $L = 2$, outperforms other schemes while essentially being as fast as the SSC-OMP method.

3.4.2 Column subset selection

Column subset selection (CSS) is a feature selection and dimensionality reduction method that processes high dimensional datasets with the goal of arriving at a succinct yet information-preserving data representation. CSS methods have received considerable attention in recent years due to their efficiency, interpretability, and provably-guaranteed performance [10, 94, 95].

At first glance, CSS is similar to the problem of determining a general low-rank approximation; however, CSS offers some unique advantages. First, since CSS is an unsupervised method and does not require labeled data, in can be efficiently applied to the scenarios where the labeled data is sparse while unlabeled data is abundant. Second, in many applications such as hiring and

education where the decision is made via a data-driven algorithm, it is of critical importance to keep the semantic interpretation of the features intact and thus enable learning interpretable models. This can be ensured by selecting a subset of available features as opposed to generating new features via an arbitrary function of the input features. Finally, compared to PCA or other methods that require matrix-matrix multiplication to project input features onto a reduced space during inference, solutions to the CSS feature selection problem can be directly applied during inference since CSS only requires selecting a subset of feature values from a new instance vector.

Formally, CSS is a constrained low-rank approximation problem that aims to identify a subset $\mathcal{S}$, $|\mathcal{S}| = k$, out of the m columns of a data matrix $\mathbf{D} \in \mathbb{R}^{n \times m}$ that best approximate the entire data matrix. In particular, the task of identifying $\mathcal{S}$ can be cast as the optimization problem

$$\begin{aligned} &\underset{\mathcal{S}}{\text{minimize}} \quad \|\mathbf{D} - \mathbf{P}(\mathcal{S})\mathbf{D}\|_F^2 \\ &\text{s.t.} \quad |\mathcal{S}| = k, \end{aligned} \tag{3.37}$$

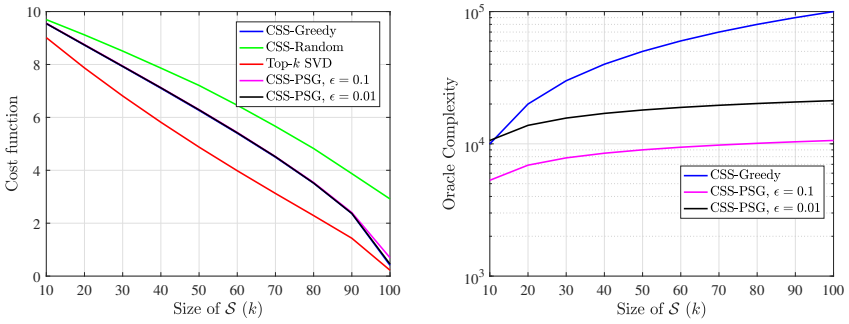
which is similar to the sparse support selection task (2.4).

The optimization task in (3.37) is an NP-hard combinatorial problem. Efficient methods for finding an approximate solution include the greedy algorithm [96] which attempts to find $\mathcal{S}$ in an iterative fashion. In particular, in the i^{th} step the greedy method identifies column j_s of $\mathbf{A}$ according to the selection rule

$$j_s = \arg\max_{j \in [n] \setminus \mathcal{S}_g^{(i-1)}} = \frac{\|\mathbf{d}_j^\top \mathbf{E}_{i-1}\|_2^2}{\|(\mathbf{I} - \mathbf{P}(\mathcal{S}_g^{(i-1)}))\mathbf{d}_j\|_2^2}, \tag{3.38}$$

where $\mathbf{E}_i$ denotes the so-called residual matrix in the i^{th} iteration and $\mathcal{S}^{(i-1)}$ collects indices of columns selected in the first i iterations; these quantities are initialized as $\mathbf{E}_0 = \mathbf{D}$ and $\mathcal{S}_0 = \emptyset$, respectively.

In this section, we explore an application of the proposed PSG method to CSS. Specifically, we apply PSG with two values of $\epsilon = 0.1$ and $\epsilon = 0.01$ to CSS and benchmark it against the greedy and random column subset selection schemes. Additionally, we use the best rank-k approximation of a matrix (i.e., top-k SVD) to serve as an upper bound on the performance, as it explicitly minimizes the Forbenius reconstruction criterion. Note that this method only serves as an upper bound and does not fall into the framework of column subset selection. We compare these algorithms on two datasets; the first one is a simulated dataset where we generate full rank matrices with a small subset of columns approximately spanning the entire column space, and the second one being real dataset.



(a) Reconstruction error (b) Oracle complexity

Figure 3.11: Performance comparison of various CSS schemes and the top-k SVD lower bound on a synthetic data.

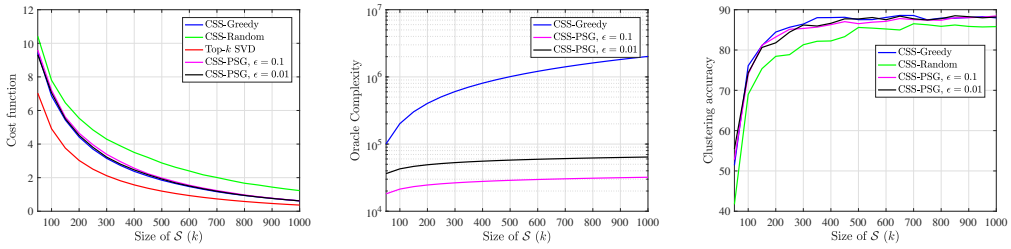
First we compare performance of various CSS schemes on a simulated

Figure 3.12: Face clustering: given images of multiple subjects, the goal is to find images that belong to the same subject (Examples from the EYaleB dataset [5]).

dataset. In each iteration of a Monte Carlo simulation with 1000 independent experiments, we generate a random matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m = 200$, $n = 1000$, as follows. First we generate $\mathbf{A}$ according to $\mathbf{A} \sim \mathcal{N}(0, 1)$, and then ensure that the first 180 columns approximately lie in the span of the last 20 columns. That is, we set $\mathbf{A}_{[180]} = \mathbf{P}_{\mathcal{L}}\mathbf{A}_{[180]} + 0.1 \times \mathbf{P}_{\mathcal{L}}^{\perp}\mathbf{A}_{[180]}$, where $\mathcal{L} = \{181, \ldots, 200\}$. Finally, we normalize all of the columns of $\mathbf{A}$ and then employ our proposed algorithm as well as the benchmarks to select k columns where k varies from 100 to 1000.

Fig. 3.11 shows the results of this experiment. As can be seen in Fig. 3.11(a), the proposed scheme delivers essentially the same reconstruction error (i.e. the same objective value for $f(\mathcal{S})$) as the greedy CSS algorithm; both methods significantly outperform random sampling. As we keep selecting more columns, the reconstruction error decreases. Note that this phenomenon is expected since the objective function of the CSS problem is monotonically increasing. Finally, in Fig. 3.11(b) we compare the number of function evaluations (i.e. oracle complexity) of the greedy and the proposed randomized greedy schemes. As seen in this figure, the proposed schemes incurs considerably lower computational cost compared to the greedy method.

Next, we test the proposed PSG algorithm on the EYaleB dataset [5]



(a) Reconstruction error (b) Oracle complexity (c) Clustering accuracy

Figure 3.13: Performance comparison of various CSS schemes on EYaleB dataset.

which contains frontal face images of 38 individuals acquired under 64 different illumination conditions. There are 2414 columns (i.e., features) in this dataset (see Fig. 3.12). We select k of out these 2414 columns, where k varies from 100 to 1000, and apply the SSC method of [2] to cluster the data points using the selected features.

Fig. 3.13 compares the performance of various column subset selection schemes as well as the top-k SVD approach. As shown in Fig. 3.13(a), reconstruction errors of the greedy and the proposed scheme are nearly identical. As the number of selected columns is increased, the reconstruction error is reduced; this is consistent with the fact that $f(\mathcal{S})$ is a monotone function. Fig. 3.13(b) shows the significant computational complexity improvement provided by the proposed scheme as compared to the greedy CSS method. Since the complexity of Algorithm 3 grows logarithmically with k, the computational overhead due to selecting larger numbers of columns is relatively small. Finally, in Fig. 3.13(c) we compare the clustering accuracy of SSC applied to the subset of features selected by different schemes. As the figure shows, clus-

tering performance of SSC combined with the proposed CSS method is nearly identical to that of the greedy scheme; both are superior to random column subset selection.

3.5 Conclusion

In this chapter, we proposed two new sparse reconstruction algorithms, namely AOLS and PSG.

AOLS, unlike state-of-the art OLS-based schemes such as Multiple Orthogonal Least-Squares (MOLS) [86], relies on a set of expressions which provide computationally efficient recursive updates of the orthogonal projection operator and enable computation of the residual vector by employing only linear equations. Additionally, AOLS allows incorporating L columns in each iteration to further reduce the complexity while achieving improved performance. In our theoretical analysis of AOLS, we showed that for coefficient matrices consisting of entries drawn from a Gaussian distribution, AOLS with high probability recovers k-sparse m-dimensional signals in at most k iterations from $\mathcal{O}(k \log \frac{m}{k+L-1})$ noiseless random linear measurements. We extended this result to the scenario where the measurements are perturbed with ℓ_2-bounded noise.

The second scheme, PSG, achieves the optimal $\mathcal{O}(k \log \frac{m}{k})$ sample complexity established by [67] while requiring only $\mathcal{O}(m \log^2 k)$ oracle calls. PSG is the first greedy algorithm that obtains optimal sample complexity at a quasilinear complexity. The proposed scheme utilizes the idea of randomly

restricting the search space of the greedy support selection methods in a progressive manner thus reducing the computational cost. We further argue the necessity of having a schedule that progressively expands the search space.

Furthermore, we considered applications to sparse subspace clustering where we employed the proposed schemes to facilitate efficient clustering of high-dimensional data points lying on the union of low-dimensional subspaces, showing superior performance compared to state-of-the-art OMP-based and BP-based methods [1–4].

Chapter 4

Evolutionary Subspace Clustering

As we saw in the previous chapter, subspace clustering is an important unsupervised learning problem that deals with clustering a collection of points lying on a union of low-dimensional subspaces. In many applications, in addition to the low-dimensional subspace structure, the data is acquired at multiple points in time. Exploiting the underlying temporal behavior provides more informative description and enables improved clustering accuracy. Therefore, in addition to the union-of-subspaces structure, there exists an underlying evolutionary structure characterizing the temporal aspect of data. Thus, it is of interest to design and investigate frameworks that exploit both union of subspaces and temporal smoothness structures to perform fast and accurate clustering, particularly in real-time applications.

In this chapter, we propose evolutionary subspace clustering, a method whose objective is to cluster a collection of evolving data points that lie on a union of low-dimensional evolving subspaces. we propose to learn a parsimonious representation of the data points at each time step by establishing a non-convex optimization framework that exploits the self-expressiveness property of the evolving data while taking into account representation from the

preceding time step. we develop a scheme based on alternating minimization that both learns the parsimonious representation as well as adaptively tunes and infers a smoothing parameter reflective of the rate of data evolution. The developed framework is successfully employed in a motion segmentation application. The content of this chapter can be found in [97].[1]

4.1 Introduction

Massive amounts of high-dimensional data collected by contemporary information processing systems create new challenges in the fields of signal processing and machine learning. High dimensionality of data presents computational and memory burdens and may adversely affect performance of the existing data analysis algorithms. An important unsupervised learning problem encountered in such settings deals with finding informative parsimonious structures characterizing large-scale high-dimensional datasets. This task is critical for detection of meaningful patterns in complex data and enabling accurate and efficient clustering. The problem of extracting low-dimensional structures for the purpose of clustering is encountered in many applications including motion segmentation and face clustering in computer vision [12,13,98], image representation and compression in image clustering [14,15], robust principal component analysis (PCA), and robust subspace recovery and track-

[1]This chapter is based on existing publication: [Hashemi, Abolfazl, and Haris Vikalo. Evolutionary self-expressive models for subspace clustering. IEEE Journal of Selected Topics in Signal Processing 12.6 (2018): 1534-1546.] The author of this dissertation is the primary contributor. Prof. Vikalo aided in editing the paper and supervised the work.

ing [16–20]. In these settings, the data can be thought of as being a collection of points lying on a union of low-dimensional subspaces. In addition to having such structural properties, data is often acquired at multiple points in time. Exploiting the underlying temporal behavior provides more informative description and enables improved clustering accuracy. For example, it is well-known that feature point trajectories associated with motion in a video lie in an affine subspace [6]. Motion during any given short time interval is related to the motion in recent past. Therefore, in addition to the union of subspaces structure of the video data, there exists an underlying *evolutionary structure* characterizing the motion. Therefore, it is of interest to design and investigate frameworks that exploit both *union of subspaces and temporal smoothness* structures to perform fast and accurate clustering, particularly in real-time applications where a clustering solution is required at each time step.

In this chapter, we formulate and study *evolutionary subspace clustering* – the task of clustering data points that lie on a union of *evolving* subspaces. We provide a mathematical formulation of evolutionary subspace clustering and introduce the *convex evolutionary self-expressive model* (CESM), an optimization framework that exploits the self-expressiveness property of data and learns sparse representations while taking into account prior representations. The task of learning parameters of the CESM leads to a non-convex optimization problem which we solve approximately by relying on the alternating minimization ideas. In the process of learning data representation, we automatically tune a smoothing parameter which characterizes the significance of

prior representations, i.e., quantifies similarity of the representation in successive time steps. The smoothing parameter is reflective of the rate of evolution of the data and signifies the amount of temporal changes in consecutive data snapshots. Note that although we only consider the case of sparse representations, the proposed framework can readily be extended to enforce any structures on the learned representations, including low rank or low rank plus sparse structures that are often encountered in subspace clustering applications. Following extensive simulations on synthetic datasets and two real-world datasets originating from real-time motion segmentation (as opposed to offline motion segmentation considered in, e.g. [3,4]) and oceanography, we demonstrate that the proposed framework significantly improves the performance and shortens runtimes of state-of-the-art *static* subspace clustering algorithms that only exploit the self-expressiveness property of the data.

4.1.1 Connection to subspace clustering

As we discussed in Section 3.4.1, subspace clustering schemes attempt to solve variants of the optimization problem

$$\min_{\mathbf{C}_t} \|\mathbf{C}_t\| \quad \text{s.t.} \quad \|\mathbf{X}_t - \mathbf{X}_t\mathbf{C}_t\|^2 \leq \epsilon, \quad \text{diag}(\mathbf{C}_t) = \mathbf{0}, \tag{4.1}$$

where $\mathbf{X}_t$ and $\mathbf{C}_t$ denote the data and representation matrices at time t, the norm in the objective function is, e.g. $\|\cdot\|_1$, $\|\cdot\|_0$, and $\|\cdot\|_*$ for SSC-BP, SSC-OMP, and LRR schemes, respectively, and ϵ is a predefined threshold that determines to what extend a representation matrix $\mathbf{C}_t$ should preserve self-expressiveness of $\mathbf{X}_t$. One then defines an affinity (or similarity) matrix

$\mathbf{A}_t = |\mathbf{C}_t| + |\mathbf{C}_t|^\top$ and applies spectral clustering [93] to find the clustering solution.

Performance of self-expressiveness-based subspace clustering schemes was analyzed in various settings. It was shown in [3, 4] that when the subspaces are disjoint (independent), the BP-based method is subspace preserving. Authors of [99, 100] take a geometric point of view to further study the performance of BP-based SSC algorithm in the setting of intersecting subspaces and in the presence of outliers. These results are extended to the OMP-based SSC [1, 2] and matching pursuit-based SSC [101].

Recently, further extensions of SSC and LRR frameworks were developed. In particular, an SSC-based approach that jointly performs representation learning and clustering is proposed in [102] while the authors of [103–107] extend the SSC framework to handle datasets with missing information. Time complexity and memory footprint challenges of the LRR framework motivated the development of its online counterpart in [108]. The temporal subspace clustering scheme [109] assumes that one data point is sampled at each time step and sets the goal of grouping the data points into sequential segments, followed by clustering the segments into their respective subspaces. However, neither of these approaches considers the possibility of an evolutionary structure in the *feature space*, the setting studied in the current chapter. Instead, prior works assume that the data points are received in an online fashion (as opposed to having evolving features) and, once acquired, are fixed and do not evolve with time. Therefore, just as the original SSC and LRR frameworks, the subsequent

variants of subspace clustering can be categorized as being *static*. In contrast, the evolutionary subspace clustering problem studied in the current chapter is focused on improving clustering quality by judiciously combining parsimonious representations from multiple time steps while exploiting the union of subspaces structure of the data.

A related problem to subspace clustering is that of robust principal component analysis (PCA) and robust subspace recovery and tracking [16–20]. There, the goal is to identify outliers (which in some applications may actually be the objects of interest) to perform PCA and find a *single* low-dimensional subspace which best fits a collection of points taken from a high-dimensional space. State-of-the-art methods perform this task by decomposing the data matrix into a sum of low rank and sparse matrices. Note that, in robust subspace recovery, the data matrix consists of all the snapshots of data which are assumed to lie on a single subspace (except for outliers). Therefore, this problem, too, is inherently different from the evolutionary subspace clustering framework that we study in the current chapter.

4.1.2 Connection to evolutionary clustering

The topic of evolutionary clustering has attracted significant attention in recent years [110–113]. The problem was originally introduced in [110] where the authors proposed a framework for evolutionary clustering by adding a temporal smoothness penalty to a static clustering objective. Evolutionary extensions of agglomerative hierarchical clustering and k-means were presented as

examples of the general framework. Evolutionary clustering has been applied in a variety of practical settings such as tracking in dynamic networks [112,114] and study of climate change [115], generally improving the performance of static clustering algorithms. Non-parametric Bayesian evolutionary clustering schemes employing hierarchical Dirichlet process are developed in [116–118]. An evolutionary affinity propagation clustering algorithm that relies on message passing between the nodes of an appropriately defined factor graph is developed in [113]. Chi et al. [119, 120] proposed two frameworks for evolutionary spectral clustering referred to as preserving cluster quality (PCQ) and preserving cluster membership (PCM) schemes. In the PCQ formulation, the temporal cost at time t is determined based on the quality of the partition formed using data from time $t-1$; in PCM, the temporal cost is a result of comparing the partition at time t with the partition at $t-1$. The authors of [121] proposed evolutionary extensions of k-means and agglomerative hierarchical clustering by filtering the feature vectors using a finite impulse response filter which combines the measurements of feature vectors and uses them to find an affinity matrix for clustering. Their approach essentially tracks clusters across time by extending the similarity between points and cluster centers to include their positions at previous time steps. However, the method in [121] is limited to the settings where the number of clusters does not change with time. Following the idea of modifying similarities followed by static clustering, Xu et al. proposed AFFECT, an evolutionary clustering method where the matrix indicating similarity between data points at a given time step is

assumed to be the sum of a deterministic matrix (the affinity matrix) and a Gaussian noise matrix [111].

To find clustering solutions at multiple points in time for evolutionary data characterized by union-of-subspaces structure, one might consider concatenating the data snapshots from the first until the current time instance and performing subspace clustering on such a set. In this approach, which we refer to as concatenate-and-cluster (C&C), finding the clustering solution at time t would involve forming the matrix $\bar{\mathbf{X}}_t = [\mathbf{X}_1^\top, \ldots, \mathbf{X}_t^\top]^\top$ and performing clustering $\bar{\mathbf{X}}_t$ via subspace clustering approaches. However, due to a significant increase in the number of features (caused by data concatenation), such a procedure would incur computational complexity that grows with time (depending on the subspace clustering method, the complexity would be either quadratic or cubic in time). Perhaps more importantly, the C&C approach lacks the ability to discover subtle temporal changes in data organization and attempts to fit a clustering solution to a single union-of-subspaces structure; in other words, clustering of concatenated data fails to account for temporal evolution of subspaces.

As an illustrative example, consider the task of *real-time* motion segmentation [122, 123] where the goal is to identify and track motions in a video sequence. Real-time motion segmentation is related to the *offline* motion segmentation task studied in [3, 4]. The difference between the two is that in the offline setting clustering is performed once, after receiving all the frames in the sequence, while in the real-time setting clustering steps are performed

after receiving each snapshot of data (See Section 4.5.2). The subspaces representing the motions evolve; while subspaces in subsequent snapshots are similar, those that are associated with snapshots separated more widely in time may be drastically different. For this reason, imposing a single structure, as in the aforementioned C&C approach, may lead to poor clustering solutions. Therefore, a scheme that judiciously exploit the evolutionary structure while acknowledging the union-of-subspaces structure is needed.

Let $\mathbf{A}_{t-1}$ and $\bar{\mathbf{A}}_t$ denote the affinity matrix at time $t-1$ and the affinity matrix constructed solely from $\mathbf{X}_t$, respectively. State-of-the-art evolutionary clustering algorithms, e.g., [111, 119–121], apply a static clustering algorithm such as spectral clustering to process the following affinity matrix

$$\mathbf{A}_t = \alpha_t \bar{\mathbf{A}}_t + (1 - \alpha_t)\mathbf{A}_{t-1}, \tag{4.2}$$

where α_t is the so-called smoothing parameter at time t. The affinity matrix $\bar{\mathbf{A}}_t$ is typically constructed from $\mathbf{X}_t$ using general similarity notions such as the negative Euclidean distance of the data points or its exponential variant.

The recursive construction of the affinity matrix shown above brings up several questions. First, note that when $\bar{\mathbf{A}}_t$ is determined from (4.2), one does not take into account representation of data points in previous time steps; as we show in our experimental studies, this may lead to poor performance in subspace clustering applications. More importantly, apart from the AFFECT algorithm [111], none of the existing evolutionary clustering schemes provides a procedure for finding the smoothing parameter α_t which determines how

much weight is placed on historic data. Instead, existing methods typically set α_t according to the user's preference for the temporal smoothness of the clustering results. AFFECT relies on an iterative shrinkage estimation approach to automatically tune α_t. However, to find the smoothing parameter, AFFECT makes certain strong assumptions on the structure of the affinity matrix. In particular, it assumes a block structure that holds only if the data at each time t is a realization of a dynamic Gaussian mixture model, which is typically not the case in practice, especially in subspace clustering applications such as motion segmentation. Indeed, as our simulation results demonstrate, typical values of smoothing parameter found by the shrinkage estimation approach of AFFECT in motion segmentation application is $\alpha_t \approx 0.5$ regardless of whether the data is static or evolutionary. This is counterintuitive since, e.g., for static data we expect $\alpha_t \approx 0$.

To address the above challenges, we develop a novel framework for clustering temporal high-dimensional data that contains points lying on a union of low-dimensional subspaces. The proposed framework exploits the self-expressiveness property of data to learn a representation for $\mathbf{X}_t$ while at the same time takes into account data representation learned in the previous time step. Moreover, we propose a novel strategy that relies on alternating minimization to automatically learn the smoothing parameter α_t at each time step. As our extensive simulation results demonstrate, the smoothing parameter inferred by the proposed CESM framework captures temporal behavior and adapts to sudden changes in data. Therefore, the smoothing parameter

found by the proposed framework is reflective of the rate of data evolution and quantifies the significance of prior representations when clustering data at time t. Note that even though in this chapter we focus on evolutionary self-expressive models with sparse representation, the proposed framework can be extended in straightforward manner to include other representation learning frameworks such as LRR.

4.2 Evolutionary Subspace Clustering

Let $\{\mathbf{x}_{t,i}\}_{i=1}^{N_t}$ be a collection of (evolving) real-valued D_t-dimensional data points at time t and let us organize those points in a matrix $\mathbf{X}_t = [\mathbf{x}_{t,1}, \ldots, \mathbf{x}_{t,N_t}] \in \mathbb{R}^{D_t \times N_t}$. The data points are drawn from a union of n_t evolving subspaces $\{\mathcal{S}_{t,i}\}_{i=1}^{n_t}$ with dimensions $\{d_{t,i}\}_{i=1}^{n_t}$. Without a loss of generality, we assume that the columns of $\mathbf{X}_t$, i.e., the data points, are normalized vectors with unit ℓ_2 norm. [2] Due to the underlying union-of-subspaces structure, the data points satisfy the self-expressiveness property [3] formally stated below.

Definition 4.2.1. *A collection of evolving data points $\{\mathbf{x}_{t,i}\}_{i=1}^{N_t}$ satisfies the self-expressiveness property if each data point has a linear representation in terms of the other points in the collection, i.e., there exist a representation matrix $\mathbf{C}_t$ such that*

$$\mathbf{X}_t = \mathbf{X}_t \mathbf{C}_t, \quad \operatorname{diag}(\mathbf{C}_t) = \mathbf{0}. \tag{4.3}$$

The goal of subspace clustering is to partition $\{\mathbf{x}_{t,i}\}_{i=1}^{N_t}$ into n_t groups

[2] As we proceed, for the simplicity of notation we may omit the time index.

such that the data points that belong to the same subspace are assigned to the same cluster. To distinguish between different methods, we refer to subspace clustering schemes that find a representation matrix $\mathbf{C}_t$ which satisfies (4.3) as the *static subspace clustering* methods. As stated in Section 4.1, in many applications the subspaces and the data points lying on the union of those subspaces evolve over time. Imposing the self-expressiveness property helps exploit the fact that the data points belong to a union of subspaces. However, (4.3) alone does not capture potential evolutionary structure of the data. To this end, we propose to find a representation matrix $\mathbf{C}_t$, for each time t, such that

$$\mathbf{C}_t = f_\theta(\mathbf{C}_{t-1}), \quad \mathbf{X}_t = \mathbf{X}_t\mathbf{C}_t, \quad \text{diag}(\mathbf{C}_t) = \mathbf{0}. \tag{4.4}$$

In other words, the representation matrix $\mathbf{C}_t$ is assumed to be a matrix-valued function parametrized by θ that captures the self-expressiveness property of data while also promoting a relation to the representation matrix at a preceding time instance, $\mathbf{C}_{t-1}$. The function $f_\theta : \mathcal{P}_\mathbf{C} \to \mathcal{P}_\mathbf{C}$ may in principle be any appropriate parametric function while the set $\mathcal{P}_\mathbf{C} \subseteq \mathbb{R}^{N\times N}$ stands for any preferred parsimonious structures imposed on the representation matrices at each time instant, e.g., sparse or low-rank representations. We refer to subspace clustering schemes that satisfy (4.4) as *evolutionary subspace clustering* methods. To find such a representation matrix $\mathbf{C}_t$, we formulate and solve the optimization

$$\begin{aligned} \min_{\theta} \quad & \|\mathbf{X}_t - \mathbf{X}_t f_\theta(\mathbf{C}_{t-1})\|^2 \\ \text{s.t.} \quad & f_\theta(\mathbf{C}_{t-1}) \in \mathcal{P}_\mathbf{C}, \end{aligned} \tag{4.5}$$

and use the resulting representation matrix $\mathbf{C}_t = f_\theta(\mathbf{C}_{t-1})$ to segment the data.

The evolutionary subspace clustering problem (4.5) is essentially a general constrained representation learning problem. Given any combination of $(f_\theta, \mathcal{P}_\mathbf{C})$, a solution to (4.5) results in a distinct evolutionary subspace clustering framework. After finding a solution to (4.5) and setting $\mathbf{C}_t = f_{\theta^*}(\mathbf{C}_{t-1})$, we construct an affinity matrix $\mathbf{A}_t = |\mathbf{C}_t| + |\mathbf{C}_t|^\top$ and then apply spectral clustering to $\mathbf{A}_t$.

In this chapter, we restrict our studies to the case where $\mathcal{P}_\mathbf{C}$ is the set of sparse representation matrices and consider a simple and interpretable form of the parametric function f_θ. Other structures and more complex parametric functions are left for future work.

4.2.1 Convex evolutionary self-expressive model

Consider the function

$$\mathbf{C}_t = f_\theta(\mathbf{C}_{t-1}) = \alpha\mathbf{U} + (1-\alpha)\mathbf{C}_{t-1}, \tag{4.6}$$

where the values of parameters $\theta = (\mathbf{U}, \alpha)$ specify the relationship between $\mathbf{C}_{t-1}$ and $\mathbf{C}_t$, and need to be learned from data. Intuitively, the *innovation representation matrix* $\mathbf{U}$ captures changes in the representation of data points between consecutive time steps. The other term on the right-hand side of (4.6), $(1-\alpha)\mathbf{C}_{t-1}$, is the part of temporal representation that carries over from the previous snapshot of data. Therefore, the parametric function in (4.6) assumes

that the representation at time t is a convex combination of the representation at $t-1$, $\mathbf{C}_{t-1}$, and the "innovation" in the representation captured by matrix $\mathbf{U}$. Parameter $0 \leq \alpha \leq 1$ quantifies significance of the previous representation on the structure of data points at time t (i.e., it is reflective of the "memory" of representations). Intuitively, if the data is static we expect parameters to take on the values ($\alpha = 0$, $\mathbf{U} = \mathbf{0}$) or ($\alpha = 1$, $\mathbf{U} = \mathbf{C}_{t-1}$). Conversely, if the temporally evolving data is characterized by a subspace structure that undergoes significant changes, we expect α to be relatively close to 0.5.

Since at each time step we seek a sparse self-representation of data, the innovation matrix $\mathbf{U}$ should be sparse and satisfy $\text{diag}(\mathbf{U}) = \mathbf{0}$. Therefore, for the evolutionary model (4.6), search for the best collection of parameters that relate $\mathbf{C}_{t-1}$ and $\mathbf{C}_t$ leads to optimization

$$\begin{aligned} \min_{\mathbf{U},\alpha} \quad & \|\mathbf{X}_t - \mathbf{X}_t(\alpha\mathbf{U} + (1-\alpha)\mathbf{C}_{t-1})\|^2 \\ \text{s.t.} \quad & \text{diag}(\mathbf{U}) = \mathbf{0}, \quad \|\mathbf{U}\|_0 \leq k, \\ & 0 \leq \alpha \leq 1. \end{aligned} \tag{4.7}$$

In the above optimization, k determines sparsity level of the innovation. Since each point in $\mathcal{S}_i$ can be expressed in terms of at most d points in $\mathcal{S}_i$, we typically set $k \leq d$.

We refer to (4.7) as the convex evolutionary self-expressive model (CESM) for the evolutionary subspace clustering. Note that due to the cardinality constraint, (4.7) is a non-convex optimization problem. In Section 4.3, we present methods that rely on alternating minimization to efficiently find an approximate solution to (4.7).

4.3 Alternating Minimization Algorithms for Evolutionary Subspace Clustering

In this section, we present alternating minimization schemes for finding the innovation representation matrix $\mathbf{U}$ and smoothing parameter α, i.e., for solving (4.7).

4.3.1 Finding parameters of the CESM model

We solve (4.7) for $\mathbf{U}$ and α in an alternating fashion. In particular, given $\mathbf{U}_{t-1}$, the innovation representation matrix found at time $t-1$, we determine value of the smoothing parameter according to

$$\alpha = \arg\min_{0 \leq \bar{\alpha} \leq 1} \|\mathbf{X}_t - \mathbf{X}_t(\bar{\alpha}\mathbf{U}_{t-1} + (1-\bar{\alpha})\mathbf{C}_{t-1})\|^2. \tag{4.8}$$

The objective function in (4.8) is unimodal and convex; in our implementation, we rely on the golden-section search algorithm [124] to efficiently find α. Having found α, we arrive at the representation learning step which requires solving

$$\begin{aligned} \min_{\mathbf{U}} \quad & \|\mathbf{X}_t - \mathbf{X}_t(\alpha\mathbf{U} + (1-\alpha)\mathbf{C}_{t-1})\|^2 \\ \text{s.t.} \quad & \operatorname{diag}(\mathbf{U}) = \mathbf{0}, \quad \|\mathbf{U}\|_0 \leq k, \end{aligned} \tag{4.9}$$

which is a non-convex optimization problem due to the cardinality constraint. Let $\widetilde{\mathbf{X}}_t = \frac{1}{\alpha}(\mathbf{X}_t - (1-\alpha)\mathbf{X}_t\mathbf{C}_{t-1})$. Then, (4.9) can equivalently be written as

$$\begin{aligned} \min_{\mathbf{U}} \quad & \|\widetilde{\mathbf{X}}_t - \mathbf{X}_t\mathbf{U}\|^2 \\ \text{s.t.} \quad & \operatorname{diag}(\mathbf{U}) = \mathbf{0}, \quad \|\mathbf{U}\|_0 \leq k. \end{aligned} \tag{4.10}$$

The optimization problem (4.10) is clearly related to static subspace clustering with sparse representation (cf. (4.1)) and, in general, to sparse reconstruction

and sparse support selection problem [8]. Similar to static sparse subspace clustering schemes [1–4], one can employ compressed sensing approaches such as basis pursuit (BP) [125] (or the related LASSO [70]), orthogonal matching pursuit (OMP) [41], and orthogonal least squares (OLS) [40] algorithms to find a suboptimal innovation matrix $\mathbf{U}$ in polynomial time.

In particular, the basis pursuit representation learning strategy leads to the convex program

$$\begin{aligned} \min_{\mathbf{U}} \quad & \|\mathbf{U}\|_1 + \frac{\lambda}{2}\|\widetilde{\mathbf{X}}_t - \mathbf{X}_t\mathbf{U}\|^2 \\ \text{s.t.} \quad & \operatorname{diag}(\mathbf{U}) = \mathbf{0}, \end{aligned} \tag{4.11}$$

which can be solved using any conventional convex solver (see Section 4.4 for an ADMM-based implementation). Here, $\lambda > 0$ is a regularization parameter that determines sparsity level of the innovation representations.

For the OMP-based strategy, to learn the representation for each data point $\mathbf{x}_{t,j}$, $j \in [N]$, we define an initial residual vector $\mathbf{r}_0 = \widetilde{\mathbf{x}}_{t,j}$ and greedily select k data points indexed by $\mathcal{A}^{(k)} = \{i_1, \ldots, i_k\} \subset [N]$ that contribute to the innovation representation of $\mathbf{x}_{t,j}$ according to

$$i_\ell = \arg\max_{i \in [N] \setminus \mathcal{A}^{(\ell-1)} \cup \{j\}} |\mathbf{r}_{\ell-1}\mathbf{x}_{t,i}|^2, \tag{4.12}$$

where $\ell \in [k]$. The residual vector is updated according to $\mathbf{r}_\ell = \mathbf{P}(\mathcal{A}^{(\ell)})^\perp \widetilde{\mathbf{x}}_{t,j}$, where $\mathbf{P}(\mathcal{A}^{(\ell-1)})$ is the projection operator onto the subspace spanned by $\mathbf{X}_{t,\mathcal{A}^{(\ell)}}$ (i.e., the columns of $\mathbf{X}_t$ that are indexed by $\mathcal{A}^{(\ell)}$). Once $\mathcal{A}^{(k)}$ is determined, the innovation representation is computed as the least square solution $\mathbf{u}_j = \mathbf{X}^\dagger_{t,\mathcal{A}^{(k)}} \widetilde{\mathbf{x}}_{t,j}$.

The OLS-based representation learning strategy is similar to that of OMP, except the selection criterion is modified to

$$i_\ell = \arg\max_{i \in [N] \setminus \mathcal{A}^{(\ell-1)} \cup \{j\}} \frac{|\mathbf{r}_{\ell-1} \mathbf{x}_{t,i}|^2}{\|\mathbf{P}(\mathcal{A}^{(\ell-1)})^\perp \mathbf{x}_{t,i}\|_2^2}. \tag{4.13}$$

Finally, (4.6) yields the desired representation matrix $\mathbf{C}_t$.

4.3.2 Complexity analysis

The computational complexity of the proposed alternating minimization schemes is analyzed next.

Since it takes $\mathcal{O}(N^2)$ to evaluate the objective functions in (4.8), the complexity of finding the smoothing parameter using the golden-section search or any other linearly convergent optimization algorithm is $\mathcal{O}(N^2)$.

The computational cost of using BP-based strategy to learn the innovation representation matrix $\mathbf{U}$ in τ iterations of the interior-point method is $\mathcal{O}(\tau DN^3)$. However, as we demonstrate in Section 4.4, by using an efficient ADMM implementation the complexity can be reduced to $\mathcal{O}(\tau_m D^2 N^2)$ where τ_m denotes the maximum number of iterations of the ADMM algorithm.

Since they require search over $\mathcal{O}(N)$ D-dimensional data points in k iterations, the complexity of learning innovation representation matrix using OMP and OLS methods is $\mathcal{O}(kDN^2)$ and $\mathcal{O}(kD^2N^2)$, respectively. In Section 4.4 we discuss how one can reduce the complexity of OMP and OLS-based representation learning methods to $\mathcal{O}(DN^2)$ using accelerated and randomized greedy strategies.

4.4 Practical Extensions

Here we discuss potential practical issues and challenges that may come up in applications, and demonstrate how the proposed frameworks can be extended to handle such cases.

4.4.1 Tracking the evolution of clusters

The CESM framework promotes consistent assignment of data points to clusters over time. However, subspaces and the corresponding clusters evolve and thus one still faces the challenge of matching the clusters formed at consecutive time steps. This task essentially entails searching over permutations of clusters at time t and identifying the one that best matches the collection of clusters at time $t-1$. Quality of a matching (i.e., the weight of a matching) is naturally quantified by the number of data points common to the pairs of matched clusters. The solution to the so-called maximum weight matching problem can be found in polynomial time using the well-known Hungarian algorithm [126], or its variants that handle more sophisticated cases such as one-to-many and many-to-one maximum weight matching [127, 128]. In our numerical studies, we use the Hungarian algorithm to match clusters across time and evaluate clustering accuracy in experiments where the ground truth is known.

4.4.2 Adding and removing data points over time

In practice, it may happen that some of the data points vanish over time while new data points are introduced. In such settings, the number of data points and hence the dimension of representation matrices varies over time. Our proposed framework readily deals with such scenarios, as explained next.

Let $\mathcal{T}$ denote the set of indices of data points introduced at time t that were not present at time $t-1$. To incorporate these points into the model, we expand $\mathbf{C}_{t-1}$ by inserting all-zero vectors in rows and columns indexed by $\mathcal{T}$. New data points do not play a role in the temporal representations of other data points but they may participate in the innovation representation matrix (i.e., $\mathbf{U}$). Finally, let $\overline{\mathcal{T}}$ denote the set of indices of data points that were present at time $t-1$ but have vanished at time t; those points are removed from the model by excluding rows and columns of $\mathbf{C}_{t-1}$ indexed by $\overline{\mathcal{T}}$.

4.4.3 Accelerated representation learning

The most computationally challenging step of the proposed evolutionary self-expressive model is the representation learning step, i.e., the task of computing the innovation representation matrix $\mathbf{U}$. Therefore, when handling evolutionary data containing a large number of high-dimensional data points, efficient representation learning methods are needed. To this end, we here discuss how to employ BP, OMP, and OLS-based strategies in an accelerated fashion.

We first develop an ADMM algorithm for finding the innovation matrix $\mathbf{U}$ in (4.11) following a similar approach to that of [4].

Define $\widetilde{\mathbf{X}}_t = \frac{1}{\alpha}(\mathbf{X}_t - (1-\alpha)\mathbf{X}_t\mathbf{C}_{t-1})$. Introduce an auxiliary matrix $\mathcal{Z}$ and consider the optimization

$$\begin{aligned} \min_{\mathbf{U},\mathcal{Z}} \quad & \|\mathbf{U}\|_1 + \frac{\lambda}{2}\|\widetilde{\mathbf{X}}_t - \mathbf{X}_t\mathcal{Z}\|^2 \\ \text{s.t.} \quad & \mathcal{Z} = \mathbf{U} - \operatorname{diag}(\mathbf{U}), \end{aligned} \tag{4.14}$$

which is equivalent to the optimization problem (4.11) considered in Section 4.3. Form the augmented Lagrangian of (4.14) to obtain

$$\begin{aligned} \mathcal{L}_\rho(\mathbf{U},\mathcal{Z},\mathbf{Y}) = \|\mathbf{U}\|_1 &+ \frac{\lambda}{2}\|\widetilde{\mathbf{X}}_t - \mathbf{X}_t\mathcal{Z}\|^2 \\ &+ \frac{\rho}{2}\|\mathcal{Z} - \mathbf{U} + \operatorname{diag}(\mathbf{U})\|^2 \\ &+ \operatorname{tr}(\mathbf{Y}^\top(\mathcal{Z} - \mathbf{U} + \operatorname{diag}(\mathbf{U}))), \end{aligned} \tag{4.15}$$

where $\rho > 0$ and $\mathbf{Y}$ are the so-called penalty parameter and dual variable, respectively. Since adding the penalty term makes the objective function (4.15) strictly convex in the optimization variables, we can apply ADMM to solve it efficiently. The ADMM consists of the following iterations:

- $\mathcal{Z}^{\ell+1} = \min_{\mathcal{Z}^\ell} \mathcal{L}_\rho(\mathbf{U}^\ell, \mathcal{Z}^\ell, \mathbf{Y}^\ell)$.

  According to [4, 129], this problem has a closed-form solution that can be expressed as

  $$\mathcal{Z}^{\ell+1} = (\lambda\widetilde{\mathbf{X}}_t^\top\widetilde{\mathbf{X}}_t + \rho\mathbf{I})^{-1}(\lambda\widetilde{\mathbf{X}}_t^\top\widetilde{\mathbf{X}}_t - \mathbf{Y}^\ell + \rho\mathbf{U}^\ell). \tag{4.16}$$

  Note that a naive way to compute matrix inversion in (4.16) requires $\mathcal{O}(N^3)$ arithmetic operations. However, employing the matrix inversion

lemma and caching the result of the inversion reduces the computational cost to $\mathcal{O}(DN^2)$.

- $\mathbf{U}^{\ell+1} = \min_{\mathbf{U}^\ell} \mathcal{L}_\rho(\mathbf{U}^\ell, \mathcal{Z}^{\ell+1}, \mathbf{Y}^\ell)$.

  Note that the update of $\mathbf{U}$ also has a closed-form solution given by

$$\begin{aligned} \mathbf{J} &= \mathcal{T}_{\frac{1}{\rho}}(\mathcal{Z}^{\ell+1} + \frac{\mathbf{Y}^\ell}{\rho}), \\ \mathbf{U}^{\ell+1} &= \mathbf{J} - \text{diag}(\mathbf{J}), \end{aligned} \tag{4.17}$$

  where $\mathcal{T}_\eta(x) = (|x| - \eta)_+ \text{sgn}(x)$ is the so-called shrinkage-thresholding operator that acts on each element of the given matrix.

- $\mathbf{Y}^{\ell+1} = \mathbf{Y}^\ell + \rho(\mathcal{Z}^{\ell+1} - \mathbf{U}^{\ell+1})$, which is a dual gradient ascent update with step size ρ.

The above three steps are repeated until convergence criteria are met or the number of iterations exceeds a predefined maximum number. Although here we focus on ADMM as the optimization method, similar update rules can be obtained by using more advanced techniques including fast and linearized ADMM [130–133].

Now, we consider the OMP and OLS-based representation learning strategies. In each iteration of the OMP and OLS-based representation learning methods, one performs search over $\mathcal{O}(N)$ data points to identify which among them contribute to the innovation representation. In the case of large-scale datasets containing many data points, having $\mathcal{O}(N)$ "oracle calls" might be prohibitive. To reduce the computational burden, we can employ the PSG

algorithm that we introduced in Section 3.3 instead of the conventional greedy strategies to accelerate the representation learning process.

The complexity of the OLS-based method can further be reduced using the AOLS algorithm, introduced in Section 3.2. Recall, AOLS improves performance of OLS while requiring significantly lower computational costs. As opposed to OLS which greedily selects data points according to (4.13), AOLS efficiently builds a collection of orthogonal vectors to represent the basis of $\mathbf{P}(\mathcal{A}^{(\ell-1)})^{\perp}$ in order to reduce the cost of projection involved in (4.13). In addition, AOLS anticipates future selections via choosing L data points in each iteration, where $L \geq 1$ is an adjustable hyper-parameter. Selecting multiple data points in each iteration essentially reduces the number of iterations required to identify the representation of data points while typically leading to improved performance. Therefore, in our implementations, we employ the AOLS strategy instead of OLS to learn the innovation matrix $\mathbf{U}$.

4.4.4 Dealing with outliers and missing entries

The evolving data may contain outliers or missing entries at some or all of the time steps. The proposed framework allows for application of convex relaxation methods to handle such cases. Specifically, let $\mathbf{E}$ denote a sparse matrix containing outliers, and let Ω denote the set of observed entries of the corrupted data $\mathbf{X}_t^c$. Define the operator $\mathcal{P}_\Omega : \mathbb{R}^{D\times N} \to \mathbb{R}^{D\times N}$ as the orthogonal projector onto the span of matrices having zero entries on $[D] \times [N]\backslash\Omega$, but agreeing with $\mathbf{X}_t^c$ on entries indexed by the set Ω. Prior to employing greedy

representation learning methods, we identify outliers and values of the missing entries by solving the convex program

$$\begin{aligned} \min_{\mathbf{X}_t,\mathbf{E}} \quad & \|\mathbf{X}_t\|_* + \lambda_e \|\mathbf{E}\|_1 \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathbf{X}_t^c) = \mathcal{P}_\Omega(\mathbf{X}_t), \quad \mathbf{X}_t^c = \mathbf{X}_t + \mathbf{E}. \end{aligned} \tag{4.18}$$

Then we can apply the CESM framework using any of the greedy representation learning methods to process the "clean" data $\mathbf{X}_t$, ultimately finding the representations and clustering results.

In contrast to the greedy representation learning methods, BP-based approach benefits from joint representation learning and corruption elimination. That is, within the CESM framework, we may solve

$$\begin{aligned} \min_{\mathbf{X}_t,\mathbf{U},\mathbf{E}} \quad & \|\mathbf{U}\|_1 + \frac{\lambda}{2}\|\widetilde{\mathbf{X}}_t - \mathbf{X}_t\mathbf{U}\|^2 + \lambda_x\|\mathbf{X}_t\|_* + \lambda_e\|\mathbf{E}\|_1 \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathbf{X}_t^c) = \mathcal{P}_\Omega(\mathbf{X}_t), \quad \mathbf{X}_t^c = \mathbf{X}_t + \mathbf{E}, \\ & \widetilde{\mathbf{X}}_t = \frac{1}{\alpha}(\mathbf{X}_t - (1-\alpha)\mathbf{X}_t\mathbf{C}_{t-1}), \quad \operatorname{diag}(\mathbf{U}) = \mathbf{0}, \end{aligned} \tag{4.19}$$

to simultaneously learn the innovation, detect the outliers, and complete the missing entries.

4.5 Numerical Experiments

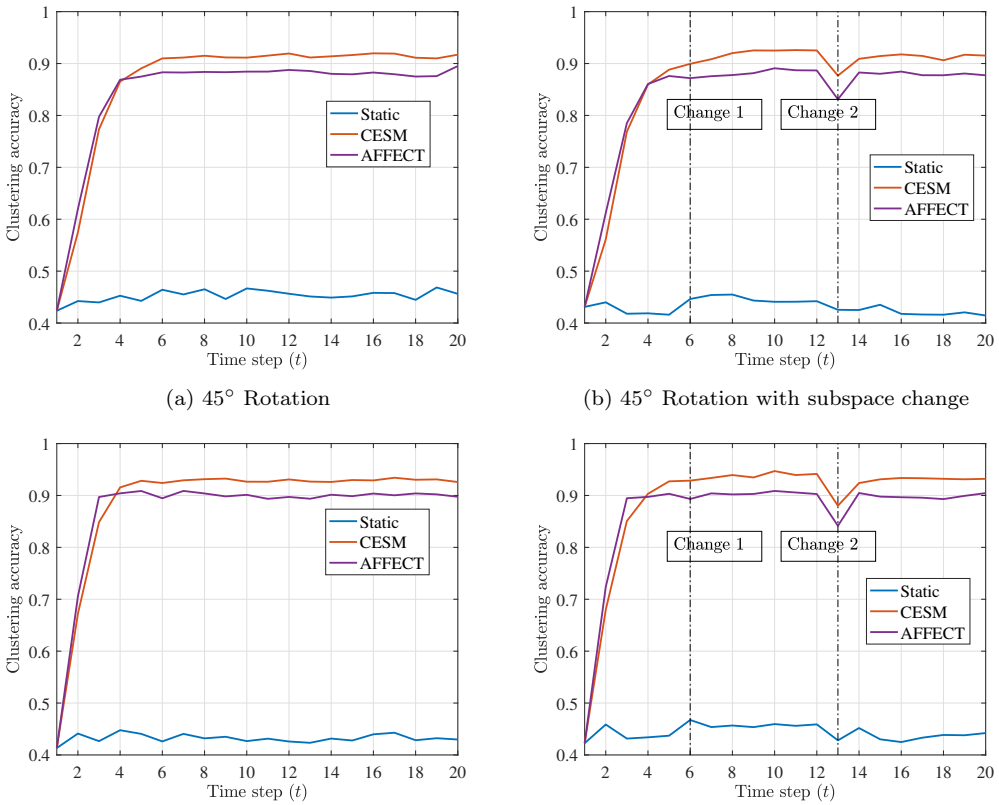
We compare performance of the proposed CESM framework to that of static subspace clustering schemes and the evolutionary clustering strategy of AFFECT [111] on synthetic, motion segmentation, and ocean water mass datasets. Note that AFFECT in general does not exploit the fact that the data points lie on a union of low-dimensional subspaces and its default choices for

affinity matrix are the negative squared Euclidean distance or its exponential form (i.e., an RBF kernel). We found that AFFECT performs poorly compared to other schemes (including static algorithms) when using default choices of affinity matrices. Hence, in all experiments we use the representation learning methods introduced in Section 4.3 for CESM as well as for AFFECT to ensure a fair assessment of the proposed evolutionary strategy.

4.5.1 Synthetic data

In a variety of applications including motion segmentation [6], the data points and their corresponding subspaces are characterized by rotational and transitional motions. Therefore, to simulate an underlying evolutionary process for data points lying on a union of subspaces, we consider the following scenario of rotating subspaces where we repeat each experiment for 150 trials.

At time $t = 1$, we construct $n = 10$ linear subspaces in $\mathbb{R}^D$, $D = 10$, each with dimension $d = 6$ by choosing their bases as the top d left singular vectors of a random Gaussian matrix in $\mathbb{R}^{D \times D}$. Then, we sample $N = 500$ data points, 50 from each subspace, by projecting random Gaussian vectors to the span of each subspace. Note that, in this setting, all the subspaces are distributed uniformly at random in the ambient space and all data points are uniformly distributed on the unit sphere of each subspace. According to the analysis in [2, 99, 100], this in turn implies that the subspace preserving property and the performance of representation learning methods based on BP, OMP, and AOLS is similar. However, we intentionally generate relatively



(a) 45° Rotation

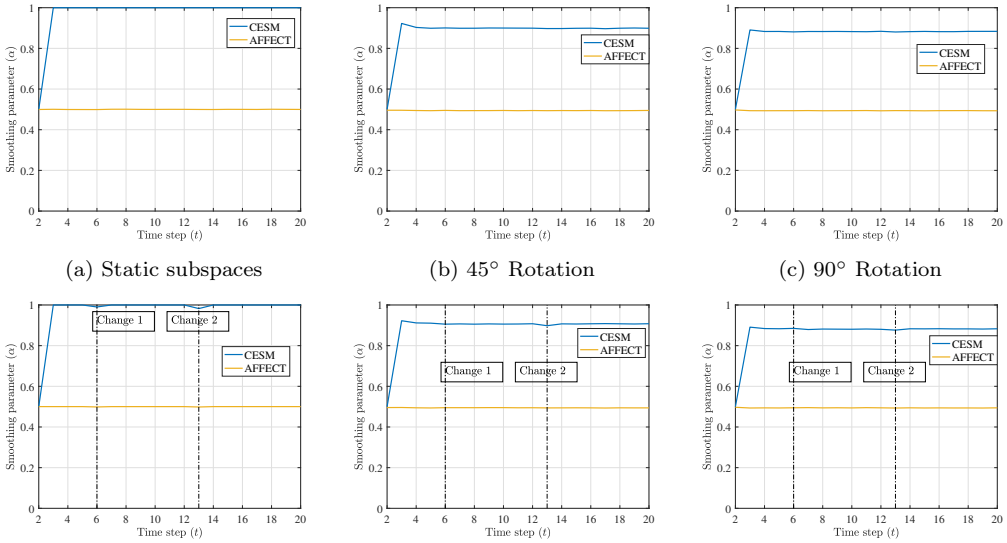
(b) 45° Rotation with subspace change

(c) 90° Rotation

(d) 90° Rotation with subspace change

Figure 4.1: Comparison of clustering accuracy of static and various evolutionary subspace clustering schemes employing OMP-based representation learning strategy on a simulated data containing 500 points that belong to a union of 10 rotating random subspaces in $\mathbb{R}^{10}$, each of dimension 6. The proposed CESM framework significantly improves the clustering accuracy and is superior to the AFFECT strategy. Moreover, CESM framework adapts to subspace changes at times $t = 6, 13$ as shown in the right-most plots.

low number of data points compared to the dimension of subspaces and the dimension of the ambient space; this creates a setting that is challenging for static subspace clustering algorithms. After constructing subspaces at time $t = 1$, we evolve the subspaces by rotating their basis 45° or 90° around a random vector and project the data points $\mathbf{X}_1$ on the span of the rotated



(a) Static subspaces (b) 45° Rotation (c) 90° Rotation

(d) Static subspaces with subspace change (e) 45° Rotation with subspace change (f) 90° Rotation with subspace change

Figure 4.2: Comparison of the smoothing parameter α for various evolutionary subspace clustering schemes employing OMP-based representation learning strategy on a simulated data containing 500 points lying on a union of 10 rotating random subspaces in $\mathbb{R}^{10}$, each of dimension 6. AFFECT's smoothing parameter remains approximately constant regardless of the underlying evolutionary behavior while the smoothing parameter for the CESM framework dynamically reflects the structure and reacts to cluster changes.

subspaces to obtain $\mathbf{X}_2$. We continue this process for $T = 20$ time steps. Note that for each subspace we perform rotation around a different random vector. Otherwise, if the rotations were around the same vector, the above setting would not be an evolutionary process as the relative positions of subspaces and data points would not vary over time. For brevity, we only present results of using OMP-based learning to find the representation matrices for static and competing evolutionary subspace clustering algorithms; however, we observed similar results for representation learning methods based on BP and AOLS.

Next, we consider a related experiment where in addition to rotation,

Figure 4.3: Example frames from the videos in the Hopkins 155 dataset [6].

at time $t = 6$ all data points generated from subspace $\mathcal{S}_{10}$ are absorbed by subspace $\mathcal{S}_9$. That is, at $t = 6$ we project $\mathbf{X}_{5,\mathcal{S}_{10}}$ to the span of $\mathcal{S}_9$. At time $t = 13$, these data points are separated from $\mathcal{S}_9$ and lie once again on $\mathcal{S}_{10}$. Hence, for $6 \leq t \leq 12$ the effective number of subspaces is $n = 9$ and there are 100 data points in $\mathcal{S}_9$.

The clustering accuracy results for these two experiments are illustrated in Fig. 4.1. For the first experiment, as seen from Fig. 4.1 (a) and Fig. 4.1 (c), the static SSC-OMP algorithm performs poorly compared to CESM and AFFECT. Since CESM and AFFECT exploit the evolutionary behavior of the data points, after a few time steps their accuracy significantly increases. We further observe that the proposed CESM framework achieves better accuracy than AFFECT; this is likely because the former exploits the self-expressiveness property of data points in the representation learning process while the latter simply combines current and prior representations to enforce the self-expressiveness property.

A comparison of the performance results in the second experiment is shown in Fig. 4.1 (b) and Fig. 4.1 (d). We observe that the performance of all evolutionary schemes suffers temporary degradations at times $t = 6$

and $t = 13$. The reason for this phenomenon is that the data points $\mathbf{X}_{6,\mathcal{S}_{10}}$ at $t = 6$ are significantly different from $\mathbf{X}_{5,\mathcal{S}_{10}}$ at time $t = 5$ due to being absorbed by $\mathcal{S}_9$ at time $t = 6$ and not belonging to $\mathcal{S}_{10}$. Therefore, since the subspaces are nearly independent, prior representations $\{\mathbf{c}_{5,i}\}_{i\in\mathcal{S}_{10}}$ and $\{\mathbf{c}_{12,i}\}_{i\in\mathcal{S}_{10}}$ are simply not well-aligned with the sudden changes taking place at times $t = 6, 13$. We further note that the deterioration in clustering accuracy is more severe for AFFECT than for CESM. We also observe from the figure that the proposed evolutionary scheme is able to quickly adapt to changes. At $t = 13$, the data points that were previously absorbed by $\mathcal{S}_9$ are projected back to the span of $\mathcal{S}_{10}$; as a result of this change, the performance of evolutionary schemes decreases. However, accuracy of the evolutionary methods recovers at $t = 14$ and improves onward as they exploit the evolving property of the data. Similar to the first experiment, due to exploiting the fact that data points lie on a union of subspaces, the proposed CESM framework outperforms the AFFECT's strategy.

Next, we investigate the value of α, i.e., the smoothing parameter discovered and used by CESM and AFFECT in the previously described experiments to further assess which scheme more accurately captures the evolutionary nature of the subspaces. Fig. 4.2 illustrates changes in the value of α over time, where in addition to the above two experiments we consider the scenario where subspaces are not rotating. The figure indicates that the smoothing parameter of AFFECT remains approximately 0.5 regardless of how rapidly the subspaces evolve. Note that the smoothing parameter essentially quantifies

evolutionary character of a dataset: if the data is static, we expect $\alpha = 0$ or $\alpha = 1$ for both CESM and AFFECT. As opposed to the AFFECT's smoothing parameter, the value of α for the CESM framework quickly converges to the anticipated level; note that we initialized α as 0.5. Fig. 4.2 (d)-(f) further suggest that the smoothing parameter of the proposed CESM framework noticeably changes at times $t = 6, 13$. This is a strong indication that CESM is capable of detecting subspace changes at $t = 6, 13$, while AFFECT fails to detect that the subspaces are rotating.

The above results suggest that the proposed framework improves performance of static subspace clustering algorithms when the data is evolving, while also being superior to state-of-the-art evolutionary clustering strategies in the considered settings. In contrast to prior schemes, the smoothing parameter of the proposed framework is meaningful and interpretable, and timely adapts to the underlying evolutionary behavior of the subspaces.

4.5.2 Real-time motion segmentation

Motion segmentation is the problem of clustering a set of two-dimensional trajectories extracted from a video sequence with multiple rigidly moving objects into groups; the resulting clusters correspond to different spatiotemporal regions (Fig. 4.3). The video sequence is often received as a stream of frames and it is desirable to perform motion segmentation in a real-time fashion [122, 123]. In the real-time setting, the t^{th} snapshot of $\mathbf{X}_t$ (a time interval consisting of multiple video frames) is of dimension $2F_t \times N_t$, where N_t is the

Table 4.1: Performance comparison of static and various evolutionary subspace clustering algorithms on real-time motion segmentation dataset. The best results for each row are in boldface fonts. For the CESM framework, the top results in each row correspond to the case of using a constant smoothing factor with the lowest average error while the bottom results in each row are achieved by using the proposed alternating minimization schemes to learn the smoothing parameter at each time step.

	Static			AFFECT			CESM		
Learning method	error	RI	runtime (s)	error	RI	runtime (s)	error	RI	runtime (s)
BP	10.76	86.29	46.16	9.86	87.78	47.35	8.88 **8.77**	**89.33** 89.14	45.10 **41.21**
OMP	31.66	62.00	1.80	14.47	86.21	3.31	**5.54** 6.85	**93.25** 88.23	**0.90** 0.93
AOLS ($L = 1$)	16.27	78.41	4.08	9.27	90.76	5.39	**6.57** 8.24	**91.97** 90.12	2.07 **1.93**
AOLS ($L = 2$)	8.54	89.10	3.75	6.17	93.08	5.17	**5.25** 5.70	**93.35** 92.85	1.85 **1.77**
AOLS ($L = 3$)	6.97	91.09	3.14	5.92	93.40	4.28	**5.49** 5.60	**94.17** 93.90	1.70 **1.69**

number of trajectories at t^{th} time interval, F_t is the number of video frames received in t^{th} time interval, n_t is the number of rigid motions at t^{th} time interval, and $F = \sum_t F_t$ denotes the total number of frames. Real-time motion segmentation falls within the scope of evolutionary subspace clustering since the received video sequence is naturally characterized by temporal properties; at t^{th} time interval, the trajectories of n_t rigid motions lie in a union of n_t low-dimensional subspaces in $\mathbb{R}^{2F_t}$, each with the dimension of at most $d_t = 3n_t$ [134].

In contrast to the real-time motion segmentation, clustering in offline settings is performed on the entire sequence, i.e., $\mathbf{X} = [\mathbf{X}_1^\top, \ldots, \mathbf{X}_T^\top]^\top$. Therefore, one expects to achieve more accurate segmentation in the offline settings. However, offline motion segmentation cannot be used in scenarios where some motions vanish or new motions appear in the video, or in cases where a real-time motion segmentation solution is desired.

To benchmark the performance of the proposed CESM framework, we consider Hopkins 155 database [6] which consists of 155 video sequences with 2 or 3 motions in each video (corresponding to 2 or 3 low-dimensional subspaces). Unlike the majority of prior work that process this data set in an offline setting, we consider the following real-time scenario: each video is divided into T data matrices $\{\mathbf{X}_t\}_{t=1}^T$ such that $F_t \geq 2n$ for a video with n motions. Then, we apply PCA on $\mathbf{X}_t$ and take its top $D = 4n$ left singular vectors as the final input to the representation learning algorithms.

We benchmark the proposed framework by comparing it to static sub-

space clustering and AFFECT; the former applies subspace clustering at each time step independently from the previous clustering results while the latter applies spectral clustering [93] on the weighted average of affinity matrices $\mathbf{A}_t$ and $\mathbf{A}_{t-1}$. The default choices for the affinity matrix in AFFECT are the negative squared Euclidean distance or its exponential form. Under these choices, AFFECT achieves a clustering error of 44.1542 and 21.9643 percent for the negative squared Euclidean distance or its exponential form, respectively, which as we present next is inferior even to the static subspace clustering algorithms. Hence, to fairly compare the performance of different evolutionary clustering strategies, we employ BP [3, 4, 125], OMP [1, 2, 41], and AOLS [64] with $L = 1, 2, 3$ to learn the representations for all schemes, including AFFECT.

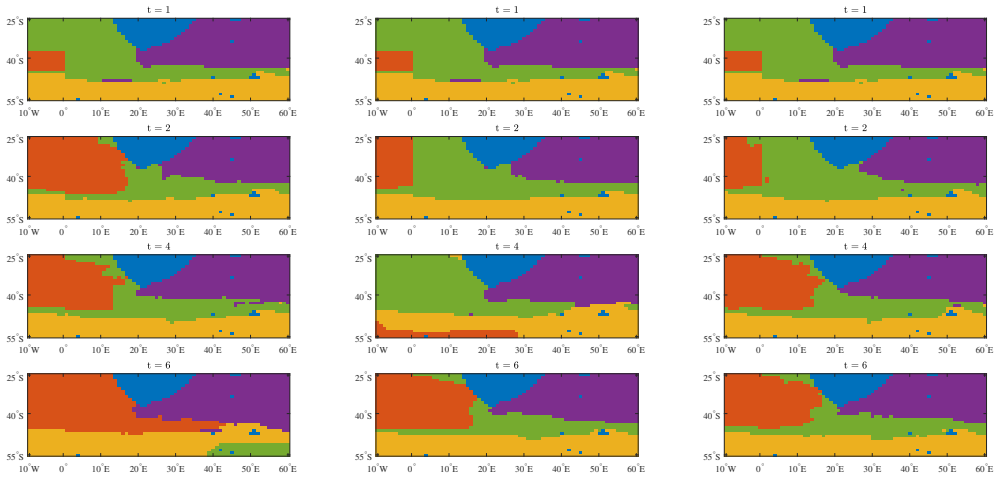
The performance of various schemes are presented in Table 4.1; there, the results are averaged over all sequences and all time intervals excluding the initial time interval $t = 1$. The initial time interval is excluded because for a specific representation learning method (e.g., BP), the results of static subspace clustering and evolutionary schemes coincide. Note that for the proposed CESM framework, the top results in each row of Table 4.1 correspond to the case of using a constant smoothing factor with the lowest average error while the bottom results in each row are achieved by using the proposed alternating minimization schemes to learn the best smoothing parameter for each time interval.

As we can see from the table, static subspace clustering has higher

clustering errors than their evolutionary counterparts; this is due to not incorporating any knowledge about the representations of the data points at other times. Furthermore, the proposed CESM framework is superior to AFFECT in terms of clustering error for all the representation learning methods. In addition, the proposed CESM framework achieves lower running time than static and AFFECT strategies, especially for the case of using OMP and AOLS as the representation learning methods. This supports the observation that CESM promotes sparser $\mathbf{U}_t$ by leveraging $\mathbf{C}_{t-1}$ in the process of learning $\mathbf{C}_t$ which in turn leads to faster convergence of OMP and AOLS. Similar to what we observed on synthetic datasets, the smoothing parameter of AFFECT (with both the default choices for the affinity matrix and the SSC-based affinity learning methods) was approximately 0.5 for all sequences and thus unable to capture evolutionary structure of the subspaces in a meaningful and interpretable manner.

4.5.3 Ocean water mass clustering

Ocean temperature and salinity has been tracked by Argo ocean observatory system comprising more than 3000 floats which provide 100,000 plus temperature and salinity profiles each year. These floats cycle between the ocean surface and 2000m depth every 10 days, taking salinity and temperature measurements at varying depths. A water mass is characterized as a body of water with a common formation and homogeneous features, such as salinity and temperature. Study of water masses can provide insight into climate



(a) static (b) AFFECT (c) CESM

Figure 4.4: Clustering results of four different types of water masses at 1000 dbar near the coast of south Africa (colored with blue). using static and various evolutionary subspace clustering schemes employing AOLS-based representation learning strategy with $L = 3$. The static subspace clustering scheme and AFFECT fail to keep track of the orange water mass at time $t = 6$ and $t = 4$, respectively. However, our proposed CESM framework detects homogeneous water masses across all time steps.

change, seasonal climatological variations, ocean biogeochemistry, and ocean circulation and its effect on transport of oxygen and organisms, which in turn affects the biological diversity of an area.

To illustrate the abilities of evolutionary subspace clustering in modeling various real-world problems, including those outside the computer vision community, we analyze the global gridded dataset produced via the Barnes method that was collected and made freely available by the international Argo program. This dataset contains monthly averages (since January 2004) of ocean temperature and salinity with 1 degree resolution worldwide [135, 136].

Table 4.2: Average salinity and temperature of four different types of water masses at 1000 dbar near the coast of south Africa identified by CESM framework employing AOLS-based representation learning strategy with $L = 3$ at different time steps. The results in top, middle and bottom for each cluster correspond to $t = 2, 4, 6$, respectively.

water mass	salinity level	temperature (°C)
orange	34.4554	3.4971
	34.3564	3.6164
	34.5008	3.2141
green	34.3452	3.5849
	34.6693	1.9910
	34.3640	3.6482
yellow	34.6603	2.0177
	34.4974	6.4445
	34.6680	2.0914
purple	34.4998	6.3313
	34.4649	3.4162
	34.4997	6.5522

In order to identify homogeneous water masses, we apply static and various evolutionary subspace clustering schemes, using AOLS-based representation learning method with $L = 3$ on the temperature and salinity data at the location near the coast of South Africa where the Indian Ocean meets the South Atlantic (specifically, the area located at latitudes 25° S to 55° S and longitudes 10° W to 60° E).

According to prior studies in [137–139], there are three well-known and strong mater masses in this area: (1) Agulhas currents, (2) the Antarctic intermediate water (AAIW), and (3) the circumpolar deep water mass. Therefore, following the discussion in [137] we set the number of clusters to $n = 4$ to further account for other water masses in the area.

The area described above accounts for $N = 1921$ evolving data points, each containing the monthly salinity and temperature from April to September for two years acquired starting in the year 2004 and 2005 ($t = 1$) until year 2014 and 2015 ($t = 6$). Temperature and salinity were normalized by subtracting the mean and dividing by the standard deviation of the entire time frame of interest. This procedure results in $24 \times N$ data matrices $\{\mathbf{X}_t\}_{t=1}^{6}$ which are then used as inputs to the evolutionary subspace clustering algorithms. As stated in Section 4.4, we employ the Hungarian method [126] to match the clustering solution at each time to the previous result.

The identified water masses by static, AFFECT, and CESM schemes using AOLS with $L = 3$ as the representation learning method are illustrated in Fig. 4.4 for $t = 1, 2, 4, 6$. The area colored with blue corresponds to the coast of south Africa and other islands in the target location. As we can see from the figure, all schemes are able to identify homogeneous water masses. However, the static subspace clustering and AFFECT schemes fail to properly detect the temporal changes in the formation of the green and orange water masses. In particular, the formation of the orange cluster evolves, as captured by the clustering results of the CESM framework. Since the CESM framework accounts for the underlying temporal behavior in the representation learning process and are able to infer appropriate smoothing factors, they are able to accurately keep track of the orange and green clusters across different time steps. Note that similarly to the results on synthetic and real-time motion segmentation datasets, the smoothing parameter of AFFECT was approxi-

mately 0.5. In addition, compared to AFFECT, CESM framework is capable of a faster adaptation to the changes in the formation of the orange water mass from $t = 1$ to $t = 2$.

The temperature and salinity averages for the water masses clustered by the CESM framework are shown in Table 4.2 where the results in top, middle and bottom for each cluster correspond to $t = 2, 4, 6$, respectively. A combination of these values, the geographic location of the clusters, and prior studies in [137, 138] suggest that the purple, orange, and yellow clusters corresponds to Agulhas currents, AAIW, and the circumpolar deep water masses, respectively.

4.6 Conclusion

In this chapter, we studied the problem of organizing data that evolves over time into clusters which is encountered in a number of practical applications in machine learning and signal processing. We introduce evolutionary subspace clustering, a method whose objective is to cluster a collection of evolving data points that lie on a union of low-dimensional evolving subspaces. To learn the parsimonious representation of the data points at each time step, we propose a non-convex optimization framework that exploits the self-expressiveness property of the evolving data while taking into account representation from the preceding time step. To find an approximate solution to the aforementioned non-convex optimization problem, we develop a scheme based on alternating minimization that both learns the parsimonious represen-

tation as well as adaptively tunes and infers a smoothing parameter reflective of the rate of data evolution. The latter addresses a fundamental challenge in evolutionary clustering – determining if and to what extent one should consider previous clustering solutions when analyzing an evolving data collection. Our experiments on both synthetic and real-world datasets demonstrate that the proposed framework outperforms state-of-the-art static subspace clustering algorithms and existing evolutionary clustering schemes in terms of both accuracy and running time, in a range of scenarios.

Chapter 5

Submodular Observation Selection in Networks

In many control, signal processing, and machine learning applications, one needs to efficiently collect the most informative observations from a potentially significantly larger set of uncertain observations. The goal of such selection is to reduce the burden on computational and communication resources while still providing accurate inference of unknown parameters.

In this chapter, motivated by the application of resource-constrained sensing systems, we examine conditions under which the mean-square error behaves similar to a submodular function. We further propose a randomized greedy algorithm for observation selection and establish performance guarantees on its achievable mean-square error. Finally, we propose a novel submodular information-exchange protocol to reduce the amount of communication in a network of sensing units operating under communication constraints. Contents of this chapter can be found in [50, 140, 141].[1]

[1]This chapter is based on existing publications: [Hashemi, Abolfazl, et al. Randomized greedy sensor selection: Leveraging weak submodularity. IEEE Transactions on Automatic Control (2020).], [Hashemi, Abolfazl, Mahsa Ghasemi, and Haris Vikalo. Submodular Observation Selection and Information Gathering for Quadratic Models. Proceedings of the 36th International Conference on Machine Learning. Vol. 97. 2019.], and [Hashemi, Abolfazl, Osman Fatih Kilic, and Haris Vikalo. Near-Optimal Distributed Estimation for a Network of Sensing Units Operating Under Communication Constraints. 2018 IEEE Con-

5.1 Introduction

Sensor networks deploy a large number of nodes that either exchange their noisy and possibly processed observations of a random process or forward those observations to a data fusion center. Due to constraints on computation, power and communication resources, instead of estimating the process using information collected by the entire network, the fusion center typically queries a relatively small subset of the available sensors. The problem of selecting the sensors that would acquire the most informative observations arises in a number of applications in control and signal processing systems including sensor selection for Kalman filtering [21–23, 142, 143], batch state and stochastic process estimation [24, 25], minimal actuator placement [26, 27], voltage control and meter placement in power networks [28–30], sensor scheduling in wireless sensor networks [21, 31], subset selection in machine learning [32–34], and graph signal processing [144, 145].

For a variety of performance criteria, finding an optimal subset of sensors requires solving a computationally challenging combinatorial optimization problem, possibly using branch-and-bound search [146]. By reducing it to the set cover problem, sensor selection was in fact shown to be NP-hard [51]. This hardness result has motivated development of numerous heuristics and approximate algorithms. For instance, [147] formulated the sensor selection problem

ference on Decision and Control (CDC). IEEE, 2018.] The author of this dissertation is the primary contributor. Mahsa Ghasemi and Osman Fatih Kilic helped with development of algorithms and implementations. Prof. Vikalo and Prof. Topcu aided in editing the papers and supervising the work.

as the maximization (minimization) of the log det of the Fisher information matrix (error covariance matrix), and found a solution by relaxing the problem to a semidefinite program (SDP). The computational complexity of finding the solution to the SDP relaxation of the sensor selection problem is cubic in the total number of available sensors, which limits its practical feasibility in large-scale networks consisting of many sensing nodes. Moreover, the solution to the SDP relaxation comes with no performance guarantees. To overcome these drawbacks, Shamaiah et al. [22] proposed a greedy algorithm guaranteed to achieve at least $(1 - 1/e)$ of the optimal objective at a complexity lower than that of the SDP relaxation. The theoretical underpinnings of the greedy approach to the sensor selection problem in [22] are drawn from the area of submodular function optimization. In particular, these results stem from the fact that the logarithm of the determinant ($\log\det$) of the Fisher information matrix is a monotone submodular function. Nemhauser et al. [48] studied maximization of such a function subject to a uniform matroid constraint and showed that the greedy algorithm, which iteratively selects items providing maximum marginal gain, achieves $(1 - 1/e)$ approximation factor. More recently, [23–25, 27], employed and analyzed greedy algorithms for finding approximate solutions to the $\log\det$ maximization problem in a number of practical settings.

Most of the existing work on greedy sensor selection has focused on optimizing the $\log\det$ of the Fisher information matrix, an objective indicative of the volume of the confidence ellipsoid. However, this criterion does not

explicitly relate to the mean-square error (MSE) which is often a natural performance metric of interest in estimation problems. The MSE, i.e., the trace of the covariance matrix of the estimation error, is not supermodular [148–153]. Therefore, its negative value, which we would like to maximize, is not submodular. Consequently, the setting and results of [48] do not apply to the MSE minimization problem.

Recently, Wang et al. [154] analyzed performance of the greedy algorithm in the general setting of maximizing a monotone non-decreasing objective function that is not necessarily submodular. They used a notion of the elemental curvature μ of the objective function to show that the greedy algorithm provides a $((1+\mu)^{-1})$-approximation under a matroid constraint. However, determining the elemental curvature defined in [154] is itself an NP-hard problem. Therefore, providing performance guarantees for the settings where the objective function is not submodular or supermodular, such as the trace of the covariance matrix of the estimation error in the sensor selection problem, remains a challenge.

Additionally, in many state estimation tasks in a network of sensing units that are capable of exchanging information, the information is gathered by the units through a nonlinear measurement model [155].

None of the schemes mentioned above consider the case of nonlinear measurement models as in these scenarios the error covariance matrix is in general unknown. Some important instances of nonlinearity are quadratic measurement models and inverse problems that occur in many natural phenomena

and real-world applications. For instance, in object tracking and localization applications in robotics and autonomous systems, the range measurements gathered by the radar systems follow a quadratic relation [156, 157].

To arrive at a (weak) submodular objective in settings where the model is nonlinear, existing schemes resort to approximation techniques, e.g., linearizing the model (the so-called local optimality approach) prior to the actual observation selection step [158–165]. However, theoretical guarantees for the performance of greedy algorithms hold only for the linearized model, i.e., for the linear approximation of the actual nonlinear model, and hence the selected subset of observations is not necessarily the most informative collection of measurements.

On another note, processing massive amounts of data collected by modern large-scale networks may be challenging even for greedy algorithms. To further reduce the computational burden of maximizing a monotone increasing, submodular function subject to cardinality constraints, the authors of [32] proposed a stochastic greedy algorithm that achieves $(1 - 1/e - \epsilon)$-approximation factor, where ϵ denotes a parameter that can be varied to explore the performance-complexity trade-off. However, the results of [32] do not apply to the sensor selection problem under the (non-submodular) MSE objective.

Given a network of units, it is further of interest to design an inference scheme that minimizes the overall estimation error [166,167]; however, in many applications it is of critical importance that each unit generates a reliable esti-

mate so as not to adversely affect decision making of other units in the network (e.g., in the context of autonomous vehicles, a unit with high estimation error may need to slow down and force other units to do the same). Therefore, we are interested in minimizing the total mean-square estimation error for the entire network while promoting balanced performance of the individual units.

In this dissertation we address the above challenges by developing weak submodularity-driven observation selection and information-exchange protocols. Specifically, under a linear model assumption, we first formulate the task of selecting sensors in a large-scale network as the problem of maximizing a monotone non-submodular objective function directly related to the mean-square estimation error. By closely inspecting curvature of the objective function, we derive sufficient conditions under which the function is weak submodular. This enables us to argue that when the measurement vectors are Gaussian or Bernoulli, as frequently encountered in reduced-dimensionality Kalman filtering via random projections [168], the MSE objective is with high probability weak submodular.

Next, we consider the task of observation selection under models where the relation between unknown states and measurements (partially) follows a quadratic equation. By drawing a connection between the classical Van Trees' inequality [169] and alphabetical optimality criteria [158], we devise new objective functions that exploit the quadratic relation of the observation model. We further prove that these functions possess two appealing properties, namely, monotonicity and (weak) submodularity under mild conditions on

the statistics of the problem and parameters of the model. These results allow us to develop a simple greedy scheme for observation selection with theoretical bounds on its achievable performance without requiring any a priori approximation step.

To decrease the cost of observation selection, under the assumption that the dynamics of the process and sensor observations is described by a state-space model, by building upon the work of Mirzasoleiman et al. [32], we propose a randomized greedy algorithm for sensor selection and derive a bound on the MSE of the state estimate formed by the Kalman filter that uses the measurements of the selected sensors. Our novel technique for the analysis of the randomized greedy algorithm provides results that improve over the existing performance guarantees of [32] for submodular maximization problems.

Finally, we formulate the task of state estimation in a network of sensing units under a constraint on communication resources and a demand for balanced performance of the individual units as the problem of maximizing a monotone objective function subject to a cardinality constraint. The proposed objective function consists of two parts: the total MSE of the network and a regularizing term that promotes balanced performance of individual units. Given the fact that the proposed formulation is NP-hard, by leveraging the notion of weak submodularity, we show that an efficient greedy algorithm achieves a constant factor approximation of the optimal schedule.

5.2 Observation Selection in Linear Model

In this section, we establish weak submodularity of the MSE objective in sensor networks gathering observations according to a linear model.

5.2.1 System model

Consider a discrete-time, linear, time-varying state-space model described by

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}_k \mathbf{x}_k + \mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k, \end{aligned} \tag{5.1}$$

where $\mathbf{x}_k \in \mathbb{R}^m$ is the state vector at time k that we aim to estimate, $\mathbf{y}_k \in \mathbb{R}^n$ is the measurement vector, $\mathbf{w}_k \in \mathbb{R}^m$ and $\mathbf{v}_k \in \mathbb{R}^n$ are zero-mean white Gaussian noise processes with covariances $\mathbf{Q}_k$ and $\mathbf{R}_k$, respectively, $\mathbf{A}_k \in \mathbb{R}^{m \times m}$ is the state transition matrix and $\mathbf{H}_k \in \mathbb{R}^{n \times m}$ is the matrix whose rows at time k are the measurement vectors $\mathbf{h}_{k,i} \in \mathbb{R}^m$, $1 \leq i \leq n$. We assume the states $\mathbf{x}_k$ are uncorrelated with $\mathbf{w}_k$ and $\mathbf{v}_k$. Additionally, we assume that $\mathbf{x}_0 \sim \mathcal{N}(0, \mathbf{\Sigma}_x)$ with $\mathbf{\Sigma}_x \succ \mathbf{0}$, and $\mathbf{R}_k = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_n^2)$. Note that, unlike the past work on greedy sensor selection in [22, 31, 43, 170], this model does not restrict the measurement noise covariance matrix to be a multiple of identity.

Due to limited resources, fusion center aims to select K out of n sensors and use their measurements to estimate the state vector $\mathbf{x}_k$ such that the trace of the covariance matrix of the estimation error, i.e., the MSE of the estimator implemented using the Kalman filter is minimized. Similar to prior work in [22, 31, 147], we assume that the measurement vectors $\mathbf{h}_{k,i}$ are available at

the fusion center. Let $\hat{\mathbf{x}}_{k|k-1}$ and $\hat{\mathbf{x}}_{k|k}$ denote the predicted and filtered linear minimum mean-square error (LMMSE) estimators of $\mathbf{x}_k$, respectively. In other words, $\hat{\mathbf{x}}_{k|k-1}$ is the LMMSE estimator of $\mathbf{x}_k$ given $\{\mathbf{y}_{S_1}, \ldots, \mathbf{y}_{S_{k-1}}\}$ and $\hat{\mathbf{x}}_{k|k}$ is the LMMSE estimator of $\mathbf{x}_k$ given $\{\mathbf{y}_{S_1}, \ldots, \mathbf{y}_{\mathcal{S}_k}\}$, where S_j denotes the set of sensors selected at time j and $\mathbf{y}_{S_j}$ denotes the vector of measurements collected by those sensors. Moreover, let $\mathbf{P}_{k|k-1}$ and $\mathbf{P}_{k|k}$ denote the predicted and filtered error covariance matrix of the Kalman filter at time instant k, respectively, i.e.,

$$\begin{aligned}\mathbf{P}_{k|k-1} &= \mathbf{A}_k \mathbf{P}_{k-1|k-1} \mathbf{A}_k^\top + \mathbf{Q}_k, \\ \mathbf{P}_{k|k} &= \left(\mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_{k,\mathcal{S}_k}^\top \mathbf{R}_{k,\mathcal{S}_k}^{-1} \mathbf{H}_{k,\mathcal{S}_k}\right)^{-1},\end{aligned}$$

where $P_{0|0} = \mathbf{\Sigma}_x$. Since $\mathbf{R}_k = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_n^2)$ and the measurements are uncorrelated across sensors, it holds that

$$\mathbf{P}_{k|k} = \left(\mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_{k,\mathcal{S}_k}^\top \mathrm{diag}(\{\sigma_i^{-2}\}_{i\in\mathcal{S}_k}) \mathbf{H}_{k,\mathcal{S}_k}\right)^{-1}.$$

Furthermore, $\mathbf{F}_{\mathcal{S}_k} = \mathbf{P}_{k|k}^{-1} = \mathbf{P}_{k|k-1}^{-1} + \sum_{i\in\mathcal{S}_k} \sigma_i^{-2} \mathbf{h}_{k,i} \mathbf{h}_{k,i}^\top$ is the corresponding Fisher information matrix. In the information form, the filtered estimator of $\mathbf{x}_k$ is expressed as

$$\hat{\mathbf{x}}_{k|k} = \mathbf{F}_{\mathcal{S}_k}^{-1} \mathbf{H}_{k,\mathcal{S}_k}^\top \mathrm{diag}(\{\sigma_i^{-2}\}_{i\in\mathcal{S}_k}) \mathbf{y}_k. \tag{5.2}$$

The MSE of the estimate found in (5.2) is given by the trace of the filtered error covariance matrix $\mathbf{P}_{k|k}$:

$$\mathrm{MSE}_{\mathcal{S}_k} = \mathbb{E}\left[\left\|\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}\right\|_2^2\right] = \mathrm{Tr}\left(\mathbf{F}_{\mathcal{S}_k}^{-1}\right). \tag{5.3}$$

To minimize (5.3), at each time step the fusion center seeks a solution to the optimization problem

$$\min_{\mathcal{S}} \quad \mathrm{Tr}\left(\mathbf{F}_{\mathcal{S}}^{-1}\right) \quad \text{s.t.} \quad \mathcal{S} \subset [n], \;\; |\mathcal{S}| = K. \tag{5.4}$$

By a reduction to the well-known set cover problem, the combinatorial optimization (5.4) can be shown to be NP-hard [51, 171]. In principle, to find the optimal solution one needs to exhaustively search over all schedules of K sensors. The techniques proposed in [147], albeit for an optimality criterion different from MSE and a simpler measurement model, find a subset of sensors that yields a sub-optimal MSE performance while being computationally much more efficient than the exhaustive search. In particular, [147] relies on finding the solution to the following SDP relaxation:

$$\begin{aligned} \min_{\mathbf{z}_k, \mathbf{Y}} \quad & \mathrm{Tr}(\mathbf{Y}) \\ \text{s.t.} \quad & 0 \leq z_{k,i} \leq 1, \;\; \forall i \in [n] \\ & \sum_{i=1}^{n} z_{k,i} = K \\ & \begin{bmatrix} \mathbf{Y} & \mathbf{I} \\ \mathbf{I} & \mathbf{P}_{k|k-1}^{-1} + \sum_{i=1}^{n} z_{k,i} \sigma_i^{-2} \mathbf{h}_{k,i} \mathbf{h}_{k,i}^{\top} \end{bmatrix} \succeq \mathbf{0}. \end{aligned} \tag{5.5}$$

The complexity of the SDP algorithm scales as $\mathcal{O}(n^3)$ which is infeasible in many practical settings. Furthermore, there are no guarantees on the achievable MSE performance of the SDP relaxation. Note that when the number of sensors in a network and the size of the state vector $\mathbf{x}_k$ are relatively large, even the greedy algorithm proposed in [22] may be computationally prohibitive.

5.2.2 Weak submodular linear sensor selection

Leveraging the idea of weak submodularity, in this section we propose a new formulation of the sensor selection problem concerned with minimizing the MSE of the Kalman filter that relies on a subset of network nodes to track states of a hidden random process.

Recall that for Kalman filtering in the resource-constrained scenario, if $\mathcal{S}_k$ is the set of sensors selected at time k then the error covariance matrix of the filtered estimate is $\mathbf{P}_{k|k} = \mathbf{F}_{\mathcal{S}_k}^{-1}$, the inverse of the corresponding Fisher information matrix. Let us define $f(\mathcal{S})$ as

$$f(\mathcal{S}) = \mathrm{Tr}\left(\mathbf{P}_{k|k-1} - \mathbf{F}_{\mathcal{S}}^{-1}\right). \tag{5.6}$$

Clearly, since $\mathbf{P}_{k|k-1}$ is known, there is a one-to-one correspondence between $f(\mathcal{S}_k)$ computed for a given subset of sensors $\mathcal{S}_k$ and the MSE of the LMMSE estimator (i.e., filtered estimate of the Kalman filter) that uses measurements acquired by the sensors in $\mathcal{S}_k$. Therefore, we can express the optimization problem (5.4) as

$$\max_{\mathcal{S}} \quad f(\mathcal{S}) \quad \text{s.t.} \quad \mathcal{S} \subset [n], \;\; |\mathcal{S}| = K. \tag{5.7}$$

We now argue that (5.7) is indeed a weak submodular optimization problem.

By defining $\mathcal{X} = [n]$ and $\mathcal{I} = \{\mathcal{S} \subset \mathcal{X} || \mathcal{S}| \leq K\}$, it is easy to see that $\mathcal{M} = (X, \mathcal{I})$ is a uniform matroid. In Proposition 5.2.1 below we characterize important properties of $f(\mathcal{S})$ and develop a recursive scheme to efficiently compute the marginal gain of querying a sensor. The formula for the marginal gain

of $f(\mathcal{S})$ is also of interest in our subsequent analysis of its weak submodularity properties.

Proposition 5.2.1. *Let* $f(\mathcal{S}) = \mathrm{Tr}\left(\mathbf{P}_{k|k-1} - \mathbf{F}_{\mathcal{S}}^{-1}\right)$. *Then,* $f(\mathcal{S})$ *is a monotonically increasing set function,* $f(\emptyset) = 0$, *and*

$$f_j(\mathcal{S}) = \frac{\mathbf{h}_{k,j}^{\top}\mathbf{F}_{\mathcal{S}}^{-2}\mathbf{h}_{k,j}}{\sigma_j^2 + \mathbf{h}_{k,j}^{\top}\mathbf{F}_{\mathcal{S}}^{-1}\mathbf{h}_{k,j}}, \tag{5.8}$$

where upon adding element $j \in \mathcal{X}\backslash\mathcal{S}$ *to* $\mathcal{S}$, $\mathbf{F}_{\mathcal{S}}$ *is updated according to*

$$\mathbf{F}_{\mathcal{S}\cup\{j\}}^{-1} = \mathbf{F}_{\mathcal{S}}^{-1} - \frac{\mathbf{F}_{\mathcal{S}}^{-1}\mathbf{h}_{k,j}\mathbf{h}_{k,j}^{\top}\mathbf{F}_{\mathcal{S}}^{-1}}{\sigma_j^2 + \mathbf{h}_{k,j}^{\top}\mathbf{F}_{\mathcal{S}}^{-1}\mathbf{h}_{k,j}}. \tag{5.9}$$

Proof. See Appendix B.1. ■

As stated, the MSE is not a supermodular function [148, 153]. Consequently, the proposed objective $f(\mathcal{S}) = \mathrm{Tr}\left(\mathbf{P}_{k|k-1} - \mathbf{F}_{\mathcal{S}}^{-1}\right)$ is also not submodular. However, as we show in Theorem 5.2.1, under certain conditions $f(\mathcal{S})$ is characterized by a bounded multiplicative curvature $\mathcal{C}_{\max}$. Theorem 5.2.1 also states a probabilistic theoretical upper bound on $\mathcal{C}_{\max}$ in scenarios where at each time step the measurement vectors $\mathbf{h}_{k,j}$'s are realizations of independent identically distributed (i.i.d.) random vectors drawn from a suitable distribution.

Before proceeding to Theorem 5.2.1 and its proof, we first state the matrix Bernstein inequality [172] and Weyl's inequality [173] which we will later use in the proof of Theorem 5.2.1.

Lemma 5.2.1. *(Matrix Bernstein inequality [172]) Let* $\{\mathbf{X}_\ell\}_{\ell=1}^n$ *be a finite collection of independent, random, Hermitian matrices in* $\mathbb{R}^{m\times m}$*. Assume that for all* $\ell \in [n]$*,*

$$\mathbb{E}\left[\mathbf{X}_\ell\right] = \mathbf{0}, \quad \lambda_{max}(\mathbf{X}_\ell) \leq L. \tag{5.10}$$

Let $\mathbf{Y} = \sum_{\ell=1}^n \mathbf{X}_\ell$*. Then, for all* $q > 0$*, it holds that*

$$\Pr\{\lambda_{max}(\mathbf{Y}) \geq q\} \leq m \exp\left(\frac{-q^2/2}{\|\mathbb{E}\left[\mathbf{Y}^2\right]\| + Lq/3}\right). \tag{5.11}$$

Lemma 5.2.2. *(Weyl's inequality [173]) Let* $\mathbf{A}$ *and* $\mathbf{B}$ *be two* $m \times m$ *real positive definite matrices. Then it holds that*

$$\lambda_l(\mathbf{A}) + \lambda_{\min}(\mathbf{B}) \leq \lambda_l(\mathbf{A}+\mathbf{B}) \leq \lambda_l(\mathbf{A}) + \lambda_{\max}(\mathbf{B}) \tag{5.12}$$

where $\lambda_l(\mathbf{A})$ *denotes the* l^{th} *largest eigenvalue of* $\mathbf{A}$*.*

We now proceed to the statement and proof of Theorem 5.2.1.

Theorem 5.2.1. *Let* c_f *be the multiplicative curvature of* $f(\mathcal{S})$*, the objective function of the sensor selection problem. Assume that* $\|\mathbf{h}_{k,j}\|_2^2 \leq C$ *for all* j *and* k*. If*

$$\lambda_{max}(\mathbf{H}_k^\top \mathbf{H}_k) \leq \left(\frac{1}{\phi} - \frac{1}{\lambda_{min}(\mathbf{P}_{k|k-1})}\right) \min_{j\in[n]} \sigma_j^2 \tag{5.13}$$

for some $0 < \phi < \lambda_{min}(\mathbf{P}_{k|k-1})$*, then it holds that*

$$c_f \leq \max_{j\in[n]} \frac{\lambda_{max}(\mathbf{P}_{k|k-1})^2(\sigma_j^2 + \lambda_{max}(\mathbf{P}_{k|k-1})C)}{\phi^2(\sigma_j^2 + \phi C)}. \tag{5.14}$$

Furthermore, if $\mathbf{h}_{k,j}$*'s are i.i.d. zero-mean random vectors with covariance matrix* $\sigma_h^2\mathbf{I}_m$ *such that* $\sigma_h^2 < C$*, then for all* $q > 0$*, with probability*

$$p \geq 1 - m\exp\left(\frac{-q^2/2}{(C-\sigma_h^2)(n\sigma_h^2 + q/3)}\right), \tag{5.15}$$

it holds that

$$\phi = \min_{j\in[n]} \left(\frac{1}{\lambda_{min}(\mathbf{P}_{k|k-1})} + \frac{n\sigma_h^2 + q}{\sigma_j^2} \right)^{-1} > 0. \tag{5.16}$$

Proof. We prove the statement of the theorem by relying on the recursive expression for the marginal gain stated in Proposition 1. We first establish a sufficient condition for weak submodularity of $f(\mathcal{S})$. In particular, from the definition of the multiplicative curvature and (5.8), for all $(S, T, j) \in \mathcal{X}_l$ we obtain

$$\begin{aligned} C_l &= \max_{(S,T,j)\in\mathcal{X}_l} \frac{(\mathbf{h}_{k,j}^\top \mathbf{F}_T^{-2} \mathbf{h}_{k,j})(\sigma_j^2 + \mathbf{h}_{k,j}^\top \mathbf{F}_{\mathcal{S}}^{-1} \mathbf{h}_{k,j})}{(\mathbf{h}_{k,j}^\top \mathbf{F}_{\mathcal{S}}^{-2} \mathbf{h}_{k,j})(\sigma_j^2 + \mathbf{h}_{k,j}^\top \mathbf{F}_T^{-1} \mathbf{h}_{k,j})} \\ &\leq \max_{(S,T,j)\in\mathcal{X}_l} \frac{\lambda_{max}(\mathbf{F}_T^{-2})(\sigma_j^2 + \lambda_{max}(\mathbf{F}_{\mathcal{S}}^{-1})\|\mathbf{h}_{k,j}\|_2^2)}{\lambda_{min}(\mathbf{F}_{\mathcal{S}}^{-2})(\sigma_j^2 + \lambda_{min}(\mathbf{F}_T^{-1})\|\mathbf{h}_{k,j}\|_2^2)}, \end{aligned} \tag{5.17}$$

where the inequality follows from the Courant–Fischer min-max theorem [173]. Notice that $\lambda_{max}(\mathbf{F}_{\mathcal{S}}^{-1}) = \lambda_{min}(\mathbf{F}_{\mathcal{S}})^{-1}$ and $\lambda_{min}(\mathbf{F}_T) \geq \lambda_{min}(\mathbf{F}_{\mathcal{S}}) \geq \lambda_{min}(\mathbf{F}_\emptyset) = \lambda_{min}(\mathbf{P}_{k|k-1}^{-1})$ by Lemma 5.2.2. This fact, along with the definition of c_f implies

$$\begin{aligned} c_f &\leq \max_{j\in[n]} \frac{\lambda_{max}(\mathbf{P}_{k|k-1})^2(\sigma_j^2 + \lambda_{max}(\mathbf{P}_{k|k-1})\|\mathbf{h}_{k,j}\|_2^2)}{\lambda_{max}(\mathbf{F}_{\mathcal{S}})^{-2}(\sigma_j^2 + \lambda_{max}(\mathbf{F}_T)^{-1}\|\mathbf{h}_{k,j}\|_2^2)} \\ &\overset{(a)}{\leq} \max_{j\in[n]} \frac{\lambda_{max}(\mathbf{P}_{k|k-1})^2(\sigma_j^2 + \lambda_{max}(\mathbf{P}_{k|k-1})\|\mathbf{h}_{k,j}\|_2^2)}{\lambda_{max}(\mathbf{F}_{[n]})^{-2}(\sigma_j^2 + \lambda_{max}(\mathbf{F}_{[n]})^{-1}\|\mathbf{h}_{k,j}\|_2^2)} \\ &\overset{(b)}{\leq} \max_{j\in[n]} \frac{\lambda_{max}(\mathbf{P}_{k|k-1})^2(\sigma_j^2 + \lambda_{max}(\mathbf{P}_{k|k-1})C)}{\lambda_{max}(\mathbf{F}_{[n]})^{-2}(\sigma_j^2 + \lambda_{max}(\mathbf{F}_{[n]})^{-1}C)}, \end{aligned} \tag{5.18}$$

where (a) follows from the fact that $\lambda_{max}(\mathbf{F}_{\mathcal{S}}) \leq \lambda_{max}(\mathbf{F}_T) \leq \lambda_{max}(\mathbf{F}_{[n]})$ and (b) holds since

$$g(x) = \frac{\sigma_j^2 + \lambda_{max}(\mathbf{P}_{k|k-1})x}{\sigma_j^2 + \lambda_{max}(\mathbf{F}_{[n]})^{-1}x} \tag{5.19}$$

is a monotonically increasing function for $x > 0$. Now, since the maximum

eigenvalue of a positive definite matrix satisfies the triangle inequality, we have

$$\begin{aligned}\lambda_{max}(\mathbf{F}_{[n]}) &\leq \frac{1}{\lambda_{min}(\mathbf{P}_{k|k-1})} + \lambda_{max}(\sum_{j=1}^{n} \frac{1}{\sigma_j^2}\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top) \\ &\leq \frac{1}{\lambda_{min}(\mathbf{P}_{k|k-1})} + \max_{j\in[n]} \frac{1}{\sigma_j^2}\lambda_{max}(\mathbf{H}_k^\top \mathbf{H}_k).\end{aligned} \tag{5.20}$$

Therefore, by combining inequalities (5.13) and (5.18) we obtain the result in (5.14).

Next, to analyze the setting of i.i.d random measurement vectors, we bound $\lambda_{max}(\mathbf{F}_{[n]})$ using Lemma 5.2.1. Let $\mathbf{X}_j = \mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top - \sigma_h^2\mathbf{I}_m$ and $\mathbf{Y} = \sum_{j=1}^{n} \mathbf{X}_j$. To use the result of Lemma 5.2.1, one should first verify expressions in (5.10). To this end, note that

$$\begin{aligned}\mathbb{E}[\mathbf{X}_j] &= \mathbb{E}[\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top - \sigma_h^2\mathbf{I}_m] \\ &= \mathbb{E}[\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top] - \sigma_h^2\mathbf{I}_m = \mathbf{0}.\end{aligned} \tag{5.21}$$

This in turn implies that $\mathbb{E}[\mathbf{Y}] = \mathbf{0}$. Since $\mathbf{X}_j$'s are independent,

$$\|\mathbb{E}[\mathbf{Y}^2]\| = \|\mathbb{E}[\sum_{j=1}^{n} \mathbf{X}_j^2]\| \leq \sum_{j=1}^{n} \|\mathbb{E}[\mathbf{X}_j^2]\| \tag{5.22}$$

by the linearity of expectation and the triangle inequality. To proceed, we need to determine $\lambda_{max}(\mathbf{X}_j)$ and $\mathbb{E}[\mathbf{X}_j^2]$. First, let us verify $\mathbf{h}_{k,j}$ is an eigenvector of $\mathbf{X}_j$ by observing that

$$\begin{aligned}\mathbf{X}_j\mathbf{h}_j &= \left(\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top - \sigma_h^2\mathbf{I}_m\right)\mathbf{h}_{k,j} \\ &= \left(\|\mathbf{h}_{k,j}\|_2^2 - \sigma_h^2\right)\mathbf{h}_{k,j},\end{aligned} \tag{5.23}$$

where $\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top - \sigma_h^2\mathbf{I}_m$ is the corresponding eigenvalue. Since $\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top$ is a rank-1 matrix, other eigenvalues of $\mathbf{X}_j$ are all equal to $-\sigma_h^2$. Hence,

$$\lambda_{max}(\mathbf{X}_j) \leq C - \sigma_h^2, \tag{5.24}$$

and we recall that $C-\sigma_h^2 > 0$. We can now establish an upper bound on $\mathbb{E}[\mathbf{X}_j^2]$ as

$$\begin{aligned}\mathbb{E}[\mathbf{X}_j^2] &= \mathbb{E}[\left(\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top - \sigma_h^2\mathbf{I}_m\right)\left(\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top - \sigma_h^2\mathbf{I}_m\right)] \\ &= \left(\|\mathbf{h}_{k,j}\|_2^2 - \sigma_h^2\right)\mathbb{E}[\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top] \\ &\quad - \sigma_h^2\mathbb{E}[\left(\mathbf{h}_{k,j}\mathbf{h}_{k,j}^\top - \sigma_h^2\mathbf{I}_m\right)] \\ &= \left(\|\mathbf{h}_{k,j}\|_2^2 - \sigma_h^2\right)\sigma_h^2\mathbf{I}_m \preceq (C-\sigma_h^2)\sigma_h^2\mathbf{I}_m,\end{aligned} \tag{5.25}$$

where we have used the fact that $\mathbb{E}[\mathbf{X}_j] = \mathbf{0}$. Thus, $L = C-\sigma_h^2$ and $\|\mathbb{E}[\mathbf{Y}^2]\| \leq n(C-\sigma_h^2)\sigma_h^2$. Now, according to Lemma 5.2.1, for all $q > 0$ it holds that $\Pr\{\lambda_{max}(\mathbf{Y}) \leq q\} \geq p$ where

$$p = 1 - m\exp\left(\frac{-q^2/2}{(C-\sigma_h^2)(n\sigma_h^2 + q/3)}\right). \tag{5.26}$$

Therefore,

$$\lambda_{max}(\mathbf{F}_{[n]}) \leq \frac{1}{\lambda_{min}(\mathbf{P}_{k|k-1})} + \max_{j\in[n]}\frac{n\sigma_h^2 + q}{\sigma_j^2} = \phi^{-1} \tag{5.27}$$

with probability p. This completes the proof. ■

Remark 5.2.1. The setting of i.i.d. random vectors described in Theorem 5.2.1 arises in scenarios where sketching techniques, such as random projections, are used to reduce dimensionality of the measurement equation (see [168] for more details). The following are often encountered examples of such settings:

1. *Multivariate Gaussian measurement vectors:* Let $\mathbf{h}_{k,j} \sim \mathcal{N}(0, \frac{1}{m}\mathbf{I}_m)$ for all j. It is straightforward to show that $\mathbb{E}[\|\mathbf{h}_{k,j}\|_2^2] = 1$. Furthermore, it can be shown that $\|\mathbf{h}_{k,j}\|_2^2$ is with high probability concentrated around its expected value. Therefore, for this case, $\sigma_h^2 = \frac{1}{m}$ and $C = 1$.

2. *Centered Bernoulli measurement vectors:* Let each entry of $\mathbf{h}_{k,j}$ be $\pm\frac{1}{\sqrt{m}}$ with equal probability. Therefore, $\|\mathbf{h}_{k,j}\|_2^2 = 1 = C$. Additionally, $\sigma_h^2 = \frac{1}{m}$ since the entries of $\mathbf{h}_{k,j}$ are i.i.d. zero-mean random variables with variance $\frac{1}{m}$.

We can interpret the conditions stated in Theorem 5.2.1 as requirements on the condition number of $\mathbf{P}_{k|k-1}$ as argued next. For a sufficiently large m and $\sigma_h^2 = \frac{1}{m}$, it holds that $C \approx 1$. Assume $\phi \geq \lambda_{max}(\mathbf{P}_{k|k-1})/\Delta$ for some $\Delta > 1$, and $\sigma_j^2 = \sigma^2$ for all $i \in [n]$. Define

$$\text{SNR} = \frac{\lambda_{max}(\mathbf{P}_{k|k-1})}{\sigma^2}, \tag{5.28}$$

and let

$$\kappa = \frac{\lambda_{max}(\mathbf{P}_{k|k-1})}{\lambda_{min}(\mathbf{P}_{k|k-1})} \geq 1 \tag{5.29}$$

be the condition number of $\mathbf{P}_{k|k-1}$. Then, following some elementary numerical approximations, we obtain the following corollary.

Corollary 5.2.1.1. *Let*

$$\Delta \geq \kappa + c_1 \frac{n}{m}\text{SNR} \tag{5.30}$$

for some $c_1 > 1$. Then with probability

$$p \geq 1 - m\exp(-\frac{n}{m}c_2) \tag{5.31}$$

it holds that $c_f \leq \Delta^3$ for some $c_2 > 0$.

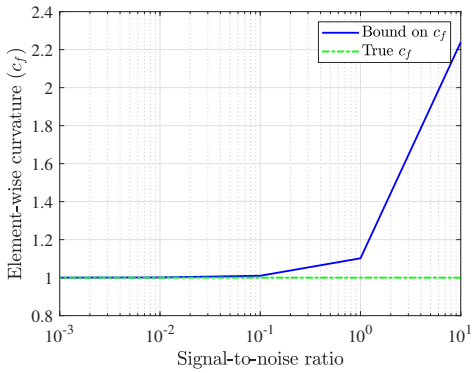


Figure 5.1: Evaluation of theoretical results in Theorem 5.2.1 for a sensor network with $m = 3$ and $n = 12$.

Informally, Theorem 5.2.1 states that for a well-conditioned $\mathbf{P}_{k|k-1}$ the curvature of $f(\mathcal{S})$ is small, which implies weak submodularity of $f(\mathcal{S})$. Furthermore, the probability of such an event exponentially increases with the number of available measurements.

5.2.3 Numerical experiments

To test the tightness of the bound established in Theorem 5.2.1, we empirically study a sensor selection problem with $n = 12$ Gaussian observations. Fig. 5.1 shows the true values of the maximum element-wise curvature found via exhaustive search as well as the bound stated in Theorem 5.2.1. As can be seen in the figure, the gap between the two is negligible at small SNR but becomes relatively loose at high SNR.

5.3 Observation Selection in Quadratic Model

In this section, we extend the result of Section 5.2 to nonlinear observation models. We establish our result by considering a multi-target tracking problem via a network of sensing units equipped with radar systems where the relation between unknown states and measurements (partially) follows a quadratic equation.

5.3.1 System model

Consider a networked sensing system where there are m sensing units in the network, sensing n objects with unknown locations. Sensing units are equipped with GPS and radar systems and can communicate with each other over locally established communication channels. Because of various practical restrictions such as power and communication constraints, only a subset of sensing units, known as leaders, can communicate to a control unit that surveys the environment via commanding the networked system. Each sensing unit acquires range and angular measurements of all the objects that are within the maximum radar detection range and transmits those measurements to its nearest leader.

Let $\mathbf{u}_k^i$ and $\mathbf{s}_k^j$ denote the location of i^{th} unit and j^{th} object at time k, respectively. Also, let $\sigma_k = [{\mathbf{s}_k^1}^\top, \ldots, {\mathbf{s}_k^n}^\top]^\top \in \mathbb{R}^{3n}$ denote the collection of unknown states evolving according to the nonlinear state equation $\sigma_k = g(\sigma_{k-1}) + \mathbf{w}_k$, where $\mathbf{w}_k$ is the zero-mean white Gaussian process noise at time k with covariance $\mathbf{Q}_k$.

If j^{th} object is within the range of i^{th} unit, the range and angular measurements of the radar system at time k have the following forms:

$$r_{ij} = \frac{1}{2}\|\mathbf{u}_k^i - \mathbf{s}_k^j\|_2^2 + \nu_{ij}, \tag{5.32}$$

$$\phi_{ij} = \arcsin\frac{u_k^i(3) - s_k^j(3)}{\|\mathbf{u}_k^i - \mathbf{s}_k^j\|_2} + \zeta_{ij}, \tag{5.33}$$

$$\alpha_{ij} = \arctan\frac{u_k^i(1) - s_k^j(1)}{u_k^i(2) - s_k^j(2)} + \eta_{ij}, \tag{5.34}$$

where ν_{ij}, ζ_{ij} and η_{ij} are zero-mean white Gaussian observation noises.[2] We denote by $\mathcal{X}_r$, $\mathcal{X}_\phi$, and $\mathcal{X}_\alpha$ the corresponding subsets of all gathered range and angular measurements and further we define $\mathcal{X} := \mathcal{X}_r \cup \mathcal{X}_\phi \cup \mathcal{X}_\alpha$. Note that depending on the location of objects and units, $3n \leq |\mathcal{X}| \leq 3nm$.

Due to limitations on the rate of communication between the leaders and the control unit that mainly stems from power limitation, and to reduce delays in tracking from high computation, only a subset $\mathcal{S}_k \subset \mathcal{X}$ of the gathered measurements is communicated to the control unit such that $|\mathcal{S}_k| \leq K$. In order to track the locations of the objects, the control unit employs extended Kalman filtering (EKF) using the received measurements. Hence, the selected subset by the unit leaders should be the one with lowest mean-square error of the EKF estimates of the objects' locations while satisfying the communication constraint.

To identify the most informative subset satisfying the communication constraint, existing locally optimal schemes (e.g., [160,162]) linearize the mea-

[2]We occasionally omit the time index for simplicity of the notation.

surement model in (5.32) – (5.34) around $\hat{\mathbf{s}}_{k-1}$, the estimate of objects' locations at time $k-1$, to obtain an approximate linearized measurement model $\mathbf{y}_k = \mathbf{H}_k \sigma_k + \mathbf{v}_k$, where $\mathbf{v}_k$ is the corresponding zero-mean white Gaussian observation noise with the diagonal covariance $\mathbf{R}_k = \text{diag}(\sigma_1^2, \dots \sigma_{|\mathcal{X}|}^2)$. Then, if for any subset of observations $\mathcal{S} \subseteq \mathcal{X}$, we consider the filtered error covariance of EKF,

$$\mathbf{P}_{k|k}(\mathcal{S}) = \left(\mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_{k,\mathcal{S}}^{\top} \mathbf{R}_{k,\mathcal{S}}^{-1} \mathbf{H}_{k,\mathcal{S}}\right)^{-1}, \tag{5.35}$$

where $\mathbf{P}_{k|k-1}$ is the predicted error covariance of EKF, observation selection is performed at each time k by optimizing trace or $\log\det$ of inverse of $\mathbf{P}_{k|k}$. That is,

$$\mathcal{S}_k = \arg\max_{|\mathcal{S}| \leq K} \quad \text{Tr}\left(\mathbf{P}_{k|k-1}\right) - \text{Tr}\left(\mathbf{P}_{k|k}(\mathcal{S})\right) \tag{5.36}$$

or

$$\mathcal{S}_k = \arg\max_{|\mathcal{S}| \leq K} \quad \log\det\left(\mathbf{P}_{k|k}^{-1}(\mathcal{S})\right) - \log\det\left(\mathbf{P}_{k|k-1}^{-1}\right). \tag{5.37}$$

Both of the above optimization problems are NP-hard. Hence, existing schemes rely on greedy heuristics or convex relaxations to find a suboptimal subset $\mathcal{S}^g$.

The major drawback of locally optimal approaches that are based on the linearized model is that the linearization step might distort the relational structure of the true nonlinear model severely. Hence, the selected subset of observations might not be the most informative collection of measurements. Although, a remedy for general and complex nonlinearities (such as angular measurements in (5.33) and (5.34)) seems rather infeasible to find, in the next

section we develop a novel framework for quadratic models (such as range measurement in (5.32)). Our proposed framework builds upon the idea of optimizing alphabetical scalarizations of the Van Trees' bound [169] on the moment of a weakly biased estimator. The Van Trees' inequality is outlined in the following theorem.

Theorem 5.3.1. *Let $\boldsymbol{\theta}$ be a collection of random unknown parameters, and let $\mathbf{y}_{\mathcal{S}} = \{y_i\}_{i\in\mathcal{S}}$ denote the collection of measurements indexed by the subset $\mathcal{S}$. For any estimator $\hat{\boldsymbol{\theta}}_{\mathcal{S}}$ that satisfies*

$$\int_{-\infty}^{+\infty} \nabla_{\boldsymbol{\Theta}} \left(p_{\boldsymbol{\theta}}(\boldsymbol{\Theta}) \mathbb{E}_{\mathbf{y};\boldsymbol{\theta}}[\hat{\boldsymbol{\theta}}_{\mathcal{S}} - \boldsymbol{\Theta}] \right) d\boldsymbol{\Theta} = \mathbf{0}, \tag{5.38}$$

it holds that

$$\mathbf{M}_{\mathcal{S}} \succeq \mathbb{E}_{\mathbf{y}_{\mathcal{S}},\boldsymbol{\theta}} \left[(\nabla_{\boldsymbol{\Theta}} \log q_{\boldsymbol{\theta}}(\boldsymbol{\Theta}))(\nabla_{\boldsymbol{\Theta}} \log q_{\boldsymbol{\theta}}(\boldsymbol{\Theta}))^{\top} \right]^{-1}, \tag{5.39}$$

where $\mathbf{M}_{\mathcal{S}} = \mathbb{E}[(\hat{\boldsymbol{\theta}}_{\mathcal{S}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}}_{\mathcal{S}} - \boldsymbol{\theta})^{\top}]$ is the so-called moment matrix associated with $\hat{\boldsymbol{\theta}}_{\mathcal{S}}$, and $q_{\boldsymbol{\theta}}(\boldsymbol{\Theta}) = p_{\boldsymbol{\theta};\mathbf{y}_{\mathcal{S}}}(\boldsymbol{\Theta};\mathbf{y})$ is the posterior distribution of $\boldsymbol{\theta}$ given $\mathbf{y}_{\mathcal{S}}$.

The condition stated in Theorem 5.3.1 essentially quantifies to what extent the estimator is biased. Indeed, for an unbiased estimator satisfying $\mathbb{E}_{\mathbf{y};\boldsymbol{\theta}}[\hat{\boldsymbol{\theta}}_{\mathcal{S}}] = \boldsymbol{\theta}$, this condition is met.

The lower bound in the Van Trees' inequality which is essentially a lower bound on the achievable mean-square error (MSE) cannot be computed in a closed-form for general nonlinear models. Nonetheless, as we show in the next section, the Van Trees' bound has a closed-form expression for the range measurements.

5.3.2 Proposed formulation

In this section, we devise a novel framework to select the most informative range measurements in a multi-object tracking sensing network. Throughout this section, we assume $\mathbb{E}[\mathbf{s}_k^j] = \hat{\mathbf{s}}_{k-1}^j$ for all $j \in [n]$. Admittedly, in the beginning of tracking, this assumption might not necessarily hold. Yet, as time passes the system generally improves the estimates of targets locations.

By defining $\tilde{\mathbf{s}}_k^j := \mathbf{s}_k^j - \hat{\mathbf{s}}_{k-1}^j$, $\mathbf{a}_{ij} := \hat{\mathbf{s}}_{k-1}^j - \mathbf{u}_k^i$, and $\tilde{r}_{ij} := r_{ij} - \frac{1}{2}\|\mathbf{a}_{ij}\|_2^2$, (5.32) can equivalently be written as

$$\tilde{r}_{ij} = \frac{1}{2}\|\tilde{\mathbf{s}}_k^j\|_2^2 + {\mathbf{a}_{ij}}^\top \tilde{\mathbf{s}}_k^j + \nu_{ij}. \tag{5.40}$$

The term $\mathbf{a}_{ij}$ can be thought of as the features or the design parameters. Let $\tilde{\sigma}_k = [\tilde{\sigma}_k^{1^\top}, \ldots, \tilde{\sigma}_k^{n^\top}]^\top \in \mathbb{R}^{3n}$, define $\mathbf{z}_{ij} := [\mathbf{0}_{3(j-1)}^\top, {\mathbf{a}_{ij}}^\top, \mathbf{0}_{3(n-j)}^\top]^\top$ and $\mathbf{X}_{ij} := \operatorname{diag}(\mathbf{0}_{3(j-1)}^\top, \mathbf{1}_3^\top, \mathbf{0}_{3(n-j)}^\top)$. Then, (5.40) can be written in terms of the concatenated vector of all centralized unknowns $\tilde{\sigma}_k$ according to

$$\tilde{r}_{ij} = \frac{1}{2}\tilde{\mathbf{s}}_k^\top \mathbf{X}_{ij} \tilde{\mathbf{s}}_k + {\mathbf{z}_{ij}}^\top \tilde{\mathbf{s}}_k + \nu_{ij}. \tag{5.41}$$

Our first theoretical result, stated in the following theorem, demonstrates that for the quadratic model in (5.41) the Van Trees' bound has a closed-form expression.

Theorem 5.3.2. *Let $\mathbf{B}_{\mathcal{S}}$ denote the lower bound in the Van Trees inequality for the quadratic model* (5.41)*. Then, for any subset $\mathcal{S} \subseteq \mathcal{X}_r$ it holds that*

$$\mathbf{B}_{\mathcal{S}} = \left(\sum_{ij \in \mathcal{S}} \frac{1}{\sigma_{ij}^2} \left(\mathbf{X}_{ij} \mathbf{P}_{k|k-1} \mathbf{X}_{ij}^\top + \mathbf{z}_{ij} \mathbf{z}_{ij}^\top \right) + \mathbf{P}_{k|k-1}^{-1} \right)^{-1}. \tag{5.42}$$

Proof. Let $\mathbf{r}$ denote the vector of all range measurements of the form (5.41), and $q_{\tilde{\sigma}_k}(\tilde{\mathbf{S}}) = p_{\mathbf{r}_\mathcal{S},\tilde{\sigma}_k}(\mathbf{r};\tilde{\mathbf{S}})$ denote the posterior distribution of $\tilde{\sigma}_k$ given $\mathbf{r}_\mathcal{S}$, and define

$$\boldsymbol{\mu}_\mathcal{S} = \mathrm{vec}(\{\frac{1}{2}\tilde{\sigma}_k^\top \mathbf{X}_{ij}\tilde{\sigma}_k + \mathbf{z}_{ij}^\top\tilde{\sigma}_k\}_{ij\in\mathcal{S}}). \tag{5.43}$$

Then the Van Trees' bound is found as

$$\begin{aligned}
\mathbf{B}_\mathcal{S}^{-1} &= \mathbb{E}_{\mathbf{r}_\mathcal{S},\tilde{\sigma}_k}[(\nabla_{\tilde{\mathbf{S}}}\log q_{\tilde{\sigma}_k}(\tilde{\mathbf{S}}))(\nabla_{\tilde{\mathbf{S}}}\log q_{\tilde{\sigma}_k}(\tilde{\mathbf{S}}))^\top] \\
&= \mathbb{E}_{\mathbf{r}_\mathcal{S},\tilde{\sigma}_k}[(\nabla_{\tilde{\mathbf{S}}}\log p_{\mathbf{r}_\mathcal{S};\tilde{\sigma}_k}(\mathbf{r};\tilde{\mathbf{S}})p_{\tilde{\sigma}_k}(\tilde{\mathbf{S}})) \\
&\qquad\qquad (\nabla_{\tilde{\mathbf{S}}}\log p_{\mathbf{r}_\mathcal{S};\tilde{\sigma}_k}(\mathbf{r};\tilde{\mathbf{S}})p_{\tilde{\sigma}_k}(\tilde{\mathbf{S}}))^\top] \\
&= \mathbb{E}_{\mathbf{r}_\mathcal{S},\tilde{\sigma}_k}[(\nabla_{\tilde{\mathbf{S}}}\log p_{\mathbf{r}_\mathcal{S};\tilde{\sigma}_k}(\mathbf{r};\tilde{\mathbf{S}})) \\
&\qquad\qquad (\nabla_{\tilde{\mathbf{S}}}\log p_{\mathbf{r}_\mathcal{S};\tilde{\sigma}_k}(\mathbf{r};\tilde{\mathbf{S}}))^\top] + \mathbf{J}_x,
\end{aligned} \tag{5.44}$$

where

$$\mathbf{J}_x = \mathbb{E}_{\mathbf{r}_\mathcal{S},\tilde{\sigma}_k}\left[(\nabla_{\tilde{\mathbf{S}}}\log p_{\tilde{\sigma}_k}(\tilde{\mathbf{S}}))(\nabla_{\tilde{\mathbf{S}}}\log p_{\tilde{\sigma}_k}(\tilde{\mathbf{S}}))^\top\right] \tag{5.45}$$

is the prior Fisher information on $\tilde{\sigma}_k$. Since in EKF settings

$$p_{\tilde{\sigma}_k}(\tilde{\mathbf{S}}) = \mathcal{N}(\mathbf{0}, \mathbf{P}_{k|k-1}), \tag{5.46}$$

then $\mathbf{J}_x = \mathbf{P}_{k|k-1}^{-1}$. Note that the conditional distribution $p_{\mathbf{r}_\mathcal{S};\tilde{\sigma}_k}(\mathbf{r};\tilde{\mathbf{S}})$ is the normal distribution $\mathcal{N}(\boldsymbol{\mu}_{\tilde{\sigma}_k}, \mathbf{R}_{k,\mathcal{S}})$. Therefore,

$$\nabla_{\tilde{\mathbf{S}}}\log p_{\mathbf{r}_\mathcal{S};\tilde{\sigma}_k}(\mathbf{r};\tilde{\mathbf{S}}) = -(\nabla_{\tilde{\mathbf{S}}}\boldsymbol{\mu}_\mathcal{S})\mathbf{R}_{k,\mathcal{S}}^{-1}(\mathbf{r}_\mathcal{S} - \boldsymbol{\mu}_\mathcal{S}), \tag{5.47}$$

where $[\nabla_{\tilde{\mathbf{S}}}\boldsymbol{\mu}_\mathcal{S}]_{ij} = \mathbf{X}_{ij}\tilde{\sigma}_k + \mathbf{z}_{ij}$. Using this result and applying the law of total expectation we obtain

$$\mathbf{B}_\mathcal{S}^{-1} = \sum_{ij\in\mathcal{S}} \frac{1}{\sigma_{ij}^2}\left(\mathbf{X}_{ij}\mathbf{P}_{k|k-1}\mathbf{X}_{ij}^\top + \mathbf{z}_{ij}\mathbf{z}_{ij}^\top\right) + \mathbf{P}_{k|k-1}^{-1}. \tag{5.48}$$

Inverting the last line that consists of an invertible positive definite matrix establishes the stated result and completes the proof. ■

Relying on the result of Theorem 5.3.2, we propose to use the trace and log det of inverse of $\mathbf{B}_{\mathcal{S}}$ as the objective functions in the observation selection task (effectively replacing $\mathbf{P}_{k|k}(\mathcal{S})$ with $\mathbf{B}_{\mathcal{S}}$ in (5.36) and (5.37)). That is, instead of linearizing the range measurements we propose to select the most informative range measurements according to one of the following optimization problems:

$$\mathcal{S}_k = \arg\max_{|\mathcal{S}|\leq K} \quad \mathrm{Tr}\left(\mathbf{P}_{k|k-1}\right) - \mathrm{Tr}\left(\mathbf{B}_{\mathcal{S}}\right), \tag{5.49}$$

$$\mathcal{S}_k = \arg\max_{|\mathcal{S}|\leq K} \quad \log\det\left(\mathbf{B}_{\mathcal{S}}^{-1}\right) - \log\det\left(\mathbf{P}_{k|k-1}^{-1}\right), \tag{5.50}$$

which are computationally challenging and NP-hard [51]. Theorem 5.3.2 opens a new avenue in the task of observation selection for quadratic models which, as we see in our simulation results, enables selection of observations leading to lower estimation error (i.e., higher information) as compared to the locally optimal approximation methods based on linearization [160, 162]. We note that the Van Trees' lower bound is asymptotically tight, i.e., it is tight in the high signal-to-noise ratio settings or in the case of sufficiently large number of observations. Hence, we expect to select a near-optimal subset by using the proposed selection criteria in such settings. In the next section, we further demonstrate monotonicity and weak submodularity of the proposed optimality criteria which in turn enables us to devise a greedy observation selection scheme with theoretical performance guarantee.

5.3.3 Greedy selection of range observations

In the following theorems, we consider trace and log det scalarizations of the Van Trees' bound $\mathbf{B}_S$ defined in Theorem 5.3.2 and show that they are monotonically non-decreasing as well as either submodular, or weak submodular. These results illustrate not only that the proposed objective functions deal with the quadratic model of range measurements without resorting to any approximations, but also that one can use the greedy observation selection method of Algorithm 1 to find a near-optimal subset of observations with performance guarantees established in Proposition 5.2.1. Proofs of the subsequent results are established by employing tools from linear algebra and matrix analysis such as Weyl's inequality, Sylvester's determinant identity, matrix inversion lemma, and Courant–Fischer min-max theorem [173].

Theorem 5.3.3. *Instate the notation and hypothesis of Theorem 5.3.2. The D-optimality of the Van Trees' bound, i.e.,*

$$f^D(\mathcal{S}) = \log\det\left(\mathbf{B}_{\mathcal{S}}^{-1}\right) - \log\det\left(\mathbf{P}_{k|k-1}^{-1}\right), \tag{5.51}$$

is monotone and submodular.

Proof. Let $ij \in \mathcal{X}_r \backslash \mathcal{S}$ be a new observation and define

$$\mathbf{J}_{ij} := \frac{1}{\sigma_{ij}^2}\left(\mathbf{X}_{ij}\mathbf{P}_{k|k-1}\mathbf{X}_{ij}^\top + \mathbf{z}_{ij}\mathbf{z}_{ij}^\top\right). \tag{5.52}$$

The marginal gain of adding a new observation to a subset $\mathcal{S}$ is

$$\begin{aligned} f_{ij}^{D}(\mathcal{S}) &= \log\det\left(\mathbf{B}_{\mathcal{S}}^{-1}+\mathbf{J}_{ij}\right)-\log\det\left(\mathbf{B}_{\mathcal{S}}^{-1}\right) \\ &\overset{(a)}{=} \log\frac{\det\mathbf{B}_{\mathcal{S}}^{-1}\det\left(\mathbf{I}+\mathbf{B}_{\mathcal{S}}^{1/2}\mathbf{J}_{ij}\mathbf{B}_{\mathcal{S}}^{1/2}\right)}{\det\mathbf{B}_{\mathcal{S}}^{-1}} \\ &= \log\det\left(\mathbf{I}+\mathbf{B}_{\mathcal{S}}^{1/2}\mathbf{J}_{ij}\mathbf{B}_{\mathcal{S}}^{1/2}\right)\overset{(b)}{\geq}0, \end{aligned} \tag{5.53}$$

where (a) follows from the fact that

$$\det\left(\mathbf{A}+\mathbf{B}\right)=\det\left(\mathbf{A}\right)\det\left(1+\mathbf{A}^{-1/2}\mathbf{B}\mathbf{A}^{-1/2}\right), \tag{5.54}$$

according to Sylvester's determinant identity, for any positive definite matrix $\mathbf{A}$ and Hermitian matrix $\mathbf{B}$ [173], and (b) holds due to $\det\left(\mathbf{I}+\mathbf{A}\right)\geq\left(1+\det\mathbf{A}\right)$ for any positive semidefinite matrix $\mathbf{A}$. Therefore f^{D} is monotonically increasing.

Now consider $\mathcal{S}\subseteq\mathcal{T}\subset\mathcal{X}_r$ and $ij\in\mathcal{X}\backslash\mathcal{T}$. Using the Sylvester's determinant identity we obtain

$$f_{ij}^{D}(\mathcal{T})/f_{ij}^{D}(\mathcal{S})=\frac{\log\det\left(\mathbf{I}+\mathbf{B}_{\mathcal{T}}^{1/2}\mathbf{J}_{ij}\mathbf{B}_{\mathcal{T}}^{1/2}\right)}{\log\det\left(\mathbf{I}+\mathbf{B}_{\mathcal{S}}^{1/2}\mathbf{J}_{ij}\mathbf{B}_{\mathcal{S}}^{1/2}\right)}\leq 1. \tag{5.55}$$

Hence, $c_{f^D}=\max_{(\mathcal{S},\mathcal{T},ij)\in\tilde{\mathcal{X}}_r} f_{ij}^{D}(\mathcal{T})/f_{ij}^{D}(\mathcal{S})\leq 1$ which in turn proves submodularity of $f^{D}(\mathcal{S})$. ■

Theorem 5.3.4. *Instate the notation and hypothesis of Theorem 5.3.2. The A-optimality of the Van Trees' bound, i.e.,*

$$f^{A}(\mathcal{S})=\mathrm{Tr}\left(\mathbf{P}_{k|k-1}\right)-\mathrm{Tr}\left(\mathbf{B}_{\mathcal{S}}\right), \tag{5.56}$$

is monotone and weak submodular and its additive and multiplicative curvatures satisfy

$$c_{f^A} \leq \max_{ij \in \mathcal{X}_r} \frac{\lambda_{\max}(\mathbf{B}_{\mathcal{X}_r}^{-1} + \mathbf{B}_{\mathcal{X}_r}^{-1}\mathbf{J}_{ij}\mathbf{B}_{\mathcal{X}_r}^{-1})}{\lambda_{\min}(\mathbf{P}_{k|k-1}^{-1} + \mathbf{P}_{k|k-1}^{-1}\mathbf{J}_{ij}\mathbf{P}_{k|k-1}^{-1})}, \tag{5.57}$$

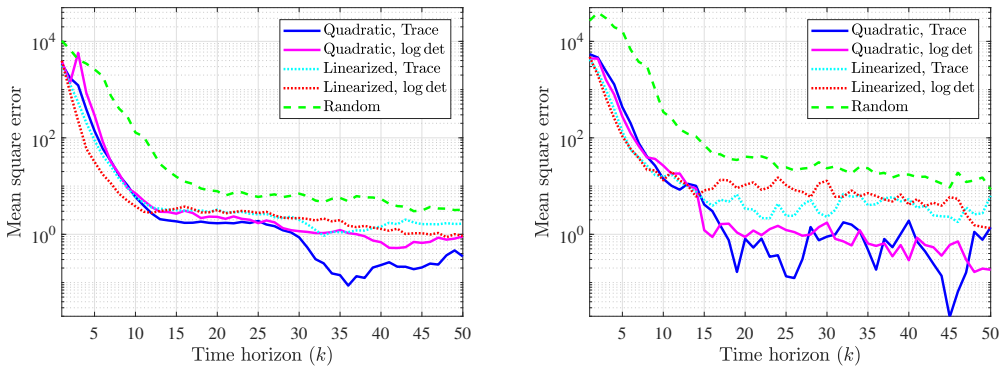
$$\begin{aligned} \epsilon_{f^A} \leq \max_{ij \in \mathcal{X}_r} & \lambda_{\max}(\mathbf{B}_{\mathcal{X}_r}^{-1} + \mathbf{B}_{\mathcal{X}_r}^{-1}\mathbf{J}_{ij}\mathbf{B}_{\mathcal{X}_r}^{-1}) \\ & - \lambda_{\min}(\mathbf{P}_{k|k-1}^{-1} + \mathbf{P}_{k|k-1}^{-1}\mathbf{J}_{ij}\mathbf{P}_{k|k-1}^{-1}), \end{aligned} \tag{5.58}$$

where $\mathbf{J}_{ij} = \frac{1}{\sigma_{ij}^2}\left(\mathbf{X}_{ij}\mathbf{P}_{k|k-1}\mathbf{X}_{ij}^\top + \mathbf{z}_{ij}\mathbf{z}_{ij}^\top\right)$*, for all* $ij \in \mathcal{X}_r$.

Proof. Proof is establish by using similar ideas employed in proof of Theorem 3. ■

The term $\mathbf{J}_{ij}$ is reflective of the amount of *information* captured by the ij^{th} observation. In this regard, Theorem 5.3.4 states that if the difference between the minimum and maximum information of individual observations is small, the objective in (5.49) is nearly submodular. Hence, the greedy observation selection scheme (Algorithm 1) is expected to find a good (informative) subset.

Theorems 5.3.3 and 5.3.4 establish monotonicity and (weak) submodularity of the proposed objective functions in (5.49) and (5.50). Hence, a suboptimal subset of range observations found by the greedy observation selection scheme (Algorithm 1) satisfies the performance bounds given in Proposition 2.3.2.



(a) tracking under identical noise powers

(b) tracking under random noise powers

Figure 5.2: Comparison of MSEs for random, linearized, and quadratic observation selection schemes in the multi-target tracking application.

5.3.4 Numerical experiments

In this section, we test the efficacy of the proposed quadratic observation selection objectives in a multi-object tracking application via UAV swarm (Fig. 5.8), using radar measurements instead of linear measurements, and compare their performance with those of random and locally optimal (linearization-based) schemes.

We consider a Monte Carlo simulation with 50 independent instances where 10 moving objects are initially uniformly distributed in a 5×10 area. At each time instance, the objects move in a random direction with a constant velocity set to 0.2. The swarm consists of 10 UAVs, equidistantly spread over the area, that move according to a periodic *parallel-path* search pattern [155]. The initial phases of the UAVs' motions are uniformly distributed to provide a better coverage of the area. The UAVs can acquire range and angular measurements of the objects that are within the maximum radar de-

tection range. The maximum radar detection range is set such that at each time step the UAVs together collect approximately 130-170 range and angular measurements. The communication bandwidth constraints limit the number of measurements transmitted to the control unit to 10% of the gathered measurements. For the proposed scheme, we select the range measurements using the proposed quadratic observation selection scheme while for angular measurements, we follow the locally optimal approach of [160, 162], i.e., linearization around the prior estimates. Performance of different schemes is assessed using the MSE of the EKF estimates of objects' locations. We consider two noise models: in the first scenario, the noise terms are i.i.d. Gaussian with $\sigma_{ij} = 0.01$ while in the second scenario, we logarithmically space the interval $(0.001, 0.01)$ to generate 10 points and select σ_i for each measurement uniformly at random from one of these 10 numbers.

The results for the first noise model are illustrated in Fig. 5.8(b). There, at the beginning of tracking all schemes have relatively high error. However, since the observations selected by the proposed schemes are chosen according to the exact range model, as time passes the MSE of the proposed schemes becomes significantly lower than those of locally optimal and random selection methods (especially under the A-optimality criterion). Fig. 5.8(b) also depicts that the MSE of estimates formed from observations selected by the proposed quadratic observation selection scheme using A-optimality is lower than the MSE achieved by selecting the observations via D-optimality. The explanation of this phenomenon is that if the estimator (here the EKF)

is a minimum variance unbiased estimator attaining (5.39) with equality, the A-optimality scalarization of the Van Trees' bound becomes equivalent to the MSE, the performance measure shown in Fig. 5.2(a). Therefore, intuitively, one expects to achieve lower MSE using the A-optimality scalarization of the Van Trees' bound, which is the case in this simulation.

The results for the second noise model are illustrated in Fig. 5.2(b) where we again observe superiority of the proposed quadratic framework to select a subset of observations with the lowest estimation error. Compared to Fig. 5.2(a), since the noise terms here are random, the MSE curves in Fig. 5.2(b) are not as smooth as those in Fig. 5.2(a).

5.4 Randomized Greedy Observation Selection

The complexity of SDP relaxation and greedy algorithms for sensor selection become prohibitive in large-scale systems. Motivated by the need for practically feasible schemes, we present a randomized greedy algorithm for finding an approximate solution to (5.7) and derive its performance guarantees.

5.4.1 Proposed scheme

Inspired by the technique in [32] proposed in the context of optimizing submodular objective functions, we develop a computationally efficient randomized greedy algorithm (see Algorithm 4) that finds an approximate solution to (5.7) with a guarantee on the achievable MSE performance of the Kalman filter that uses only the observations of the selected sensors. Algo-

Algorithm 4 Randomized Greedy Sensor Scheduling

Input: $\mathbf{P}_{k|k-1}$, $\mathbf{H}_k$, K, ϵ.
Output: Subset $\mathcal{S}_k \subseteq [n]$ with $|\mathcal{S}_k| = K$.
Initialize $\mathcal{S}_k^{(0)} = \emptyset$, $\mathbf{F}^{-1}_{\mathcal{S}_k^{(0)}} = \mathbf{P}_{k|k-1}$.
for $i = 0, \ldots, K-1$
    Choose R by sampling $s = \frac{n}{K}\log(1/\epsilon)$ indices uniformly at random from $[n] \backslash \mathcal{S}_k^{(i)}$.
    $i_s = \arg\max_{j \in R} \frac{\mathbf{h}_{k,j}^\top \mathbf{F}^{-2}_{\mathcal{S}_k^{(i)}} \mathbf{h}_{k,j}}{\sigma_j^2 + \mathbf{h}_{k,j}^\top \mathbf{F}^{-1}_{\mathcal{S}_k^{(i)}} \mathbf{h}_{k,j}}$.
    Set $\mathcal{S}_k^{(i+1)} = \mathcal{S}_k^{(i)} \cup \{i_s\}$.
    $\mathbf{F}^{-1}_{\mathcal{S}_k^{(i+1)}} = \mathbf{F}^{-1}_{\mathcal{S}_k^{(i)}} - \frac{\mathbf{F}^{-1}_{\mathcal{S}_k^{(i)}} \mathbf{h}_{k,i_s} \mathbf{h}_{k,i_s}^\top \mathbf{F}^{-1}_{\mathcal{S}_k^{(i)}}}{\sigma_j^2 + \mathbf{h}_{k,i_s}^\top \mathbf{F}^{-1}_{\mathcal{S}_k^{(i)}} \mathbf{h}_{k,i_s}}$
end for
return $\mathcal{S}_k = \mathcal{S}_k^{(K)}$.

rithm 4 performs the task of sensor scheduling in the following way. At each iteration of the algorithm, a subset R of size s is sampled uniformly at random and without replacement from the set of available sensors. The marginal gain provided by each of these s sensors to the objective function is computed using (5.8), and the one yielding the highest marginal gain is added to the set of selected sensors. Then the efficient recursive formula in (5.9) is used to update $\mathbf{F}_S^{-1}$ so it can be analyzed when making the selection in the next iteration. This procedure is repeated K times.

Remark 5.4.1. The parameter ϵ in Algorithm 4, $e^{-K} \leq \epsilon < 1$, is a predefined constant that is chosen to strike a desired balance between performance and complexity. When $\epsilon = e^{-K}$, each iteration includes all of the non-selected

sensors in R and Algorithm 4 coincides with the conventional greedy scheme. However, as ϵ approaches 1, $|R|$ and thus the overall computational complexity decreases.

5.4.2 Performance analysis of the proposed scheme

In this section we analyze Algorithm 4 and in Theorem 5.4.1 provide a bound on the performance of the proposed randomized greedy scheme when applied to finding an approximate solution to maximization problem (5.7).

Before deriving the main result, we first provide Lemma 5.4.1 that establishes a lower bound on the expected marginal gain.

Lemma 5.4.1. *Let $\mathcal{S}_k^{(i)}$ be the set of selected sensors at the end of the i^{th} iteration of Algorithm 4. Then*

$$\mathbb{E}\left[f_{(i+1)_s}(\mathcal{S}_k^{(i)})|\mathcal{S}_k^{(i)}\right] \geq \frac{1-\epsilon^\beta}{K} \sum_{j\in\mathcal{S}_k^*\setminus\mathcal{S}_k^{(i)}} f_j(\mathcal{S}_k^{(i)}), \tag{5.59}$$

where $\mathcal{S}_k^*$ is the set of optimal sensors at time k, $(i+1)_s$ is the index of the selected sensor at the $(i+1)^{st}$ iteration, $\beta = 1 + \max\{0, \frac{s}{2n} - \frac{1}{2(n-s)}\}$, and $s = \frac{n}{K}\log(1/\epsilon)$.

Proof. See Appendix B.2. ■

Theorem 5.4.1 below specifies how accurate the approximate solution to the sensor selection problem found by Algorithm 4 is. In particular, if $f(\mathcal{S})$ is characterized by a bounded multiplicative curvature, Algorithm 4 returns a

subset of sensors yielding an objective that is on average within a multiplicative factor of the objective achieved by the optimal schedule.

Theorem 5.4.1. *Let c_f be the multiplicative curvature of $f(\mathcal{S})$, i.e., the objective function of sensor scheduling problem in (5.7). Let $\mathcal{S}_k$ denote the subset of sensors selected by Algorithm 4 at time k, and let $\mathcal{S}_k^*$ be the optimum solution to (5.7) such that $|\mathcal{S}_k^*| = K$. Then $f(\mathcal{S}_k)$ is on expectation a multiplicative factor away from $f(\mathcal{S}_k^*)$. That is,*

$$\mathbb{E}\left[f(\mathcal{S}_k)\right] \geq \left(1 - e^{-\frac{1}{c}} - \frac{\epsilon^\beta}{c}\right) f(\mathcal{S}_k^*), \tag{5.60}$$

where $c = \max\{c_f, 1\}$, $e^{-K} \leq \epsilon < 1$, and $\beta = 1 + \max\{0, \frac{s}{2n} - \frac{1}{2(n-s)}\}$. Furthermore, the computational complexity of Algorithm 4 is $\mathcal{O}(nm^2 \log(\frac{1}{\epsilon}))$ where n is the total number of sensors and m is the dimension of $\mathbf{x}_k$.

Proof. Consider $\mathcal{S}_k^{(i)}$, the set generated by the end of the i^{th} iteration of Algorithm 4. Employing Lemma 2.3.1 with $\mathcal{S} = \mathcal{S}_k^{(i)}$ and $\mathcal{T} = \mathcal{S}_k^* \cup \mathcal{S}_k^{(i)}$, and using monotonicity of f, yields

$$\begin{aligned} \frac{f(\mathcal{S}_k^*) - f(\mathcal{S}_k^{(i)})}{\frac{1}{r}(1 + (r-1)c_f)} &\leq \frac{f(\mathcal{S}_k^* \cup \mathcal{S}_k^{(i)}) - f(\mathcal{S}_k^{(i)})}{\frac{1}{r}(1 + (r-1)c_f)} \\ &\leq \sum_{j \in \mathcal{S}_k^* \backslash \mathcal{S}_k^{(i)}} f_j(\mathcal{S}_k^{(i)}), \end{aligned} \tag{5.61}$$

where $|\mathcal{S}_k^* \backslash \mathcal{S}_k^{(i)}| = r$. Now, using Lemma 5.4.1 we obtain

$$\mathbb{E}\left[f_{(i+1)_s}(\mathcal{S}_k^{(i)}) | \mathcal{S}_k^{(i)}\right] \geq \left(1 - \epsilon^\beta\right) \frac{f(\mathcal{S}_k^*) - f(\mathcal{S}_k^{(i)})}{\frac{K}{r}(1 + (r-1)c_f)}. \tag{5.62}$$

Applying the law of total expectation yields

$$\begin{aligned}\mathbb{E}\left[f_{(i+1)_s}(\mathcal{S}_k^{(i)})\right] &= \mathbb{E}\left[f(\mathcal{S}_k^{(i+1)}) - f(\mathcal{S}_k^{(i)})\right] \\ &\geq \left(1-\epsilon^{\beta}\right)\frac{f(\mathcal{S}_k^*) - \mathbb{E}\left[f(\mathcal{S}_k^{(i)})\right]}{\frac{K}{r}(1+(r-1)c_f)}.\end{aligned} \tag{5.63}$$

Define

$$g(r) := \frac{1}{r}(1+(r-1)c_f). \tag{5.64}$$

It is easy to verify, e.g., by taking the derivative, that $g(r)$ is decreasing (increasing) with respect to r if $c_f < 1$ ($c_f > 1$). Let $c = \max\{c_f, 1\}$. Then

$$\frac{1}{r}(1+(r-1)c_f) \leq \frac{1}{r}(1+(r-1)c_f) \leq c. \tag{5.65}$$

Hence,

$$\mathbb{E}\left[f(\mathcal{S}_k^{(i+1)}) - f(\mathcal{S}_k^{(i)})\right] \geq \frac{1-\epsilon^{\beta}}{Kc}\left(f(\mathcal{S}_k^*) - \mathbb{E}\left[f(\mathcal{S}_k^{(i)})\right]\right). \tag{5.66}$$

Using an inductive argument and due to the fact that $f(\emptyset) = 0$, we obtain

$$\mathbb{E}[f(\mathcal{S}_k)] \geq \left(1 - \left(1 - \frac{1-\epsilon^{\beta}}{Kc}\right)^K\right) f(\mathcal{S}_k^*). \tag{5.67}$$

Finally, using the fact that $(1+x)^y \leq e^{xy}$ for $y > 0$ and the easily verifiable fact that $e^{ax} \leq 1 + axe^a$ for $0 < x < 1$,

$$\begin{aligned}\mathbb{E}[f(\mathcal{S}_k)] &\geq \left(1 - e^{-\frac{1-\epsilon^{\beta}}{c}}\right) f(\mathcal{S}_k^*) \\ &\geq \left(1 - e^{-\frac{1}{c}} - \frac{\epsilon^{\beta}}{c}\right) f(\mathcal{S}_k^*).\end{aligned} \tag{5.68}$$

To take a closer look at computational complexity of Algorithm 4, note that step 6 costs $\mathcal{O}(\frac{n}{K}m^2\log(\frac{1}{\epsilon}))$ since one needs to compute $\frac{n}{K}\log(\frac{1}{\epsilon})$ marginal

gains, each requiring $\mathcal{O}(m^2)$ operations. Furthermore, step 8 requires $\mathcal{O}(m^2)$ arithmetic operations. Since there are K such iterations, running time of Algorithm 4 is $\mathcal{O}(nm^2 \log(\frac{1}{\epsilon}))$. This completes the proof. ■

Using the definition of $f(\mathcal{S})$ we obtain Corollary 5.4.1.1 stating that, at each time step, the achievable MSE in (5.3) obtained by forming an estimate using sensors selected by the randomized greedy algorithm is within a factor of the optimal MSE.

Corollary 5.4.1.1. *Consider the notation and assumptions of Theorem 5.4.1 and introduce* $\alpha = 1 - e^{-\frac{1}{c}} - \frac{\epsilon^\beta}{c}$. *Let* $\text{MSE}_{\mathcal{S}_k}$ *denote the mean-square estimation error obtained by forming an estimate using information provided by the sensors selected by Algorithm 4 at time* k, *and let* MSE_o *be the optimal mean-square error formed using information collected by the sensors specified by the optimum solution of* (5.7). *Then the expected* $\text{MSE}_{\mathcal{S}_k}$ *is bounded as*

$$\mathbb{E}\left[\text{MSE}_{\mathcal{S}_k}\right] \leq \alpha \text{MSE}_o + (1-\alpha)\text{Tr}(\mathbf{P}_{k|k-1}). \tag{5.69}$$

Remark 5.4.2. Since the proposed sensor selection scheme is a randomized algorithm, the analysis of its *expected* MSE, as provided by Theorem 5.4.1 and Corollary 5.4.1.1, is a meaningful performance characterization. Notice that, as expected, α is decreasing in both c and ϵ. If $f(\mathcal{S})$ is characterized by a small curvature, then $f(\mathcal{S})$ is nearly submodular and the randomized greedy algorithm delivers a near-optimal sensor scheduling. As we decrease ϵ, α increases which in turn leads to a better approximation factor. Moreover,

by following an argument similar to that of the classical analysis in [48], one can show that the approximation factor for the greedy algorithm is given by $\alpha_g = 1 - e^{-\frac{1}{c}}$ (see also [152, 170]). Therefore, the term $\frac{\epsilon}{c}$ in α denotes the difference between the approximation factors of the proposed randomized greedy algorithm and the conventional greedy scheme.

Remark 5.4.3. The computational complexity of the greedy method for sensor selection that finds marginal gains via the efficient recursion given in Proposition 1 is $\mathcal{O}(Knm^2)$. Hence, our proposed scheme provides a reduction in complexity by $K/\log(\frac{1}{\epsilon})$ which may be particularly beneficial in large-scale networks, as illustrated in our simulation results.

Remark 5.4.4. In contrast to the results of [32] derived in the context of maximizing monotone submodular functions, Theorem 5.4.1 relaxes the submodularity assumption and states that the randomized greedy algorithm does not require submodularity to achieve near-optimal performance. Rather, if the set function is *weak submodular*, Algorithm 4 still selects a subset of sensors that provide an MSE near that achieved by the optimal subset of sensors. In addition, even if the function is submodular (e.g., if we use the log det objective instead of the MSE), the results of Theorem (5.4.1) offer an improvement over the theoretical results of [32] due to a tighter approximation bound stemming from the analysis presented in the proof of Theorem (5.4.1). Moreover, a major assumption in [32] is that R is constructed by sampling with replacement. Clearly, this contradicts the fact that a sensor selected in one iteration will not be in R in the subsequent iteration with probability one. On the contrary,

we assume R is constructed by sampling without replacement and carry out the analysis in this setting that matches the actual randomized greedy sensor selection strategy.

The randomized selection step of Algorithm 4 can be interpreted as an approximation of the marginal gains of the selected sensors using a greedy scheme [22]. More specifically, for the i^{th} iteration it holds that $f_{j_{rg}}(\mathcal{S}_k^{(i)}) = \eta_k^{(i)} f_{j_g}(\mathcal{S}_k^{(i)})$, where subscripts rg and g refer to the sensors selected by the randomized greedy (Algorithm 4) and the greedy algorithm, respectively, and $\{\eta_k^{(i)}\}_{i=1}^K$ are random variables with mean $\mu_i(\epsilon)$ that satisfy $0 < \ell_i(\epsilon) \leq \eta_k^{(i)} \leq 1$ for all $i \in [K]$.[3] In view of this argument, we obtain Theorem 5.4.2 which states that if $f(\mathcal{S})$ is characterized by a bounded multiplicative curvature and $\{\eta_k^{(i)}\}_{i=1}^K$ are independent random variables, Algorithm 4 returns a subset of sensors yielding an objective that with high probability is only a multiplicative factor away from the objective achieved by the optimal schedule.

Theorem 5.4.2. *Instate the notation and assumptions of Theorem 5.4.1. Let* $\{\eta_k^{(i)}\}_{i=1}^K$ *denote a collection of random variables such that* $0 < \ell_i(\epsilon) \leq \eta_k^{(i)} \leq 1$, *and* $\mathbb{E}[\eta_k^{(i)}] = \mu_i(\epsilon)$ *for all* i *and* k. *Let* $\ell_{min}(\epsilon) = \min_{i,k}\{\ell_i(\epsilon)\}$ *and* $\mu_{min}(\epsilon) = \min_{i,k}\{\mu_i(\epsilon)\}$. *Then,*

$$f(\mathcal{S}_k) \geq \left(1 - e^{-\frac{\ell_{min}(\epsilon)}{c}}\right) f(\mathcal{S}_k^*). \tag{5.70}$$

[3]Notice that $\ell_i(\epsilon)$ and $\mu_i(\epsilon)$ are time-varying quantities where the time index is omitted for the simplicity of notation.

Furthermore, if $\{\eta_k^{(i)}\}_{i=1}^K$ *are independent, then for all* $0 < q < 1$ *with probability at least* $1 - e^{-CK}$*, it holds that*

$$f(\mathcal{S}_k) \geq \left(1 - e^{-\frac{(1-q)\mu_{min}(\epsilon)}{c}}\right) f(\mathcal{S}_k^*), \tag{5.71}$$

for some $C > 0$*.*

Proof. Consider $\mathcal{S}_k^{(i)}$, the set generated by the end of the i^{th} iteration of Algorithm 4 and let $(i+1)_g$ and $(i+1)_{rg}$ denote the sensors selected by the greedy and randomized greedy algorithm in the i^{th} iteration, respectively. Let $c = \max\{c_f, 1\}$. Employing Lemma 2.3.1 with $S = \mathcal{S}_k^{(i)}$ and $T = \mathcal{S}_k^* \cup \mathcal{S}_k^{(i)}$, and using monotonicity of f, yields

$$\begin{aligned} f(\mathcal{S}_k^*) - f(\mathcal{S}_k^{(i)}) &\leq f(\mathcal{S}_k^* \cup \mathcal{S}_k^{(i)}) - f(\mathcal{S}_k^{(i)}) \\ &\leq c \sum_{j \in \mathcal{S}_k^* \setminus \mathcal{S}_k^{(i)}} f_j(\mathcal{S}_k^{(i)}). \end{aligned} \tag{5.72}$$

Using the fact that

$$f_j(\mathcal{S}_k^{(i)}) \leq f_{(i+1)_{rg}}(\mathcal{S}_k^{(i)}) \leq f_{(i+1)_g}(\mathcal{S}_k^{(i)}) \tag{5.73}$$

for all j, we obtain

$$f(\mathcal{S}_k^*) - f(\mathcal{S}_k^{(i)}) \leq cK f_{(i+1)_g}(\mathcal{S}_k^{(i)}). \tag{5.74}$$

On the other hand,

$$\begin{aligned} f(\mathcal{S}_k^{(i+1)}) - f(\mathcal{S}_k^{(i)}) &= f_{(i+1)_{rg}}(\mathcal{S}_k^{(i)}) \\ &= \eta_k^{(i+1)} f_{(i+1)_g}(\mathcal{S}_k^{(i)}). \end{aligned} \tag{5.75}$$

Combining (5.74) and (5.75) yields

$$f(\mathcal{S}_k^{(i+1)}) - f(\mathcal{S}_k^{(i)}) \geq \frac{\eta_k^{(i+1)}}{Kc}\left(f(\mathcal{S}_k^*) - f(\mathcal{S}_k^{(i)})\right). \tag{5.76}$$

Using an inductive argument similar to the one in the proof of Theorem 5.4.1, and noting that $f(\emptyset) = 0$,

$$\begin{aligned} f(\mathcal{S}_k) &\geq \left(1 - \left(1 - \sum_{i=1}^{K} \frac{\eta_k^{(i)}}{Kc}\right)\right) f(\mathcal{S}_k^*) \\ &\overset{(a)}{\geq} \left(1 - e^{-\sum_{i=1}^{K} \frac{\eta_k^{(i)}}{Kc}}\right) f(\mathcal{S}_k^*), \end{aligned} \tag{5.77}$$

where to obtain (a) we use the fact that $(1+x)^y \leq e^{xy}$ for $y > 0$. Therefore, since by assumption $\ell_{min}(\epsilon) \leq \ell_i(\epsilon) \leq \eta_k^{(i)} \leq 1$, we establish (5.70).

To show the second statement, i.e., prove (5.71) holds in the setting of independent $\{\eta_k^{(i)}\}_{i=1}^K$, we apply the Bernstein's inequality [174] to the sum of independent random variables $\sum_{i=1}^K \eta_k^{(i)}$. Since $\{\eta_k^{(i)}\}$ are bounded random variables, from Popoviciu's inequality [174] for all $i \in [K]$, it follows that

$$\mathrm{Var}[\eta_k^{(i)}] \leq \frac{1}{4}(1 - \ell_i(\epsilon))^2. \tag{5.78}$$

Hence, based on the Bernstein's inequality, for all $0 < q < 1$

$$\Pr\{\sum_{i=1}^{K} \eta_k^{(i)} < (1-q)\sum_{i=1}^{K} \mu_i\} < p, \tag{5.79}$$

where

$$\begin{aligned} p &= \exp\left(-\frac{(1-q)^2(\sum_{i=1}^K \mu_i(\epsilon))^2}{\frac{1-q}{3}\sum_{i=1}^K \mu_i(\epsilon) + \frac{1}{4}\sum_{i=1}^K (1-\ell_i(\epsilon))^2}\right) \\ &\overset{(b)}{\leq} \exp\left(-\frac{K(1-q)^2\mu_{min}^2(\epsilon)}{\frac{1-q}{3}\mu_{min}(\epsilon) + \frac{1}{4}(1-\ell_{min}(\epsilon))^2}\right) \\ &= e^{-C(\epsilon,q)K}, \end{aligned} \tag{5.80}$$

where (b) follows because p increases as we replace $\mu_i(\epsilon)$ and $\ell_i(\epsilon)$ by their lower bounds. Finally, substituting this results in (5.77) yields

$$f(\mathcal{S}_k) \geq \left(1 - e^{-\frac{(1-q)\mu_{min}(\epsilon)}{c}}\right) f(\mathcal{S}_k^*), \tag{5.81}$$

with probability at least $1 - e^{C(\epsilon,q)K}$. This completes the proof. ■

Similar to Corollary 5.4.1.1, we can now obtain a probabilistic bound on the MSE (5.3) achievable at each time step using the proposed randomized greedy algorithm. This result is stated in Corollary 5.4.2.1 below.

Corollary 5.4.2.1. *Consider the notation and assumptions of Corollary 5.4.1.1 and Theorem 5.4.2. Let* $0 < q < 1$ *and define* $\alpha = 1 - \exp(-\frac{(1-q)\mu_{min}(\epsilon)}{c})$. *Then, with probability at least* $1 - e^{-CK}$ *it holds that*

$$\text{MSE}_{\mathcal{S}_k} \leq \alpha \text{MSE}_o + (1 - \alpha)\text{Tr}(\mathbf{P}_{k|k-1}), \tag{5.82}$$

for some $C > 0$.

5.4.3 Numerical Experiments

To test the performance of the proposed randomized greedy algorithm, we compare it with the classic greedy algorithm and the SDP relaxation in a variety of settings as detailed next. We implemented the greedy and randomized greedy algorithms in MATLAB and the SDP relaxation scheme via CVX [175]. All simulations were run on a laptop with 2.0 GHz Intel Core i7-4510U CPU and 8.00 GB of RAM.

5.4.3.1 Kalman filtering in random sensor networks

We first consider the problem of state estimation in a linear time-varying system via Kalman filtering. For simplicity, we assume the state transition matrix to be identity, i.e., $\mathbf{A}_k = \mathbf{I}_m$. At each time step, the measurement vectors, i.e., the rows of the measurement matrix $\mathbf{H}_k$, are drawn according to $\mathcal{N} \sim (0, \frac{1}{m}\mathbf{I}_m)$. The initial state is a zero-mean Gaussian random vector with covariance $\mathbf{\Sigma_x} = \mathbf{I}_m$; and the process and measurement noise are zero-mean Gaussian with covariance matrices $\mathbf{Q} = 0.05\mathbf{I}_m$ and $\mathbf{R} = 0.05\mathbf{I}_n$, respectively.

The MSE of the filtered estimator and running time of each scheme is averaged over 100 Monte-Carlo simulations. The time horizon for each run is $T = 10$ seconds.

We first consider a system having state dimension $m = 50$ and the total number of sensors $n = 400$. We set a constraint on the number of sensors allowed to be queried at each time step to $K = 55$ and compare the MSE achieved by each sensor selection method over the time horizon of interest. For the randomized greedy algorithm we set $\epsilon = 0.001$. Fig. 5.3 shows that the greedy method consistently yields the lowest estimation MSE while the MSE provided by the randomized greedy algorithm is slightly higher. The MSE performance achieved by solving the SDP relaxation is considerably larger than those of the greedy and randomized greedy algorithms. The time it takes each method to select K sensors is given in Table 5.1. Both the greedy algorithm and the randomized greedy algorithm are much faster than the SDP formulation. Moreover, the randomized greedy scheme is nearly two

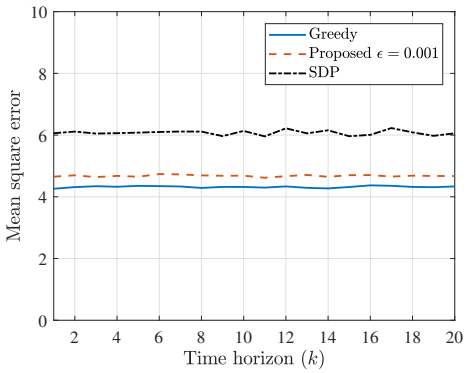


Figure 5.3: MSE comparison of randomized greedy, greedy, and SDP relaxation sensor selection schemes employed in Kalman filtering.

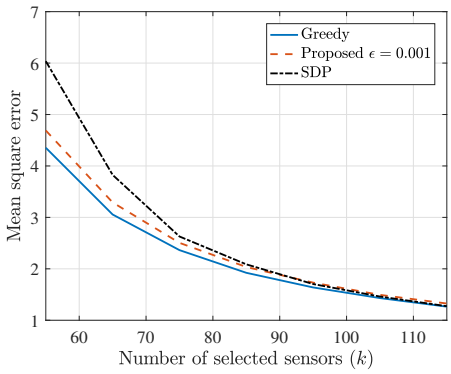
times faster than the greedy method.

Note that, in this example, in each iteration of the sensor selection procedure the randomized scheme only computes the marginal gain for a sampled subset of size 50. In contrast, the classic greedy approach computes the marginal gain for all 400 sensors. In summary, the greedy method yields slightly lower MSE but is much slower than the proposed randomized greedy algorithm.

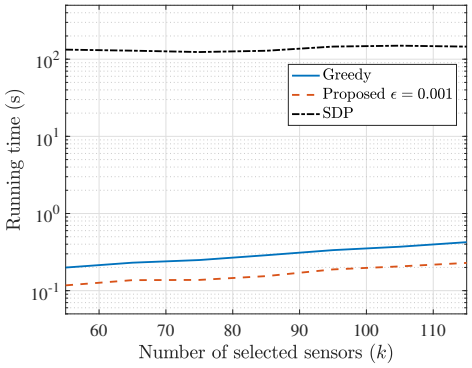
To study the effect of the number of selected sensors on the MSE performance, we vary K from 55 to 115 with increments of 10. The MSE values at the last time step for each algorithm are shown in Fig. 5.4(a). As the number of selected sensors increases, the estimation becomes more accurate,

Randomized Greedy	Greedy	SDP Relaxation
0.20 s	0.38 s	249.86 s

Table 5.1: Running time comparison of the randomized greedy, greedy, and SDP relaxation sensor selection schemes ($m = 50$, $n = 400$, $K = 55$, $\epsilon = 0.001$).



(a) Comparing MSE performance of different schemes.



(b) Running time comparison.

Figure 5.4: Comparison of randomized greedy, greedy, and SDP relaxation schemes as the number of selected sensors increases.

as reflected by the MSE of the estimates provided by each algorithm. Moreover, the differences between the MSE values achieved by different schemes monotonically decrease as more sensors are selected. The sensor selection running times shown in Fig. 5.4(b) indicate that the randomized greedy scheme is nearly twice as fast as the greedy method, while the SDP method is orders of magnitude slower than both greedy and randomized greedy algorithms.

Finally, to empirically verify the results of Theorem 5.4.2, in Fig. 5.5 we compare histograms of MSE achieved by the greedy and the proposed randomized greedy sensor selection schemes with various choices of ϵ when $K = 60$. As the figure shows, the MSE of sets selected by the proposed scheme is relatively close to that selected by state-of-the-art greedy algorithm. In addition, as ϵ decreases, the MSE of the randomized greedy algorithm approaches that of the greedy algorithm. These empirical observations coincide

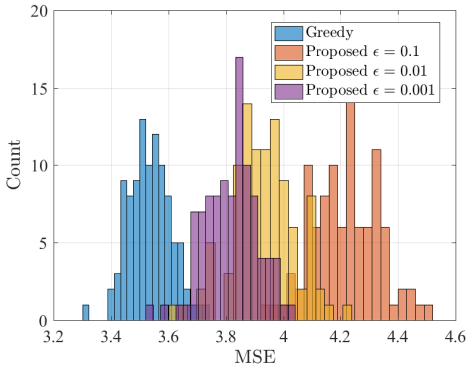
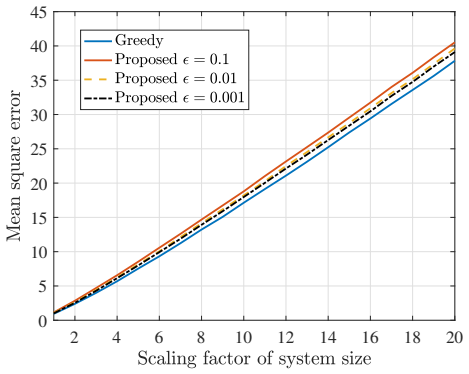


Figure 5.5: Histogram of MSE values for 100 independent realization of a sensor scheduling task for a sensor network with $m = 50$, $K = 60$, and $n = 400$.

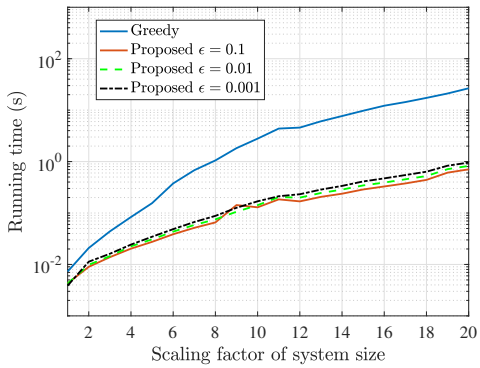
with our theoretical results in Theorem 5.4.2. That is, the proposed algorithm, although a randomized scheme, returns a near-optimal subset of sensors for each individual sensor selection task.

5.4.3.2 State estimation in large-scale networks

Next, we compare the performance of the randomized greedy algorithm to that of the greedy algorithm as the size of the system increases. We run both methods for 20 different system dimensions. The initial dimensions are set to $m = 20$, $n = 200$, and $K = 25$ and all three parameters are scaled by γ where γ varies from 1 to 20. In addition, to evaluate the effect of ϵ on the performance and runtime of the randomized greedy approach, we repeat experiments for $\epsilon \in \{0.1, 0.01, 0.001\}$. Note that the computational complexity of the SDP relaxation scheme is prohibitive in this setting and hence it is omitted. Fig. 5.6(a) illustrates the MSE comparison of the greedy and randomized greedy schemes.



(a) Comparing MSE performance of different schemes.



(b) Running time comparison.

Figure 5.6: A comparison of the randomized greedy and greedy algorithms for varied network size.

It shows that the difference between the MSEs is negligible. The running time is plotted in Fig. 5.6(b). As the figure illustrates, the gap between the running times grows with the size of the system and the randomized greedy algorithm performs nearly 28 times faster than the greedy method for the largest network. Fig. 5.6 shows that using a smaller ϵ results in a lower MSE while it slightly increases the running time. These results suggest that, for large systems, the randomized greedy provides almost the same MSE while being much faster than the greedy algorithm.

5.4.3.3 Accelerated multi-object tracking

Finally, we study a multi-object tracking application (See Fig. 5.7). Specifically, we consider a scenario where twenty moving objects are initially uniformly distributed in a 5×10 area. At each time instance, each object moves in a random direction with a constant speed set to 0.2. Twenty UAVs,

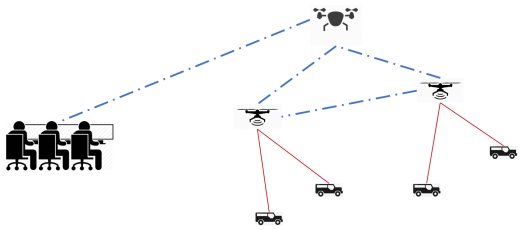
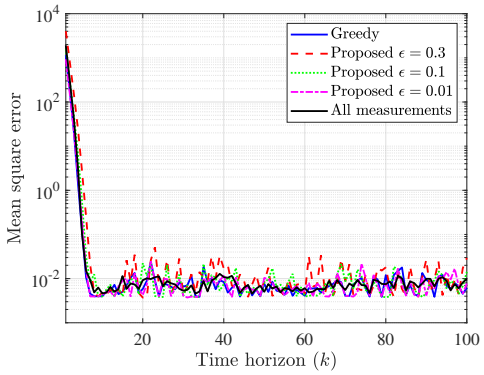
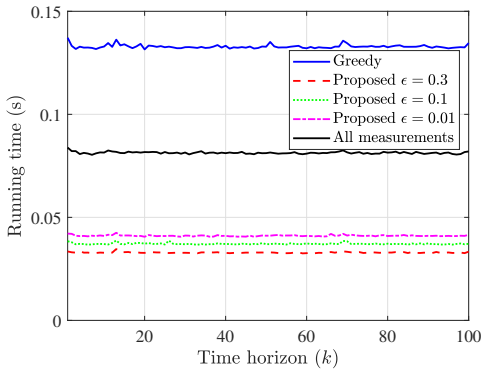


Figure 5.7: Multi-object tracking via a swarm of UAVs. The UAVs can communicate with each other and are equipped with GPS and radar systems. The objective is to select a small subset of range and angular measurements gathered by the UAVs to communicate to the control unit.



(a) Comparing MSE performance of different schemes.



(b) Running time comparison.

Figure 5.8: A comparison of the randomized greedy and greedy algorithms for a multi-object tracking application.

equidistantly spread over the area, move according to a periodic *parallel-path* search pattern [155]. The initial phases of the UAVs' motions are uniformly distributed to provide a better coverage of the area. The UAVs can acquire range and angular measurements of the objects that are within the maximum radar detection range. The maximum radar detection range is set such that at each time step the UAVs together collect approximately 600 range and angular measurements. The communication bandwidth constraints limit the number

of measurements transmitted to the control unit to $K = 100$. Note that since the radar measurement model is nonlinear, the control unit tracks objects via the extended Kalman filter. Fig. 5.8 shows a comparison in terms of the MSE and running time between the greedy and randomized greedy schemes for various values of ϵ. In the same figure we show performance of the scheme that ignores communications constraints and uses all the available measurements gathered by the UAVs. As Fig. 5.8(a) illustrates, the MSE performance of the greedy and proposed schemes are relatively close and similar to the performance of the scheme that uses all the measurements. However, a closer look at the running time comparison shown in Fig. 5.8(b) reveals that the combined runtime of randomized greedy sensor selection and Kalman filtering tasks is approximately 2 times faster than the runtime of the Kalman filter that uses all the measurements, and approximately 4 times faster than the combined runtime of the classical greedy sensor selection and Kalman filtering. Therefore, not only does the proposed scheme satisfy the communication constraint and perform nearly as well as using all the measurements, but it also significantly reduces the time needed to perform sensor selection and process the selected measurements in extended Kalman filtering.

5.5 Submodular Information-Exchange Communication Protocol

In this section, we propose a communication protocol based on weak submodular optimization to schedule communication among the sensors in

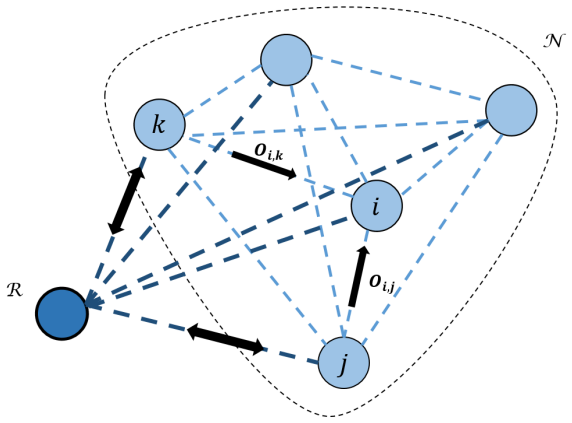


Figure 5.9: A fully connected network of units with sensing, communication, and processing capabilities; the communication between the units is constrained. Also shown is a scheduler R that organizes exchange of observations $\mathcal{O}_{i,j}$ and $\mathcal{O}_{i,k}$ to node i from nodes j and k, respectively.

a sensor network by using a central scheduler. For simplicity, we only consider alinear observation model [cf. Section 5.2] and rely on a simple greedy solver [cf. Algorithm 1]. Nonetheless, the result of this section can be readily extended to quadratic models and one can further employ the efficient randomized greedy algorithm that we proposed in Section 5.4.

5.5.1 System model

We consider a fully connected distributed network of m nodes with sensing, communication, and processing capabilities. The network also includes a scheduler that organizes the exchange of information among the units, i.e., the scheduler decides which information should be communicated from one unit to another.

One can think of the described network as having an undirected graph

structure, where vertices and edges represent the nodes and the connections among them, respectively. An example of such a network is illustrated in Fig. 5.9. There, the scheduler R organizes exchange of observations $\mathcal{O}_{i,j}$ and $\mathcal{O}_{i,k}$ to node i from nodes j and k, respectively. For instance, in a multi-target tracking application using a swarm of UAVs [142, 176, 177] [cf. Fig. 5.7], each unit is a UAV equipped with radar, GPS, and Lidar systems, while the swarm leader schedules exchange of information among the units.

To model the dynamics of the underlying hidden state $\mathbf{x} \in \mathbb{R}^n$, we assume a state-space model

$$\mathbf{x}_{t+1} = \mathbf{A}_t \mathbf{x}_t + \mathbf{w}_t,$$

where $\mathbf{A}_t \in \mathbb{R}^{n \times n}$ is the state-transition matrix and $\mathbf{w}_t \in \mathbb{R}^n$ is the zero-mean Gaussian state noise with covariance $\mathbf{Q} \in \mathbb{R}^{n \times n}$. We further assume that the state $\mathbf{x}_t$ is uncorrelated with $\mathbf{w}_t$ and the initial state $\mathbf{x}_0$ is sampled from a Gaussian distribution, i.e., $\mathbf{x}_0 \sim \mathcal{N}(0, \mathbf{\Sigma}_x)$.

The i^{th} node in the network acquires partial noisy linear observations of the underlying state according to

$$\mathbf{y}_{i,t} = \mathbf{H}_{i,t} \mathbf{x}_t + \mathbf{n}_{i,t},$$

where $i \in [m]$ and $\mathbf{H}_{i,t}$ denotes the matrix collects components of the underlying state sensed by i^{th} node. Let $\mathcal{L}_{i,t}$ denote the set of noisy observations of the components of $\mathbf{x}_t$ available to the i^{th} node (i.e., the noisy observations collected by the vector $\mathbf{y}_{i,t}$). Here, we do not make any assumptions on the

structure of $\mathbf{H}_{i,t}$. We assume that the observation noise $\mathbf{n}_{i,t} \in \mathbb{R}^{|\mathcal{L}_{i,t}|}$ is spatially and temporally independent zero-mean Gaussian noise with covariance $\mathbf{R}_{i,t} = \sigma_i^2 \mathbf{I}_{|\mathcal{L}_{i,t}|}$.

Without communication, each node only uses its acquired local measurements and performs Kalman filtering to estimate the underlying state $\mathbf{x}_t$ by minimizing the mean-squared error of the linear least-mean-square error (LLMSE) estimator. However, cooperation can greatly enhance the learning capabilities of the individual units as well as the entire network.

Let $\mathbf{P}_{\mathcal{L}_i,t}$ be the filtered error covariance matrix of the i^{th} agent at time t obtained by using only the local measurements $\mathcal{L}_{i,t}$. Then,

$$\mathbf{P}_{\mathcal{L}_i,t} = \left(\mathbf{P}_{i,t-1}^{-1} + \frac{1}{\sigma_i^2} \sum_{i_j \in \mathcal{L}_{i,t}} \mathbf{h}_{i_j} \mathbf{h}_{i_j}^\top \right)^{-1} \tag{5.83}$$

where

$$\mathbf{P}_{i,t} = \mathbf{A}_t \mathbf{P}_{\mathcal{L}_i,t} \mathbf{A}_t^\top + \mathbf{Q} \tag{5.84}$$

is the prediction error covariance matrix. If the scheduler R at time t allocates observations to node i, agent i receives the subset $\mathcal{O}_i$ of the measurements from the agents selected by R. Let

$$\mathbf{F}_{i,t} = \mathbf{P}_{i,t-1}^{-1} + \frac{1}{\sigma_i^2} \sum_{i_j \in \mathcal{L}_{i,t}} \mathbf{h}_{i_j} \mathbf{h}_{i_j}^\top \tag{5.85}$$

be the Fisher information matrix associated with the i^{th} node that determines the prior information and confidence of the node i before receiving the partial

observation set $\mathcal{O}_i$. The filtered error covariance matrix of the i^{th} node will then be updated according to

$$\mathbf{P}_{\mathcal{L}_i \cup \mathcal{O}_i} = \left(\mathbf{F}_{i,t} + \sum_{j_k \in \mathcal{O}_i} \frac{1}{\sigma_j^2} \mathbf{h}_{j_k} \mathbf{h}_{j_k}^\top \right)^{-1}. \tag{5.86}$$

The global MSE of the network at time t is defined as the sum of the MSEs of the individual nodes. In particular,

$$\text{MSE}_t = \sum_{i=1}^{m} \mathbb{E} \|\mathbf{x}_t - \hat{\mathbf{x}}_{i,t}\|^2, \tag{5.87}$$

where $\hat{\mathbf{x}}_{i,t}$ is the linear estimate of $\mathbf{x}_t$ computed by the i^{th} unit at time t. Since the MSE is equivalent to the trace of the filtered error covariance matrix,

$$\text{MSE}_t = \sum_{i=1}^{m} \text{Tr} \left(\left(\mathbf{F}_{i,t} + \sum_{j_k \in \mathcal{O}_i} \frac{1}{\sigma_j^2} \mathbf{h}_{j_k} \mathbf{h}_{j_k}^\top \right)^{-1} \right). \tag{5.88}$$

The amount of information allowed to be exchanged among the nodes of the network at any given time step is limited. More specifically, we assume that the subsets of partial observations $\{\mathcal{O}_i\}_{i=1}^m$ scheduled to be communicated to each agent should satisfy $\sum_{i=1}^m |\mathcal{O}_i| \leq K$, where K denotes the total number of observations that are allowed to be exchanged among the nodes of the network. The scheduler decides how to allocate measurements to individual nodes by solving the optimization problem

$$\begin{aligned} \min_{\mathcal{O}_i} \quad & \sum_{i=1}^{m} \text{Tr} \left(\left(\mathbf{F}_{i,t} + \sum_{j_k \in \mathcal{O}_i} \frac{1}{\sigma_k^2} \mathbf{h}_{j_k} \mathbf{h}_{j_k}^\top \right)^{-1} \right) \\ \text{s.t.} \quad & \mathcal{O}_i \subset \cup_{i=1}^m \mathcal{L}_i, \quad \forall i \in [m] \\ & \sum_{i=1}^{m} |\mathcal{O}_i| \leq K. \end{aligned} \tag{5.89}$$

A comparison of (5.89) to sensor selection problem [cf. Section 5.2] reveals that finding the optimal solution to (5.89) is generally NP-hard. In addition to being computationally challenging, optimization (5.89) does not necessary lead to a solution that would promote balanced MSE performance of the individual units; this point is illustrated later in our numerical results. To this end, we next add a regularization term to the objective function so as to promote balanced performance while still finding a near-optimal solution to the MSE estimation problem for the entire network.

5.5.2 Proposed formulation

Let $\mathcal{S} \subseteq \mathbf{X}$ where $\mathbf{X} = [m] \times [m] \times [\max_{i\in[m]} |\mathcal{L}_i|]$ is a ground set for the set function

$$f(\mathcal{S}) \quad = \quad \sum_{i=1}^{m} \mathrm{Tr}\left(\mathbf{F}_{i,t}^{-1}\right) \; - \; \mathrm{Tr}\left(\left(\mathbf{F}_{i,t} + \sum_{(i,j,k)\in\mathcal{S}} \frac{1}{\sigma_j^2}\mathbf{h}_{j_k}\mathbf{h}_{j_k}^{\top}\right)^{-1}\right). \quad (5.90)$$

The triplet (i, j, k) denotes that the k^{th} measurement of node j is communicated to node i. The function $f(\mathcal{S})$ is inversely related to the total MSE of the network. To arrive at a measurement exchange scheme that promotes balanced performance across the network units, we propose the optimization problem

$$\begin{aligned} \max_{\mathcal{S}} \quad & f(\mathcal{S}) + \gamma g(\mathcal{S}) \\ \text{s.t.} \quad & \mathcal{O}_i \subset \cup_{i=1}^{m} \mathcal{L}_i, \quad \forall i \in [m] \\ & \sum_{(i,-,-)\in\mathcal{S}} |\mathcal{O}_i| \leq K \\ & \mathcal{S} \subseteq [m] \times [m] \times [\max_{i\in[m]} |\mathcal{L}_i|], \end{aligned} \quad (5.91)$$

where

$$g(S) = \sum_{(i,-,-)\in\mathcal{S}} \log\left(1 + \frac{|\mathcal{O}_i|}{|\mathcal{L}_i|}\right) \tag{5.92}$$

is a regularization function and $\gamma \geq 0$ denotes the regularization parameter that determines the significance of balancing with respect to the goal of minimizing the total MSE of the entire network. On one hand, when $\gamma = 0$ the relay node R attempts to find a schedule that results in the lowest total MSE while disregarding potential imbalance in performance of the individual units. On the other hand, when γ is relatively large, the exchange of information determined by R is such that the differences between the MSEs of individual sensing nodes in the network become as small as possible. Notation $(i,-,-) \in \mathcal{S}$ in (5.91) and (5.92) implies that it does not matter for $g(\mathcal{S})$ which measurements are communicated to the i^{th} node; instead, it is the number of communicated measurements that is used to promote balanced performance.

Note that the proposed formulation (5.91) is an NP-hard combinatorial optimization problem, as it generalizes (5.7). However, as we show next, the proposed objective function $u(\mathcal{S}) = f(\mathcal{S}) + \gamma g(\mathcal{S})$ is monotone weak submodular, i.e., under some mild conditions it is characterized with a bounded multiplicative curvature. Hence, one can find an approximate solution to (5.91) using a greedy algorithm, as we state in the next section.

We proceed by providing two propositions to characterize the combinatorial properties of $f(\mathcal{S})$ and $g(\mathcal{S})$. For simplicity of the stated results, we assume that $\mathbf{R}_{i,t} = \sigma^2\mathbf{I}_{|\mathcal{L}_{i,t}|}$, i.e., use the same measurement noise statistics for all sensing nodes of the network (a generalization is straightforward).

Proposition 5.5.1. *Let* $\lambda_M = \max_{i\in[m]} \lambda_{max}(\mathbf{F}_{i,t})$, $\lambda_m = \min_{i\in[m]} \lambda_{min}(\mathbf{F}_{i,t})$, *and* $\mathbf{H}_t = [\mathbf{H}_{1,t}^\top, \ldots, \mathbf{H}_{1,m}^\top]^\top$. *Let* c_f *be the multiplicative curvature of* $f(\mathcal{S})$. *If*

$$\frac{1}{\sigma^2}\lambda_{max}(\mathbf{H}_t^\top \mathbf{H}_t) \leq \lambda_M, \tag{5.93}$$

then it holds that

$$c_f \leq \left(2\frac{\lambda_M}{\lambda_m}\right)^3. \tag{5.94}$$

Proof. Note that $f(\mathcal{S})$ is the sum of the additive inverse of the MSE of the sensing nodes that receive partial observations. Let i be one such node. Theorem 1 in [142] states that if

$$\frac{1}{\sigma^2}\lambda_{max}(\mathbf{H}_{i,t}^\top \mathbf{H}_{i,t}) \leq \lambda_{max}(\mathbf{F}_{i,t}), \tag{5.95}$$

then the multiplicative curvature of the additive inverse of the MSE of node i, c_{f_i}, satisfies

$$\begin{aligned} c_{f_i} &\leq \left(2\frac{\lambda_{max}(\mathbf{F}_{i,t})}{\lambda_{min}(\mathbf{F}_{i,t})}\right)^3 \frac{1+\sigma^2\lambda_{min}(\mathbf{F}_{i,t})}{1+2\sigma^2\lambda_{max}(\mathbf{F}_{i,t})} \\ &\leq \left(2\frac{\lambda_{max}(\mathbf{F}_{i,t})}{\lambda_{min}(\mathbf{F}_{i,t})}\right)^3. \end{aligned} \tag{5.96}$$

It is straightforward to see that the condition stated in (5.93) implies (5.95) and we have $\max_{i\in[m]} c_{f_i} \leq (2\frac{\lambda_M}{\lambda_m})^3$. Hence, definition of λ_M, λ_m, and $f(\mathcal{S})$ yields $c_f \leq \max_{i\in[m]} c_{f_i}$ which in turn completes the proof. ■

Proposition 5.5.2. *The set function* $g(\mathcal{S})$ *is a monotone submodular function.*

Proof. In order to prove the results, we first find the marginal gain $g_{(i,-,-)}(\mathcal{S})$ that in the following argument is denoted by $g_i(\mathcal{S})$ (with a slight abuse of notations for the sake of readability). By the definition of $g(\mathcal{S})$ and the marginal gain,

$$\begin{aligned} g_i(\mathcal{S}) &= \log\left(1+\frac{|\mathcal{O}_i|+1}{|\mathcal{L}_i|}\right) - \log\left(1+\frac{|\mathcal{O}_i|}{|\mathcal{L}_i|}\right) \\ &= \log\left(1+\frac{1}{|\mathcal{O}_i|+|\mathcal{L}_i|}\right). \end{aligned} \tag{5.97}$$

Since $\log(.)$ is a monotonically increasing function, $|\mathcal{O}_i| \geq 0$, $|\mathcal{L}_i| > 0$, $g_i(\mathcal{S}) > 0$ and hence $g(\mathcal{S})$ is monotone. We now prove the second part of the statement, i.e., submodularity of $g(\mathcal{S})$. Specifically, we should prove that the marginal gain of adding (i,j,k) to $\mathcal{S}$ is greater than adding it to a larger set $\mathcal{S} \cup \{(i',j',k')\}$ where $(i,j,k) \neq (i',j',k')$. Two cases might happen. First, assume that $i \neq i'$. Then,

$$g_i(\mathcal{S}) = g_i(\mathcal{S} \cup \{(i',j',k')\}) = \log\left(1+\frac{1}{|\mathcal{O}_i|+|\mathcal{L}_i|}\right). \tag{5.98}$$

Now assume $i = i'$. Then,

$$g_i(\mathcal{S} \cup \{(i',j',k')\}) = \log\left(1+\frac{1}{|\mathcal{O}_i|+|\mathcal{L}_i|+1}\right) < g_i(\mathcal{S}). \tag{5.99}$$

Combining (5.98) and (5.99) we conclude $g_i(\mathcal{S}) \geq g_i(\mathcal{S} \cup \{(i',j',k')\})$ which in turn implies submodularity. ■

By combining the results of Proposition 1 and Proposition 2, and by employing the matrix inversion lemma [178], we obtain the following theorem about the proposed objective function $u(\mathcal{S})$.

Theorem 5.5.1. *The utility set function $u(\mathcal{S})$ is a monotone, weak submodular function, $u(\emptyset) = 0$, and*

$$u_{(i,j,k)}(\mathcal{S}) = f_{(i,j,k)}(\mathcal{S}) + \gamma g_{(i,j,k)}(\mathcal{S}). \tag{5.100}$$

Proof. First note that it clearly holds that $u(\emptyset) = f(\emptyset) + \gamma g(\emptyset) = 0$. Furthermore, since $u(\mathcal{S}) = f(\mathcal{S}) + \gamma g(\mathcal{S})$ is the sum of a monotone weak submodular and a submodular function, it is also monotone weak submodular and $\mathcal{C}_u \leq (2\frac{\lambda_M}{\lambda_m})^3$. Finally, we introduce $g_{(i,j,k)}(\mathcal{S})$ in (5.97) and recursively find $f_{(i,j,k)}$ as

$$f_{(i,j,k)}(\mathcal{S}) = \frac{\mathbf{h}_{j_k}^\top \mathbf{F}_{i,\mathcal{S}}^{-2} \mathbf{h}_{j_k}}{\sigma_j^2 + \mathbf{h}_{j_k}^\top \mathbf{F}_{i,\mathcal{S}}^{-1} \mathbf{h}_{j_k}}, \tag{5.101}$$

$$\mathbf{F}_{i,\mathcal{S}\cup(i,j,k)}^{-1} = \mathbf{F}_{i,\mathcal{S}}^{-1} - \frac{\mathbf{F}_{i,\mathcal{S}}^{-1} \mathbf{h}_{j_k} \mathbf{h}_{j_k}^\top \mathbf{F}_{i,s}^{-1}}{\sigma_j^2 + \mathbf{h}_{j_k}^\top \mathbf{F}_{i,\mathcal{S}}^{-1} \mathbf{h}_{j_k}}, \tag{5.102}$$

where

$$\mathbf{F}_{i,\mathcal{S}} = \mathbf{F}_{i,t} + \sum_{(i,j,k)\in\mathcal{S}} \frac{1}{\sigma_j^2} \mathbf{h}_{j_k} \mathbf{h}_{j_k}^\top.$$

■

The analysis of combinatorial characteristics of the proposed utility set function reveals that the optimization problem in (5.91) is that of maximizing a monotone weak submodular set function subject to cardinality constraint. Therefore, in order to find a near-optimal scheduling of the observations exchange, we resort to their greedy selection. More specifically, at each time step t, the scheduler observes the performances of the local nodes and calculates

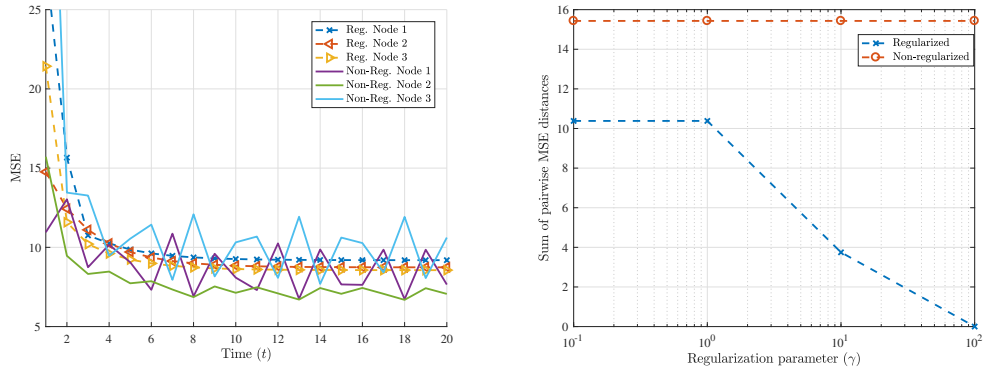
Algorithm 5 Greedy Information-Exchange Scheduling

1: **Input:** $\mathbf{P}_{i,t-1}$, $\mathbf{H}_{i,t}$, K, for $i = 1, \ldots, m$.
2: **Output:** Subset $\mathcal{S}_t \subseteq \mathbf{X}$ with $|S_t| = K$.
3: Initialize $\mathcal{S}_t = \emptyset$, $\mathbf{F}_{i,\mathcal{S}} = \mathbf{F}_{i,t}$ for $i = 1, \ldots, m$.
4: **for** $k = 1, \ldots, K$
5: $\quad (i, j, k) = \arg\max_{(i',j',k') \notin \mathcal{S}_t} u_{(i',j',k')}(\mathcal{S}_t)$.
6: $\quad$ Update $\mathcal{S}_t \leftarrow \mathcal{S}_t \cup \{(i, j, k)\}$.
7: $\quad$ Update $\mathbf{F}_{i,\mathcal{S}}^{-1} \leftarrow \mathbf{F}_{i,\mathcal{S}}^{-1} - \frac{\mathbf{F}_{i,\mathcal{S}}^{-1}\mathbf{h}_{j_k}\mathbf{h}_{j_k}(t)^\top\mathbf{F}_{i,\mathcal{S}}^{-1}}{\sigma_j^2 + \mathbf{h}_{j_k}^\top\mathbf{F}_{i,\mathcal{S}}^{-1}\mathbf{h}_{j_k}}$.
8: **end for**
9: **return** S_t.

the marginal gain of the possible distribution patterns (i, j, k) using (5.100). Then it adds the pattern yielding the highest marginal gain to the scheduling set $\mathcal{S}_t$ and updates the performance records $\mathbf{F}_{i,\mathcal{S}}$ for each node using (5.102). After repeating this procedure K times, the scheduler sends the instructions for the exchange of observations to the individual nodes.

The proposed method is formalized as Algorithm 5. Performance and complexity of the greedy algorithm are characterized by the following theoretical result which its proof is an immediate result of Proposition 2.3.2.

Theorem 5.5.2. *Let $\mathcal{C}_u$ be the multiplicative curvature of $u(\mathcal{S})$, i.e., the objective function of the balanced performance promoting scheduling problem in* (5.91). *Let $\{\mathcal{O}_i\}_{(i,-,-)\in\mathcal{S}}$ denote the set $\mathcal{S}$ of the observations selected to be communicated through the network by Algorithm 1 at time t, and let $\{\mathcal{O}_i^*\}_{(i,-,-)\in\mathcal{S}^*}$ be the optimal schedule of* (5.91) *such that $\sum_{(i,-,-)\in\mathcal{S}} |\mathcal{O}_i| \leq K$,*



(a) MSEs of the individual units

(b) Running time comparison.

Figure 5.10: Comparison of sum of pairwise node-level MSE distances

and $\sum_{(i,-,-)\in\mathcal{S}^*} |\mathcal{O}_i^*| \leq K$. *Then, it holds that*

$$u(\mathcal{S}) \geq (1 - e^{-\frac{1}{c}})u(\mathcal{S}^*), \tag{5.103}$$

where $c = \max\{1, \mathcal{C}_u\}$. *Furthermore, the computational complexity of Algorithm 5 is* $\mathcal{O}(Kmn^2 \sum_{i=1}^{m} |\mathcal{L}_{i,t}|)$.

5.5.3 Numerical experiments

In this section, we study the performance of the proposed algorithm in different scenarios. In particular, we simulate a fully connected network having 3 nodes, set the dimension of the state vector to $n = 50$, and assume that a scheduler is given information about the observation matrices of the individual nodes. For the state-transition matrix of the linear dynamical system, we set $\mathbf{A}_t = 0.8\mathbf{I}_n$ and randomly generate partial observation matrices $\mathbf{H}_{i,t}$. The observation patterns of the nodes vary with different runs; however, we preserve the rank of the matrices – in particular, $\text{rank}(\mathbf{H}_{1,t}) = 21$, $\text{rank}(\mathbf{H}_{2,t}) = 37$ and

rank($\mathbf{H}_{3,t}$) = 5. We assume a zero-mean Gaussian process noise and a zero-mean Gaussian observation noise at individual nodes with covariance matrices $\mathbf{Q} = 0.2\mathbf{I}_n$ and $\mathbf{R}_{i,t} = 0.05\mathbf{I}_n$, respectively. We run 10 Monte-Carlo simulations and select time horizons for each run as $T = 20$.

We first consider the MSE performances of the individual nodes in the network under regularized ($\gamma > 0$) and non-regularized ($\gamma = 0$) settings. The total number of measurements that can be exchanged among the units is set to $K = 40$. The regularization coefficients are set to $\gamma = 200$ and $\gamma = 0$; the large difference between the regularization coefficients will emphasize the effect of the balancing term on the individual node performance. In Fig. 5.10(a), we observe that the regularization term balances the individual node performances. We also observe that in the absence of regularization the nodes exhibit temporally rapidly varying MSE performance. This is primarily due to a deterministic nature of the greedy selection of the set of observations shared among the agents. In particular, when the regularization term is set to zero, at each time step the algorithm greedily schedules most of the observations to the node with the highest MSE. On the other hand, the non-zero regularization term ensures a temporally smoother and balanced MSE performances of the individual units.

Finally, we investigate the effect of the regularization parameter γ on balancing the individual performances of the nodes in the network. We set $K = 40$ and vary γ for the regularized network from 0.1 to 100 with log-scale increments. We compare the sum of pairwise MSE distances of the nodes in

the regularized and non-regularized networks in Fig. 5.10(b). We observe that the use of higher regularization coefficients results in a more balanced performances between individual nodes.

5.6 Conclusion

In this chapter, we studied the task of observation selection and information sharing in large-scale sensor networks where we relied on weak submodular optimization to designing efficient algorithms with provable guarantees.

First, we studied the problem of state estimation in large-scale linear time-varying dynamical systems where we provided a probabilistic theoretical bound on the multiplicative curvature of a monotone objective function that is inversely related to MSE criterion.

Next, we considered networked sensing systems following a (partially) quadratic measurement models. For this setting, we derived new optimality criteria by relying on the Van Trees' inequality and proved that they are monotone and (weak) submodular set functions. In particular, we showed that the $\log\det$ of the inverse of the Van Trees' bound is submodular while its trace is weak submodular under certain conditions on the unknown states, noise statistics, and the parameters of the model. Following these results, we developed an efficient greedy observation selection algorithm for networked sensing systems with theoretical bounds on its achievable utility that efficiently exploits the quadratic structure of the measurement model in its selection criteria.

We further proposed a randomized greedy algorithm for selecting sensors to query such that their choice minimizes the estimator's mean-square error at each time step. We established the performance guarantee for the proposed algorithm and analyzed its computational complexity.

Finally, we considered the task of distributed state estimation in a communication-constrained network of sensing units where in addition to minimizing the total mean-square error, a certain level of performance balancing is desired throughout the network. We formulated this task as the maximization of a monotone objective function subject to cardinality constraint. The proposed objective function is the sum of two monotone set functions: the first function, which is weak submodular, is inversely related to the total MSE of the network while the second one is submodular and favors a schedule of observation exchange that promotes balanced performance of individual units. Through a series of simulations, we demonstrated that the proposed formulation minimizes the total MSE of the network while balancing individual units performance.

Chapter 6

Compressed Decentralized Optimization via Multiple Gossip Steps

In this chapter, we study the problem of communication-efficient decentralized learning over networks where the goal of participating clients is to collaboratively optimize a global objective with a finite sum structure. Due to the fact that the learning task is large-scale, full communication among the clients is prohibitive. To this end, we propose DeLi-CoCo, a decentralized linearly convergent optimization algorithm. In DeLi-CoCo, each local gradient update is followed by *multiple* compressed communication steps to increase the capability of the clients to collaboratively accomplish the underlying learning task. We study the convergence property of our algorithm and show that by performing multiple compressed communication steps, DeLi-CoCo converges linearly to a near-optimal solution for smooth nonconvex objectives which satisfy the Polyak-Lojasiewicz condition. We show similar results hold for smooth and strongly convex problems. This convergence rate matches the same rate as that of decentralized gradient descent with no communication compression.

6.1 Introduction

We consider distributed optimization over a network with n client nodes where the objective function is possibly nonconvex. Formally, we are interested in

$$\min_{\mathbf{x}} \left[f(\mathbf{x}) := \sum_{i=1}^{n} f_i(\mathbf{x}) \right], \tag{6.1}$$

where $f_i : \mathbb{R}^d \to \mathbb{R}$ for $i \in [n] := \{1, ..., n\}$ is the local objective function of the i^{th} client. The goal of the clients in the network is to collaboratively solve the above optimization problem by passing messages over a graph that connects them [52].

The optimization task in (6.1) arises in many important distributed machine learning (ML) tasks, i.e., training and optimization of ML models in a distributed/decentralized manner [35–37]. Solving such distributed tasks is often facilitated by communication of agents' local model parameters over a network that governs their communication capabilities. Compared to a centralized optimization framework, distributed optimization enables locality of data storage and model updates which in turn offers computational advantages by delegating computations to multiple clients, and further promotes preservation of privacy of user information [35].

As the size of ML models grows, exchanging information across the network becomes a major challenge in distributed optimization [37]. It is therefore imperative to design communication-efficient strategies which reduce the amount of communicated data by performing compressed communication

while at the same time, despite the use of compressed communication, achieve a convergence properties that is on par with the performance of centralized and distributed methods utilizing uncompressed information [37–39].

We consider decentralized nonconvex ML tasks in a communication-constrained settings. In such scenarios, the clients may need to compress their local updates (using, e.g., quantization and/or sparsification) before transmitting them to their neighbors. Our goal is to establish a communication-efficient decentralized scheme with accelerated convergence rates for nonconvex tasks. Specific contributions of this chapter can be summarized as follows:

1. We propose **De**centralized **Li**near Learning with **Co**mmunication **Co**mpression (DeLi-CoCo), an iterative decentralized algorithm with arbitrary communication compression (both biased and unbiased compression operators) that performs multiple gossip steps in each iteration for faster convergence.

2. By employing $Q > 1$ steps of compressed communication after each local gradient update, DeLi-CoCo achieves a linear rate of convergence to a near-optimal solution for smooth nonconvex objectives satisfying the Polyak-Lojasiewicz condition. This rate matches the convergence rate of decentralized gradient descent (DGD) [53] with no communication compression. The proposed Q-step gossiping further helps to arbitrarily decrease the sub-optimality radius of the near-optimal solution, thereby improving upon the results of DGD [53].

3. We demonstrate that DeLi-CoCo compares favorably to centralized

and decentralized schemes without communication compression in a variety of convex and nonconvex learning tasks.

Table 6.1: Comparison of convergence rates of different decentralized optimization algorithms under smoothness. In the table, $\rho, \rho_1, \rho_2, \rho_3 \in (0, 1)$ denote the rate of linear convergence. α_1, α, and C depend on network and function properties. Further, $\alpha < 1$, $Q > 1$ is the number of rounds of consensus, and C depends on compression rate. In the table, SC stands for strong convexity.

Algorithm	Convergence	Setting	Compression
DGD [52]	$\mathcal{O}(1/T)$	SC, full gradient	✗
DGD [53]	$\mathcal{O}(\rho_1^T) + \alpha_1$	Restricted SC, full gradient	✗
EXTRA, SSDA [36, 54]	$\mathcal{O}(\rho_2^T)$	SC, full gradient	✗
DIGing [179]	$\mathcal{O}(\rho_3^T)$	SC, full gradient	✗
Choco-SGD [39]	$\mathcal{O}(1/T)$	SC, stochastic gradient	✓
This work	$\mathcal{O}(\rho^T) + C\alpha^Q$	PL condition, full gradient	✓

6.1.1 Significance and Related Work

Decentralized learning and optimization have drawn significant attention in the past few years due to the increasing importance of privacy and high data communication costs of centralized methods [38, 39, 60, 180–183]. Decentralized topologies overcome the aforementioned challenges by allowing each client to exchange messages only with their neighbors without exchanging their local data, showing great potential in terms of scalability and privacy-preserving capabilities.

Consensus with Compressed Communication. The study of decentralized optimization problems dates back to 1980s [184]. The main focus of early research in this area was on the task of average consensus where the goal of a

network is to find the average of local variables (i.e., agents' model vectors) in a decentralized manner. Conditions for asymptotic and non-asymptotic convergence of the decentralized average consensus in a variety of settings including directed and undirected time-varying graphs have been established in the seminal works [55,56,185–190]. Recently, a pioneering work [39] proposed the first communication-efficient average consensus/gossip algorithm that achieves a linear convergence rate and significantly improves the performance of existing quantized gossip methods [191–194]. In [39] a stochastic decentralized algorithm only for strongly convex and smooth objectives is further developed. Such linearly convergent gossip methods have also recently been extended to the scenario where the communication graph of agents is directed and time-varying [195,196]. In our work, we consider general nonconvex learning tasks and employ the proposed gossiping scheme of [39] as a subroutine. However, we propose a new decentralized algorithm with arbitrary compression that leverages multiple gossip steps to collaboratively solve nonconvex problems under the Polyak-Lojasiewicz condition [57,58].

Distributed Optimization with Compressed Communication. Distributed optimization is one of the richest topics at the intersection of machine learning, signal processing and control. Consensus/gossip algorithms have enabled distributed optimization of (non)convex objectives (e.g., empirical risk minimization) by modeling the task of decentralized optimization as noisy consensus. Examples include the celebrated distributed (sub)gradient descent algorithms (DGD) [52,53,197]. These schemes consider small-scale

problems where the clients can communicate uncompressed messages to their neighbors. Designing communication-efficient distributed optimization algorithms is an active area of research motivated by the desire to reduce the communication burden of multi-core and parallel optimization of ML models [60, 61, 198]. Majority of the existing works consider distributed optimization tasks with master-slave architectures where the compression of communication is accomplished by using methods based on sparsification or quantization of gradients [60, 62, 198–201]. An example of such a setting is federated learning [35, 202] which enables distributed learning of an ML model in a cloud while the training data remains distributed across a large number of clients. Recent federated learning schemes that promote communication efficiency either focus on compressing the size of the client-to-cloud messages or decreasing the number of communication rounds [180, 203–207]. In contrast to that line of work, we consider a general decentralized learning scenario and exploit the error feedback mechanism of [60, 199, 200] as part of our proposed scheme to enable arbitrary compression while maintaining a linear convergence rate. Additionally, unlike a majority of decentralized optimization schemes including those with uncompressed communication that require strong convexity to a achieve linear rate, e.g. [36, 53, 54, 179] – except the recent results in [208, 209] with full communication – we only assume the Polyak-Lojasiewicz condition which enables us to analyze nonconvex learning tasks. To our knowledge, the proposed algorithm is the first scheme to achieve linear convergence in the decentralized setting with compressed communication under the Polyak-

Lojasiewicz condition.

6.2 Multi-step Gossip Decentralized Gradient Descent

In this section, we present our proposed algorithm for solving (2.24) iteratively in a decentralized manner where the agents are restricted to communicate compressed information.

The proposed DeLi-CoCo scheme (see Algorithm 6) consists of two main subroutines: (i) update of the local variable $\mathbf{x}_i$ via gradient descent, and (ii) exchange of compressed messages between neighboring clients by performing $Q \geq 1$ compressed gossiping steps.

Let $t = 1, \ldots, T$ denote the t^{th} iteration of DeLi-CoCo and let $q = 0, \ldots . Q-1$ denote the q^{th} compressed gossiping/consensus step. Each client i maintains three local variables: $\mathbf{x}_{t,i}^{(q)}$, $\mathbf{z}_{t,i}^{(q)}$, and $\mathbf{s}_{t,i}^{(q)}$. Here, $\mathbf{x}_{t,i}^{(q)}$ denotes the vector of current local parameters of node i, while $\mathbf{z}_{t,i}^{(q)}$, and $\mathbf{s}_{t,i}^{(q)}$ are maintained locally to keep track of the compression noise and be used as an error feedback for subsequent iterations, respectively [39, 60].

At iteration t, each client updates its own local parameters by performing a simple gradient descent update according to

$$\mathbf{x}_{t,i}^{(0)} = \mathbf{x}_{t-1,i}^{(Q)} - \eta \nabla f_i(\mathbf{x}_{t-1,i}^{(Q)}), \tag{6.2}$$

where $\eta > 0$ is a constant learning rate specified in Theorem 6.3.1. Following the gradient update, we propose to perform Q compressed gossiping steps in a decentralized manner to further update the local parameters as well as the

error feedback variables. Intuitively, this Q-step procedure is a crucial part of DeLi-CoCo that enables updated parameters $\mathbf{x}_{t,i}^{(0)}$ to converge to their average value.

To perform the $(q+1)^{\text{st}}$ gossiping step, each node generates the message $\mathcal{C}(\mathbf{x}_{t,i}^{(q)}-\mathbf{z}_{t,i}^{(q)})$, where $\mathcal{C}:\mathbb{R}^d\to\mathbb{R}^d$ denotes the (potentially random) compression operator, and $\mathbf{z}_{t,i}^{(q)}$ is a parameter that keeps track of the compression error. The compressed message $\mathcal{C}(\mathbf{x}_{t,i}^{(q)}-\mathbf{z}_{t,i}^{(q)})$ is communicated to update $\mathbf{s}_{t,i}^{(q)}$, and then it is further used by the transmitting client as an error feedback to update $\mathbf{z}_{t,i}^{(q)}$:

$$\mathbf{s}_{t,i}^{(q+1)}=\mathbf{s}_{t,i}^{(q+1)}+\sum_{j=1}^{n}w_{ij}\mathcal{C}(\mathbf{x}_{t,j}^{(q)}-\mathbf{z}_{t,j}^{(q)}),\quad \mathbf{z}_{t,i}^{(q+1)}=\mathbf{z}_{t,i}^{(q)}+\mathcal{C}(\mathbf{x}_{t,i}^{(q)}-\mathbf{z}_{t,i}^{(q)}). \tag{6.3}$$

Intuitively, $\mathbf{z}_{t,i}^{(q)}$ and the error feedback mechanism enable all the local information to be transmitted eventually with a delay that depends on the compression operator $\mathcal{C}$.

Then, to accomplish the $(q+1)^{\text{st}}$ gossiping step, each client performs

$$\mathbf{x}_{t,i}^{(q+1)}=\mathbf{x}_{t,i}^{(q)}+\gamma(\mathbf{s}_{t,i}^{(q+1)}-\mathbf{z}_{t,i}^{(q+1)}), \tag{6.4}$$

where $0<\gamma\leq 1$ is the gossiping/consensus learning rate whose exact value will be specified in Theorem 6.3.1. After performing compressed gossiping for Q steps, the t^{th} iteration of DeLi-CoCo is complete.

The above update rules are summarized in Algorithm 6 where we use an equivalent and useful matrix notation where $\mathbf{x}_{t,i}^{(q)}$, $\mathbf{s}_{t,i}^{(q)}$, and $\mathbf{z}_{t,i}^{(q)}$ are stored as the i^{th} column of $\mathbf{X}_t^{(q)}$, $\mathbf{S}_t^{(q)}$, and $\mathbf{Z}_t^{(q)}$, respectively.

Algorithm 6 The DeLi-CoCo Algorithm

Input: stepsize η, consensus stepsize γ, number of gradient iterations T, number of consensus steps per gradient iteration Q, mixing matrix W; initialize $\mathbf{X}_0^{(Q)}$, $\mathbf{Z}_0^{(0)} = \mathbf{S}_0^{(0)} = \mathbf{0}$.
for $t = 1, \dots, T$
  $\mathbf{X}_t^{(0)} = \mathbf{X}_{t-1}^{(Q)} - \eta \nabla F(\mathbf{X}_{t-1}^{(Q)})$ (local gradient update)
  for $q = 0, 1, \dots, Q-1$
    $\mathbf{S}_t^{(q+1)} = \mathbf{S}_t^{(q)} + \mathcal{C}(\mathbf{X}_t^{(q)} - \mathbf{Z}_t^{(q)})\mathbf{W}$ (Exchanging messages)
    $\mathbf{Z}_t^{(q+1)} = \mathbf{Z}_t^{(q)} + \mathcal{C}(\mathbf{X}_t^{(q)} - \mathbf{Z}_t^{(q)})$ (Compression error feedback)
    $\mathbf{X}_t^{(q+1)} = \mathbf{X}_t^{(q)} + \gamma(\mathbf{S}_t^{(q+1)} - \mathbf{Z}_t^{(q+1)})$ (Local gossip update)
  end for
  $\mathbf{Z}_{t+1}^{(0)} = \mathbf{Z}_t^{(Q)}$, $\mathbf{S}_{t+1}^{(0)} = \mathbf{S}_t^{(Q)}$
end for

Remark 6.2.1. Let $Q = 1$, $\gamma = 1$, and assume there is no compression, i.e. $\mathcal{C}(\mathbf{x}_{t,i}^{(q)} - \mathbf{z}_{t,i}^{(q)}) = \mathbf{x}_{t,i}^{(q)} - \mathbf{z}_{t,i}^{(q)}$. Then the proposed DeLi-CoCo scheme reduces to the DGD [53]. If $Q = 1$, $\eta = \mathcal{O}(1/T)$, and clients perform local stochastic gradient updates, the proposed scheme reduces to Choco-SGD [39]. As we will discuss in Section 4, by performing $Q > 1$ gossiping steps, DeLi-CoCO achieves a convergence rate that compares favorably with DGD.

6.3 Convergence Analysis

In this section we analyze the convergence properties of DeLi-CoCo. Recall that $\|\mathbf{X}_t^{(q)} - \mathbf{X}^*\|^2 = \sum_{i=1}^n \|\mathbf{x}_{t,i}^{(q)} - \mathbf{x}^*\|_2^2$ where $\mathbf{x}^* \in \mathcal{X}^* := \operatorname{argmin} f(\mathbf{x})$ and $F(\mathbf{X}^*) = f(\mathbf{x}^*) := f^*$. Equivalently, we refer to $\mathcal{X}^*$ as the set of optimal points of f.

Theorem 6.3.1. *Assume Assumptions 2.4.1, 2.4.2,2.4.4, and 2.4.5 hold. De-*

fine

$$\begin{aligned}\beta(\delta,\omega) &:= 16\delta+\delta^2-8\delta\omega+(4+2\delta)\lambda_{\max}^2(\mathbf{I}-\mathbf{W}),\\ \Delta^2 &:= \max_{\mathbf{x}^*\in\mathcal{X}^*}\sum_{i=1}^{n}\|\nabla f_i(\mathbf{x}^*)\|_2^2, \quad R_0 := F(\mathbf{X}_0^Q)-f^*,\\ \rho &:= 1-\frac{2\mu}{n\hat{L}}+\frac{\mu L}{n\hat{L}^2}<1-\frac{\mu}{n\hat{L}}, \quad \xi := 23\left(1-\frac{\delta\gamma}{2}\right)^Q,\end{aligned} \tag{6.5}$$

where $\mathcal{X}^* := \arg\min f(\mathbf{x})$*, and* δ *and* ω *are the spectral gap of* W *and the level of compression, respectively. Let the parameters of DeLi-CoCo satisfy*

$$\eta=\frac{1}{\hat{L}}, \quad 0<\gamma=\frac{\delta\omega}{\beta(\delta,\omega)}<1, \quad Q>\left\lceil\left(\log\rho-\log 24\right)\Big/\log\left(1-\frac{\delta\gamma}{2}\right)\right\rceil. \tag{6.6}$$

Then, after T *iterations, the iterates of DeLi-CoCo satisfy*

$$\begin{aligned}&\mathbb{E}_{\mathcal{C}}[F(\mathbf{X}_T^{(Q)})]-f^* \le \frac{21\Delta^2}{\hat{L}(1-\xi)}\left(1-\frac{\delta\gamma}{2}\right)^{\frac{Q}{2}}\\ &+\rho^T\left(\left[\frac{1-\frac{\mu}{n\hat{L}}+\frac{L}{2\mu}}{\rho}+\frac{13L\sqrt{1-\xi}}{2\mu(\rho-\xi)}\left(1-\frac{\delta\gamma}{2}\right)^{\frac{Q}{2}}\right]R_0+\frac{\|\mathbf{X}_0^{(Q)}\|^2\hat{L}\sqrt{1-\xi}}{\left(1-\frac{\delta\gamma}{2}\right)^{\frac{Q}{2}}}\right).\end{aligned} \tag{6.7}$$

Additionally, if the nodes are initialized such that $\mathbf{X}_0^{(Q)}=0$*, by considering the dominant terms in the expression above we have*

$$\begin{aligned}&\mathbb{E}_{\mathcal{C}}[F(\mathbf{X}_T^{(Q)})]-f^*\\ &=\mathcal{O}\left(\left(1+\frac{L}{\mu}\frac{\sqrt{1-\xi}}{\rho-\xi}\left(1-\frac{\delta\gamma}{2}\right)^{\frac{Q}{2}}\right)R_0\rho^T+\frac{\Delta^2}{\hat{L}\sqrt{1-\xi}}\left(1-\frac{\delta\gamma}{2}\right)^{\frac{Q}{2}}\right),\end{aligned} \tag{6.8}$$

where the $\mathcal{O}$ *notation does not hide any terms depending on* Q *or* T*.*

Proof. See Appendix C.1. ■

Remark 6.3.1. We highlight the following observations:

1. Comparison to DGD: With no compression, using tighter analysis the term $(1-\delta\gamma/2)$ can be further improved to $1-\gamma\delta$ [39,55]. In this setting, with $\gamma = 1$ we may compare our result to the prior work in [53,210]. First, in contrast to [53,210], our analysis is carried out under PLC without assuming (restricted) strong convexity. The radius of the near-optimal neighborhood in [53] (see Theorem 4 there) is proportional to Δ/δ while in our case, by using the proposed Q- step compressed gossiping procedure, the radius is proportional to $\Delta^2(1-\delta)^{\frac{Q}{2}}$; in fact, we can make the bound arbitrarily small by performing a sufficiently large number of gossiping steps Q (see Corollary 6.3.1.1).

2. Effect of Compression: Our results reveal that compression of messages using contraction operators can be thought of as weakening the connectivity property of the communication graph by inducing spectral gap $\delta' = \delta\omega$. As ω approaches zero, the consensus learning rate decreases. Hence, as per intuition, a larger Q is required to satisfy the conditions in the statement of Theorem 6.3.1.

3. Almost Linear Convergence: Our analysis further reveals that at the cost of increased number of rounds of communication, suboptimality can be arbitrary reduced. In particular, $\mathbb{E}_{\mathcal{C}}[F(\mathbf{X}_t^{(Q)})] - f^* \leq \epsilon$ accuracy can be achieved after $\mathcal{O}(\log^2(1/\epsilon))$ rounds of communication by setting $Q = T = \log(1/\epsilon)$. However, in practice it suffices to use a small Q to achieve a competitive performance compared to centralized and decentralized schemes with no compression.

4. Power of Overparameterization: Consider the case that (6.1) corresponds to a decentralized regression or classification task wherein the model architecture is expressive enough to completely fit or *interpolate* the training data distributed among the clients [211–213], e.g. in the case of over-parameterized neural networks or functions satisfying a certain growth condition [214, 215]. Then any stationary point of f will also be a stationary point of each of the f_i's and thus $\Delta^2 = 0$. Therefore, in the this setting and under PLC, Deli-CoCo converges exactly at a linear rate of $\mathcal{O}(\log(1/\epsilon))$ by setting Q to be a constant independent of ϵ.

Corollary 6.3.1.1. *Instate the notation and hypotheses of Theorem 6.3.1. Then, in order to achieve* $\mathbb{E}_{\mathcal{C}}[F(\mathbf{X}_T^{(Q)})] - f^* \leq \epsilon$, *Deli-CoCo requires* $\tau = \mathcal{O}(\log^2(1/\epsilon))$ *rounds of communication if* $\Delta \neq 0$, *and* $\tau = \mathcal{O}(\log(1/\epsilon))$ *if* $\Delta = 0$.

To our knowledge, DeLi-CoCo is the first algorithm attaining a linear convergence rate for decentralized nonconvex optimization with compressed communication in the interpolation regime.

5. Implications for Federated Learning: Theorem 6.3.1 also implies a near linear convergence rate for federated learning tasks – in their simplest form – satisfying PLC under compressed communication. This scenario corresponds to a decentralized learning problem over a network with $\delta = 1$ [216] under which DeLi-CoCo efficiently delivers a stationary solution. Nonetheless, there are some open problems and issues such as delayed communication and intermittent client availability in federated learning [217] that are not considered here.

Since PLC is implied by strong convexity, Theorem 6.3.1 provides a convergence rate for strongly convex and smooth objectives. We can make this more explicit in Theorem 6.3.2 below.

Theorem 6.3.2. *Assume Assumptions 2.4.1, 2.4.2,2.4.3, and 2.4.5 hold. Define*

$$\ell := (1-\eta\hat{\mu})^2 < 1, \quad \xi := (1-\frac{\delta\gamma}{2})^Q(3+20\eta^2\hat{L}^2) \tag{6.9}$$

Let the parameters of DeLi-CoCo satisfy

$$\begin{aligned} &\gamma = \frac{\delta\omega}{\beta(\delta,\omega)}, \quad \eta \leq \min\{\frac{2}{\hat{L}+\hat{\mu}}, \frac{2}{L+\mu}\}, \\ &Q > \left\lceil \left(\log\ell - \log(3+20\eta^2\hat{L}^2) \right) \Big/ \log\left(1-\frac{\delta\gamma}{2}\right) \right\rceil. \end{aligned} \tag{6.10}$$

Then, after T iterations, the iterates of DeLi-CoCo satisfy

$$\begin{aligned} \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_T^{(Q)} - \mathbf{X}^*\|^2 \leq &\left[\frac{T}{\ell}\left(\frac{13L\eta^2}{\ell(\ell-\xi)}\left(1-\frac{\delta\gamma}{2}\right)^Q D_0 + \|\mathbf{X}_0\|^2\right) + D_0\right]\ell^T \\ &+ \frac{20\eta^2\Delta^2}{(1-\ell)(1-\xi)}\left(1-\frac{\delta\gamma}{2}\right)^Q, \end{aligned} \tag{6.11}$$

Additionally, if the nodes are initialized such that $\mathbf{X}_0^Q = 0$, by considering the dominant terms in the expression above we have

$$\begin{aligned} &\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_T^{(Q)} - \mathbf{X}^*\|^2 \\ &= \mathcal{O}\left(\left[1+\frac{TL\eta^2}{\ell(\ell-\xi)}\left(1-\frac{\delta\gamma}{2}\right)^Q\right]D_0\ell^T + \frac{\eta^2\Delta^2}{(1-\ell)(1-\xi)}\left(1-\frac{\delta\gamma}{2}\right)^Q\right), \end{aligned} \tag{6.12}$$

where the $\mathcal{O}$ notation does not hide any terms depending on Q or T.

Proof. See Appendix C.2. ■

6.4 Numerical Experiments

We show the effectiveness of DeLi-CoCo on three common machine learning problems - logistic regression, linear regression and non-linear regression. Following [39], for all the experiments we plot the sub-optimality, i.e. $f(\bar{\mathbf{x}}_t) - f^*$ against the number of local gradient computations (or steps). Here, f^* is the optimal value obtained by running vanilla gradient descent with the entire data on a single machine – we shall refer to this setting as "Centralized GD" throughout this section. We consider the compression schemes proceeding the statement of Assumption 2.4.5, and explore some network topologies commonly used in literature, namely ring, torus, fully-connected and disconnected topologies (see, e.g. [39, 55]). All plots are averaged over 3 independent runs.

Datasets: Let $\{s_1^{(i)}, \ldots, s_{n_i}^{(i)}\}$ denote the samples being processed in the i^{th} node where n_i is the total number of samples in the i^{th} node. Then, $f_i(\mathbf{x}) = \frac{1}{n_i} \sum_{j=1}^{n_i} \ell(\mathbf{x}, s_j^{(i)})$, where $\ell(.)$ denotes the loss function of the regression tasks that we explain next.

(i) Logistic regression: We use a binary version of MNIST [218] where the first five classes are treated as class 0 and the rest as class 1. We train a classifier with the binary cross-entropy loss. We consider a decentralized setting where the data is evenly distributed among all the nodes in a challenging sorted setting (sorted based on labels) where at most one node acquires examples from both classes.

(ii) Linear regression: We train a linear regression model on $m = 10000$ synthetic data samples $\{(\mathbf{a}_i, y_i)\}_{i=1}^m$ generated according to $y_i = \langle \boldsymbol{\theta}^*, \mathbf{a}_i \rangle + e_i$, where $\boldsymbol{\theta}^* \in \mathbb{R}^{2000}$, the i^{th} input $\mathbf{a}_i \sim \mathcal{N}(\mathbf{0}, I_{2000})$, and noise $e_i \sim \mathcal{N}(0, 0.05)$. We refer to this synthetic dataset as SYN-1. Here, we use the squared loss function with ℓ_2-regularization value $= 0.001$.

(iii) Non-linear regression: We train a non-linear regression model on $m = 10000$ synthetic data samples $\{(\mathbf{a}_i, y_i)\}_{i=1}^m$ generated as $y_i = \text{relu}(\langle \boldsymbol{\theta}^*, \mathbf{a}_i \rangle) + e_i$, where $\boldsymbol{\theta}^* \in \mathbb{R}^{2000}$, the i^{th} input $\mathbf{a}_i \sim \mathcal{N}(\vec{0}, I_{2000})$, $e_i \sim \mathcal{N}(0, 0.05)$ and $\text{relu}(z) = \max(z, 0)$ (i.e. the standard ReLU function). We call this synthetic dataset SYN-2 henceforth. We model this task as training a one-layer neural network having ReLU activation with the squared loss function and ℓ_2-regularization value $= 0.001$.

Importance of Q. The backbone of DeLi-CoCo is the introduction of performing Q mixing steps. To better understand the role of this stage under communication compression, in Fig. 6.1 we depict the performance of DeLi-CoCo for different values of Q under varied consensus learning rates γ where we compare its performance to centralized GD without compression. Our experiments suggest that DeLi-CoCo can achieve the same rate of convergence as Centralized GD for suitable values of Q, thereby illustrating the value of having multiple mixing steps. We further observe that with higher γ, DeLi-CoCo requires fewer mixing steps to match the performance of Centralized GD.

Impact of Topology. Fig. 6.2 shows the effect of topology and size of

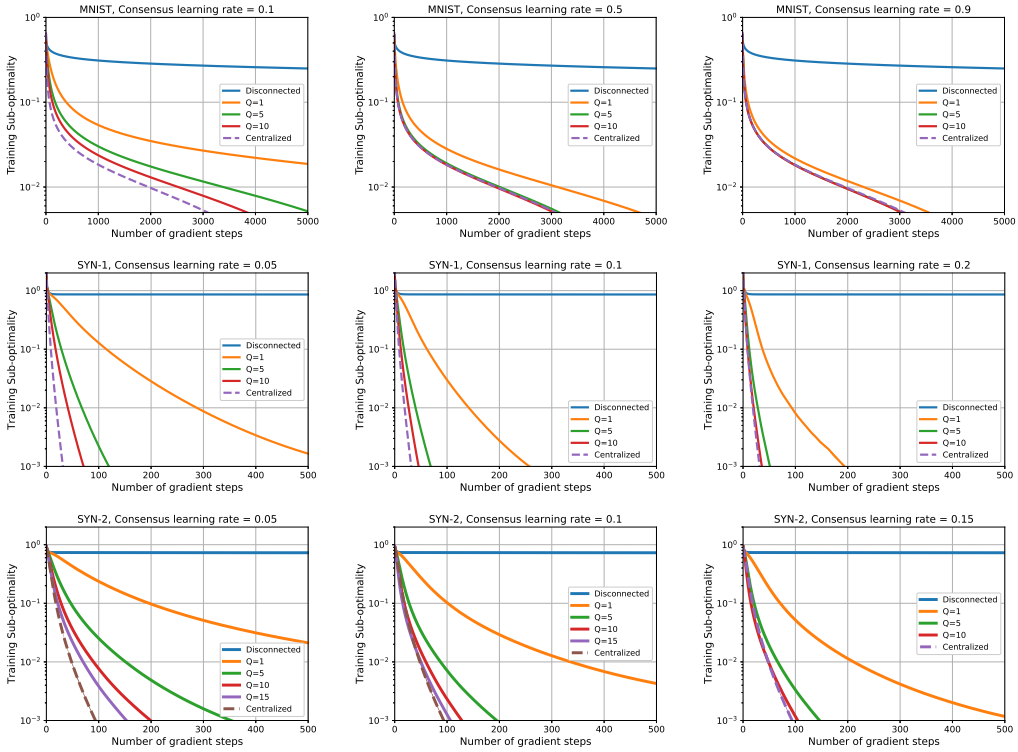


Figure 6.1: Effect of varying Q under different consensus learning rates γ. MNIST setting (top row): $n = 9$, top(0.05); SYN-1 setting (mid row): $n = 16$, qsgd_2; SYN-2 (bottom row): $n = 16$, qsgd_2, ℓ_2-regularization value $= 0.001$. We used $\eta = 0.2$ and torus topology for all datasets.

the network on the convergence of DeLi-CoCo under communication compression. We repeatedly observe that fully connected topology outperforms torus which is followed by the ring topology, consistent with the intuition that better connectivity leads to faster convergence. Notice that increasing the number of nodes worsens the convergence across all topologies. However, this effect is less severe for the networks with stronger connectivity properties. This observation is consistent with the intuition that communication compression leads to weakening of network connectivity. Also note that increasing Q improves

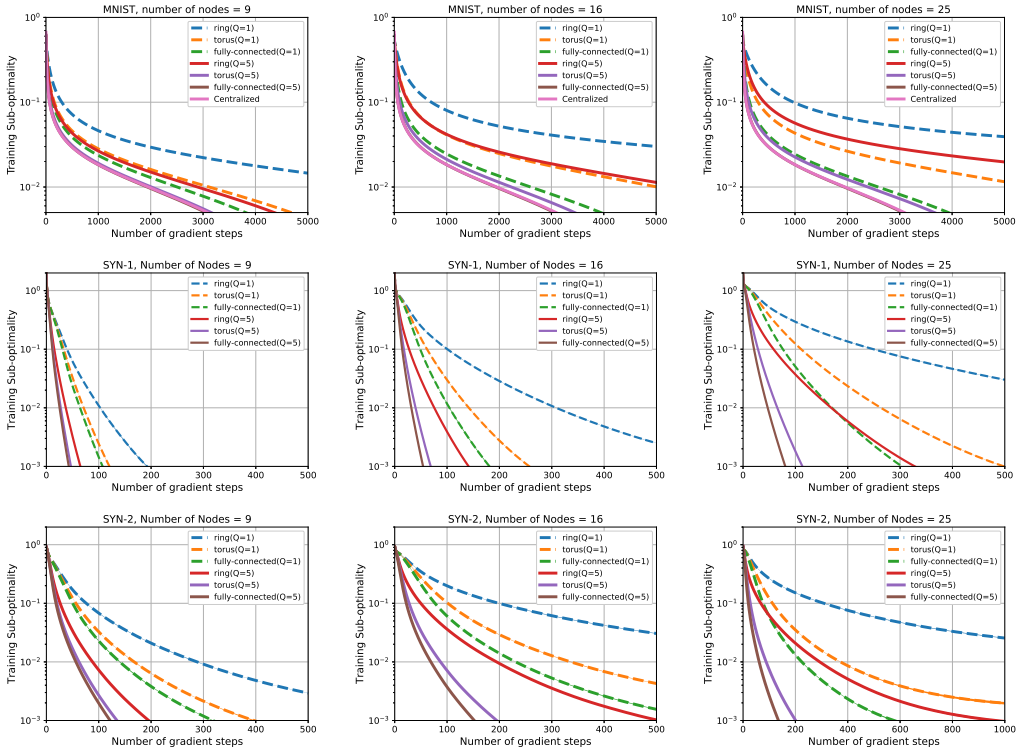


Figure 6.2: Effect of network topology settings on the convergence rates; showing dependence on the number of nodes n and mixing steps Q. MNIST setting(top row): top(0.05), $\gamma = 0.5$, $\eta = 0.2$. SYN-1 setting (bottom row): qsgd_2, $\gamma = 0.1$; We used $\eta = 0.2$ for $n = 9, 16$ and $\eta = 0.15$ for $n = 25$, respectively. SYN2 setting (bottom row): qsgd_2, $\eta = 0.2$, ℓ_2-regularization parameter $= 0.001$; We used $\gamma = 0.1$ for $n = 9, 16$ and $\gamma = 0.15$ for $n = 25$.

the rate of convergence, reinforcing our earlier observations in Fig. 6.1.

Impact of Compression Operator. In Fig. 6.3, we show the effect of different compression schemes on the rate of convergence. In particular, we compare DeLi-CoCo with the DGD [52, 53] that operates under no communication compression (i.e., full communication). Note that with majority of the compression schemes, DeLi-CoCo achieves similar rates as DGD.

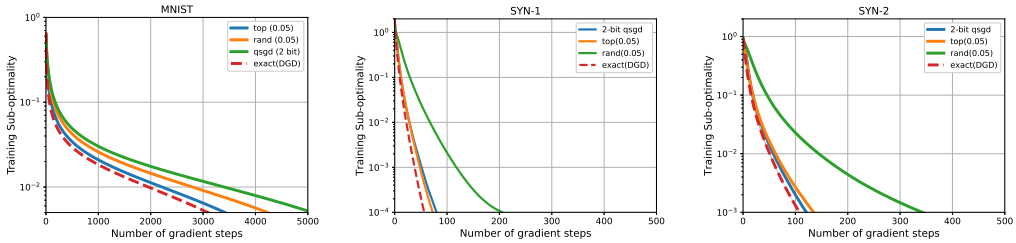


Figure 6.3: Comparison of various compression operators over torus topology. MNIST (Left): $n = 9, \eta = 0.2$; For qsgd_2, top(0.05), rand(0.05), we used $Q = \{10, 15, 15\}$ and $\gamma = \{0.05, 0.1, 0.05\}$, respectively. SYN-1 (Middle): $n = 16, \eta = 0.2$, $Q = 5$; For qsgd_2, top(0.05), and rand(0.05), we used $\gamma = 0.2$, 0.2 and 0.05, respectively. SYN2 (Right): $n = 16, \eta = 0.2$, $Q = 5$, ℓ_2-regularization parameter $= 0.001$; For qsgd_2, top(0.05), and rand(0.05) we used $\gamma = 0.2$, $\gamma = 0.2$, and $\gamma = 0.05$, respectively.

6.5 Conclusion

In this chapter, we considered decentralized nonconvex ML tasks in a communication-constrained settings where the clients may need to compress their local updates before transmitting them to their neighbors. We proposed (DeLi-CoCo), an iterative decentralized algorithm with arbitrary communication compression (both biased and unbiased compression operators) that performs multiple gossip steps in each iteration for faster convergence. By employing $Q > 1$ steps of compressed communication after each local gradient update, DeLi-CoCo achieves a linear rate of convergence to a near-optimal solution for smooth nonconvex objectives satisfying the Polyak-Lojasiewicz condition. The proposed Q-step gossiping further helps to arbitrarily decrease the sub-optimality radius of the near-optimal solution. We further showed that a similar convergence rate for smooth and strongly convex objectives. Finally, we demonstrated that DeLi-CoCo compares favorably to schemes without communication compression in a variety of learning tasks.

Chapter 7

Conclusion and Future Work

The aim of this thesis dissertation is to develop and analyze efficient methods for inference and learning from contemporary large-scale datasets. These large-scale and high-dimensional datasets have hidden low-dimensional structures, e.g. sparsity, and are gathered by a network of resource constrained systems capable of exchanging information.

7.1 Conclusions

The first contribution of this dissertation was two-fold: we first studied the task of sparse reconstruction and support selection where we proposed two efficient greedy algorithms and theoretically established the conditions for their exact reconstruction performance in a variety of settings. Then, we showed that the proposed algorithms can be utilized in structured data clustering problems where the data is a collection of points lying on a union of low-dimensional and evolving subspaces. To this end, we proposed a non-convex optimization framework that exploits the self-expressiveness property of the evolving data while taking into account representation from the preceding time step. To find an approximate solution to the aforementioned non-convex opti-

mization problem, we developed a scheme based on alternating minimization that both learns the parsimonious representation as well as adaptively tunes and infers a smoothing parameter reflective of the rate of data evolution.

As a second contribution, we focused on observation selection and information gathering in networks where we studied state-estimation tasks of dynamically-evolving systems through large-scale sensor networks. We established that the mean-square error criterion is weak submodular in networks governed by a linear observation model. We further proposed a new weak submodular observation selection criterion by relying on the Van Trees' inequality. Additionally, we proposed efficient greedy observation selection and communication scheduling schemes and established their near-optimal performance.

As a final contribution, we devised a new optimization algorithm for collaborative learning problems where the communication among the participating agents is limited. The proposed algorithm leverages multiple compressed communication steps as well as a compression error feedback mechanism to accomplish the learning task. We further analyzed the performance of the proposed scheme and established that it achieves near-optimal convergence rate for general nonconvex learning tasks that satisfy the Polyak-Lojasiewicz condition.

7.2 Future Work

As part of future work for the sparse reconstruction and support selection problems discussed in Chapter 3, it would be valuable to extend the

presented analysis which was performed under the assumption that the sensing matrix is distributed according to a zero mean Gaussian distribution, and study performance of AOLS and PSG for hybrid dictionaries [83]. It is also of interest to analytically characterize performance of the AOLS-based sparse subspace clustering scheme using the techniques established in [2, 99]. Additionally, it is of interest to determine whether $\mathcal{O}(m(\log k))$ is in fact the minimum number of oracle calls required to achieve the optimal sample complexity for greedy sparse reconstruction algorithms. A trivial lower bound, as argued by [67], is $\mathcal{O}(m)$.

The evolutionary subspace clustering framework and the CESM algorithm that we developed in Chapter 4 pave the way towards interesting future research directions. Firstly, it would be of interest to extend the CESM framework to other subspace clustering algorithms, including the schemes that rely on finding low rank representations of data points (see, e.g. [219]). It is also valuable to exploit the theoretical foundation of subspace clustering established by [2, 99] to analyze the performance of the proposed frameworks, e.g., in the setting of rotating random subspaces that we considered in Chapter 4. Finally, it would be of interest to develop more complex models for the evolutionary subspace clustering problem, e.g., by using neural networks as the parametric function or a matrix of smoothing parameters in place of the proposed convex evolutionary self-expressive model. An interesting effort in this direction has recently been made by [220].

Contributions made in Chapter 5 on the topic of weak submodular ob-

servation selection and information gathering in communication constrained networks were based on the consideration that the selected subset of information can be communicated exactly to the corresponding receiver nodes in the network. Often, as it has been argued in [149], the information might be lost due to communication and link failures in the network. Thus, it would be of interest to study robust variants of the methodologies developed in Chapter 5 by building upon the techniques developed in [161, 221]. Another important direction for future research is to design weak submodular communication scheduling approaches that trade performance for increased privacy and security of the communicated information. Such algorithms entail integrating privacy-preserving concepts such as differential privacy [222] to guarantee the robustness of the algorithms to malicious activities.

The collaborative learning algorithm that we developed in Chapter 6 requires access to the full local gradients. It would be of interest to extend the proposed algorithm to the scenario where only a stochastic and unbiased estimate of the true gradient is available to each local client. Furthermore, the analysis carried out in Chapter 6 requires the global objective to satisfy the Polyak-Lojasiewicz condition in the nonconvex case and strongly convex assumption in the convex setting. It would be valuable to theoretically establish the convergence rate of the proposed algorithm under milder conditions. Finally, extensions to time-varying and directed network topologies as well as incorporating momentum techniques such as [59] for improved theoretical and empirical performance is another interesting research direction.

Appendices

Appendix A

Missing Proofs from Chapter 3

A.1 Useful Lemmas

We first state a number of intermediate lemmas. Lemma A.1.1 states that the Euclidean norm of a normally distributed vector is concentrated around its expected value.

Lemma A.1.1. *Let* $\mathbf{a} \in \mathbb{R}^n$ *be a vector consisting of entries that are drawn independently from* $\mathcal{N}(0, 1/n)$*. Then it holds that* $\mathbb{E}\left\|\mathbf{P}(\mathcal{S}^{(k)})\mathbf{u}\right\|_2^2 = \frac{k}{n}\mathbb{E}\left\|\mathbf{u}\right\|_2^2$*. Furthermore, one can show that*

$$\Pr\left((1-\gamma)\frac{k}{n} < \left\|\mathbf{P}(\mathcal{S}^{(k)})\mathbf{u}\right\|_2^2 < (1+\gamma)\frac{k}{n}\right) \geq 1 - 2e^{-nc_0(\gamma)}, \tag{A.1}$$

where $c_0(\gamma) = \frac{\gamma^2}{4} - \frac{\gamma^3}{6}$ *for* $0 < \gamma < 1$*.*

Proof. The lemma aims to characterize the length of the projection of a random vector onto a low-dimensional subspace. In the following argument we show that the distribution of the length of the projected vector is invariant to rotation which in turn enables us to find the projection in a straightforward manner.

Recall that $\mathbf{P}(\mathcal{S}^{(k)})$ is an orthogonal projection operator for a subspace $\mathcal{L}_k$ spanned by the columns of $\mathbf{A}_k$. Let $\mathcal{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_k\}$ denote an or-

thonormal basis for $\mathcal{L}_k$. There exist a rotation operator $\mathcal{R}$ such that $\mathcal{R}(\mathcal{B}) = \{\mathbf{e}_1, \ldots, \mathbf{e}_k\}$, where $\mathbf{e}_i$ is the i^{th} standard unit vector. Let $\mathbf{u} \sim \mathcal{N}(0, 1/n)$. Since a multivariate Gaussian distribution is spherically symmetric [223], distribution of $\mathbf{u}$ remains unchanged under rotation, i.e., $\mathcal{R}(\mathbf{u}) \sim \mathcal{N}(0, 1/n)$. Therefore, it holds that $\mathbb{E}\|\mathcal{R}(\mathbf{u})\|_2 = \mathbb{E}\|\mathbf{u}\|_2$. In addition, since after rotation $\{\mathbf{e}_1, \ldots, \mathbf{e}_k\}$ is a basis for the rotation of $\mathcal{L}_k$, $\left\|\mathbf{P}(\mathcal{S}^{(k)})\mathbf{u}\right\|_2$ has the same distribution as the length of a vector consisting of the first k components of $\mathcal{R}(\mathbf{u})$. It then follows from the i.i.d. assumption and linearity of expectation that $\mathbb{E}\left\|\mathbf{P}(\mathcal{S}^{(k)})\mathbf{u}\right\|_2^2 = \frac{k}{n}\mathbb{E}\|\mathbf{u}\|_2^2 = \frac{k}{n}$.

We now prove the statement in the second part of the lemma. Let $\mathbf{u}_k^{\mathcal{R}}$ be the vector collecting the first k coordinates of $\mathcal{R}(\mathbf{u})$. The above argument implies $\left\|\mathbf{P}(\mathcal{S}^{(k)})\mathbf{u}\right\|_2^2$ has the same distribution as $\left\|\mathbf{u}_k^{\mathcal{R}}\right\|_2^2$. In addition, $n\left\|\mathbf{u}_k^{\mathcal{R}}\right\|_2^2$ is distributed as χ_k^2 because of the spherical symmetry property of $\mathbf{u}$. Let $\lambda > 0$; we will specify the value of λ shortly. Now,

$$\begin{aligned}
\Pr\{\left\|\mathbf{P}(\mathcal{S}^{(k)})\mathbf{u}\right\|_2^2 \leq (1-\gamma)\frac{k}{n}\} &= \Pr\{n\left\|\mathbf{u}_k^{\mathcal{R}}\right\|_2^2 \leq (1-\gamma)k\} \\
&= \Pr\{-\frac{\lambda}{2}n\left\|\mathbf{u}_k^{\mathcal{R}}\right\|_2^2 \geq -\frac{\lambda k(1-\gamma)}{2}\} \\
&= \Pr\{e^{-\frac{\lambda}{2}n\left\|\mathbf{u}_k^{\mathcal{R}}\right\|_2^2} \geq e^{-\frac{\lambda k(1-\gamma)}{2}}\} \\
&\overset{(a)}{\leq} e^{\frac{\lambda k(1-\gamma)}{2}}\mathbb{E}\{e^{-\frac{\lambda}{2}n\left\|\mathbf{u}_k^{\mathcal{R}}\right\|_2^2}\} \\
&\overset{(b)}{=} e^{\frac{\lambda k(1-\gamma)}{2}}(1+\lambda)^{\frac{-k}{2}}
\end{aligned} \tag{A.2}$$

where (a) follows from the Markov inequality and (b) is due to the definition of the Moment Generating Function (MGF) for χ_k^2-distribution. Now, let

$\lambda = \frac{\gamma}{1-\gamma}$. It follows that

$$\Pr\{\left\|\mathbf{P}(\mathcal{S}^{(k)})\mathbf{u}\right\|_2^2 \leq (1-\gamma)\frac{k}{n}\} \leq e^{\frac{\lambda k(1-\gamma)}{2}}(1-\gamma)^{\frac{k}{2}} = e^{\frac{k}{2}(\gamma+\log(1-\gamma))} \leq e^{\frac{-k\gamma^2}{4}} \tag{A.3}$$

where in the last inequality we used the fact that $\log(1-\gamma) \leq -\gamma - \frac{\gamma^2}{2}$. Following the same line of argument, one can show that

$$\Pr\{\left\|\mathbf{P}(\mathcal{S}^{(k)})\mathbf{u}\right\|_2^2 \geq (1+\gamma)\frac{k}{n}\} \leq e^{-k(\frac{\gamma^2}{4}-\frac{\gamma^3}{6})}. \tag{A.4}$$

The combination of (A.3) and (A.4) using Boole's inequality leads to the stated result. ■

Lemma A.1.2 (Corollary 2.4.5 in [224]) states inequalities between the maximum and minimum singular values of a matrix and its submatrices.

Lemma A.1.2. *Let* $\mathbf{C}$ *be a full rank tall matrix and let* $\mathbf{A}$ *be a submatrix of* $\mathbf{C}$*. Then*

$$\sigma_{\min}(\mathbf{A}) \geq \sigma_{\min}(\mathbf{C}), \quad \sigma_{\max}(\mathbf{A}) \leq \sigma_{\max}(\mathbf{C}). \tag{A.5}$$

Lemma A.1.3 from [225] establishes a probabilistic bound on the smallest singular value of a normally distributed matrix.

Lemma A.1.3. *Let* $\mathbf{A} \in \mathbb{R}^{n\times k}$ *denote a tall matrix whose entries are drawn independently from* $\mathcal{N}(0, 1/n)$*. Then for any* $\delta > 0$ *it holds that*

$$\Pr(\sigma_{\min}(\mathbf{A}) \geq 1 - \sqrt{\frac{k}{n}} - \delta) \geq 1 - \exp(-\delta^2\frac{n}{2}), \tag{A.6}$$

and

$$\Pr(\sigma_{\max}(\mathbf{A}) \geq 1 + \sqrt{\frac{k}{n}} + \delta) \geq 1 - \exp(-\delta^2\frac{n}{2}). \tag{A.7}$$

Lemma A.1.4 (Lemma 5.1 in [227]) establishes bounds on the singular values of $\mathbf{A}_k$, i.e., a submatrix of $\mathbf{A}$ with k columns.

Lemma A.1.4. *Let $\mathbf{A} \in \mathbb{R}^{n \times m}$ denote a matrix with entries that are drawn independently from $\mathcal{N}(0, 1/n)$. Then, for any $0 < \delta < 1$ and for all $\mathbf{x} \in$ Range$(\mathbf{A}_k)$, it holds that*

$$\Pr\{|\frac{\|\mathbf{A}_k\mathbf{x}\|_2}{\|\mathbf{x}\|_2} - 1| \leq \delta\} \geq 1 - 2(\frac{12}{\delta})^k e^{-nc_0(\frac{\delta}{2})}. \tag{A.8}$$

Lemma A.1.5 (Proposition 4 in [78]) establishes an upper bound on the inner product of two independent random vectors.

Lemma A.1.5. *Let $\mathbf{a} \in \mathbb{R}^n$ denote a vector with entries that are drawn independently from $\mathcal{N}(0, 1/n)$. Let $\mathbf{u} \in \mathbb{R}^n$ be a random vector such that $\|\mathbf{u}\|_2 = 1$ and let $\mathbf{u}$ and $\mathbf{a}$ be statistically independent. Then for $\delta > 0$ it holds*

$$\Pr(|\mathbf{a}^\top \mathbf{u}| \leq \delta) \geq 1 - \exp(-\delta^2 \frac{n}{2}). \tag{A.9}$$

A.2 Proof of Theorem 3.2.2

The proof is inspired by the inductive framework first introduced in [78].[1] We can assume, without a loss of generality, that the nonzero components of $\mathbf{x}$ are in the first k locations. This implies that $\mathbf{A}$ can be written as $\mathbf{A} = \left[\bar{\mathbf{A}} \;\; \widetilde{\mathbf{A}}\right]$, where $\bar{\mathbf{A}} \in \mathbb{R}^{n \times k}$ has columns with indices in $\mathcal{S}_{true}$ and

[1] Our analysis relies on (3.1) rather than the computationally efficient recursions in (3.7). Nonetheless, we have shown the equivalence between the two criteria in Theorem 3.2.1.

$\widetilde{\mathbf{A}} \in \mathbb{R}^{n\times(m-k)}$ has columns with indices in $\mathcal{I}\backslash\mathcal{S}_{true}$. For $\mathcal{T}_1 \subset \mathcal{I}$ and $\mathcal{T}_2 \subset \mathcal{I}$ such that $\mathcal{T}_1 \cap \mathcal{T}_2 = \emptyset$, define

$$\mathbf{b}_j^{\mathcal{T}_1} = \frac{\mathbf{a}_j}{\|\mathbf{P}(\mathcal{T}_1)^{\perp}\mathbf{a}_j\|_2}, \ \ j \in \mathcal{T}_2, \tag{A.10}$$

where $\mathbf{P}_{\mathcal{T}_1}^{\perp}$ denotes the projection matrix onto the orthogonal complement of the subspace spanned by the columns of $\mathbf{A}$ with indices in $\mathcal{T}_1$. Using the notation of (A.10), (3.1) becomes

$$j_s = \arg\max\nolimits_{j\in\mathcal{I}\backslash\mathcal{S}_{i-1}} \left|\mathbf{r}_{i-1}^{\top}\mathbf{b}_j^{\mathcal{S}_{i-1}}\right|. \tag{A.11}$$

In addition, let $\mathbf{\Phi}_{\mathcal{S}^{(i)}} = [\mathbf{b}_j^{\mathcal{S}^{(i)}}] \in \mathbb{R}^{n\times(k-i)}$, $j \in \mathcal{S}_{true}\backslash\mathcal{S}^{(i)}$, and $\mathbf{\Psi}_{\mathcal{S}^{(i)}} = [\mathbf{b}_j^{\mathcal{S}^{(i)}}] \in \mathbb{R}^{n\times(m-k)}$, $j \in \mathcal{I}\backslash\mathcal{S}_{true}$. Assume that in the first i iterations AOLS selects columns from $\mathcal{S}_{true}$. Let $|\psi_{o_1}^{\top}\mathbf{r}_i| \leq \cdots \leq |\psi_{o_{m-k}}^{\top}\mathbf{r}_i|$ be an ordering of the set $\{|\psi_1^{\top}\mathbf{r}_i|, \ldots, |\psi_{m-k}^{\top}\mathbf{r}_i|\}$. According to the selection rule in (A.11), AOLS identifies at least one true column in the $(i+1)^{\text{st}}$ iteration if the maximum correlation between $\mathbf{r}_i$ and columns of $\mathbf{\Phi}_{\mathcal{S}^{(i)}}$ is greater than the $|\mathcal{P}(\mathbf{\Psi}_{\mathcal{S}^{(i)}}^{\top}\mathbf{r}_i)_{m-k-L+1}|$. Therefore,

$$\rho(\mathbf{r}_i) = \frac{|\mathcal{P}(\mathbf{\Psi}_{\mathcal{S}^{(i)}}^{\top}\mathbf{r}_i)_{m-k-L+1}|}{\|\mathbf{\Phi}_{\mathcal{S}^{(i)}}^{\top}\mathbf{r}_i\|_{\infty}} < 1 \tag{A.12}$$

guarantees that AOLS selects at least one true column in the $(i+1)^{\text{st}}$ iteration. Hence, $\rho(\mathbf{r}_i) < 1$ for $i \in \{0, \ldots, k-1\}$ ensures recovery of $\mathbf{x}$ in k iterations. In other words, $\max_i \rho(\mathbf{r}_i) < 1$ is sufficient condition for AOLS to successfully recover the support of $\mathbf{x}$, i.e., if Σ denotes the event that AOLS succeeds, then $\Pr\{\Sigma\} \geq \Pr\{\max_i \rho(\mathbf{r}_i) < 1\}$. We may upper bound $\rho(\mathbf{r}_i)$ as

$$\rho(\mathbf{r}_i) \leq \frac{|\mathcal{P}(\widetilde{\mathbf{A}}^{\top}\mathbf{r}_i)_{m-k-L+1}|}{\|\bar{\mathbf{A}}^{\top}\mathbf{r}_i\|_{\infty}} \frac{\max_{j\in\mathcal{S}_{true}} \|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_j\|_2}{\min_{j\notin\mathcal{S}_{true}} \|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a}_j\|_2}. \tag{A.13}$$

According to Lemma A.1.1,

$$\begin{aligned}\rho(\mathbf{r}_i) &\leq \frac{|\mathcal{P}(\widetilde{\mathbf{A}}^\top \mathbf{r}_i)_{m-k-L+1}|}{\|\bar{\mathbf{A}}^\top \mathbf{r}_i\|_\infty} \sqrt{\frac{1+\gamma}{1-\gamma}} \sqrt{\frac{(n-i)/n}{(n-i)/n}} \frac{\mathbb{E}\|\mathbf{a}_{j_{\max}}\|_2}{\mathbb{E}\|\mathbf{a}_{j_{\min}}\|_2} \\ &= \sqrt{\frac{1+\gamma}{1-\gamma}} \frac{|\mathcal{P}(\widetilde{\mathbf{A}}^\top \mathbf{r}_i)_{m-k-L+1}|}{\|\bar{\mathbf{A}}^\top \mathbf{r}_i\|_\infty}\end{aligned} \tag{A.14}$$

with probability exceeding $p_1 = \left(1 - 2e^{-(n-k+1)c_0(\gamma)}\right)^2$ for $0 \leq i < k$. Let $c_1(\gamma) = \sqrt{\frac{1-\gamma}{1+\gamma}}$. Using a simple norm inequality and exploiting the fact that $\bar{\mathbf{A}}^\top \mathbf{r}_i$ has at most $k-i$ nonzero entries leads to

$$\rho(\mathbf{r}_i) \leq \frac{\sqrt{k-i}}{c_1(\gamma)} \frac{|\mathcal{P}(\widetilde{\mathbf{A}}^\top \mathbf{r}_i)_{m-k-L+1}|}{\|\bar{\mathbf{A}}^\top \mathbf{r}_i\|_2} = \frac{\sqrt{k-i}}{c_1(\gamma)} \|\widetilde{\mathbf{A}}^\top \widetilde{\mathbf{r}}_i\|_\infty, \tag{A.15}$$

where $\widetilde{\mathbf{r}}_i = \mathbf{r}_i / \|\bar{\mathbf{A}}^\top \mathbf{r}_i\|_2$. According to Lemma A.1.4, for any $0 < \delta < 1$, $\Pr\{\|\widetilde{\mathbf{r}}_i\|_2 \leq \frac{1}{1-\delta}\} \geq 1 - 2(\frac{12}{\delta})^k e^{-nc_0(\frac{\delta}{2})} = p_2$. Subsequently,

$$\begin{aligned}\Pr\{\Sigma\} &\geq p_1 p_2 \Pr\{\max_{0 \leq i < k} |\mathcal{P}(\widetilde{\mathbf{A}}^\top \mathbf{r}_i)_{m-k-L+1}| < c_1(\gamma)\} \\ &\geq p_1 p_2 \prod_{j=1}^{m-k-L+1} \Pr\{\max_{0 \leq i < k} \left|\widetilde{\mathbf{a}}_{o_j}^\top \widetilde{\mathbf{r}}_i \sqrt{k-i}\right| < c_1(\gamma)\} \\ &= p_1 p_2 \Pr\{\max_{0 \leq i < k} \left|\widetilde{\mathbf{a}}_{o_1}^\top \widetilde{\mathbf{r}}_i \sqrt{k-i}\right| < c_1(\gamma)\}^{m-k-L+1},\end{aligned} \tag{A.16}$$

where we used the assumption that the columns of $\widetilde{\mathbf{A}}$ are independent. Note that the random vectors $\{\widetilde{\mathbf{r}}_i \sqrt{k-i}\}_{i=0}^{k-1}$ are bounded with probability exceeding p_2 and are statistically independent of $\widetilde{\mathbf{A}}$. Now, recall that the entries of $\mathbf{A}$ are drawn independently from $\mathcal{N}\left(0, \frac{1}{n}\right)$. Since the random variable $X_i = \widetilde{\mathbf{a}}_{o_1}^\top \widetilde{\mathbf{r}}_i \sqrt{k-i}$ is distributed as $\mathcal{N}(0, \sigma^2)$ with $\sigma^2 \leq \frac{k-i}{n(1-\delta)^2}$, by using a Gaussian tail bound and Boole's inequality it is straightforward to show that

$$\Pr\{\max_{0 \leq i < k} |X_i| < c_1(\gamma)\} \geq 1 - \sum_{i=0}^{k-1} e^{-\frac{n}{k-i} c_1(\gamma)^2 (1-\delta)^2}. \tag{A.17}$$

Thus, $\Pr\{\Sigma\} \geq p_1 p_2 p_3$, where

$$p_3 = \left(1 - \sum_{i=0}^{k-1} e^{-\frac{n}{k-i} c_1(\gamma)^2 (1-\delta)^2}\right)^{m-k-L+1}. \tag{A.18}$$

This completes the proof.

A.3 Proof of Theorem 3.2.3

Here we follow the outline of the proof of Theorem 3.2.2. Note that, in the presence of noise, $\bar{\mathbf{A}}^\top \mathbf{r}_i$ in (A.14) has at most k nonzero entries. After a straightforward modification of (A.15), we obtain

$$\rho(\mathbf{r}_i) \leq \frac{\sqrt{k}}{c_1(\gamma)} |\mathcal{P}(\widetilde{\mathbf{A}}^\top \mathbf{r}_i)_{m-k-L+1}|. \tag{A.19}$$

The most important difference between the noisy and noiseless scenarios is that $\mathbf{r}_i$ in the latter does not belong to the range of $\bar{\mathbf{A}}$; therefore, further restrictions are needed to ensure that $\{\widetilde{\mathbf{r}}_i\}_{i=0}^{k-1}$ remains bounded. To this end, we investigate lower bounds on $\|\bar{\mathbf{A}}^\top \mathbf{r}_i\|_2$ and upper bounds on $\|\widetilde{\mathbf{r}}_i\|_2$. Recall that in the i^{th} iteration

$$\mathbf{r}_i = \mathbf{P}(\mathcal{S}^{(i)})^\perp \mathbf{y} = \mathbf{P}(\mathcal{S}^{(i)})^\perp \left(\bar{\mathbf{A}}\bar{\mathbf{x}} + \mathbf{e}\right), \tag{A.20}$$

where $\bar{\mathbf{x}} \in \mathbb{R}^k$ is a subvector of $\mathbf{x}$ that collects nonzero components of $\mathbf{x}$. We can write $\mathbf{e}$ equivalently as

$$\mathbf{e} = \bar{\mathbf{A}}\mathbf{w} + \mathbf{e}^\perp, \tag{A.21}$$

where $\mathbf{e}^\perp = \mathbf{P}(\mathcal{S}^{(k)})^\perp \mathbf{e}$ is the projection of $\mathbf{e}$ onto the orthogonal complement of the subspace spanned by the columns of $\mathbf{A}$ corresponding to nonzero entries of

$\mathbf{x}$, and $\mathbf{w} = \bar{\mathbf{A}}^{\dagger}\mathbf{e}$. Substituting (A.21) into (A.20) and noting that $\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\mathbf{a} = 0$ if $\mathbf{a}$ is selected in previous iterations as well as observing that $\mathcal{L}_i \subset \mathcal{L}_k$, we obtain

$$\mathbf{r}_i = \mathbf{e}^{\perp} + \mathbf{P}(\mathcal{S}^{(i)})^{\perp}\bar{\mathbf{A}}_{i^c}\mathbf{c}_{i^c}, \tag{A.22}$$

where $\mathbf{c} = \bar{\mathbf{x}} + \mathbf{w}$ and subscript i^c denotes the set of correct columns that have not yet been selected. Evidently, (A.22) demonstrates that $\mathbf{r}_i$ can be written as a sum of orthogonal terms. Therefore,

$$\|\mathbf{r}_i\|_2^2 = \|\mathbf{e}^{\perp}\|_2^2 + \|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\bar{\mathbf{A}}_{i^c}\mathbf{c}_{i^c}\|_2^2. \tag{A.23}$$

Applying (A.22) yields

$$\begin{aligned}
\|\bar{\mathbf{A}}^{\top}\mathbf{r}_i\|_2 &= \|\bar{\mathbf{A}}^{\top}\left(\mathbf{e}^{\perp} + \mathbf{P}(\mathcal{S}^{(i)})^{\perp}\bar{\mathbf{A}}_{i^c}\mathbf{c}_{i^c}\right)\|_2 \\
&\overset{(a)}{=} \|\bar{\mathbf{A}}^{\top}\mathbf{e}^{\perp} + \bar{\mathbf{A}}_{i^c}^{\top}\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\bar{\mathbf{A}}_{i^c}\mathbf{c}_{i^c}\|_2 \\
&\overset{(b)}{=} \|\bar{\mathbf{A}}_{i^c}^{\top}\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\bar{\mathbf{A}}_{i^c}\mathbf{c}_{i^c}\|_2 \\
&\overset{(c)}{\geq} \sigma_{\min}^2(\bar{\mathbf{A}})\|\mathbf{c}_{i^c}\|_2,
\end{aligned} \tag{A.24}$$

where (a) holds because $\mathbf{P}(\mathcal{S}^{(i)})^{\perp}$ projects onto the orthogonal complement of the space spanned by the columns of $\bar{\mathbf{A}}_i$, (b) follows from the fact that columns of $\bar{\mathbf{A}}$ and $\mathbf{e}^{\perp}$ lie in orthogonal subspaces, and (c) follows from Lemma A.1.2 and the fact that $\mathbf{P}(\mathcal{S}^{(i)})^{\perp}$ is a projection matrix.

We now bound the norm of $\widetilde{\mathbf{r}}_i$. Substitute (A.23) and (A.24) in the

definition of $\widetilde{\mathbf{r}}_i$ to arrive at

$$\begin{aligned}\|\widetilde{\mathbf{r}}_i\|_2 &\leq \frac{\left[\|\mathbf{e}^{\perp}\|_2^2 + \|\mathbf{P}(\mathcal{S}^{(i)})^{\perp}\bar{\mathbf{A}}_{i^c}\mathbf{c}_{i^c}\|_2^2\right]^{\frac{1}{2}}}{\sigma_{\min}^2(\bar{\mathbf{A}})\|\mathbf{c}_{i^c}\|_2} \\ &\overset{(a)}{\leq} \frac{\left[\|\mathbf{e}^{\perp}\|_2^2 + \sigma_{\max}^2(\bar{\mathbf{A}})\|\mathbf{c}_{i^c}\|_2^2\right]^{\frac{1}{2}}}{\sigma_{\min}^2(\bar{\mathbf{A}})\|\mathbf{c}_{i^c}\|_2} \\ &= \frac{\left[\|\mathbf{e}^{\perp}\|_2^2/\|\mathbf{c}_{i^c}\|_2^2 + \sigma_{\max}^2(\bar{\mathbf{A}})\right]^{\frac{1}{2}}}{\sigma_{\min}^2(\bar{\mathbf{A}})}\end{aligned} \tag{A.25}$$

where (a) follows from Lemma A.1.2 and the fact that $\mathbf{P}(\mathcal{S}^{(i)})^{\perp}$ is a projection matrix. In addition,

$$\|\mathbf{e}^{\perp}\|_2 = \|\mathbf{P}(\mathcal{S}^{(k)})^{\perp}\mathbf{e}\|_2 \leq \|\mathbf{e}\|_2 \leq \gamma_{\mathbf{e}}. \tag{A.26}$$

Defining $\mathbf{x}_{\min} = \min_j |\bar{\mathbf{x}}_j|$ and $\mathbf{c}_{\min} = \min_j |\mathbf{c}_j|$, it is straightforward to see that

$$\mathbf{c}_{\min} \geq \mathbf{x}_{\min} - \|\mathbf{w}\|_2. \tag{A.27}$$

Moreover, we impose $\mathbf{x}_{\min} \geq (1+\delta)\|\mathbf{w}\|_2$. Therefore,

$$\begin{aligned}\|\mathbf{c}_{i^c}\|_2^2 &\geq (k-i)\mathbf{c}_{\min}^2 \\ &\geq (k-i)\left(\mathbf{x}_{\min} - \|\mathbf{w}\|_2\right)^2 \\ &= (k-i)(\mathbf{x}_{\min} - \|\bar{\mathbf{A}}^{\dagger}\mathbf{e}\|_2)^2 \\ &\geq (k-i)(\mathbf{x}_{\min} - \sigma_{\max}(\bar{\mathbf{A}}^{\dagger})\|\mathbf{e}\|_2)^2 \\ &= (k-i)(\mathbf{x}_{\min} - \sigma_{\min}(\bar{\mathbf{A}})\gamma_{\mathbf{e}})^2.\end{aligned} \tag{A.28}$$

Combining (A.25), (A.26), and (A.28) implies that

$$\begin{aligned}\|\widetilde{\mathbf{r}}_i\|_2 &\leq \frac{\left[\frac{\gamma_{\mathbf{e}}^2}{(k-i)(\mathbf{x}_{\min} - \sigma_{\min}(\bar{\mathbf{A}})\gamma_{\mathbf{e}})^2} + \sigma_{\max}^2(\bar{\mathbf{A}})\right]^{\frac{1}{2}}}{\sigma_{\min}^2(\bar{\mathbf{A}})} \\ &\leq \frac{\left[\frac{\gamma_{\mathbf{e}}^2}{(k-i)(\mathbf{x}_{\min} - (1+\delta)\gamma_{\mathbf{e}})^2} + (1+\delta)^2\right]^{\frac{1}{2}}}{(1-\delta)^2}\end{aligned} \tag{A.29}$$

with probability exceeding p_2. Thus, imposing the constraint

$$\mathbf{x}_{\min} \geq (1+\delta+t)\gamma_{\mathbf{e}} \tag{A.30}$$

where $t > 0$[2] establishes

$$\|\widetilde{\mathbf{r}}_i\|_2 \leq \frac{\left[\frac{1}{(k-i)t^2} + (1+\delta)^2\right]^{\frac{1}{2}}}{(1-\delta)^2}. \tag{A.31}$$

By following the steps of the proof of Theorem 3.2.2 and exploiting independence of the columns of $\widetilde{\mathbf{A}}$, we arrive at

$$\Pr\{\Sigma\} \geq p_1 p_2 \Pr\{\max_{0\leq i<k} \left|\widetilde{\mathbf{a}}_{o_1}^{\top}\widetilde{\mathbf{r}}_i\right| < \frac{c_1(\gamma)}{\sqrt{k}}\}^{m-k-L+1}. \tag{A.32}$$

Recall that $\{\widetilde{\mathbf{r}}_i\}_{i=0}^{k-1}$ are statistically independent of $\widetilde{\mathbf{A}}$ and that with probability higher than p_2 they are bounded. By using Boole's for the random variable $X_i = \widetilde{\mathbf{a}}_{o_1}^{\top}\widetilde{\mathbf{r}}_i$ we obtain

$$\Pr\{\max_{0\leq i<k} |X_i| < \frac{c_1(\gamma)}{\sqrt{k}}\} \geq 1 - \sum_{i=0}^{k-1} e^{-\frac{nc_1(\gamma)^2(1-\delta)^4}{k\left[\frac{1}{(k-i)t^2}+(1+\delta)^2\right]}}. \tag{A.33}$$

Let us denote

$$p_3 = \left(1 - \sum_{i=0}^{k-1} e^{-\frac{nc_1(\gamma)^2(1-\delta)^4}{k\left[\frac{1}{(k-i)t^2}+(1+\delta)^2\right]}}\right)^{m-k-L+1}. \tag{A.34}$$

Then from (A.32) and (A.33) follows that $\Pr\{\Sigma\} \geq p_1 p_2 p_3$, which completes the proof.

Remark A.3.1. Note that in the absence of noise the first term in the numerator of (A.31) vanishes, leading to $\|\widetilde{\mathbf{r}}_i\|_2 \leq \frac{1}{1-\delta} + \frac{2\delta}{(1-\delta)^2}$. A comparison with the proof of Theorem 3.2.2 suggests that the term $\frac{2\delta}{(1-\delta)^2}$ is a modification which stems from the presence of noise.

[2]This is consistent with our previous condition $\mathbf{x}_{\min} \geq (1+\delta)\|\mathbf{w}\|_2$.

A.4 Proof of Theorem 3.3.1

Before proceeding to the proof, we state a useful lemma from [228].

Lemma A.4.1. *For every* $|a| \leq 1$ *and* $b \geq 1$ *it holds that* $(1+a)^b \geq e^{ab}(1-a^2b)$.

Proof. Let x $= ab$, $|x| \leq b$. Consider $g(x) = e^{-x}(1+\frac{x}{b})^b - (1-\frac{x^2}{b})$. At $x = 0$, both $g(x)$ and $f'(x)$ are zero. If $f'(x) = 0$ for any other x in the interval, for such x we have

$$e^{-x}(1+\frac{x}{b})^b = 2 + \frac{2x}{b}.$$

Therefore, for such x

$$g(x) = \frac{(x+1)^2}{b} + 1 - \frac{1}{b} > 0.$$

Furthermore, since $g(b) > 0$ for all b while $g(-b) > 0$ for $b > 1$ and $g(-b) = 0$ for $b = 1$, all other points we must have $g(x) > 0$. ■

To prove the theorem, we first establish Lemma A.4.2 below that demonstrates that the probability of success of PSG is product of two terms: (i) $\prod_{i=0}^{k-1} p_{psg}^{(i)}$ that characterizes the likelihood that $\mathcal{R}_{psg}^{(i)}$ contains at least a new element of $\mathcal{S}^\star$ for all $i = 0, \ldots, k-1$, (ii) $\prod_{i=0}^{k-1} q_{psg}^{(i)}$ that determines the chance of selecting one of the elements in the nonempty intersection of search space $\mathcal{R}_{psg}^{(i)}$ and $\mathcal{S}^\star \backslash \mathcal{S}_{psg}^{(i)}$. The first term $\prod_{i=0}^{k-1} p_{psg}^{(i)}$ is of particular interest as it can be thought of as being a general upper bound on success probability, a fact that is used in the proof of Theorem 3.3.3.

Lemma A.4.2. *Let $\mathcal{S}_{psg}^{(k)}$ denote the subset selected by PSG, and let $\mathcal{R}_{psg}^{(i)}$ denote the randomly selected search space of PSG in i^{th} iteration. Then, it holds that*

$$\Pr\left(\mathcal{S}_{psg}^{(k)} = \mathcal{S}^{\star}\right) = \prod_{i=0}^{k-1} p_{psg}^{(i)} \prod_{i=0}^{k-1} q_{psg}^{(i)}, \tag{A.35}$$

where

$$p_{psg}^{(i)} = \Pr\left(\mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^{\star} \backslash \mathcal{S}_{psg}^{(i)}) \neq \emptyset \mid \mathcal{S}_{psg}^{(i)} \subset \mathcal{S}^{\star} \ , \ |\mathcal{S}_{psg}^{(i)}| = i\right), \tag{A.36}$$

and

$$q_{psg}^{(i)} = \Pr\left(\mathcal{S}_{psg}^{(i+1)} \subset \mathcal{S}^{\star} \mid \mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^{\star} \backslash \mathcal{S}_{psg}^{(i)}) \neq \emptyset, \ |\mathcal{S}_{psg}^{(i)}| = i\right). \tag{A.37}$$

Proof. Let $\mathcal{A}_{psg}^{(i)}$ denote the event $\{\mathcal{S}_{psg}^{(i+1)} \cap \mathcal{S}_{psg}^{(i)} \neq \emptyset, \mathcal{S}_{psg}^{(i+1)} \subseteq \mathcal{S}^{\star}\}$. Then the probability of success of PSG can be expressed as

$$\begin{aligned} \Pr\left(\mathcal{S}_{psg}^{(k)} = \mathcal{S}^{\star}\right) &= \Pr\left(\cap_{i=0}^{k-1} \mathcal{A}_{psg}^{(i)}\right) \\ &= \prod_{i=0}^{k-1} \Pr\left(\mathcal{A}_{psg}^{(i)} \mid \cap_{j=0}^{i-1} \mathcal{A}_{psg}^{(j)}\right) \\ &= \prod_{i=0}^{k-1} \Pr\left(\mathcal{A}_{psg}^{(i)} \mid \mathbf{B}_{psg}^{(i)}\right), \end{aligned} \tag{A.38}$$

where $\mathbf{B}_{psg}^{(i)} = \{\mathcal{S}_{psg}^{(i)} \subset \mathcal{S}^{\star}, |\mathcal{S}_{psg}^{(i)}| = i\}$. Note that $\mathcal{A}_{psg}^{(i)}$ can equivalently be written as

$$\mathcal{A}_{psg}^{(i)} = \{\mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^{\star} \backslash \mathcal{S}_{psg}^{(i)}) \neq \emptyset, \mathcal{S}_{psg}^{(i+1)} \subseteq \mathcal{S}^{\star}\}. \tag{A.39}$$

This can be written by further conditioning as

$$\Pr\left(\mathcal{S}_{psg}^{(k)} = \mathcal{S}^{\star}\right) = \prod_{i=0}^{k-1} p_{psg}^{(i)} \prod_{i=0}^{k-1} q_{psg}^{(i)}, \tag{A.40}$$

where $p_{psg}^{(i)}$ and $q_{psg}^{(i)}$ are given by (A.36) and (A.37), respectively. ■

Therefore, to prove the theorem it suffices to derive nontrivial lower bounds on $\prod_{i=0}^{k-1} p_{psg}^{(i)}$ and $\prod_{i=0}^{k-1} q_{psg}^{(i)}$.

First we establish a preliminary general result in Lemma A.4.3 which provides a lower bound on $\prod_{i=0}^{k-1} p_{psg}^{(i)}$.

Lemma A.4.3. *Let* $r_i = \min(\frac{m}{k-i} \log \frac{1}{\epsilon}, m)$*, for all* $i = 0, \ldots, k-1$*. Then,*

$$\prod_{i=0}^{k-1} p_{psg}^{(i)} \geq \exp\left(-\epsilon k + \epsilon \log \frac{1}{\epsilon}\right)\left(1 - \epsilon^2 k + \epsilon^2 \log \frac{1}{\epsilon}\right). \tag{A.41}$$

Proof. First note that since $r_i = m$ for all $i \geq k - \log \frac{1}{\epsilon}$, it follows that $p_{psg}^{(i)} = 1$. Let us first consider the setting of sampling with replacement. It holds that

$$\begin{aligned} \prod_{i=0}^{k-1} p_{psg}^{(i)} &= \prod_{i=0}^{k-\log \frac{1}{\epsilon}-1} \left(1 - \left(1 - \frac{k-i}{m}\right)^{r_i}\right) \\ &\geq \prod_{i=0}^{k-\log \frac{1}{\epsilon}-1} \left(1 - \exp\left(-r_i \frac{k-i}{m}\right)\right) \\ &= (1-\epsilon)^{k - \log \frac{1}{\epsilon}}. \end{aligned} \tag{A.42}$$

Finally, to obtain (A.41) we apply Lemma A.4.1.

Next, we consider the setting of sampling without replacement. For

every $i < k - \log \frac{1}{\epsilon}$,

$$\begin{aligned} p_{psg}^{(i)} &= 1 - \prod_{l=0}^{r_i-1} \left(1 - \frac{k-i}{m-l}\right) \\ &\overset{(a)}{\geq} 1 - \left(1 - \frac{k-i}{r_i} \sum_{l=0}^{r_i-1} \frac{1}{m-l}\right)^{r_i} \\ &\geq 1 - \left(1 - \frac{k-i}{m}\right)^{r_i} \\ &\overset{(b)}{\geq} 1 - \exp\left(-r_i \frac{k-i}{m}\right) \\ &= 1 - \epsilon, \end{aligned} \tag{A.43}$$

where (a) is obtained by using the inequality of arithmetic and geometric means, and (b) is due to the fact that $(1+x)^y \leq e^{xy}$ for any real number $y \geq 1$. Therefore, just as in the case of sampling with replacement, (A.41) holds. ■

Since we established a lower bound on $\prod_{i=0}^{k-1} p_{psg}^{(i)}$ in (A.41), it just remains to derive a nontrivial lower bound on $q_{psg}^{(i)}$ in order to show existence of a sufficient condition for the exact identification of $\mathcal{S}^\star$ and establish a lower bound on the probability of success of PSG. A lower bound on $q_{psg}^{(i)}$ can be obtained by considering the conditions under which the largest marginal gain of elements in $\mathcal{R}_{psg}^{(i)} \cap \mathcal{S}^\star$ exceeds that in $\mathcal{R}_{psg}^{(i)} \backslash \mathcal{S}^\star$ for all $i = 0, \ldots, k-1$, that is,

$$\max_{j \in \mathcal{R}_{psg}^{(i)} \backslash \mathcal{S}^\star} g_j(\mathcal{S}_{psg}^{(i)}) < \max_{j \in \mathcal{R}_{psg}^{(i)} \cap \mathcal{S}^\star} g_j(\mathcal{S}_{psg}^{(i)}), \tag{A.44}$$

with high probability. In Lemma A.4.4, we show the sufficient condition defined in (A.44) holds with high probability for PSG applied to the problem of

sparse support selection.

Lemma A.4.4. *Let* $\mathbf{x} \in \mathbb{R}^m$ *be an arbitrary sparse vector with* $k < m$ *non-zero entries and let* $\mathbf{A} \in \mathbb{R}^{n \times m}$ *denote a random matrix with entries drawn independently from* $\mathcal{N}(0, 1/n)$*. Given noiseless measurements* $\mathbf{y} = \mathbf{A}\mathbf{x}$*, for PSG with parameter* $e^{-k} \leq \epsilon \leq e^{-\frac{k}{m}}$ *it holds that* $\prod_{i=0}^{k-1} q_{psg}^{(i)} \geq \tilde{q}_1 \tilde{q}_2$ *where*

$$\begin{aligned} \tilde{q}_1 &= \left(1 - 2\exp\left(-n(\frac{\gamma^2}{4} - \frac{\gamma^3}{6})\right)\right)^m - \exp(-\delta^2 \frac{n}{2}), \text{ and} \\ \tilde{q}_2 &= \left(1 - \exp\left(-\frac{1-\gamma}{1+\gamma}(1 - \sqrt{\frac{k}{n}} - \delta)^2 \frac{n}{2k}\right)\right)^{k(m-k)}, \end{aligned} \tag{A.45}$$

for any $0 < \gamma < 1$ *and* $\delta > 0$*.*

We now proceed with the proof of Lemma A.4.4. Let $\mathbf{r}_i := (\mathbf{I}_n - \mathbf{P}(\mathcal{S}_{psg}^{(i)}))\mathbf{y}$ be the *residual* vector in the i^{th} iteration of PSG. Note that if in the previous iterations PSG selected columns of $\mathbf{A}$ with indices from $\mathcal{S}^\star$, the selected columns are orthogonal to $\mathbf{r}_i$.

To prove the stated result it is sufficient to establish a lower bound on the probability of (A.44). Given the selection criterion of OMP for sparse support selection, it is straightforward to see that

$$\rho(\mathbf{r}_i) := \max_{j \in \mathcal{R}_{psg}^{(i)} \setminus \mathcal{S}^\star} \frac{|\mathbf{a}_j^\top \mathbf{r}_i|}{\|\mathbf{a}_j\|_2} \Big/ \max_{j \in \mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \setminus \mathcal{S}_{psg}^{(i)})} \frac{|\mathbf{a}_j^\top \mathbf{r}_i|}{\|\mathbf{a}_j\|_2} < 1 \tag{A.46}$$

is a sufficient condition for successful identification of an element from $\mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \setminus \mathcal{S}_{psg}^{(i)})$. Our goal in this theorem is to prove that with high probability $\rho(\mathbf{r}_i) < 1$ in each iteration i. This in turn will establish a lower bound on $q_{psg}^{(i)}$, $i = 0, \ldots, k-1$. To this end, following [78] we employ an induction

technique to show that $\rho(\mathbf{r}_i) < 1$ if $\mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \backslash \mathcal{S}_{psg}^{(i)}) \neq \emptyset$ and $\mathcal{S}_{psg}^{(i)} \subseteq \mathcal{S}^\star$. Since computing $\rho(\mathbf{r}_i)$ appears challenging, to establish the desired results we show that a judicious upper bound on $\rho(\mathbf{r}_i)$ is with overwhelming probability smaller than 1. In particular, note that one may upper bound $\rho(\mathbf{r}_i)$ as

$$\begin{aligned}\rho(\mathbf{r}_i) &\leq \frac{\max_{j \in \mathcal{R}_{psg}^{(i)} \backslash \mathcal{S}^\star} |\mathbf{a}_j^\top \mathbf{r}_i|}{\max_{j \in \mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \backslash \mathcal{S}_{psg}^{(i)})} |\mathbf{a}_j^\top \mathbf{r}_i|} \cdot \frac{\max_{j \in \mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \backslash \mathcal{S}_{psg}^{(i)})} \|\mathbf{a}_j\|_2}{\min_{j \in \mathcal{R}_{psg}^{(i)} \backslash \mathcal{S}^\star} \|\mathbf{a}_j\|_2}, \\ &\leq \frac{\max_{j \in \mathcal{R}_{psg}^{(i)} \backslash \mathcal{S}^\star} |\mathbf{a}_j^\top \mathbf{r}_i|}{\max_{j \in \mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \backslash \mathcal{S}_{psg}^{(i)})} |\mathbf{a}_j^\top \mathbf{r}_i|} \cdot \frac{\max_{j \in [m]} \|\mathbf{a}_j\|_2}{\min_{j \in [m]} \|\mathbf{a}_j\|_2}.\end{aligned} \tag{A.47}$$

Let $\mathbf{Z}_1$ denote the event that

$$\frac{\max_{j \in [m]} \|\mathbf{a}_j\|_2}{\min_{j \in [m]} \|\mathbf{a}_j\|_2} \leq \sqrt{\frac{1+\gamma}{1-\gamma}} \tag{A.48}$$

for some $\gamma \in (0, 1)$. Then, from Lemma A.1.1 it follows that

$$\Pr(\mathbf{Z}_1) \geq \left(1 - 2e^{-c_0(\gamma)n}\right)^m. \tag{A.49}$$

In other words, since $\|\mathbf{a}_j\|_2$'s are highly concentrated around one, one can approximate (A.47) by disregarding the second factor on the right-hand side. Additionally, let $\mathbf{Z}_2$ denote the event that $\sigma_{\min}(\mathbf{A}_{\mathcal{S}^\star}) \geq 1 - \sqrt{\frac{k}{n}} - \delta$ for some $\delta > 0$. Then, from Lemma A.1.3 we have

$$\Pr(\mathbf{Z}_2) \geq 1 - \exp(-\delta^2 \frac{n}{2}). \tag{A.50}$$

Therefore, by conditioning

$$\Pr\left(\rho(\mathbf{r}_i) < 1\right) \geq \Pr\left(\rho(\mathbf{r}_i) < 1 \mid \mathbf{Z}_1 \cap \mathbf{Z}_2\right) \Pr(\mathbf{Z}_1 \cap \mathbf{Z}_2). \tag{A.51}$$

Note that occurrence of $\mathbf{Z}_1$ and $\mathbf{Z}_2$ in the $i = 0$ iteration implies $\mathbf{Z}_1$ and $\mathbf{Z}_1$ occur throughout the algorithm. Thus, $\mathbf{Z}_1$ and $\mathbf{Z}_2$ are in a sense *global* events. Note that $\Pr(\mathbf{Z}_1 \cap \mathbf{Z}_2)$ can be bounded according to

$$\begin{aligned}\Pr(\mathbf{Z}_1 \cap \mathbf{Z}_2) &= \Pr(\mathbf{Z}_1) + \Pr(\mathbf{Z}_2) - \Pr(\mathbf{Z}_1 \cup \mathbf{Z}_2), \\ &\geq \Pr(\mathbf{Z}_1) + \Pr(\mathbf{Z}_2) - 1, \\ &\geq \left(1 - 2e^{-c_0(\gamma)n}\right)^m - \exp(-\delta^2 \frac{n}{2}) := \tilde{q}_1.\end{aligned} \tag{A.52}$$

Now, note that $\max_{j \in \mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \setminus \mathcal{S}_{psg}^{(i)})} |\mathbf{a}_j^\top \mathbf{r}_i|$ can alternatively be written as an ℓ_∞-norm of its argument. Furthermore, since $|\mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \setminus \mathcal{S}_{psg}^{(i)})| \leq |\mathcal{S}^\star| \leq k$, there are at most k inner products $|\mathbf{a}_j^\top \mathbf{r}_i|$ to consider (i.e., $1 \leq j \leq k$). Finally, since for a k-dimensional vector $\mathbf{a}$ holds that $\sqrt{k}\|\mathbf{a}\|_\infty \geq \|\mathbf{a}\|_2$, by conditioning on $\mathbf{Z}_1 \cap \mathbf{Z}_2$ we have

$$\begin{aligned}\rho(\mathbf{r}_i) &\leq \sqrt{k} \frac{\max_{j \in \mathcal{R}_{psg}^{(i)} \setminus \mathcal{S}^\star} |\mathbf{a}_j^\top \mathbf{r}_i|}{\|\mathbf{A}^\top_{\mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \setminus \mathcal{S}_{psg}^{(i)})} \mathbf{r}_i\|_2} \sqrt{\frac{1+\gamma}{1-\gamma}}, \\ &= \frac{\sqrt{k}}{c_1(\gamma)} \max_{j \in \mathcal{R}_{psg}^{(i)} \setminus \mathcal{S}^\star} |\mathbf{a}_j^\top \tilde{\mathbf{r}}_i|,\end{aligned} \tag{A.53}$$

where $c_1(\gamma) = \sqrt{\frac{1-\gamma}{1+\gamma}}$ and $\tilde{\mathbf{r}}_i = \mathbf{r}_i / \|\mathbf{A}^\top_{\mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \setminus \mathcal{S}_{psg}^{(i)})} \mathbf{r}_i\|_2$. Note that $\tilde{\mathbf{r}}_i$ is introduced in part to help us apply the concentration results established by Lemma A.1.5. Since $\mathbf{A}_{\mathcal{R}_{psg}^{(i)} \cap (\mathcal{S}^\star \setminus \mathcal{S}_{psg}^{(i)})}$ is a submatrix of $\mathbf{A}_{\mathcal{S}^\star}$, by conditioning on $\mathbf{Z}_1 \cap \mathbf{Z}_2$,

properties of singular values, and Lemma A.1.2 we obtain

$$\begin{aligned}\|\tilde{\mathbf{r}}_i\|_2 &= \frac{\|\mathbf{r}_i\|_2}{\|\mathbf{A}^{\top}_{\mathcal{R}^{(i)}_{psg}\cap(\mathcal{S}^{\star}\backslash\mathcal{S}^{(i)}_{psg})}\mathbf{r}_i\|_2}, \\ &\leq \frac{1}{\sigma_{\min}\big(_{\mathcal{R}^{(i)}_{psg}\cap(\mathcal{S}^{\star}\backslash\mathcal{S}^{(i)}_{psg})}\big)}, \\ &\leq \frac{1}{\sigma_{\min}(\mathbf{A}_{\mathcal{S}^{\star}})}, \\ &\leq \frac{1}{1-\sqrt{\frac{k}{n}}-\delta}.\end{aligned} \tag{A.54}$$

By defining $\bar{\mathbf{r}}_i = \sigma_{\min}(\mathbf{A}_{\mathcal{S}^{\star}})\tilde{\mathbf{r}}_i$, $\|\bar{\mathbf{r}}_i\|_2 = 1$, conditioning on $\mathbf{Z}_1 \cap \mathbf{Z}_2$ (A.53) can be written as

$$\begin{aligned}\rho(\mathbf{r}_i) &\leq \frac{\sqrt{k}}{c_1(\gamma)(1-\sqrt{\frac{k}{n}}-\delta)} \max_{j\in\mathcal{R}^{(i)}_{psg}\backslash\mathcal{S}^{\star}} |\mathbf{a}_j^{\top}\bar{\mathbf{r}}_i| \\ &\leq \frac{\sqrt{k}}{c_1(\gamma)(1-\sqrt{\frac{k}{n}}-\delta)} \max_{j\in[m]\backslash\mathcal{S}^{\star}} |\mathbf{a}_j^{\top}\bar{\mathbf{r}}_i|\end{aligned} \tag{A.55}$$

Thus, conditioning on $\mathbf{Z}_1$ and $\mathbf{Z}_2$

$$\max_{j\in[m]\backslash\mathcal{S}^{\star}} |\mathbf{a}_j^{\top}\bar{\mathbf{r}}_i| < \frac{c_1(\gamma)(1-\sqrt{\frac{k}{n}}-\delta)}{\sqrt{k}} \tag{A.56}$$

is a sufficient condition for successful identification of an element from $\mathcal{R}^{(i)}_{psg} \cap (\mathcal{S}^{\star}\backslash\mathcal{S}^{(i)}_{psg})$. Note that since by the hypothesis of the inductive argument $\mathcal{R}^{(i)}_{psg} \cap (\mathcal{S}^{\star}\backslash\mathcal{S}^{(i)}_{psg}) \neq \emptyset$ and $\mathcal{S}^{(i)}_{psg} \subseteq \mathcal{S}^{\star}$ hold, $\bar{\mathbf{r}}_i$ is in the span of $\mathbf{A}_{\mathcal{S}^{\star}}$, and subsequently $\bar{\mathbf{r}}_i$ and $\mathbf{a}_j$'s are statistically independent for all $j \in [m]\backslash\mathcal{S}^{\star}$. Therefore, by Lemma

A.1.5 and the fact that $\mathbf{a}_j$'s are i.i.d. normal random vectors

$$\begin{aligned}
&\Pr\left(\max_{j\in[m]\setminus\mathcal{S}^\star}|\mathbf{a}_j^\top\bar{\mathbf{r}}_i|<\frac{c_1(\gamma)(1-\sqrt{\frac{k}{n}}-\delta)}{\sqrt{k}}\right)\\
&=\Pr\left(|\mathbf{a}_1^\top\bar{\mathbf{r}}_i|<\frac{c_1(\gamma)(1-\sqrt{\frac{k}{n}}-\delta)}{\sqrt{k}}\right)^{(m-k)}\\
&\geq\left(1-\exp\left(-c_1(\gamma)^2(1-\sqrt{\frac{k}{n}}-\delta)^2\frac{n}{2k}\right)\right)^{(m-k)}\\
&:=\tilde{q}_2^{\frac{1}{k}}
\end{aligned}\tag{A.57}$$

Finally, noting $\prod_{i=0}^{k-1}q_{psg}^{(i)}\geq\tilde{q}_1\prod_{i=0}^{k-1}\tilde{q}_2^{\frac{1}{k}}=\tilde{q}_1\tilde{q}_2$ establishes the stated results.

We now proceed with the reminder of proof of Theorem 3.3.1. Let us take a closer look to $\tilde{q}_1$. We may bound $\tilde{q}_1$ using the inequality $(1-x)^l\geq 1-lx$, valid for $x\leq 1$ and $l\geq 1$ according to

$$\tilde{q}_1\geq 1-2m\exp\left(-n(\frac{\gamma^2}{4}-\frac{\gamma^3}{6})\right)-\exp(-\delta^2\frac{n}{2}).\tag{A.58}$$

Since our goal is to show the optimal sample complexity is achieved by PSG, comparing $\tilde{q}_1$ and $\tilde{q}_2$ we can conclude $\tilde{q}_1$ can be easily excluded from our numerical approximations as the exponent in $\tilde{q}_1$ increases linearly with n while exponent in $\tilde{q}_2$ increases fairly more slowly. Alternatively, we can multiply $\tilde{q}_1$ and $\tilde{q}_2$, and by discarding positive higher order terms achieve the same conclusion.

Now, lets turn our attention towards the lower bound on $\prod_{i=0}^{k-1}p_{psg}^{(i)}$. Although in Lemma A.4.3 we presented a relatively tight bound, the proof of

Lemma A.4.3 reveals a simpler lower bound

$$\prod_{i=0}^{k-1} p_{psg}^{(i)} \geq (1-\epsilon)^k \geq 1 - k\epsilon. \tag{A.59}$$

Next, we find a simple lower bound on $\tilde{q}_2$. Assume, $(1 - \sqrt{\frac{k}{n}} - \delta)^2 \geq 1 - c$ for some $c > 0$. Then, it holds that $n \geq C_2 k$, where $C_2 := (1 - \sqrt{1-c} + \delta)^{-2}$. Thus, employing $(1-x)^l \geq 1 - lx$ once again yields

$$\tilde{q}_2 \geq 1 - k(m-k) \exp\left(-\frac{1-\gamma}{1+\gamma}(1-c)\frac{n}{2k}\right). \tag{A.60}$$

Let $C_1 := \frac{1-\gamma}{1+\gamma}\frac{1-c}{2}$. Given that $k(m-k) \leq \frac{1}{4}(\frac{m}{k})^6$ for $m > k\sqrt{k}$, we obtain

$$\tilde{q}_2 \geq 1 - \frac{1}{4}(\frac{m}{k})^6 \exp\left(-C_1\frac{n}{k}\right). \tag{A.61}$$

Now, since $(1-\beta)^2 \geq 1 - 2\beta$, in order to establish $\Pr\left(\mathcal{S}_{psg}^{(k)} = \mathcal{S}^\star\right) \geq 1 - 2\beta$, it suffices to show

$$1 - k\epsilon > 1 - \beta, \text{ and} \qquad 1 - \frac{1}{4}(\frac{m}{k})^6 \exp\left(-C_1\frac{n}{k}\right) > 1 - \beta. \tag{A.62}$$

Therefore, the condition on ϵ, i.e., $\epsilon < \frac{\beta}{k}$, and the results emerge by rearranging the above inequalities.

A.5 Proof of Theorem 3.3.2

We can follow the steps in the proof of Theorem 3.3.1 and Lemma A.4.4 to obtain

$$\begin{aligned} \rho(\mathbf{r}_i) &\leq \sqrt{k}\frac{\max_{j\in\mathcal{R}_{psg}^{(i)}\setminus\mathcal{S}^\star} |\mathbf{a}_j^\top \mathbf{r}_i|}{\|\mathbf{A}^\top_{\mathcal{R}_{psg}^{(i)}\cap(\mathcal{S}^\star\setminus\mathcal{S}_{psg}^{(i)})}\mathbf{r}_i\|_2}\sqrt{\frac{1+\gamma}{1-\gamma}}, \\ &= \frac{\sqrt{k}}{c_1(\gamma)} \max_{j\in\mathcal{R}_{psg}^{(i)}\setminus\mathcal{S}^\star} |\mathbf{a}_j^\top \tilde{\mathbf{r}}_i|, \end{aligned} \tag{A.63}$$

by conditioning on $\mathbf{Z}_1 \cap \mathbf{Z}_2$, where $c_1(\gamma) = \sqrt{\frac{1-\gamma}{1+\gamma}}$ and $\tilde{\mathbf{r}}_i = \mathbf{r}_i / \|\mathbf{A}^{\top}_{\mathcal{R}^{(i)}_{psg} \cap (\mathcal{S}^{\star} \setminus \mathcal{S}^{(i)}_{psg})} \mathbf{r}_i\|_2$. The main difference compared to Theorem 3.3.1 is the approach we need to take to bound $\tilde{\mathbf{r}}_i$.

To this end, recall that in the i^{th} iteration

$$\mathbf{r}_i = \mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg})\mathbf{y} = \mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg}) \left(\mathbf{A}_{\mathcal{S}^*} \bar{\mathbf{x}} + \mathbf{n}\right), \tag{A.64}$$

where $\bar{\mathbf{x}} \in \mathbb{R}^k$ is a subvector of $\mathbf{x}$ that collects the top k components of $\mathbf{x}$ (i.e. the nonzero entries of the best k-sparse approximation $\hat{\mathbf{x}}$), and

$$\mathbf{n} = \mathbf{e} + \mathbf{A}_{[m] \setminus \mathcal{S}^*} \tilde{\mathbf{x}}, \tag{A.65}$$

where $\tilde{\mathbf{x}} \in \mathbb{R}^{m-k}$ collects the bottom $m-k$ components of $\mathbf{x}$. Note that $\mathbf{n}$ can also be written as

$$\mathbf{n} = \mathbf{A}_{\mathcal{S}^*} \mathbf{w} + \mathbf{n}^{\perp}, \tag{A.66}$$

where $\mathbf{n}^{\perp} = \mathbf{P}^{\perp}(\mathcal{S}^*)\mathbf{n}$ is the projection of $\mathbf{n}$ onto the orthogonal complement of the subspace spanned by the columns of $\mathbf{A}$ corresponding to the top k entries in $\mathbf{x}$, and $\mathbf{w} = \mathbf{A}^{\dagger}_{\mathcal{S}^*}\mathbf{n}$. Substituting (A.66) into (A.64) and noting that $\mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg})\mathbf{a} = 0$ if $\mathbf{a}$ is selected in previous iterations as well as observing that we obtain

$$\mathbf{r}_i = \mathbf{n}^{\perp} + \mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg}) \mathbf{A}_{\mathcal{S}^{\star} \setminus \mathcal{S}^{(i)}_{psg}} \mathbf{c}_{\mathcal{S}^{\star} \setminus \mathcal{S}^{(i)}_{psg}}, \tag{A.67}$$

where $\mathbf{c} = \bar{\mathbf{x}} + \mathbf{w} \in \mathbb{R}^k$. Note that the result in (A.67) essentially states that $\mathbf{r}_i$ can be written as a sum of two orthogonal terms. Consequently,

$$\|\mathbf{r}_i\|_2^2 = \|\mathbf{n}^{\perp}\|_2^2 + \|\mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg}) \mathbf{A}_{\mathcal{S}^{\star} \setminus \mathcal{S}^{(i)}_{psg}} \mathbf{c}_{\mathcal{S}^{\star} \setminus \mathcal{S}^{(i)}_{psg}}\|_2^2. \tag{A.68}$$

Now, using (A.67) we can proceed by bounding the normalizing factor in $\tilde{\mathbf{r}}_i$

$$\begin{aligned}
\|\mathbf{A}^{\top}_{\mathcal{R}^{(i)}_{psg}\cap(\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg})}\mathbf{r}_i\|_2 &= \|\mathbf{A}^{\top}_{\mathcal{R}^{(i)}_{psg}\cap(\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg})}\left(\mathbf{n}^{\perp}+\mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg})\mathbf{A}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\mathbf{c}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\right)\|_2 \\
&= \|\mathbf{A}^{\top}_{\mathcal{R}^{(i)}_{psg}\cap(\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg})}\mathbf{n}^{\perp}+\mathbf{A}^{\top}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg})\mathbf{A}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\mathbf{c}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\|_2 \\
&\overset{(a)}{=} \|\mathbf{A}^{\top}_{\mathcal{R}^{(i)}_{psg}\cap(\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg})}\mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg})\mathbf{A}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\mathbf{c}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\|_2 \\
&\overset{(b)}{\geq} \sigma^2_{\min}(\mathbf{A}_{\mathcal{S}^*})\|\mathbf{c}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\|_2,
\end{aligned} \tag{A.69}$$

where (a) follows from the fact that columns of $\mathbf{A}_{\mathcal{R}^{(i)}_{psg}\cap(\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg})}$ and $\mathbf{n}^{\perp}$ lie in orthogonal subspaces, and (b) follows from Lemma A.1.2 and the fact that $\mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg})$ is a projection matrix.

Having established a lower bound for the normalizing factor, we now proceed to bound the norm of $\tilde{\mathbf{r}}_i$. Substitute (A.68) and (A.69) in the definition of $\tilde{\mathbf{r}}_i$ to arrive at

$$\begin{aligned}
\|\tilde{\mathbf{r}}_i\|_2 &\leq \frac{\left[\|\mathbf{n}^{\perp}\|_2^2+\|\mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg})\mathbf{A}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\mathbf{c}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\|_2^2\right]^{\frac{1}{2}}}{\sigma^2_{\min}(\mathbf{A}_{\mathcal{S}^*})\|\mathbf{c}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\|_2} \\
&\overset{(a)}{\leq} \frac{\left[\|\mathbf{n}^{\perp}\|_2^2+\sigma^2_{\max}(\mathbf{A}_{\mathcal{S}^*})\|\mathbf{c}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\|_2^2\right]^{\frac{1}{2}}}{\sigma^2_{\min}(\mathbf{A}_{\mathcal{S}^*})\|\mathbf{c}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\|_2} \\
&= \frac{\left[\|\mathbf{n}^{\perp}\|_2^2/\|\mathbf{c}_{\mathcal{S}^{\star}\setminus\mathcal{S}^{(i)}_{psg}}\|_2^2+\sigma^2_{\max}(\mathbf{A}_{\mathcal{S}^*})\right]^{\frac{1}{2}}}{\sigma^2_{\min}(\mathbf{A}_{\mathcal{S}^*})}
\end{aligned} \tag{A.70}$$

where (a) follows from Lemma A.1.2 and the fact that $\mathbf{P}^{\perp}(\mathcal{S}^{(i)}_{psg})$ is a projection matrix. In addition,

$$\|\mathbf{n}^{\perp}\|_2 = \|\mathbf{P}^{\perp}(\mathcal{S}^*)\mathbf{n}\|_2 \leq \|\mathbf{n}\|_2 \leq \epsilon_{\mathbf{n}}. \tag{A.71}$$

Define $\hat{x}_{\min} = \min_j |\bar{\mathbf{x}}_j|$ and $\mathbf{c}_{\min} = \min_j |\mathbf{c}_j|$ where $\hat{x}_{\min}$ is the smallest entry in the best k-sparse approximation of $\mathbf{x}$. It is evident to see that

$$\mathbf{c}_{\min} \geq \hat{x}_{\min} - \|\mathbf{w}\|_2. \tag{A.72}$$

Note that by the assumption on the smallest entry of $\hat{\mathbf{x}}$ (see also (A.75)) $\mathbf{c}_{\min} > 0$. Therefore,

$$\begin{aligned} \|\mathbf{c}_{\mathcal{S}^\star \setminus \mathcal{S}^{(i)}_{psg}}\|_2^2 &\geq (k-i)\mathbf{c}_{\min}^2 \\ &\geq (k-i)\left(\hat{x}_{\min} - \|\mathbf{w}\|_2\right)^2 \\ &= (k-i)(\hat{x}_{\min} - \|\mathbf{A}^\dagger_{\mathcal{S}^*}\mathbf{n}\|_2)^2 \\ &\geq (k-i)(\hat{x}_{\min} - \sigma_{\max}(\mathbf{A}^\dagger_{\mathcal{S}^*})\|\mathbf{n}\|_2)^2 \\ &= (k-i)(\hat{x}_{\min} - \frac{\epsilon_{\mathbf{n}}}{\sigma_{\min}(\mathbf{A}_{\mathcal{S}^*})})^2. \end{aligned} \tag{A.73}$$

Let $\mathbf{Z}_3$ denote the event that $\sigma_{\max}(\mathbf{A}_{\mathcal{S}^\star}) \leq 1 + \sqrt{\frac{k}{n}} + \delta$ for some $\delta > 0$. Conditioning on $\mathbf{Z}_1 \cap \mathbf{Z}_2 \cap Z_3$, and combining (A.70), (A.71), and (A.73) implies that

$$\begin{aligned} \|\tilde{\mathbf{r}}_i\|_2 &\leq \frac{\left[\frac{\epsilon_{\mathbf{n}}^2}{(k-i)(\hat{x}_{\min} - \frac{\epsilon_{\mathbf{n}}}{\sigma_{\min}(\mathbf{A}_{\mathcal{S}^*})})^2} + \sigma^2_{\max}(\mathbf{A}_{\mathcal{S}^*})\right]^{\frac{1}{2}}}{\sigma^2_{\min}(\mathbf{A}_{\mathcal{S}^*})} \\ &\leq \frac{\left[\frac{\epsilon_{\mathbf{n}}^2}{(k-i)(\hat{x}_{\min} - (1-\sqrt{\frac{k}{n}}-\delta)^{-1}\epsilon_{\mathbf{n}})^2} + (1+\sqrt{\frac{k}{n}}+\delta)^2\right]^{\frac{1}{2}}}{(1-\sqrt{\frac{k}{n}}-\delta)^2} \end{aligned} \tag{A.74}$$

Thus, imposing the constraint

$$\hat{x}_{\min} \geq \left[(1-\sqrt{\frac{k}{n}}-\delta)^{-1} + t\right]\epsilon_{\mathbf{n}} \tag{A.75}$$

where $t > 0$ establishes

$$\|\tilde{\mathbf{r}}_i\|_2 \leq \frac{\left[\frac{1}{(k-i)t^2} + (1+\sqrt{\frac{k}{n}}+\delta)^2\right]^{\frac{1}{2}}}{(1-\sqrt{\frac{k}{n}}-\delta)^2}. \tag{A.76}$$

With the adjusted bound on $\|\tilde{\mathbf{r}}_i\|_2$ we can now proceed to finalize the proof in a near identical manner as that of Theorem 3.3.1 to arrive to the counterpart of (A.57), i.e.

$$\begin{aligned}
&\Pr\left(\max_{j\in[m]\setminus\mathcal{S}^\star} |\mathbf{a}_j^\top \bar{\mathbf{r}}_i| < \frac{c_1(\gamma)(1-\sqrt{\frac{k}{n}}-\delta)^2}{\left[\frac{1}{(k-i)t^2} + (1+\sqrt{\frac{k}{n}}+\delta)^2\right]^{\frac{1}{2}}\sqrt{k}}\right) \\
&= \Pr\left(|\mathbf{a}_1^\top \bar{\mathbf{r}}_i| < \frac{c_1(\gamma)(1-\sqrt{\frac{k}{n}}-\delta)^2}{\left[\frac{1}{(k-i)t^2} + (1+\sqrt{\frac{k}{n}}+\delta)^2\right]^{\frac{1}{2}}\sqrt{k}}\right)^{(m-k)} \\
&\geq \left(1-\exp\left(-\frac{nc_1(\gamma)^2(1-\sqrt{\frac{k}{n}}-\delta)^4}{2k\left[\frac{1}{(k-i)t^2} + (1+\sqrt{\frac{k}{n}}+\delta)^2\right]}\right)\right)^{(m-k)}.
\end{aligned} \tag{A.77}$$

where $\bar{\mathbf{r}}_i = \tilde{\mathbf{r}}_i/\|\tilde{\mathbf{r}}_i\|_2$, $\|\bar{\mathbf{r}}_i\|_2 = 1$, and we again use Lemma A.1.5 and the fact that $\mathbf{a}_j$'s are i.i.d. normal random vectors. Now upon using union bound define

$$\tilde{q}_4 = \left(1-\sum_{i=0}^{k-1}\exp\left(-\frac{nc_1(\gamma)^2(1-\sqrt{\frac{k}{n}}-\delta)^4}{2k\left[\frac{1}{(k-i)t^2} + (1+\sqrt{\frac{k}{n}}+\delta)^2\right]}\right)\right)^{(m-k)} \tag{A.78}$$

It now remains to bound the probability of the conditioning event, i.e. , This

can be done simply by noting

$$\begin{aligned}\Pr(\mathbf{Z}_1 \cap \mathbf{Z}_2 \cap \mathbf{Z}_3) &= \Pr(\mathbf{Z}_1 \cap \mathbf{Z}_2) + \Pr(\mathbf{Z}_3) - \Pr((\mathbf{Z}_1 \cap \mathbf{Z}_2) \cup \mathbf{Z}_3) \\ &\geq \Pr(\mathbf{Z}_1 \cap \mathbf{Z}_2) + \Pr(\mathbf{Z}_3) - 1 \\ &\geq \tilde{q}_1 + \Pr(\mathbf{Z}_3) - 1 \\ &\geq \left(1 - 2e^{-c_0(\gamma)n}\right)^m - 2\exp(-\delta^2 \frac{n}{2}) := \tilde{q}_3.\end{aligned} \tag{A.79}$$

Thus, we establish $\prod_{i=0}^{k-1} q_{psg}^{(i)} \geq \tilde{q}_3 \tilde{q}_4$. Therefore, with probability exceeding $\prod_{i=0}^{k-1} p_{psg}^{(i)} \prod_{i=0}^{k-1} q_{psg}^{(i)}$ we identify the indices corresponding to the top k entries in $\mathbf{x}$. This in turn implies that the approximation error satisfies

$$\|\mathbf{x}_{psg} - \hat{\mathbf{x}}\|_2^2 = \|\mathbf{A}_{\mathcal{S}^*}^{\dagger}(\mathbf{A}_{\mathcal{S}^*}\hat{\mathbf{x}} + \mathbf{e}) - \hat{\mathbf{x}}\|^2 \leq \|\mathbf{A}_{\mathcal{S}^*}^{\dagger}\mathbf{e}\|^2 \leq \frac{\|\mathbf{n}\|^2}{\sigma_{\min}(\mathbf{A}_{\mathcal{S}^*})^2} \leq \frac{\epsilon_{\mathbf{n}}}{(1 - \sqrt{\frac{k}{n}} - \delta)^2} \tag{A.80}$$

with probability exceeding $\prod_{i=0}^{k-1} p_{psg}^{(i)} \prod_{i=0}^{k-1} q_{psg}^{(i)}$.

Finally, we can obtain the order of sample complexity (which is again the optimal order of $\mathcal{O}(k \log \frac{m}{k})$ if $m > k\sqrt{k}$) using a near identical approach as the one we used in the end of proof of Theorem 3.3.1. Note however that for the case of noisy measurements case the constants are inferior compared to those in the noiseless setting, implying a more demanding sampling requirement for the former. In particular Assume, $(1 - \sqrt{\frac{k}{n}} - \delta)^2 \geq 1 - c'$ for some $c' > 0$. Then, it holds that $n \geq C_2 k$, where $C_2' := (1 - \sqrt[4]{1 - c'} + \delta)^{-2}$. Also, we have $1 + \sqrt{\frac{k}{n}} + \delta \leq 1 + \delta + 1/C_2'$. Now, define

$$C_1' = \frac{2c_1(\gamma)(1 - c)}{\frac{1}{t^2} + 1 + \delta + 1/C_2'}. \tag{A.81}$$

with these constants we obtain (3.33).

A.6 Proof of Theorem 3.3.3

First, we present Lemma A.6.1 that establishes an upper bound on the probability that ALG identifies $\mathcal{S}^\star$.

Lemma A.6.1. *Let $r < m$ and assume $\mathcal{R}_{alg}^{(i)}$ in each iteration of ALG is constructed via sampling with replacement. Then*

$$\Pr\left(\mathcal{S}_{alg}^{(k)} = \mathcal{S}^\star\right) \leq \min\left(q, 1 - \left(1 - \frac{1}{m}\right)^r\right), \tag{A.82}$$

where

$$q := \left(1 - \frac{1}{\ell}\sum_{i=i_{\min}}^{k-1} \exp\left(-\frac{r(k-i)}{m}\right)\left(1 - \frac{r(k-i)^2}{m^2}\right)\right)^\ell, \tag{A.83}$$

and $\ell = \min(k, \lfloor\sqrt{m^2/r}\rfloor)$ and $i_{\min} = k - \ell$.

Proof. Since we assume ALG uses sampling with replacement to construct $\mathcal{R}_{alg}^{(i)}$, we can compute $p_{alg}^{(i)}$ according to

$$\begin{aligned} p_{alg}^{(i)} &= 1 - \Pr\left(\mathcal{R}_{alg}^{(i)} \cap (\mathcal{S}^\star \backslash \mathcal{S}_{alg}^{(i)}) = \emptyset | \mathbf{B}_{alg}^{(i)}\right) \\ &= 1 - \left(1 - \frac{|\mathcal{S}^\star \backslash \mathcal{S}_{alg}^{(i)}|}{|\mathcal{X}|}\right)^r \\ &= 1 - \left(1 - \frac{k-i}{m}\right)^r. \end{aligned} \tag{A.84}$$

Note that since $p_{alg}^{(i)} \leq 1$, it follows that

$$\prod_{i=0}^{k-1} p_{alg}^{(i)} \leq p_{alg}^{(k-1)} = 1 - \left(1 - \frac{1}{m}\right)^r. \tag{A.85}$$

Let $\ell = \min(k, \lfloor \sqrt{m^2/r} \rfloor)$ and define $i_{\min} := k - \ell$. Then, using $p_{alg}^{(i)} \leq 1$, $0 \leq i < i_{\min}$, and Lemma A.4.1 with $a = -\frac{k-i}{m}$ and $b = r$ for $i_{\min} \leq i \leq k-1$ leads to

$$\prod_{i=0}^{k-1} p_{alg}^{(i)} \leq \prod_{i=i_{\min}}^{k-1} \left(1 - \exp\left(-\frac{r(k-i)}{m}\right)\left(1 - \frac{r(k-i)^2}{m^2}\right)\right). \tag{A.86}$$

Applying the inequality of arithmetic and geometric means yields

$$\prod_{i=0}^{k-1} p_{alg}^{(i)} \leq \left(1 - \frac{1}{\ell}\sum_{i=i_{\min}}^{k-1} \exp\left(-\frac{r(k-i)}{m}\right)\left(1 - \frac{r(k-i)^2}{m^2}\right)\right)^{\ell}. \tag{A.87}$$

Finally, we obtain the bound stated in (A.82) by taking the minimum of (A.85) and (A.87). ■

We now proceed to the proof of Theorem 3.3.3. First, consider the setting where $r \leq k^{\alpha-1}m$, $0 < \alpha < 1$. Since by Lemma A.6.1 we have (A.82), to establish the result (i) it suffices to show that

$$\limsup_{m,k\to\infty} 1 - \left(1 - \frac{1}{m}\right)^r = 0. \tag{A.88}$$

Using Lemma A.4.1 yields

$$\begin{aligned} 1 - \left(1 - \frac{1}{m}\right)^r &\leq 1 - \exp\left(-\frac{r}{m}\right)\left(1 - \frac{r}{m^2}\right) \\ &\leq 1 - \exp\left(-k^{\alpha-1}\right)\left(1 - \frac{k^{\alpha-1}}{m}\right), \end{aligned} \tag{A.89}$$

where for the last inequality we recall the assumption $r \leq k^{\alpha-1}m$. The result is then established by noting $\limsup_{m,k\to\infty} \frac{k^{\alpha-1}}{m} = 0$, $\limsup_{k\to\infty} \exp\left(-k^{\alpha-1}\right) = 1$, and using the squeeze theorem.

Next, consider the setting in (ii), i.e., $r \leq \alpha_1 m$, $0 < \alpha_1 < 1$. Following a similar approach, one obtains

$$\Pr\left(\mathcal{S}_{alg}^{(k)} = \mathcal{S}^{\star}\right) \leq 1 - \exp\left(-\alpha_1\right)\left(1 - \frac{\alpha_1}{m}\right). \tag{A.90}$$

Since the bound in (A.90) converges to $1 - \exp\left(-\alpha_1\right)$ as $m, k \to \infty$, it holds that $\Pr\left(\mathcal{S}_{alg}^{(k)} = \mathcal{S}^{\star}\right) \leq 1 - \exp\left(-\alpha_1\right) := \delta_2 < 0.63$.

Finally, we establish the lower bound on $\Pr\left(\mathcal{S}_{alg}^{(k)} = \mathcal{S}^{\star}\right)$. It holds that

$$\begin{aligned}
\Pr\left(\mathcal{S}_{alg}^{(k)} = \mathcal{S}^{\star}\right) &= \prod_{i=0}^{k-1} p_{alg}^{(i)} \\
&\geq \prod_{i=0}^{k-1} \left(1 - \exp(-r\frac{k-i}{m})\right) \\
&= \prod_{i=0}^{k-1} \left(1 - \exp(-\alpha_1(k-i))\right) \\
&= \prod_{n=1}^{k} \left(1 - \exp(-\alpha_1 n)\right).
\end{aligned} \tag{A.91}$$

A crude lower bound can be easily achieved by replacing the product terms with the smallest term, i.e., $1 - \exp(-\alpha_1)$. However, such lower bound converges to zero as $k \to \infty$. Therefore, we resort to a different approach. Note that for every $0 < \alpha_1 < 1$, there exists a positive integer $q(\alpha_1)$ such that $\exp(\alpha_1 n) \geq n^2$ for all $n \geq q(\alpha_1)$. For instance, if $\alpha_1 \geq \log(2) \approx 0.6931$, then $q(\alpha_1) \leq 4$. Thus,

$$\prod_{n=q(\alpha_1)}^{\infty} \left(1 - \frac{1}{n^2}\right) \leq \prod_{n=q(\alpha_1)}^{\infty} \left(1 - \frac{1}{\exp(\alpha_1 n)}\right). \tag{A.92}$$

It then follows

$$\begin{aligned}
\limsup_{m,k\to\infty} \Pr\left(\mathcal{S}_{alg}^{(k)} = \mathcal{S}^{\star}\right) &\geq \prod_{n=1}^{\infty}\left(1-\frac{1}{\exp(\alpha_1 n)}\right) \\
&\geq \left(1-\frac{1}{\exp(\alpha_1)}\right)\prod_{n=2}^{q(\alpha_1)-1}\frac{\left(1-\frac{1}{\exp(\alpha_1 n)}\right)}{\left(1-\frac{1}{n^2}\right)}\prod_{n=2}^{\infty}\left(1-\frac{1}{n^2}\right) \quad \text{(A.93)} \\
&= \frac{1}{2}\left(1-\frac{1}{\exp(\alpha_1)}\right)\prod_{n=2}^{q(\alpha_1)-1}\frac{\left(1-\frac{1}{\exp(\alpha_1 n)}\right)}{\left(1-\frac{1}{n^2}\right)} := \delta_1 > 0.
\end{aligned}$$

For instance, if $\alpha_1 \geq \log(2)$, then $\delta_1 \geq 0.2461$.

Appendix B

Missing Proofs from Chapter 5

B.1 Proof of Proposition 5.2.1

First, note that

$$f(\emptyset) = \mathrm{Tr}\left(\mathbf{P}_{k|k-1} - \mathbf{F}_{\emptyset}^{-1}\right) = \mathrm{Tr}\left(\mathbf{P}_{k|k-1} - \mathbf{P}_{k|k-1}\right) = 0. \tag{B.1}$$

Now, for $j \in [n]\backslash S$ it holds that

$$\begin{aligned} f_j(\mathcal{S}) &= f(\mathcal{S}\cup\{j\}) - f(\mathcal{S}) \\ &= \mathrm{Tr}\left(\mathbf{P}_{k|k-1} - \mathbf{F}_{\mathcal{S}\cup\{j\}}^{-1}\right) - \mathrm{Tr}\left(\mathbf{P}_{k|k-1} - \mathbf{F}_{\mathcal{S}}^{-1}\right) \\ &= \mathrm{Tr}\left(\mathbf{F}_{\mathcal{S}}^{-1}\right) - \mathrm{Tr}\left(\mathbf{F}_{\mathcal{S}\cup\{j\}}^{-1}\right) \\ &= \mathrm{Tr}\left(\mathbf{F}_{\mathcal{S}}^{-1}\right) - \mathrm{Tr}\left(\left(\mathbf{F}_{\mathcal{S}} + \sigma_j^{-2}\mathbf{h}_{k,j}\mathbf{h}_{k,j}^{\top}\right)^{-1}\right) \\ &\overset{(a)}{=} \mathrm{Tr}\left(\frac{\mathbf{F}_{\mathcal{S}}^{-1}\mathbf{h}_{k,j}\mathbf{h}_{k,j}^{\top}\mathbf{F}_{\mathcal{S}}^{-1}}{\sigma_j^2 + \mathbf{h}_{k,j}^{\top}\mathbf{F}_{\mathcal{S}}^{-1}\mathbf{h}_{k,j}}\right) \\ &\overset{(b)}{=} \frac{\mathbf{h}_{k,j}^{\top}\mathbf{F}_{\mathcal{S}}^{-2}\mathbf{h}_{k,j}}{\sigma_j^2 + \mathbf{h}_{k,j}^{\top}\mathbf{F}_{\mathcal{S}}^{-1}\mathbf{h}_{k,j}} \end{aligned} \tag{B.2}$$

where (a) is obtained by applying matrix inversion lemma (Sherman-Morrison formula) [173] to $(\mathbf{F}_{\mathcal{S}} + \sigma_j^{-2}\mathbf{h}_{k,j}\mathbf{h}_{k,j}^{\top})^{-1}$, and (b) follows from the properties of the matrix trace operator. Finally, since $\mathbf{F}_{\mathcal{S}}$ is a symmetric positive definite matrix, $f_j(\mathcal{S}) > 0$ which in turn implies monotonicity.

B.2 Proof of Lemma 5.4.1

First, we aim to bound the probability of an event that a random set R contains at least one index from the optimal set of sensors which is a necessary condition to reach the optimal MSE. Let us consider $S_t^{(i)}$, the set of sensors selected by the end of i^{th} iteration of Algorithm 4 and let $\Phi = R \cap (\mathcal{S}_k^* \backslash S_t^{(i)})$. It holds that[1]

$$\begin{aligned}\Pr\{\Phi = \emptyset\} &= \prod_{l=0}^{s-1} \left(1 - \frac{|\mathcal{S}_k^* \backslash \mathcal{S}_k^{(i)}|}{|[n] \backslash \mathcal{S}_k^{(i)}| - l}\right) \\ &\overset{(a)}{\leq} \left(1 - \frac{|\mathcal{S}_k^* \backslash \mathcal{S}_k^{(i)}|}{s} \sum_{l=0}^{s-1} \frac{1}{|[n] \backslash \mathcal{S}_k^{(i)}| - l}\right)^s \\ &\overset{(b)}{\leq} \left(1 - \frac{|\mathcal{S}_k^* \backslash \mathcal{S}_k^{(i)}|}{s} \sum_{l=0}^{s-1} \frac{1}{n-l}\right)^s\end{aligned} \tag{B.3}$$

where (a) holds due to the inequality of arithmetic and geometric means, and (b) holds since $|[n] \backslash S_i| \leq n$. Now recall that for any integer p,

$$H_p = \sum_{l=1}^{p} \frac{1}{p} = \log p + \gamma + \zeta_p, \tag{B.4}$$

where H_p is the p^{th} harmonic number, γ is the Euler-Mascheroni constant, and $\zeta_p = \frac{1}{2p} - \mathcal{O}(\frac{1}{p^4})$ is a monotonically decreasing sequence related to Hurwitz

[1] Without a loss of generality, we assume that s is an integer.

zeta function [230]. Therefore, using the identity (B.4) we obtain

$$\begin{aligned}\Pr\{\Phi=\emptyset\} &\leq (1-\frac{|\mathcal{S}_k^*\backslash\mathcal{S}_k^{(i)}|}{s}(H_n-H_{n-s}))^s \\ &= (1-\frac{|\mathcal{S}_k^*\backslash\mathcal{S}_k^{(i)}|}{s}(\log(\frac{n}{n-s})+\zeta_n-\zeta_{n-s}))^s \\ &\overset{(c)}{\leq} (1-\frac{|\mathcal{S}_k^*\backslash\mathcal{S}_k^{(i)}|}{s}(\log(\frac{n}{n-s})-\frac{s}{2n(n-s)}))^s \\ &\overset{(d)}{\leq} ((1-\frac{s}{n})e^{\frac{s}{2n(n-s)}})^{|\mathcal{S}_k^*\backslash\mathcal{S}_k^{(i)}|},\end{aligned} \tag{B.5}$$

where (c) follows since $\zeta_n-\zeta_{n-s}=\frac{1}{2n}-\frac{1}{2(n-s)}+\mathcal{O}(\frac{1}{(n-s)^4})$, and (d) is due to the fact that $(1+x)^y\leq e^{xy}$ for any real number $y\geq 1$. Next, the fact that $\log(1-x)\leq -x-\frac{x^2}{2}$ for $0<x<1$ yields

$$(1-\frac{s}{n})e^{\frac{s}{2n(n-s)}}\leq e^{-\frac{\beta_1 s}{n}}, \tag{B.6}$$

where $\beta_1=1+(\frac{s}{2n}-\frac{1}{2(n-s)})$. On the other hand, we can also upper bound $\Pr\{\Phi=\emptyset\}$ as

$$\begin{aligned}\Pr\{\Phi=\emptyset\} &\leq \left(1-\frac{|\mathcal{S}_k^*\backslash\mathcal{S}_k^{(i)}|}{s}\sum_{l=0}^{s-1}\frac{1}{n-l}\right)^s \\ &\leq \left(1-\frac{|\mathcal{S}_k^*\backslash\mathcal{S}_k^{(i)}|}{n}\right)^s \\ &\leq e^{-\frac{s}{n}|\mathcal{S}_k^*\backslash\mathcal{S}_k^{(i)}|},\end{aligned} \tag{B.7}$$

where we again employed the inequality $(1+x)^y\leq e^{xy}$. Let us denote $\beta=\max\{1,\beta_1\}$. Then

$$\Pr\{\Phi\neq\emptyset\}\geq 1-e^{-\frac{\beta s}{n}|\mathcal{S}_k^*\backslash\mathcal{S}_k^{(i)}|}\geq\frac{1-\epsilon^\beta}{K}(|\mathcal{S}_k^*\backslash\mathcal{S}_k^{(i)}|), \tag{B.8}$$

by the definition of s and the fact that $1-e^{-\frac{\beta s}{n}x}$ is a concave function. Finally, according to Lemma 2 in [32],

$$\mathbb{E}[f_{(i+1)_s}(\mathcal{S}_k^{(i)})|\mathcal{S}_k^{(i)}] \geq \frac{\Pr\{\Phi \neq \emptyset\}}{|\mathcal{S}_k^* \backslash \mathcal{S}_k^{(i)}|} \sum_{j \in \mathcal{S}_k^* \backslash \mathcal{S}_k^{(i)}} f_o(\mathcal{S}_k^{(i)}). \tag{B.9}$$

Combining (B.8) and (B.9) yields the stated results.

Appendix C

Missing Proofs from Chapter 6

First, we introduce notation and state a few useful facts. Let $\mathcal{L}$ be a *linear subspace* of $d \times n$ matrices having identical columns with the projection operator $\mathcal{P}_{\mathcal{L}}(.)$ such that for all $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ we have $\mathcal{P}_{\mathcal{L}}(X) = [\bar{\mathbf{x}}, \ldots, \bar{\mathbf{x}}]$, where $\bar{\mathbf{x}} = \frac{1}{n}\sum_i \mathbf{x}_i$.

It then becomes evident that the update rule of the average iterate can be written equivalently as

$$\begin{aligned} \bar{\mathbf{X}}_{t+1} &= \mathcal{P}_{\mathcal{L}}(\bar{\mathbf{X}}_t - \eta \nabla F(\bar{\mathbf{X}}_t)) \\ &= \bar{\mathbf{X}}_t - \eta \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)) \end{aligned} \tag{C.1}$$

where $\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)) = \frac{1}{n}[\nabla f(\bar{\mathbf{x}}_t), \ldots, \nabla f(\bar{\mathbf{x}}_t)]$. While the second line does not hold in general, it does in our case due to the definition of $\mathcal{P}_{\mathcal{L}}$ and the fact that $\bar{\mathbf{X}}_t \in \mathcal{L}$. Note that due to the defined projection operator, for $\mathbf{X}^* = [\mathbf{x}^*, \ldots, \mathbf{x}^*]$ where $\mathbf{x}^*$ is any of the global optima, it holds that $\mathbf{X}^* \in \mathcal{L}$ and $\mathcal{P}_{\mathcal{L}}(\nabla F(\mathbf{X}^*)) = \mathbf{0} \in \mathcal{L}$.

Next, recall the non-expansiveness of projection (see Lemma 2.2.7 and Corollary 2.2.3 in [59]),

$$\|\mathcal{P}_{\mathcal{L}}(\mathbf{X}) - \mathcal{P}_{\mathcal{L}}(\mathbf{Y})\| \leq \|\mathbf{X} - \mathbf{Y}\|. \tag{C.2}$$

Finally, we explore smoothness and strong convexity of $F(\cdot)$ (in the convex case). Since $\bar{\mathbf{X}}_i \in \mathcal{L}$ for all $i \in [T]$ and $\bar{\mathbf{X}}^* = \mathbf{X}^* \in \mathcal{L}$, by strong convexity and smoothness of $f(\cdot)$ it holds that

$$F(\bar{\mathbf{X}}_i) \leq F(\bar{\mathbf{X}}_j) + \langle \bar{\mathbf{X}}_i - \bar{\mathbf{X}}_j, \nabla F(\bar{\mathbf{X}}_j) \rangle + \frac{L}{2} \|\bar{\mathbf{X}}_i - \bar{\mathbf{X}}_j\|^2 \tag{C.3}$$

$$F(\bar{\mathbf{X}}_i) \geq F(\bar{\mathbf{X}}_j) + \langle \bar{\mathbf{X}}_i - \bar{\mathbf{X}}_j, \nabla F(\bar{\mathbf{X}}_j) \rangle + \frac{\mu}{2} \|\bar{\mathbf{X}}_i - \bar{\mathbf{X}}_j\|^2 \tag{C.4}$$

since $\|\bar{\mathbf{X}}_i - \bar{\mathbf{X}}_j\|^2 = n\|\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j\|^2$. Therefore, $F(\cdot)$ is μ-strongly convex (in the convex scenario) and L-smooth on $\mathcal{L}$ and $\hat{\mu}$-strongly convex (in the convex scenario) and $\hat{L}$-smooth everywhere else.

Remark C.0.1. In the following proofs, for simplicity we make the simplifying assumption that the clients initialize their parameters such that $F(\mathbf{X}_0^{(Q)}) = F(\bar{\mathbf{X}}_0)$. This can hold easily by setting all initial vectors to be equal to a vector $\mathbf{x}_0 \in \mathbb{R}^d$.

C.1 Proof of Theorem 6.3.1

The proof relies on a perturbation analysis and interpreting the iterates of the proposed scheme as random perturbations of a virtual sequence $\bar{\mathbf{X}}_t$ having identical columns. Then, leveraging linear convergence of the gossip subroutine and the fact $F(\bar{\mathbf{X}}_t) - f^*$ decreases linearly, we can show linear convergence of the expected difference $\mathbb{E}_{\mathcal{C}}\|\bar{\mathbf{X}}_t - \mathbf{X}_t^{(Q)}\|$ and, in turn, the error of the proposed scheme. Recall that

$$f^* := \min_{\mathbf{x}} \left[f(\mathbf{x}) := \sum_{i=1}^{n} f_i(\mathbf{x}) \right]. \tag{C.5}$$

We first state a number of intermediate lemmas

C.1.1 Useful lemmas

Lemma C.1.1 establishes linear convergence of the virtual average sequence in terms of the suboptimality of the function values.

Lemma C.1.1. *Let* $\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{X}}_t - \eta \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))$ *denote a sequence such that* $\bar{\mathbf{X}}_t = [\bar{\mathbf{x}}_t, \ldots, \bar{\mathbf{x}}_t]$*, and let* $\eta \leq 1/L$*. Then*

$$F(\bar{\mathbf{X}}_t) - f^* \leq [F(\bar{\mathbf{X}}_0) - f^*] \left(1 - 2\frac{\mu}{n}\eta(1 - \frac{L\eta}{2})\right)^t. \tag{C.6}$$

Proof. First, the equivalent update rule for $\bar{\mathbf{X}}_t$, i.e.

$$\bar{\mathbf{X}}_{t+1} = \arg\min_{\bar{\mathbf{Y}} \in \mathcal{L}} \{g(\bar{\mathbf{Y}}) := F(\bar{\mathbf{X}}_t) + \langle \nabla F(\bar{\mathbf{X}}_t), \bar{\mathbf{Y}} - \bar{\mathbf{X}}_t \rangle + \frac{1}{2\eta}\|\bar{\mathbf{Y}} - \bar{\mathbf{X}}_t\|^2\}. \tag{C.7}$$

to establish a useful smoothness result.

From the optimally condition for the convex function $g(\bar{\mathbf{Y}})$ (see Theorem 2.2.9 in [59]) and the fact that $\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{X}}_t - \eta \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))$, it follows that

$$\langle \nabla F(\bar{\mathbf{X}}_t) - \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \bar{\mathbf{Y}} - \bar{\mathbf{X}}_{t+1} \rangle \geq 0 \tag{C.8}$$

for any $\bar{\mathbf{Y}} \in \mathcal{L}$. Now, using this result and choosing $\bar{\mathbf{Y}} = \bar{\mathbf{X}}_t \in \mathcal{L}$ we have

$$\begin{aligned} F(\bar{\mathbf{X}}_t) &\geq F(\bar{\mathbf{X}}_t) + \langle \nabla F(\bar{\mathbf{X}}_t) - \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \bar{\mathbf{X}}_{t+1} - \bar{\mathbf{X}}_t \rangle \\ &\geq g(\bar{\mathbf{X}}_{t+1}) - \frac{1}{2\eta}\|\bar{\mathbf{X}}_{t+1} - \bar{\mathbf{X}}_t\|^2 - \langle \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \bar{\mathbf{X}}_{t+1} - \bar{\mathbf{X}}_t \rangle \\ &= g(\bar{\mathbf{X}}_{t+1}) + \frac{\eta}{2}\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2, \end{aligned} \tag{C.9}$$

where we used the definition of $g(\cdot)$ and the update rule

$$\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{X}}_t - \eta \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)). \tag{C.10}$$

Now, L-smoothness of $F(\cdot)$ on $\mathcal{L}$ and the fact that $\eta \leq 1/L$ imply

$$\begin{aligned} F(\bar{\mathbf{X}}_t) &\geq F(\bar{\mathbf{X}}_{t+1}) + (\frac{1}{2\eta} - \frac{L}{2})\|\bar{\mathbf{X}}_{t+1} - \bar{\mathbf{X}}_t\|^2 + \frac{\eta}{2}\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2 \\ &= F(\bar{\mathbf{X}}_{t+1}) + \eta(1 - \frac{\eta L}{2})\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2. \end{aligned} \tag{C.11}$$

Finally, using the definition of $\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))$, we can relate this last result to the gradient of f at $\bar{x}_t$ to obtain

$$F(\bar{\mathbf{X}}_{t+1}) \leq F(\bar{\mathbf{X}}_t) - \frac{\eta}{n}(1 - \frac{L\eta}{2})\|\nabla f(\bar{\mathbf{x}}_t)\|_2^2. \tag{C.12}$$

Having established the above *smoothness* result and recalling that $f(\cdot)$ satisfies PLC we have

$$\|\nabla f(\bar{\mathbf{x}}_t)\|_2^2 \geq 2\mu[f(\bar{\mathbf{x}}_t) - f^*]. \tag{C.13}$$

Subtracting f^* from both sides of (C.12), using (C.13), and noting $F(\bar{\mathbf{X}}_t) = f(\bar{\mathbf{x}}_t)$ yields

$$F(\bar{\mathbf{X}}_{t+1}) - f^* \leq [F(\bar{\mathbf{X}}_t) - f^*]\left(1 - 2\frac{\mu}{n}\eta(1 - \frac{L\eta}{2})\right). \tag{C.14}$$

Finally, recursively applying the above result establishes the stated expression.

■

The next Lemma establishes a bound on the gradient norm that will assist us in the proof of Theorem 6.3.1.

Lemma C.1.2. *Recall F is L-smooth and satisfies PLC with parameter μ on $\mathcal{L}$. For any $\bar{\mathbf{X}} \in \mathcal{L}$, let $\bar{\mathbf{X}}^*$ be the projection of $\bar{\mathbf{X}}$ on the solution set of $\min_{\mathbf{x}} f(\mathbf{x})$. Then, we have*

$$\|\nabla F(\bar{\mathbf{X}}) - \nabla F(\bar{\mathbf{X}}^*)\|^2 \leq \frac{L^2}{2\mu}(F(\bar{\mathbf{X}}) - f^*). \tag{C.15}$$

Proof. First, by smoothness we have

$$\|\nabla F(\bar{\mathbf{X}}) - \nabla F(\bar{\mathbf{X}}^*)\|^2 \leq L^2 \|\bar{\mathbf{X}} - \bar{\mathbf{X}}^*\|^2. \tag{C.16}$$

Additionally, PLC implies that F satisfies the so-called quadratic growth condition (see Appendix A in [58] as well as the related works [233, 234]),

$$2\mu \|\bar{\mathbf{X}} - \bar{\mathbf{X}}^*\|^2 \leq F(\bar{\mathbf{X}}) - F^*, \text{ where } F^* = \min_X F(\mathbf{X}). \tag{C.17}$$

Combining these two results and noting $F^* = f^*$ establishes the claim of the lemma. ■

The next two Lemmas collectively establish a bound on amount of perturbation of the DeLi-CoCo's iterates $\mathbf{X}_t^{(Q)}$ compared to $\bar{\mathbf{X}}_t$.

Lemma C.1.3. *Let $\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{X}}_t - \eta \nabla F(\bar{\mathbf{X}}_t)$. Under the conditions of DeLi-CoCo, with $\gamma = \frac{\delta\omega}{\beta(\delta,\omega)}$, it holds that*

$$\begin{aligned} &\mathbb{E}_{\mathcal{C}} \|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}_t\|^2 + \mathbb{E}_{\mathcal{C}} \|\mathbf{X}_t^{(Q)} - \mathbf{Z}_t^{(Q)}\|^2 \\ &\leq (1 - \frac{\delta\gamma}{2})^Q \left(\mathbb{E}_{\mathcal{C}} \|\mathbf{X}_t^{(0)} - \bar{\mathbf{X}}_t\|^2 + \mathbb{E}_{\mathcal{C}} \|\mathbf{X}_t^{(0)} - \mathbf{Z}_t^{(0)}\|^2 \right), \end{aligned} \tag{C.18}$$

where $\beta(\delta, \omega) = 16\delta + \delta^2 - 8\delta\omega + (4 + 2\delta)\lambda_{\max}^2(\mathbf{I} - \mathbf{W})$.

Proof. See Theorem 2 and its proof in [39]. The main ingredients of the proof are: (i) the fact that the error feedback sequence $\mathbf{Z}_t^{(Q)}$ approaches $\mathbf{X}_t^{(Q)}$ due to the contraction property of the compression operator $\mathcal{C}$, and (ii) the linear mixing rate of the gossiping matrix W that is determined by the spectral gap δ. ■

Lemma C.1.4. *Let* $\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{X}}_t - \eta \nabla F(\bar{\mathbf{X}}_t)$ *and let* $\{\mathbf{X}_t^{(Q)}\}$ *denote the sequence generated by DeLi-CoCo. Let* $e_t^2 = \mathbb{E}_{\mathcal{C}} \|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}_t\|^2 + \mathbb{E}_{\mathcal{C}} \|\mathbf{X}_t^{(Q)} - \mathbf{Z}_t^{(Q)}\|^2$. *For any* $0 < \eta \leq 1/L$, *let* Q *be such that*

$$\zeta := \rho - \xi > 0, \quad \rho := 1 - 2\frac{\mu}{n}\eta(1 - \frac{L\eta}{2}) < 1, \quad \xi := (1 - \frac{\delta\gamma}{2})^Q (3 + 20\eta^2 \hat{L}^2) < 1. \tag{C.19}$$

Then, it holds that

$$e_t^2 \leq \frac{13\eta^2 L^2}{2\mu\zeta}(1 - \frac{\delta\gamma}{2})^Q [F(\bar{\mathbf{X}}_0) - f^*]\rho^t + e_0^2 \rho^t + \frac{20\eta^2\Delta^2}{1-\xi}(1 - \frac{\delta\gamma}{2})^Q. \tag{C.20}$$

Additionally, if all nodes initialize such that $\mathbf{X}_0^Q = 0$, *by considering the dominant terms in above we have*

$$e_t^2 = \mathcal{O}\left(\frac{\eta^2 L^2}{\mu\zeta}(1 - \frac{\delta\gamma}{2})^Q [F(\bar{\mathbf{X}}_0) - f^*]\rho^t + \frac{\eta^2\Delta^2}{1-\xi}(1 - \frac{\delta\gamma}{2})^Q\right), \tag{C.21}$$

where the $\mathcal{O}$ *notation does not hide any terms depending on* Q *or* t.

Before presenting the proof we highlight again that if f is interpolating [211–213], e.g. an overparameterized neural network or a function satisfying the growth condition [214, 215], then $\Delta = 0$ and the second term disappears. Additionally, if there is no communication compression and the graph is fully connected ($\delta = 1$), the term $1 - \gamma/2$ can be improved to $1 - \gamma$ (see, e.g. [55]). Therefore, by using the gossip learning rate $\gamma = 1$, the second term in the error bound collapses to 0. Further, (C.19) is not necessary.

Proof. It holds by definitions $\mathbf{X}_{t+1}^{(0)} = \mathbf{X}_t^{(Q)} - \eta \nabla F(\mathbf{X}_t^{(Q)})$ and $\mathbf{Z}_{t+1}^{(0)} = \mathbf{Z}_t^{(Q)}$ that

$$\begin{aligned}
&\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(0)} - \bar{\mathbf{X}}_{t+1}\|^2 + \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(0)} - \mathbf{Z}_{t+1}^{(0)}\|^2 \\
&= \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}_{t+1} - \bar{\mathbf{X}}_t + \bar{\mathbf{X}}_t - \eta \nabla F(\mathbf{X}_t^{(Q)})\|^2 \\
&\quad + \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{Z}_t^{(Q)} - \eta \nabla F(\mathbf{X}_t^{(Q)})\|^2 \\
&\leq 3e_t^2 + 3\eta^2 \|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2 + 5\eta^2 \mathbb{E}_{\mathcal{C}}\|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\bar{\mathbf{X}}_t) + \nabla F(\bar{\mathbf{X}}_t)\|^2,
\end{aligned} \tag{C.22}$$

where we used the fact that $\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{Z}_t^{(Q)}\|^2 \geq 0$ and the smoothness of $F(\cdot)$. Let $\bar{\mathbf{X}}_t^*$ be the projection of $\bar{\mathbf{X}}_t$ to the optimal set. Now, we proceed by using the smoothness property, the non-expansiveness property of projection (cf. (C.2)) as well as the fact that $\mathcal{P}_{\mathcal{L}}(\nabla F(\mathbf{X}^*)) = \mathbf{0}$ to obtain

$$\begin{aligned}
&\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(0)} - \bar{\mathbf{X}}_{t+1}\|^2 + \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(0)} - \mathbf{Z}_{t+1}^{(0)}\|^2 \\
&\quad \leq 3e_t^2 + 3\eta^2 \|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)) - \mathcal{P}_{\mathcal{L}}(\nabla F(\mathbf{X}^*))\|^2 \\
&\qquad + 5\eta^2 \mathbb{E}_{\mathcal{C}}\|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\bar{\mathbf{X}}_t) + \nabla F(\bar{\mathbf{X}}_t)\|^2 \\
&\quad \leq 3e_t^2 + 13\eta^2 \|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*)\|^2 \\
&\qquad + 10\eta^2 \mathbb{E}_{\mathcal{C}}\|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\bar{\mathbf{X}}_t) + \nabla F(\mathbf{X}^*)\|^2 \\
&\quad \leq 3e_t^2 + 13\eta^2 \|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*)\|^2 \\
&\qquad + 20\hat{L}^2\eta^2 \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}_t\|^2 + 20\eta^2 \|\nabla F(\mathbf{X}^*)\|^2 \\
&\quad \leq 3e_t^2 + 13\eta^2 \|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*)\|^2 \\
&\qquad + 20\hat{L}^2\eta^2 (\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}_t\|^2 + \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{Z}_t^{(Q)}\|^2) + 20\eta^2 \|\nabla F(\mathbf{X}^*)\|^2 \\
&\quad \leq (3 + 20\eta^2\hat{L}^2)e_t^2 + 13\eta^2 \|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\bar{\mathbf{X}}_t^*)\|^2 + 20\eta^2\Delta^2.
\end{aligned} \tag{C.23}$$

To bound $\|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\bar{\mathbf{X}}_t^*)\|$ we can use the result of Lemma C.1.2. Thus,

by Lemma C.1.3 and the above result we obtain the recursion

$$\begin{aligned} e_{t+1}^2 &\leq (1-\frac{\delta\gamma}{2})^Q \left((3+20\eta^2\hat{L}^2)e_t^2 + \frac{13}{2\mu}L^2\eta^2(F(\bar{\mathbf{X}}_t)-f^*) + 20\eta^2\Delta^2 \right) \\ &:= \xi e_t^2 + \nu_t + u, \end{aligned} \tag{C.24}$$

where

$$\xi := (1-\frac{\delta\gamma}{2})^Q(3+20\eta^2\hat{L}^2), \tag{C.25}$$

$$\nu_t := \frac{13}{2\mu}(1-\frac{\delta\gamma}{2})^Q L^2\eta^2(F(\bar{\mathbf{X}}_t)-f^*) \tag{C.26}$$

is a linearly decreasing sequence, i.e. $\nu_t \leq \rho^t\nu_0$, where by Lemma C.1.1

$$\rho := 1-2\frac{\mu}{n}\eta(1-\frac{L\eta}{2}), \qquad \nu_0 := \frac{13}{2\mu}(1-\frac{\delta\gamma}{2})^Q L^2\eta^2[F(\bar{\mathbf{X}}_0)-f^*], \tag{C.27}$$

and

$$u := 20(1-\frac{\delta\gamma}{2})^Q\eta^2\Delta^2. \tag{C.28}$$

Given the fact that ν_t vanishes linearly, we expect e_t^2 to converge linearly because for a large enough Q, we have $\xi < 1$ (see the conditions in the statement of Theorem 6.3.1). We now prove this statement using induction.

Define h_t such that $h_0 = e_0^2$, and let $h_{t+1} = ah_t + \nu_t$. Using simple algebra it follows that

$$h_t = \xi^t e_0^2 + \sum_{i=0}^{t-1} \xi^{t-i-1}\rho^i\nu_0. \tag{C.29}$$

Similarly, we can expand the recursion of e_t^2 to obtain

$$e_t^2 \leq \xi^t e_0^2 + \sum_{i=0}^{t-1} \xi^{t-i-1}\rho^i\nu_0 + u\sum_{i=0}^{t-1}\xi^i \leq h_t + \frac{u}{1-\xi}. \tag{C.30}$$

Note that if h_t linearly converges to zero, then e_t^2 linearly converges as well. Since $\rho - \xi = \zeta > 0$ by assumption, using simple algebra we can show

$$h_t \leq \rho^t (e_0^2 + \frac{\nu_0}{\zeta}). \tag{C.31}$$

Thus,

$$e_t^2 \leq \rho^t (e_0^2 + \frac{\nu_0}{\zeta}) + \frac{u}{1-\xi}, \tag{C.32}$$

and the proof is complete by noting the definitions of ρ, ξ, ν_0, u, and the fact that by definition $e_0^2 = \|\mathbf{X}_0^{(Q)}\|^2$. ■

C.1.2 Proof of the main theorem

The main challenge in decentralized learning under PLC is that we cannot use the co-coercivity property. To this end, we leverage the fact that PLC relates to suboptimality of the function value and exploit a judiciously chosen step size to simplify the analysis.

Let $\mathbf{X}_t^{(q)} = \bar{\mathbf{X}}_t + \mathbf{E}_t^{(q)}$ for some (random) error matrix $\mathbf{E}_t^{(q)} \in \mathbb{R}^{d \times n}$. By $\hat{L}$-smoothness of $F(\cdot)$ for all X and the fact that $\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{X}}_t - \eta \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))$ we have

$$F(\mathbf{X}_{t+1}^{(Q)}) \leq F(\bar{\mathbf{X}}_t) + \langle -\eta \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)) + \mathbf{E}_{t+1}^{(Q)}, \nabla F(\bar{\mathbf{X}}_t) \rangle + \frac{\hat{L}}{2} \|\eta \nabla F(\bar{\mathbf{X}}_t) - \mathbf{E}_{t+1}^{(Q)}\|^2. \tag{C.33}$$

Let $\eta = 1/\hat{L} \leq 1/L$ and hence the condition of Lemma C.1.1 is satisfied.

Expanding the last inequality by using $\eta = 1/\hat{L}$ we obtain

$$F(\mathbf{X}_{t+1}^{(Q)}) \leq F(\bar{\mathbf{X}}_t) - \frac{1}{\hat{L}}\langle \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \nabla F(\bar{\mathbf{X}}_t)\rangle + \langle \nabla F(\bar{\mathbf{X}}_t) - \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \mathbf{E}_{t+1}^{(Q)}\rangle + \frac{\hat{L}}{2}\|\mathbf{E}_{t+1}^{(Q)}\|^2 + \frac{1}{2\hat{L}}\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2. \tag{C.34}$$

Next, we need to take care of the cross terms. First, note that

$$\begin{aligned} &- \langle \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \nabla F(\bar{\mathbf{X}}_t)\rangle \\ &= \langle -\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \nabla F(\bar{\mathbf{X}}_t) + \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)) - \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\rangle \\ &= -\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2 + \langle \mathbf{0} - \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \nabla F(\bar{\mathbf{X}}_t) - \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\rangle \\ &\leq -\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2, \end{aligned} \tag{C.35}$$

where the last inequality follows by the variational characterization of the projection (see Lemma 2.2.7 in [59]), i.e.

$$\langle \mathbf{Y} - \mathcal{P}_{\mathcal{L}}(\mathbf{Y}), \mathbf{X} - \mathcal{P}_{\mathcal{L}}(\mathbf{Y})\rangle \leq 0, \quad \mathbf{X} \in \mathcal{L}, \tag{C.36}$$

and the fact that $\mathbf{0} \in \mathcal{L}$.

We now bound the second cross term in (C.34). From Young's inequality, for all $\alpha > 0$ it holds that

$$
\begin{aligned}
&\langle \nabla F(\bar{\mathbf{X}}_t) - \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \mathbf{E}_{t+1}^{(Q)} \rangle \\
&\leq \frac{\alpha}{2}\|\mathbf{E}_{t+1}^{(Q)}\|^2 + \frac{1}{2\alpha}\|\nabla F(\bar{\mathbf{X}}_t) - \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2 \\
&= \frac{\alpha}{2}\|\mathbf{E}_{t+1}^{(Q)}\|^2 + \frac{1}{2\alpha}\|\nabla F(\bar{\mathbf{X}}_t)\|^2 + \frac{1}{2\alpha}\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2 \\
&\quad - \frac{1}{\alpha}\langle \mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)), \nabla F(\bar{\mathbf{X}}_t) \rangle \\
&\leq \frac{\alpha}{2}\|\mathbf{E}_{t+1}^{(Q)}\|^2 + \frac{1}{2\alpha}\|\nabla F(\bar{\mathbf{X}}_t)\|^2 - \frac{1}{2\alpha}\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2 \quad \text{(Using (C.35))} \\
&\leq \frac{\alpha}{2}\|\mathbf{E}_{t+1}^{(Q)}\|^2 + \frac{1}{2\alpha}\|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\bar{\mathbf{X}}_t^*) + \nabla F(\bar{\mathbf{X}}_t^*)\|^2 \\
&(\bar{\mathbf{X}}_t^* \text{ is the projection of } \bar{\mathbf{X}}_t \text{ on f } \mathcal{X}^* \text{ – see Lemma C.1.2}) \\
&\leq \frac{\alpha}{2}\|\mathbf{E}_{t+1}^{(Q)}\|^2 + \frac{1}{\alpha}\|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\bar{\mathbf{X}}_t^*)\|^2 + \frac{1}{\alpha}\Delta^2 \\
&(\text{recall } \Delta^2 := \max_{\mathbf{x}^* \in \mathcal{X}^*} \sum_{i=1}^{n} \|\nabla f_i(\mathbf{x}^*)\|_2^2) \\
&\leq \frac{\alpha}{2}\|\mathbf{E}_{t+1}^{(Q)}\|^2 + \frac{L^2}{2\alpha\mu}[F(\bar{\mathbf{X}}_t) - f^*] + \frac{1}{\alpha}\Delta^2,
\end{aligned}
\tag{C.37}
$$

In the last step, we use the result of Lemma C.1.2 by noting the fact that F on $\mathcal{L}$, similar to f, satisfies the PL condition with parameter μ.

Putting the bounds on the cross terms in (C.34) together, we obtain

$$
\begin{aligned}
F(\mathbf{X}_{t+1}^{(Q)}) \leq F(\bar{\mathbf{X}}_t) - \frac{1}{2\hat{L}}\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2 + \frac{\hat{L}+\alpha}{2}\|\mathbf{E}_{t+1}^{(Q)}\|^2 \\
+ \frac{L^2}{2\alpha\mu}[F(\bar{\mathbf{X}}_t) - f^*] + \frac{1}{\alpha}\Delta^2.
\end{aligned}
\tag{C.38}
$$

Subtracting f^* from both sides, using the PL condition of f along the fact that $\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2 = \|\nabla f(\bar{\mathbf{x}}_t)\|_2^2/n$ and $f(\bar{\mathbf{x}}_t) = F(\bar{\mathbf{X}}_t)$, and taking the

expectation yields

$$\begin{aligned}\mathbb{E}_{\mathcal{C}}[F(\mathbf{X}_{t+1}^{(Q)})] - f^* \leq (1 - \frac{\mu}{n\hat{L}} + \frac{L^2}{2\alpha\mu}) \quad \big(F(\bar{\mathbf{X}}_t) - f^*\big) \\ + \frac{\hat{L}+\alpha}{2}\mathbb{E}_{\mathcal{C}}\|\mathbf{E}_{t+1}^{(Q)}\|^2 + \frac{1}{\alpha}\Delta^2.\end{aligned} \tag{C.39}$$

Recall from Lemma C.1.1 and C.1.4 that with the specific choice of stepsize $\eta = 1/\hat{L}$ and the fact that $\hat{L} \geq L$ we have

$$F(\bar{\mathbf{X}}_t) - f^* \leq [F(\bar{\mathbf{X}}_0) - f^*]\rho^t \tag{C.40}$$

and

$$\begin{aligned}\mathbb{E}_{\mathcal{C}}\|\mathbf{E}_t^{(Q)}\|^2 \leq e_t^2 \leq \frac{13\eta^2 L^2}{2\mu\zeta}(1 - \frac{\delta\gamma}{2})^Q[F(\bar{\mathbf{X}}_0) - f^*]\rho^t \\ + \|\mathbf{X}_0^{(Q)}\|^2\rho^t + \frac{20\eta^2\Delta^2}{1-\xi}(1 - \frac{\delta\gamma}{2})^Q,\end{aligned} \tag{C.41}$$

where

$$\zeta := \rho - \xi > 0, \quad \rho := 1 - 2\frac{\mu}{n\hat{L}}(1 - \frac{L}{2\hat{L}}) < 1 - \frac{\mu}{n\hat{L}}, \quad \xi := 23(1 - \frac{\delta\gamma}{2})^Q < \rho. \tag{C.42}$$

To state a simple and clear result, we make a simplifying assumption that $\alpha \geq \hat{L}$. Indeed, in our regime of interest where $(1 - \frac{\delta\gamma}{2})^Q$ is small, if

$$\alpha = \frac{\hat{L}\sqrt{1-\xi}}{(1 - \frac{\delta\gamma}{2})^{\frac{Q}{2}}}, \tag{C.43}$$

following simple algebra we can show this assumption holds if Q satisfies

$$24(1 - \frac{\delta\gamma}{2})^Q < 1. \tag{C.44}$$

Therefore,

$$\frac{\hat{L}+\alpha}{2} \leq \alpha, \quad (1 - \frac{\mu}{n\hat{L}} + \frac{L^2}{2\alpha\mu}) \leq 1 - \frac{\mu}{n\hat{L}} + \frac{L}{2\mu}. \tag{C.45}$$

Thus, it holds that

$$\begin{aligned}\mathbb{E}_{\mathcal{C}}[F(\mathbf{X}_t^{(Q)})] - f^* \leq & \frac{21\Delta^2}{\hat{L}(1-\xi)}(1-\frac{\delta\gamma}{2})^{\frac{Q}{2}} \\ & + \rho^t \left[\frac{1-\frac{\mu}{n\hat{L}}+\frac{L}{2\mu}}{\rho} + \frac{13L\sqrt{1-\xi}}{2\mu\zeta}\left(1-\frac{\delta\gamma}{2}\right)^{\frac{Q}{2}}\right]\left[F(\bar{\mathbf{X}}_0) - f^*\right] \\ & + \rho^t \frac{\|\mathbf{X}_0^{(Q)}\|^2 \hat{L}\sqrt{1-\xi}}{\left(1-\frac{\delta\gamma}{2}\right)^{\frac{Q}{2}}}.\end{aligned} \tag{C.46}$$

That is, the above result establishes the linear convergence of the proposed scheme under smoothness and PLC.

C.2 Proof of Theorem 6.3.2

The proof follows a near identical perturbation analysis in Theorem 6.3.1. To formalize the arguments, we start by providing some intermediate lemmas.

C.2.1 Useful lemmas

The first Lemma establishes linear convergence of the "unperturbed sequence".

Lemma C.2.1. *Let* $\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{X}}_t - \eta\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))$ *denote a sequence such that* $\bar{\mathbf{X}}_0 \in \mathcal{L}$, $\bar{\mathbf{X}}_t = [\bar{\mathbf{x}}_t, \ldots, \bar{\mathbf{x}}_t]$, *and* $\eta \leq 2/(L+\mu)$. *Then*

$$\|\bar{\mathbf{X}}_t - \mathbf{X}^*\| \leq (1-\mu\eta)^t \|\bar{\mathbf{X}}_0 - \mathbf{X}^*\|. \tag{C.47}$$

Proof. Given the update of the average iterates it holds that

$$\begin{aligned}\|\bar{\mathbf{X}}_{t+1} - \mathbf{X}^*\|^2 &= \|\mathcal{P}_{\mathcal{L}}(\bar{\mathbf{X}}_t - \eta\nabla F(\bar{\mathbf{X}}_t)) - \mathcal{P}_{\mathcal{L}}(\mathbf{X}^* - \nabla F(\mathbf{X}^*))\|^2 \\ &\leq \|\bar{\mathbf{X}}_t - \eta\nabla F(\bar{\mathbf{X}}_t) - \mathbf{X}^* + \eta\nabla F(\mathbf{X}^*)\|^2 \\ &= \|\bar{\mathbf{X}}_t - \mathbf{X}^*\|^2 + \eta^2\|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*)\|^2 \\ &\quad - 2\eta\langle\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*), \bar{\mathbf{X}}_t - \mathbf{X}^*\rangle,\end{aligned} \tag{C.48}$$

where to obtain the inequality we use the non-expansiveness of projection (see Lemma 2.2.7 and Corollary 2.2.3 in [59]),

$$\|\mathcal{P}_{\mathcal{L}}(\mathbf{X}) - \mathcal{P}_{\mathcal{L}}(\mathbf{Y})\| \leq \|\mathbf{X} - \mathbf{Y}\|. \tag{C.49}$$

Now, we use Theorem 2.1.11 in [59], i.e.,

$$\langle\nabla F(\mathbf{Z}) - \nabla F(\mathbf{Y}), \mathbf{Z} - \mathbf{Y}\rangle \geq \frac{\mu L}{\mu + L}\|\mathbf{Z} - \mathbf{Y}\|^2 + \frac{1}{\mu + L}\|F(\mathbf{Z}) - \nabla F(\mathbf{Y})\|^2, \tag{C.50}$$

for $\mathbf{Z} = \bar{\mathbf{X}}_t$ and $\mathbf{Y} = \mathbf{X}^*$ to bound the inner-product on the RHS of (C.48),

$$\begin{aligned}\|\bar{\mathbf{X}}_{t+1} - \mathbf{X}^*\|^2 &\leq (1 - 2\eta\frac{\mu L}{\mu + L})\|\bar{\mathbf{X}}_t - \mathbf{X}^*\|^2 \\ &\quad + (\eta^2 - \frac{2\eta}{\mu + L})\|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*)\|^2 \\ &\leq (1 - 2\eta\frac{\mu L}{\mu + L} + \eta^2\mu^2 - 2\eta\frac{\mu^2}{\mu + L})\|\bar{\mathbf{X}}_t - \mathbf{X}^*\|^2 \\ &= (1 - \eta\mu)^2\|\bar{\mathbf{X}}_t - \mathbf{X}^*\|^2,\end{aligned} \tag{C.51}$$

where to obtain the inequality we use the fact that $\eta \leq 2/(L + \mu)$ as well as the strong convexity of $F(\cdot)$ – in particular, the inequality

$$\|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*)\| \geq \mu\|\bar{\mathbf{X}}_t - \mathbf{X}^*\|. \tag{C.52}$$

Finally, recursively applying the result of (C.51) establishes the stated expression. ■

The next Lemma establishes a bound on the amount of perturbation of the DeLi-CoCo's iterates $\mathbf{X}_t^{(Q)}$ compared to $\bar{\mathbf{X}}_t$.

Lemma C.2.2. *Let $\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{X}}_t - \eta\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))$ and let $\{\mathbf{X}_t^{(Q)}\}$ denote the sequence generated by DeLi-CoCo. Let $e_t^2 = \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}_t\|^2 + \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{Z}_t^{(Q)}\|^2$. For any $0 < \eta \leq 2/(L+\mu)$, let Q be such that*

$$\zeta := \rho - \xi > 0, \quad \rho := (1-\eta\mu)^2 < 1, \quad \xi := (1-\frac{\delta\gamma}{2})^Q(3+20\eta^2\hat{L}^2) < 1. \tag{C.53}$$

Then, it holds that

$$e_t^2 \leq \frac{13L\eta^2}{\zeta}(1-\frac{\delta\gamma}{2})^Q\|\mathbf{X}_0^{(Q)} - \mathbf{X}^*\|^2\rho^t + \rho^t\|\mathbf{X}_0^{(Q)}\|^2 + \frac{20\eta^2\Delta^2}{1-\xi}(1-\frac{\delta\gamma}{2})^Q. \tag{C.54}$$

Additionally, if all nodes initialize such that $\mathbf{X}_0^Q = 0$, by considering the dominant terms in above we obtain

$$e_t^2 = \mathcal{O}\left(\frac{L\eta^2}{\zeta}(1-\frac{\delta\gamma}{2})^Q\|\bar{\mathbf{X}}_0 - \mathbf{X}^*\|^2\rho^t + \frac{\eta^2\Delta^2}{1-\xi}(1-\frac{\delta\gamma}{2})^Q\right), \tag{C.55}$$

where the $\mathcal{O}$ notation does not hide any term depending on Q or t.

Proof. It holds by definitions $\mathbf{X}_{t+1}^{(0)} = \mathbf{X}_t^{(Q)} - \eta\nabla F(\mathbf{X}_t^{(Q)})$ and $\mathbf{Z}_{t+1}^{(0)} = \mathbf{Z}_t^{(Q)}$ that

$$\begin{aligned}
&\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(0)} - \bar{\mathbf{X}}_{t+1}\|^2 + \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(0)} - \mathbf{Z}_{t+1}^{(0)}\|^2 \\
&\quad= \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}_{t+1} - \bar{\mathbf{X}}_t + \bar{\mathbf{X}}_t - \eta\nabla F(\mathbf{X}_t^{(Q)})\|^2 \\
&\qquad+ \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{Z}_t^{(Q)} - \eta\nabla F(\mathbf{X}_t^{(Q)})\|^2 \\
&\quad\leq 3e_t^2 + 3\eta^2\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))\|^2 + 5\eta^2\mathbb{E}_{\mathcal{C}}\|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\bar{\mathbf{X}}_t) + \nabla F(\bar{\mathbf{X}}_t)\|^2,
\end{aligned} \tag{C.56}$$

where we used the update rule of the average iterates $\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{X}}_t - \eta\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t))$. Now, we proceed by using the smoothness property of $F(\cdot)$, the non-expansiveness

property of projection (cf. (C.2)) as well as the fact that $\mathcal{P}_{\mathcal{L}}(\nabla F(\mathbf{X}^*)) = \mathbf{0}$ to obtain

$$\begin{aligned}
&\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(0)} - \bar{\mathbf{X}}_{t+1}\|^2 + \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(0)} - \mathbf{Z}_{t+1}^{(0)}\|^2 \\
&\quad \leq 3e_t^2 + 3\eta^2\|\mathcal{P}_{\mathcal{L}}(\nabla F(\bar{\mathbf{X}}_t)) - \mathcal{P}_{\mathcal{L}}(\nabla F(\mathbf{X}^*))\|^2 \\
&\qquad + 5\eta^2\mathbb{E}_{\mathcal{C}}\|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\bar{\mathbf{X}}_t) + \nabla F(\bar{\mathbf{X}}_t)\|^2 \\
&\quad \leq 3e_t^2 + 13\eta^2\|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*)\|^2 \\
&\qquad + 10\eta^2\mathbb{E}_{\mathcal{C}}\|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\bar{\mathbf{X}}_t) + \nabla F(\mathbf{X}^*)\|^2 \\
&\quad \leq 3e_t^2 + 13\eta^2\|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*)\|^2 \\
&\qquad + 20\hat{L}^2\eta^2\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}_t\|^2 + 20\eta^2\|\nabla F(\mathbf{X}^*)\|^2 \\
&\quad \leq 3e_t^2 + 13\eta^2\|\nabla F(\bar{\mathbf{X}}_t) - \nabla F(\mathbf{X}^*)\|^2 \\
&\qquad + 20\hat{L}^2\eta^2(\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}_t\|^2 + \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{Z}_t^{(Q)}\|^2) + 20\eta^2\|\nabla F(\mathbf{X}^*)\|^2 \\
&\quad \leq (3 + 20\eta^2\hat{L}^2)e_t^2 + 13\eta^2L^2\|\bar{\mathbf{X}}_t - \mathbf{X}^*\|^2 + 20\eta^2\|\nabla F(\mathbf{X}^*)\|^2,
\end{aligned} \tag{C.57}$$

where we used the fact that $\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{Z}_t^{(Q)}\|^2 \geq 0$ and exploited smoothness of $F(\cdot)$. Thus, by Lemma C.1.3 and the result of (C.57) we obtain the recursion

$$e_{t+1}^2 \leq \xi e_t^2 + \nu_t + u, \tag{C.58}$$

where

$$\xi := (1 - \frac{\delta\gamma}{2})^Q(3 + 20\eta^2\hat{L}^2), \tag{C.59}$$

$$\nu_t := 13L\eta^2(1 - \frac{\delta\gamma}{2})^Q\|\bar{\mathbf{X}}_t - \mathbf{X}^*\|^2, \tag{C.60}$$

and

$$u := 20(1 - \frac{\delta\gamma}{2})^Q\eta^2\|\nabla F(\mathbf{X}^*)\|^2 = 20(1 - \frac{\delta\gamma}{2})^Q\eta^2\Delta^2. \tag{C.61}$$

Note that by Lemma C.2.1, ν_t is a linearly converging sequence, i.e. $\nu_t \leq \rho^t \nu_0$ where

$$\rho := (1 - \eta\mu)^2, \qquad \nu_0 := 13L\eta^2(1 - \frac{\delta\gamma}{2})^Q \|\bar{\mathbf{X}}_0 - \mathbf{X}^*\|^2. \tag{C.62}$$

Since for a large enough Q and small enough η (see the conditions in the statement of Theorem **??**) $a < 1$, given the fact that ν_t vanishes linearly, we expect e_t^2 to converge linearly. In the following we prove this statement using induction.

Define h_t such that $h_0 = e_0^2$, and $h_{t+1} = \xi h_t + \nu_t$. Using simple algebra it follows that

$$h_t = \xi^t e_0^2 + \sum_{i=0}^{t-1} \xi^{t-i-1} \rho^i \nu_0, \tag{C.63}$$

Similarly, we can expand the recursion of e_t^2 to obtain

$$e_t^2 \leq \xi^t e_0^2 + \sum_{i=0}^{t-1} \xi^{t-i-1} \rho^i \nu_0 + u \sum_{i=0}^{t-1} \xi^i \leq h_t + \frac{u}{1-\xi}. \tag{C.64}$$

Note that if h_t linearly converges to zero, hence, e_t^2 linearly converges as well. Since $\rho - \xi = \zeta > 0$ by assumption, using simple algebra we can show

$$h_t \leq \rho^t (e_0^2 + \frac{\nu_0}{\zeta}). \tag{C.65}$$

Thus,

$$e_t^2 \leq \rho^t (e_0^2 + \frac{\nu_0}{\zeta}) + \frac{u}{1-\xi}, \tag{C.66}$$

and the proof is complete by noting the definitions of ξ, ρ, ν_0, and u. ■

C.2.2 Proof of the main theorem

Let $\mathbf{X}_t^{(q)} = \bar{\mathbf{X}}_t + \mathbf{E}_t^{(q)}$ for some (random) error matrix $\mathbf{E}_t^{(q)} \in \mathbb{R}^{d\times n}$. Recall $\mathcal{P}_{\mathcal{L}}(.)$ denotes the projection operator onto the linear subspace (denoted by $\mathcal{L}$) of $d \times n$ matrices with identical columns. Noting the fact that $\mathbf{X}^* = \mathcal{P}_{\mathcal{L}}(\bar{\mathbf{X}}^* - \eta\nabla F(\bar{\mathbf{X}}^*))$ and $\bar{\mathbf{X}}_{t+1} = \mathcal{P}_{\mathcal{L}}(\bar{\mathbf{X}}_t - \eta\nabla F(\bar{\mathbf{X}}_t))$, and using the update rule of DeLi-CoCo we have

$$\begin{aligned}
\|\mathbf{X}_{t+1}^{(Q)} - \mathbf{X}^*\|^2 &= \|\mathbf{E}_{t+1}^{(Q)}\|^2 + \|\mathcal{P}_{\mathcal{L}}(\mathbf{X}_t^{(Q)} - \eta\nabla F(\mathbf{X}_t^{(Q)})) - \mathcal{P}_{\mathcal{L}}(\mathbf{X}^* - \eta\nabla F(\mathbf{X}^*))\|^2 \\
&\quad + 2\langle \mathbf{E}_{t+1}^{(Q)}, \mathcal{P}_{\mathcal{L}}(\mathbf{X}_t^{(Q)} - \eta\nabla F(\mathbf{X}_t^{(Q)})) - \mathcal{P}_{\mathcal{L}}(\mathbf{X}^*)\rangle \\
&\leq \|\mathbf{E}_{t+1}^{(Q)}\|^2 + 2\langle \mathbf{E}_{t+1}^{(Q)}, \mathcal{P}_{\mathcal{L}}(\mathbf{X}_t^{(Q)} - \eta\nabla F(\mathbf{X}_t^{(Q)})) \\
&\quad - \mathcal{P}_{\mathcal{L}}(\mathbf{X}^* - \eta\nabla F(\mathbf{X}^*))\rangle + \|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}^* - \eta\nabla F(\mathbf{X}_t^{(Q)}) + \eta\nabla F(\mathbf{X}^*)\|^2,
\end{aligned} \tag{C.67}$$

where to obtain the inequality we used the non-expansiveness property of projection (cf. (C.2)). Next, we aim to bound each of the individual terms above. First, we know from the definition of e_t that $\mathbb{E}_{\mathcal{C}}\|\mathbf{E}_{t+1}^{(Q)}\| \leq e_{t+1}$. Secondly, by using the Cauchy-Schwarz inequality, the cross term can be entangled and dealt with according to

$$\begin{aligned}
&\mathbb{E}_{\mathcal{C}}[\langle \mathbf{E}_{t+1}^{(Q)}, \mathcal{P}_{\mathcal{L}}(\mathbf{X}_t^{(Q)} - \eta\nabla F(\mathbf{X}_t^{(Q)})) - \mathcal{P}_{\mathcal{L}}(\mathbf{X}^* - \eta\nabla F(\mathbf{X}^*))\rangle] \\
&\qquad \leq \mathbb{E}_{\mathcal{C}}\|\mathbf{E}_{t+1}^{(Q)}\|\|\mathcal{P}_{\mathcal{L}}(\mathbf{X}_t^{(Q)} - \eta\nabla F(\mathbf{X}_t^{(Q)})) - \mathcal{P}_{\mathcal{L}}(\mathbf{X}^* - \eta\nabla F(\mathbf{X}^*))\| \\
&\qquad \leq \mathbb{E}_{\mathcal{C}}\|\mathbf{E}_{t+1}^{(Q)}\|\|\mathbf{X}_t^{(Q)} - \bar{\mathbf{X}}^* - \eta\nabla F(\mathbf{X}_t^{(Q)}) + \eta\nabla F(\mathbf{X}^*)\|,
\end{aligned} \tag{C.68}$$

by using the non-expansiveness property of projection (cf. (C.2)). Thus, by taking the expectation of both sides in (C.67) and using the above arguments

we obtain

$$\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(Q)} - \mathbf{X}^*\|^2 \leq e_{t+1}^2 + 2e_{t+1}\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{X}^* - \eta\nabla F(\mathbf{X}_t^{(Q)}) + \eta\nabla F(\mathbf{X}^*)\| + \mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{X}^* - \eta\nabla F(\mathbf{X}_t^{(Q)}) + \eta\nabla F(\mathbf{X}^*)\|^2. \tag{C.69}$$

Now we bound the last term on the RHS of the above expression. Using Theorem 2.1.11 in [59], i.e.

$$\langle \nabla F(Z) - \nabla F(Y), Z - Y\rangle \geq \frac{\hat{\mu}\hat{L}}{\hat{\mu}+\hat{L}}\|Z-Y\|^2 + \frac{1}{\hat{\mu}+\hat{L}}\|F(Z) - \nabla F(Y)\|^2, \tag{C.70}$$

for $Z = \mathbf{X}_t^{(Q)}$ and $Y = \mathbf{X}^*$ we have

$$\begin{aligned}
&\|\mathbf{X}_t^{(Q)} - \mathbf{X}^* - \eta\nabla F(\mathbf{X}_t^{(Q)}) + \eta\nabla F(\mathbf{X}^*)\|^2 \leq \|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|^2 \\
&\quad + \eta^2\|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\mathbf{X}^*)\|^2 \\
&- 2\frac{\hat{\mu}\hat{L}}{\hat{\mu}+\hat{L}}\eta\|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|^2 - \frac{2\eta}{\hat{\mu}+\hat{L}}\|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\mathbf{X}^*)\|^2 \\
&= \|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|^2\left(1 - 2\frac{\hat{\mu}\hat{L}}{\hat{\mu}+\hat{L}}\eta\right) + \|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\mathbf{X}^*)\|^2\left(\eta - \frac{2}{\hat{\mu}+\hat{L}}\right)\eta \\
&\leq \|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|^2\left(1 - 2\frac{\hat{\mu}\hat{L}}{\hat{\mu}+\hat{L}}\eta + \eta(\eta - \frac{2}{\hat{\mu}+\hat{L}})\hat{\mu}^2\right) \\
&= \|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|^2(1-\eta\hat{\mu})^2 := \|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|^2\ell^2,
\end{aligned} \tag{C.71}$$

for any $\eta \leq 2/(\hat{L}+\hat{\mu})$. The strong convexity of $F(\cdot)$ implies

$$\|\nabla F(\mathbf{X}_t^{(Q)}) - \nabla F(\mathbf{X}^*)\| \geq \hat{\mu}\|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|. \tag{C.72}$$

Thus, we can now put together a bound on the error term according to

$$\begin{aligned}
\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(Q)} - \mathbf{X}^*\|^2 &\leq \ell^2\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|^2 + e_{t+1}^2 + 2\ell e_{t+1}\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\| \\
&= (e_{t+1} + \ell\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|)^2.
\end{aligned} \tag{C.73}$$

Therefore,

$$\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(Q)} - \mathbf{X}^*\| \leq e_{t+1} + \ell\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|. \tag{C.74}$$

Recall from Lemma C.2.2 that the sequence e_t^2 converges linearly, i.e.,

$$e_t^2 \leq \frac{13L\eta^2}{\zeta}(1-\frac{\delta\gamma}{2})^Q\|\mathbf{X}_0^{(Q)} - \mathbf{X}^*\|^2\rho^t + \rho^t\|\mathbf{X}_0^{(Q)}\|^2 + \frac{20\eta^2\Delta^2}{1-\xi}(1-\frac{\delta\gamma}{2})^Q, \tag{C.75}$$

where

$$\zeta := \rho - \xi > 0, \quad \rho := (1-\eta\mu)^2 < 1, \quad \xi := (1-\frac{\delta\gamma}{2})^Q(3+20\eta^2\hat{L}^2) < 1. \tag{C.76}$$

Note that for any η we have $\ell \geq \rho$ and we can upper bound the bound on e_t^2 by replacing ρ with ℓ. Using a similar technique as the one we used towards the end of the proof of Lemma C.2.2, we can show for any two sequence $h_t^{(1)}$ and $h_t^{(2)} := \ell^t h_0^{(2)}$ that satisfy

$$h_t^{(1)} \leq \ell h_t^{(1)} + h_t^{(2)} + u^{(1)}, \tag{C.77}$$

it holds that

$$h_t^{(1)} \leq \ell^t(h_0^{(1)} + t\frac{h_0^{(2)}}{\ell}) + \frac{u^{(1)}}{1-\ell}. \tag{C.78}$$

Thus, replacing $h_t^{(1)}$ and $h_t^{(2)}$ with $\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|^2$ and e_t^2 we obtain that $\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_{t+1}^{(Q)} - \mathbf{X}^*\|$ converges according to

$$\begin{aligned}\mathbb{E}_{\mathcal{C}}\|\mathbf{X}_t^{(Q)} - \mathbf{X}^*\|^2 = t\ell^{t-1}&\left(\frac{13L\eta^2}{\ell\zeta}\left(1-\frac{\delta\gamma}{2}\right)^Q\|\mathbf{X}_0 - \mathbf{X}^*\|^2 + \|\mathbf{X}_0\|^2\right)\\ &+ \|\mathbf{X}_0 - \mathbf{X}^*\|^2\ell^t\\ &+ \frac{20\eta^2}{1-\ell}\frac{\Delta^2}{1-\xi}\left(1-\frac{\delta\gamma}{2}\right)^Q,\end{aligned} \tag{C.79}$$

which is the stated result.

Bibliography

[1] E. L. Dyer, A. C. Sankaranarayanan, and R. G. Baraniuk, "Greedy feature selection for subspace clustering," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2487–2517, Sep. 2013.

[2] C. You, D. Robinson, and R. Vidal, "Scalable sparse subspace clustering by orthogonal matching pursuit," in *Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016, pp. 3918–3927.

[3] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009, pp. 2790–2797.

[4] ——, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

[5] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.

[6] R. Tron and R. Vidal, "A benchmark for the comparison of 3-D motion segmentation algorithms," in *Conf. Computer Vision and Pattern*

Recognition (CVPR). IEEE, 2007, pp. 1–8.

[7] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM journal on computing*, vol. 24, no. 2, pp. 227–234, Apr. 1995.

[8] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[9] M. Elad, M. A. Figueiredo, and Y. Ma, "On the role of sparse and redundant representations in image processing," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 972–982, June 2010.

[10] J. A. Tropp, "Column subset selection, matrix factorization, and eigenvalue optimization," in *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 2009, pp. 978–986.

[11] D. Du, F. K. Hwang, and F. Hwang, *Combinatorial group testing and its applications*. World Scientific, 2000, vol. 12.

[12] A. Y. Yang, J. Wright, Y. Ma, and S. S. Sastry, "Unsupervised segmentation of natural images via lossy data compression," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 212–225, May 2008.

[13] R. Vidal, R. Tron, and R. Hartley, "Multiframe motion segmentation with missing data using PowerFactorization and GPCA," *Int. Journal of Computer Vision*, vol. 79, no. 1, pp. 85–105, Aug. 2008.

[14] J. Ho, M.-H. Yang, J. Lim, K.-C. Lee, and D. Kriegman, "Clustering appearances of objects under varying illumination conditions," in *Conf. Computer Vision and Pattern Recognition (CVPR)*, vol. 1. IEEE, 2003, pp. I–I.

[15] W. Hong, J. Wright, K. Huang, and Y. Ma, "Multiscale hybrid linear models for lossy image representation," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3655–3671, Dec. 2006.

[16] H. Xu, C. Caramanis, and S. Sanghavi, "Robust PCA via outlier pursuit," in *Advances in Neural Information Processing Systems (NIPS)*, 2010, pp. 2496–2504.

[17] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, p. 11, May 2011.

[18] B. Yang, "Projection approximation subspace tracking," *IEEE Trans. Signal Process.*, vol. 43, no. 1, pp. 95–107, Jan. 1995.

[19] M. Rahmani and G. K. Atia, "High dimensional low rank plus sparse matrix decomposition," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 2004–2019, Apr. 2017.

[20] ——, "Randomized robust subspace recovery and outlier detection for high dimensional data matrices," *IEEE Trans. Signal Process.*, vol. 65, no. 6, pp. 1580–1594, Mar. 2017.

[21] A. Nordio, A. Tarable, F. Dabbene, and R. Tempo, "Sensor selection and precoding strategies for wireless sensor networks," *IEEE trans. on signal process.*, vol. 63, no. 16, pp. 4411–4421, 2015.

[22] M. Shamaiah, S. Banerjee, and H. Vikalo, "Greedy sensor selection: Leveraging submodularity," in *Conf. on Decision and Control*. IEEE, 2010, pp. 2572–2577.

[23] V. Tzoumas, A. Jadbabaie, and G. J. Pappas, "Sensor placement for optimal kalman filtering: Fundamental limits, submodularity, and algorithms," in *Proceedings of the American Control Conference (ACC)*. IEEE, Jun. 2016, pp. 191–196.

[24] V. Tzoumas, N. A. Atanasov, A. Jadbabaie, and G. J. Pappas, "Scheduling nonlinear sensors for stochastic process estimation," *arXiv preprint arXiv:1609.08536*, 2016.

[25] V. Tzoumas, A. Jadbabaie, and G. J. Pappas, "Near-optimal sensor scheduling for batch state estimation: Complexity, algorithms, and limits," in *Proceedings of Conference on Decision and Control (CDC)*. IEEE, Dec. 2016, pp. 2695–2702.

[26] T. H. Summers, F. L. Cortesi, and J. Lygeros, "On submodularity and controllability in complex dynamical networks," *IEEE Trans. Control Netw. Syst.*, vol. 3, no. 1, pp. 91–101, 2016.

[27] V. Tzoumas, M. A. Rahimian, G. J. Pappas, and A. Jadbabaie, "Minimal actuator placement with optimal control constraints," in *Proceedings of American Control Conference (ACC)*. IEEE, Jul. 2015, pp. 2081–2086.

[28] M. G. Damavandi, V. Krishnamurthy, and J. R. Martí, "Robust meter placement for state estimation in active distribution systems," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1972–1982, 2015.

[29] N. Gensollen, V. Gauthier, M. Marot, and M. Becker, "Submodular optimization for control of prosumer networks," in *Int. Conf. Smart Grid Communications*. IEEE, 2016, pp. 180–185.

[30] Z. Liu, A. Clark, P. Lee, L. Bushnell, D. Kirschen, and R. Poovendran, "Towards scalable voltage control in smart grid: a submodular optimization approach," in *Int. Conf. Cyber-Physical Systems*. IEEE Press, 2016, p. 20.

[31] M. Shamaiah, S. Banerjee, and H. Vikalo, "Greedy sensor selection under channel uncertainty," *IEEE Wireless Commun. Lett.*, vol. 1, no. 4, pp. 376–379, 2012.

[32] B. Mirzasoleiman, A. Badanidiyuru, A. Karbasi, J. Vondrak, and A. Krause, "Lazier than lazy greedy," in *Conf. Artificial Intelligence*. AAAI, 2015.

[33] M. Ghasemi and U. Topcu, "Online active perception for partially observable markov decision processes with limited budget," in *2019 IEEE*

58th Conference on Decision and Control (CDC). IEEE, 2019, pp. 6169–6174.

[34] ——, "Perception-aware point-based value iteration for partially observable markov decision processes," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. AAAI Press, 2019, pp. 2371–2377.

[35] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*, 2017, pp. 1273–1282.

[36] K. Seaman, F. Bach, S. Bubeck, Y. T. Lee, and L. Massoulié, "Optimal algorithms for smooth and strongly convex distributed optimization in networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3027–3036.

[37] M. Assran, N. Loizou, N. Ballas, and M. Rabbat, "Stochastic gradient push for distributed deep learning," *arXiv preprint arXiv:1811.10792*, 2018.

[38] L. He, A. Bian, and M. Jaggi, "CoLa: Decentralized linear learning," in *Advances in Neural Information Processing Systems*, 2018, pp. 4536–4546.

[39] A. Koloskova, S. Stich, and M. Jaggi, "Decentralized stochastic optimization and gossip algorithms with compressed communication," in *Inter-*

national Conference on Machine Learning, 2019, pp. 3478–3487.

[40] S. Chen, S. A. Billings, and W. Luo, "Orthogonal least squares methods and their application to non-linear system identification," *Int. Journal of Control*, vol. 50, no. 5, pp. 1873–1896, Nov. 1989.

[41] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Asilomar Conf. Signals, Syst. and Computers*. IEEE, 1993, pp. 40–44.

[42] A. Hashemi and H. Vikalo, "Accelerated orthogonal least-squares for large-scale sparse reconstruction," *Digital Signal Process.*, vol. 82, pp. 91–105, Nov. 2018.

[43] L. Chamon and A. Ribeiro, "Approximate supermodularity bounds for experimental design," in *Advances in Neural Information Processing Systems*, Dec. 2017, pp. 5409–5418.

[44] M. Sviridenko, J. Vondrák, and J. Ward, "Optimal approximation for submodular and supermodular optimization with bounded curvature," *Mathematics of Operations Research*, vol. 42, no. 4, pp. 1197–1218, 2017.

[45] A. Das and D. Kempe, "Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection," in *Int. Conf. Machine Learning*, 2011, pp. 1057–1064.

[46] E. R. Elenberg, R. Khanna, A. G. Dimakis, S. Negahban *et al.*, "Restricted strong convexity implies weak submodularity," *The Annals of Statistics*, vol. 46, no. 6B, pp. 3539–3568, 2018.

[47] T. Horel and Y. Singer, "Maximization of approximately submodular functions," in *Proc. Advances in Neural Information Processing Systems*, 2016, pp. 3045–3053.

[48] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions I," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[49] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *Proc. national conference on Artificial intelligence*, vol. 7. AAAI Press, 2007, pp. 1650–1654.

[50] A. Hashemi, M. Ghasemi, H. Vikalo, and U. Topcu, "Randomized greedy sensor selection: Leveraging weak submodularity," *IEEE Transactions on Automatic Control*, 2020.

[51] D. P. Williamson and D. B. Shmoys, *The design of approximation algorithms*. Cambridge university press, 2011.

[52] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.

[53] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," *SIAM Journal on Optimization*, vol. 26, no. 3, pp. 1835–1854, 2016.

[54] W. Shi, Q. Ling, G. Wu, and W. Yin, "EXTRA: An exact first-order algorithm for decentralized consensus optimization," *SIAM Journal on Optimization*, vol. 25, no. 2, pp. 944–966, 2015.

[55] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems & Control Letters*, vol. 53, no. 1, pp. 65–78, 2004.

[56] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on automatic control*, vol. 48, no. 6, pp. 988–1001, 2003.

[57] B. T. Polyak, "Gradient methods for minimizing functionals," *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, vol. 3, no. 4, pp. 643–653, 1963.

[58] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the Polyak-Lojasiewicz condition," in *European Conference on Machine Learning and Knowledge Discovery in Databases-Volume 9851*, 2016, pp. 795–811.

[59] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2013, vol. 87.

[60] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, "Sparsified SGD with memory," in *Advances in Neural Information Processing Systems*, 2018, pp. 4447–4458.

[61] S. P. Karimireddy, Q. Rebjock, S. U. Stich, and M. Jaggi, "Error feedback fixes SignSGD and other gradient compression schemes," *arXiv preprint arXiv:1901.09847*, 2019.

[62] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-efficient SGD via gradient quantization and encoding," in *Advances in Neural Information Processing Systems*, 2017, pp. 1709–1720.

[63] A. Hashemi, H. Vikalo, and G. de Veciana, "Progressive stochastic greedy sparse reconstruction and support selection," *arXiv*, pp. arXiv–1907, 2019.

[64] A. Hashemi and H. Vikalo, "Sparse linear regression via generalized orthogonal least-squares," in *Global Conf. Signal and Information Processing (GlobalSIP)*. IEEE, Dec. 2016, pp. 1305–1309.

[65] ——, "Recovery of sparse signals via branch and bound least-squares," in *IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4760–4764.

[66] A. Hashemi, R. Shafipour, H. Vikalo, and G. Mateos, "A novel scheme for support identification and iterative sampling of bandlimited graph

signals," in *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2018, pp. 778–782.

[67] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[68] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[69] E. J. Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9, pp. 589–592, 2008.

[70] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, Jan. 1996.

[71] S. Consul, A. Hashemi, and H. Vikalo, "A map framework for support recovery of sparse signals using orthogonal least squares," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5127–5131.

[72] T. Zhang, "Sparse recovery with orthogonal matching pursuit under RIP," *IEEE Transactions on Information Theory*, vol. 57, no. 9, pp.

6215–6221, Sep. 2011.

[73] M. A. Davenport and M. B. Wakin, "Analysis of orthogonal matching pursuit using the restricted isometry property," *IEEE Transactions on Information Theory*, vol. 56, no. 9, pp. 4395–4401, Sep. 2010.

[74] Q. Mo and Y. Shen, "A remark on the restricted isometry property in orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3654–3656, June. 2012.

[75] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.

[76] T. T. Cai and L. Wang, "Orthogonal matching pursuit for sparse signal recovery with noise," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4680–4688, July 2011.

[77] T. Zhang, "On the consistency of feature selection using greedy least squares regression," *Journal of Machine Learning Research*, vol. 10, pp. 555–568, Mar. 2009.

[78] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[79] A. K. Fletcher and S. Rangan, "Orthogonal matching pursuit: A Brownian motion analysis," *IEEE Transactions on Signal Processing*, vol. 60, no. 3, pp. 1010–1021, Mar. 2012.

[80] S. Rangan and A. K. Fletcher, "Orthogonal matching pursuit from noisy random measurements: A new analysis," in *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, 2009, pp. 540–548.

[81] S. Foucart, "Stability and robustness of weak orthogonal matching pursuits," in *Recent advances in harmonic analysis and applications*. Springer, 2012, pp. 395–405.

[82] J. Wang, S. Kwon, P. Li, and B. Shim, "Recovery of sparse signals via generalized orthogonal matching pursuit: A new analysis," *IEEE Transactions on Signal Processing*, vol. 64, no. 4, pp. 1076–1089, Feb. 2016.

[83] C. Soussen, R. Gribonval, J. Idier, and C. Herzet, "Joint k-step analysis of orthogonal matching pursuit and orthogonal least squares," *IEEE Trans. Inf. Theory*, vol. 59, no. 5, pp. 3158–3174, May 2013.

[84] C. Herzet, A. Drémeau, and C. Soussen, "Relaxed recovery conditions for OMP/OLS by exploiting both coherence and decay," *IEEE Trans. Inf. Theory*, vol. 62, no. 1, pp. 459–470, Jan. 2016.

[85] C. Herzet, C. Soussen, J. Idier, and R. Gribonval, "Exact recovery conditions for sparse representations with partial support information," *IEEE*

Transactions on Information Theory, vol. 59, no. 11, pp. 7509–7524, Nov. 2013.

[86] J. Wang and P. Li, "Recovery of sparse signals using multiple orthogonal least squares," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 2049–2062, Apr. 2017.

[87] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 1094–1121, Feb. 2012.

[88] J. Wang, S. Kwon, and B. Shim, "Generalized orthogonal matching pursuit," *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6202–6216, Dec. 2012.

[89] S. Kwon, J. Wang, and B. Shim, "Multipath matching pursuit," *IEEE Transactions on Information Theory*, vol. 60, no. 5, pp. 2986–3001, Mar. 2014.

[90] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, May 2009.

[91] M. Grant and S. Boyd, "CVX: MATLAB software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[92] ——, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control*, ser. Lecture Notes in Control and Information Sciences, V. Blondel, S. Boyd, and H. Kimura, Eds. Springer-Verlag Limited, 2008, pp. 95–110, http://stanford.edu/~boyd/graph_dcp.html.

[93] A. Y. Ng, M. I. Jordan, Y. Weiss *et al.*, "On spectral clustering: Analysis and an algorithm," in *the Advances in Neural Information Processing Systems (NIPS)*, vol. 14, no. 2, 2001, pp. 849–856.

[94] V. Guruswami and A. K. Sinop, "Optimal column-based low-rank matrix reconstruction," in *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*. SIAM, 2012, pp. 1207–1214.

[95] C. Boutsidis, P. Drineas, and M. Magdon-Ismail, "Near-optimal column-based matrix reconstruction," *SIAM Journal on Computing*, vol. 43, no. 2, pp. 687–717, 2014.

[96] A. K. Farahat, A. Elgohary, A. Ghodsi, and M. S. Kamel, "Greedy column subset selection for large-scale data sets," *Knowledge and Information Systems*, vol. 45, no. 1, pp. 1–34, 2015.

[97] A. Hashemi and H. Vikalo, "Evolutionary self-expressive models for subspace clustering," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1534–1546, 2018.

[98] ——, "Evolutionary subspace clustering: Discovering structure in self-expressive time-series data," in *IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, 2019.

[99] M. Soltanolkotabi and E. J. Candes, "A geometric analysis of subspace clustering with outliers," *The Annals of Statistics*, pp. 2195–2238, Aug. 2012.

[100] M. Soltanolkotabi, E. Elhamifar, and E. J. Candes, "Robust subspace clustering," *The Annals of Statistics*, vol. 42, no. 2, pp. 669–699, Apr. 2014.

[101] M. Tschannen and H. Bölcskei, "Noisy subspace clustering via matching pursuits," *arXiv preprint arXiv:1612.03450*, Dec. 2016.

[102] C.-G. Li, C. You, and R. Vidal, "Structured sparse subspace clustering: A joint affinity learning and subspace clustering framework," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2988–3001, Jun. 2017.

[103] E. Elhamifar, "High-rank matrix completion and clustering under self-expressive models," in *Advances in Neural Information Processing Systems (NIPS)*, 2016, pp. 73–81.

[104] C.-G. Li and R. Vidal, "A structured sparse plus structured low-rank framework for subspace clustering and completion," *IEEE Trans. Signal Process.*, vol. 64, no. 24, pp. 6557–6570, Dec. 2016.

[105] J. Fan and T. W. Chow, "Sparse subspace clustering for data with missing entries and high-rank matrix completion," *Neural Networks*, vol. 93, pp. 36–44, Sep. 2017.

[106] Z. Charles, A. Jalali, and R. Willett, "Subspace clustering with missing and corrupted data," *arXiv preprint arXiv:1707.02461*, Jan. 2018.

[107] M. C. Tsakiris and R. Vidal, "Theoretical analysis of sparse subspace clustering with missing entries," *arXiv preprint arXiv:1801.00393*, Feb. 2018.

[108] J. Shen, P. Li, and H. Xu, "Online low-rank subspace clustering by basis dictionary pursuit," in *Int. Conf. Machine Learning (ICML)*, 2016, pp. 622–631.

[109] S. Li, K. Li, and Y. Fu, "Temporal subspace clustering for human motion segmentation," in *Int. Conf. Computer Vision (ICCV)*. IEEE, 2015, pp. 4453–4461.

[110] D. Chakrabarti, R. Kumar, and A. Tomkins, "Evolutionary clustering," in *Int. Conf. Knowledge Discovery and Data Mining (SIGKDD)*. ACM, 2006, pp. 554–560.

[111] K. S. Xu, M. Kliger, and A. O. Hero III, "Adaptive evolutionary clustering," *Data Mining and Knowledge Discovery*, vol. 28, no. 2, pp. 304–336, Mar. 2014.

[112] F. Folino and C. Pizzuti, "An evolutionary multiobjective approach for community discovery in dynamic networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8, pp. 1838–1852, Aug. 2014.

[113] N. M. Arzeno and H. Vikalo, "Evolutionary affinity propagation," in *Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2681–2685.

[114] N. Czink, R. Tian, S. Wyne, F. Tufvesson, J.-P. Nuutinen, J. Ylitalo, E. Bonek, and A. F. Molisch, "Tracking time-variant cluster parameters in MIMO channel measurements," in *Int. Conf. Communications and Networking in China (CHINACOM)*. IEEE, 2007, pp. 1147–1151.

[115] S. Günnemann, H. Kremer, C. Laufkötter, and T. Seidl, "Tracing evolving subspace clusters in temporal climate data," *Data Mining and Knowledge Discovery*, vol. 24, no. 2, pp. 387–410, Mar. 2012.

[116] A. Ahmed and E. Xing, "Dynamic non-parametric mixture models and the recurrent Chinese restaurant process: With applications to evolutionary clustering," in *Int. Conf. Data Mining (SDM)*. SIAM, 2008, pp. 219–230.

[117] T. Xu, Z. Zhang, S. Y. Philip, and B. Long, "Evolutionary clustering by hierarchical Dirichlet process with hidden Markov state," in *Int. Conf. Data Mining (ICDM)*. IEEE, 2008, pp. 658–667.

[118] A. Ahmed and E. P. Xing, "Timeline: A dynamic hierarchical Dirichlet process model for recovering birth/death and evolution of topics in text stream," *arXiv preprint arXiv:1203.3463*, Mar. 2012.

[119] Y. Chi, X. Song, D. Zhou, K. Hino, and B. L. Tseng, "Evolutionary spectral clustering by incorporating temporal smoothness," in *Int. Conf. Knowledge Discovery and Data Mining (SIGKDD)*. ACM, 2007, pp. 153–162.

[120] ——, "On evolutionary spectral clustering," *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 4, p. 17, Nov. 2009.

[121] J. Rosswog and K. Ghose, "Detecting and tracking spatio-temporal clusters with adaptive history filtering," in *Int. Conf. Data Mining Workshops (ICDMW)*. IEEE, 2008, pp. 448–457.

[122] S. M. Smith and J. M. Brady, "ASSET-2: Real-time motion segmentation and shape tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 814–820, Aug. 1995.

[123] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1631–1643, Aug. 2005.

[124] J. Kiefer, "Sequential minimax search for a maximum," *the American Mathematical Society*, vol. 4, no. 3, pp. 502–506, Sep. 1953.

[125] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM review*, vol. 43, no. 1, pp. 129–159, Feb. 2001.

[126] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics*, vol. 2, no. 1-2, pp. 83–97, Mar. 1955.

[127] D. Greene, D. Doyle, and P. Cunningham, "Tracking the evolution of communities in dynamic social networks," in *Int. Conf. Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 2010, pp. 176–183.

[128] P. Bródka, S. Saganowski, and P. Kazienko, "GED: The method for group evolution discovery in social networks," *Social Network Analysis and Mining*, vol. 3, no. 1, pp. 1–14, Mar. 2013.

[129] 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, Jan. 2011.

[130] J. Yang and Y. Zhang, "Alternating direction algorithms for ℓ_1-problems in compressive sensing," *SIAM Journal on Scientific Computing*, vol. 33, no. 1, pp. 250–278, Feb. 2011.

[131] J. Yang and X. Yuan, "Linearized augmented lagrangian and alternating direction methods for nuclear norm minimization," *Mathematics of Computation*, vol. 82, no. 281, pp. 301–329, Jan. 2013.

[132] T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk, "Fast alternating direction optimization methods," *SIAM Journal on Imaging Sciences*, vol. 7, no. 3, pp. 1588–1623, Aug. 2014.

[133] Z. Lin, R. Liu, and H. Li, "Linearized alternating direction method with parallel splitting and adaptive penalty for separable convex programs in machine learning," *Machine Learning*, vol. 99, no. 2, pp. 287–325, May 2015.

[134] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: A factorization method," *Int. Journal of Computer Vision*, vol. 9, no. 2, pp. 137–154, Nov. 1992.

[135] D. Roemmich and J. Gilson, "The 2004–2008 mean and annual cycle of temperature, salinity, and steric height in the global ocean from the Argo program," *Progress in Oceanography*, vol. 82, no. 2, pp. 81–100, Aug. 2009.

[136] H. Li, F. Xu, W. Zhou, D. Wang, J. S. Wright, Z. Liu, and Y. Lin, "Development of a global gridded Argo data set with Barnes successive corrections," *Journal of Geophysical Research: Oceans*, vol. 122, no. 2, pp. 866–889, Jan. 2017.

[137] N. M. Arzeno-González, "Outcome prediction and structure discovery in healthcare data," Ph.D. dissertation, 2016.

[138] G. L. Pickard and W. J. Emery, *Descriptive physical oceanography: An introduction.* Elsevier, 2016.

[139] N. M. Arzeno-González and H. Vikalo, "Evolutionary clustering via message passing," *Submitted*, 2018.

[140] A. Hashemi, M. Ghasemi, and H. Vikalo, "Submodular observation selection and information gathering for quadratic models," in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97, 2019.

[141] A. Hashemi, O. F. Kilic, and H. Vikalo, "Near-optimal distributed estimation for a network of sensing units operating under communication constraints," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 2890–2895.

[142] A. Hashemi, M. Ghasemi, H. Vikalo, and U. Topcu, "A randomized greedy algorithm for near-optimal sensor scheduling in large-scale sensor networks," in *Proc. American Control Conference*. IEEE, 2018, pp. 1027–1032.

[143] M. Ghasemi, A. Hashemi, U. Topcu, and H. Vikalo, "On submodularity of quadratic observation selection in constrained networked sensing systems," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 4671–4676.

[144] A. Hashemi, R. Shafipour, H. Vikalo, and G. Mateos, "Sampling and reconstruction of graph signals via weak submodularity and semidefinite

relaxation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4179–4183.

[145] ——, "Accelerated greedy sampling of graph signals: A weak submodular optimization framework," *arXiv preprint arXiv:1807.07222*, 2018.

[146] W. J. Welch, "Branch-and-bound search for experimental designs based on d optimality and other criteria," *Technometrics*, vol. 24, no. 1, pp. 41–48, 1982.

[147] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 451–462, 2009.

[148] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *J. Mach. Learn. Res.*, vol. 9, no. Feb, pp. 235–284, 2008.

[149] S. T. Jawaid and S. L. Smith, "Submodularity and greedy algorithms in sensor scheduling for linear dynamical systems," *Automatica*, vol. 61, pp. 282–288, 2015.

[150] H. Zhang, R. Ayoub, and S. Sundaram, "Sensor selection for optimal filtering of linear dynamical systems: Complexity and approximation," in *Conference on Decision and Control (CDC)*. IEEE, Dec. 2015, pp. 5002–5007.

[151] P. Singh, S. Z. Yong, and E. Frazzoli, "Supermodular batch state estimation in optimal sensor scheduling," *IEEE Control Systems Letters*, vol. 1, no. 2, pp. 292–297, Oct. 2017.

[152] P. Singh, M. Chen, L. Carlone, S. Karaman, E. Frazzoli, and D. Hsu, "Supermodular mean squared error minimization for sensor scheduling in optimal Kalman filtering," in *American Control Conference (ACC)*. IEEE, 2017, pp. 5787–5794.

[153] A. Olshevsky, "On (non) supermodularity of average control energy," *IEEE Trans. Control Netw. Syst.*, 2017.

[154] Z. Wang, B. Moran, X. Wang, and Q. Pan, "Approximation for maximizing monotone non-decreasing set functions with a greedy method," *J. Comb. Optim.*, vol. 31, no. 1, pp. 29–43, 2016.

[155] P. Vincent and I. Rubin, "A framework and analysis for cooperative search using UAV swarms," in *Proceedings of the 2004 ACM symposium on Applied computing*. ACM, 2004, pp. 79–86.

[156] M. I. Skolnik, "Radar handbook," 1970.

[157] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *Computer*, vol. 34, no. 8, pp. 57–66, 2001.

[158] K. Chaloner and I. Verdinelli, "Bayesian experimental design: A review," *Statistical Science*, pp. 273–304, 1995.

[159] P. Sebastiani and H. P. Wynn, "Maximum entropy sampling and optimal Bayesian experimental design," *Journal of the Royal Statistical Society: Series B*, vol. 62, no. 1, pp. 145–157, 2000.

[160] P. Flaherty, A. Arkin, and M. I. Jordan, "Robust design of biological experiments," in *Proc. Advances in neural information processing systems*, 2006, pp. 363–370.

[161] A. Krause, H. B. McMahan, C. Guestrin, and A. Gupta, "Robust submodular observation selection," *Journal of Machine Learning Research*, vol. 9, no. Dec, pp. 2761–2801, 2008.

[162] S. Rao, S. P. Chepuri, and G. Leus, "Greedy sensor selection for nonlinear models," in *IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*. IEEE, 2015, pp. 241–244.

[163] X. Shen, S. Liu, and P. K. Varshney, "Sensor selection for nonlinear systems in large sensor networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 4, pp. 2664–2678, 2014.

[164] S. P. Chepuri and G. Leus, "Sparsity-promoting sensor selection for nonlinear measurement models," *IEEE Transactions on Signal Processing*, vol. 63, no. 3, pp. 684–698, 2015.

[165] M. Davidian, *Nonlinear models for repeated measurement data*. Routledge, 2017.

[166] S. Grime and H. F. Durrant-Whyte, "Data fusion in decentralized sensor networks," *Control engineering practice*, vol. 2, no. 5, pp. 849–863, 1994.

[167] R. Olfati-Saber, "Distributed kalman filtering for sensor networks," in *Decision and Control, 2007 46th IEEE Conference on*. IEEE, 2007, pp. 5492–5498.

[168] D. Berberidis and G. B. Giannakis, "Data sketching for large-scale Kalman filtering," in *Int. Conf. Acoustics, Speech and Signal Process.* IEEE, 2016, pp. 6195–6199.

[169] H. L. Van Trees, *Detection, estimation, and modulation theory, part I: detection, estimation, and linear modulation theory*. John Wiley & Sons, 2004.

[170] L. F. Chamon, G. J. Pappas, and A. Ribeiro, "The mean square error in Kalman filtering sensor selection is approximately supermodular," in *Conference on Decision and Control (CDC)*. IEEE, Dec. 2017, pp. 343–350.

[171] U. Feige, "A threshold of ln n for approximating set cover," *Journal of the ACM*, vol. 45, no. 4, pp. 634–652, Jul. 1998.

[172] J. A. Tropp *et al.*, "An introduction to matrix concentration inequalities," *Foundations and Trends® in Machine Learning*, vol. 8, no. 1-2, pp. 1–230, 2015.

[173] R. Bellman, *Introduction to matrix analysis*. SIAM, 1997.

[174] R. V. Hogg and A. T. Craig, *Introduction to mathematical statistics*. Upper Saddle River, New Jersey: Prentice Hall, 1995.

[175] M. Grant, S. Boyd, and Y. Ye, "CVX: Matlab software for disciplined convex programming," 2008.

[176] N. Sorensen and W. Ren, "A unified formation control scheme with a single or multiple leaders," in *American Control Conference, 2007. ACC'07.* IEEE, 2007, pp. 5412–5418.

[177] W. Ren and R. W. Beard, *Distributed consensus in multi-vehicle cooperative control*. Springer, 2008.

[178] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.

[179] A. Nedic, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM Journal on Optimization*, vol. 27, no. 4, pp. 2597–2633, 2017.

[180] A. Koloskova, T. Lin, S. U. Stich, and M. Jaggi, "Decentralized deep learning with arbitrary communication compression," in *International Conference on Learning Representations*, 2020.

[181] H. Tang, S. Gan, C. Zhang, T. Zhang, and J. Liu, "Communication compression for decentralized training," in *Advances in Neural Information Processing Systems*, 2018, pp. 7652–7662.

[182] Z. Shen, A. Mokhtari, T. Zhou, P. Zhao, and H. Qian, "Towards more efficient stochastic decentralized learning: Faster convergence and sparse communication," in *International Conference on Machine Learning*, 2018, pp. 4624–4633.

[183] A. Reisizadeh, H. Taheri, A. Mokhtari, H. Hassani, and R. Pedarsani, "Robust and communication-efficient collaborative learning," in *Advances in Neural Information Processing Systems*, 2019, pp. 8386–8397.

[184] J. N. Tsitsiklis, "Problems in decentralized decision making and computation." Massachusetts Inst of Tech Cambridge Lab for Information and Decision Systems, Tech. Rep., 1984.

[185] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on automatic control*, vol. 50, no. 5, pp. 655–661, 2005.

[186] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control systems magazine*, vol. 27, no. 2, pp. 71–82, 2007.

[187] K. Cai and H. Ishii, "Average consensus on general strongly connected digraphs," *Automatica*, vol. 48, no. 11, pp. 2750–2761, 2012.

[188] ——, "Average consensus on arbitrary strongly connected digraphs with time-varying topologies," *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 1066–1071, 2014.

[189] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.* IEEE, 2003, pp. 482–491.

[190] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE transactions on information theory*, vol. 52, no. 6, pp. 2508–2530, 2006.

[191] R. Carli, F. Bullo, and S. Zampieri, "Quantized average consensus via dynamic coding/decoding schemes," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 20, no. 2, pp. 156–175, 2010.

[192] R. Carli, F. Fagnani, P. Frasca, and S. Zampieri, "Gossip consensus algorithms via quantized communication," *Automatica*, vol. 46, no. 1, pp. 70–80, 2010.

[193] J. Fang and H. Li, "Distributed estimation of Gauss-Markov random fields with one-bit quantized data," *IEEE Signal Processing Letters*, vol. 17, no. 5, pp. 449–452, 2010.

[194] T. Li, M. Fu, L. Xie, and J.-F. Zhang, "Distributed consensus with limited communication data rate," *IEEE Transactions on Automatic Control*, vol. 56, no. 2, pp. 279–292, 2010.

[195] H. Taheri, A. Mokhtari, H. Hassani, and R. Pedarsani, "Quantized push-sum for gossip and decentralized optimization over directed graphs,"

arXiv preprint arXiv:2002.09964, 2020.

[196] Y. Chen, A. Hashem, and H. Vikalo, "Communication-efficient algorithms for distributed optimization over directed graphs," *arXiv preprint arXiv:2005.13189*, 2020.

[197] B. Johansson, M. Rabi, and M. Johansson, "A randomized incremental subgradient method for distributed optimization in networked systems," *SIAM Journal on Optimization*, vol. 20, no. 3, pp. 1157–1170, 2010.

[198] J. Konevcny and P. Richtarik, "Randomized distributed mean estimation: Accuracy vs. communication," *Frontiers in Applied Mathematics and Statistics*, vol. 4, p. 62, 2018.

[199] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu, "1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[200] N. Strom, "Scalable distributed DNN training using commodity GPU cloud computing," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[201] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, "SignSGD: Compressed optimisation for non-convex problems," in *International Conference on Machine Learning*, 2018, pp. 560–569.

[202] J. Konecny, H. B. McMahan, F. X. Yu, P. Richtarik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[203] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization," *arXiv preprint arXiv:1909.13014*, 2019.

[204] J. Sun, T. Chen, G. Giannakis, and Z. Yang, "Communication-efficient distributed learning via lazily aggregated quantized gradients," in *Advances in Neural Information Processing Systems*, 2019, pp. 3365–3375.

[205] J. Wang and G. Joshi, "Cooperative SGD: A unified framework for the design and analysis of communication-efficient SGD algorithms," *arXiv preprint arXiv:1808.07576*, 2018.

[206] D. Basu, D. Data, C. Karakus, and S. Diggavi, "Qsparse-local-SGD: Distributed SGD with quantization, sparsification and local computations," in *Advances in Neural Information Processing Systems*, 2019, pp. 14 668–14 679.

[207] T. Chen, G. Giannakis, T. Sun, and W. Yin, "LAG: Lazily aggregated gradient for communication-efficient distributed learning," in *Advances in Neural Information Processing Systems*, 2018, pp. 5050–5060.

[208] X. Yi, S. Zhang, T. Yang, K. H. Johansson, and T. Chai, "Linear convergence of first- and zeroth-order primal-dual algorithms for distributed nonconvex optimization," *arXiv preprint arXiv:1912.12110*, 2020.

[209] Y. Tang and N. Li, "Distributed zero-order algorithms for nonconvex multi-agent optimization," in *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2019, pp. 781–786.

[210] A. Rogozin and A. Gasnikov, "Projected gradient method for decentralized optimization over time-varying networks," *arXiv preprint*.

[211] S. Ma, R. Bassily, and M. Belkin, "The power of interpolation: Understanding the effectiveness of SGD in modern over-parametrized learning," *arXiv preprint arXiv:1712.06559*, 2017.

[212] S. Vaswani, F. Bach, and M. Schmidt, "Fast and faster convergence of SGD for over-parameterized models and an accelerated perceptron," *arXiv preprint arXiv:1810.07288*, 2018.

[213] S. Vaswani, A. Mishkin, I. Laradji, M. Schmidt, G. Gidel, and S. Lacoste-Julien, "Painless stochastic gradient: Interpolation, line-search, and convergence rates," in *Advances in Neural Information Processing Systems*, 2019, pp. 3727–3740.

[214] M. Schmidt and N. L. Roux, "Fast convergence of stochastic gradient descent under a strong growth condition," *arXiv preprint arXiv:1308.6370*,

2013.

[215] V. Cevher and B. C. Vũ, "On the linear convergence of the stochastic gradient method with constant step-size," *Optimization Letters*, vol. 13, no. 5, pp. 1177–1187, 2019.

[216] J. Wang, A. K. Sahu, Z. Yang, G. Joshi, and S. Kar, "Matcha: Speeding up decentralized SGD via matching decomposition sampling," *arXiv preprint arXiv:1905.09435*, 2019.

[217] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *arXiv preprint arXiv:1912.04977*, 2019.

[218] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[219] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.

[220] D. Xu, T. Long, and J. Gao, "Lstm-assisted evolutionary self-expressive subspace clustering," *arXiv preprint arXiv:1910.08862*, 2019.

[221] J. B. Orlin, A. S. Schulz, and R. Udwani, "Robust monotone submodular function maximization," in *Proc. International Conference on Integer*

Programming and Combinatorial Optimization. Springer, 2016, pp. 312–324.

[222] C. Dwork, "Differential privacy: A survey of results," in *International conference on theory and applications of models of computation*. Springer, 2008, pp. 1–19.

[223] S. Bochner, *Lectures on Fourier Integrals*. Princeton University Press, 1959, vol. 42.

[224] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU Press, 2012, vol. 3.

[225] K. R. Davidson and S. J. Szarek, "Local operator theory, random matrices and banach spaces," *Handbook of the geometry of Banach spaces*, vol. 1, no. 317-366, p. 131, 2001.

[226] M. Ghasemi, A. Hashemi, H. Vikalo, and U. Topcu, "Learning in markov decision processes with varying losses: High-probability regret bounds under bandit feedback," *arXiv*, 2020.

[227] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, Dec. 2008.

[228] M. E. Levenson, J. F. Locke, and H. Tate, "3766," *The American Mathematical Monthly*, vol. 45, no. 1, pp. 56–58, 1938. [Online]. Available: http://www.jstor.org/stable/2303491

[229] A. Hashemi, B. Zhu, and H. Vikalo, "Sparse tensor decomposition for haplotype assembly of diploids and polyploids," *BMC genomics*, vol. 19, no. 4, p. 191, 2018.

[230] S. Lang, *Algebraic number theory*. Springer Science & Business Media, 2013, vol. 110.

[231] R. Shafipour, A. Hashemi, G. Mateos, and H. Vikalo, "Online topology inference from streaming stationary graph signals," in *2019 IEEE Data Science Workshop (DSW)*, 2019.

[232] A. Hashemi, A. Acharya, R. Das, H. Vikalo, S. Sanghavi, and I. Dhillon, "On the benefits of multiple gossip steps in communication-constrained decentralized optimization," *arXiv preprint arXiv:2011.10643*, 2020.

[233] J. Bolte, T. P. Nguyen, J. Peypouquet, and B. W. Suter, "From error bounds to the complexity of first-order descent methods for convex functions," *Mathematical Programming*, vol. 165, no. 2, pp. 471–507, 2017.

[234] H. Zhang, "New analysis of linear convergence of gradient-type methods via unifying error bound conditions," *Mathematical Programming*, vol. 180, no. 1, pp. 371–416, 2020.