2015

Network topology identification based on measured data

Mahdi Zamanighomi
Iowa State University

Network topology identification based on measured data

by

Mahdi Zamanighomi

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Electrical Engineering

Program of Study Committee:

Zhengdao Wang, Major Professor

Kris De Brabanter

Nicola Elia

Aditya Ramamoorthy

Namrata Vaswani

Iowa State University

Ames, Iowa

2015

DEDICATION

I dedicate my thesis to my loving family that have continued to provide encouragement and support throughout my journey in the doctoral program.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

First, I would like to begin by thanking my advisor, Dr. Zhengdao Wang, for his unwavering support and advice that has continued to motivate me and shape my research during the Ph.D program. His mentorship has made my experience at Iowa State University better than I could have ever imagined.

Second, I would like to thank my committee members: Dr. Kris De Brabanter, Dr. Nicola Elia, Dr. Aditya Ramamoorthy, and Dr. Namrata Vaswani for their insightful feedback and questions that have guided my research.

Additionally, I would like to thank Dr. Georgios B. Giannakis and Dr. Mostafa Zamanian, for the trainings that they have each provided me in their respective fields. Dr. Giannakis's training on modern big data optimization and Dr. Zamanian's training on bioinformatics have played an invaluable role in the evolution of my work.

Finally, I would like to thank Dr. Konstantinos Slavakis for the discussion on the filter design problem and Dr. Michael Kimber for the comments on gene network inference.

ABSTRACT

We consider the problem of modeling of systems and learning of models from a limited number of measurements. We also contribute to the development of inference algorithms that require high-dimensional data processing. As an inspiring example, a growing interest in biology is to determine dependencies among genes. Such problem, known as gene regulatory network inference, often leads to identifying of large networks through relatively small gene expression data.

The main purpose of the thesis is to develop models and learning methods for data based applications. In particular, we first build a dynamical model for gene-gene interactions to learn the topology of gene regulatory networks from gene expression data. Our proposed model is applicable to such complex gene regulatory networks that contain loops and non-linear dependencies between genes. We seek to use dynamical gene expression data when a system is perturbed. Ideally, such dynamical changes result from local genetic or chemical perturbations of systems in steady state that can be captured in a time-dependent manner. We present a low-complexity inference method that can be adapted to incorporate other information measured across a biological system. The performance of our method is examined employing both simulated and real datasets. This work can potentially inform biological discovery relating to interactions of genes in disease-relevant networks, synthetic networks, and networks immediate to drug response.

Along with the main objective of the thesis, we next seek to estimate high-dimensional covariance matrices based on a few partial observations. Notably, covariance matrices can be utilized to form networks or improve network inference. We assume that the true covariance matrix can be modeled as a sum of Kronecker products of two lower dimensional matrices. To estimate covariance, we propose a convex optimization approach computationally affordable in high-dimensional setting and applicable to missing data. Regardless of whether the process

producing missing values is random or not, our novel scheme can be used without employing any imputation methods. We characterize the symmetry and positive definiteness of the estimated covariance and further shed light on its square error performance. The effect of missing values on the estimation error is mathematically presented and numerical results are illustrated to validate our method.

In addition to the modeling and learning, we improve inference algorithms that involve high-dimensional data processing. Specifically, we attempt to reduce the complexity of the linear minimum mean-square error (LMMSE) estimation when observation vectors have high-dimensionality and contain missing entries. In this context, the standard LMMSE estimator must be re-computed whenever missing values take place at different positions. Instead, we propose a method to first construct the LMMSE estimator based on complete data statistics. We then apply this estimator to the data vector with missing values replaced by zeros. We finally establish a low-complexity update according to missing data patterns to modify our estimation and preserve the LMMSE optimality.

CHAPTER 1. INTRODUCTION

Data analysis of large-scale systems with respect to the modeling, the learning, and the inference is a fundamental and challenging problem in variety of applications. Examples of such applications include bioinformatics (gene regulatory networks, genomic sequence analysis), social networks (twitter, facebook), and weather forecast. As data become more available, we seek for converting data to rational and intuitive information amendable for human decision and discovery. To successfully arrive at this actionable information, we require flexible models that sufficiently describe the physical or chemical behavior of systems as well as learning of models from measured data.

The thesis mainly develops generic models and low-complexity algorithms suitable for data based applications. In particular, we investigate two fundamental problems as follows. We first build a framework to learn the topology of gene regulatory networks from dynamical gene expression data. In this scenario, gene networks are directed, present complex gene-gene interactions, and contain loops while the measured data is limited. Next, we seek to estimate high-dimensional covariance matrices based on Kronecker product models and observations that are partially captured. The estimated covariance matrix can be used to construct a graphical model or improve network inference. In addition to the modeling and learning problems, we contribute to the improvement of inference algorithms that involve high-dimensional data processing. Specifically, we derive an algorithm to reduce the complexity of LMMSE estimation for high-dimensional data with missing values. In the next sections, we briefly demonstrate the significance of the mentioned subjects as well as data structures, problem settings, and proposed methods.

1.1 Gene Regulatory Network Inference

Numerous biological processes take place in the cells. These processes are regulated based on information within the living systems genome. To obtain a better understanding and demonstration of the regulatory processes, it is necessary to shed light on interactions among many elements of biological systems using available experimental data on the DNA and RNA level as well as the protein and metabolite level. In this scenario, a gene regulatory network presents dependencies between biological components and often genes are modeled as nodes and their corresponding interactions as edges [2, 3].

Inferring gene regulatory networks from gene expression data has proved to be useful in biological network discovery [4, 5]. For example, it can be used to find the gene regulatory networks immediate to drug response, to predict interactions among genes in disease-relevant networks, and to probe of properties of synthetic networks. Moreover, the inference data can be employed to determine and prioritize candidate genetic elements for downstream biological and *in vivo* validation.

The development of computational methods to infer gene regulatory networks from gene expression datasets is an important challenge. Several different approaches for gene network reconstruction have been proposed such as Bayesian models [6], Boolean models [7], and graphical Gaussian models [8]. However, these techniques are often bound to fail in large-scale settings, are related to particular biological and experimental structures, and require biological information that is typically unavailable and difficult to determine [9].

The recent advances in higher-throughput sequencing technologies, combined with more precise modes of genetic perturbation, offers an opportunity to obtain efficient approaches for gene network inference [10–13]. Previous studies for network recovery via perturbation tend to be restricted to the analysis of steady-state gene expression [14, 15]. Here, we seek to develop algorithms for network inference that depends on dynamical gene expression data coupled to genetic or chemical perturbation.

In chapter 2, we first present a system of nonlinear ordinary differential equations to model eukaryotic gene regulation, which offers a new extension of an existing thermodynamic and statistical mechanic approach to modeling polymerase binding [16, 17].

We then propose a step-wise technique to identify gene-gene interactions that expand from a known point of genetic or chemical perturbation of systems in steady state. Our approach seeks to use information contained in the dynamic gene expression changes that occur when systems are perturbed. The novel approach sequentially detects genes that fall out of steady state and incorporates them into an increasing series of low-complexity optimization problems. In this process, we consider an important feature of gene regulatory networks that assumes genes are sparsely connected. We emphasize that the new approach can be adapted to employ other information measured across biological systems. We finally elucidate the identifiability of gene regulatory networks and show promising results of our algorithms using simulated and real datasets.

1.2 High-Dimensional Covariance Matrix Estimation with Missing Data

The problem of high-dimensional covariance matrix estimation is fundamental and has received high attention in numerous applications such as portfolio risk assessment [18], genomics [19], user-ratings data [20], and weather forecast [21]. Such high-dimensional approximation is intrinsically challenging especially when the model dimension is comparable or significantly larger than the sample size. To deal with the curse of high dimensionality, recent studies have been focused on low rank and sparse covariance estimation [22–24]. However, covariance matrices are not necessarily low rank or sparse and could exhibit different structures. For example, a class of covariance matrices follows Kronecker product structure, that is when the covariance can be represented as a sum of Kronecker products of two lower dimensional matrices. This model occurs in many applications, for instance, geostatistics [25], bioinformatics [26], and wind speed prediction [21].

In practical applications, measurements may not be fully captured, which leads to observation vectors with missing entries. It is well known that ignoring missing data in statistical analysis could lead to an unacceptable bias in parameter estimates [27]. Thus, the design of appropriate methods for dealing with missing data is essential for the estimation of variables.

There are several approaches to handle missing values, such as maximum likelihood and multiple imputation [28], but each of them results in different performance. One simple approach is to exclude variables for which observations are missing and then limit the analysis to the fully observed measurements. In gene expression data where the majority of genes are affected by missing data, we are left with few variables and consequently, removing variables is waste of available measurements.

An alternative approach is to impute missing values and then proceed to a desired analysis of the available and imputed data. The most common methods for estimating missing data is the maximum likelihood, expectation-maximization (EM) algorithm, and multiple imputation which require restrictive assumptions on missing data distributions and they are computationally expensive in high-dimensional setting [29].

In chapter 3, we consider covariance matrices with the Kronecker product structure and seek for their estimations through partial observations. In particular, we generalize the permuted rank-penalized least square method [21] to the case of missing data. We assume that the position of missing values are available, but missing data mechanisms are unknown. We emphasize that our method can be applied to any missing data patterns, whether the process producing missing values is random or not, and does not depend on imputation techniques. We also assume that observation vectors are independent and identically distributed (i.i.d) multivariate Gaussian provided that no missing value occurs.

We introduce a novel unbiased estimator that utilizes the standard sample covariance matrix, even though the sample covariance can not be computed due to the presence of missing data. We employ this new unbiased estimator to propose a convex optimization approach for estimating covariance matrices with Kronecker product structure. For this scenario, we characterize the symmetry and positive definiteness of the estimated covariance. We further establish a tight upper bound on the square error (SE) of our procedure and mathematically reveal the

consequences of missing data on the SE performance. Numerical simulations are presented to validate our approach.

1.3 High-Dimensional LMMSE Estimation with Missing Data

In statistics, the LMMSE estimation refers to a method that linearly approximates a desired signal from an observation through minimizing the mean-square error. This approach is widely applied to many fields, such as control theory [30, 31], signal processing [32], and communications [33].

The LMMSE estimator designed for vector observations involves the inverse of the data covariance matrix. Such matrix inversion may be computationally expensive in high-dimensional setting. The multistage Wiener filter proposed by [34] is an alternative procedure to implement the LMMSE filter and avoid the direct inversion of the data covariance matrix. In this scenario, decompositions based on orthogonal projections are employed to prevent covariance inversion.

In practical applications, observation vectors may contain missing entries as discussed in the previous section. Given all data statistics and the position of missing values, the LMMSE estimator can still be obtained and applied to observations that are fully captured. Nevertheless, the required computations could be intensive in high-dimensional setup since the LMMSE estimator must be re-derived for different missing data patterns.

In chapter 4, we consider high-dimensional data with missing values and seek to build an algorithm to reduce the complexity of LMMSE estimation. We first design the LMMSE filter based on the data statistics. Then, we apply this full-data processing to observation vectors with missing values where all missing entries are replaced by zeros. We finally derive a low-complexity update, that depends on missing data patterns, to modify our estimate. Notably, our procedure maintains the LMMSE optimality and also achieves lower complexity compared to constructing a new LMMSE filter whenever the position of missing data changes. Moreover, the proposed update is applicable to the multistage Wiener filter and dose not hurt its LMMSE optimality.

CHAPTER 2. GENE REGULATORY NETWORK INFERENCE FROM PERTURBED TIME-SERIES EXPRESSION DATA

Modified from a paper submitted to *IEEE/ACM Transactions on Computational Biology and Bioinformatics*

Mahdi Zamanighomi[1], Mostafa Zamanian[2], Michael J. Kimber[3], and Zhengdao Wang[1]

We focus on the modeling and learning of gene-gene interactions based on dynamical datasets, known as gene regulatory network inference. The reconstruction of gene regulatory networks from gene expression data is a challenging problem due to the complexity of interactions among genes as well as limited sources of measured data. A variety of models and methods have been developed to address different aspects of this important problem. However, these techniques are often difficult to scale, are narrowly focused on particular biological and experimental platforms, require experimental data that are typically unavailable and difficult to ascertain. The more recent availability of higher-throughput sequencing platforms, combined with more precise modes of genetic perturbation, presents an opportunity to formulate more robust and comprehensive approaches to gene network inference. Here, we propose a step-wise framework for identifying gene-gene regulatory interactions that expand from a known point of genetic or chemical perturbation using time series gene expression data. This novel approach sequentially identifies non-steady state genes post-perturbation and incorporates them into a growing series of low-complexity optimization problems. The governing ordinary differential equations of this model are rooted in the biophysics of stochastic molecular events that underlie gene regulation, delineating roles for both protein and RNA-mediated gene regulation. We

[1]Department of Electrical and Computer Engineering, Iowa State University, Ames, IA USA
[2]Department of Molecular Biosciences, Northwestern University, Evanston, IL USA
[3]Department of Biomedical Sciences, Iowa State University, Ames, IA USA

show the successful application of our core algorithms for network inference using simulated and real datasets.

2.1 Introduction

The elucidation of gene regulatory networks is fundamental to understanding the dynamic functions of genes in biochemical, cellular and physiological contexts. The architectures of networks comprised of small numbers of genes are generally deciphered using classical experimental techniques, where biophysical data describing the interactions of genes and their products can lead to useful models and well-characterized systems. While this validated experimental tract continues to provide valuable biological insight, it is ultimately laborious and costly, and often demands strategies uniquely tailored to individual biological systems and problems. Furthermore, the models that result from these efforts tend to be limited to a very modest subset of genes, typically suffer from a lack of temporal resolution, and focus narrowly on very particular modes of interaction.

To complement these established approaches, there is a great impetus to develop more efficient and uniformly applicable *in silico* methods for gene network inference and discovery [2, 12, 35–39]. Of particular interest is the goal of gene network inference using perturbed gene expression data [10, 13, 40–46], whereby gene expression levels are measured under the influence of either genetic or chemical perturbations of the system. Previous attempts at network reconstruction via perturbation tend to be limited to the analysis of steady-state gene expression. The growing ubiquity of next-generation sequencing technologies presents a powerful high-throughput substrate for capturing the dynamic and non steady-state aspects of gene expression.

In this work, we seek to develop a robust framework for network inference that relies on temporal gene expression data coupled to genetic or chemical perturbation. In a departure from previous attempts, our formulation does not require *a priori* knowledge beyond the set of temporal gene expression measurements, acknowledges the non-steady state and dynamic nature of gene expression, incorporates both RNA and protein-mediated regulation, sequentially absorbs a growing number of genes into the regulatory network immediate to perturbation,

aims for sparsity in network topology, and reduces an otherwise complex optimization problem into a convex form that can be solved efficiently.

Notation: Throughout this chapter $\{d, i, j, k, l\}$ count integer numbers. Column vectors and matrices are indicated by bold lower-case and upper-case letters, respectively. We use $\mathbf{1}$ to show a vector with all entries 1 and $\mathbf{0}$ a vector with all entries 0. The set of real numbers is denoted as $\mathbb{R}$ and positive real numbers $\mathbb{R}^+$. The indicator function $\mathbb{I}_{\mathbb{R}^+}\{x\}$ has the value one when $x \in \mathbb{R}^+$, otherwise zero. The operator sign($\mathbf{x}$) replaces each entry of $\mathbf{x}$ with its sign function value. We use $(\mathbf{X})^T$ to denote transpose of $\mathbf{X}$, $dx(t)/dt$ and $x'(t)$ the first derivative of $x(t)$ with respect to time t, $\|\mathbf{x}\|_1$ the 1-norm of vector $\mathbf{x}$, $\|\mathbf{x}\|_2$ the 2-norm of vector $\mathbf{x}$, and $\|\mathbf{X}\|$ the largest singular value of matrix $\mathbf{X}$. We explicitly state a function of time in the form $\mathbf{x}(t)$. This is to be distinguished from vectors of the form $\mathbf{x}(i)$, where i is a positive integer representing the ith entry of the vector $\mathbf{x}$.

2.2 System Model

2.2.1 Gene Expression Datasets and Perturbation

Let $x_i(t)$ and $y_i(t)$ denote the RNA-level and protein-level expression of gene i at time t, respectively. We define an $m \times n$ gene expression matrix

$$\mathbf{X} = \begin{pmatrix} x_1(t_1) & \dots & x_1(t_n) \\ \vdots & \ddots & \vdots \\ x_m(t_1) & \dots & x_m(t_n) \end{pmatrix},$$

where m indicates the total number of genes in the system and n the total number of samples in the time series. In practical cases, with expression data originating from microarray or RNA-Seq experiments, $m \gg n$.

Here, we are concerned with datasets with known points of perturbation. In this experimental scheme, a gene x_i^p is specifically targeted for perturbation via either gene suppression or gene over-expression. Perturbation is triggered at a known time point after a series of presumably steady state measurements. Without loss of generality, it is assumed that the starting point of perturbation occurs at t_1 and prior measurements are approximately steady state. Datasets

from experiments that conform to this scheme are in the following form, where $x_i^p(t_1)$ represents the point of perturbation and L denotes the total number of samples post-perturbation.

$$\mathbf{X}^p := \begin{pmatrix} \dots & x_1(t_0) & x_1(t_1) & \dots & x_1(t_L) \\ \ddots & \vdots & \vdots & \ddots & \vdots \\ \dots & x_i(t_0) & x_i^p(t_1) & \dots & x_i^p(t_L) \\ \ddots & \vdots & \vdots & \ddots & \vdots \\ \dots & x_m(t_0) & x_m(t_1) & \dots & x_m(t_L) \end{pmatrix}$$

2.2.2 Conceptual Description of Inference Approach

We consider a non-perturbed system as one with genes in steady state, i.e., where $dx_i(t)/dt$ and $dy_i(t)/dt$ are approximately zero. After a series of steady state expression measurements, a protein-encoding gene in this system is perturbed to bring about a dramatic change in its expression level, i.e., where $|dx_i^p(t)/dt| \gg 0$, followed by a series of post-perturbation measurements. The discrete set of expression measurements, with appropriate temporal resolution, can be used to produce continuous gene trajectory curves.

For a short period of time post-perturbation, the perturbed gene falls out of steady state while all other genes remain effectively in steady state. The induced change in RNA expression, Δx_i^p, is coupled to a delayed change in protein expression, Δy_i^p. This shift in protein availability leads, through the immediate regulatory network of the perturbed protein, to changes in the expression levels of other genes.

Consider the set of all genes that are affected by Δy_i^p at time t. We divide this set into protein and miRNA-encoding subsets. The set of all indices that correspond to protein-encoding genes is shown as $G(t)$, and $M(t)$ is set of all indices that correspond to miRNA-encoding genes. We define the collection of RNA expression data for these subsets as $\mathcal{X}_{G(t)} := \{x_i(t)|i \in G(t)\}$ and $\mathcal{X}_{M(t)} := \{x_i(t)|i \in M(t)\}$, respectively. We further define the collection of protein expression levels for subset G as $\mathcal{Y}_{G(t)} := \{y_i(t)|i \in G(t)\}$.

In principle, we can identify genes that fall out of steady state in an ordered manner with gene trajectory analysis. The growing set of non-steady state actors in the system, both

members of $G(t)$ and $M(t)$, can then be sequentially incorporated into a growing network of interactions to be modeled.

2.2.3 Governing Regulatory Equations

Gene and protein expression dynamics are often modeled in the form of ordinary differential equations [47–49], with gene-specific rate constants for molecular synthesis and degradation and gene-specific functions accounting for the regulatory effects of proteins. We introduce miRNA-mediated gene regulation into this model and establish functions for both protein and RNA regulatory interactions that complement our overall approach to network inference. The architecture of the gene regulatory circuit under consideration is depicted in Figure 2.1.

This circuit can be represented in the following form:

$$\frac{dx_i(t)}{dt} = \tau_i f_i(\mathcal{Y}_{G(t)}) - \left(\lambda_i^{RNA} + g_i(\mathcal{X}_{M(t)})\right) x_i(t) \tag{2.1}$$

$$\frac{dy_i(t)}{dt} = \left(r_i - h_i(\mathcal{X}_{M(t)})\right) x_i(t) - \lambda_i^{Prot} y_i(t), \tag{2.2}$$

where τ_i is the rate of transcription when RNA polymerase (RNAP) is bound, $f_i(\mathcal{Y}_{G(t)})$ is the probability of RNAP binding, λ_i^{RNA} is the rate of basal RNA degradation, $g_i(\mathcal{X}_{M(t)})$ incorporates the effect of miRNA-mediated RNA degradation, r_i is the rate of translation, $h_i(\mathcal{X}_{M(t)})$ accounts for the effect of miRNA-mediated translational inhibition, and λ_i^{Prot} is the rate of protein degradation. It follows from the biological definitions of the system that parameters τ_i, λ_i^{RNA}, r_i, and λ_i^{Prot} are to be positive and $h_i(\mathcal{X}_{M(t)}) \leq r_i$.

2.2.4 Protein-Mediated Regulation

For each gene, i, we employ an existing statistical thermodynamic framework [16, 17] to model the equilibrium probability of RNAP binding to a gene of interest as a function of protein regulators, $f_i(\mathcal{Y}_{G(t)})$. We extend a previous derivation of multiple protein regulators operating on a single gene [15] and explicitly show that the general form can be expressed as a function of non-steady state genes, $G(t)$ (see Appendix A). Although steady state regulators play an active role in gene regulation, we can effectively restrict our binding probability function

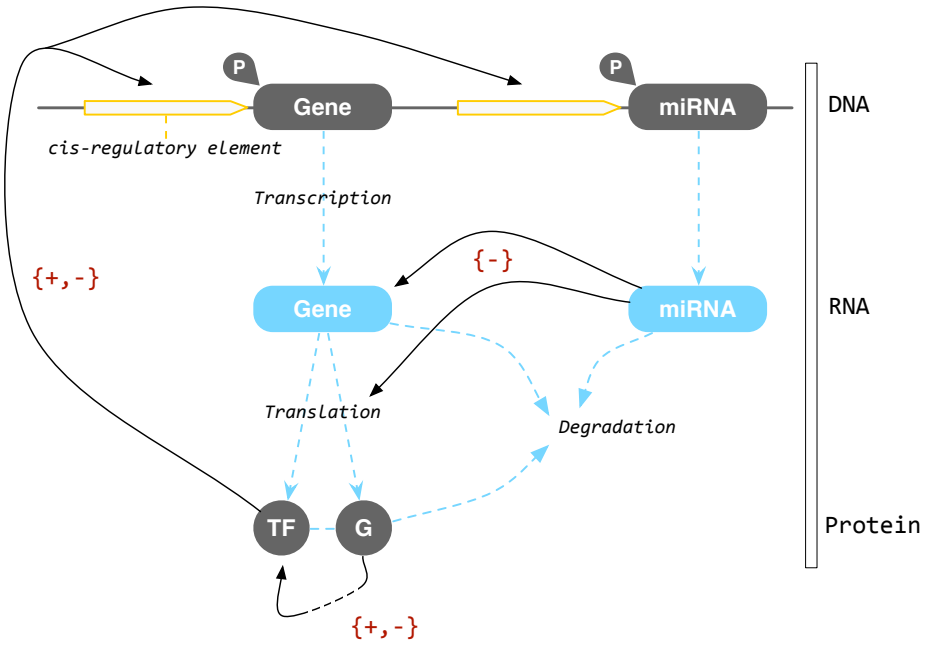

Figure 2.1: Gene regulatory circuit. 'Gene' represents protein-encoding genes and 'miRNA' represents miRNA-encoding genes. Protein-encoding genes can give rise to transcription factors ('TF') that directly exert influence on the cis regions of other genes, as well as non-TF proteins ('G') that can indirectly act through TFs and various biochemical cascades. These protein regulators ultimately affect the equilibrium probability of RNA polymerase ('P') being bound to a promoter of interest. Additionally, miRNAs can directly repress expression via targetted RNA degradation or translational repression. All proteins and RNAs in this system undergo varying rates of chemical degradation.

to the activities of perturbed regulators. This function is shown below:

$$f_i(\mathcal{Y}_{G(t)}) = \frac{a_{i0} + \sum_{j=1}^{N(t)} a_{ij} \prod_{k \in S_{ij}(t)} y_k(t)}{1 + \sum_{j=1}^{N(t)} b_{ij} \prod_{k \in S_{ij}(t)} y_k(t)} \tag{2.3}$$

where $S_{ij}(t)$, $0 \leq j \leq N(t)$, is the list of all possible protein products of genes within set $G(t)$ that interact to form regulatory complexes. For instance when $G(t) = \{1, 2\}$, there are $N(t) + 1 = 4$ complexes as the empty set $S_{i0} = \{\emptyset\}$, $S_{i1} = \{1\}$, $S_{i2} = \{2\}$, and $S_{i3} = \{1, 2\}$. To reduce the complexity of this model, we restrict $S_{ij}(t)$ to all terms up to the second-

order, accounting for the interactions of no more than two proteins bound together. In this arrangement, a complex represents either the products of a single gene or the interaction of the products of any two genes that can form a regulatory agent. However, any number of complexes can additively combine to regulate single genes. The numbering of complexes is an arbitrary labeling of genes and gene-pairs in the system. The coefficients $0 \leq a_{ij} \leq b_{ij}$ depend on the binding energies of regulator complexes that act on a promoter region, and a_{i0} and b_{i0} correspond to the case where no regulators are bound to the promoter region ($\prod_{k \in S_{i0}(t)} y_k(t) := 1$). It is assumed all coefficients are normalized so that $b_{i0} = 1$.

2.2.5 miRNA-Mediated Regulation

To account for the effects of miRNA regulation on each gene, we draw on previous mass-law (linear) models [50,51] that acknowledge two primary routes of inhibitory regulation: (i) cleavage or degradation of target transcript and (ii) translational repression. These are represented by functions $g_i(\mathcal{X}_{M(t)})$ and $h_i(\mathcal{X}_{M(t)})$, respectively. The former is a modifier of the RNA degradation rate constant, λ_i^{RNA}, while the latter detracts from RNA available to the translational machinery without affecting RNA concentration as assayed. These functions are shown below.

$$g_i(\mathcal{X}_{M(t)}) = \sum_{j \in \mathcal{X}_{M(t)}} \lambda_{ij}^{RNA} x_j(t) \tag{2.4}$$

$$h_i(\mathcal{X}_{M(t)}) = \sum_{j \in \mathcal{X}_{M(t)}} \lambda_{ij}^{Prot} x_j(t) \tag{2.5}$$

where both λ_{ij}^{RNA} and λ_{ij}^{Prot} are greater than or equal to zero.

We impose the constraint that any given miRNA can only inhibit the expression of a particular target mRNA through one mode of regulation, either transcript cleavage or translational repression. This is reasonable, given that the particular pathway of inhibition is determined by the specificity of binding between a particular miRNA and a seed site on a target transcript, which is a fixed interaction for each miRNA-mRNA pairing [52–54]. This constraint takes the following mathematical form

$$\mathbb{I}_{\mathbb{R}^+}\{\lambda_{ij}^{RNA}\} + \mathbb{I}_{\mathbb{R}^+}\{\lambda_{ij}^{Prot}\} = 1.$$

2.3 Network Inference Algorithm

Subsections 2.3.1 - 2.3.4 contain all the core algorithmic components in our proposed inference pipeline. A graphical overview of how these modular algorithms form a framework for gene network inference is shown in Figure 2.2. This linear ordering of post-processing and inference steps, although designed for a normalized gene expression dataset involving a precise perturbation, is robust and flexible.

2.3.1 Modeling and Estimation of Gene Expression

Normalized gene expression values, such that $x_i(t) \leq 1$, are the given input for the algorithms described in this and subsequent sections. In reality, gene expression trajectories are inevitably noisy, which perturb the model parameters away from the true values. To reduce this noise effect, we first represent gene expressions as a linear combination of basis functions in the following form

$$x_i(t) = \sum_{d=1}^{D} \theta_{id}\varphi_d(t) = \boldsymbol{\varphi}(t)^T \boldsymbol{\theta}_i, \tag{2.6}$$

where D is the total number of bases and θ_{id} the coefficient of the dth basis function, $\varphi_{id}(t)$. The basis functions are chosen to take the form of a B-spline (See Appendix A). Although all genes are associated with a common set of basis functions in (2.6), one can consider different sets of basis functions for different genes.

The form of (2.6) allows us to fit a continuous function for a set of discrete gene expression measurements, using the following minimization

$$\text{(P1)} \qquad \min_{\boldsymbol{\theta}_i} \left\| \sum_{j=1}^{L} \left(x_i(t_j) - \boldsymbol{\varphi}(t_j)^T \boldsymbol{\theta}_i \right) \right\|_2 + \gamma_\theta \boldsymbol{\theta}_i^T \mathbf{K} \boldsymbol{\theta}_i,$$

where the roughness penalty $\boldsymbol{\theta}_i^T \mathbf{K} \boldsymbol{\theta}_i = \int_{t_1}^{t_L} \left(d^2 x_i(t)/dt^2 \right)^2 dt$ and $\mathbf{K}$ is a roughness matrix with the (j,k)th entry $\int_{t_1}^{t_L} \varphi_j''(t)\varphi_k''(t)dt$. Here, the first term is intended to diminish noise within measurements and the second term is intended to smooth our approximations. The parameter γ_θ is tuned by cross validation where training data is available, otherwise it can be drawn from a characterized network from the nearest available biological system.

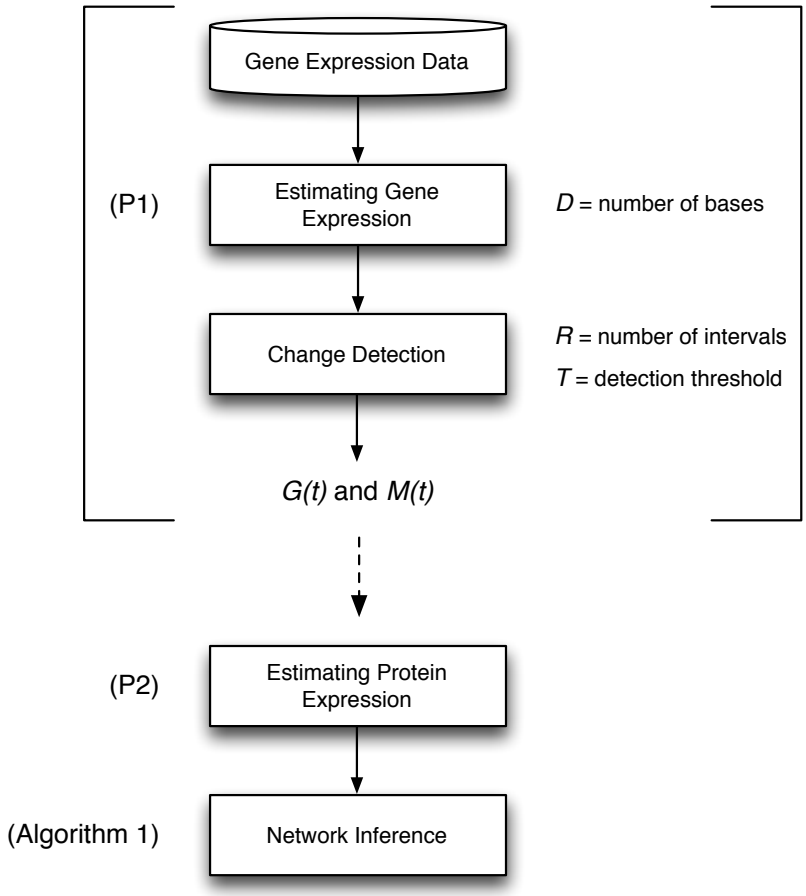

Figure 2.2: Overview of gene inference pipeline, beginning with a normalized gene expression dataset. The first stage involves the estimation of all gene trajectories as noise-free and continuous curves (P1), followed by segmentation into equally-spaced intervals for detection of significant changes in expression. The time-dependent expansion of $G(t)$ and $M(t)$, along with the result of (P1), seed downstream network inference. In the next stage, (P2) is used to estimate protein expression, and finally all obtained results are considered in algorithm 1 to produce a regulatory network map. Figure 2.3 provides a graphical description of the bracketed pre-inference stages.

Employing (P1), our estimation to $x_i(t)$, denoted as $\hat{x}_i(t)$, is a continuous function in time and its first derivative can be easily calculated as

$$\frac{d\hat{x}_i(t)}{dt} \simeq \frac{\hat{x}_i(t+\Delta t) - \hat{x}_i(t)}{\Delta t}. \tag{2.7}$$

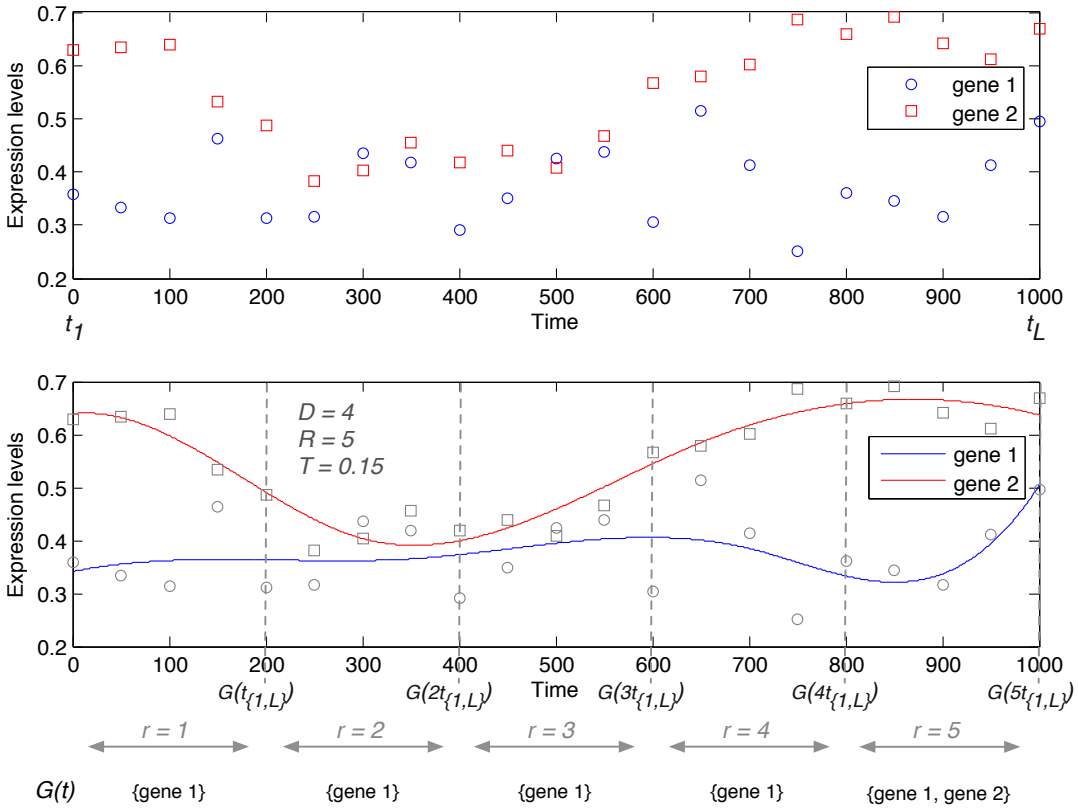

Figure 2.3: The bracketed pre-inferenced stages of the pipeline in Figure 2.2 are shown graphically. Discrete expression data from two genes and a small number of basis functions are utilized to produce continuous models of expression (P1), followed by segmentation and change detection. In this simple example, a change in gene 2 is detected in sub-interval $r = 1$, and a change in gene 1 is detected in sub-interval $r = 5$.

Throughout the rest of the chapter, it is assumed that our samples are taken from $\hat{x}_i(t)$ and therefore, any arbitrary number of samples, L, is achievable. We further replace $\hat{x}_i(t)$ with $x_i(t)$ for notational convenience.

2.3.2 Detection of Perturbed Genes

We can introduce a simple first approach for detecting when individual genes exit steady state post-perturbation. Gene expression models generated via (P1) are essentially smooth and noise-free when the total number of bases is restricted to an appropriately small number, D. High-frequency gene trajectories, whether a product of noise or periodicity in expression

[55, 56], are converted into flat trajectories. This property allows us to detect when significant non-periodic deviations occur with respect to the initial steady state measurement(s). More precisely, time interval $[t_1, t_L)$ is divided into R sub-intervals as $[rt_{\{1,L\}}, (r+1)t_{\{1,L\}})$ for all $1 \leq r \leq R$, where $t_{\{1,L\}} := (t_1 - t_L)/(R+1)$. We choose R with respect to the nature of the original expression data, such that $R \geq D$.

For each sub-interval, we look for the maximum and minimum values of trajectories. The sets $G(t)$ and $M(t)$ are then expanded as follows. At sub-interval r, gene i is included within either $G(t)$ or $M(t)$ for $t > rt_{\{1,L\}}$ provided that the deviation from the steady state measurement of gene i is greater than a desired threshold, T. In the simulations described in this chapter, T was set in the range of $[0.15, 0.20]$ for normalized expression data. Both R and this threshold can be modified to better reflect the frequency of gene expression measurements for a given biological system. If more complex change detection schemes are preferred, a number of alternative approaches can be adapted for this purpose [57–59].

2.3.3 Modeling and Estimation of Protein Expression

Formulation: Similar to (2.6), we express the protein level $y_i(t)$ as

$$y_i(t) = \sum_{d=1}^{D} \alpha_{id}\varphi_d(t) = \boldsymbol{\varphi}(t)^T \boldsymbol{\alpha}_i. \tag{2.8}$$

Our objective is first to find $\boldsymbol{\alpha}_i$ through the ordinary differential equation (ODE) (2.2) resulting in an estimation of the protein level $y_i(t)$. The calculated $y_i(t)$'s are in turn used to approximate unknown variables associated with the ODE (2.1). One of the challenges of solving non-linear ODEs is that the solution does not usually have a closed form. We propose to transform the non-linear ODE (2.2) into a linear regression problem. To motivate our method of constructing the ODE solution, we consider the first derivative of $y_i(t)$ as

$$y_i'(t) = \boldsymbol{\varphi}'(t)^T \boldsymbol{\alpha}_i,$$

and ODE (2.2) is consequently represented as

$$\boldsymbol{\varphi}'(t)^T \boldsymbol{\alpha}_i = \left(r_i - \sum_{j \in M(t)} \lambda_{ij}^{Prot} x_j(t) \right) x_i(t) - \lambda_i^{Prot} \boldsymbol{\varphi}(t)^T \boldsymbol{\alpha}_i.$$

We rewrite the above equation in the following form

$$r_i x_i(t) - \mathbf{x}_{M(t)}{}^T \boldsymbol{\lambda}_i^R(t) - \mathbf{b}_i(t)^T \boldsymbol{\alpha}_i = 0, \tag{2.9}$$

where $\mathbf{b}_i^T(t) \coloneqq [\lambda_i^{Prot} \boldsymbol{\varphi}(t)^T + \boldsymbol{\varphi}'(t)^T]$ and $\boldsymbol{\lambda}_i^R(t)$ is the column vector with entries λ_{ij}^{Prot}, $\forall j \in M(t)$. The miRNA expressions corresponding to $\boldsymbol{\lambda}_i^R(t)$ are indicated by the vector $\mathbf{x}_{M(t)}$ such that both vectors, $\boldsymbol{\lambda}_i^R(t)$ and $\mathbf{x}_{M(t)}$, have the same index order. For notational convenience, we assume that all entries of $\mathbf{x}_{M(t)}$ are multiplied by $x_i(t)$.

Consider gene expressions at times t_l, $1 \leq l \leq L$. Setting all available gene expressions in equation (2.9), we arrive at

$$\mathbf{A}_i \left(-r_i, \mathbf{z}_i^T, \boldsymbol{\alpha}_i^T\right)^T = 0,$$

where

$$\mathbf{A}_i \coloneqq \begin{pmatrix} x_i(t_1) & (\mathbf{x}_{M(t_1)}{}^T, \mathbf{0}(t_1)^T) & \mathbf{b}_i(t_1)^T \\ x_i(t_2) & (\mathbf{x}_{M(t_2)}{}^T, \mathbf{0}(t_2)^T) & \mathbf{b}_i(t_2)^T \\ \vdots & \vdots & \vdots \\ x_i(t_L) & \mathbf{x}_{M(t_L)}{}^T & \mathbf{b}_i(t_L)^T \end{pmatrix}, \ \mathbf{z}_i \coloneqq \boldsymbol{\lambda}_i^R(t_L),$$

and $\mathbf{0}(t_l)$ is the zero column vector with length $\mathbf{card}(M(t_L)) - \mathbf{card}(M(t_l))$. When the length is zero, we do not consider the vector $\mathbf{0}(t_l)$, e.g., $(\mathbf{x}_{M(t_L)}{}^T, \mathbf{0}(t_L)^T)$ is replaced by $\mathbf{x}_{M(t_L)}{}^T$ in the last row of $\mathbf{A}_i$. Matrix $\mathbf{A}_i$ has L rows and $\mathbf{card}(M(t_L)) + D + 1$ columns. Given that r_i is positive, we normalize $\left(-r_i, \mathbf{z}_i^T, \boldsymbol{\alpha}_i^T\right)^T$ with respect to r_i and represent the normalized vector as $\left(-1, \mathbf{z}_i^T, \boldsymbol{\alpha}_i^T\right)^T$, acknowledging abuse of notation. Given λ_i^{Prot} and $M(t)$, matrix $\mathbf{A}_i$ is completely determined.

Algorithm: We need to solve the linear system model

$$\mathbf{A}_i \begin{pmatrix} -1 \\ \mathbf{z}_i \\ \boldsymbol{\alpha}_i \end{pmatrix} = 0 \tag{2.10}$$

for $\mathbf{z}_i$ and $\boldsymbol{\alpha}_i$ when matrix $\mathbf{A}_i$ is determined. For identifiability of $\mathbf{z}_i$ and $\boldsymbol{\alpha}_i$, we require that $L \geq \mathbf{card}(M(t_L)) + D$, that is the number of equations is no smaller than the number of

unknown parameters. However the sparsity in $\mathbf{z}_i$, given that only a small number of miRNAs typically act on a common gene [60], reduces the number of required equations.

To account for measurement noise and encourage $\mathbf{z}_i$ to be sparse, we will minimize the 2-norm error described in (2.10) with 1-norm regularization $\|\mathbf{z}_i\|_1$. Furthermore, we adopt the analogous roughness penalty $\boldsymbol{\alpha}_i^T \mathbf{K} \boldsymbol{\alpha}_i$ as used in (P1). Thus, we propose to obtain the ODE (2.2) solution with the following convex optimization

$$\text{(P2)} \qquad \min_{\{\mathbf{z}_i, \boldsymbol{\alpha}_i\}} \left\| \mathbf{A}_i \begin{pmatrix} -1 \\ \mathbf{z}_i \\ \boldsymbol{\alpha}_i \end{pmatrix} \right\|_2 + \gamma_z \|\mathbf{z}_i\|_1 + \gamma_\alpha \boldsymbol{\alpha}_i^T \mathbf{K} \boldsymbol{\alpha}_i,$$

$$\text{subject to} \quad \mathbf{z}_i \geq \mathbf{0}$$

$$(\mathbf{x}_{M(t_l)}{}^T, \mathbf{0}(t_l)^T)\mathbf{z}_i \leq x_i(t) \quad \forall 1 \leq l \leq L$$

where γ_z and γ_α are chosen using cross validation. The second constraint ensures that the total rate of translation, $r_i - h_i(\mathcal{X}_{M(t)})$, is not negative. Due to the convex nature of this problem, it can be quickly solved for large gene datasets. This recovery of protein expression is dependent on prior knowledge of individual protein degradation rates, λ_i^{Prot}. In the absence of this experimental data, we can fix the value of λ_i^{Prot} to 1 for the entire system and still achieve accurate network reconstruction as shown in subsequent sections.

2.3.4 Gene Regulatory Inference

Formulation: The model given by ODEs (2.1) and (2.2) describes the evolution of RNA and protein expressions provided that we know all the regulatory parameters, e.g., a_{ij}, b_{ij}, and τ_i. Coefficients a_{ij} and b_{ij} are difficult to experimentally determine and it is currently infeasible to carry out the relevant measurements simultaneously for a complex system with a large number of genes and gene products under consideration. Our goal is to estimate these coefficients so that the ODE models can be temporally fitted to large gene expression data. Specifically, we will use the previously described estimations of protein and RNA expression to approximate a_{ij} and b_{ij}, and to infer a regulatory network map.

To improve the reliability of the inferred network, we take into account time-dependent changes in gene levels and construct a set of equations accordingly. This is an important departure from standard steady state treatments. In this scenario, we first assume that the non-perturbed system is in an initial steady state, where RNA and protein levels are near constant (i.e., $dx_i(t)/dt = dy_i(t)/dt \simeq 0$). As previously mentioned, the perturbation of protein-encoding gene $x_i^p(t_1)$ first leads to fluctuations in the expression levels of genes in its immediate regulatory network. Genes that have exited a steady-state expression profile at any time up to t, $G(t)$ and $M(t)$, expand to contain greater numbers of genes that interact to form a putative regulatory network.

Considering changes in gene levels $x_i(t)$ at time t_l, $1 \leq l \leq L$, with the exception of $x_i^p(t_1)$, the term $\tau_i f_i(\mathcal{Y}_{G(t_l)})$ in equation (2.1) can be rewritten as follows

$$\tau_i f_i(\mathcal{Y}_{G(t_l)}) = \frac{\tau_i a_{i0} + \sum_{j=1}^{N(t_l)} \tau_i a_{ij} \prod_{k \in S_{ij}(t_l)} y_k(t_l)}{1 + \sum_{j=1}^{N(t_l)} b_{ij} \prod_{k \in S_{ij}(t_l)} y_k(t_l)} := \frac{\mathbf{p}_i^T(t_l)\mathbf{a}_i}{\mathbf{p}_i^T(t_l)\mathbf{b}_i}, \tag{2.11}$$

where $\mathbf{a}_i$ is a vector with $(j+1)$th entry $\tau_i a_{ij}$, $0 \leq j \leq N(t_L)$. The $(j+1)$th element of vector $\mathbf{p}_i(t_l)$ is described by $\prod_{k \in S_{ij}(t_l)} y_k(t_l)$ when $0 \leq j \leq N(t_l)$ and zero for $N(t_l)+1 \leq j \leq N(t_L)$. Vector $\mathbf{b}_i$ is defined such that the first entry is 1 and $(j+1)$th, $1 \leq j \leq N(t_L)$, is b_{ij}.

Remark 2.1. *Given that $y_i(t)$s are normalized with respect to r_i, a_{ij} and b_{ij} include the multiplier term $\prod_{k \in S_{ij}(t_l)} r_k$ so that the normalization can be vanished. Similarly, τ_i can be absorbed into the coefficients a_{ij}, where we assume $\tau_i < 1$ to maintain the algorithm constraint $0 \leq a_i \leq b_i$.*

We also represent

$$\left(\lambda_i^{RNA} + \sum_{j \in M(t_l)} \lambda_{ij}^{RNA} x_j(t_l) \right) x_i(t_l) + \left. \frac{dx_i(t)}{dt} \right|_{t=t_l} := \mathbf{u}_i^T(t_l)\boldsymbol{\lambda}_i, \tag{2.12}$$

in which $\mathbf{u}_i(t_l)$ and $\boldsymbol{\lambda}_i$ are defined as follows. First and second entries of vector $\mathbf{u}_i(t_l)$ are $dx_i(t)/dt|_{t=t_l}$ and $x_i(t_l)$, respectively. The remaining entries are $x_j(t_l)x_i(t_l)$, $j \in M(t_l)$. Making the same arrangement of array as $\mathbf{u}_i(t_l)$, vector $\boldsymbol{\lambda}_i$ is determined by first entry 1, second entry λ_i^{RNA}, and subsequent entries λ_{ij}^{RNA}, $j \in M(t_l)$.

Using (2.11)–(2.12), equation (2.1) can be reformulated as

$$\Omega_l(\mathbf{a}_i, \mathbf{b}_i, \boldsymbol{\lambda}_i) := \mathbf{p}_i^T(t_l)\mathbf{a}_i - \mathbf{u}_i^T(t_l)\boldsymbol{\lambda}_i\mathbf{b}_i^T\mathbf{p}_i(t_l) = 0. \tag{2.13}$$

Algorithm: We need to solve the non-convex problem

$$\begin{aligned} \text{(P3)} \qquad & \min_{\{\mathbf{a}_i, \mathbf{b}_i, \boldsymbol{\lambda}_i\}} \Gamma(\mathbf{a}_i, \mathbf{b}_i, \boldsymbol{\lambda}_i) \\ & \text{subject to} \quad 0 \le \mathbf{a}_i \le \mathbf{b}_i, 0 \le \boldsymbol{\lambda}_i \\ & \qquad\qquad \mathbf{b}_i(1) = 1 \\ & \qquad\qquad \boldsymbol{\lambda}_i(1) = 1, \boldsymbol{\lambda}_i(2) = \lambda_i^{RNA} \end{aligned}$$

with

$$\Gamma(\mathbf{a}_i, \mathbf{b}_i, \boldsymbol{\lambda}_i) := \sum_{l=1}^{L} \Omega_l(\mathbf{a}_i, \mathbf{b}_i, \boldsymbol{\lambda}_i)^2 + \frac{\gamma_1}{2}\left(\|\boldsymbol{\lambda}_i\|_2^2 + \|\mathbf{b}_i\|_2^2\right) + \gamma_2\|\mathbf{b}_i\|_1 + \gamma_3\|\boldsymbol{\lambda}_i\|_1.$$

The first term in the above equation follows from (2.13). The second term associated with $\gamma_1/2$ motivates grouping effect among variables $\mathbf{b}_i$ and $\boldsymbol{\lambda}_i$ [61,62]. Due to the assumption that each gene has only a few regulators, 1-norm regularizations are considered to encourage sparse solutions. Note that in the absence of miRNAs (all $\lambda_{ij}^{RNA} = 0$), terms $\|\boldsymbol{\lambda}_i\|_2$ and $\|\boldsymbol{\lambda}_i\|_1$ are no longer needed.

Non-convex optimizations are generally hard to solve in a reasonable time. Hence, we seek to identify a special treatment that reduces the computational complexity and provides desired solutions. Optimization (P3) is convex in $\{\mathbf{a}_i, \mathbf{b}_i\}$ for fixed $\boldsymbol{\lambda}_i$ and vice versa, and therefore the problem is bi-convex and can be solved using a variation of the alternating-direction method of multipliers (ADMM) which cycles over two groups of variables [63] (see Appendix A). Here, given the absence of dual variables, ADMM is reduced to simple alternating minimization. The proposed solver entails an iterative procedure compromising two steps per iteration $k = 1, 2, \ldots$

This iterative procedure implements a block coordinate descent method [64]. At each minimization, the variables that are not being updated are treated as fixed and are replaced with their most updated values. Then the iteration alternates between two sets of variables, $\{\mathbf{b}_i, \mathbf{a}_i\}$ and $\boldsymbol{\lambda}_i$.

Algorithm 1 : Gene regulatory network inference

input $\mathbf{a}_i, \mathbf{b}_i, \boldsymbol{\lambda}_i$

initialize $\mathbf{a}_i[0], \mathbf{b}_i[0]$, **and** $\boldsymbol{\lambda}_i[0]$ **at random with respect to**

$$\mathbf{b}_i(1) = 1, \boldsymbol{\lambda}_i(1) = 1, \text{and } \boldsymbol{\lambda}_i(2) = \lambda_i^{RNA}$$

for $k = 0, 1, \ldots$ **do**

[S1] Update primal variables $\mathbf{a}_i$ **and** $\mathbf{b}_i$**:**

$$\{\mathbf{a}_i[k+1], \mathbf{b}_i[k+1]\} = \arg \min_{\{\mathbf{b}_i, \mathbf{a}_i\}} \Gamma(\mathbf{a}_i, \mathbf{b}_i, \boldsymbol{\lambda}_i[k])$$

$$\text{subject to} \quad \mathbf{0} \leq \mathbf{a}_i \leq \mathbf{b}_i$$

$$\mathbf{b}_i(1) = 1$$

[S2] Update primal variable $\boldsymbol{\lambda}_i$**:**

$$\boldsymbol{\lambda}_i[k+1] = \arg \min_{\boldsymbol{\lambda}_i} \Gamma(\mathbf{a}_i[k], \mathbf{b}_i[k], \boldsymbol{\lambda}_i)$$

$$\text{subject to} \quad \boldsymbol{\lambda}_i \geq \mathbf{0}$$

$$\boldsymbol{\lambda}_i(1) = 1, \boldsymbol{\lambda}_i(2) = \lambda_i^{RNA}$$

end for

return $\mathbf{a}_i, \mathbf{b}_i, \boldsymbol{\lambda}_i$

One difficulty with the proposed solver is that it may result in stationary points which are not necessarily globally optimal. This occurs since optimization (P3) is not convex in $\{\mathbf{b}_i, \mathbf{a}_i, \boldsymbol{\lambda}_i\}$. Motivated by the proposition 1 in [65], the next theorem offers a global optimality certificate upon the convergence of the solver.

Theorem 2.1. *Let* $\{\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i\}$ *be a stationary point of (P3). If*

$$\left\| \sum_{l=1}^{L} \Omega_l(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i) \mathbf{u}_i(t_l) \mathbf{p}_i^T(t_l) \right\| \leq \frac{\gamma_1}{2}, \tag{2.14}$$

then $\{\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i\}$ *is the globally optimal solution of (P3).*

Proof. See Appendix A. □

Remark 2.2. *For non-convex problems, ADMM offers no convergence guarantees. Nevertheless, there are evidences in the literature that show empirical convergence of ADMM, particularly when the non-convex exhibits specific structures. For example in our scenario, problem (P3) is bi-convex and admits unique closed form solutions for sub-problems* ***[S1]*** *and* ***[S2]****. This observation along with desired properties, Theorem 4.5 and 4.9 in [66], are indeed a sufficient case for successful convergence. A formal proof of convergence is beyond the scope of this work.*

Algorithm 1 is intended for the case in which the RNA degradation rates, λ_i^{RNA}, are available. However, experimentally measuring λ_i^{RNA} is a difficult task. We offer a simple modification to the algorithm so that network inference can be still obtained without prior knowledge of RNA degradation rates.

For simplicity of explanation, we can first remove miRNAs from our model. ODE (2.1) can then be rewritten as

$$\Omega_l(\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i) \coloneqq \mathbf{p}_i^T(t_l) \left(\mathbf{a}_i - \mathbf{b}_i \frac{dx_i(t_l)}{dt} - \mathbf{c}_i x_i(t_l) \right) = 0, \tag{2.15}$$

and $\mathbf{c}_i \coloneqq \lambda_i^{RNA} \mathbf{b}_i$. Employing the above reformulation, unknown variables $\mathbf{a}_i$, $\mathbf{b}_i$, and $\mathbf{c}_i$ are estimated through the following convex optimization

$$\begin{aligned} \text{(P4)} \quad \arg \min_{\{\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i\}} & \sum_{l=1}^{L} \Omega_l(\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i) + \gamma_2(\|\mathbf{b}_i\|_1 + \|\mathbf{c}_i\|_1) \\ \text{subject to} \quad & \mathbf{0} \leq \mathbf{a}_i \\ & \mathbf{a}_i \leq \mathbf{b}_i \\ & \lambda_{min} \mathbf{b}_i \leq \mathbf{c}_i \leq \lambda_{max} \mathbf{b}_i, \end{aligned} \tag{2.16}$$

where λ_{min} and λ_{max} specify an lower and upper bound for λ_i^{RNA}, respectively. Variable $\mathbf{c}_i$ is introduced to remove λ_i^{RNA} from our optimization. However, the new variable expands

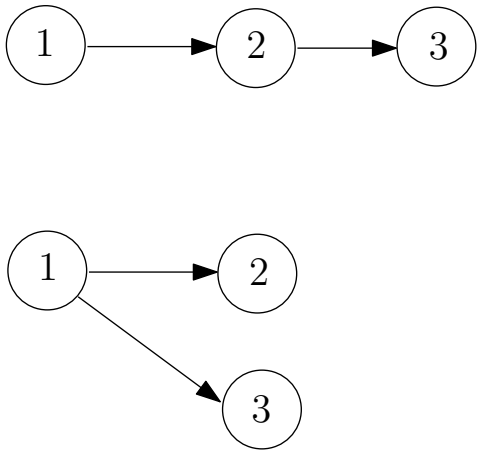

Figure 2.4: Map of two gene regulatory networks with similar gene levels

the feasible set of solutions, which might create an answer different from the true value. To reduce this effect, we add constraint (2.16) to (P4) to tighten the feasible set of solutions. Given that $\lambda_i^{RNA}/\tau_i \geq 1$, we can take on the additional constraint $\mathbf{a}_i \leq \mathbf{c}_i$. In the subsequent simulations, λ_{min} is in the near-zero range $[0.001, 0.01]$, and λ_{max} is selected in the range $[0.1, 1]$. It is straightforward to generalize the introduced approach within the framework of (P3). Derivations are removed to avoid repetition in the chapter.

2.4 Identifiability of Gene Regulatory Networks

New RNA sequencing workflows in conjunction with gene expression data are dramatically emerging. However, they must be evaluated to determine which of datasets are of value for the gene network inference. In fact, we need to understand what system knowledge can be obtained from gene expression data alone? What are limitations in gene expression data and current biological information for inferring gene networks. To clarify the difficulty here, consider Figure 2.4. In this example, two gene regulatory networks (top and bottom) with a total number of three genes are depicted. We assume that gene 1 is perturbed and gene 2 (also 3) approximately corresponds to similar expression levels in both networks. Then, theses two networks may not be distinguishable, regardless of any type of inference procedure.

To shed light on the above questions, a formal identifiability method is introduced to inves-

tigate which of unknown parameters in gene regulatory networks can be estimated using gene expression data. In particular, we utilize results from differential algebra techniques to perform the structural identifiabilty analysis of our system model.

2.4.1 Structural Identifiabilty Definition

The structural identifiability analysis of nonlinear dynamical models has been well studied [67–71]. The focus in this area is to employ differential algebra methods coupled with Gröbner Bases, Lie derivatives and the Taylor series in order to find globally identifiable parameters of nonlinear dynamical models. Such models are usually assumed to be in the form of

$$\begin{aligned} \mathbf{s}'(t,\mathbf{p}) &= \mathbf{f}(\mathbf{s}(t,\mathbf{p}),\mathbf{u}(t),\mathbf{p}),\\ \mathbf{w}(t,\mathbf{p}) &= \mathbf{g}(\mathbf{s}(t,\mathbf{p}),\mathbf{p}),\\ \mathbf{s}(0,\mathbf{p}) &= \mathbf{s}_0(\mathbf{p}), \end{aligned} \tag{2.17}$$

where $\mathbf{s}(t,\mathbf{p}) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^s$, and $\mathbf{w}(t,\mathbf{p}) \in \mathbb{R}^m$ are the state variables, the input functions and the observation functions, respectively. The entries of vectors $\mathbf{f}$ and $\mathbf{g}$ are polynomials or fractions of polynomial in $\mathbf{s}$, $\mathbf{u}$, and the parameter vector $\mathbf{p}$. Here, our goal is to deduce whether $\mathbf{p}$ is structurally identifiable or not. Structural identifiability is defined as follows [69]:

Definition 2.1. *Let $\mathbf{p} \in \Omega \subseteq \mathbb{R}^d$ and $\mathbf{s}_0(\mathbf{p})$ be the initial condition in (2.17). Consider the map $\mathcal{M}_{\mathbf{p},\mathbf{s}_0(\cdot)} : \mathbf{u}(\cdot) \mapsto \mathbf{w}(\cdot,\mathbf{p})$. The parameter vectors $\mathbf{p}$ and $\mathbf{p}^*$ are said to be equivalent, denoted by $\mathbf{p} \sim \mathbf{p}^*$, if and only if $\mathcal{M}_{\mathbf{p},\mathbf{s}_0(\cdot)}(\mathbf{u}) = \mathcal{M}_{\mathbf{p}^*,\mathbf{s}_0(\cdot)}(\mathbf{u})$ for all $\mathbf{u} \in \mathcal{U}$ where $\mathcal{U}$ is a given class of input functions. Then, we have*

1. *The parameter vector $\mathbf{p}$ is said to be globally structurally identifiable if for almost all $\mathbf{p}^* \in \Omega$, $\mathbf{p} \sim \mathbf{p}^*$ implies $\mathbf{p} = \mathbf{p}^*$.*

2. *The parameter vector $\mathbf{p}$ is said to be locally structurally identifiable if there exists an open set $W \subset \Omega$ (with respect to the Euclidean topology) such that for almost all $\mathbf{p}^* \in W$, $\mathbf{p} \sim \mathbf{p}^*$ implies $\mathbf{p} = \mathbf{p}^*$.*

3. *The parameter vector $\mathbf{p}$ is said to be structurally unidentifiable if $\mathbf{p}$ is not locally structurally identifiable.*

2.4.2 Structural Identifiabilty Analysis

To determine identifiable parameters, Ollivier in [68] eliminates non-observable state variables, $\mathbf{s}$, to obtain relations among inputs, outputs, and unknown parameters. Such input-output relations are polynomials in the variables $\{\mathbf{u}, \mathbf{u}', \mathbf{u}'', \ldots, \mathbf{w}, \mathbf{w}', \mathbf{w}'', \ldots\}$ with rational coefficients in the parameter vector $\mathbf{p}$. By analysis of the coefficients of input-output polynomials, it is possible to establish the identifiability of $\mathbf{p}$.

The input-output polynomials can be obtained from the *characteristic set* [67]. The characteristic set is a "minimal" set of differential polynomials that generate the same differential ideal as the ideal generated by (2.17); see [72]. The first m elements of the characteristic set forms the input-output polynomials

$$A_1(\mathbf{u}, \mathbf{w}, \mathbf{p}), \ldots, A_m(\mathbf{u}, \mathbf{w}, \mathbf{p}). \tag{2.18}$$

Input-output polynomials can be represented as $A_j(\mathbf{u}, \mathbf{w}, \mathbf{p}) = \sum_i e_i(\mathbf{p}) B_i(\mathbf{u}, \mathbf{w})$, where $e_i(\mathbf{p})$ is a rational function in $\mathbf{p}$ and $B_i(\mathbf{u}, \mathbf{w})$ is a monomial function in $\{\mathbf{u}, \mathbf{u}', \mathbf{u}'', \ldots, \mathbf{w}, \mathbf{w}', \mathbf{w}'', \ldots\}$. To perform the structural identifiabilty, let $\mathbf{p}^*$ be an arbitrary point in parameter space. We then set $A_j(\mathbf{u}, \mathbf{w}, \mathbf{p}) = A_j(\mathbf{u}, \mathbf{w}, \mathbf{p}^*)$, $1 \leq j \leq m$, which leads to $\sum_i (e_i(\mathbf{p}) - e_i(\mathbf{p}^*)) B_i(\mathbf{u}, \mathbf{w}) = 0$. Since the characteristic set is constructed based on a prime ideal [73], $B_i(\mathbf{u}, \mathbf{w})$ are linearly independent and globally identifiable. Therefore, our identifiability problem is reduced to injectivity of the map from $\mathbf{p}$ to the coefficients of the input-output polynomials, $e_i(\mathbf{p})$. Specifically, model (2.17) is structurally identifiable if and only if

$$e_i(\mathbf{p}) = e_i(\mathbf{p}^*) \tag{2.19}$$

for all i imply that $\mathbf{p} = \mathbf{p}^*$ for any arbitrary $\mathbf{p}^*$. The model is locally structurally identifiable if and only if there are finite distinct solutions for $\mathbf{p}$. The model is structurally unidentifiable if and only if there are infinite solutions for $\mathbf{p}$. The solutions of (2.19) are usually computed by finding a Gröbner basis and using elimination [72].

We summarize the structural identifiabilty analysis as three steps:

1. Find input-output polynomials, $A_j(\mathbf{u}, \mathbf{w}, \mathbf{p})$, based on the characteristic set.

2. Recover the coefficients of input-output polynomials that are functions of parameters, $e_i(\mathbf{p})$.

3. If the coefficient map from $\mathbf{p}$ to $e_i(\mathbf{p})$ for all i is 1-to-1, our model is identifiable.

A detailed description of the above procedures can be found in [69, 72]. In the following, we present an example to illustrate the introduced steps.

Example: Consider the dynamical model [74]

$$\begin{aligned} s_1' &= p_1 s_1^2 + p_2 s_1 s_2, \\ s_2' &= p_3 s_1^2 + p_4 s_1 s_2, \\ w_1 &= s_1, \end{aligned}$$

which corresponds to the input-output polynomial

$$-ww'' + w'^2 + (p_2 p_3 - p_1 p_4) w^4 + (p_1 + p_4) w' w^2.$$

Thus, the coefficients of the input-output polynomial are

$$-1, 1, p_2 p_3 - p_1 p_4, p_1 + p_4.$$

We conclude the model in this example is not identifiable since there are 4 parameters and only 2 coefficients involving the parameters.

2.4.3 Network Identifiability

The aforementioned model (2.17) incorporates the ODEs of the form (2.1) and (2.2). More precisely, these ODES can be represented by

$$\begin{aligned} \mathbf{x}'(t, \mathbf{p}) &= \mathbf{f}_1(\mathbf{x}(t, \mathbf{p}), \mathbf{y}(t, \mathbf{p}), \mathbf{p}), \\ \mathbf{y}'(t, \mathbf{p}) &= \mathbf{f}_2(\mathbf{x}(t, \mathbf{p}), \mathbf{y}(t, \mathbf{p}), \mathbf{p}), \end{aligned} \tag{2.20}$$

where vector $\mathbf{p}$ contains unknown parameters associated with ODEs (2.1) and (2.2), vector $\mathbf{x}(t, \mathbf{p})$ genes levels at time t, and vector $\mathbf{y}(t, \mathbf{p})$ proteins levels at time t. The vector functions

$\mathbf{f}_1$ and $\mathbf{f}_2$ are fractions of polynomial (or polynomials) that follow our system model. Let us define the new vector

$$\mathbf{s} := \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}. \tag{2.21}$$

Consequently, ODEs (2.1) and (2.2) can be shown as

$$\begin{aligned} \mathbf{s}'(t, \mathbf{p}) &= \mathbf{f}(\mathbf{s}(t, \mathbf{p}), \mathbf{p}), \\ \mathbf{w}(t, \mathbf{p}) &= \mathbf{g}(\mathbf{s}(t, \mathbf{p}), \mathbf{p}), \end{aligned} \tag{2.22}$$

with $\mathbf{f}(\cdot) = (\mathbf{f}_1(\cdot)^T, \mathbf{f}_2(\cdot)^T)^T$ and $\mathbf{g}(\cdot) = \mathbf{x}$. As it appears, the above model preserves the structure (2.17), which presents an opportunity to utilize structural identifiability and determine whether gene regulatory networks are globally (or locally) identifiable. However, gene regulatory networks are sparse since only a few genes can affect one gene. Therefore, the structural identifiability must be derived for sparse networks. In fact, it is unlikely to identify such networks without considering sparsity.

We extend the structural identifiability to the case in which the parameter vector $\mathbf{p}$ has only k non-zero entries. In this scenario, we reduce the parameter space to the k-sparse vectors and seek for the injectivity of the coefficient map. More precisely, we have

Theorem 2.2. *Assume k elements from the parameter vector $\mathbf{p}$ are non-zero. If the coefficient map from the space of k-sparse parameter vector to the coefficients of input-output polynomials, $e_i(\mathbf{p})$, is 1-to-1, then the network is globally identifiable.*

To clarify the above theorem, let us assume $\mathbf{p} \in \mathbb{R}^d$ with k non-zero entries. We then have $\frac{d!}{k!(d-k)!}$ possibilities for k-sparse vectors, each corresponding to a subspace in the parameter space, donated by S_l, $1 \leq l \leq \frac{d!}{k!(d-k)!}$. The network is globally identifiable if (i) the map from any S_l to the coefficients of the input-output polynomials is 1-to-1 and (ii) the map from each S_l results in a distinct set of coefficients. Conditions (i) and (ii) guarantee that the coefficient map in Theorem 2.2 is 1-to-1. Note that the network identifiability is performed similar to the previous subsection, but, the injectivity of the coefficient map is analyzed with respect to the space of k-sparse parameter vector.

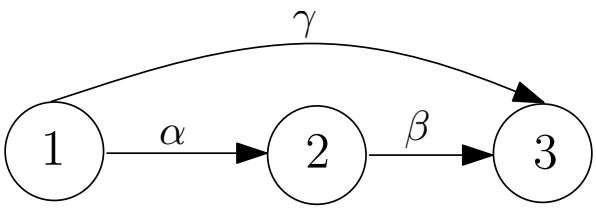

Figure 2.5: Map of gene regulatory network with a linear dynamical model. Parameters α, β, and γ exhibits the relations among genes according to the dynamical model.

Remark 2.3. *In the case that only condition (ii) holds, we are still able to identify the network topology. In other words, condition (ii) enables us to determine support of the parameter vector* **p**.

The following examples are provided to demonstrate the network identifiability analysis. In the first example, the network is identifiable while in the second example, the network is not identifiable unless the parameter space is limited to the k-sparse vectors.

Example: Consider the gene network depicted in Figure 2.5 with the following dynamical model

$$x_1' = 1 - x_1,$$
$$x_2' = \alpha x_1 - x_2,$$
$$x_3' = \gamma x_1 + \beta x_2 - x_3,$$

where the gene levels x_1, x_2, and x_3 are measured. For simplicity, we have assumed that our dynamical model is linear and has a few unknown parameters (the network is sparse). The input-output polynomials are

$$x_1' - 1 + x_1 \quad , x_2' - \alpha x_1 + x_2 \quad , x_3' - \gamma x_1 - \beta x_2 + x_3,$$

which correspond to the coefficients (involving parameters)

$$\alpha, \beta, \gamma.$$

Therefore, the network is identifiable since we have 3 parameters and 3 distinct coefficients.

Example: Consider the following coefficients that are obtained based on input-output polynomials of a network:

$$p_1 + p_5 + 4p_6 + 2p_7,\ p_2 + 2p_5 + 3p_6 + p_7,\ p_3 + 3p_5 + 2p_6 + 4p_7,\ p_4 + 4p_5 + p_6 + 3p_7.$$

The above coefficients can be represented as $\mathbf{Ap}$ where

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 4 & 2 \\ 0 & 1 & 0 & 0 & 2 & 3 & 1 \\ 0 & 0 & 1 & 0 & 3 & 2 & 4 \\ 0 & 0 & 0 & 1 & 4 & 1 & 3 \end{pmatrix}, \quad \mathbf{p} = (p_1, p_2, p_3, p_4, p_5, p_6, p_7)^T.$$

The network is not identifiable since the number of parameters is greater than the number of coefficients. Let us assume there are only two non-zero parameters. To invistigate the injectivity of the coefficient map, let $\mathbf{p}*$ be an arbitrary point in parameter space and $\mathbf{Ap} = \mathbf{Ap}*$. We then have $\mathbf{A}(\mathbf{p} - \mathbf{p}*) = 0$. Since vector $(\mathbf{p} - \mathbf{p}*)$ contains at most four non-zero entries and any four columns of $\mathbf{A}$ are linearly independent, we arrive at $\mathbf{p} = \mathbf{p}*$. We conclude that the network is globally identifiable when at most two parameters are non-zero.

2.5 Simulations

2.5.1 Small Gene Network with Prior Knowledge of Degradation Rates

To demonstrate the proposed time-series approach, we consider the three-gene network described by the following systems of ODEs for gene expression

$$\begin{aligned} \frac{dx_1(t)}{dt} =& \frac{0.1 + 0.05y_1(t)y_2(t) + 0.025y_1(t)y_3(t)}{1 + 0.1y_1(t) + 10y_3(t) + 0.05y_1(t)y_2(t) + 0.025y_1(t)y_3(t)} \\ & - 0.1x_1(t), \\ \frac{dx_2(t)}{dt} =& \frac{0.1 + 0.1y_1(t) + 0.1y_1(t)y_2(t)}{1 + 0.1y_1(t) + 0.1y_1(t)y_2(t) + 10y_1(t)y_3(t)} - 0.1x_2(t), \\ \frac{dx_3(t)}{dt} =& \frac{0.1 + 0.1y_2(t)}{1 + 0.1y_2(t) + .1y_3(t)} - 0.1x_3(t), \end{aligned} \tag{2.23}$$

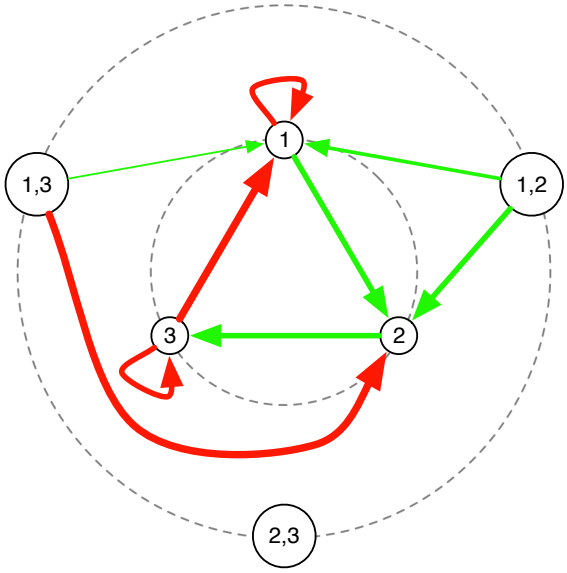

Figure 2.6: Map of gene regulatory network described by equations (2.23) and (2.24). First-order (single) and and second-order (combined) regulators are depicted in concentric circles. Green arrows specify gene activation and red arrows specify gene repression. The relative magnitudes of activation and repression are roughly represented by arrow thickness.

and the following system of ODEs for protein expression

$$\begin{aligned}\frac{dy_1(t)}{dt} &= x_1(t) - 0.5y_1(t),\\ \frac{dy_2(t)}{dt} &= 2x_2(t) - 0.5y_2(t),\\ \frac{dy_3(t)}{dt} &= x_3(t) - 0.5y_3(t).\end{aligned} \tag{2.24}$$

The above toy model, visualized in Figure 2.6, is provided to better explain our algorithms. Although a small network is examined, many of the same qualitative characteristics of large network are investigated in this example. The explicit system of ODEs, describing the kinetics of the system [75], allows us to generate samples to fit our model and to also compare recovered solutions with the ground truth. This model also incorporates complex modes of regulation, including self-regulation and combined regulators.

To generate data, arbitrary initial conditions are assigned to ODEs (2.23) and (2.24) and the system is allowed to resolve to a steady state. To perturb this steady state, the expression

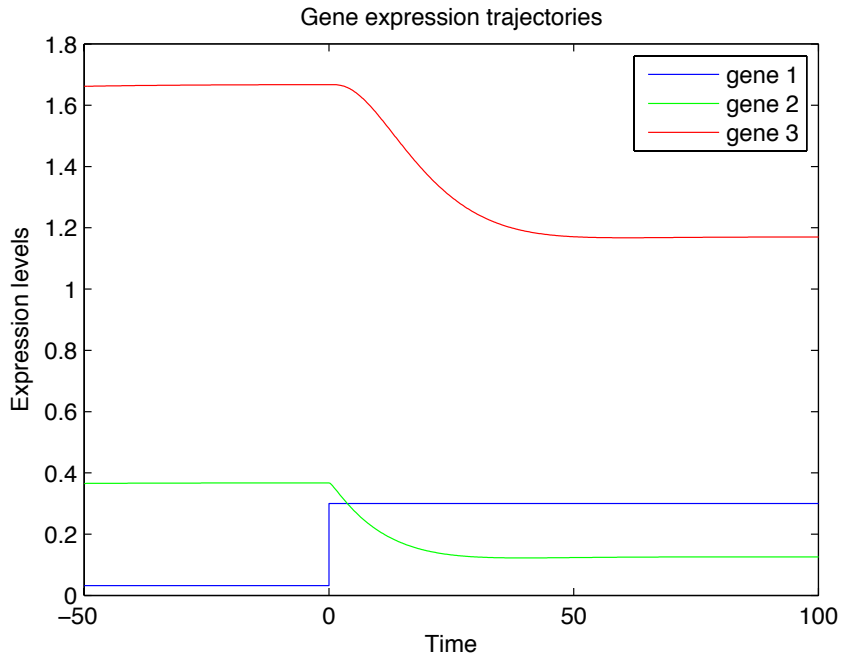

Figure 2.7: Gene expression trajectories (unnormalized) before and during the imposed perturbation. The system is in steady state before time 0. Gene 1 is artificially perturbed at time zero, leading to changes in gene expression levels. A new steady state is eventually achieved at approximately time 50. We sample expression levels between time 0 (the starting point of perturbation) and 50 (the new steady state) and use them as data in our algorithm.

level of gene 1, $x_1(t)$, is artificially fixed to 0.3, leading to fluctuations in the expression levels of other genes. Figure 2.7 illustrates expression trajectories before and during the perturbation.

We collect 12 samples from each gene expression level. The samples are chosen uniformly from time interval $[0, 50]$. Points 0 and 50 specify the times at which the perturbation starts and the system reaches a new steady state, respectively. Using these sampled data, we solve optimization (P2) to effectively recover protein expressions as shown in Figure 2.8.

We finally examine Algorithm 1, (P3), for the goal of network recovery. In this scenario, our target is to estimate vectors $\mathbf{a}_i$ and $\mathbf{b}_i$. We assume that the degradation rates are known in advance and therefore, since the system does not contain any miRNA in this particular example, $\boldsymbol{\lambda}_i$ is completely at hand. Let us consider gene 3 where the true value of $\mathbf{a}_3 = (0.1, 0, 0.1, 0, 0, 0, 0)$ and $\mathbf{b}_3 = (1, 0, 0.1, 0.1, 0, 0, 0)$. Vectors $\mathbf{a}_3$ and $\mathbf{b}_3$ are indexed with regard

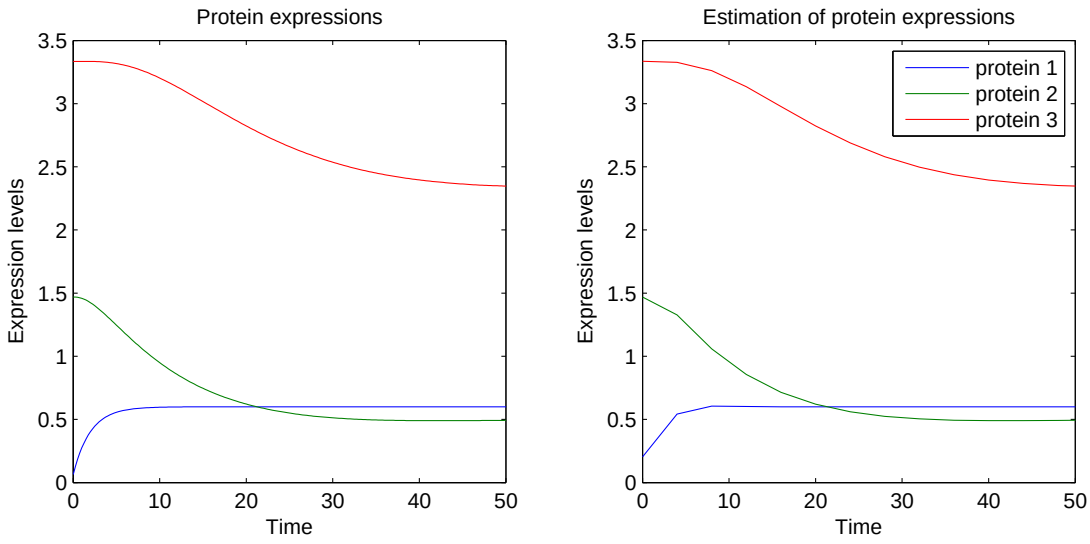

Figure 2.8: Exact protein expression curves derived from model ODEs (2.23) and (2.24) (left), and their recovered estimations using 12 unnormalized timepoint samples via (P2) (right). For convenience of graphical comparison, the values of r_i were drawn from the system equations. Protein expression is otherwise normalized with respect to r_i, but this would result in a transformed scale for this qualitative comparison.

to

$$\mathbf{p}_3(t_l) = (1, y_1(t_l), y_2(t_l), y_3(t_l), y_1(t_l)y_2(t_l), y_1(t_l)y_3(t_l), y_2(t_l)y_3(t_l)).$$

Applying our method, we obtain $\mathbf{a}_3 \simeq (0.1, 0, 0.083, 0, 0, 0, 0)$ and $\mathbf{b}_3 \simeq (1, 0, 0.083, 0.08, 0, 0, 0)$. Table 2.1 demonstrates that as the sampling frequency increases, we attain more accurate approximations. Furthermore, it can be seen that the estimations achieve similar accuracy after a small number of samples.

Employing the aforementioned single perturbation, we are only able to recover the strongest edge of gene 2, $\mathbf{b}_2(6) = 10$. The difficulty here is due to the sharp change in y_1 (Figure 2.8), which provides us with a minimal amount of dynamic information. y_1 near-instantaneously switches between two steady-state levels of expression, resulting in less accurate recovery of the underlying dynamics. However, expression patterns in perturbed biological settings tend to be

Table 2.1: Inference of binding coefficients describing energies of regulator complex-promoter interactions based on number of samples.

# of Samples	Variables	Estimated vector entries						
8	$\mathbf{a}_3$	0.097	0	0.082	0	0	0	0
	$\mathbf{b}_3$	1	0	0.082	0.06	0	0	0
16	$\mathbf{a}_3$	0.1	0	0.093	0	0	0	0
	$\mathbf{b}_3$	1	0	0.093	0.089	0	0	0
24	$\mathbf{a}_3$	0.1	0	0.1	0	0	0	0
	$\mathbf{b}_3$	1	0	0.1	0.092	0	0	0

more dynamic and are unlikely to contain this type of expression pattern. In this example, the removal of sharp instantaneous expression changes leads to complete recovery of the gene regulatory network.

Remark 2.4. *The recovery of regulatory networks using this proposed approach is tightly associated with the presence of dynamic changes in gene expression. These changes can provide us with a certain amount of information which predominantly specifies the accuracy of estimation. The achievable accuracy depends on many factors such as nonlinearity in changes or similarity in the range of changes.*

2.5.2 Medium (10-gene) Simulated Network With Noise

We extend our approach to simulated networks of 10 genes, generated as part of the DREAM4 *in silico* network inference challenge [76]. Each network dataset includes a simulated time series of gene expression in response to five chemical perturbations, along with single steady-state expression levels for wild-type, knockdown, knockout, and multifactorial perturbations. These datasets also simulate internal network noise and incorporate measurement noise. We use these data to assess the robustness of our approach in a non-ideal setup.

Our approach is geared towards precise genetic and chemical perturbations, while these datasets simulate chemicals that are non-specific in their interactions. To place us at further

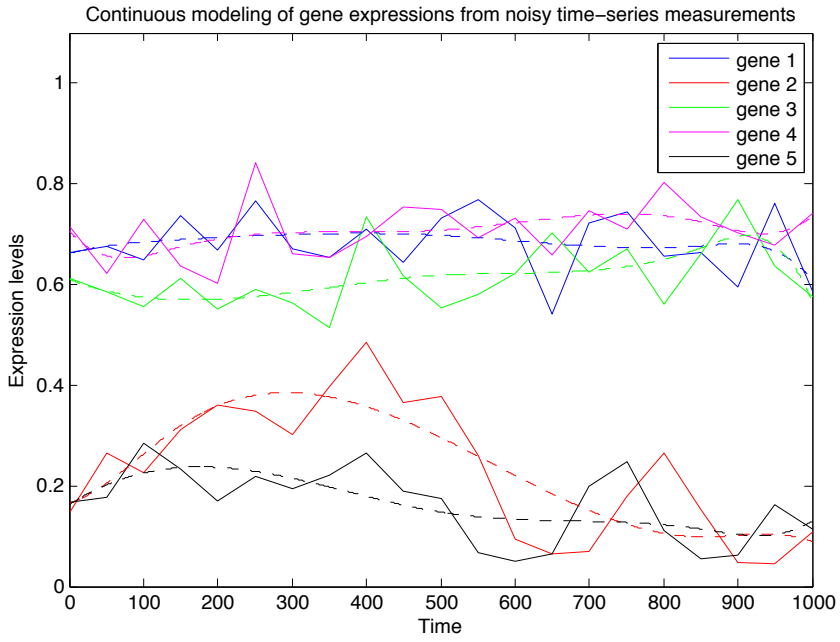

Figure 2.9: Time series gene expression measurements from simulated DREAM4 datasets are shown with connected solid lines. Dashed lines of corresponding color show that application of (P1) effectively produces noise-free (smooth) and continuous gene expression curves.

disadvantage, we attempt network recovery using only the time series perturbations, forgoing all other datasets available to solvers. Lastly, our approach works best under conditions where RNA and protein degradation rates are known. Given that this information is unavailable, this exercise also serves as a test of our simplifying assumptions for such situations. Unlike simulations in the previous section, the rules of this challenge stipulate no self-regulation and no combined regulators.

DREAM4 Challenge 2 datasets for Networks 1 and 2 are used to infer gene regulatory networks and to inspect predictions of network topology using the official scoring pipeline. First, we use (P1) to produce smooth and continuous gene expression trajectories from the discrete and noisy time series datasets (Figure 2.9). Perturbed genes are identified and incorporated as described in Section 2.3.2. Network inference is carried out using Algorithm 1. In the absence of RNA degradation rates, λ_{min} is set to either 0.001 or 0.01, and λ_{max} is set to 0.1 or 1. If a directed network edge is identified, the probability of the edge is set to 1 for weighted edges, and

0 otherwise. This is done to allow scoring of our network with the provided scripts, given our non-probabilistic formulation. Algorithm 1 minimization values are filtered against abnormal values that could represent underfitting and overfitting of data.

For Network 1, we report the area under the receiver operating characteristic curve (AUROC) = 0.81 and the area under the precision-recall curve (AUPR) = 0.75, and for Network 2, AUROC = 0.76 and AUPR = 0.68. These results compare very favorably to other time series-based methods applied to the same datasets [9]. In fact, for Networks 1 and 2, the AUROC and AUPR values represent improvements over the top reported results.

2.5.3 Network Inference From Yeast Cell Cycle Time Series

In order to probe real biological data with inherent noise, we apply parts of our pipeline to a classical yeast cell cycle microarray dataset [77]. This data is provided as a 25 point time-series with a 5 minute sampling interval. Given the yeast cell is in an incredibly dynamic stage post synchronization with α-factor pheromone, this again represents a vast departure from ideal near steady-state conditions with a precise and local perturbation. We chose to focus our analysis on a set of primary regulatory genes and complexes involved in core cell cycle control and that showed greater than 15% changes in expression over the time course [1]. This led to retainment of 7 genes. We use (P2) to fit smooth continuous functions to the noisy gene expression measurements exhibited in Figure 2.10. We next examine our proposed scheme, (P4), to infer a gene regulatory network among these genes.

The inferred network is shown in Figure 2.10, with arrows indicating directed edges for gene-gene excitatory and inhibitory interactions. Of the 12 regulatory interactions inferred, 6 are correct in both directionality and influence (i.e. inhibition vs activation) and 2 are correct only in directionality. Further, 3 can be considered conditionally correct, whereby the predicted influence is mediated by a single intermediate node that was absent from the model. A single edge was labeled as a false positive, even though an argument can be made for mediation of that influence by two intermediate nodes. Strikingly, the algorithm correctly predicts a role for combined regulators and recovers the only example of self-regulation in the reference pathway. This is promising, given the absence of data relating to protein degradation,

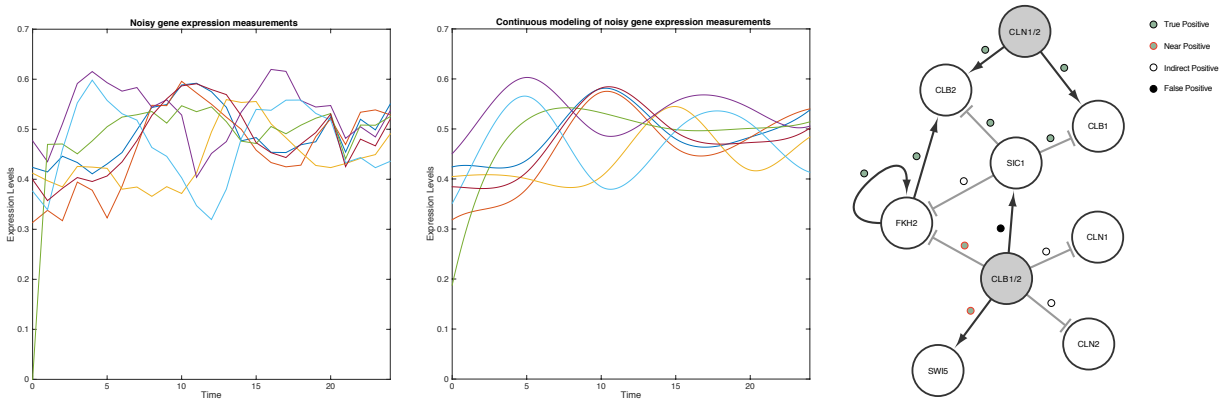

Figure 2.10: Time-series gene expression measurements of yeast cell cycle-associated genes filtered at a stringent change detection threshold ($T = 0.15$) (left), and their recovered estimations using (P2) (center). The inferred network via (P4) is shown on the right, compared to the network as it's presently understood [1]. "True positives" represent edges recapitulated by the inference algorithm in both direction and influence, "near positives" represent edges correct in direction but with reversed influence, "indirect positives" represent edges of correct direction and influence with a missing intermediate node, and "false positive" indicates an edge not found in the reference network and that cannot be explained through a single intermediate node.

contextless inference, and the non step-wise nature of changes in expression that would be preferred in our proposed experimental scheme.

2.6 Summary

The gene inference pipeline described in this work helps establish a robust framework for network discovery from perturbed expression data. The system of equations used to model eukaryotic gene regulation include the novel extension of a thermodynamic and statistical mechanic approach to polymerase binding. This pipeline is best suited for the processing of expression measurements from high-resolution time series experiments involving precise genetic or chemical perturbation of a steady state system. Genetic perturbation is best in the form

of induced over-expression or RNAi-mediated gene knockdown. Chemical perturbation is best in the form of a chemical that has a specific protein interaction and limited off-target effects. However, we establish that this approach can yield insights under non-ideal conditions.

The modular nature of our pipeline allows for the modification of different stages to best fit a given biological system and of expression information. Alternative approaches can be implemented for the stags that precede the core inference algorithm, including change detection. The performance of this approach can further be improved with *a priori* knowledge of protein expression levels, protein and RNA degradation rates, along with the labeling of non-coding RNAs. Technologies are continually being improved for the purpose of capturing these data in a genome-wide manner [78–81], to complement gene expression measurements. Our gene inference approach can readily utilize protein expression data, protein and RNA degradation data, and miRNA labeling data.

While we expect such inference approaches to work better for homogenous and synchronized single-cell or single-tissue systems, we also expect to capture the most prominent and meaningful aspects of the aggregate dynamics of heterogenous mixed-cell populations, multi-tissue systems, and whole organisms. Future directions include the more comprehensive validation and refinement of these algorithms for synthetic networks and higher-order eukaryotic systems, adaptations of more sophisticated change detection schemes, and surveys of a broader range of system-specific sampling frequencies.

This inference method has broad application in biological network discovery. For example, it can be used to identify the topology of gene regulatory networks immediate to drug response, and can be used to identify new interactions for genes implicated in disease. The inference data can then be used to seed and prioritize candidates for downstream biological and *in vivo* validation.

CHAPTER 3. HIGH-DIMENSIONAL COVARIANCE MATRIX ESTIMATION BASED ON KRONECKER PRODUCTS AND PARTIAL OBSERVATIONS

A paper to be submitted

Mahdi Zamanighomi[1] and Zhengdao Wang[1]

Learning of large-scale network topology based on measurements is a fundamental problem, and also the main goal of the thesis. The availability of covariance matrices, combined with observations, leads to a better estimation of network topology [82, 83]. In this chapter, we specifically study the problem of high-dimensional covariance matrix estimation with partial observations. We assume that the true covariance matrix can be represented as an expansion of Kronecker products and observations suffer from missing values. In the absence of missing data, observation vectors are assumed to be i.i.d multivariate Gaussian. In particular, we propose a new procedure computationally affordable in high dimension to extend the permuted rank-penalized least-square method [21] to the case of missing data. Our approach is applicable to a large variety of missing data mechanisms, whether the process generating missing values is random or not, and does not require imputation techniques. We introduce a novel unbiased estimator and characterize its convergence rate to the true covariance matrix measured by the spectral norm of a permutation operator. We also show that the estimator is positive definite as the number of samples goes to infinity. We establish a tight outer bound on the square error of our procedure and elucidate consequences of missing values on the estimation performance. Different schemes are compared by numerical simulations in order to evaluate several missing data models.

[1]Department of Electrical and Computer Engineering, Iowa State University, Ames, IA USA

3.1 Introduction

The problem of covariance estimation with partial observations is fundamental and occurs in variety of applications such as gene expression profile analyses [84,85], machine learning [86], climate studies [87], and graphical models [88].

In practical applications, measurements may not be fully obtained, which results in an observation data vector with missing entries. We can view the observation vector as having less entries provided that missing entries take place at fixed positions of the vector. However, missing entries may occur at positions that randomly change with time, requiring more complex estimation methods.

There are many ways to deal with missing values, each of which results in different performance [89–93]. Excluding missing data is the simplest approach, yet has noticeable flaws [28]. In this method, we remove all variables for which observations are missing and then limit the statistical analysis to the fully observed variables. However, in some applications such as gene expression data where the majority of genes are disturbed by missing data, we are left with few variables and many available observations are wasted.

An alternative approach is to fill in missing values based on imputation techniques. For this arrangement, existing procedures involve intensive computations to approximate missing elements, e.g., EM algorithm [87].

Many recent applications involve huge datasets with both large sample size N and large dimension d where the number of dimensions may drastically exceed the number of observations. This problem leads to the idea of dimension reduction, also known as sparse or low rank constraints, that is finding good low-dimensional representations of massive datasets. Promising methods in several fields, such as compressed sensing [94], have been proposed to perform dimension reduction [22, 95–102].

Covariance matrices are not necessarily low rank and may follow different structures. For instance, a well known class of covariance is positive definite matrices that are full rank. A covariance matrix exhibits the Kronecker product (KP) structure [103, 104] if the covariance can be represented as a sum of Kronecker products of two lower dimensional matrices. This

model is favorable in variety of applications, for instance, matrix normal distributions are used in nonparametric Bayesian approaches especially in learning Gaussian Processes for multiple outputs prediction [105]. It is well known that the covariance matrix of the Gaussian Process prior takes the form of KP. Additionally, the KP structure has applications in genomics [26, 106,107], collaborative filtering [25,108], geostatistics [109], and multivariate repeated measures data [110, 111].

Recently, [21] has proposed a convex optimization approach to estimate convariance matrices with the KP structure and has derived a tight high-dimensional SE convergence rate as N and d go to infinity. This method, called the Permuted Rank-penalized Least Squares (PRLS), illustrates promising results in the spatio-temporal linear least squares prediction of multivariate wind speed datasets. The PRLS however is not applicable to a large variety of problems imposed by missing data.

In this chapter, we generalize the PRLS method [21] to the case of missing data. In particular, we seek to estimate high-dimensional covariance matrices with the KP structure through partial observations. We propose a novel method for the treatment of missing data, which requires neither imputing missing observations nor discarding any available observations to recover covariance matrix. Notably, this novel approach utilizes the empirical covariance matrix (ECM), even though we have no access to the ECM as a result of missing observations. Furthermore, we show that our estimator achieves the same SE convergence rate as [21], wherein all observations are fully captured. However, we obtain that the estimator convergence rate holds with a different probability due to the impact of missing values. Interestingly, our analysis reveals circumstances under which high convergence probability is guaranteed.

The chapter is organized as follows. Section 3.2 introduces a new unbiased estimator and generalizes the PRLS covariance estimation method to missing observations. Then, Section 3.3 characterizes the symmetry and positive definiteness of our estimator. Employing an operator norm bound from Section 3.4, the SE convergence rate of the proposed method is established in Section 3.5. Numerical simulations are given in Section 3.6 to demonstrate the effectiveness of the proposed algorithm. Finally, Section 3.7 summarizes the chapter and describes some remaining open problems associated with missing values.

Notation: Throughout this chapter $\{i, j, t\}$ are integer indices. Column vectors and matrices are indicated by bold lower-case and upper-case letters, respectively. Symbol $\mathbf{x}(i)$ indicates the ith entry of vector $\mathbf{x}$ and $\mathbf{X}(i, j)$ shows the (i, j)th element of matrix $\mathbf{X}$. We use $\mathbf{X}^T$ to denote the transpose of matrix $\mathbf{X}$, vec$(\mathbf{X})$ the vectorized form of matrix $\mathbf{X}$ (stacking the columns of $\mathbf{X}$ into one column), $\|\mathbf{X}\|_F$ the Frobenius norm of matrix $\mathbf{X}$, $\|\mathbf{X}\|_*$ the nuclear norm of matrix $\mathbf{X}$, $\|\mathbf{X}\|_\infty$ the largest singular value of matrix $\mathbf{X}$, and $\|\mathbf{X}\|_0$ the smallest singular value of matrix $\mathbf{X}$. The operator $\circ$ indicates the Hadamard product and $\otimes$ the Kronecker product.

For a $d_1 d_2 \times d_1 d_2$ matrix $\mathbf{X}$, $\{\mathbf{X}[i, j]\}_{i,j=1}^{d_1}$ represents its $d_2 \times d_2$ block submatrices, where submatrices are in the form of $\mathbf{X}[i, j] = \mathbf{X}\,(1 + (i-1)d_2 : id_2, 1 + (j-1)d_2 : jd_2)$. We define the permutation map $\mathcal{P} : \mathbb{R}^{d_1 d_2 \times d_1 d_2} \to \mathbb{R}^{d_1^2 \times d_2^2}$, in which the $(i-1)d_1 + j$ row of $\mathcal{P}(\mathbf{X})$ is equal to vec$(\mathbf{X}[i, j])^T$. For instance, the permuted version of a 4×4 matrix $\mathbf{A}$ when $d_1 = d_2 = 2$ is equal to

$$\begin{pmatrix} A(1,1) & A(2,1) & A(1,2) & A(2,2) \\ A(1,3) & A(2,3) & A(1,4) & A(2,4) \\ A(3,1) & A(4,1) & A(3,2) & A(4,2) \\ A(3,3) & A(4,3) & A(3,4) & A(4,4) \end{pmatrix}.$$

We use vec$^{-1}(\cdot)$ and $\mathcal{P}^{-1}(\cdot)$ to show the inverse operator for vec$(\cdot)$ and $\mathcal{P}(\cdot)$, respectively. The operation $\mathbf{X}^{(\alpha)}$ takes each element $\mathbf{X}(i, j)$ to $\mathbf{X}(i, j)^\alpha$ and similarly, $\mathbf{x}^{(\alpha)}$ transforms $\mathbf{x}(i)$ into $\mathbf{x}(i)^\alpha$. We define $\mathcal{S}_d = \{\mathbf{X} \in \mathbb{R}^{d_1 \times d} : \mathbf{X} = \mathbf{X}^T\}$ to denote the set of real symmetric matrices, $\mathcal{S}_d^+$ the set of real symmetric positive semidefinite matrices, $\mathcal{S}_d^{++}$ the set of real symmetric positive definite matrices, and $\mathcal{N}_d = \{\mathbf{x} \in \mathbb{R}^d : \mathbf{x}^t\mathbf{x} = 1\}$ the unite Euclidean sphere.

3.2 System Model

Let $\{\mathbf{x}_t\}_{t=1}^n$, $\mathbf{x}_t \in \mathbb{R}^d$, be i.i.d. multivariate Gaussian vectors with zero mean and unknown covariance matrix $\mathbf{\Sigma}_0$. We observe n i.i.d random vectors $\{\mathbf{z}_t\}_{t=1}^n$ as

$$\mathbf{z}_t = \mathbf{\Gamma}_t \mathbf{x}_t, 1 \le t \le n \tag{3.1}$$

where $\mathbf{\Gamma}_t$ is defined as the $d \times d$ diagonal matrix with $\mathbf{\Gamma}_t(i, i) = 0$, $1 \le i \le d$, if $\mathbf{x}_t(i)$ is missing and 1 otherwise. We emphasize that our analysis is not limited to such data that are missing

completely at random (MCR), missing at random (MR), or not missing at random (NMR) [90]. Particularly, we consider model (3.1) for all possible arrangements of $\mathbf{\Gamma}_t$ since several random and non-random processes could simultaneously generate missing values and even further, we may not be able to model the missing data mechanism [112]. To the best of our knowledge, existing missing data techniques such as maximum likelihood and multiple imputation, are effective under specific structures for missing values and often computationally expensive in high-dimensional setup.

Our goal is to estimate $\mathbf{\Sigma}_0$ given partial observations $\{\mathbf{z}_t\}_{t=1}^n$. We assume that (i) the positions of missing data, $\mathbf{\Gamma}_t$ for all $1 \leq t \leq n$, are known and (ii) the covariance matrix can be written as a sum of KPs of two lower dimensional matrices:

$$\mathbf{\Sigma}_0 = \sum_{i=1}^{r} \mathbf{A}_i \otimes \mathbf{B}_i, \tag{3.2}$$

where $\{\mathbf{A}_i\}_{i=1}^r$ are $d_1 \times d_1$ linearly independent matrices, $\{\mathbf{B}_i\}_{i=1}^r$ are $d_2 \times d_2$ linearly independent matrices, and $d = d_1 d_2$. We additionally assume that the factor dimensions d_1 and d_2 are given. The integer r denotes the total number of KPs in the summation and is supposed to be less than $\min(d_1^2, d_2^2)$ [21]. The mentioned model (3.2) can be interpreted as a low rank principle component decomposition where components are KPs, but neither orthogonal nor normalized.

Given observations with no missing data, a sufficient statistic to estimate the true covariance matrix $\mathbf{\Sigma}_0$ is the ECM:

$$\frac{1}{n} \sum_{t=1}^{n} \mathbf{x}_t \mathbf{x}_t^T \tag{3.3}$$

However, the above unbiased estimator is not functional since we only have access to $\mathbf{z}_t$. We thus consider the following alternative

$$\mathbf{\Sigma}_n^{\mathbf{\Gamma}} = \frac{1}{n} \sum_{t=1}^{n} \mathbf{z}_t \mathbf{z}_t^T. \tag{3.4}$$

This estimator concentrates around its mean, $\mathbf{\Sigma}_0^{\mathbf{\Gamma}} := E[\mathbf{\Sigma}_n^{\mathbf{\Gamma}}]$, which could be far away from $\mathbf{\Sigma}_0$ and lead to unacceptably large biases in parameter estimates [27, 113]. To remove the introduced bias, let us first rewrite $\mathbf{\Sigma}_0^{\mathbf{\Gamma}}$ as follows

$$\mathbf{\Sigma}_0^{\mathbf{\Gamma}} = E[\frac{1}{n} \sum_{t=1}^{n} \mathbf{z}_t \mathbf{z}_t^T] = E[\frac{1}{n} \sum_{t=1}^{n} \mathbf{\Gamma}_t \mathbf{x}_t \mathbf{x}_t^T \mathbf{\Gamma}_t] = \frac{1}{n} \sum_{t=1}^{n} \mathbf{\Gamma}_t \mathbf{\Sigma}_0 \mathbf{\Gamma}_t = \mathbf{W} \circ \mathbf{\Sigma}_0, \tag{3.5}$$

where $\mathbf{W}$ is the weight matrix with entries $\mathbf{W}(i,j) = \frac{1}{n}\sum_{t=1}^{n} \mathbf{\Gamma}_t(i,i)\mathbf{\Gamma}_t(j,j)$. Although entries of $\mathbf{W}$ belong to the interval $[0,1]$, we assume $\mathbf{W}(i,j) \in (0,1]$ acknowledging that all variables are successfully measured in at least one time point. Therefore equation (3.5) can be represented as

$$\mathbf{\Sigma}_0 = \mathbf{W}^{(-1)} \circ \mathbf{\Sigma}_0^{\mathbf{\Gamma}}, \tag{3.6}$$

leading to the following unbiased estimator of $\mathbf{\Sigma}_0$ when the dataset contains missing observations:

$$\hat{\mathbf{\Sigma}}_n := \mathbf{W}^{(-1)} \circ \mathbf{\Sigma}_n^{\mathbf{\Gamma}}. \tag{3.7}$$

This unbiased estimator not only takes advantage of all available information to estimate $\mathbf{\Sigma}_0$ but also can be employed whether missing patterns are random or not. The model (3.7) suffers from high variance when the number of samples, n, is smaller than the number of dimensions, d. To tackle this challenge, a low rank approximation to $\hat{\mathbf{\Sigma}}_n$ is usually considered. The popular low rank approximation called the standard principal component analysis (PCA) performs the Eigen-Decomposition of $\hat{\mathbf{\Sigma}}_n$ to retain the top r principle components. The PCA estimator then takes the form of

$$\hat{\mathbf{\Sigma}}_n^{PCA} = \sum_{i=1}^{r} \sigma_i^2 \mathbf{v}_i \mathbf{v}_i^T, \tag{3.8}$$

where σ_i is the ith largest singular value associated with the right singular vector $\mathbf{v}_i$. In high-dimensional setting however, the PCA can be affected by excessive bias and bound to fail [114]. This phenomenon is mainly connected to known inconsistency results for sample eigenvalues and eigenvectors as d increases.

In [115], an alternative method to derive the low rank covariance estimation is proposed as the solution of the following penalized minimization problem:

$$\hat{\mathbf{\Sigma}}_n^{\lambda} := \arg\min_{\mathbf{\Sigma} \in \mathbb{S}_{++}^{p}} \|\hat{\mathbf{\Sigma}}_n - \mathbf{\Sigma}\|_F^2 + \lambda \operatorname{Tr}(\mathbf{\Sigma}), \tag{3.9}$$

in which λ is a tuning parameter and $\operatorname{Tr}(\mathbf{\Sigma})$ is equivalent to the 1-norm on the eigenvalues of $\mathbf{\Sigma}$. The estimator (3.9) is developed for the case where all $\mathbf{\Gamma}_t(i,i)$ are i.i.d Bernoulli random variables with the same parameter δ and independent of $\{\mathbf{x}_t\}_{t=1}^{n}$. For this scenario, the unbiased

estimator $\hat{\mathbf{\Sigma}}_n$ is simplified to

$$(\delta^{-1} - \delta^{-2})\text{diag}(\mathbf{\Sigma}_n^{\mathbf{\Gamma}}) + \delta^{-2}\mathbf{\Sigma}_n^{\mathbf{\Gamma}}. \tag{3.10}$$

Corollary 1 in [115] proves that the solution to the convex problem (3.9) converges to $\mathbf{\Sigma}$ at a minimax optimal rate.

Here, we propose a penalized empirical risk minimization problem analogous to (3.9), but applicable to any $\mathbf{\Gamma}_t$, and generalize the PRLS [21] to the case in which observations are partially captured, cf. models (3.1) and (3.2). More precisely, we propose the following convex optimization to estimate the permuted version of $\mathbf{\Sigma}_0$:

$$\text{(P1)} \qquad \hat{\mathbf{P}}_n^{\gamma} = \arg \min_{\mathbf{P} \in \mathbb{R}^{d_1^2 \times d_2^2}} \|\hat{\mathbf{P}}_n - \mathbf{P}\|_F^2 + \gamma \|\mathbf{P}\|_*,$$

where $\hat{\mathbf{P}}_n \coloneqq \mathcal{P}(\hat{\mathbf{\Sigma}}_n)$ (cf. Notation), $\mathbf{P} \coloneqq \mathcal{P}(\mathbf{\Sigma})$, and γ is a rank-controlling parameter. The term $\|\hat{\mathbf{P}}_n - \mathbf{P}\|_F^2$ is equivalent to $\|\hat{\mathbf{\Sigma}}_n - \mathbf{\Sigma}\|_F^2$ (Theorem 2.1 in [116]). To shed light on the need of $\|\mathbf{P}\|_*$, let's consider equation (3.2). It is easy to show $\mathbf{P}_0 \coloneqq \mathcal{P}(\mathbf{\Sigma}_0) = \sum_{i=1}^{r} \text{vec}(\mathbf{A}_i)\text{vec}(\mathbf{B}_i)^T$. This suggests that $\mathbf{P}$ must be of rank r at most and therefore (P1) is a convex relaxation of

$$\begin{aligned} &\min_{\mathbf{P} \in \mathbb{R}^{d_1^2 \times d_2^2}} \|\hat{\mathbf{P}}_n - \mathbf{P}\|_F^2 \\ &\text{subject to} \quad \text{rank}(\mathbf{P}) \leq r. \end{aligned} \tag{3.11}$$

In particular, to obtain the convex relaxation of the above NP-hard problem, we leverage from recent developments in compressive sampling [95] and substitute the ℓ_0-norm with its ℓ_1-norm surrogate, which here corresponds to the nuclear norm $\|\mathbf{P}\|_*$.

It is well known that the solution of (P1) is in the following closed form [117]:

$$\hat{\mathbf{P}}_n^{\gamma} = \sum_{i=1}^{\min(d_1^2, d_2^2)} \max\left(0, \sigma_i(\hat{\mathbf{P}}_n) - \frac{\gamma}{2}\right) \mathbf{u}_i \mathbf{v}_i^T, \tag{3.12}$$

where $\sigma_i(\hat{\mathbf{P}}_n)$ is the ith largest singular value of $\hat{\mathbf{P}}_n$ corresponding to the left and right singular vectors $\mathbf{u}_i$ and $\mathbf{v}_i$, respectively. The answer $\hat{\mathbf{P}}^{\gamma}$ is essentially transformed back to the original matrix space $\mathbb{R}^{d \times d}$ employing $\hat{\mathbf{\Sigma}}_n^{\gamma} \coloneqq \mathcal{P}^{-1}(\hat{\mathbf{P}}_n^{\gamma})$ (cf. Notation). In the next section, we explore the symmetric and positive definiteness of our estimation $\hat{\mathbf{\Sigma}}_n^{\gamma}$.

Remark 3.1. *Fast algorithms for solving convex optimization problems with a nuclear norm regularization, such as (P1), have been recently proposed [117–120]. Numerical results suggest that these methods are amenable to very large scale problems and recover low rank matrices with nearly a billion unknowns. We conclude that the proposed (P1) can be efficiently solved when matrix* $\mathbf{P}_0$ *is low rank, equivalently* $r \ll \min(d_1^2, d_2^2)$*. Moreover, we note that the obtained solution is unique because (P1) is strictly convex for* $\gamma > 0$*.*

Remark 3.2. *In statistics, an outlier is defined as an observation far from other observations, which could be due to mixture of different distributions or experimental errors. One may wish to exclude them from datasets or design methods that are robust to outliers. Here, we propose to use the estimated covariance matrix* $\hat{\mathbf{\Sigma}}_n^\gamma$ *to detect outliers. Interestingly [121] presents a fast algorithm based on known distributions, which minimizes the covariance determinant to recognize the most extreme measurements. We then discard the detected outliers, treated as missing data, and invoke (P1) to re-estimate the covariance. We eventually perform the mentioned process for a few iterations to refine our covariance matrix approximation.*

Remark 3.3. *In the case that the position of missing data is not available, we treat missing values as outliers and follow the above remark to locate where missing values occur.*

3.3 Symmetry and Positive Definiteness

Here, we investigate consequences of missing data on the de-permuted solution $\hat{\mathbf{\Sigma}}_n^\gamma$ to illuminate the possibility of successful arrival at a symmetric and positive definite estimation. Specifically, we show that $\hat{\mathbf{\Sigma}}_n^\gamma$ is symmetric with probability 1 and, furthermore, is positive definite with at least a probability, which exponentially increases as the number of samples grows.

Employing the essence of Theorem 1 in [21], we discern that the de-permuted solution of (P1), $\hat{\mathbf{\Sigma}}_n^\gamma$, is

1. symmetric with probability 1 if $\hat{\mathbf{\Sigma}}_n$ is symmetric.

2. positive definite with probability 1 if $\hat{\mathbf{\Sigma}}_n$ is positive definite.

In the context of [21], the estimator $\hat{\mathbf{\Sigma}}_n$ is equivalent to the ECM (3.3) and positive definite for $n \geq d$. However, this estimator in our setup is not necessarily positive definite due to the presence of missing values. Therefore, to utilize the above results, we must explore cases where the symmetry and positive definiteness of $\hat{\mathbf{\Sigma}}_n$ hold. Next theorem proposes an inner bound on the probability that $\hat{\mathbf{\Sigma}}_n$ is positive definite.

Theorem 3.1. *The unbiased estimator $\hat{\mathbf{\Sigma}}_n$ is*

1. *symmetric.*
2. *positive definite with probability 1 provided that $n \geq d_1 d_2$ and $\mathbf{W}^{(-1)}$ is positive semidefinite.*
3. *positive definite with probability at least*

$$1 - 2e^{-n\frac{\|\mathbf{\Sigma}_0\|_0^2}{C_1 C \|\mathbf{\Sigma}_0\|_\infty^2 + C_2\sqrt{C}\|\mathbf{\Sigma}_0\|_0\|\mathbf{\Sigma}_0\|_\infty}} \tag{3.13}$$

for $n \geq d_1 d_2$ and $C := \max_{\mathbf{u}\in\mathcal{N}_{d_1 d_2}} {\mathbf{u}^{(2)}}^T \mathbf{W}^{(-2)} \mathbf{u}^{(2)}$, $C_1 := \frac{8e}{\sqrt{6\pi}}$, $C_2 := 2e\sqrt{2}$.

Proof. See Appendix B. □

To elucidate the effect of missing data on the above probability, let us first introduce an upper bound for C, which is independent of $\mathbf{u}$. Using Appendix B, we obtain that ${\mathbf{u}^{(2)}}^T \mathbf{W}^{(-2)} \mathbf{u}^{(2)} = \left\|\mathbf{W}^{(-1)} \circ (\mathbf{u}\mathbf{u}^T)\right\|_F^2$. Then, we have

$$\left\|\mathbf{W}^{(-1)} \circ (\mathbf{u}\mathbf{u}^T)\right\|_F^2 \leq \left(\max_{i,j} \mathbf{W}^{(-2)}(i,j)\right) \|\mathbf{u}\mathbf{u}^T\|_F^2 = \max_{i,j} \mathbf{W}^{(-2)}(i,j), \tag{3.14}$$

and thus, $C \leq \max_{i,j} \mathbf{W}^{(-2)}(i,j)$. Consider a situation in which 25% of elements within the set $\{\mathbf{x}_t(i)\}_{t=1}^n$ is not available for each i. In this scenario, the percentage of missing values is quite noticeable in comparison with contemporary datasets. One can easily show that $16/9 \leq W^{(-2)}(i,j) \leq 4$, resulting in $C \leq 4$. Hence, to diminish the involvement of missing observations from the probability (3.13), we need at most $4n$ samples. Moreover, taking into account that the probability (3.13) exponentially increases in n and the value C is finite, the probability (3.13) is almost 1 for an appropriate number of samples.

Remark 3.4. *Matrix* $\mathbf{W}$ *is positive semidefinite since* $\mathbf{W}$ *can be written as* $\mathbf{H}\mathbf{H}^T$ *with* $\mathbf{H}(i,j) \coloneqq \frac{1}{\sqrt{n}}\mathbf{\Gamma}_j(i,i)$, $\forall 1 \leq i \leq d$ *and* $1 \leq j \leq n$. *Problem 7.5.P12 in [64] states that for the positive semidefinite* $\mathbf{W}$, *the Hadamard inverse matrix* $\mathbf{W}^{(-1)}$ *is also positive semidefinite if and only if* $\mathbf{W}$ *is rank one. Thus, the sufficient and necessary condition for credibility of the second claim in Theorem 3.1 is to arrive at rank one* $\mathbf{W}$. *For instance, the rank one criteria ensures that the estimator* $\hat{\mathbf{\Sigma}}_n$ *is positive definite with probability 1 provided that missing entries take place with a pattern guaranteeing the same vector* $(\mathbf{\Gamma}_1(i,i), \ldots, \mathbf{\Gamma}_n(i,i))$ *for all* i. *In addition, the second requirement of Theorem 3.1 is satisfied when all observations are at hand.*

3.4 Spectral Norm Bound

In this section, we characterize a bound on the spectral norm of $\mathbf{D}_n \coloneqq \hat{\mathbf{P}}_n - \mathbf{P}_0$. We will take advantage of this result to derive a tight outer bound on the SE estimation error.

Theorem 3.2. *Fix* $\epsilon \in [0, \frac{1}{2})$ *and define* $N \coloneqq \max(d_1, d_2, n)$ *and* $C_0 \coloneqq \max(C_1 C_P, C_2\sqrt{C_P})$ *where*

$$C_P \coloneqq \max_{\mathbf{u}\in\mathcal{N}_{d_1^2}, \mathbf{v}\in\mathcal{N}_{d_2^2}} \mathbf{u}^{(2)} \mathcal{P}\left(\mathbf{W}^{(-2)}\right) \mathbf{v}^{(2)}.$$

Assume $q \geq \max\left(\sqrt{2C_1 C_P \ln(1+\frac{2}{\epsilon})}, 2C_2\sqrt{C_P}\ln(1+\frac{2}{\epsilon})\right)$. *Then, we have*

$$\|\mathbf{D}_n\|_\infty \leq \frac{q\|\mathbf{\Sigma}_0\|_\infty}{1-2\epsilon} \max\left(\frac{d_1^2+d_2^2+\log N}{n}, \sqrt{\frac{d_1^2+d_2^2+\log N}{n}}\right) \tag{3.15}$$

with probability at least $1 - 2N^{-\frac{q}{2C_0}}$.

Proof. Following the proof of Theorem 3 in [21], combined with Lemma B.2 (see Appendix B), the proof is established. □

We emphasize that Lemma B.2 plays an essential role in the proof of the above theorem. In fact, the new lemma allows us to generalize the operator norm bound on the permuted ECM (3.3), derived in [21], to our unbiased estimator (3.7).

Clearly, Theorem 3.2 demands no condition on the covariance matrix $\mathbf{\Sigma}_0$. However, for the theorem to be of any practical interest, we require the outer bound in (3.15) to be small,

leading to

$$n \geq \beta C_P (d_1^2 + d_2^2 + \log N), \tag{3.16}$$

where $\beta > 0$ is a sufficiently large constant number. This appealing criteria reveals the impact of missing data, C_P, on the number of measurements sufficient to guarantee an accurate approximation to the spectral norm of $\mathbf{\Sigma}_0$. We note that the required number of samples does not dramatically grow as a response to missing data, that is because $C_P \leq \max_{i,j} \mathbf{W}^{(-2)}(i,j)$ (similar argument as Section 3.3) is a small number in variety of applications.

3.5 SE Bound

Here, we establish a tight outer bound on the SE $\|\hat{\mathbf{\Sigma}}_n^{\gamma} - \mathbf{\Sigma}_0\|_F^2$. This result is built using a bound on the Frobenius norm of $\|\hat{\mathbf{P}}_n - \mathbf{P}_0\|_F^2$ and the fluctuation of $\mathbf{D}_n$ measured by the spectral norm, Theorem 3.2.

Theorem 3.3. *Choose γ equal to*

$$\frac{2q\|\mathbf{\Sigma}_0\|_\infty}{1-2\epsilon} \max\left(\frac{d_1^2 + d_2^2 + \log N}{n}, \sqrt{\frac{d_1^2 + d_2^2 + \log N}{n}}\right),$$

where the introduced parameters are characterized based on Theorem 3.2. Then, we have

$$\|\hat{\mathbf{\Sigma}}_n^{\gamma} - \mathbf{\Sigma}_0\|_F^2 \leq \inf_{\mathbf{P}:\mathrm{rank}(\mathbf{P})\leq r} \|\mathbf{P} - \mathbf{P}_0\|_F^2 + \frac{\left(1+\sqrt{2}\right)^2}{4}\gamma^2 \mathrm{rank}(\mathbf{P}) \tag{3.17}$$

with probability at least $1 - 2N^{-\frac{q}{2C_0}}$.

Proof. See Appendix B. □

The above theorem provides some insight on the tuning of the regularization parameter γ. We notice this choice of γ depends on $\|\mathbf{\Sigma}_0\|_\infty$, which is generally unknown. Thus, we suggest using $\|\hat{\mathbf{\Sigma}}_n\|_\infty$ instead so that γ can be specified based on available information.

Given that $\mathbf{\Sigma}_0$ takes the form of (3.2), the estimation error $\min_{\mathbf{P}:\mathrm{rank}(\mathbf{P})\leq r} \|\mathbf{P} - \mathbf{P}_0\|_F^2$ is zero. Therefore for large enough n, Theorem 3.3 offers that the SE approximation error $\|\hat{\mathbf{\Sigma}}_n^{\gamma} - \mathbf{\Sigma}_0\|_F^2$ is of order $r\frac{d_1^2+d_2^2+\log N}{n}$, with probability not less than $1-2N^{-\frac{q}{2C_0}}$. Indeed, this asymptotic SE convergence rate of the covariance estimation with partial observations coincides with the same

rate achieved in [21], where all observations are available. However, the probability $1-2N^{-\frac{q}{2C_0}}$ exhibits a change, that is the consequence of missing data. More precisely, the elements q and C_0 have greater values, compared to the non-missing case. Furthermore, the larger value of q causes the regularization parameter γ to increase, demonstrating a greater emphasis on the rank constraint in (P1).

The order of mean-square error (MSE) convergence rate for the standard sample covariance matrix is $\frac{d_1^2 d_2^2}{n}$, which is clearly less than the convergence rate of $\hat{\mathbf{\Sigma}}_n$. Therefore, we realize from Theorem 3.3 that the SE convergence rate of (P1) is significantly lower than the MSE convergence rate of the unbiased sample covariance matrix $\hat{\mathbf{\Sigma}}_n$, provided that rank $r \ll \min(d_1^2, d_2^2)$.

We finally discern from Theorem 3.3 that the solution of (P1) takes a structure similar to (3.2) to satisfy the infimum (3.17), where each term in the expansion, $\mathbf{A}_i$ and $\mathbf{B}_i$, can be of any arbitrary rank. This freedom, nevertheless, can not be offered by the PCA procedure since each term is limited to rank one.

3.6 Simulation

In order to provide a quantitative illustration of the results in this chapter, we compare the SE performance obtained by the PRLS (solution of (4) in [21]), the Generalized PRLS (solution of (P1)), the ECM (equation (3.3)), and the Generalized ECM (equation (3.7)). We emphasize that the PRLS and ECM methods can not tolerate missing values while the Generalized PRLS and the Generalized ECM are applicable to missing data.

We construct the true covariance matrix $\mathbf{\Sigma}_0$ employing model (3.2) with $d_1 = d_2 = 10$ and $r = 3$. Factors A_i and B_i take the form of $\mathbf{S}\mathbf{S}^T$, $\mathbf{S}$ is a square random matrix whose columns follow a Gaussian distribution, which results in positive definite $\mathbf{\Sigma}_0$. We then generate 100-dimensional observation vectors based on the Gaussian distribution with zero mean and covariance matrix $\mathbf{\Sigma}_0$. To include missing values, we randomly force 10, 20, and 30 entries of each generated vector to be zero. For these three scenarios, the SE perfomance as a function of sample size is shown in Figure 3.1. As predicted by Theorem 3.3, the Generalized PRLS performs quite close to the PRLS when 10 and 20 percent of entries are gone. Furthermore for

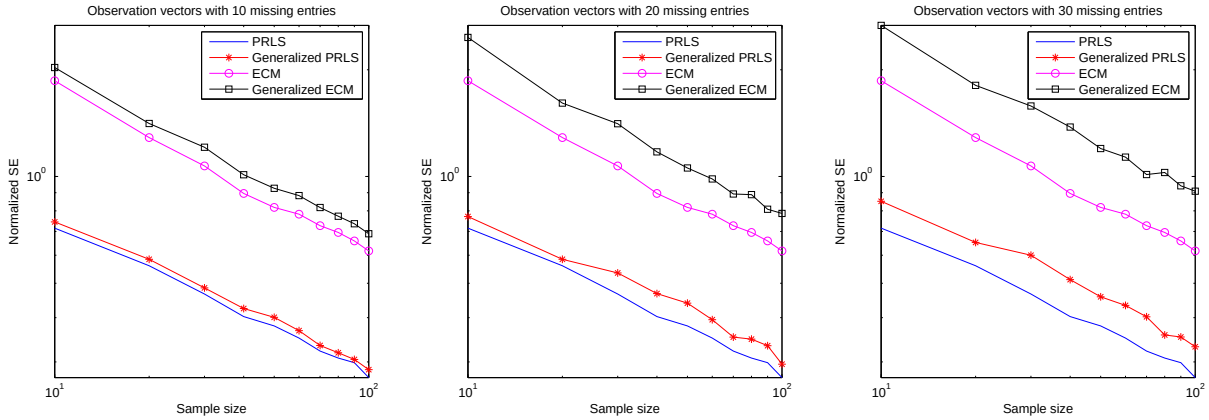

Figure 3.1: SE performance normalized with respect to $\|\mathbf{\Sigma}_0\|_F^2$ versus the number of available samples n. The Generalized PRLS, $\hat{\mathbf{\Sigma}}_n$, and the Generalized ECM, $\hat{\mathbf{\Sigma}}_n$, are derived using 100-dimensioanl observation vectors with 10 missing entries (left), 20 missing entries (center), and 30 missing entries (right). The true covariance matrix is constructed based on model (3.2) with $d_1 = d_2 = 10$ and $r = 3$. We observe that the emprical perfomance of the Generalized PRLS is close to the PRLS method and it also outperforms the ECM and Generalized ECM. Note that the PRLS and ECM require all observations to be compeletely captured.

datasets containing tremendous number of missing values, such as right panel in Figure 3.1, we still achieve an acceptable performance in comparison with the PRLS. We finally observe that the Generalized PRLS notably outperforms the ECM and Generalized ECM.

3.7 Summary

We have shown that the PRLS method can be generalized to datasets that contain missing values. The spirit is to replace the standard ECM with the unbiased estimator (3.7). The novel estimator is applicable to a large variety of missing data patterns, such as MCR, MR, and NMR, as long as all variables are observed in at least one time point. We have analyzed the solution of the Generalized PRLS and shown that the approximated covaraince is positive definite with a probability close to one for an appropriate number of samples. We have further derived an analysis on the concentration of measure phenomenon for observation vectors that

suffer from missing data, cf. Lemma B.2. Using this result, we have established a spectral norm bound and also a SE bound to illustrate the performance of our procedure. We have established that the Generalized PRLS achieves the same convergence rate as the PRLS, but it holds with a different probability because of missing data. We have finally observed from numerical results that the Generalized PRLS preforms quite close to the PRLS.

For the future, it is desirable to obtain all possible cases in which the estimator (3.7) is positive definite with probability 1, see Theorem 3.1. This result will lead us to a complete characterization of positive definiteness of the Generalized PRLS solution. Moreover, it would be interesting to establish an inner bound on the SE or MSE performance and compare it with the proposed SE outer bound. We believe that the Cramér-Rao lower bound (CRLB) intended for biased estimators can be used to derive the MSE inner bound. Finally, we would like to demonstrate the performance of the Generalized PRLS on real world applications that are affected by missing data.

CHAPTER 4. LINEAR MINIMUM MEAN-SQUARE ERROR ESTIMATION BASED ON HIGH-DIMENSIONAL DATA WITH MISSING VALUES

Modified from a paper published in *Annual Conference on Information Sciences and Systems*

Mahdi Zamanighomi[1], Zhengdao Wang[1], Konstantinos Slavakis[2], and Georgios B Giannakis[2]

This chapter intends to develop low-complexity algorithms arising from large-scale data analysis to improve statistical inference and prediction. Specifically, we consider LMMSE estimation problems where the observation data may have missing entries. Processing such data vectors exhibits high complexity if the observation data vector has high-dimensionality and the LMMSE estimator must be re-derived whenever there are missing values. In this context, a means of reducing the computational complexity is introduced when the number of missing entries is relatively small. All first- and second-order data statistics are assumed known, and the positions of the missing values are also known. The proposed method works by first applying the LMMSE estimator on the data vector with missing values replaced by zeros, and then applying a low-complexity update that depends on the positions of the missing. The method achieves exact LMMSE based on only observed data with lower complexity compared to the direct implementation of a time-varying LMMSE filter based on the incomplete data. We also show that if LMMSE imputation is used to fill the missing entires first based on the non-missing entries, and then a complete-data LMMSE filter is applied to the completed data vector, then the same linear MMSE is also achieved, but with higher complexity.

[1]Department of Electrical and Computer Engineering, Iowa State University, Ames, IA USA
[2]Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN USA

4.1 Introduction

LMMSE estimation is a method to linearly estimate a desired signal from an observation so that the mean-square error is minimized. It has many applications in signal processing, communications, and control theory to name a few.

For vector observations, the resulting estimator involves the inverse of the data covariance matrix. When the dimensionality of the observation vector is small, such inversion can be readily implemented. When the dimensionality is high, however, direct inversion may not be feasible, and alternative methods are well motivated.

Goldstein et. al. in [34] presented the *multistage Wiener filter* (MSWF) method to implement the LMMSE filter. In this approach, decompositions based on orthogonal projections are used to derive a multistage structure which avoids explicit inversion of the data covariance matrix.

In practical applications, the observations may not be captured completely, which results in an observation data vector with missing entries. If such missing entries occur at fixed positions of the observation vector, then one can simply view the vector as having less entries. On the other hand, if the missing entries occur at positions that change with time, then the estimation problem is more involved. If all data statistics are known and the missing positions are also known, then the LMMSE estimator can still be derived and applied to the data values that are non-missing. Such computations however may be expensive because a different estimator is needed for each missing data pattern.

In this paper, we seek to reduce the complexity of LMMSE estimation when the data vector with missing entries has high-dimensionality. Our approach is to first apply the full-data processing to the data vector with missing entries, by treating the missing entries as zeros. Then we modify the estimate with a relatively low-complexity update. Such a method reduces the overall complexity of processing the observation data vectors with missing entries while preserving the LMMSE optimality.

Notation: Throughout the paper N, M, and T are integers. We use $t \in \{1, \dots T\}$ to denote the time index, while i and j are integer indices. Vectors and matrices are indicated by lower

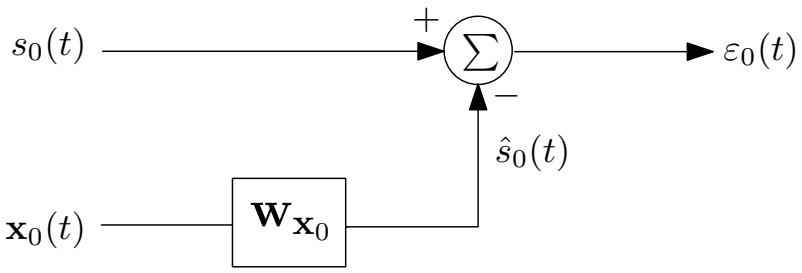

Figure 4.1: Wiener filter

and upper case bold symbols, respectively. We let $\mathbb{E}[\cdot]$ denote expectation, $(\cdot)^H$ the conjugate transpose, and $\mathcal{O}(\cdot)$ the order of complexity.

4.2 LMMSE Estimation

The classical LMMSE estimation problem is depicted in Figure 4.1. The signal $s_0(t) \in \mathbb{C}$ is a scalar desired signal, $\mathbf{x}_0(t) \in \mathbb{C}^{N\times 1}$ is the zero-mean observed data vector, and the vector $\mathbf{w}_{\mathbf{x}_0} \in \mathbb{C}^{N\times 1}$ is the sought LMMSE filter. The signal $s_0(t)$ is estimated as $\hat{s}_0(t) = \mathbf{w}_{\mathbf{x}_0}^H \mathbf{x}_0(t)$. The covariance matrix of $\mathbf{x}_0(t)$ is given by

$$\mathbf{R}_{\mathbf{x}_0} = \mathbb{E}[\mathbf{x}_0(t)\mathbf{x}_0^H(t)] \tag{4.1}$$

and assumed to be nonsingular. The variance of $s_0(t)$ is

$$\sigma_{s_0}^2 = \mathbb{E}[|s_0(t)|^2] \tag{4.2}$$

and the cross correlation between $s_0(t)$ and $\mathbf{x}_0(t)$ is

$$\mathbf{r}_{\mathbf{x}_0 s_0} = \mathbb{E}[\mathbf{x}_0(t)s_0^*(t)] \tag{4.3}$$

where $(\cdot)^*$ denotes conjugation. The error

$$\varepsilon_0(t) \coloneqq s_0(t) - \hat{s}_0(t) = s_0(t) - \mathbf{w}_{\mathbf{x}_0}^H \mathbf{x}_0(t) \tag{4.4}$$

is minimized in the mean-squared sense by the vector

$$\mathbf{w}_{\mathbf{x}_0} = \mathbf{R}_{\mathbf{x}_0}^{-1}\mathbf{r}_{\mathbf{x}_0 s_0}. \tag{4.5}$$

The minimum mean-square error (MMSE) achieved is

$$\mathbb{E}[|\varepsilon_0(t)|^2] = \sigma_{s_0}^2 - \mathbf{r}_{\mathbf{x}_0 s_0}^H \mathbf{R}_{\mathbf{x}_0}^{-1}\mathbf{r}_{\mathbf{x}_0 s_0}. \tag{4.6}$$

4.2.1 Missing Data

When the observation vector $\mathbf{x}_0(t)$ has missing entries, the weight vector in (4.5) cannot be directly applied to obtain the optimal output $\hat{s}_0(t)$. However, if the positions of the missing entries are known, the optimal filtering vector can be derived for the observable (non-missing) part.

We assume that the positions of the missing entries in $\mathbf{x}_0(t)$ are known. Let $\bar{\mathbf{x}}_0(t) \in \mathbb{C}^{M(t)}$ denote the observable part of $\mathbf{x}_0(t)$, which is a vector obtained by removing all entries that correspond to missing values. We refer to $\bar{\mathbf{x}}_0(t)$ as *incomplete data*, and the vector $\mathbf{x}_0(t)$ as *complete data*. The remaining values $\tilde{\mathbf{x}}_0(t)$ that are missing are referred to as *missing data*.

The covariance matrix corresponding to the incomplete data $\bar{\mathbf{x}}_0(t)$ is denoted by $\mathbf{R}_{\bar{\mathbf{x}}_0(t)}$, and the cross correlation vector between $\bar{\mathbf{x}}_0(t)$ and $s_0(t)$ by $\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0}$. The optimal LMMSE estimator is $\mathbf{w}^H_{\bar{\mathbf{x}}_0(t)}\bar{\mathbf{x}}_0(t)$, where

$$\mathbf{w}_{\bar{\mathbf{x}}_0(t)} = \mathbf{R}^{-1}_{\bar{\mathbf{x}}_0(t)}\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0}. \tag{4.7}$$

Since the missing data positions in general may be different at different times, a direct implementation of the LMMSE estimator, which involves the inversion of the covariance matrix of the observable part, is computationally expensive, especially when the dimensionality of $\bar{\mathbf{x}}_0(t)$ namely $M(t)$, is large. Suppose that $M(t)$ is close N, or equivalently the number of missing entries $N - M(t)$ is small in comparison with N. Our objective is to realize time-dependent LMMSE estimation with low complexity, by avoiding the recalculation of $\mathbf{R}^{-1}_{\bar{\mathbf{x}}_0(t)}$ at different times.

4.3 LMMSE Estimation with Incomplete Data

Our approach to reduce complexity is by updating the LMMSE filter output intended for the complete data whenever the input data is not fully available. Instead of re-evaluating the filter coefficients, a small update is applied so that the same MMSE optimality is retained.

To derive the update, suppose first that $\mathbf{R}^{-1}_{\mathbf{x}_0}$ is already computed. Without loss of generality, we assume the missing entries in $\mathbf{x}_0(t)$ are in the last positions such that we can write

the complete data as

$$\mathbf{x}_0(t) = \begin{pmatrix} \bar{\mathbf{x}}_0(t) \\ \tilde{\mathbf{x}}_0(t) \end{pmatrix} \tag{4.8}$$

where $\tilde{\mathbf{x}}_0(t) \in \mathbb{C}^{M\times 1}$ represents the missing entries. Define the $N \times 1$ vector

$$\mathbf{z}_0(t) := \begin{pmatrix} \bar{\mathbf{x}}_0(t) \\ \mathbf{0} \end{pmatrix} \tag{4.9}$$

which is obtained by setting the missing entries in $\mathbf{x}_0(t)$ to zero. In general, the missing data could occur at different entries of $\mathbf{x}_0(t)$. However, $\mathbf{x}_0(t)$ in the desired form can be obtained by applying the appropriate permutation to the observed data vector based on the positions of missing entries.

The update $\mathbf{u}(t)$ we are seeking is such that we can apply the complete-data LMMSE filter to $\mathbf{z}_0(t)$ first, and then update the filter output so that the subsequent LMMSE solution based on $\bar{\mathbf{x}}_0(t)$ is obtained. Specifically, we would like the update to be such that

$$\left(\mathbf{R}_{\mathbf{x}_0}^{-1}\mathbf{r}_{\mathbf{x}_0 s_0}\right)^H \mathbf{z}_0(t) - \mathbf{u}(t) = \left(\mathbf{R}_{\bar{\mathbf{x}}_0(t)}^{-1}\mathbf{r}_{\bar{\mathbf{x}}_0(t) s_0}\right)^H \bar{\mathbf{x}}_0(t). \tag{4.10}$$

The covariance matrix $\mathbf{R}_{\mathbf{x}_0}$ can be partitioned as

$$\mathbf{R}_{\mathbf{x}_0} = \begin{pmatrix} \mathbf{R}_{\bar{\mathbf{x}}_0(t)} & \mathbf{R}_{\tilde{\mathbf{x}}_0(t)\bar{\mathbf{x}}_0(t)}^H \\ \mathbf{R}_{\tilde{\mathbf{x}}_0(t)\bar{\mathbf{x}}_0(t)} & \mathbf{R}_{\tilde{\mathbf{x}}_0(t)} \end{pmatrix}. \tag{4.11}$$

Using the lemma in the appendix, $\mathbf{R}_{\mathbf{x}}^{-1}$ can be written as

$$\mathbf{R}_{\mathbf{x}_0}^{-1} = \begin{pmatrix} \mathbf{R}_{\bar{\mathbf{x}}_0(t)}^{-1} + \mathbf{L}_{21}^H \mathbf{S}^{-1}\mathbf{L}_{21} & -\mathbf{L}_{21}^H\mathbf{S}^{-1} \\ -\mathbf{S}^{-1}\mathbf{L}_{21} & \mathbf{S}^{-1} \end{pmatrix} := \begin{pmatrix} \mathbf{E} & \mathbf{F} \\ \mathbf{G} & \mathbf{H} \end{pmatrix} \tag{4.12}$$

where

$$\mathbf{L}_{21} = \mathbf{R}_{\tilde{\mathbf{x}}_0(t)\bar{\mathbf{x}}_0(t)}\mathbf{R}_{\bar{\mathbf{x}}_0(t)}^{-1} \tag{4.13}$$

$$\mathbf{S} = \mathbf{R}_{\tilde{\mathbf{x}}_0(t)} - \mathbf{R}_{\tilde{\mathbf{x}}_0(t)\bar{\mathbf{x}}_0(t)}\mathbf{R}_{\bar{\mathbf{x}}_0(t)}^{-1}\mathbf{R}_{\tilde{\mathbf{x}}_0(t)\bar{\mathbf{x}}_0(t)}^H. \tag{4.14}$$

Using (4.12), $\mathbf{R}_{\mathbf{x}_0}^{-1}\mathbf{r}_{\mathbf{x}_0 s_0}$ can be written as

$$\begin{pmatrix} \mathbf{E} & \mathbf{F} \\ \mathbf{G} & \mathbf{H} \end{pmatrix} \begin{pmatrix} \mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0} \\ \mathbf{r}_{\tilde{\mathbf{x}}_0(t)s_0} \end{pmatrix} = \begin{pmatrix} \mathbf{E}\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0} + \mathbf{F}\mathbf{r}_{\tilde{\mathbf{x}}_0(t)s_0} \\ \mathbf{G}\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0} + \mathbf{H}\mathbf{r}_{\tilde{\mathbf{x}}_0(t)s_0} \end{pmatrix} \tag{4.15}$$

and consequently

$$\left(\mathbf{R}_{\mathbf{x}_0}^{-1}\mathbf{r}_{\mathbf{x}_0 s_0}\right)^H \mathbf{z}_0(t) = \left(\mathbf{E}\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0}\right)^H \bar{\mathbf{x}}_0(t) + \left(\mathbf{F}\mathbf{r}_{\tilde{\mathbf{x}}_0(t)s_0}\right)^H \bar{\mathbf{x}}_0(t). \tag{4.16}$$

Substituting $\left(\mathbf{R}_{\bar{\mathbf{x}}_0}^{-1} + \mathbf{F}\mathbf{H}^{-1}\mathbf{G}\right)$ for $\mathbf{E}$ in (4.16) (see Appendix C), yields

$$\left(\mathbf{R}_{\mathbf{x}_0}^{-1}\mathbf{r}_{\mathbf{x}_0 s_0}\right)^H \mathbf{z}_0(t) = \left(\mathbf{R}_{\bar{\mathbf{x}}_0(t)}^{-1}\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0}\right)^H \bar{\mathbf{x}}_0(t) + \left(\mathbf{F}\mathbf{H}^{-1}\mathbf{G}\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0}\right)^H \bar{\mathbf{x}}_0(t) + \left(\mathbf{F}\mathbf{r}_{\tilde{\mathbf{x}}_0(t)s_0}\right)^H \bar{\mathbf{x}}_0(t). \tag{4.17}$$

Comparing (4.10) with (4.17), the update at time t is

$$\mathbf{u}(t) = \left(\mathbf{F}\mathbf{H}^{-1}\mathbf{G}\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0}\right)^H \bar{\mathbf{x}}_0(t) + \left(\mathbf{F}\mathbf{r}_{\tilde{\mathbf{x}}_0(t)s_0}\right)^H \bar{\mathbf{x}}_0(t). \tag{4.18}$$

This update needs to be subtracted from the complete data filter output to yield the optimal LMMSE estimate based on $\bar{\mathbf{x}}_0(t)$.

In summary, per slot t, the LMMSE solution $\widehat{\bar{s}}_0(t)$ based on $\bar{\mathbf{x}}_0(t)$ can be obtained as follows:

1. Given $\{\mathbf{R}_{\mathbf{x}_0}, \mathbf{r}_{\mathbf{x}_0 s_0}\}$, find $\mathbf{w}_{\mathbf{x}_0}$ as in (4.5) and compute $\mathbf{w}_{\mathbf{x}_0}^H \mathbf{z}_0(t)$.
2. Obtain update $\mathbf{u}(t)$ as in (4.18).
3. Find $\widehat{\bar{s}}_0(t) = \mathbf{w}_{\mathbf{x}_0}^H \mathbf{z}_0(t) - \mathbf{u}(t)$.

4.4 Discussion

4.4.1 Alternative Methods

Missing data is a common occurrence and may have noticeable effect on the final results drawn from observations. Deletion and imputation have been proposed to deal with missing data. In the deletion method, an entire record is excluded from analysis if any single value is missing [**?**]. Partial deletion is also possible. However, discarding missing values introduces bias in the results. Imputation is an alternate technique to mitigate bias effects. In this method, missing data are replaced with substituted values based on available observations [122]. Such replacement requires prior knowledge on the correlation statistics of the data.

4.4.2 Complexity

Assuming that the inverse of the full-data covariance matrix $\mathbf{R}_{\mathbf{x}_0}$ has been computed once, the complexity of our proposed update at time t is $\mathcal{O}((N-M(t))^3)+\mathcal{O}((N-M(t))M(t))$. The first term is due to the need to invert the matrix $\mathbf{H}$, and the second term is due to the matrix-vector products. Compared to the direct inversion of an $M(t)\times M(t)$ matrix $\mathbf{R}_{\bar{\mathbf{x}}_0(t)}$ which has complexity $\mathcal{O}(M^3(t))$, the complexity of our proposed method is much lower especially if the amount of missing data $N-M(t)$ is small.

4.4.3 MSE Comparison

To evaluate the impact of missing data on the MSE performance, we list below the achievable MSE in different scenarios:

Case 1: *No Missing Data.* If there is no missing data, the achievable MMSE is given by (c. f. (4.6))

$$\sigma^2_{\varepsilon_0} := \sigma^2_{s_0} - \mathbf{r}^H_{\mathbf{x}_0 s_0}\mathbf{R}^{-1}_{\mathbf{x}_0}\mathbf{r}_{\mathbf{x}_0 s_0}. \tag{4.19}$$

Case 2: *Missing Data with Partial Deletion.* If missing data are simply set to zero, and then a complete-data LMMSE filter is applied to $\mathbf{z}_0(t)$, the achieved MSE is given by

$$\sigma^2_{s_0} - 2\,\mathrm{Re}\left\{\bar{\mathbf{w}}^H(t)\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0}\right\} + \bar{\mathbf{w}}^H(t)\mathbf{R}^{-1}_{\mathbf{x}_0}\bar{\mathbf{w}}(t) \tag{4.20}$$

where $\bar{\mathbf{w}}(t) = \mathbf{D}(t)\mathbf{R}^{-1}_{\mathbf{x}_0}\mathbf{r}_{\mathbf{x}_0 s_0}$, and $\mathbf{D}(t)$ is a diagonal matrix with 0 at (i,i)th entry if $\mathbf{e}_i^T\mathbf{x}(t)$ is missing and one elsewhere.

Case 3: *LMMSE accounting for Missing Data.* Applying the LMMSE estimator such as in our proposed method on the incomplete data, the achieved MSE is

$$\sigma^2_{\bar{\varepsilon}_0}(t) := \sigma^2_{s_0} - \mathbf{r}^H_{\bar{\mathbf{x}}_0(t)s_0}\mathbf{R}^{-1}_{\bar{\mathbf{x}}_0(t)}\mathbf{r}_{\bar{\mathbf{x}}_0(t)s_0}. \tag{4.21}$$

Note that the MSE in Cases 1 and 3 are indeed the (linear) MMSE, but this is not generally true for Case 2.

Case 4: *LMMSE imputation.* If the missing entries are replaced with their LMMSE estimates based on the incomplete data, then the complete-data LMMSE filter $\mathbf{w}_{\mathbf{x}_0}$ can be applied

to the completed data vector. In this case, the LMMSE estimate of the missing entries based on the incomplete data is given by

$$\hat{\bar{\mathbf{x}}}_0(t) = (\mathbf{R}^{-1}_{\bar{\mathbf{x}}_0(t)}\mathbf{R}_{\bar{\mathbf{x}}(t)\tilde{\mathbf{x}}(t)})^H \bar{\mathbf{x}}_0(t) = \mathbf{L}_{21}\bar{\mathbf{x}}_0(t), \tag{4.22}$$

where $\mathbf{L}_{21}$ is as in (4.13). The completed data vector with the imputed entries is

$$\hat{\mathbf{x}}_0(t) = \begin{bmatrix} \mathbf{I} \\ \mathbf{L}_{21} \end{bmatrix} \bar{\mathbf{x}}_0(t). \tag{4.23}$$

Applying the complete-data LMMSE filter $\mathbf{w}_{\mathbf{x}_0} = \mathbf{R}^{-1}_{\mathbf{x}_0}\mathbf{r}_{\mathbf{x}_0 s_0}$ (cf. (4.5)) to $\hat{\mathbf{x}}_0(t)$ yields the estimate of $s_0(t)$ as

$$\mathbf{w}^H_{\mathbf{x}_0}\hat{\mathbf{x}}_0(t) = \mathbf{r}^H_{\mathbf{x}_0 s_0}\mathbf{R}^{-1}_{\mathbf{x}_0} \begin{bmatrix} \mathbf{I} \\ \mathbf{L}_{21} \end{bmatrix} \bar{\mathbf{x}}_0(t), \tag{4.24}$$

which by using the expression of $\mathbf{R}^{-1}_{\mathbf{x}_0}$ in (4.12) can be verified to be equal to

$$\mathbf{r}^H_{\mathbf{x}_0 s_0} \begin{bmatrix} \mathbf{R}^{-1}_{\bar{\mathbf{x}}_0(t)} \\ 0 \end{bmatrix} \bar{\mathbf{x}}_0(t) = \mathbf{r}^H_{\bar{\mathbf{x}}_0 s_0}\mathbf{R}^{-1}_{\bar{\mathbf{x}}_0(t)}\bar{\mathbf{x}}_0(t). \tag{4.25}$$

Note that this is exactly the LMMSE estimate of $s_0(t)$ based on the incomplete data only. We summarize the result in the following proposition.

Proposition 4.1. *The LMMSE estimate of s_0 obtained by applying the complete-data LMMSE filter vector $\mathbf{w}_{\mathbf{x}_0}$ to the completed data vector $\hat{\mathbf{x}}_0(t)$ with LMMSE imputation for missing entries is identical to the LMMSE estimate of s_0 based on the incomplete data alone.*

We note that although LMMSE imputation enables us to achieve the same MSE as that of the LMMSE estimation based directly on the incomplete data, estimating the missing data does incur additional complexity. The computational complexity of $\mathbf{R}^{-1}_{\bar{\mathbf{x}}_0(t)}$ as required in the imputation process can be reduced by leveraging the same technique as in our proposed reduced-complexity LMMSE scheme in Section 4.3. Such imputation may be useful in cases where the missing entries in the data vectors also need to be estimated in addition to the unknown signal $s_0(t)$.

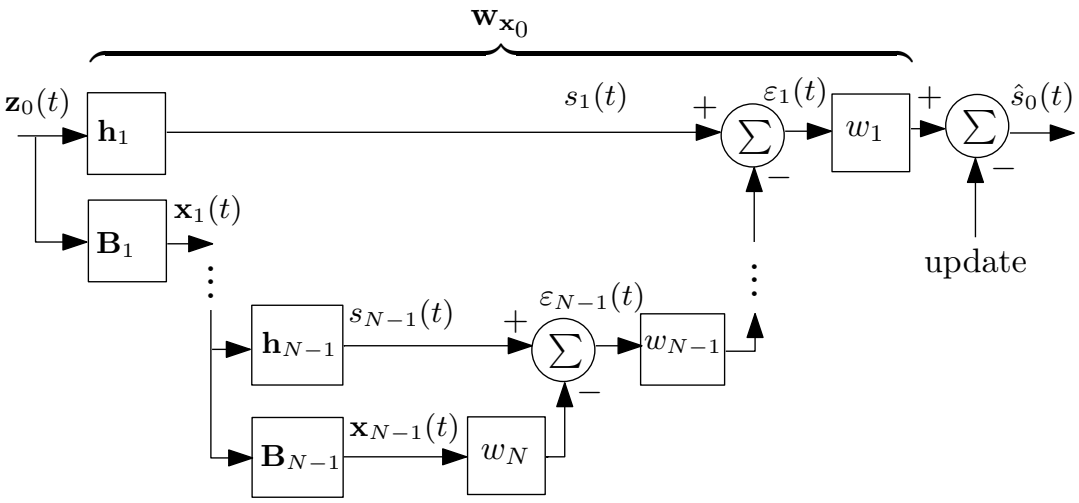

Figure 4.2: The nested chain of Wiener filters

4.4.4 MSWF Update

The MSWF was developed in [34] as an LMMSE solution that does not require inversion of the observed-data covariance matrix. In MSWF, a series of transformations is applied to the input data so that the resulting components associated with the cross-correlation of each stage are recognized and separated. As such, the MSWF avoids the explicit inversion of the data covariance matrix, and thus reduces the complexity.

If the observation data has missing values however, the MSWF is not optimal. Applying the MSWF to the data vector with missing entries (whose values are set to zero) result in suboptimal performance, corresponding to Case 2 discussed in Section 4.4.3.

The multistage method works by applying the following recursion N times with increasing i (see Figure 4.2)

$$s_i(t) = \mathbf{h}_i^H \mathbf{x}_{i-1}(t) \tag{4.26}$$

$$\text{where } \mathbf{h}_i \propto \mathbb{E}[\mathbf{x}_{i-1}(t) s_{i-1}(t)], \quad \|\mathbf{h}_i\| = 1 \tag{4.27}$$

$$\mathbf{x}_i(t) = \mathbf{B}_i^H \mathbf{x}_{i-1}(t), \quad \mathbf{B}_i^H \mathbf{h}_i = 0 \tag{4.28}$$

and applying the backward recursion

$$\varepsilon_i(t) = s_i(t) - w_{i+1}^* \varepsilon_{i+1}(t), \quad i = N-1, N-2 \ldots, 1,$$

with appropriately chosen scalars w_{i+1}.

When there are missing data, it is non-trivial to derive the MSWF coefficients for the incomplete data vector, without recomputing all coefficients involved. More precisely, for each time slot when entries of the observed data are missing, we must first remove the missing entries from the data (dimensionality reduction) before computing the MSWF corresponding to the data without missing entries.

The described update in Section 4.3 addresses this challenge especially when the input signals are of high dimensions. In fact, provided that the inverse of $\mathbf{R}_{\mathbf{x}_0}$ is computed once, instead of re-deriving the MSWF coefficients per time slot, one only needs a simple update on the MSWF involving the covariance matrix inverse corresponding to the missing data.

4.4.5 Numerical Example

For illustration purpose, we provide a small numerical example to demonstrate the effect of missing data on the overall achievable MSE. For $\rho > 1$, let $[\mathbf{R}_{\mathbf{x}_0}]_{i,j} = \rho^{-|i-j|}$, $N = 15$, $\sigma_{s_0}^2 = 1$, and

$$[\mathbf{r}_{\mathbf{x}_0 s_0}]_i = a \cos\left(2\pi \frac{i - \frac{N}{2}}{4N-1}\right), \quad i \in \{1, \ldots, N\} \tag{4.29}$$

where a is an adjustable parameter. We will choose $\rho = 1.2$ in our example. For simplicity, it is assumed that the last four entries of $\mathbf{x}_0(t)$ are missing per time slot. Running simulations based on (4.19)–(4.20) for different values of a we arrive at Table 4.1, where the improvement caused by the update is noticeable and significant for small MMSE.

4.5 Summary

We proposed a method to perform optimal LMMSE filtering of observation data with missing entries whose positions may be time varying. Our method applies the complete-data MMSE filter to the partially-deleted data (setting missing entries to zero) and then updates the filter output with an additional term. The proposed method offers reduced complexity per time slot when the number of missing entries in a time slot is small. It is possible to apply the

Table 4.1: The impact of missing data on MSE results

a	Case#1	Case#2	Case#3
0.55	0.4949	0.5525	0.5246
0.60	0.3989	0.4674	0.4343
0.65	0.2945	0.3749	0.3360
0.70	0.1818	0.2751	0.2300
0.75	0.0608	0.1678	0.1160

proposed update to e.g., multi-stage Wiener filter, so that optimality in MSE is maintained when processing data with missing entries.

It would be interesting to investigate the filter design problem without the knowledge of missing entries positions. Moreover, finding an update when the observed covariance matrix is rank-deficient would also be interesting.

CHAPTER 5. CONCLUSION AND FUTURE WORKS

We have mainly considered modeling and learning of networks based on measurements, specifically gene regulatory network inference and high-dimensional covaraince approximation. Furthermore, we have contributed to the development of low-complex inference algorithms responsible for high-dimensional data processing.

We have introduced a step-wise framework for gene regulatory network discovery from dynamical expression data that result from genetic or chemical perturbation of a steady state system. The ordinary differential equations used to model eukaryotic gene regulation presents the new generalization of a thermodynamic and statistical mechanic approach to polymerase binding. We have established robust and low-complexity algorithms to infer gene regulatory networks, which is best suited for the genetic perturbation. However, we have shown that this approach can still work under non-ideal conditions. Notably, our procedure allows us for the modifications of its steps to improve the network inference, for instance alternative methods for change detection. We can also improve the inference performance with a priori knowledge measured across biological systems, such as protein and RNA degradation rates. Our method can exhibit promising results for such real datasets that suffer from the absence of information relating to degradation rates, contextless inference, and the non step-wise nature of changes in gene expression. Our work can be further extended as follows:

- It is not always possible to construct gene regulatory networks based on gene expression profiling experiments. For instance, the availability of small number of noisy observations can potentially create non-identifiable problems. Moreover, biological structures are the dominant factor to determine gene expression levels and the ability of network inference will significantly depend on the complexity of systems. Therefore, gene expression data may not be enough to identify networks. This present issue, that is investigated as identi-

fiability of gene regulatory networks in chapter 2, leads to the question: what additional biological and experimental data can be used to guarantee network identifiability? In fact, new sources of information measured across biological systems may be considered together with gene expression data, but they must be analyzed to deduce what type of data is appropriate for the purpose of network inference.

- It would be interesting to propose a Kalman filter based approach to identification of gene regulatory networks that is suitable for our system model [123]. This direction is indeed beneficial since the Kalman filter method is an online iterative algorithm with the capability of estimating a large number of parameters based on a few observations.

We have also investigated the problem of high-dimensional covariance matrix estimation based on partial observations. We have proposed a convex optimization approach suitable for any missing data patterns to estimate covariance matrices with Kronecker product structure. We have shown that the estimated covariance is positive definite with a probability close to one for an appropriate number of samples. Furthermore, we have established a spectral norm bound and a square error bound to elucidate the performance of the proposed method. Our scheme achieves high-dimensional consistency with a convergence rate quite faster than the standard sample covariance matrix. Mathematical derivations has presented to reveal consequences of missing data on the performance and numerical simulations has taken into account to verify our results. There remain several open problems as follows:

- We would like to establish an inner bound on the SE or mean-square error performance of our method and compare it with the SE outer bound. This analysis will elucidate whether the proposed procedure is optimal or an opportunity to reach better performance is possible. We believe that the Cramer-Rao lower bound intended for biased estimators could be taken into account to derive the inner bound.

- It is desired to demonstrate the performance of the proposed method on real world applications that suffer from missing data. We suggest that wind speed datasets with missing values are appropriate for our study [21].

Finally, we have developed an algorithm to perform the optimal linear minimum mean-square error filter for high-dimensional observation vectors that contain missing values. The proposed method presents lower complexity compared to re-deriving the filter whenever the position of missing data changes. We have shown that our method can be applied to multistage Wiener filter and preserves its optimality. Future research directions are as follows:

- In chapter 4, we have assumed that the position of missing data are available. However, the filter design problem when the position of missing entries are unknown remains open.
- we would like to construct updates according to specific covariance structures, such as low rank and sparse properties, to improve computational complexities.

APPENDIX A. SUPPORTING INFORMATION FOR CHAPTER 2

Treatment of protein regulators

Consider a gene for which the probability of RNAP being bound to a specific promoter site, S, is under the potential influence of a single non-steady state regulator, Regulator 1, and the collection of all available regulators still in steady state. The steady state regulators are encapsulated as a single super-protein complex, SS, that is fixed as bound to the promoter region. Suppose that we have P RNAP, R_1 Regulator 1, and R_{SS} super-protein complex.

We apply the following notation: ε_P^{NS} is used to denote the energy of the case in which RNAP is bound to a non-specific (NS) DNA binding site, $\varepsilon_{P,i0}^{S}$ the energy when RNAP is only bound to the S binding site, $\varepsilon_{P,i1}^{S}$ the energy when RNAP is specifically bound to the promoter-regulator complex, ε_{SS}^{NS} the energy when the SS is bound to the NS binding site, ε_{SS}^{S} the energy when the SS is bound to the S binding site, ε_{i1}^{NS} the energy when Regulator 1 is bound to the NS binding site, ε_{i1}^{S} the energy when Regulator 1 is bound to the S binding site, and

$$\Delta\varepsilon_{P,i0} := \varepsilon_{P,i0}^{S} - \varepsilon_P^{NS}, \Delta\varepsilon_{P,i1} := \varepsilon_{P,i1}^{S} - \varepsilon_P^{NS}, \Delta\varepsilon_{i1} := \varepsilon_{i1}^{S} - \varepsilon_{i1}^{NS}.$$

Also define

$$Z(P, R_1, R_{SS} - 1) := \frac{m! e^{-P\beta\varepsilon_P^{NS}} e^{-R_1\beta\varepsilon_{i1}^{NS}} e^{-(R_{SS}-1)\beta\varepsilon_{SS}^{NS}} e^{-\beta\varepsilon_{SS}^{S}}}{P! R_1! (R_{SS}-1)! (m - P - R_1 - R_{SS} + 1)!},$$

where $Z(P, R_1, R_{SS} - 1)$ gives the total number of arrangements for RNAP and R1 at NS binding sites, weighted by a Boltzmann factor providing a relative energy for each state.

The available configurations of the system with corresponding unnormalized probabilities are enumerated as follows: (i) Regulator 1 and RNAP unbound: $Z(P, R_1, R_{SS} - 1)$, (ii) only Regulator 1 bound: $Z(P, R_1 - 1, R_{SS} - 1)e^{-\beta\varepsilon_{i1}^{S}}$, (iii) only RNAP bound: $Z(P - 1, R_1, R_{SS} -$

$1)e^{-\beta\varepsilon^S_{P,i0}}$, and (iv) both Regulator 1 and RNAP bound: $Z(P-1,R_1-1,R_{SS}-1)e^{-\beta\varepsilon^S_{P,i1}}$. To derive the probability of RNAP binding, we sum the probabilities of configurations in which RNAP is bound to the specific site and divide over the sum of probabilities of all potential configurations, Z_{total}. Here, in parallel to [15], it is shown how the effect of steady state proteins can effectively be removed from the protein regulator formulation, under the aforementioned arrangement. To represent the probability of RNAP binding to the cis regulatory region of gene i, we define p_i^{bound} as follows.

$$
\begin{aligned}
p_i^{\text{bound}} &= \left(Z(P-1,R_1,R_{SS}-1)e^{-\beta\varepsilon^S_{P,i0}} + Z(P-1,R_1-1,R_{SS}-1)e^{-\beta\varepsilon^S_{P,i1}}\right)\Big/ Z_{\text{total}} \\
&= \left(Z(P-1,R_1,R_{SS}-1)e^{-\beta\varepsilon^S_{P,i0}} + Z(P-1,R_1-1,R_{SS}-1)e^{-\beta\varepsilon^S_{P,i1}}\right)\Big/ \\
&\quad \Big(Z(P,R_1,R_{SS}-1) + Z(P,R_1-1,R_{SS}-1)e^{-\beta\varepsilon^S_{i1}} + Z(P-1,R_1,R_{SS}-1)e^{-\beta\varepsilon^S_{P,i0}} \\
&\quad + Z(P-1,R_1-1,R_{SS}-1)e^{-\beta\varepsilon^S_{P,i1}}\Big) \\
&= \left(\frac{m}{R_1}e^{\beta\varepsilon^{NS}_P}e^{-\beta\varepsilon^S_{P,i0}} + e^{\beta\varepsilon^{NS}_P}e^{\beta\varepsilon^{NS}_{i1}}e^{-\beta\varepsilon^S_{P,i1}}e^{-\beta\varepsilon^S_{i1}}\right)\Big/\left(\frac{m}{R_1}e^{\beta\varepsilon^{NS}_P}e^{-\beta\varepsilon^S_{P,i0}} + e^{\beta\varepsilon^{NS}_P}e^{\beta\varepsilon^{NS}_{i1}}e^{-\beta\varepsilon^S_{P,i1}}e^{-\beta\varepsilon^S_{i1}}\right. \\
&\quad \left. + \frac{m^2}{PR_1} + \frac{m}{P}e^{\beta\varepsilon^{NS}_{i1}}e^{-\beta\varepsilon^S_{i1}}\right)
\end{aligned}
$$

$$
= \frac{\frac{1}{y_1}e^{-\beta\Delta\varepsilon_{P,i0}} + e^{-\beta\Delta\varepsilon_{P,i1}}e^{-\beta\Delta\varepsilon_{i1}}}{\frac{1}{y_1}e^{-\beta\Delta\varepsilon_{P,i0}} + e^{-\beta\Delta\varepsilon_{P,i1}}e^{-\beta\Delta\varepsilon_{i1}} + \frac{1}{Py_1} + \frac{1}{P}e^{-\beta\Delta\varepsilon_{i1}}}
$$

$$
= \frac{Pe^{-\beta\Delta\varepsilon_{P,i0}} + y_1Pe^{-\beta\Delta\varepsilon_{P,i1}}e^{-\beta\Delta\varepsilon_{i1}}}{Pe^{-\beta\Delta\varepsilon_{P,i0}} + y_1Pe^{-\beta\Delta\varepsilon_{P,i1}}e^{-\beta\Delta\varepsilon_{i1}} + 1 + y_1e^{-\beta\Delta\varepsilon_{i1}}}
$$

$$
= \frac{Pe^{-\beta\Delta\varepsilon_{P,i0}} + y_1Pe^{-\beta\Delta\varepsilon_{P,i1}}e^{-\beta\Delta\varepsilon_{i1}}}{(1+Pe^{-\beta\Delta\varepsilon_{P,i0}}) + y_1e^{-\beta\Delta\varepsilon_{i1}}(1+Pe^{-\beta\Delta\varepsilon_{P,i1}})}
$$

where we have applied the approximation

$$
m!/P!R_1!(R_{SS}-1)!(m-P-R_1-R_{SS}+1)! \approx m^P m^{R_1} m^{R_{SS}}/P!R_1!(R_{SS}-1)!.
$$

We introduce y_1, the protein product of Regulator 1 defined as R_1/m, for the purposes of normalization and in keeping with the protein designations used throughout this paper. We

additionally note that P in the final steps of the derivation above is also normalized to m, but we retain the same notation for simplicity.

The final derivation can be generalized to, for an indefinite number of first and second order regulators.

$$f_i(\mathcal{Y}_{G(t)}) = \frac{\sum_{j=0}^{N(t)} P e^{-\beta\Delta\varepsilon_{P,ij}} e^{-\beta\Delta\varepsilon_{ij}} \prod_{k\in S_{ij}(t)} y_k(t)}{\sum_{j=0}^{N(t)} (1 + P e^{-\beta\Delta\varepsilon_{P,ij}}) e^{-\beta\Delta\varepsilon_{ij}} \prod_{k\in S_{ij}(t)} y_k(t)}, \tag{A.1}$$

where $\Delta\varepsilon_{ij}$ is the binding energy of the jth complex to the promoter, $\Delta\varepsilon_{P,ij}$ is the energy of RNAP being bound to the promoter-regulator complex j, and P is the concentration of RNAP. Setting $a_{ij} = P e^{-\beta\Delta\varepsilon_{P,ij}} e^{-\beta\Delta\varepsilon_{ij}}$ and $b_{ij} = (1 + P e^{-\beta\Delta\varepsilon_{P,ij}}) e^{-\beta\Delta\varepsilon_{ij}}$, we arrive at the form given in the section Protein-Mediated Regulation.

B-splines

B-splines have been well investigated in approximation theory and numerical analysis, leading to a variety of important properties such as computational efficiency and numerical stability. Particularly, the B-spline basis functions have the best approximation capacity based on the Stone-Weierstrass Approximation Theorem. Polynomial functions are also used to estimate continuous functions. However, the B-spline bases are shown to be optimally stable [124].

A set of B-spline basis functions in variable t is determined by the degree of a piecewise polynomial, P, and a knot sequence [125]. The knot sequence is a set of points that divides a real interval into a number of sub-intervals. More precisely, D bases of degree P are parameterized by $D+P+1$ knots, $\{t_0, t_1, \ldots, t_{D+P}\}$ where $t_0 \leq t_1 \leq \ldots \leq t_{D+P}$. Employing this set of knots and the De Boor recursion in [126], the dth B-spline basis of degree P, written as $\varphi_d^{(P)}(t)$, is derived recursively as follows:

$$\varphi_d^{(0)}(t) = \begin{cases} 1 & \text{if } t_{d-1} \leq t \leq t_d \\ 0 & \text{if otherwise} \end{cases}, \tag{A.2}$$

$$\varphi_d^{(p)}(t) = \frac{t - t_{d-1}}{t_{p+d-1} - t_{d-1}} \varphi_d^{(p-1)}(t) + \frac{t_{p+d} - t}{t_{p+d} - t_d} \varphi_{d+1}^{(p-1)}(t), \tag{A.3}$$

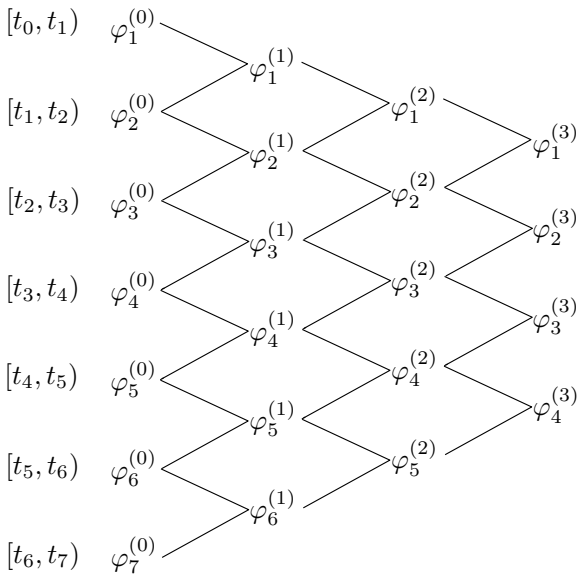

Figure A.1: The De Boor recursion for $P = 3$ and $D = 4$.

for $1 \leq d \leq D + P - p$ where $p = 0$ in (A.2) and $1 \leq p \leq P$ in (A.3). The above recursion is visualized in Figure A.1 (reconstructed from [125]).

The degree $P = 3$ or 4 is sufficient in most applications. The number of basis functions should be large enough to arrive at accurate estimation but not too large to cause overfitting. In our case, gene and protein levels do not contain high frequency changes and therefore, a small number of basis functions are sufficient to represent gene and protein expressions.

Bi-Convex Problems

Bi-convex optimization is a generalization of convex optimization where the objective function and the constraint set can be bi-convex [66].

Definition A.1. *Let $\mathcal{X} \subseteq \mathbb{R}^n$ and $\mathcal{Y} \subseteq \mathbb{R}^n$ be two non-empty convex sets. The set $E \subseteq \mathcal{X} \times \mathcal{Y}$ is called bi-convex if $B_x := \{y \in \mathcal{Y} : (x, y) \in B\}$ is convex for each x, and $B_y := \{x \in \mathcal{X} : (x, y) \in B\}$ is convex for each y.*

Definition A.2. *A function $f(x, y) : B \to \mathbb{R}$ is called bi-convex if $f(x, y)$ is convex on B_x for every fixed x and also convex on B_y for every fixed y.*

A common method to solve a bi-convex problem is ADMM [127]. The ADMM is an iterative augmented Lagrangian method that uses partial updates for dual variables and replaces joint minimization by simpler sub-problems. However, the mentioned procedure does not guarantee global optimality of the solution.

Proof of Theorem 1

The stationary points $\{\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i\}$ of (P3) are derived by setting sub-gradients to zero as follows

$$\nabla_{\mathbf{a}_i}\Gamma(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i) = 2\sum_{l=1}^{L}\Omega_l(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i)\mathbf{p}_i(t_l) = \mathbf{0} \tag{A.4}$$

$$\nabla_{\mathbf{b}_i}\Gamma(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i) = -2\sum_{l=1}^{L}\Omega_l(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i)\mathbf{u}_i^T(t_l)\bar{\boldsymbol{\lambda}}_i\mathbf{p}_i(t_l) + \gamma_1\bar{\mathbf{b}}_i + \gamma_2\,\mathrm{sign}(\bar{\mathbf{b}}_i) = \mathbf{0} \tag{A.5}$$

$$\nabla_{\boldsymbol{\lambda}_i}\Gamma(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i) = -2\sum_{l=1}^{L}\Omega_l(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i)\mathbf{p}_i^T(t_l)\bar{\mathbf{b}}_i\mathbf{u}_i(t_l) + \gamma_1\bar{\boldsymbol{\lambda}}_i + \gamma_3\,\mathrm{sign}(\bar{\boldsymbol{\lambda}}_i) = \mathbf{0} \tag{A.6}$$

with respect to constraints $0 \leq \bar{\mathbf{a}}_i \leq \bar{\mathbf{b}}_i$ and $\bar{\boldsymbol{\lambda}}_i \geq 0$. These constraints admit that $\mathrm{sign}(\cdot)$ can be replaced by vector $\mathbf{1}$ in the above equations. It is obvious from (A.5)–(A.6) that $\bar{\mathbf{b}}_i^T\nabla_{\mathbf{b}_i}\Omega(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i) = \bar{\boldsymbol{\lambda}}_i^T\nabla_{\boldsymbol{\lambda}_i}\Omega(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i) = 0$, which results in

$$2\sum_{l=1}^{L}\Omega_l(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i, \bar{\boldsymbol{\lambda}}_i)\mathbf{p}_i^T(t_l)\bar{\mathbf{b}}_i\mathbf{u}_i^T(t_l)\bar{\boldsymbol{\lambda}}_i = \gamma_1\bar{\mathbf{b}}_i^T\bar{\mathbf{b}}_i + \gamma_2\bar{\mathbf{b}}_i^T\mathbf{1} = \gamma_1\bar{\boldsymbol{\lambda}}_i^T\bar{\boldsymbol{\lambda}}_i + \gamma_3\bar{\boldsymbol{\lambda}}_i^T\mathbf{1}. \tag{A.7}$$

Consider the convex optimization

$$\text{(P5)} \quad \min_{\{\mathbf{a}_i, \mathbf{G}_i, \mathbf{W}_1, \mathbf{W}_2\}} \sum_{l=1}^{L}\left(\mathbf{p}_i^T(t_l)\mathbf{a}_i - \mathbf{u}_i^T(t_l)\mathbf{G}_i\mathbf{p}_i(t_l)\right)^2 + \gamma_1\kappa(\mathbf{W}_1, \mathbf{W}_2)$$

$$\text{subject to} \quad \mathbf{W} := \begin{pmatrix} \mathbf{W}_1 & \mathbf{G}_i \\ \mathbf{G}_i^T & \mathbf{W}_2 \end{pmatrix} \succeq 0,$$

where $\kappa(\mathbf{W}_1, \mathbf{W}_2) := \frac{1}{2}\left(\mathrm{Tr}(\mathbf{W}_1) + \mathrm{Tr}(\mathbf{W}_2)\right)$.

Minimizing (P5) with respect to $\{\mathbf{W}_1, \mathbf{W}_2\}$ leads to

$$\|\mathbf{G}_i\|_* = \min_{\{\mathbf{W}_1, \mathbf{W}_2\}} \kappa(\mathbf{W}_1, \mathbf{W}_2) \quad \text{subject to} \quad \mathbf{W} \succeq 0,$$

which is the alternative characterization of the nuclear norm [95]. Taking advantage of the nuclear norm, we can restrict matrix $\mathbf{G}_i$ to be rank one as $\boldsymbol{\lambda}_i\mathbf{b}_i^T$. Also, $\kappa(\cdot,\cdot)$ is able to satisfy the required sparsity for $\{\boldsymbol{\lambda}_i, \mathbf{b}_i^T\}$. To investigate these claims, recall constraints (**??**) and set $\mathbf{G}_i := \boldsymbol{\lambda}_i\mathbf{b}_i^T$, $\mathbf{W}_1 := \boldsymbol{\lambda}_i\boldsymbol{\lambda}_i^T + \frac{\gamma_3}{\gamma_1}\operatorname{diag}(\boldsymbol{\lambda}_i)$, and $\mathbf{W}_2 := \mathbf{b}_i\mathbf{b}_i^T + \frac{\gamma_2}{\gamma_1}\operatorname{diag}(\mathbf{b}_i)$ where $\operatorname{diag}(\boldsymbol{\lambda}_i)$ is the diagonal matrix with (j,j)th entry equal to $\boldsymbol{\lambda}_i(j)$. Then, the triple $(\mathbf{G}_i, \mathbf{W}_1, \mathbf{W}_2)$ is feasible for (P5) due to

$$\begin{aligned}\begin{pmatrix}\mathbf{W}_1 & \mathbf{G}_i\\ \mathbf{G}_i^T & \mathbf{W}_2\end{pmatrix} &= \begin{pmatrix}\boldsymbol{\lambda}_i\boldsymbol{\lambda}_i^T + \frac{\gamma_3}{\gamma_1}\operatorname{diag}(\boldsymbol{\lambda}_i) & \boldsymbol{\lambda}_i\mathbf{b}_i^T\\ \mathbf{b}_i\boldsymbol{\lambda}_i^T & \mathbf{b}_i\mathbf{b}_i^T + \frac{\gamma_2}{\gamma_1}\operatorname{diag}(\mathbf{b}_i)\end{pmatrix}\\ &= \begin{pmatrix}\boldsymbol{\lambda}_i\\ \mathbf{b}_i\end{pmatrix}\begin{pmatrix}\boldsymbol{\lambda}_i\\ \mathbf{b}_i\end{pmatrix}^T + \frac{1}{\gamma_1}\begin{pmatrix}\gamma_3\operatorname{diag}(\boldsymbol{\lambda}_i) & 0\\ 0 & \gamma_2\operatorname{diag}(\mathbf{b}_i)\end{pmatrix} \succeq 0. \end{aligned} \tag{A.8}$$

In addition, we have

$$\gamma_1\kappa(\mathbf{W}_1,\mathbf{W}_2) = \gamma_1\left(\|\boldsymbol{\lambda}_i\|_2^2 + \|\mathbf{b}_i\|_2^2\right) + \gamma_2\|\mathbf{b}_i\|_1 + \gamma_3\|\boldsymbol{\lambda}_i\|_1,$$

and therefore the same objective function for (P3) and (P5) are obtained. This proves any feasible solution of (P5) yields an inner bound for (P3).

We next establish that the proposed inner bound is always equal to (P3) upon satisfying the condition introduced in Theorem 2.1 and conclude the two problems are equivalent. The equivalence ensures that the stationary point of (P3) (which exhibits Theorem 2.1 condition) is in fact globally optimal. To show this, the Lagrangian is first formed as

$$\mathcal{L}(\mathbf{G}_i,\mathbf{a}_i,\mathbf{W}_1,\mathbf{W}_2,\mathbf{M}) = \sum_{l=1}^{L}\left(\mathbf{p}_i^T(t_l)\mathbf{a}_i - \mathbf{u}_i^T(t_l)\mathbf{G}_i\mathbf{p}_i(t_l)\right)^2 + \gamma_1\kappa(\mathbf{W}_1,\mathbf{W}_2) - \langle\mathbf{M},\mathbf{W}\rangle,$$

and $\mathbf{M}$ indicates the dual variable associated with the constraint $\mathbf{W}\succeq 0$. In accordance with the block structure of $\mathbf{W}$ in (P5), we define $M_1 := [M]_{11}$, $M_2 := [M]_{12}$, $M_3 := [M]_{22}$, and $M_4 := [M]_{21}$. The optimal solution of (P5) must

(i) null the sub-gradients

$$\nabla_{\mathbf{a}_i}\mathcal{L}(\mathbf{G}_i,\mathbf{a}_i,\mathbf{W}_1,\mathbf{W}_2,\mathbf{M}) = 2\sum_{l=1}^{L}\left(\mathbf{p}_i^T(t_l)\mathbf{a}_i - \mathbf{u}_i^T(t_l)\mathbf{G}_i\mathbf{p}_i(t_l)\right)\mathbf{p}_i(t_l) \tag{A.9}$$

$$\nabla_{\mathbf{G}_i}\mathcal{L}(\mathbf{G}_i,\mathbf{a}_i,\mathbf{W}_1,\mathbf{W}_2,\mathbf{M}) = -2\sum_{l=1}^{L}\left(\mathbf{p}_i^T(t_l)\mathbf{a}_i - \mathbf{u}_i^T(t_l)\mathbf{G}_i\mathbf{p}_i(t_l)\right)\mathbf{u}_i(t_l)\mathbf{p}_i^T(t_l) - \mathbf{M}_2 - \mathbf{M}_4^T \tag{A.10}$$

$$\nabla_{\mathbf{W}_1}\mathcal{L}(\mathbf{G}_i,\mathbf{a}_i,\mathbf{W}_1,\mathbf{W}_2,\mathbf{M}) = \frac{\gamma_1}{2}\mathbf{I} - \mathbf{M}_1 \tag{A.11}$$

$$\nabla_{\mathbf{W}_2}\mathcal{L}(\mathbf{G}_i,\mathbf{a}_i,\mathbf{W}_1,\mathbf{W}_2,\mathbf{M}) = \frac{\gamma_1}{2}\mathbf{I} - \mathbf{M}_3 \tag{A.12}$$

and also satisfy

(ii) the complementary slackness condition $\langle \mathbf{M}, \mathbf{W}\rangle = 0$;

(iii) primal feasibility $\mathbf{W} \succeq 0$;

(iv) dual feasibility $\mathbf{M} \succeq 0$.

Consider the stationary points of (P3), and choose the candidate primal variables $\tilde{\mathbf{a}}_i := \bar{\mathbf{a}}_i$, $\tilde{\mathbf{G}}_i := \bar{\boldsymbol{\lambda}}_i\bar{\mathbf{b}}_i^T$, $\tilde{\mathbf{W}}_1 := \bar{\boldsymbol{\lambda}}_i\bar{\boldsymbol{\lambda}}_i^T + \frac{\gamma_3}{\gamma_1}\operatorname{diag}(\bar{\boldsymbol{\lambda}}_i)$, $\tilde{\mathbf{W}}_2 := \bar{\mathbf{b}}_i\bar{\mathbf{b}}_i^T + \frac{\gamma_2}{\gamma_1}\operatorname{diag}(\bar{\mathbf{b}}_i)$; and the dual variables $\tilde{\mathbf{M}}_1 := \frac{\gamma_1}{2}\mathbf{I}$, $\tilde{\mathbf{M}}_3 := \frac{\gamma_1}{2}\mathbf{I}$, $\tilde{\mathbf{M}}_2 := -\sum_{l=1}^{L}\Omega_l(\bar{\mathbf{a}}_i,\bar{\mathbf{b}}_i,\bar{\boldsymbol{\lambda}}_i)\mathbf{u}_i(t_l)\mathbf{p}_i^T(t_l)$, and $\tilde{\mathbf{M}}_4 := \tilde{\mathbf{M}}_2^T$. Then, condition (i) holds because the sub-gradients (A.9)–(A.12) are zero when substituting the introduced primal and dual variables. The requirement (ii) is also true since

$$\begin{aligned}
\langle\tilde{\mathbf{M}},\tilde{\mathbf{W}}\rangle &= \langle\tilde{\mathbf{M}}_1,\tilde{\mathbf{W}}_1\rangle + \langle\tilde{\mathbf{M}}_3,\tilde{\mathbf{W}}_2\rangle + 2\langle\tilde{\mathbf{M}}_2,\tilde{\mathbf{G}}_i\rangle \\
&= \frac{\gamma_1}{2}\operatorname{Tr}\left(\bar{\boldsymbol{\lambda}}_i\bar{\boldsymbol{\lambda}}_i^T + \frac{\gamma_3}{\gamma_1}\operatorname{diag}(\bar{\boldsymbol{\lambda}}_i)\right) + \frac{\gamma_1}{2}\operatorname{Tr}\left(\bar{\mathbf{b}}_i\bar{\mathbf{b}}_i^T + \frac{\gamma_2}{\gamma_1}\operatorname{diag}(\bar{\mathbf{b}}_i)\right) \\
&\quad - 2\operatorname{Tr}\left(\sum_{l=1}^{L}\Omega_l(\bar{\mathbf{a}}_i,\bar{\mathbf{b}}_i,\bar{\boldsymbol{\lambda}}_i)\mathbf{p}_i^T(t_l)\bar{\mathbf{b}}_i\mathbf{u}_i^T(t_l)\bar{\boldsymbol{\lambda}}_i\right) \\
&= \frac{1}{2}\operatorname{Tr}\left(\gamma_1\bar{\boldsymbol{\lambda}}_i\bar{\boldsymbol{\lambda}}_i^T + \gamma_3\operatorname{diag}(\bar{\boldsymbol{\lambda}}_i)\right) + \frac{1}{2}\operatorname{Tr}\left(\gamma_1\bar{\mathbf{b}}_i\bar{\mathbf{b}}_i^T + \gamma_2\operatorname{diag}(\bar{\mathbf{b}}_i)\right) - \operatorname{Tr}\left(\gamma_1\bar{\boldsymbol{\lambda}}_i\bar{\boldsymbol{\lambda}}_i^T + \gamma_3\operatorname{diag}(\bar{\boldsymbol{\lambda}}_i)\right) = 0,
\end{aligned}$$

where the last equality follows from (A.7). Moreover, (iii) is confirmed similar to (A.8). In order to meet the last criterion (iv), according to a Schur complement argument [64], it is sufficient to invoke $\|\tilde{\mathbf{M}}_2\| \le \gamma_1/2$.

Consequently, by choosing the proposed candidates that have been proved to be optimal, one can easily verify (P5) coincides with (P3). This completes the proof.

APPENDIX B. SUPPORTING INFORMATION FOR CHAPTER 3

Proof of Theorem 3.1

Matrix $\hat{\mathbf{\Sigma}}_n$ is the Hadamard product of two symmetric matrices (cf. (3.7)) and therefore is symmetric.

Using Lemma B.1, $\hat{\mathbf{\Sigma}}_n$ is positive definite if $\mathbf{\Sigma}_n^{\mathbf{\Gamma}}$ is positive definite (equivalently $n \geq d_1 d_2$) and $\mathbf{W}^{(-1)}$ (where all entries are positive) is positive semidefinite.

The last claim is proved based on techniques from compressed sensing and concentration of measure inequalities for Gaussian matrices, see Lemma 2 in [21] and Appendix A in [128]. Let $\mathbf{u} = (u_1, u_2, \ldots, u_{d_1 d_2})^T \in \mathcal{N}_{d_1 d_2}$ and define $\mathbf{\Lambda}_n \coloneqq \hat{\mathbf{\Sigma}}_n - \mathbf{\Sigma}_0$. Then, we have

$$\mathbf{u}^T \mathbf{\Lambda}_n \mathbf{u} = \mathbf{u}^T (\mathbf{W}^{(-1)} \circ \mathbf{\Sigma}_n^{\mathbf{\Gamma}}) \mathbf{u} - \mathbf{u}^T \mathrm{E}[\mathbf{W}^{(-1)} \circ \mathbf{\Sigma}_n^{\mathbf{\Gamma}}] \mathbf{u} \tag{B.1}$$

$$= \mathrm{tr}\left((\mathbf{W}^{(-1)} \circ \frac{1}{n} \sum_{t=1}^{n} \mathbf{z}_t \mathbf{z}_t^T) \mathbf{u}\mathbf{u}^T \right) - \mathrm{tr}\left(\mathrm{E}[\mathbf{W}^{(-1)} \circ \frac{1}{n} \sum_{t=1}^{n} \mathbf{z}_t \mathbf{z}_t^T] \mathbf{u}\mathbf{u}^T \right) \tag{B.2}$$

$$= \frac{1}{n} \sum_{t=1}^{n} \left(\mathbf{z}_t^T \mathbf{M} \mathbf{z}_t - \mathrm{E}[\mathbf{z}_t^T \mathbf{M} \mathbf{z}_t] \right) \tag{B.3}$$

$$= \frac{1}{n} \sum_{t=1}^{n} \left(\mathbf{x}_t^T \mathbf{\Gamma}_t^T \mathbf{M} \mathbf{\Gamma}_t \mathbf{x}_t - \mathrm{E}[\mathbf{x}_t^T \mathbf{\Gamma}_t^T \mathbf{M} \mathbf{\Gamma}_t \mathbf{z}_t] \right) \tag{B.4}$$

$$= \frac{1}{n} \sum_{t=1}^{n} \phi_t \tag{B.5}$$

where $\mathbf{M} \coloneqq \mathbf{W}^{(-1)} \circ (\mathbf{u}\mathbf{u}^T)$ and $\phi_t \coloneqq \mathbf{x}_t^T \mathbf{\Gamma}_t^T \mathbf{M} \mathbf{\Gamma}_t \mathbf{x}_t - \mathrm{E}[\mathbf{x}_t^T \mathbf{\Gamma}_t^T \mathbf{M} \mathbf{\Gamma}_t \mathbf{z}_t]$. We used the important result $\mathbf{\Sigma}_0 = \mathrm{E}[\hat{\mathbf{\Sigma}}_n]$ and also (3.4) to derive (B.1). Equation (B.2) follows from the definition (3.7) and equation (B.3) follows from the property that $\mathrm{tr}\left((\mathbf{A} \circ \mathbf{B})\mathbf{C}\right) = \mathrm{tr}\left(\mathbf{A}(\mathbf{B}^T \circ \mathbf{C})\right)$. Replacing the missing data model (3.1), we arrive at the equality (B.4). To simplify our derivation, we employ the joint Gaussian property of $\{\mathbf{x}_t\}_{t=1}^n$ and introduce a random vector with i.i.d

standard normal elements denoted by $\boldsymbol{\upsilon}_t := \boldsymbol{\Sigma}_0^{-\frac{1}{2}}\mathbf{x}_t \sim N(\mathbf{0}, \mathbf{I}_{d_1 d_2})$. Consequently, $\boldsymbol{\upsilon}_t^T \tilde{\mathbf{M}}_t \boldsymbol{\upsilon}_t$ is obtained as a stochastic equivalence to $\mathbf{x}_t^T \boldsymbol{\Gamma}_t^T \mathbf{M} \boldsymbol{\Gamma}_t \mathbf{x}_t$, where $\tilde{\mathbf{M}}_t := \boldsymbol{\Sigma}_0^{\frac{1}{2}} \boldsymbol{\Gamma}_t^T \mathbf{M} \boldsymbol{\Gamma}_t \boldsymbol{\Sigma}_0^{\frac{1}{2}}$. Using this decomposition, we have

$$
\begin{aligned}
\mathrm{E}|\phi_t|^2 &= \mathrm{E}|\boldsymbol{\upsilon}_t^T \tilde{\mathbf{M}}_t \boldsymbol{\upsilon}_t - \mathrm{E}[\boldsymbol{\upsilon}_t^T \tilde{\mathbf{M}}_t \boldsymbol{\upsilon}_t]|^2 \\
&= \mathrm{E}\left| \sum_{i,j=1}^{d_1 d_2} \boldsymbol{\upsilon}_t(i)\boldsymbol{\upsilon}_t(j)\tilde{\mathbf{M}}_t(i,j) - \sum_{i=1}^{d_1 d_2} \tilde{\mathbf{M}}_t(i,i) \right|^2 \\
&= \mathrm{E}\left| \sum_{i\neq j}^{d_1 d_2} \boldsymbol{\upsilon}_t(i)\boldsymbol{\upsilon}_t(j)\tilde{\mathbf{M}}_t(i,j) + \sum_{i=1}^{d_1 d_2} (\boldsymbol{\upsilon}_t(i)^2 - 1)\tilde{\mathbf{M}}_t(i,i) \right|^2 \\
&= \sum_{i\neq j}^{d_1 d_2} \sum_{i'\neq j'}^{d_1 d_2} \mathrm{E}[\boldsymbol{\upsilon}_t(i)\boldsymbol{\upsilon}_t(j)\boldsymbol{\upsilon}_t(i')\boldsymbol{\upsilon}_t(j')]\tilde{\mathbf{M}}_t(i,j)\tilde{\mathbf{M}}_t(i',j') \\
&+ \sum_{i=1}^{d_1 d_2} \sum_{i'=1}^{d_1 d_2} \mathrm{E}[(\boldsymbol{\upsilon}_t(i)^2 - 1)(\boldsymbol{\upsilon}_t(i')^2 - 1)]\tilde{\mathbf{M}}_t(i,i)\tilde{\mathbf{M}}_t(i',i') \\
&= \sum_{i\neq j}^{d_1 d_2} \tilde{\mathbf{M}}_t(i,j)^2 + 2\sum_{i=1}^{d_1 d_2} \tilde{\mathbf{M}}_t(i,i)^2 \\
&= \sum_{i,j=1}^{d_1 d_2} \tilde{\mathbf{M}}_t(i,j)^2 + \sum_{i=1}^{d_1 d_2} \tilde{\mathbf{M}}_t(i,i)^2 \leq 2\|\tilde{\mathbf{M}}_t\|_F^2 \\
&\leq 2\|\boldsymbol{\Sigma}_0\|_\infty^2 \|\boldsymbol{\Gamma}_t^T \mathbf{M} \boldsymbol{\Gamma}_t\|_F^2 \leq 2\|\boldsymbol{\Sigma}_0\|_\infty^2 \|\mathbf{M}\|_F^2.
\end{aligned} \tag{B.6}
$$

We next derive an upper bound on $\|\mathbf{M}\|_F^2$ as follows

$$
\begin{aligned}
\|\mathbf{M}\|_F^2 &= \mathrm{tr}\left((\mathbf{W}^{(-1)} \circ \mathbf{u}\mathbf{u}^T)(\mathbf{W}^{(-1)} \circ \mathbf{u}\mathbf{u}^T) \right) \\
&= \mathrm{tr}\left((\mathbf{W}^{(-1)} \circ \mathbf{W}^{(-1)})\mathbf{u}\mathbf{u}^T \circ \mathbf{u}\mathbf{u}^T \right) \\
&= \mathrm{tr}\left(\mathbf{W}^{(-2)} \mathbf{u}^{(2)} \mathbf{u}^{(2)^T} \right) \\
&\leq \max_{\mathbf{u}\in\mathcal{N}_{d_1 d_2}} \mathbf{u}^{(2)^T} \mathbf{W}^{(-2)} \mathbf{u}^{(2)}.
\end{aligned} \tag{B.7}
$$

Employing a moment bound for the random variable ϕ_t (page 65 in [129]) and also Stirling's formula, Theorem 1.1 in [128] yields

$$
\mathrm{E}|\phi_t|^p \leq p! Z^{p-2} \tau_t / 2,
$$

for all $p \geq 2$ and

$$Z = e\left(\mathrm{E}|\phi_t|^2\right)^{\frac{1}{2}} \leq e\sqrt{2C}\|\mathbf{\Sigma}_0\|_\infty$$
$$\tau_t = \frac{2e}{\sqrt{6\pi}}\mathrm{E}|\phi_t|^2 \leq \frac{4Ce}{\sqrt{6\pi}}\|\mathbf{\Sigma}_0\|_\infty^2,$$

where $C := \max_{\mathbf{u}\in\mathcal{N}_{d_1 d_2}} \mathbf{u}^{(2)^T}\mathbf{W}^{(-2)}\mathbf{u}^{(2)}$. The above inequalities follow from (B.6) and (B.7). To end this proof, we consider Bernstein's inequality leading to

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{t=1}^{n}\phi_t\right| \geq \delta\right) \leq 2e^{-\frac{1}{2}\frac{n^2\delta^2}{n\tau_1+nZ\delta}}$$
$$\leq 2e^{-n\frac{\delta^2}{C_1 C\|\mathbf{\Sigma}_0\|_\infty^2+C_2\sqrt{C}\delta\|\mathbf{\Sigma}_0\|_\infty}}$$

with $C_1 = \frac{8e}{\sqrt{6\pi}}$ and $C_2 = 2e\sqrt{2}$. Choosing $\delta = \|\mathbf{\Sigma}_0\|_0$, we can conclude that $\mathbf{\Sigma}_n^{\mathbf{\Gamma}}$ is positive definite with probability at least

$$1 - 2e^{-n\frac{\|\mathbf{\Sigma}_0\|_0^2}{C_1 C\|\mathbf{\Sigma}_0\|_\infty^2+C_2\sqrt{C}\|\mathbf{\Sigma}_0\|_0\|\mathbf{\Sigma}_0\|_\infty}}$$

This completes the proof.

Lemma B.1

Lemma B.1. *If matrix* $\mathbf{A}$ *is positive definite,* $\mathbf{B}$ *is positive semidefinite, and all entries on the diagonal of* $\mathbf{B}$ *are positive then* $\mathbf{A} \circ \mathbf{B}$ *is positive definite.*

Proof. This is the immediate result of the Schur product theorem [130, 131]. □

Lemma B.2

The concentration of measure phenomenon is an important subject in probability analysis. There is now a vast literature on this area [132]. Specifically, [21] explores a class of concentration of measure for coupled Gaussian Chaos based on i.i.d mutivarite normal observations. However, these observation vectors in our study, $\{\mathbf{z}_t\}_{t=1}^n$, are corrupted by missing data. Next lemma performs a systematic investigation of the concentration of measure phenomenon for $\{\mathbf{z}_t\}_{t=1}^n$.

Lemma B.2. *Consider observation vectors* $\{\mathbf{z}_t\}_{t=1}^n$ *as introduced in the System Model. Let* $\mathbf{u} = (u_1, u_2, \ldots, u_{d_1^2})^T \in \mathcal{N}_{d_1^2}$, $\mathbf{v} = (v_1, v_2, \ldots, u_{d_2^2})^T \in \mathcal{N}_{d_2^2}$, *and recall* $\mathbf{D}_n$ *(see Section 3.4). We then have for all* $\delta \geq 0$,

$$\mathbb{P}\left(\left|\mathbf{u}^T \mathbf{D}_n \mathbf{v}\right| \geq \delta\right) \leq 2e^{-n\frac{\delta^2}{C_1 C_P \|\mathbf{\Sigma}_0\|_\infty^2 + C_2\sqrt{C_P}\delta\|\mathbf{\Sigma}_0\|_\infty}}. \tag{B.8}$$

Proof. Let us rewrite $\mathbf{D}_n$ as follows

$$\begin{aligned}
\mathbf{D}_n &= \mathcal{P}(\hat{\mathbf{\Sigma}}_n) - \mathcal{P}(\mathbf{\Sigma}_0) \\
&= \mathcal{P}(\mathbf{W}^{(-1)} \circ \mathbf{\Sigma}_n^{\mathbf{\Gamma}}) - \mathcal{P}(\mathrm{E}[\mathbf{W}^{(-1)} \circ \mathbf{\Sigma}_n^{\mathbf{\Gamma}}]) \\
&= \frac{1}{n}\sum_{t=1}^n \mathcal{P}(\mathbf{W}^{(-1)}) \circ \left(\mathcal{P}(\mathbf{z}_t\mathbf{z}_t^T) - \mathrm{E}[\mathcal{P}(\mathbf{z}_t\mathbf{z}_t^T)]\right) \\
&= \frac{1}{n}\sum_{t=1}^n \mathcal{P}(\mathbf{W}^{(-1)}) \circ \\
&\begin{pmatrix}
\mathrm{vec}(\mathbf{z}_t[1]\mathbf{z}_t[1]^T)^T - \mathrm{E}\left[\mathrm{vec}(\mathbf{z}_t[1]\mathbf{z}_t[1]^T)^T\right] \\
\mathrm{vec}(\mathbf{z}_t[1]\mathbf{z}_t[2]^T)^T - \mathrm{E}\left[\mathrm{vec}(\mathbf{z}_t[1]\mathbf{z}_t[2]^T)^T\right] \\
\vdots \\
\mathrm{vec}(\mathbf{z}_t[d_1]\mathbf{z}_t[d_1]^T)^T - \mathrm{E}\left[\mathrm{vec}(\mathbf{z}_t[d_1]\mathbf{z}_t[d_1]^T)^T\right]
\end{pmatrix},
\end{aligned}$$

where $\mathbf{z}_t[i] := \mathbf{z}_t(1 + (i-1)d_2 : id_2)$ is the ith subvector of $\mathbf{z}_t$. Hence, introducing $\mathbf{F} := \mathbf{W}^{(-1)} \circ (\mathbf{U} \otimes \mathbf{V})$ with $\mathbf{U} := \mathcal{P}^{(-1)}(\mathbf{u})$ and $\mathbf{V} := \mathcal{P}^{(-1)}(\mathbf{v})$, some elementary algebra gives that

$$\begin{aligned}
\mathbf{u}^T\mathbf{D}_n\mathbf{v} &= \sum_{t=1}^n \mathrm{tr}\left((\mathbf{W}^{(-1)} \circ \mathbf{z}_t\mathbf{z}_t^T)\mathbf{U} \otimes \mathbf{V}\right) - \mathrm{E}\left[\mathrm{tr}\left((\mathbf{W}^{(-1)} \circ \mathbf{z}_t\mathbf{z}_t^T)\mathbf{U} \otimes \mathbf{V}\right)\right] \\
&= \sum_{t=1}^n \mathrm{tr}\left(\left(\mathbf{W}^{(-1)} \circ (\mathbf{U} \otimes \mathbf{V})\right)\mathbf{z}_t\mathbf{z}_t^T\right) - \mathrm{E}\left[\mathrm{tr}\left(\left(\mathbf{W}^{(-1)} \circ (\mathbf{U} \otimes \mathbf{V})\right)\mathbf{z}_t\mathbf{z}_t^T\right)\right] \\
&= \frac{1}{n}\sum_{t=1}^n \left(\mathbf{z}_t^T\mathbf{F}\mathbf{z}_t - \mathrm{E}[\mathbf{z}_t^T\mathbf{F}\mathbf{z}_t]\right).
\end{aligned}$$

Following the proof of Theorem 3.1, combined with

$$\begin{aligned}\|\mathbf{F}\|_F^2 &= \left\|\mathbf{W}^{(-1)} \circ (\mathbf{U} \otimes \mathbf{V})\right\|_F^2 \\ &= \operatorname{tr}\left[\left(\mathbf{W}^{(-1)} \circ (\mathbf{U} \otimes \mathbf{V})\right)\left(\mathbf{W}^{(-1)} \circ (\mathbf{U} \otimes \mathbf{V})\right)\right] \\ &= \operatorname{tr}\left[\mathbf{W}^{(-2)}(\mathbf{U}^{(2)} \otimes \mathbf{V}^{(2)})\right] \\ &= \mathbf{u}^{(2)}\mathcal{P}\left(\mathbf{W}^{(-2)}\right)\mathbf{v}^{(2)} \\ &\leq \max_{\mathbf{u}\in\mathcal{N}_{d_1^2},\mathbf{v}\in\mathcal{N}_{d_2^2}} \mathbf{u}^{(2)}\mathcal{P}\left(\mathbf{W}^{(-2)}\right)\mathbf{v}^{(2)},\end{aligned}$$

we arrive at (B.8). □

Proof of Theorem 3.3

Although we follow similar logic as Appendix E in [21], our new Theorem 3.2 is taken into account to prove the result here.

Assuming that $\|\mathbf{D}_n\|_\infty \leq \frac{\gamma}{2}$, with probability one, we have from Theorem 2 in [21] (see also Theorem 1 in [115])

$$\|\hat{\mathbf{P}}_n^\gamma - \mathbf{P}_0\|_F^2 \leq \inf_{\mathbf{P}:\operatorname{rank}(\mathbf{P})\leq r} \|\mathbf{P} - \mathbf{P}_0\|_F^2 + \frac{(1+\sqrt{2})^2}{4}\gamma^2\operatorname{rank}(\mathbf{P}) \tag{B.9}$$

We introduce the event

$$\omega_r := \left\{\|\hat{\mathbf{P}}_n^\gamma - \mathbf{P}_0\|_F^2 > \inf_{\mathbf{P}:\operatorname{rank}(\mathbf{P})\leq r} \|\mathbf{P} - \mathbf{P}_0\|_F^2 + \frac{(1+\sqrt{2})^2}{4}\gamma^2\operatorname{rank}(\mathbf{P})\right\}$$

Employing (B.9), we attain

$$\begin{aligned}\mathbb{P}(\omega_r) &= \mathbb{P}\left(\omega_r \cap \left\{\|\mathbf{D}_n\|_\infty > \frac{\gamma}{2}\right\}\right) + \mathbb{P}\left(\omega_r \cap \left\{\|\mathbf{D}_n\|_\infty \leq \frac{\gamma}{2}\right\}\right) \\ &= \mathbb{P}\left(\omega_r \cap \left\{\|\mathbf{D}_n\|_\infty > \frac{\gamma}{2}\right\}\right) \\ &= \mathbb{P}\left(\omega_r \middle| \|\mathbf{D}_n\|_\infty > \frac{\gamma}{2}\right)\mathbb{P}\left(\|\mathbf{D}_n\|_\infty > \frac{\gamma}{2}\right) \\ &\leq \mathbb{P}\left(\|\mathbf{D}_n\|_\infty > \frac{\gamma}{2}\right).\end{aligned}$$

Choosing $\gamma = \frac{2q\|\mathbf{\Sigma}_0\|_\infty}{1-2\epsilon}\max\left(\frac{d_1^2+d_2^2+\log N}{n}, \sqrt{\frac{d_1^2+d_2^2+\log N}{n}}\right)$, Theorem 3.2 implies that

$$\mathbb{P}\left(\|\mathbf{D}_n\|_\infty > \frac{\gamma}{2}\right) \leq 2N^{-\frac{q}{2C_0}}.$$

Tanking into account that $\|\hat{\mathbf{P}}_n - \mathbf{P}_0\|_F^2$ is equivalent to $\|\hat{\mathbf{\Sigma}}_n - \mathbf{\Sigma}_0\|_F^2$, the proof is completed.

APPENDIX C. SUPPORTING INFORMATION FOR CHAPTER 4

Inversion of Block Matrix

The following result is useful for computing the inverse of a sub-matrix of a given matrix based on the inverse of the whole matrix; see e.g., [133] and [134, p.572]

Lemma C.1. *Let* $\mathbf{A}$ *be a nonsingular and symmetric matrix. Matrix* $\mathbf{A}$ *can be described based on* $\mathbf{LDL}^T$ *factorization as*

$$\mathbf{A} := \begin{pmatrix} \mathbf{A}_{11} & \mathbf{B}_{21}^T \\ \mathbf{B}_{21} & \mathbf{A}_{22} \end{pmatrix} = \begin{pmatrix} \mathbf{I} & \\ \mathbf{L}_{21} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{A}_{11} & \\ & \mathbf{A}_{22} - \mathbf{B}_{21}\mathbf{A}_{11}^{-1}\mathbf{B}_{21}^T \end{pmatrix} \begin{pmatrix} \mathbf{I} & \mathbf{L}_{21}^T \\ & \mathbf{I} \end{pmatrix},$$

where $\mathbf{L}_{21} = \mathbf{B}_{21}\mathbf{A}_{11}^{-1}$ *and* $\mathbf{S} = \mathbf{A}_{22} - \mathbf{B}_{21}\mathbf{A}_{11}^{-1}\mathbf{B}_{21}^T$ *is the Schur complement. The inversion of* $\mathbf{A}$ *can be expressed by*

$$\mathbf{A}^{-1} = \begin{pmatrix} \mathbf{A}_{11}^{-1} + \mathbf{L}_{21}^T\mathbf{S}^{-1}\mathbf{L}_{21} & -\mathbf{L}_{21}^T\mathbf{S}^{-1} \\ -\mathbf{S}^{-1}\mathbf{L}_{21} & \mathbf{S}^{-1} \end{pmatrix}. \tag{C.1}$$

Equation (C.1) suggests that once $\mathbf{A}^{-1}$ is known, the task of computing $\mathbf{A}_{11}^{-1}$ can be reduced to the following steps.

1. Calculate $\mathbf{S}$ by inverting $\mathbf{S}^{-1}$.
2. Rewrite $\mathbf{L}_{21}^T\mathbf{S}^{-1}\mathbf{L}_{21} = \mathbf{L}_{21}^T\mathbf{S}^{-1}\mathbf{S}\mathbf{S}^{-1}\mathbf{L}_{21}$ and compute it using $-\mathbf{L}_{21}^T\mathbf{S}^{-1}$, $\mathbf{S}$, and $-\mathbf{S}^{-1}\mathbf{L}_{21}$.
3. Subtract $\mathbf{L}_{21}^T\mathbf{S}^{-1}\mathbf{L}_{21}$ from the corresponding sub-block matrix in $\mathbf{A}^{-1}$ to obtain $\mathbf{A}_{11}^{-1}$.

Remark C.1. *The complexity of* $\mathbf{LDL}^T$ *factorization for an arbitrary* $N \times N$ *matrix* $\mathbf{A}$ *is* $\mathcal{O}(N^3)$. *However, in some applications, for instance when* $\mathbf{A}$ *is sparse, the computation of* $\mathbf{LDL}^T$ *factorization is reduced significantly [133], [135]. Thus, the term* $\mathbf{L}_{21}^T\mathbf{S}^{-1}\mathbf{L}_{21}$ *can be alternatively computed as i) obtain* $\mathbf{L}_{21}$ *from* $\mathbf{LDL}^T$ *factorization of* $\mathbf{A}$*; and ii) compute* $\mathbf{L}_{21}^T\mathbf{S}^{-1}\mathbf{L}_{21}$ *using* $\mathbf{S}^{-1}$ *and* $\mathbf{L}_{21}$.

BIBLIOGRAPHY

[1] J. Bähler, "Cell-cycle control of gene expression in budding and fission yeast," *Annual Review of Genetics*, vol. 39, pp. 69–94, 2005.

[2] M. Hecker, S. Lambeck, S. Toepfer, E. Van Someren, and R. Guthke, "Gene regulatory network inference: data integration in dynamic modelsa review," *Biosystems*, vol. 96, no. 1, pp. 86–103, 2009.

[3] B. Ristevski, "A survey of models for inference of gene regulatory networks," *Nonlinear Anal Model Control*, vol. 18, no. 4, pp. 444–465, 2013.

[4] H. Kitano, "Systems biology: a brief overview," *Science*, vol. 295, no. 5560, pp. 1662–1664, 2002.

[5] E. C. Butcher, E. L. Berg, and E. J. Kunkel, "Systems biology in drug discovery," *Nature Biotechnology*, vol. 22, no. 10, pp. 1253–1259, 2004.

[6] D. Peer, A. Regev, G. Elidan, and N. Friedman, "Inferring subnetworks from perturbed expression profiles," *Bioinformatics*, vol. 17, no. suppl 1, pp. S215–S224, 2001.

[7] S. Kauffman, C. Peterson, B. Samuelsson, and C. Troein, "Random Boolean network models and the yeast transcriptional network," *Proceedings of the National Academy of Sciences*, vol. 100, no. 25, pp. 14796–14799, 2003.

[8] A. De La Fuente, N. Bing, I. Hoeschele, and P. Mendes, "Discovery of meaningful associations in genomic data using partial correlation coefficients," *Bioinformatics*, vol. 20, no. 18, pp. 3565–3574, 2004.

[9] C. A. Penfold and D. L. Wild, "How to infer gene networks from expression profiles, revisited," *Interface Focus*, vol. 1, no. 6, pp. 857–870, 2011.

[10] T. E. Ideker, V. Thorsson, and R. M. Karp, "Discovery of regulatory interactions through perturbation: inference and experimental design," *Pacific Symposium on Biocomputing*, vol. 5, pp. 302–313, 2000.

[11] D. E. Zak, G. E. Gonye, J. S. Schwaber, and F. J. Doyle, "Importance of input perturbations and stochastic gene expression in the reverse engineering of genetic regulatory networks: insights from an identifiability analysis of an in silico network," *Genome Research*, vol. 13, no. 11, pp. 2396–2405, 2003.

[12] M. Bansal, V. Belcastro, A. Ambesi-Impiombato, and D. Di Bernardo, "How to infer gene networks from expression profiles," *Molecular Systems Biology*, vol. 3, no. 1, 2007.

[13] J. Tegner, M. S. Yeung, J. Hasty, and J. J. Collins, "Reverse engineering gene networks: integrating genetic perturbations with dynamical modeling," *Proceedings of the National Academy of Sciences*, vol. 100, no. 10, pp. 5944–5949, 2003.

[14] I. Shmulevich, I. Gluhovsky, R. F. Hashimoto, E. R. Dougherty, and W. Zhang, "Steady-state analysis of genetic regulatory networks modelled by probabilistic boolean networks," *Comparative and Functional Genomics*, vol. 4, no. 6, pp. 601–608, 2003.

[15] A. Meister, Y. H. Li, B. Choi, W. H. Wong, et al., "Learning a nonlinear dynamical system model of gene regulation: a perturbed steady-state approach," *The Annals of Applied Statistics*, vol. 7, no. 3, pp. 1311–1333, 2013.

[16] L. Bintu, N. E. Buchler, H. G. Garcia, U. Gerland, T. Hwa, J. Kondev, and R. Phillips, "Transcriptional regulation by the numbers: models," *Current Opinion in Genetics & Development*, vol. 15, no. 2, pp. 116–124, 2005.

[17] L. Bintu, N. E. Buchler, H. G. Garcia, U. Gerland, T. Hwa, J. Kondev, T. Kuhlman, and R. Phillips, "Transcriptional regulation by the numbers: applications," *Current Opinion in Genetics & Development*, vol. 15, no. 2, pp. 125–135, 2005.

[18] J. Bai and S. Shi, "Estimating high dimensional covariance matrices and its applications," *Annals of Economics & Finance*, vol. 12, no. 2, 2011.

[19] J. Schäfer and K. Strimmer, "A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics," *Statistical Applications in Genetics and Molecular Biology*, vol. 4, no. 1, 2005.

[20] G. I. Allen and R. Tibshirani, "Transposable regularized covariance models with an application to missing data imputation," *The Annals of Applied Statistics*, vol. 4, no. 2, pp. 764–790, 2010.

[21] T. Tsiligkaridis and A. O. Hero, "Covariance estimation in high dimensions via kronecker product expansions," *IEEE Transactions on Signal Processing*, vol. 61, no. 21, pp. 5347–5360, 2013.

[22] J. Fan, Y. Fan, and J. Lv, "High dimensional covariance matrix estimation using a factor model," *Journal of Econometrics*, vol. 147, no. 1, pp. 186–197, 2008.

[23] A. J. Rothman, P. J. Bickel, E. Levina, J. Zhu, et al., "Sparse permutation invariant covariance estimation," *Electronic Journal of Statistics*, vol. 2, pp. 494–515, 2008.

[24] M. Yuan and Y. Lin, "Model selection and estimation in the gaussian graphical model," *Biometrika*, vol. 94, no. 1, pp. 19–35, 2007.

[25] N. Cressie, "Statistics for spatial data," *Terra Nova*, vol. 4, no. 5, pp. 613–617, 1992.

[26] J. Yin and H. Li, "Model selection and estimation in the matrix normal graphical model," *Journal of Multivariate Analysis*, vol. 107, pp. 119–140, 2012.

[27] M. P. Jones, "Indicator and stratification methods for missing explanatory variables in multiple linear regression," *Journal of the American Statistical Association*, vol. 91, no. 433, pp. 222–230, 1996.

[28] P. D. Allison, "Missing data techniques for structural equation modeling," *Journal of Abnormal Psychology*, vol. 112, no. 4, pp. 545–557, 2003.

[29] P. D. Allison, *Missing data*, Sage publications, 2001.

[30] O. L. V. Costa and S. Guerra, "Stationary filter for linear minimum mean square error estimator of discrete-time markovian jump systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 8, pp. 1351–1356, 2002.

[31] O. Costa, "Linear minimum mean square error estimation for discrete-time markovian jump linear systems," *IEEE Transactions on Automatic Control*, vol. 39, no. 8, pp. 1685–1689, 1994.

[32] T. J. Lim and Y. Ma, "The kalman filter as the optimal linear minimum mean-squared error multiuser CDMA detector," *IEEE Transactions on Information Theory*, vol. 46, no. 7, pp. 2561–2566, 2000.

[33] C. D. Frank, E. Visotsky, P. Choudhary, and A. Ghosh, "Linear minimum mean square error equalization with interference cancellation for mobile communication forward links utilizing orthogonal codes covered by long pseudorandom spreading codes," Oct. 18 2005, US Patent 6,956,893.

[34] J. S. Goldstein, I. S. Reed, and L. Scharf, "A multistage representation of the Wiener filter based on orthogonal projections," *IEEE Transactions on Information Theory*, vol. 44, no. 7, pp. 2943–2959, 1998.

[35] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman, "Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data," *Nature Genetics*, vol. 34, no. 2, pp. 166–176, 2003.

[36] W.-P. Lee and W.-S. Tzou, "Computational methods for discovering gene networks from expression data," *Briefings in Bioinformatics*, vol. 10, no. 4, pp. 408–423, 2009.

[37] R. De Smet and K. Marchal, "Advantages and limitations of current network inference methods," *Nature Reviews Microbiology*, vol. 8, no. 10, pp. 717–729, 2010.

[38] D. Marbach, R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, and G. Stolovitzky, "Revealing strengths and weaknesses of methods for gene network inference," *Proceedings of the National Academy of Sciences*, vol. 107, no. 14, pp. 6286–6291, 2010.

[39] B. Jia and X. Wang, "Gene regulatory network inference by point-based gaussian approximation filters incorporating the prior information," *EURASIP Journal on Bioinformatics and Systems Biology*, vol. 2013, no. 1, pp. 1–16, 2013.

[40] R. C. Jansen, "Studying complex biological systems using multifactorial perturbation," *Nature Reviews Genetics*, vol. 4, no. 2, pp. 145–151, 2003.

[41] T. S. Gardner, D. Di Bernardo, D. Lorenz, and J. J. Collins, "Inferring genetic networks and identifying compound mode of action via expression profiling," *Science*, vol. 301, no. 5629, pp. 102–105, 2003.

[42] M. Bansal, G. Della Gatta, and D. Di Bernardo, "Inference of gene regulatory networks and compound mode of action from time course gene expression profiles," *Bioinformatics*, vol. 22, no. 7, pp. 815–822, 2006.

[43] B. A. Logsdon and J. Mezey, "Gene expression network reconstruction by convex feature selection when incorporating genetic perturbations," *PLOS Computational Biology*, vol. 6, no. 12, 2010.

[44] G. Krouk, J. Lingeman, A. M. Colon, G. Coruzzi, D. Shasha, et al., "Gene regulatory networks in plants: learning causality from time and perturbation," *Genome Biology*, vol. 14, no. 6, 2013.

[45] X. Cai, J. A. Bazerque, and G. B. Giannakis, "Inference of gene regulatory networks with sparse structural equation models exploiting genetic perturbations," *PLOS Computational Biology*, vol. 9, no. 5, 2013.

[46] H. Bolouri, "Modeling genomic regulatory networks with big data," *Trends in Genetics*, vol. 30, no. 5, pp. 182–191, 2014.

[47] B. C. Goodwin, "Oscillatory behavior in enzymatic control processes," *Advances in Enzyme Regulation*, vol. 3, pp. 425–437, 1965.

[48] T. Chen, H. L. He, G. M. Church, et al., "Modeling gene expression with differential equations.," *Pacific Symposium on Biocomputing*, vol. 4, no. 29, 1999.

[49] A. Ay and D. N. Arnosti, "Mathematical modeling of gene expression: a guide for the perplexed biologist," *Critical Reviews in Biochemistry and Molecular Biology*, vol. 46, no. 2, pp. 137–151, 2011.

[50] R. Khanin and V. Vinciotti, "Computational modeling of post-transcriptional gene regulation by microRNAs," *Journal of Computational Biology*, vol. 15, no. 3, pp. 305–316, 2008.

[51] J. Hausser, A. P. Syed, N. Selevsek, E. Van Nimwegen, L. Jaskiewicz, R. Aebersold, and M. Zavolan, "Timescales and bottlenecks in miRNA-dependent gene regulation," *Molecular Systems Biology*, vol. 9, no. 1, 2013.

[52] E. Van Rooij, "The art of microRNA research," *Circulation Research*, vol. 108, no. 2, pp. 219–234, 2011.

[53] G. Hutvágner and P. D. Zamore, "A microRNA in a multiple-turnover RNAi enzyme complex," *Science*, vol. 297, no. 5589, pp. 2056–2060, 2002.

[54] J. G. Doench and P. A. Sharp, "Specificity of microRNA target selection in translational repression," *Genes & Development*, vol. 18, no. 5, pp. 504–511, 2004.

[55] S. Wichert, K. Fokianos, and K. Strimmer, "Identifying periodically expressed transcripts in microarray time series data," *Bioinformatics*, vol. 20, no. 1, pp. 5–20, 2004.

[56] Y. Luan and H. Li, "Model-based methods for identifying periodically expressed genes based on time course microarray gene expression data," *Bioinformatics*, vol. 20, no. 3, pp. 332–339, 2004.

[57] M. Rooman, J. Albert, Y. Dehouck, and A. Haye, "Detection of perturbation phases and developmental stages in organisms from DNA microarray time series data," *PlOS ONE*, vol. 6, no. 12, 2011.

[58] E. F. Glynn, J. Chen, and A. R. Mushegian, "Detecting periodic patterns in unevenly spaced gene expression time series using lomb–scargle periodograms," *Bioinformatics*, vol. 22, no. 3, pp. 310–316, 2006.

[59] P. Chaudhuri and J. S. Marron, "Sizer for exploration of structures in curves," *Journal of the American Statistical Association*, vol. 94, no. 447, pp. 807–823, 1999.

[60] S. Zhang, Q. Li, J. Liu, and X. J. Zhou, "A novel computational framework for simultaneous integration of multiple types of genomic data to identify microRNA-gene regulatory modules," *Bioinformatics*, vol. 27, no. 13, pp. i401–i409, 2011.

[61] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, 2005.

[62] T. Shimamura, S. Imoto, R. Yamaguchi, A. Fujita, M. Nagasaki, and S. Miyano, "Recursive regularization for inferring gene networks from time-course gene expression profiles," *BMC Systems Biology*, vol. 3, no. 1, 2009.

[63] D. P. Bertsekas, *Nonlinear programming*, Athena Scientific Belmont, 1999.

[64] R. A. Horn and C. R. Johnson, *Matrix analysis*, Cambridge University Press, 2012.

[65] M. Mardani, G. Mateos, and G. Giannakis, "Decentralized sparsity-regularized rank minimization: Algorithms and applications," *IEEE Transactions on Signal Processing*, vol. 61, no. 21, pp. 5374–5388, Nov 2013.

[66] J. Gorski, F. Pfeuffer, and K. Klamroth, "Biconvex sets and optimization with biconvex functions: a survey and extensions," *Mathematical Methods of Operations Research*, vol. 66, no. 3, pp. 373–407, 2007.

[67] F. Ollivier, *Le problème de l'identifiabilité structurelle globale: approche théorique, méthodes effectives et bornes de complexité*, Ph.D. thesis, Palaiseau, Ecole Polytechnique, 1990.

[68] F. Ollivier, "Identifiabilité des systèmes," Tech. Rep., GAGE, Ecole Polytechnique, 1997.

[69] G. Margaria, E. Riccomagno, M. J. Chappell, and H. P. Wynn, "Differential algebra methods for the study of the structural identifiability of rational function state-space models in the biosciences," *Mathematical Biosciences*, vol. 174, no. 1, pp. 1–26, 2001.

[70] N. Meshkat, M. Eisenberg, and J. J. DiStefano III, "An algorithm for finding globally identifiable parameter combinations of nonlinear ODE models using Gröbner bases," *Mathematical Biosciences*, vol. 222, no. 2, pp. 61–72, 2009.

[71] N. Meshkat, C. Anderson, and J. J. DiStefano, "Finding identifiable parameter combinations in nonlinear ODE models and the rational reparameterization of their input–output equations," *Mathematical Biosciences*, vol. 233, no. 1, pp. 19–31, 2011.

[72] G. Bellu, M. P. Saccomani, S. Audoly, and L. DAngiò, "Daisy: a new software tool to test global identifiability of biological and physiological systems," *Computer Methods and Programs in Biomedicine*, vol. 88, no. 1, pp. 52–61, 2007.

[73] F. Boulier, D. Lazard, F. Ollivier, and M. Petitot, "Computing representations for radicals of finitely generated differential ideals," *Applicable Algebra in Engineering, Communication and Computing*, vol. 20, no. 1, pp. 73–121, 2009.

[74] S. Vajda, "Identifiability of polynomial systems: structural and numerical aspects," *Identifiability of Parametric Models*, vol. 4, pp. 42–49, 1987.

[75] G. Karlebach and R. Shamir, "Modelling and analysis of gene regulatory networks," *Nature Reviews Molecular Cell Biology*, vol. 9, no. 10, pp. 770–780, 2008.

[76] A. Greenfield, A. Madar, H. Ostrer, and R. Bonneau, "Dream4: Combining genetic and dynamic information to identify biological networks and dynamical models," *PlOS ONE*, vol. 5, no. 10, 2010.

[77] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher, "Comprehensive identification of cell cycle–regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization," *Molecular Biology of the Cell*, vol. 9, no. 12, pp. 3273–3297, 1998.

[78] M. Breker and M. Schuldiner, "The emergence of proteome-wide technologies: systematic analysis of proteins comes of age," *Nature Reviews Molecular Cell Biology*, vol. 15, no. 7, pp. 453–464, 2014.

[79] M. Rabani, J. Z. Levin, L. Fan, X. Adiconis, R. Raychowdhury, M. Garber, A. Gnirke, C. Nusbaum, N. Hacohen, N. Friedman, et al., "Metabolic labeling of RNA uncovers principles of RNA production and degradation dynamics in mammalian cells," *Nature Biotechnology*, vol. 29, no. 5, pp. 436–442, 2011.

[80] T. Maier, A. Schmidt, M. Güell, S. Kühner, A.-C. Gavin, R. Aebersold, and L. Serrano, "Quantification of mRNA and protein and integration with protein turnover in a bacterium," *Molecular Systems Biology*, vol. 7, no. 1, 2011.

[81] M. K. Doherty and R. J. Beynon, "Protein turnover on the scale of the proteome," *Expert Rev Proteomics*, vol. 3, no. 1, pp. 97–110, 2006.

[82] G. Rasool, N. Bouaynaya, H. M. Fathallah-Shaykh, and D. Schonfeld, "Inference of genetic regulatory networks using regularized likelihood with covariance estimation," in *IEEE Statistical Signal Processing Workshop (SSP)*, pp. 560–563, 2012.

[83] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.

[84] J. Xie and P. M. Bentler, "Covariance structure models for gene expression microarray data," *Structural Equation Modeling*, vol. 10, no. 4, pp. 566–582, 2003.

[85] S. Oba, M.-a. Sato, I. Takemasa, M. Monden, K.-i. Matsubara, and S. Ishii, "A bayesian missing value estimation method for gene expression profile data," *Bioinformatics*, vol. 19, no. 16, pp. 2088–2096, 2003.

[86] R. J. Little, "Robust estimation of the mean and covariance matrix from data with missing values," *Applied Statistics*, vol. 37, no. 1, pp. 23–38, 1988.

[87] T. Schneider, "Analysis of incomplete climate data: estimation of mean values and covariance matrices and imputation of missing values.," *Journal of Climate*, vol. 14, no. 5, 2001.

[88] P.-L. Loh, M. J. Wainwright, et al., "Structure estimation for discrete graphical models: generalized covariance matrices and their inverses," *The Annals of Statistics*, vol. 41, no. 6, pp. 3022–3049, 2013.

[89] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*, John Wiley & Sons, 2002.

[90] P. D. Allison, "Missing data: quantitative applications in the social sciences," *British Journal of Mathematical and Statistical Psychology*, vol. 55, no. 1, pp. 193–196, 2002.

[91] A. Afifi and R. Elashoff, "Missing observations in multivariate statistics i. review of the literature," *Journal of the American Statistical Association*, vol. 61, no. 315, pp. 595–604, 1966.

[92] L. M. Collins, J. L. Schafer, and C.-M. Kam, "A comparison of inclusive and restrictive strategies in modern missing data procedures.," *Psychological Methods*, vol. 6, no. 4, pp. 330, 2001.

[93] M. Zamanighomi, Z. Wang, K. Slavakis, and G. B. Giannakis, "Linear minimum mean-square error estimation based on high-dimensional data with missing values," in *IEEE 48th Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–5, 2014.

[94] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[95] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Review*, vol. 52, no. 3, pp. 471–501, 2010.

[96] E. Candes and T. Tao, "The dantzig selector: statistical estimation when p is much larger than n," *The Annals of Statistics*, pp. 2313–2351, 2007.

[97] P. J. Bickel, Y. Ritov, and A. B. Tsybakov, "Simultaneous analysis of lasso and dantzig selector," *The Annals of Statistics*, pp. 1705–1732, 2009.

[98] I. M. Johnstone and A. Y. Lu, "On consistency and sparsity for principal components analysis in high dimensions," *Journal of the American Statistical Association*, vol. 104, no. 486, 2009.

[99] P. Ravikumar, M. J. Wainwright, G. Raskutti, B. Yu, et al., "High-dimensional covariance estimation by minimizing 1-penalized log-determinant divergence," *Electronic Journal of Statistics*, vol. 5, pp. 935–980, 2011.

[100] E. J. Candes and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 925–936, 2010.

[101] J. Mairal, J. Ponce, G. Sapiro, A. Zisserman, and F. R. Bach, "Supervised dictionary learning," in *Advances in neural information processing systems*, pp. 1033–1040, 2009.

[102] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[103] T. A. Barton and D. R. Fuhrmann, "Covariance structures for multidimensional data," *Multidimensional Systems and Signal Processing*, vol. 4, no. 2, pp. 111–123, 1993.

[104] K. Werner, M. Jansson, and P. Stoica, "On estimation of covariance matrices with kronecker product structure," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 478–491, 2008.

[105] E. Bonilla, K. M. Chai, and C. Williams, "Multi-task gaussian process prediction," in *Advances in Neural Information Processing Systems*, pp. 153–160, 2008.

[106] M. Lynch and B. Walsh, *Genetics and analysis of quantitative traits*, Sinauer Sunderland, 1998.

[107] S. L. Teng and H. Huang, "A statistical framework to infer functional gene relationships from biologically interrelated microarray experiments," *Journal of the American Statistical Association*, vol. 104, no. 486, pp. 465–473, 2009.

[108] K. Yu, W. Chu, S. Yu, V. Tresp, and Z. Xu, "Stochastic relational models for discriminative link prediction," in *Advances in Neural Information Processing Systems*, pp. 1553–1560, 2006.

[109] H. Wackernagel, *Multivariate geostatistics*, Springer Science & Business Media, 2003.

[110] A. T. Galecki, "General class of covariance structures for two or more repeated factors in longitudinal data analysis," *Communications in Statistics-Theory and Methods*, vol. 23, no. 11, pp. 3105–3119, 1994.

[111] R. J. Boik, "Scheffés mixed model for multivariate repeated measures: a relative efficiency evaluation," *Communications in Statistics-Theory and Methods*, vol. 20, no. 4, pp. 1233–1255, 1991.

[112] J. W. Graham, S. M. Hofer, and D. P. MacKinnon, "Maximizing the usefulness of data obtained with planned missing value patterns: An application of maximum likelihood procedures," *Multivariate Behavioral Research*, vol. 31, no. 2, pp. 197–218, 1996.

[113] M. Glasser, "Linear regression analysis with missing observations among the independent variables," *Journal of the American Statistical Association*, vol. 59, no. 307, pp. 834–844, 1964.

[114] S. Lee, F. Zou, and F. A. Wright, "Convergence and prediction of principal component scores in high-dimensional settings," *Annals of Statistics*, vol. 38, no. 6, pp. 3605, 2010.

[115] K. Lounici, "High-dimensional covariance matrix estimation with missing observations," *Bernoulli*, vol. 20, no. 3, pp. 1029–1058, 2014.

[116] C. V. Loan and N. Pitsianis, *Approximation with Kronecker products*, Springer, 1993.

[117] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[118] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pacific Journal of Optimization*, vol. 6, no. 15, pp. 615–640, 2010.

[119] M. Jaggi, M. Sulovsk, et al., "A simple algorithm for nuclear norm regularized problems," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 471–478, 2010.

[120] T. K. Pong, P. Tseng, S. Ji, and J. Ye, "Trace norm regularization: reformulations, algorithms, and multi-task learning," *SIAM Journal on Optimization*, vol. 20, no. 6, pp. 3465–3489, 2010.

[121] P. J. Rousseeuw and K. V. Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999.

[122] A. Gelman, *Data analysis using regression and multilevel/hierarchical models*, Cambridge University Press, 2007.

[123] Z. Wang, X. Liu, Y. Liu, J. Liang, and V. Vinciotti, "An extended kalman filtering approach to modeling nonlinear dynamic gene regulatory networks via short gene expression time series," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 6, no. 3, pp. 410–419, 2009.

[124] J. M. Peña, "B-splines and optimal stability," *Mathematics of Computation*, vol. 66, no. 220, pp. 1555–1560, 1997.

[125] X. Hong, S. Chen, Y. Gong, and C. Harris, "Nonlinear equalization of hammerstein OFDM systems," *IEEE Transactions on Signal Processing*, vol. 62, no. 21, pp. 5629–5639, 2014.

[126] C. De Boor, "A practical guide to splines," *Mathematics of Computation*, 1978.

[127] 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, 2011.

[128] H. Rauhut, K. Schnass, and P. Vandergheynst, "Compressed sensing and redundant dictionaries," *IEEE Transactions on Information Theory*, vol. 54, no. 5, pp. 2210–2219, 2008.

[129] M. Ledoux and M. Talagrand, *Probability in Banach spaces: isoperimetry and processes*, Springer Science & Business Media, 2013.

[130] J. Schur, "Bemerkungen zur theorie der beschränkten bilinearformen mit unendlich vielen veränderlichen.," *Journal für die reine und Angewandte Mathematik*, vol. 140, pp. 1–28, 1911.

[131] R. A. Horn and R. Mathias, "Block-matrix generalizations of Schur's basic theorems on hadamard products," *Linear Algebra and its Applications*, vol. 172, pp. 337–346, 1992.

[132] M. Ledoux, *The concentration of measure phenomenon*, American Mathematical Society, 2005.

[133] L. Lin, C. Yang, J. Lu, L. Ying, and W. E, "A fast parallel algorithm for selected inversion of structured sparse matrices with application to 2d electronic structure calculations," *SIAM Journal on Scientific Computing*, vol. 33, no. 3, pp. 1329–1351, 2011.

[134] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*, Prentice Hall, 1993.

[135] A. Gupta, G. Karypis, and V. Kumar, "Highly scalable parallel algorithms for sparse matrix factorization," *IEEE Transactions on Parallel and Distributed Systems*, vol. 8, no. 5, pp. 502–520, 1997.