Tractable Problems in Estimation and Control Subject to Sparsity and Structural Constraints

A Dissertation Presented

by

Wang, Yin

to

The Department of Electrical and Computer Engineering

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in

Electrical Engineering

Northeastern University
Boston, Massachusetts

January 2016
To my parents.
# Contents

List of Figures vii

List of Tables ix

Acknowledgments x

Abstract of the Dissertation xi

## 1 Introduction

1.1 Estimator and Controller Design under Information Structure 1

1.2 Regression Modeling with the Sensor Fault 3

1.3 GARCH Model Identification of the Covariance Feature 4

1.4 Organization of the Dissertation 5

1.5 Notations 5

## 2 Background

2.1 Moments: An SDP Relaxation Method with Global Optimality 8

2.1.1 The General Problem of Moments 9

2.1.2 Extension on 0-1 Optimization 10

2.2 Q Parametrization 11

2.3 Atomic Norm Representation and Frank-Wolfe Algorithm 14

2.3.1 Atomic Norm and Sparsity 14

2.3.2 Frank-Wolfe Algorithm 16

## 3 Worst-case Optimal Estimator based on Sensor Selection

3.1 Previous Works and Problem Statement 18

3.2 Proposed Solution 20

3.2.1 $\ell_{2}$ Bounded Noise 20

3.2.2 $\ell_{\infty}$ Bounded Noise 23

3.3 Results and Experiments 24

3.3.1 Estimation with Sensor Switching Strategy in Linear System 24

3.3.2 Estimation with Sensor Selection Strategy in Sensing System 27
4 $\ell_\infty$ Worst-case Optimal Estimators for Switched ARX Systems

4.1 Previous Works and Problem Statement ........................................ 29
4.2 Proposed Solution ........................................................................ 31
   4.2.1 A Semi-Algebraic Optimization Reformulation and Convex Relaxation 31
   4.2.2 Computational Complexity Considerations .............................. 34
   4.2.3 Consistency Considerations ................................................. 35
   4.2.4 Handling Parametric Uncertainty ....................................... 36
4.3 Handling Specific Switching Rule ............................................... 36
4.4 Results and Experiments ............................................................ 37
   4.4.1 Example without Model Uncertainty ................................. 37
   4.4.2 Example with Parametric Uncertainty ............................... 38
   4.4.3 Example with Switching Rule .......................................... 40

5 Sparse Structured Output Feedback Controller Design .................. 42
5.1 Previous Works .......................................................................... 42
5.2 Sparse Structured Static Output Feedback Controller Design ...... 43
5.3 Proposed Solution ....................................................................... 44
   5.3.1 Computational Complexity Considerations and Low Order Relaxations 48
   5.3.2 Extension to Unknown Sparse Pattern ................................ 49
   5.3.3 Extension to Continuous-time Case .................................... 50
5.4 Results and Experiments ............................................................ 51
   5.4.1 Design Sparse Gain with Non-Quadratic Invariant Pattern .... 51
   5.4.2 Design Sparse Output-Feedback Gain with Unknown Sparse Pattern 54
5.5 Sparse Structured Dynamic Output Feedback Controller Design 55
5.6 Statement of the Problem ........................................................... 56
5.7 Sparse Controller Synthesis via Convex Optimization ................. 56
   5.7.1 Finding Sparse Approximations of a Given Controller ........ 57
   5.7.2 Sparse Controller Approximation as a Filtering Problem .... 58
   5.7.3 Stability Analysis ............................................................ 59
   5.7.4 Proposed Algorithm ......................................................... 61
5.8 Illustrative Example ................................................................. 61

6 Identifying Sparse Dynamical Graphical Models via Super-Atomic Norm Minimization Approach 64
6.1 Motivation and Previous Works ................................................ 64
6.2 Problem Statement .................................................................... 66
6.3 Super Atoms and Block Sparsity ............................................... 68
6.4 Sparse Network Identification via Super-Atomic Norm Minimization 69
6.5 Extensions .................................................................................. 71
   6.5.1 External inputs ................................................................. 71
   6.5.2 Missing data ................................................................. 73
6.6 Reweighted Heuristic Super-Atomic Norm Minimization .......... 73
6.7 Examples .................................................................................... 73
   6.7.1 Synthetic Example without External Input ....................... 74
   6.7.2 Synthetic Example with External Input ............................ 76
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Linear fractional interconnection of $P$ and $K$</td>
<td>12</td>
</tr>
<tr>
<td>3.1</td>
<td>Comparison of Estimation Performance based on Strategies listed in Table 3.1.</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>Top (no constraints): Red squares (proposed method): Mean=-5.00e-3, Std=1.02e-1; Blue circles (random selection): Mean=6.18e-2, Std=3.86e-1; Green triangles: Mean=-8.38e-3, Std=9.43e-2. Center ($e_1$ active at most twice): Red squares (proposed method): Mean=1.34e-2, Std=2.10e-1; Blue circles (random selection): Mean=2.60e-2, Std=3.68e-1. Bottom ($e_1$ active at most once): Red squares (proposed method): Mean=4.43e-2, Std=2.28e-1; Blue circles (random selection): Mean=1.12e-1, Std=4.93e-1.</td>
<td></td>
</tr>
<tr>
<td>3.2</td>
<td>Estimation Error for Different Sensor Selections.</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>Red squares (proposed method): Mean=-6.40e-3, Std=7.67e-2; Blue circles (max-det selection): Mean=1.07e-2, Std=1.12e-1; Green triangles (All sensors): Mean=-5.56e-3, Std=5.63e-2.</td>
<td></td>
</tr>
<tr>
<td>4.1</td>
<td>Estimation with Determined Models.</td>
<td>38</td>
</tr>
<tr>
<td></td>
<td>Blue line: $[\hat{y}_t - 0.5, \hat{y}_t + 0.5]$; Magenta box: suboptimal bounds by Algorithm 3; Red star: $y_t$; Black circle: estimation of proposed estimator.</td>
<td></td>
</tr>
<tr>
<td>4.2</td>
<td>Estimation Error with Determined Models.</td>
<td>38</td>
</tr>
<tr>
<td>4.3</td>
<td>Estimation with 5% Parametric Uncertainty.</td>
<td>39</td>
</tr>
<tr>
<td>4.4</td>
<td>Estimation with 10% Parametric Uncertainty.</td>
<td>40</td>
</tr>
<tr>
<td>4.5</td>
<td>Absolute Value of the Actual Estimation Error for Different Uncertainty Levels.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Blue star: $A_k = C_k = 0$, green cross: $A_k = C_k = 5%$, red circle: $A_k = C_k = 10%$.</td>
<td>40</td>
</tr>
<tr>
<td>4.6</td>
<td>Performance comparison between estimation with switching rule and without switching rule.</td>
<td>41</td>
</tr>
<tr>
<td>5.1</td>
<td>Impulse Response of Static Controller with Specific Sparse Pattern in Discrete-time Case</td>
<td>52</td>
</tr>
<tr>
<td>5.2</td>
<td>Impulse Response of Static Controller with Specific Sparse Pattern in Continuous-time Case</td>
<td>54</td>
</tr>
<tr>
<td>5.3</td>
<td>Impulse Response of Static Controller with maximal Sparse Pattern</td>
<td>55</td>
</tr>
<tr>
<td>5.4</td>
<td>The setup for sparse control synthesis</td>
<td>56</td>
</tr>
</tbody>
</table>
5.5 Example where seeking sparse approximations by minimizing the distance from the controller to the desired sparsity set does not necessarily yield the best performing sparse approximation. ......................................................... 57
5.6 Sparse control synthesis as a filtering problem. .................................................. 58
5.7 Setup for stability analysis when using the sparsity constrained controller. ............ 59
5.8 Values of $\|T_{uw}(C) - C_sT_{yw}(C)\|_\infty$ for different sparse controller orders. ........ 63
5.9 Closed loop $\mathcal{H}_\infty$ norm achieved by the different controllers. ..................... 63
6.1 Recovery of External Input Signal ................................................................. 76
6.2 Sample Frames of UT Sequence 6 showing the causally interacting groups identified using different methods. Top: Proposed Method. Center: reweighted GLEI. Bottom: COLS. ................................................................. 78
6.3 Sample Frames of UT Sequence 16 showing the causally interacting groups identified using different methods. Top: Proposed Method. Center: reweighted GLEI. Bottom: COLS. The red circle denotes the agent position recovered by solving (6.16) ...... 79
6.4 Causally interacting groups in Double Tennis. Top: Proposed Method. Center: reweighted GLEI. Bottom: COLS. ................................................................. 80
7.1 Sample Regression Problems in Computer Vision. Left: Line fitting; Center: Surface fitting from 3D cloud data points; Right: Linear discriminant analysis (LDA) for face recognition. ................................................................. 82
7.2 Synthetic Data Experiments: Fitting a 5-dimensional hyperplane. The plots show the Geometric Mean of Precision and Recall for all the evaluated algorithms. .............. 93
7.3 Synthetic Data Experiments: Fitting a 5-dimensional hyperplane. The plots show the Model Error for all the evaluated algorithms. .................................................. 94
7.4 Face recovery results: In order from left to right, top to bottom: original image, occluded image, best possible recovery with given basis, proposed, BPRR, eRPCA, LMedS, Mestimator, MLESAC, MSAC, RANSAC, and RR. ................................. 96
8.1 Two examples where covariance features are used to describe a target. On the left, the appearance of the target car has roughly constant covariance. On the right, the covariance of the appearance of the spinning ball changes over time. ................. 98
8.2 Mean estimation error from 20 trials for the synthetic data experiment. .................. 113
8.3 Tracking under occlusion. Top: sample training data. Bottom: tracking results using different filters. ................................................................. 114
8.4 Sample frames from Youtube videos. Left: original sequences. Right: sequences corrupted by noise. ................................................................. 117
List of Tables

3.1 Sensor Choices for Different Algorithms ............................................. 25
6.1 Mean Computing Time as a Function of System Order ($N = 500, n_a = 10$) .... 75
6.2 Mean Computing Time as a Function of Number of Links ($N = 500, r = 6$) .... 75
6.3 Mean Computing Time as a Function of the Total Number of Nodes ($r = 6, n_a = 10$) 75
7.1 Running times for the experiments with synthetic data. .......................... 94
7.2 Fitting to original image error. .......................................................... 95
7.3 Model estimation error. ..................................................................... 96
7.4 Fitting to original image error (allowing extra time to the *SAC algorithms). .... 97
7.5 Model estimation error (allowing extra time to the *SAC algorithms). .......... 97
8.1 Mean Estimation Error and Running Time for the Experiments Using Youtube Video Clips (number in parathesis denotes the system order of JBRF model) ............... 116
Acknowledgments

First of all, I would like to give my most sincere appreciation to my advisor, Professor Mario Sznaier. Thanks for his patient and inspiring instructions while writing this dissertation. He’s not only a good research advisor, but also a good advisor in my career life. I’m very grateful for his advise and help in my future career plans.

I also want to express my appreciation to Professor Octavia Camps for opening my sight to the computer vision field. Auditing her class and reading the papers she recommended gave me an understanding of this interesting area. She also gave many valuable comments on the computer vision contents in this dissertation.

It is my great honor to have the committee members, Professor Dana Brooks and Professor Rifat Sipahi. I sincerely thank them for their advice which help me in putting this dissertation on solid theoretical ground.

To my dear friends and colleagues in the Robust Systems Lab, I appreciate your help and all the YASI’s (yet another stupid idea) we have discussed together.

Last but not the least, I want to express my greatest appreciation to my parents for their eternal love, persistent support and webcam companionship on every weekend. Their solicitude gave me the courage to face the setbacks in my study and life.
Abstract of the Dissertation

Tractable Problems in Estimation and Control Subject to Sparsity and Structural Constraints

by

Wang, Yin

Doctor of Philosophy in Electrical and Computer Engineering
Northeastern University, January 2016
Dr. Mario Sznaier, Adviser

Recent advances in sensing technology have led to an explosion on the amount of data that can be harvested from systems. However, in order to benefit from the availability of much richer information, engineers must face the “curse of dimensionality”. Classical system design techniques are not directly applicable to “data deluged” scenario due to their poor scaling properties, and their inability to handle structural constraints on the information flow. Motivated by these difficulties, in the past few years, many research efforts have been devoted toward developing computationally tractable approaches to handle “Big Data”. These ideas include exploiting the so called concentration of measure (inherent underlying sparsity) and self-similarity (high degree of spatio-temporal correlation in the data).

In this dissertation, using recent results from semi-algebraic optimization, Q parameterization, compressive sensing and manifold geometry, we show that many seemingly hard problems involving Big Data can be relaxed to tractable convex optimizations, in many cases with optimality certificates. Based on this analysis, computationally attractive convex tools are proposed for tasks including sensor selection, sparse controller design, robust regression, covariance feature propagation, etc. The potential of these tools is illustrated through numerical and practical experiments by comparing against existing methods. Finally, some open problems and directions for future research are stated.
Chapter 1

Introduction

Since the beginning of this century, human life has experienced an explosive development in the microelectronic field. The sensor integration, coupled with unceasing electronic miniaturization, leads to the extremely affordable sensing devices with considerable processing power. The cheap and numerous information-sensing devices lead to large scaled network systems and huge amount data collection, which gives rise to Big Data. Big Data is both a blessing and a curse for engineering design. On the bright side, accuracy in big data may lead to more confident decision making. But on the dark side, the data sets are so large or complex that the traditional data processing tools are inadequate. Therefore, data must be processed with advanced tools (analytics and algorithms) to reveal meaningful information.

In this dissertation, based on the recent results from the convex optimization theory, we present several tractable tools for three categories of problems arising from the era of big data: (1) the information structure constrained estimator and controller design; (2) robust regression of the data collection under sensor fault; (3) the propagating estimation of covariance feature, a fixed-size feature that captures stochastic information from large amount of data.

1.1 Estimator and Controller Design under Information Structure

Given the deluge of data collected by the numerous sensing devices, processing every single information is nearly impossible. This is not only because of the computational cost required to process the data, but also due to the communication difficulties and the limit of human operators in practice. Therefore, the goal is to achieve acceptable performance in some given criteria using only part of the available information channels, which leads into the sparse structure constraints in system
design. This type of constraint is also known as the information structure constraint, which has drawn substantial research interest in the past decade. In this research, the information structure constraint is considered in the estimator design and controller design problems, which are the two cornerstones of feedback control theory. It’s well known that designing an estimator or controller with the sparse structure constraints is not an easy problem. In fact, these problems are generically NP-hard\[1, 2]. However, as we show in this category, tractable approaches can be achieved in several practical problems. The discussion can be divided into three cases: (i) the worst-case optimal estimation, in which we focus on the estimator design based on sensor selection strategy and the estimator design of switched ARX system; (ii) sparse structured output feedback control, where we propose convex approaches for both static control and dynamic control; (iii) sparse dynamical graphical model identification, in which we involve the concept of Granger causality.

In the first case, we consider the estimator design under sparse information structure. Depending on the type of information channel, we divide this topic into two situations: the estimation based on sensor selection and the estimation of switched autoregressive exogenous (ARX) system. For the sensor selection based estimator, we consider the sparse structure as choosing a subset of sensors in all candidates (sparse structure on measurement channels), with the goal of achieving optimality on a given criterion. This is a widely considered problem in many areas, such as robotics, chemical process control, structural monitoring and target localization and tracking. This problem is essentially difficult because the combinatorial nature of the decision making. Several relaxations have been considered in \[3, 4, 5, 6, 7, 8, 9]\ under the Gaussian noise assumption. Although this assumption holds in many situations, it does not cover the practically important case of bounded noise. In this case, the noise is known to belong to a convex set characterized by a bounded norm description. Here, the objective is to estimate a given quantity based on the noisy measurements with no more than \(n\) sensors out of a total of \(N\) available ones. The advantages of the proposed approach are twofold: we consider the case that the only information available about the noise is the deterministic set-membership description; some additional selection constraints can be easily involved in the proposed formulation. A similar methodology is extended to solve the switched ARX model estimation problem, in which we consider the \(\ell_\infty\) worst-case optimal estimator. This problem is ubiquitous in applications ranging from manufacturing processes, communication systems and biology to reconfigurable control. In this scenario, we consider the sparse structure available on the information of switching signal. This information, which has the binary pattern, indicates which subsystem is active at each time instance. The problem is known to be NP-hard when the switching signal is not observable, which is the case we are interested in.
On the other hand, controller design under sparse structure constraints has also drawn substantial research interest during the last decade. This problem is also known as decentralized control. In [10, 11, 12], it has been shown that all the stabilizing controllers can be parametrized using a convex form with respect to a free parameter. The parametrization is usually known as Q parametrization, Youla parametrization or YBJK parametrization. However, the convexity of this parametrization may fail to hold, once the sparse structure constraints are considered while designing the controller. Given the difficulty and even NP-hardness[13, 14] of the design problem, many pioneer works have been devoted to the study of practical convex relaxations. Depending on the memory strategies, the existing methods can be divided into two classes: the static control and the dynamic control. Although these methods work well in their scopes, restricted assumptions are usually required either on the sparsity or the structure of variables. In consideration of the restrictiveness, we present two convex approaches for static and dynamic control under general sparse structure. Compared to the existing methods, the proposed methods are based on more general formulations, which lead to the success of proposed approaches while other methods are not applicable.

In the third case of the information structured system design, we present an efficient approach to identifying sparse dynamical graphical models from experimental input/output data. Each node of the dynamical graphical model is associated with a time series, while the edges relate the values of these series at different time instants. This problem is a generalization to the dynamical setting of the classical graphical modeling in statistical analysis. Applications can be found in fields as diverse as neuroscience, systems biology and chemistry, economics and video-analytics. While identifying a dynamical graphical model is generally ill posed due to the potentially infinite number of topologies, the sparse structure is usually enforced on the edges as the regularization. This leads to the identification of the sparse dynamical graphical model. In the proposed work, we show that this problem can be efficiently solved by recasting it into an expanded atomic-norm minimization framework that allows for enforcing block-sparsity. The approach leads to computationally efficient algorithms capable of comfortably handling large data sets, unknown inputs and fragmented data records. It is worth mentioning that the proposed algorithms achieve overwhelming efficiency in large-scaled cases, as compared with the latest commercial solvers.
1.2 Regression Modeling with the Sensor Fault

Given the collected data, the linear regression model has been widely used in many decision making problems. In the second category, we discuss the linear regression problem, considering a practical situation where part of the data-collecting sensors fail to work, i.e., sensor fault. Sensor fault is a general phenomenon in large-scaled network systems. Given the massive and widely deployed sensing devices, it’s nearly impossible to require that each device functions through the entire task. As the consequence of the sensor fault, the data collected will not only be corrupted by noise, but also contain outliers. Therefore, it is essential to reject the outliers while estimating the linear regression model from the experimental data, which leads to the intensive studies of robust regression. While directly exploiting the sparse structure of outliers in this problem leads to NP-hardness, several tractable relaxations[15, 16, 17, 18] have been recently proposed for linear robust regression along with the theoretical conditions guaranteeing exact recovery of the model parameters. However, these relaxations may perform poorly in cases where the fitting error for the outliers is very large. In the proposed work, we introduce a self-scaled $\ell_1$ regularized robust regression approach, where the cost function is automatically scaled with scalings that depend on the a-priori information. Thus, the proposed approach achieves substantially better performance than the traditional regularized approaches in cases where the outliers are far from the linear manifold spanned by the inliers, while at the same time exhibits the same theoretical recovery properties.

1.3 GARCH Model Identification of the Covariance Feature

Given a large number of data samples in a practical task, it is usually unnecessary to dive into the information of each sample individually. This is not only because the computational cost considerations, but also because the noise and outliers in the data can lead to incorrect decisions. This difficulties can be circumvented by exploiting stochastic properties of the data. A widely used stochastic informations is the covariance matrix. It can be easily observed that the size of the covariance matrix is immune to the number of samples, which makes it a reasonable choice for data representation in big data problems. This compressed representation, also known as covariance feature, has shown many successes in various applications[19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37].

In practice, to achieve better performance, it is necessary to update the covariance features, which reduces to the problem of estimating the future values of a Positive Definite matrix from the
past historical data. A potential difficulty here stems from the nature of the positive definite matrices. It is known that covariance matrices live on a manifold instead of Euclidean space. Therefore, it is necessary to consider the manifold structure when designing the covariance feature updating methods. While several methods have been proposed in the literature\cite{20, 27} for propagating past data in this context, the problem of identifying the propagation models from experimental data is still open. In order to identify a “manifold-aware” model, we proposed a local convex approach based on a geometry-aware distance metric, called \textit{Jensen Bregman LogDet Deivergence (JBLD)}\cite{26}. By further combining this metric with on-manifold \textit{generalized auto-regressive conditional heteroskedasticity (GARCH)} models, we show that identifying a geometry-aware on-manifold model from experimental data can be solved efficiently by considering a convex optimization problem. Finally, by exploiting the identified model in a Bayesian maximum likelihood setup, we propose an optimal filter for updating the covariance features.

1.4 Organization of the Dissertation

The rest of the dissertation is organized as follows: In section 1.5, we summarize the notations used throughout this dissertation. Then we introduce the relevant background knowledge in chapter 2, which includes moment-theory based polynomial optimization techniques, \(Q\) parametrization, atomic-norm based representation and the Frank-Wolfe algorithm. Building on these convex optimization techniques, we present our main contributions in chapters 3–8. We first discuss the worse-case optimal estimator design based on sensor selection strategy in chapter 3. Then, based on a similar methodology, \(\ell_\infty\) worst-case optimal estimator for switched ARX system is presented in chapter 4. In chapter 5, we introduce the convex approaches for sparse structured static and dynamic controller design. The super-atom norm based efficient identification method of sparse dynamical graphical models is proposed in chapter 6. For the second category, we present the self scaled regularized robust regression approach in chapter 7. Finally, we present the geometry-aware on-manifold optimal filter for the covariance feature propagation in chapter 8. In chapter 9, we summarize the main contributions of this thesis and point out to several directions for future research.

1.5 Notations

\(\mathbb{R}\) \hspace{1cm} set of real numbers
\(\mathbb{N}_n\) \hspace{1cm} set of integers from 1 to \(n\)
CHAPTER 1. INTRODUCTION

$S^n$ set of symmetric matrices in $\mathbb{R}^{n \times n}$

$S^n_+ (S^n_{++})$ set of positive-semidefinite (-definite) matrix in $S^n$

$\text{conv}(\mathcal{A})$ Convex hull of the set $\mathcal{A}$

$x$ a vector in $\mathbb{R}^n$

$x^{(j)}, x_j$ $j^{th}$ component of the vector $x$

$\|x\|_2$ $\ell_2$ norm of a vector: $\|x\|_2^2 = \sum_i x_i^2$

$\|x\|_0$ $\ell_0$ quasi-norm, number of non-zero elements in $x$

$\|x\|_\infty$ $\ell_\infty$ norm, $\|x\|_\infty \doteq \max_i |x_i|$

$\|\{x\}\|_p$ $\ell_p$ norm of a vector valued sequence $\{x_i\}_{i=1}^m$ where each $x_i \in \mathbb{R}^N$

$\|\{x\}\|_p \doteq \left( \sum_{i=1}^m \|x_i\|_p^p \right)^{1/p}, 1 \leq p < \infty$

$\|\{x\}\|_\infty \doteq \max_{1 \leq i \leq m} \|x_i\|_\infty$

$\|\{x\}\|_o$ $\ell_o$-quasinorm $\doteq$ number of non-zero vectors in the sequence (i.e. cardinality of the set $\{i|x_i \neq 0, i \in [1, m]\}$)

$M$ a matrix in $\mathbb{R}^{n \times m}$

$e_i$ $i^{th}$ vector of the canonical basis: $e_i \doteq [0, \ldots, 1, \ldots 0]^T$

$I$ Identity matrix

$I_{\mathcal{I}}$ Matrix whose columns are the vectors $e_i$ corresponding to indexes $i \in \mathcal{I} \subset \mathbb{N}_n$.

$M \succeq N$ the matrix $M - N$ is positive semidefinite

$\sigma_i(M)$ the $i$-th largest singular value of matrix $M$

$\sigma(M) (\sigma(M))$ maximum (minimum) singular value of $M$

$\|M\|_F$ Frobenious norm of the matrix $M$

$|M|, \det[M]$ determinant of $M$

$M(j,:)$ $j^{th}$ row of the matrix $M$

$M(:,j)$ $j^{th}$ column of matrix $M$.

$|E|$ cardinality (e.g. number of elements) of the set $E$

$\mathcal{N}(\mu, \Sigma)$ multivariate Gaussian distribution with mean $\mu$ and covariance $\Sigma$

$M^{1/2}$ for $M \in S^n_+, M^{1/2} = V S^{1/2} V^T$ where $V SV^T$ is the svd of $M$

$\text{Proj}_+(M)$ projection of $M$ onto $S_+$ using eigenvalue decomposition

$O(x^n)$ infinitesimal of order $x^n$ as $x \to 0$
1.5. NOTATIONS

Pattern(.) For any transfer matrix $G \in \mathbb{R}^{m \times n}$, Pattern($G$) is a binary matrix with entries

$$[\text{Pattern}(G)]_{i,j} = \begin{cases} 0 & \text{if } G_{i,j} = 0 \\ 1 & \text{otherwise} \end{cases}$$

$\ell_2$ Banach spaces of vector valued real sequences with finite energy, equipped with the norm

$$\|y\|_{\ell_2}^2 \doteq \sum_{i=0}^{\infty} \|y_i\|^2$$

$\mathcal{B}\ell_2$ unit ball in $\ell_2$.

$\mathcal{H}_\infty$ Banach space of transfer matrices analytic outside the unit disk, equipped with the norm

$$\|G(z)\|_{\mathcal{H}_\infty} = \sup_{0 \leq \theta < 2\pi} \left\{ \sigma_{\max} \left[ G(e^{j\theta}) \right] \right\}$$

$\mathcal{RH}_\infty$ subspace of $\mathcal{H}_\infty$ formed by real rational transfer matrices.

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}$$ Compact representation of a Linear Time Invariant (LTI) systems with a model of the form

$$x_{k+1} = Ax_k + Bw_k$$

$$y_k = Cx_k + Dw_k$$

Alternatively these systems will be represented using the transfer matrix $G(z) \doteq C(zI - A)^{-1}B + D$.

$\mathcal{F}_\ell(G, C)$ Lower Linear Fractional Transformation:

$$\mathcal{F}_\ell(G, C) \doteq G_{11} + G_{12}C(I - G_{22}C)^{-1}G_{21}$$

$G^-$ Adjoint of the transfer matrix $G(z)$: $G(z)^- \doteq GT\left(\frac{1}{z}\right)$. 
Chapter 2

Background

2.1 Moments: An SDP Relaxation Method with Global Optimality

Given a compact semi-algebraic set $\mathcal{K} \subset \mathbb{R}^n$, which is defined by $l$ multivariate polynomial inequalities as

$$\mathcal{K} = \{ \mathbf{x} \in \mathbb{R}^n \mid g_k(\mathbf{x}) \geq 0, \ k = 1, \cdots, l \}$$

and the polynomial optimization problem of the form

$$\min_{\mathbf{x}} \ p(\mathbf{x})$$

$$\text{s.t.} \quad \mathbf{x} \in \mathcal{K}$$

If we are only interested in local minimum, plenty of methods can be used, such as optimality conditions and descent algorithms using basic tools from real and convex analysis and linear algebra. Sometimes, we also can relax this problem into a convex surrogate and solve this convex relaxation, which doesn’t guarantee to find the global optimal solution too. However, under many circumstances, a global optimal solution is important. In order to find the global minimum of (2.2), the property that $p(\mathbf{x})$ and $g_k(\mathbf{x})$ are polynomials can help. In the next, we are going to introduce a SDP (semi-definite programming) relaxation, known as moments, that has been shown of ability to find the global optimal solution of polynomial optimizations. Afterwards, an extension about solving 0-1 optimization problem using moments will be introduced, as a tool of solving engineering problems discussed in this dissertation.
2.1. MOMENTS: AN SDP RELAXATION METHOD WITH GLOBAL OPTIMALITY

2.1.1 The General Problem of Moments

Apparently, finding the global minimum $p^*$ of polynomial optimization problem (2.2) is equivalent to solve

$$
p^* = \sup \lambda \\
\text{s.t. } p(x) - \lambda \geq 0 \\
x \in \mathcal{K}
$$

(2.3)

following Putinar’s Positivstellensatz, above problem can be recast as characterizing the nonnegative polynomial

$$
p(x) - p^* = \sigma_0(x) + \sum_{k=1}^{l} \sigma_k(x)g_k(x)
$$

(2.4)

for some SOS polynomials $(\sigma_k) \subset \mathbb{R}[x]$. Based on the theory of moments, which is in duality with the theory of nonnegative polynomials and Hilbert’s 17th problem on the representation of nonnegative polynomial, [38] has shown that the polynomial optimization problem (2.2) can be replaced as the equivalent problem:

$$
p^* = \min_{\mu \in \mathcal{P}(\mathcal{K})} \int p(x)\mu(dx)
$$

(2.5)

and furthermore, reduced into a monotonically convergent sequence of Linear Matrix Inequality (LMI) optimization problems.

In equation (2.5), $\mathcal{P}(\mathcal{K})$ denotes the space of finite Borel signed measures on $\mathcal{K}$. Determine whether there exists such a measure $\mu$ in $\mathcal{P}(\mathcal{K})$ with $\{m_\alpha\}$ as the $\alpha$-th moments under $\mu$ is called the $\mathcal{K}$-moment problem. $\{m_\alpha\}$ is defined as:

$$
m_\alpha = E_\mu(x^\alpha) = \int x^\alpha \mu(dx)
$$

(2.6)

where $x^\alpha = x_1^{\alpha_1}x_2^{\alpha_2} \cdots x_n^{\alpha_n}$. In [38], they have shown that the existence of $\mu$ is equivalent to several positive semi-definite constraints on the so called moment matrix $M(m_\alpha)$ and localizing matrix $L(g_km_\alpha)$. Next, let’s introduce the truncated version of moment matrix and localizing matrix of order $T$, containing moments of order up to $2T$. Given a basis of monomials up to order $T$ as:

$$\{1, x_1, x_2, \cdots, x_1^2, x_1x_2, \cdots, x_1x_n, x_2x_3, \cdots, x_n^2, \cdots, x_1^T, \cdots, x_n^T\}
$$

(2.7)

let’s denote each element from the basis as $x^{\alpha_i}$ for $i = 1, 2, \cdots, \left(\begin{array}{c}n+T \\ n\end{array}\right)$. Then the truncated moment matrix of order $T$ is given by

$$M_T(m)_{i,j} = m_{\alpha_i+\alpha_j}, \quad \text{for all } i,j \leq \left(\begin{array}{c}n+T \\ n\end{array}\right)
$$

(2.8)
Similarly, given \( g_k(x) = \sum_{\beta} g_{k,\beta} x^{\beta} \), where \( g_{k,\beta} \) denotes the coefficients of monomial \( x^{\beta} \), the corresponding truncated localizing matrix of order \( T \) will be written as

\[
L_T(g_k m)_{i,j} = \sum_{\beta} g_{k,\beta} m_{\beta_i + \alpha_i + \alpha_j}, \quad \text{for all } i, j \leq \binom{n + T - \delta_k}{n}
\]  

(2.9)

where, \( \delta_k \) is the polynomial degree of \( g_k(x) \). Based on the truncated moment matrix and localizing matrix, we can give the form of the monotonically convergent sequence of LMI mentioned above. Write the polynomial \( p(x) \) in (2.2) as

\[
p(x) = \sum_{\alpha} p_{\alpha} x^{\alpha}
\]

(2.10)

the LMI problem is given by

\[
p_T^* = \min_m \sum_{\alpha} p_{\alpha} m_{\alpha} \\
\text{s.t. } M_T(m) \succeq 0 \\
L_T(g_k m) \succeq 0
\]

(2.11)

with \( p_T^* \to p^* \) from below, as \( T \to \infty \).

Next, let’s introduce a property allowing the relaxation sequence to stop in early order of moment matrix. If for some \( T \), we have \( \text{rank}[M_T(m)] = \text{rank}[M_{T-1}(m)] \), the so called flat extension property, then the \( T \)-th order relaxation is indeed exact[39].

### 2.1.2 Extension on 0-1 Optimization

This section is concerned with the 0-1 polynomial optimization problem. Consider a polynomial optimization problem with constraints that the variables has to be either 0 or 1:

\[
\min_x p(x) \\
\text{s.t. } x \in \mathcal{K} \\
x_i \in \{0, 1\}
\]

(2.12)

where, \( \mathcal{K} \subset \mathbb{R}^n \) is a compact semi-algebraic set defined in (2.1). The 0-1 constraint is well known to be difficult to deal with due to its non-convex property. However, this constraint can be substituted as the equivalent polynomial type constraint \( x_i = x_i^2 \), in which case problem (2.12) can be treated as a general polynomial optimization problem and solved as a sequence of SDP problems. Even better part is, as shown in [40], any order moment problems for problem (2.12) involve at most \( 2^n - 1 \).
2.2. Q PARAMETRIZATION

moment variables and for all $T > n$,

$$\text{rank}[M_T(m)] = \text{rank}[M_n(m)]$$ (2.13)
	herefore the relaxation sequence is guaranteed to find optimal solution for some finite order, which is also the global minimum of problem (2.12).

2.2 Q Parametrization

In previous section, we have introduced a convex relaxation for polynomial optimization, which is generally difficult due to non-linearity property. In this section, we are going to talk about convex parametrization of all stabilizing controllers and admissible filters given a system plant, which is referred as Q parametrization in the following.

In control aspect, this parametrization, which is also known as Youla or YBJK parametrization, was developed by Youla and coworkers in the continuous time case[10, 11] and, independently, by Kucera[12] in discrete time case. The advantage of this parametrization is, briefly speaking, that all the closed-loop stabilizing controller given a plant can be parametrized in terms of a free variable $Q(s)$, which is a stable proper plant.

**Definition 1.** If $F(s)$ is real-rational, then $F \in \mathbb{RH}_\infty$ if and only if $F$ is proper and stable.

**Definition 2.** Two matrices $F$ and $G$ is in $\mathbb{RH}_\infty$ are right-coprime (over $\mathbb{RH}_\infty$) if they have equal number of columns and there exists matrices $X$ and $Y$ in $\mathbb{RH}_\infty$ such that

$$\begin{bmatrix} X & Y \end{bmatrix} \begin{bmatrix} F \\ G \end{bmatrix} = XF + YG = I$$

Similarly with left-coprime.

**Definition 3.** Let $G$ be a proper real-rational matrix, right-coprime factorization of $G$ is

$$G = N_r M_r^{-1}$$

where $N_r$ and $M_r$ are right-coprime matrices in $\mathbb{RH}_\infty$. Left-coprime factorization of $G$ is

$$G = M_l^{-1} N_l$$

where $N_l$ and $M_l$ are left-coprime.
Notice that $M_r$ and $M_l$ in Definition 3 is implied to be square and non-singular. Next let’s look at a special coprime factorization:

**Lemma 1.** [41] For each proper real-rational matrix $G$ there exist eight $\mathcal{RH}_\infty$-matrices satisfying the equations

$$G = N_r M_r^{-1} = M_l^{-1} N_l$$  \hspace{1cm} (2.14)

$$
\begin{bmatrix}
X_l & -Y_l \\
-N_l & M_l
\end{bmatrix}
\begin{bmatrix}
M_r & Y_r \\
N_r & X_r
\end{bmatrix}
= I 
$$  \hspace{1cm} (2.15)

Given a coprime factorization, in [11], it has been shown that all the stabilizing controllers can be parametrized with a stable proper parameter $Q(s)$ and the resulting closed-loop maps with the stabilizing controllers are affine with respect to $Q(s)$. This development allows finding a optimal stabilizing controller using convex optimization methods. Bring in double coprime factorization in Lemma 1, in [41], it shows that

**Theorem 1.** The set of all controllers (proper) stabilizing $G$ is

$$C_{\text{stab}} = \{(Y_r - M_r Q)(X_r - N_l Q)^{-1} \mid Q \in \mathcal{RH}_\infty\}$$

$$= \{(X_l - Q N_l)^{-1}(Y_l - Q M_l) \mid Q \in \mathcal{RH}_\infty\}$$  \hspace{1cm} (2.16)

![Figure 2.1: Linear fractional interconnection of $P$ and $K$](image)

Consider a plant shown in Fig 2.1, suppose $P$ is stabilizable, which means that $P$ and $G$ share the same unstable poles, then given the stabilizing controller defined in equation (2.16), we can derive the closed-loop map from $w$ to $z$ is

$$f(P, K) = P_{11} + P_{12}(I - KG)^{-1}KP_{21}$$

$$= P_{11} + P_{12}M_r(Y_l - QM_l)P_{21}$$

$$= T_1 + T_2QT_3$$  \hspace{1cm} (2.17)
2.2. Q PARAMETRIZATION

where $T_1 = P_{11} + P_{12}M_rY_rP_{21}$, $T_2 = -P_{12}M_r$ and $T_3 = M_rP_{21}$.

Analogous to Q parametrization in controller design, there exists a well known parametrizing method for all admissible filter, see for instance [42, 43, 44]. Consider a linear time-invariant system as

$$x_{k+1} = Ax_k + Bw_k$$
$$z_k = C_1x_k$$
$$y_k = C_2x_k + Dw_k$$

(2.18)

where, $w$ is a vector containing both processing noise and measurement noise and $y$ denotes measurement. The filtering problem is to estimate $z$ given measurement $y$. Let’s first introduce a definition for admissible filter:

**Definition 4.** Given system (2.18), an LTI filter $F$ is said to be admissible if it’s real rational, stable and estimation error $e \to 0$ for any initial condition $x_0$ given $w_k \equiv 0$.

As shown in [43, 44], all admissible filters can be parametrized as an affine equation with respect to a parameter $Q(s)$, which is required to be stable.

**Theorem 2.** Given system (2.18), suppose $(A, C_2)$ is detectable, let $L$ be any real matrix such that $(A - LC_2)$ is asymptotically stable, then all admissible filters can be written as formula that

$$F = F_1 + QF_2$$

(2.19)

with

$$F_1 = \begin{bmatrix}
A - LC_2 & L \\
C_1 & 0
\end{bmatrix}$$

$$F_2 = \begin{bmatrix}
A - LC_2 & L \\
-Rc^{-1/2}C_2 & Rc^{-1/2}
\end{bmatrix}$$

(2.20)

where, $R_c > 0$ is a scaling matrix. Using this admissible filter form, the resulting map from noise $w$ to estimation error $e$ is $T_{ew} = T_1 + QT_2$ with

$$T_1 = \begin{bmatrix}
A - LC_2 & B - LD \\
C_1 & 0
\end{bmatrix}$$

$$T_2 = \begin{bmatrix}
A - LC_2 & B - LD \\
-Rc^{-1/2}C_2 & -Rc^{-1/2}D
\end{bmatrix}$$

(2.21)

The map from noise to estimation error is apparently affine with respect to parameter $Q$. 

Based on this scheme, in [45], it is shown that a filter with sparse structure can be solved by manipulating the freedom in parameter $Q$ to do element cancellation, resulting in a convex optimization problem as below

$$
\min_Q \|T_1 + QT_2\|_2
$$

s.t. $Q \in \mathcal{RH}_\infty$

$$
\Phi = F_1 + QF_2 \in S
$$

where $S$ denotes a subspace satisfying the given sparse pattern. By approximating stable plant with FIR models, above problem results in

$$
\min_Q \sum_{i=0}^{n-1} \|Q_i\|_2^2
$$

s.t. $Q = \sum_{i=0}^{n-1} Q_i z^{-i}$

$$
(Q)_r(F_2)^c = -(F_1)_r, \forall (r,c) \in C
$$

where, $C = \{(r,c) \mid S(r,c) = 0\}$.

Obviously, using Q parametrization (for both controller and filter case), we can parametrize all candidate controller or filter as an affine form with respect to the stable, proper parameter $Q(s)$, this renders adopting convex optimization methods to design controller and observer with some (convex) norm described performance specification to achieve optimality. This advantage will be taken in this dissertation.

### 2.3 Atomic Norm Representation and Frank-Wolfe Algorithm

In this section, we first describe an unified convex formulation which induces sparsity in more general case. Then we give a brief introduction on a projection-free first order method, known as Frank-Wolfe algorithm. The Frank-Wolfe method has been firstly introduced in [46], and re-gained much interests in recent years due to computationally efficient performance in several areas. For a latest review, interested reader is referred to [47] and references therein.

#### 2.3.1 Atomic Norm and Sparsity

Recently, in applications throughout science and engineering, the “sparsity” penalty is a general idea, used to search for information of low complexity, or model of simple structure, i.e.
sparse vector and low rank matrix. With respect to different scenario, various convex penalties are used, for instance \( \ell_1 \) norm for the sparse cardinality of vector, nuclear norm for the low rank of matrix. In [48], an unified convex formulation is introduced for inducing sparsity, which is known as atomic norm.

In order to define an atomic norm, a general model is considered. In the model, a target is written as a nonnegative combination of a few elements from an atomic set

\[
x = \sum_{i=1}^{k} c_i a_i , \quad a_i \in \mathcal{A}, c_i \geq 0
\]  

(2.24)

where, \( \mathcal{A} \) is atomic set containing (possibly infinite) elementary atoms. Each atom represents the simplest building block with respect to the target. For example, an atom can be defined as the unit-norm one-sparse vectors (i.e. \( e_i \)) when \( x \) denotes a sparse vector, or the unit-norm rank-one matrix when \( x \) is a low rank matrix. It’s worth mentioning that the atomic set containing unit-norm rank-one matrices is an infinite set.

Following the model (2.24), a gauge function can be induced given a compact set of atoms \( \mathcal{A}[49] \). Let \( \text{conv}(\mathcal{A}) \) be the convex hull, whose extreme points are the elements \( a_i \in \mathcal{A} \), we have the gauge function \( \| \cdot \|_\mathcal{A} \) denoted as

\[
\|x\|_\mathcal{A} = \inf \{ t > 0 : x \in t \text{conv}(\mathcal{A}) \}
\]  

(2.25)

this gauge function is always a convex, extended real-valued function for any set \( \mathcal{A} \). By appropriate recentering the convex hull, it can be rewritten as[50]

\[
\|x\|_\mathcal{A} = \inf \left\{ \sum_{a \in \mathcal{A}} c_a : x = \sum_{a \in \mathcal{A}} c_a a, c_a \geq 0, \forall a \in \mathcal{A} \right\}
\]  

(2.26)

If \( \mathcal{A} \) is centrally symmetric about the origin, this gauge function is a norm, which is called atomic norm induced by \( \mathcal{A} \). It can be easily shown that an equivalent definition is

\[
\|x\|_\mathcal{A} = \inf \left\{ \sum_{a \in \mathcal{A}} |c_a| : x = \sum_{a \in \mathcal{A}} c_a a \right\}
\]  

(2.27)

The support function of the convex hull \( \text{conv}(\mathcal{A}) \) is given by

\[
\|x\|^{\ast}_\mathcal{A} = \sup \{ \langle x, s \rangle : s \in \text{conv}(\mathcal{A}) \}
\]  

(2.28)

If \( \| \cdot \|_\mathcal{A} \) is a norm, then the support function is exactly its dual norm. Notice that \( \text{conv}(\mathcal{A}) \) is a convex hull despite of the structure of atoms \( a \in \mathcal{A} \). Therefore, from the definition of convex hull, it
can be directly shown that the support function of $\text{conv} (A)$ can be evaluated only on the extreme points of the convex hull, i.e. the atoms,

$$\|x\|_A^* = \sup\{ \langle x, a \rangle : a \in A \}$$  \hspace{1cm} (2.29)

This is often easier to compute than a maximum over the full domain $\text{conv} (A)$. As pointed in [47], this is the key property which enables the efficient application of the Frank-Wolfe algorithm for atomic domains. A brief introduction of Frank-Wolfe type algorithm will be given in the following section.

The effectiveness of using atomic norm in convex optimization method searching sparse solution has been shown in [48], and an analysis of exact and robust recovery is given therein. Besides inducing sparsity of solution, it is also worth mentioning that, by building proper atoms, one can actually involves the physical meaning of the practical problem. For instance, in order to identify a rational proper transfer function, [51] built the atoms as the simple rational transfer functions with single real pole or pair of complex conjugate poles.

### 2.3.2 Frank-Wolfe Algorithm

In previous section, we have introduced a general unified convex formulation with the goal of recovering the sparse solution. However, there is always another question one should answer, which is the computational issue of designing efficient algorithm for the formulated problem.

In [48], the project gradient descent method is proposed to solve the atomic norm minimization problem. In addition, proximal methods are also used to solve atomic norm minimization. While the project gradient descent and proximal methods works well on atomic norm minimization, a range of interests have been given to the Frank-Wolfe algorithm [46] due to its good scalability and ability to remain sparse combination of atoms. Differing from the project gradient descent and proximal methods which involves a quadratic problem in each iteration, Frank-Wolfe algorithm solves a linear approximation of the objective function. One of the many variants of Frank-Wolfe type algorithms are introduced in Algorithm 1, considering a convex optimization problems of the form:

$$\min_{x} f(x)$$

s.t. \hspace{1cm} $\|x\|_A \leq \tau$  \hspace{1cm} (2.30)

where, the atomic norm constraint is used to induce sparsity. Further, it has been shown in [47] that above problem can be solved efficiently with convergence rate of $O(\frac{1}{t})$. 
Algorithm 1 Generic Frank-Wolfe algorithm to minimize a convex function over the $\tau$-scaled atomic norm ball

1. **Initialize:** $x_0 \leftarrow \tau a_0$ for arbitrary $a_0 \in A$

2. **for** $t = 0, 1, 2, \cdots$ **do**

3. $a_t \leftarrow \arg\min_{a \in A} \langle \partial f(x_t), a \rangle$

4. $\alpha_t \leftarrow \arg\min_{\alpha \in [0,1]} f(x_t + \alpha[\tau a_t - x_t])$

5. $x_{t+1} \leftarrow x_t + \alpha_t[\tau a_t - x_t]$

6. **end for**
Chapter 3

Worst-case Optimal Estimator based on Sensor Selection

3.1 Previous Works and Problem Statement

In this chapter, we discuss an design approach of worst-case optimal estimator based on the sensor selection strategy. Among the existing works regarding sensor selection problems, a large portion of efforts have been devoted to stochastic approach, where the authors want to select a subset of all sensors minimizing the covariance of estimation error\cite{3} or the Shannon entropy\cite{52}. Alternative efforts have been devoted to the topic of sensor placement, in which the aim is to cover as much area or many targets as possible with minimum number of sensors. A detailed survey can be found in \cite{53}. In this chapter, we propose a convex optimization based sensor selection algorithm considering the case in which the noise is known to belong to a norm bounded convex set. This scenario is contrary to most of the relaxation approaches\cite{3, 4, 5, 6, 7, 8, 9} which consider noise with Gaussian distribution.

In the following sections, by involving the idea of Information Based Complexity (IBC)\cite{54}, we introduce a set-membership method choosing a subset of sensors from all candidates with the goal of minimizing the worst-case estimation error over the set of estimate consistent with the noise description. The estimation is called worst-case optimal. In \cite{55}, the worst-case optimal estimation problem has been introduced under $\ell_2$ and $\ell_\infty$ norm description with all sensors available. The proposed results motivate our approach in the following.

Consider three Banach spaces, $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ over the real field, a linear operator $S_y : \mathcal{X} \rightarrow$
3.1. PREVIOUS WORKS AND PROBLEM STATEMENT

\( Y \) of the form that \( y = Cx + \eta \), where \( C \) is of full column rank and \( \eta \) denotes noise only known to belong to a bounded set \( \mathcal{N} \). Then given a measurement \( y \in \mathcal{Y} \), we can form a consistent set of all possible \( x \) that can generate this measurement:

\[
\mathcal{T}(y) = \{ x \in \mathcal{X} \mid y = Cx + \eta, \ \eta \in \mathcal{N} \} \tag{3.1}
\]

Let \( S_z : \mathcal{X} \rightarrow \mathcal{Z} \) denotes another linear operator, the goal of IBC based worst-case estimation is to design an optimal estimator of \( z \) from \( y \).

An estimation algorithm \( A \) generating estimation of \( z \) as \( \hat{z} = A(y) \) for a given measurement \( y \) associates an local error

\[
e(A, y) \doteq \sup_{x \in \mathcal{T}(y)} \| S_z x - A(y) \|
\]

then, the global error is defined as

\[
e(A) \doteq \sup_y e(A, y) \tag{3.3}
\]

Therefore, an estimation algorithm is called globally optimal if it generates least global error, intuitively speaking, minimizes the worst-case error

\[
e(A_o) = e_o \doteq \inf_A e(A) \tag{3.4}
\]

In [56, 57, 55, 58], several globally optimal algorithms have been extensively studied. An important result from [55] shows that in the \( \ell_2 \) or \( \ell_\infty \) norm equipped case, the worst-case optimal estimator is of linear form such as

\[
\hat{z} = S_z C^\dagger y
\]

with \( C^\dagger \) denotes a suitably constructed left inverse of \( C \). Apparently in this case, given \( \mathcal{N} \) as an \( \varepsilon \)-ball of \( \ell_p \) norm, estimating \( z \) in \( \ell_q \) norm equipped space generates an optimal worst-case error

\[
e_o = \| S_z C^\dagger \|_{\ell_p \rightarrow \ell_q} \cdot \varepsilon \tag{3.6}
\]

In the case that no additional constraint is required on the selection scheme, [55] has shown that: 1) in the \( \ell_2 \) equipped case, the worst-case optimal estimator involves all available sensors; 2) in the \( \ell_\infty \) equipped case, a subset, with cardinality equalling dimension of \( \mathcal{X} \), of all available sensors is involved in \( C^\dagger \). However, in practice, there are many situations that the sensors should be manipulated under some additional constraints, such as transmission medium access constraints or power consideration. These extra requirements may prohibit certain sensors from simultaneously active, or limit the number of total active sensors in a subset.
In the following section of this chapter, we will introduce a convex optimization based procedure of building the worst-case optimal estimator based on selecting sensors, and show that some additional constraints on the selecting scheme can be handled easily by the proposed method. The problem of interest can be described as follows,

**Problem 1.** Given $N$ candidates and some selection constraints, decide a strategy for choosing no more than $n$ sensors such that the resulting estimator

$$\hat{z} = S_z C_{\text{selected}}^\dagger y_{\text{selected}}$$

(3.7)

is globally (worst-case) optimal.

Formally speaking, Problem 1 can be treated as an optimization problem

$$\min C_{s} \| S_z C_{s}^\dagger \|$$

(3.8)

where, $C_s, s = 1, 2, \cdots, n_s \leq \binom{N}{n}$ is all admissible choice of $n$ sensors from $N$ candidates, and $\| \cdot \|$ denotes a suitable norm. Obviously, this is a combinatorial optimization problem, which is generally known to be NP-hard. Next, we will show how to manipulate this formation into a polynomial optimization problem and use the moments-based method to achieve the global optimal solution.

### 3.2 Proposed Solution

In this section, we first show how to deduce Problem 1 into a solvable formation in $\ell_2$ norm equipped space. Then we extend the idea with $\ell_\infty$ norm case. For simplicity, the information which is going to be estimated is assumed to be a scalar $z$. This assumption is without loss of generality in the $\ell_\infty$ case, since vector optimal estimators simply consist of the optimal scalar estimator $\hat{z}_i$ of each component of $z$[55]. As goes to the $\ell_2$ case, it is equivalent to designing independent estimators (hence potentially using different set of sensors) for each component, rather than a single coupled estimator.

#### 3.2.1 $\ell_2$ Bounded Noise

For a given choice $s$, the $C^\dagger$ in equation (3.6) can be written as $(C_s^T C_s)^{-1} C_s^T$, where $C_s$ is the submatrix formed by the chosen sensors. Define 0-1 variables of number $N$, say $v_i$ for $i = 1, 2, \cdots, N$, and assign each of them to one sensor in $C$, we can form a matrix as $C_v \equiv$


3.2. PROPOSED SOLUTION

\[
\begin{bmatrix}
 v_1 c_1^T & v_2 c_2^T & \cdots & v_N c_N^T
\end{bmatrix}^T,
\]
where \( c_i \) denotes the \( i \)-th row of \( C \). Therefore, the problem (3.8) can be recast as

\[
\begin{aligned}
\min & \quad \| S_z (C_v^T C_v)^{-1} C_v^T \|_2 \\
\text{s.t.} & \quad 1^T v = n \\
& \quad v_i \in \{0, 1\}, \forall i = 1, 2, \cdots, N \\
& \quad v \in S_v
\end{aligned}
\]  

(3.9)

where, 1 is a vector with all entries of one, and \( S_v \) denotes the set of admissible selections. Realize that the objective function in (3.9) can be rewritten as a rational polynomial form as

\[
\begin{aligned}
\| S_z (C_v^T C_v)^{-1} C_v^T \|_2^2 \\
= \left\| S_z \frac{\text{adj}(C_v^T C_v)}{\text{det}(C_v^T C_v)} C_v^T \right\|_2^2 \\
= \| P_v \|_2^2 \\
= \frac{p^2(v)}{d^2(v)}
\end{aligned}
\]  

(3.10)

where, \( \text{adj}(\cdot) \) and \( \text{det}(\cdot) \) represents the adjugate operation and determinant operation of a given matrix respectively, and \( P(v) = \left[ p_1(v) \ p_2(v) \ \cdots \ p_N(v) \right] \) is a vector with entries as polynomials in the variable of \( v \). Since \( d(v) = \text{det}(C_v^T C_v) \) is also a polynomial, in equation (3.10), we result in a rational polynomial. In order to write a polynomial optimization form, we introduce an extra variable \( \beta \), such that \( \frac{p^2(v)}{d^2(v)} = \beta \). Therefore, the problem (3.9) is equivalent to the following polynomial optimization problem

\[
\begin{aligned}
\min & \quad \beta \\
\text{s.t.} & \quad p^2(v) - \beta d^2(v) = 0 \\
& \quad 1^T v = n \\
& \quad v_i - v_i^2 = 0, \forall i = 1, \cdots, N \\
& \quad v \in S_v
\end{aligned}
\]  

(3.11)

Notice, the constraint \( v_i \in \{0, 1\} \) is enforced by the equality \( v_i - v_i^2 = 0 \). Hence, in the case that \( S_v \) is semi-algebraic, the above problem can be solved by moments-based method.

Next, let’s show some additional selection constraints that we can easily handle:

- no more than \( n_i \) sensors can be simultaneously chosen from a given set \( S_i \).
- sensors \( v_i \) and \( v_j \) cannot be simultaneously chosen.
these constraints can be directly incorporated while building the moment matrix $M(m)$, instead of using additional localizing matrix. For example, in the first case, it can be accomplished by setting the $\alpha$-th moments, $m_\alpha = E_\mu(v_1^{\alpha_1}v_2^{\alpha_2}\cdots v_n^{\alpha_n}) = 0$ whenever $\sum_{i \in S_i} \alpha_i > n_i$. Similarly, the second case can be handled by setting $m_\alpha = 0$ whenever $\alpha_i + \alpha_j > 1$.

Note that in problem (3.11), $\beta$ is the only variable which is not in $\{0, 1\}$. In order to adopt the nice property of result from [40], we perform a line search on $\beta$ instead of treating it as an optimization variable and present a more efficient algorithm in Algorithm 2.

### Algorithm 2 \ell_2 norm based sensor selection

**Initialize:** $\beta_0$ equals to a positive scalar

repeat

Solve

$$v_k = \arg\min_v \ p^2(v) - \beta_k d^2(v)$$

s.t. $1^T v = n$

$$v_i - v_i^2 = 0, \ \forall i = 1, \cdots, N$$

$v \in S_v$

(3.12)

Update

$$\beta_{k+1} = \frac{p^2(v_k)}{d^2(v_k)}$$

(3.13)

until $\beta_{k+1}$ converges.

---

**Remark 1.** Note that, since that $v_i \in \{0, 1\}$ and $\sum_i v_i = n$, each iteration in Algorithm 2 only involves polynomial of degree at most $n$. And according to Theorem 1 in [40], we can state that only moments of order up to $2n$ need to be considered.

**Theorem 3.** The sequence $\beta_k$ in Algorithm 2 converges to the solution of Problem (3.11).

**Proof.** To establish this proof, we begin with showing that the sequence $\beta_k$ is non-increasing. Let $J(v, \beta) = p^2(v) - \beta d^2(v)$, since $v_k$ is a feasible solution of the optimization at step $k + 1$, it follows that $J(v_{k+1}, \beta_{k+1}) \leq J(v_k, \beta_{k+1}) = 0$. Thus $p^2(v_{k+1}) - \beta_{k+1} d^2(v_{k+1}) \leq 0 \Rightarrow \beta_{k+2} = \frac{p^2(v_{k+1})}{d^2(v_{k+1})} \leq \beta_{k+1}$. Since $\beta_k$ is non-increasing and bounded below by zero, it follows that $\beta_k$ converges from above to some limit $\tilde{\beta}$. Similarly, $v_k$ has a convergent subsequence $v_{k_i} \to \tilde{v}$ because it is
bounded. By construction \( p^2(\tilde{u}) - \tilde{\beta}d^2(\tilde{u}) = 0 \), hence, for any \( \epsilon > 0 \), there exists some \( K \) such that

\[
p^2(v_k) - \beta_k d^2(v_k) \geq -\epsilon, \quad \forall k \geq K
\]  

(3.14)

Let \( \beta^* \) and \( v^* \) be the solution of (3.11), then

\[
p^2(v_k) - \beta_k d^2(v_k) \leq p^2(v^*) - \beta_k d^2(v^*)
\]
\[
\leq p^2(v^*) - \tilde{\beta} d^2(v^*)
\]
\[
= -d^2(v^*)(\tilde{\beta} - \beta^*)
\]

(3.15)

If \( \tilde{\beta} > \beta^* \), then choosing \( \epsilon < d^2(v^*)(\tilde{\beta} - \beta^*) \) leads to a contradiction between (3.14) and (3.15).

\[ \square \]

### 3.2.2 \( \ell_\infty \) Bounded Noise

According to [55], in \( \ell_\infty \) norm equipped case, the worst-case optimal estimator has the form of \( \hat{z} = \tilde{C}^{-1}\tilde{y} \), with \( \tilde{C} \) a full rank submatrix formed from rows of \( C \) and the corresponding measurements represented by \( \tilde{y} \). Hence, the corresponding worst-case optimal estimation error is

\[
e_o \doteq \| S_z \tilde{C}^{-1} \|_{\ell_\infty \rightarrow \ell_\infty}
\]

(3.16)

Following similar procedure shown in \( \ell_2 \) norm case, we can have the optimal cost in (3.16) written as

\[
\min \| S_z (C^T_v C_v)^{-1} C^T_v \|_{\ell_\infty \rightarrow \ell_\infty}
\]

s.t. \( 1^T v = n \)

\[
v_i \in \{0, 1\}, \quad \forall i = 1, 2, \cdots, N
\]

\[
v \in S_v
\]

(3.17)

where, it is enforced that exactly \( n \) sensors must be selected from all available. Afterwards, we can derive the following semi-algebraic optimization

\[
\min \frac{\sum_j |p_j(v)|}{d(v)}
\]

s.t. \( 1^T v = n \)

\[
v_i \in \{0, 1\}, \quad \forall i = 1, 2, \cdots, N
\]

\[
v \in S_v
\]

(3.18)
By introduce extra variable $\beta$, it leads to

$$\min_{\beta, x_+, x_-, v} \beta$$

s.t. 

$x_+ \geq 0$

$x_- \geq 0$

$x_+(j) - x_-(j) = p_j(v)$

$$\sum_j [x_+(j) + x_-(j)] = \beta d(v)$$

$1^T v = n$

$v_i - v_i^2 = 0, \forall i = 1, \cdots, N$

$v \in \mathcal{S}_v$

this problem can be reduces into a sequence of LMI optimization problems by adopting moments-based relaxation. It is worth emphasizing that in this case, due to the additional real variables $\beta$, $x_+$ and $x_-$, the result in $\ell_2$ bounded noise case no longer apply. However, as we introduced in section 2.1, the sequence of LMI problems converge monotonically to the optimal solution and provide optimality certificates for finite order relaxation, the so called flat extension property.

### 3.3 Results and Experiments

In this section, we will present two numerical experiments, showing the effectiveness of the proposed method by comparing with two existing methods.

#### 3.3.1 Estimation with Sensor Switching Strategy in Linear System

Consider a linear system:

$$x(k + 1) = \begin{bmatrix} -0.7321 & 0.7321 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} x(k)$$

$$y(k) = C x(k) + \eta(k)$$

$$C = \begin{bmatrix} -4.1466 & -2.7676 & -3.8185 \\ 0.5857 & -0.7998 & -0.1319 \\ 0.1676 & 0.9364 & -0.3085 \end{bmatrix}$$
### 3.3. RESULTS AND EXPERIMENTS

<table>
<thead>
<tr>
<th>Bound on the number of times that sensor 1 can be active</th>
<th>3</th>
<th>2</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>proposed method (time instant, sensor)</td>
<td>{(1,1) (2,1) (3,1)}</td>
<td>{(1,1) (2,1) (3,2)}</td>
<td>{(1,1) (2,2) (3,3)}</td>
</tr>
<tr>
<td>method in [55]</td>
<td>All</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>max-det criterion used in [3]</td>
<td>{(1,1) (2,1) (3,1)}</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>random selection</td>
<td>{(1,1) (2,2) (3,1)}</td>
<td>{(1,2) (2,1) (3,1)}</td>
<td>{(1,1) (2,2) (3,2)}</td>
</tr>
</tbody>
</table>

Table 3.1: Sensor Choices for Different Algorithms

where \(\|\eta\|_2 \leq 1\). Given that the sensor \(c_1\) (the first row of \(C\)) generates high signal-to-noise ratio in the price of high energy expense. Therefore, it’s of interest to limit its use. The goal is to estimate, at each instant \(x(k)\) using the measurements \(y(k+2)\), \(y(k+1)\) and \(y(k)\). Table 3.1 summarizes the sensor choice obtained by using the proposed method, method in [55], adopting criterion introduced in [3] and a random choice. 100 estimation trials with random initial conditions corresponding to these strategies has been shown in Fig 3.1. As illustrated there, as the constraints on usage of sensor 1 tighten, the estimation error increases. In all cases, the proposed method can handle the constraints and give estimation with error remaining below the worst-case bound (dashed line), while the error corresponding to a random sensor selection exceeds it.
Figure 3.1: Comparison of Estimation Performance based on Strategies listed in Table 3.1.
Top (no constraints): Red squares (proposed method): Mean=-5.00e-3, Std=1.02e-1; Blue circles (random selection): Mean=6.18e-2, Std=3.86e-1; Green triangles ([55]): Mean=-8.38e-3, Std=9.43e-2.
Center ($c_1$ active at most twice): Red squares (proposed method): Mean=1.34e-2, Std=2.10e-1; Blue circles (random selection): Mean=2.60e-2, Std=3.68e-1.
Bottom ($c_1$ active at most once): Red squares (proposed method): Mean=4.43e-2, Std=2.28e-1; Blue circles (random selection): Mean=1.12e-1, Std=4.93e-1.
3.3. RESULTS AND EXPERIMENTS

3.3.2 Estimation with Sensor Selection Strategy in Sensing System

Consider now a sensing system of up to 9 measurements of a vector $x \in \mathbb{R}^3$ given by $y = Cx + \eta$ with $\|\eta\|_2 = 1$ and

$$C = \begin{bmatrix}
-4.0988 & 2.2718 & 2.1652 \\
-4.1893 & 2.3651 & -4.1625 \\
3.9169 & 3.8809 & 0.6934 \\
3.4704 & -3.7338 & -2.9529 \\
-4.5053 & 1.9238 & -2.0671 \\
2.4785 & 0.3474 & -1.7208 \\
4.2102 & 3.2847 & 0.2299 \\
-3.7732 & -4.0766 & 1.7296 \\
2.1043 & 4.8087 & 4.6235
\end{bmatrix}$$

In this case, the goal is to select no more than 3 sensors from all available to estimate a quantity $z = x_1 + x_2 + x_3$. The optimal choice according to max-det criterion is sensor #2, #4 and #8, while the proposed method selects #2, #3 and #9. 100 random trials have been experimented and estimation error corresponding to these selections has been shown in Fig 3.2. Observe that in some trials the max-det selection error (blue circles) indeed exceeds the optimal worst-case bound achieved by the proposed method (red squares). For comparison purposes, we also show the error achieved by a worst-case optimal estimator that uses all the sensors available (green triangles), illustrating the additional cost incurred when considering sparse structure on the sensing strategy.
Figure 3.2: Estimation Error for Different Sensor Selections.

Red squares (proposed method): Mean=-6.40e-3, Std=7.67e-2; Blue circles (max-det selection): Mean=1.07e-2, Std=1.12e-1; Green triangles (All sensors): Mean=-5.56e-3, Std=5.63e-2.
Chapter 4

$\ell_\infty$ Worst-case Optimal Estimators for Switched ARX Systems

4.1 Previous Works and Problem Statement

In previous chapter, we present an algorithm in which variables of \{0, 1\} are assigned to all candidate sensors to potentially parametrize all admissible choices. In this section, we extend the similar idea to show that an $\ell_\infty$ worst-case optimal estimator for switched autoregressive exogenous systems can be designed by using \{0, 1\} variables to parametrize all admissible switching paths.

Generally, switched linear systems contains two kinds of signals: the discrete state, also known as mode, denoting which subsystem is active at each time instant; the continuous state denoting the output (in either discrete-time or continuous-time case) of switched linear systems. Among the many efforts in the switched linear systems estimation problem, the existing results can be roughly classified into three categories.

The first class of approaches consider the situation that the mode is accessible, in which it can be solved by considering for instance a gain switched Luenberger observer[59]. This approach involves solving a set of linear matrix inequalities. In the case of $\ell_\infty$ bounded noise, an on-line linear programming procedure has been introduced for the worst-case optimal filters synthesis[60]. This approach involves $O(r)$ variables, where $r$ is the memory of the filter.

The second category assumes a relaxed knowledge of the mode signal, rather than the exact mode information. The problem is formulated in a probabilistic context, where either a probability vector is provided for different modes or a first-order Markov chain is used to model the mode
transitions[61, 62]. However, the probabilistic information of the mode signal is not generally accessible in many practical situations.

In the third category that neither direct access to the mode signal nor direct evaluation on the probability of transitions is available, an estimator is constructed to estimate the mode from the knowledge of the inputs and outputs[63, 64, 65, 66]. In [63, 66], a multi-model observer is designed with a location observer deciding the active subsystem and a continuous observer estimating the continuous state. These approaches require the switching system to be observable, which is not trivial to check. Alternative approaches circumvent this limit by avoiding the estimation of mode signal. An asymptotic observer[64] is constructed directly from the measured data using an algebraic approach, while the switching is restricted to the measurement equations with systems sharing common dynamics. In [65], both the mode and the continuous states are jointly estimated by minimizing a receding-horizon quadratic cost function that penalizes weighted $\ell_2$ norms of the estimation errors in the state, measurement and process noise over the set $\mathcal{P}$ of switching pattern compatible with the experimental data observed so far. A potential difficulty here stems from the computational complexity entailed in minimizing the objective function over all patterns in $\mathcal{P}$, and in propagating this set. In [67], a non-smooth optimization approach is introduced to estimate the continuous state without explicitly estimating the mode variables. The idea is to fit estimations with the observed input/output data to sparsify the residues. However, this scheme doesn’t consider the consistency of noise with the given support set.

Consider a multi-input, multi-output (MIMO) switched autoregressive exogenous (SARX) models of the form:

$$y_t = \sum_{k=1}^{n_a} A_k(\sigma_t) y_{t-k} + \sum_{k=1}^{n_c} C_k(\sigma_t) u_{t-k}$$

$$\tilde{y}_r = y_r + \eta_r, \quad r = t, t-1, \cdots, t-n_a$$

where $u_t \in \mathbb{R}^{n_u}, y_t, \tilde{y}_t \in \mathbb{R}^{n_y}$ and $\sigma_t \in \mathbb{N}_{n_s}$ denote the input, output, its noisy measurements, and the mode signal, respectively. No assumptions are made in terms of dwell time, thus the system can switch arbitrarily fast among the $n_s$ submodels, say $G_i$’s, each associated with a set of its coefficient matrices $\{A_1(i), \cdots, A_{n_a}(i), C_1(i), \cdots, C_{n_c}(i)\}$. The goal is to estimate a (scalar) linear combination of values of $y$ using the most recent $r$ noisy measurements $\tilde{y}$, where $r$ is a design parameter.

**Problem 2.** Given a nominal switched ARX system of the form (4.1), an a-priori bound $\epsilon$ on the $\ell_\infty$
4.2 Proposed Solution

In this section, a convex optimization based algorithm is proposed for finding the bounded complexity \( \ell_\infty \) (point wise) optimal estimators. The main idea is to first recast this problem into a linear optimization form over a semi-algebraic set which defines all admissible continuous state, then relax this polynomial optimization problem into a sequence of convex optimization problems by exploiting the moments-based method.

4.2.1 A Semi-Algebraic Optimization Reformulation and Convex Relaxation

Given a sequence of input/output data \( \{\tilde{y}_t, u_t\} \), \( t' = t - r + 1, \cdots, t \), define the consistency set \( T(\tilde{y}, u) \) as a set of all possible values of \( y \) that are consistent with the observed data,

\[
T(\tilde{y}, u) = \{ \{y_{t'}\}_{t'-r+1}^t \mid (4.1) \text{ holds for some } \|\eta\|_\infty \leq \epsilon \text{ and } \{\sigma_{t'}\}_{t'-r+1+n_a}^{t-r+1+n_a} \in \mathbb{N}_{n_a} \} \tag{4.3}
\]

Next, we show that \( T(\tilde{y}, u) \) has a semi-algebraic representation by assigning \( \{0, 1\} \) variables to each potential mode candidate, therefore parametrizing all admissible mode signal in this case. Clearly, (4.1) holds if and only if the following set of equations is feasible:

\[
s_{\sigma, \tau} \|y_{t'} - \sum_{k=1}^{n_a} A_k(\sigma) y_{t'-k} - \sum_{k=1}^{n_c} C_k(\sigma) u_{t'-k}\|_2^2 = 0 \\
s_{\sigma, \tau} = \frac{s_{\sigma, \tau}^2}{s_{\sigma, \tau}} \\
\sum_{\sigma} s_{\sigma, \tau} = 1
\tag{4.4}
\]
where $\sigma = 1, \cdots, n_s$ and $\tau = t - r + 1 + n_a, \cdots, t$. Also, notice that we use $s_{\sigma, \tau} = s_{\sigma, \tau}^2$ to force $s_{\sigma, \tau} \in \{0, 1\}$. Hence, it follows that $T(\tilde{y}, u)$ can be written as the below equivalent form,

$$T'(\tilde{y}, u) = \{\{y_{\tau'}\}_{t-r+1}^t | (4.4) \text{ holds}, \|\eta_{\tau'}\| \leq \epsilon, y_{\tau'} = \tilde{y}_{\tau'} - \eta_{\tau'}, \text{ for all } \tau' = t - r + 1, \cdots, t\} \quad (4.5)$$

Next, we present the proposed estimator based on solving two optimization problems. Given $\{\tilde{y}_{\tau'}, u_{\tau'}\}_{t-r+1}^t$, define $\hat{z}_t^+$ and $\hat{z}_t^-$ as the solutions to the following optimization problems:

$$\hat{z}_t^+ = \max_{y \in T'(\tilde{y}, u)} \text{Trace}(H^TY_t) \quad (4.6)$$

$$\hat{z}_t^- = \min_{y \in T'(\tilde{y}, u)} \text{Trace}(H^TY_t) \quad (4.7)$$

and define the (central) estimator

$$\hat{z}_t = \frac{\hat{z}_t^+ + \hat{z}_t^-}{2} \quad (4.8)$$

**Lemma 2.** $\hat{z}_t$ is a point wise worst-case optimal estimator of $z_t$.

**Proof.** This result is followed immediately from Theorem 2.4 in [68] by noting that $\hat{z}_t$ is the Chebyshev center of the set

$$\mathcal{H}T' = \{z \mid z = \text{Trace}(H^TY) \text{ for some } \{y\}_{t-r+1}^t \in T'(\tilde{y}, u)\}$$

□

**Remark 2.** By directly substituting $y_{\tau} = \tilde{y}_{\tau} - \eta_{\tau}$ into (4.4) it follows that the optimization problems (4.6) and (4.7) can be rewritten as:

$$\hat{z}_t^+ (\hat{z}_t^-) = \max_{\eta} (\min_{\eta}) \text{Trace}(H^T\Xi) + f_0(\tilde{y})$$

\[ s.t. \quad s_{\sigma, \tau} p_{\sigma, \tau}(\eta) = 0 \]

\[ \|\eta\|_{\infty} \leq \epsilon \]

\[ s_{\sigma, \tau} = s_{\sigma, \tau}^2 \]

\[ \sum_{\sigma} s_{\sigma, \tau} = 1 \]

where $\Xi \triangleq [\eta_t \cdots \eta_{t-n_s}]$, $p_{\sigma, \tau}(\eta)$ is a (quadratic) SOS polynomial, and where $f_0(\tilde{y})$ is a function of the measured data only.

Notice that problem (4.9) is computationally challenging due to the non-convexity of polynomial constraints. However, we can use moments-based method to relax this problem into a sequence of convex LMI problems generating monotonic sequence of bounds from which convergent
4.2. PROPOSED SOLUTION

Suboptimal estimators can be obtained. Consider \( p_{\sigma,\tau} \) in (4.9) of form \( p_{\sigma,\tau} = c_T^T \eta^\alpha \), where the coefficients \( c_\tau \) are functions of the parameters of the SARX model and the measured data \( \{\tilde{y}, u\} \). \( \eta^\alpha \) is a vector containing monomials of unknown noise \( \eta \) in a suitable ordering. From the results in section 2.1, we result in a sequence of LMI problems,

\[
\hat{z}_{t,N}^+ (\hat{z}_{t,N}^-) = \max_m (\min_m) \; \text{Trace}(H^T P_N M_N) + f_o(\tilde{y})
\]

s.t.

\[
M_N(m) \succeq 0
\]

\[
L_N(m) \succeq 0
\]

where \( m \) denotes the moments of \( \{\eta^\tau\}_{t-r+1}^t \) up to order \( 2N \), \( P_N \) denotes a selection matrix selecting relevant moment variables from moment matrix \( M_N \), \( L_N \) is localizing matrix corresponding to semi-algebraic constraints.

**Lemma 3.** The estimator \( \hat{z}_{t,N} \) satisfies \( \max_{z \in \mathcal{H}^T'} |z - \hat{z}_{t,N}| \leq e_{t,N} \). Moreover, \( e_{t,N} \) is a non-increasing sequence with \( \lim_{N \to \infty} e_{t,N} = e_t \), the optimal worst-case error.

**Proof.** From the results in section 2.1, it can be easily shown that \( \hat{z}_{t,N}^+ \geq \hat{z}_{t,N} \) and \( \hat{z}_{t,N}^- \leq \hat{z}_{t,N} \). Therefore, given any \( z \in \mathcal{H}^T' \), we have

\[
z - \hat{z}_{t,N} \leq \hat{z}_{t,N}^+ - \hat{z}_{t,N} = \frac{\hat{z}_{t,N}^+ - \hat{z}_{t,N}^-}{2} \]

\[
z - \hat{z}_{t,N} \geq \hat{z}_{t,N}^- - \hat{z}_{t,N} = -\frac{\hat{z}_{t,N}^+ - \hat{z}_{t,N}^-}{2}
\]

Hence, \( |z - \hat{z}_{t,N}| \leq e_{t,N} \) for all \( z \in \mathcal{H}^T' \).

The facts that \( e_{t,N} \) is non-increasing and convergent follows from the results in section 2.1, showing that \( \hat{z}_{t,N}^+, \hat{z}_{t,N}^- \) are convergent non-increasing/non-decreasing sequences respectively. \( \square \)

**Theorem 4.** The optimal value in problem (4.10) is achieved for some \( N \leq N_o = r - n_\alpha + 1 \).

**Proof.** The proof is based upon showing that the objective function in (4.9) admits an expansion of the form

\[
\text{Trace}(H^T \Xi) - z^* = u_o + \sum u_i f_i
\]

where \( z^* \) denotes the optimal value, \( f_i \) denote the constraints, and \( u_o, u_i \) are sum of squares (SOS) polynomials such that degree of \( u_o \leq 2N_o \) and degree of \( u_i f_i \leq 2N_o \). The desired result follows then from Putinar’s Positivstellensatz. \( \square \)
4.2.2 Computational Complexity Considerations

In this section, we consider reducing the computational complexity from two aspects. Firstly, we introduce a sparse structure in moments-based method, known as running intersection property.

Definition 5. [69] Assume that the polynomial \( p \) can be partitioned into \( p = p_1 + \cdots + p_l \), such that each \( p_i \) and the constraints \( g_k \) that define the semi-algebraic set \( K \) contain only variables indexed by elements of some subset \( I_i \subset \{1, \cdots, n\} \). If there exists a reordering \( I_{i'} \) of \( I_i \) such that for every \( i' = 1, \cdots, l-1 \),

\[
I_{i'+1} \cap \bigcup_{j=1}^{i'} I_j \subset I_q, \text{ for some } q \leq i'
\]  

(4.11)

then the running intersection property is satisfied.

For problems satisfying the running intersection property, it can be shown that it’s possible to construct a hierarchy of semi-definite programs of smaller size. Specially, partition the objective function \( \{p_i\}_{i=1}^l \) according to the set \( \{I_i\} \) and consider the problem

\[
p_N^* = \min_m \sum_{i=1}^l \sum_{\alpha_{I_i}} p_{\alpha_{I_i}} m_{\alpha_{I_i}}
\]

s.t. \( M_N(m_{I_i}) \succeq 0, i = 1, \cdots, l \)  
\[
L_N(g_k m_{I_i}) \succeq 0, i = 1, \cdots, l
\]

where \( \alpha_{I_i}(j) = 0, \forall j \notin I_i \). Then as shown in [69], the convergence \( p_N^* \uparrow p^* \) still holds. It’s worth emphasizing that this property renders a substantial reduction in computational complexity in practice. For instance, in the case of generic polynomials and constraints, an \( N^{th} \) order relaxation requires considering moment and localizing matrices containing \( O(n^{2N}) \) variables. On the other hand, if the running intersection property holds and it is possible to define \( l \) sets of smaller sized matrices each containing variables only in \( I_i \) (i.e. number of variables is \( O(\kappa^{2N}) \), where \( \kappa \) is the maximum cardinality of \( I_i \)). In many practical application, including the one we proposed here, \( \kappa \ll n \). Hence, exploiting the sparse structure substantially reduces the number of variables in the optimization and hence computational complexity, while still providing convergent relaxations.

It can be easily seen that, in problem (4.10), the objective and constraints can be partitioned into \( r - n_a \) subsets \( I_i \), each containing the noise variables for \( n_a + 1 \) consecutive time instants and the \( n_s \) indicator (\( \{0, 1\} \)) variables associated with the index \( i \). Thus, the running intersection property is
4.2. PROPOSED SOLUTION

satisfied and it can be solved by considering \( n_a \) smaller moment matrices of size \( (N+n_a+n_y+n_s) \) rather than a single one of size \( (N+n_a+n_y+n_s) \). While this results in a substantial reduction on the number of variables involved (from \( O((r_n+n_a+n_y+n_s)^2N) \) to \( O((n_a+n_y+n_s+n_s)^2N) \)), the computational burden can limit the “memory” of the estimator (the design parameter \( r \)) that can be used in practice. To circumvent this difficulty, in the sequel we present a suboptimal filter that allows for using a shorter memory \( r_0 < r \) by propagating past bounds on the estimation error in a receding horizon fashion, leading to Algorithm 3.

**Algorithm 3** Receding Horizon Estimator

Choose the window length \( r_0, r_0 < r \). Set the number of iterations to \( r - r_0 + 1 \).

Set \( z^+_r = +\infty, z^-_r = -\infty, \tau = 1, \cdots, r \).

For \( i = r_0, \cdots, r \), repeat

1- Solve problem (4.10) with \( r = r_0 \) subject to the additional constraints:

\[ \text{Trace}(H^T Y_{\tau-n_z:}\tau) \leq z^+_\tau, \]
\[ \text{Trace}(H^T Y_{\tau-n_z:}\tau) \geq z^-_\tau, \quad \tau = i - r_0 + 1, \cdots, i \]

2- Update \( z^+_i \) and \( z^-_i \) with the results from [1-].

end for.

4.2.3 Consistency Considerations

Note that the optimization problems (4.6) and (4.7) are uncoupled, in the sense that the worst-case noise and switching sequence could potentially be different. Hence the estimator (4.8) is not interpolatory, in the sense that it does not necessarily belong to the consistency set \( T^\tau(\tilde{y}, u) \). This raises the issue of whether the search for optimal estimators should be restricted to those inside the consistency set, for instance by enforcing the constraint that the optimizing sequences in (4.6) and (4.7) are the same. As we show next with a simple example, as long as one is interested in minimizing the worst case estimation error, the answer to this question is negative: estimators outside the consistency set can have smaller worst-case error than those inside it\(^1\). To this effect consider a system that switches between two first order models:

\[ y_{k+1} = y_k \quad \text{(sys1)} \]

\(^1\)This is related to the fact that, unless one is working in a Hilbert space, the Chebyshev center of a set can be outside it, even if the set is convex.
with noise bound $|\epsilon| \leq 1$. Assume that two noisy measurements are available

$$
\tilde{y}_1 = y_1 + \epsilon_1 = 2 \\
\tilde{y}_2 = y_2 + \epsilon_2 = 0
$$

The goal is to find a worst-case optimal estimator for $y_2$. Optimizing $y^+ - y^-$ (the diameter of information) subject to the constraint that both $y^+$ and $y^-$ have to be generated by the same switching sequence leads to $y_{sys1}^+ = y_{sys1}^- = 1$ for the case where the measurements are both generated by the first system and $y_{sys2}^+ = -0.5$, $y_{sys2}^- = -1$ if both are generated by the second one. The corresponding central estimators are given by $y_{c,sys1} = 1$ and $y_{c,sys2} = -0.75$. Following these estimation, the worst-case error of $y_{c,sys1}$ can be $1 - (-1) = 2$, achieved if the true signal was generated by the system 2. Similarly, the worst-case estimation error of $y_{c,sys2}$ is $1 - (-0.75) = 1.75$, achieved if the true signal was generated by system 1. On the other hand, optimizing $y^+$ and $y^-$ separately leads to $y^+ = 1$, $y^- = -1$ and $y_c = 0$ with a worst-case error of 1, regardless of whether the true system is the first or the second one. Note that the estimation $y_c = 0$ is outside the consistency set, which is $\{1\} \cup [-1.5, -0.5]$.

### 4.2.4 Handling Parametric Uncertainty

In this section, we briefly outline how to extend the proposed estimator to account for parametric uncertainty. Assume that some coefficients in (4.1) are only known to belong to some semi-algebraic set, e.g. $A_i \in A_i$, $C_j \in C_j$, for some $i, j$. In this case, treating $A_i$, $C_j$ as unknowns in (4.4) and adding the constraints $A_i \in A_i$, $C_j \in C_j$ to the description (4.5) leads to a semi-algebraic optimization problem that can be solved using the techniques outlined above.

### 4.3 Handling Specific Switching Rule

In this section, we extend the proposed estimator to handling a-priori constraints on switching rule. In practice, although the exact and probabilistic information of the mode is usually not available, there are cases that we can have information about the switching rule, such as subsystem 1 can never switch to subsystem 2. Note that this information doesn’t tell us exact or probabilistic clue where the mode will be. But this information can be used in proposed estimator to narrow down

\footnote{simple examples are structured additive uncertainty $M + \Delta$ and multiplicative uncertainty $(1 + \delta)M$.}
the admissible switching signal, hence the consistency set. Therefore, it will make the proposed estimator more practical with the ability to handle this switching rule. We show that the proposed approach can easily handle this type constraints by introducing additional monomial equality as

$$s_{1,\tau}s_{2,\tau+1} = 0$$

(4.14)

Notice that, by adding this constraint, the running intersection property doesn’t hold with $I_i$ containing only variables involved at each time instant, but two connected time instants.

### 4.4 Results and Experiments

In this section, we present several numerical examples to illustrate the effectiveness of the propose estimator.

#### 4.4.1 Example without Model Uncertainty

Consider first the case of a switched ARX system that switches arbitrarily fast between the following two subsystems

$$y_t = 0.2y_{t-1} + 0.24y_{t-2} + 2u_{t-1} \quad (G1)$$

$$y_t = -1.4y_{t-1} - 0.53y_{t-2} + u_{t-1} \quad (G2)$$

The goal is to estimate $z_t = y_t$ from the noisy measurements

$$\tilde{y}_t = y_t + \eta, \quad ||\eta||_{\infty} = 0.5$$

Fig 4.1 and Fig 4.2 shows the performance of the proposed estimator with memory $r_0 = 11$, over a time horizon $T = 30$. As illustrated there, the proposed estimator substantially reduces the uncertainty in the estimation.
4.4.2 Example with Parametric Uncertainty

In this example, we show the ability of the proposed estimators to handle parametric uncertainty. We assume that the coefficients of the nominal model are subject to time-varying
multiplicative parametric uncertainty leading to descriptions of the form

\[ y_t = \sum_{k=1}^{n_a} (1 + \delta_{A_{k,t}}) A_k(\sigma_t) y_{t-k} + \sum_{k=1}^{n_c} (1 + \delta_{C_{k,t}}) C_k(\sigma_t) u_{t-k} \]

\[ \bar{y}_\tau = y_\tau + \eta_{\tau}, \quad \tau = \ell, \cdots, t - n_a \]

where the only information available about \( \delta_{A_{k,t}} \) and \( \delta_{C_{k,t}} \) is a bound of the form

\[ \|\delta_{A_{k,t}}\|_\infty \leq \epsilon_{A_k} \text{ and } \|\delta_{C_{k,t}}\|_\infty \leq \epsilon_{C_k} \]

This uncertainty can be handled by proposed algorithm by adding new variables \( \delta_{A_{k,t}} \) and \( \delta_{C_{k,t}} \) to the problem, and suitably modifying the moments sequence \( m \) and corresponding matrix \( M(m) \) to include these new terms, and adding localizing matrices \( L(m) \) corresponding to the constraints (4.16).

Fig 4.3 and Fig 4.4 shows the results of two experiments with 5% and 10% uncertainty, respectively (e.g. \( \epsilon_{A_k} = \epsilon_{C_k} = 0.05 \) and \( \epsilon_{A_k} = \epsilon_{C_k} = 0.10 \)). Finally, we show comparison of estimation error for both cases as well as the one obtained in determined model case in Fig 4.5. As expected, model uncertainty results in large estimation errors.
4.4.3 Example with Switching Rule

In this example, we consider 3 subsystems showing as below, with switching rule that subsystem #1 cannot to subsystem #3, subsystem #2 cannot to subsystem #1, subsystem #3 cannot to...
4.4. RESULTS AND EXPERIMENTS

(a) Comparison of \((\hat{z}^+ - \hat{z}^-)/2\) with switching rule v.s. without switching rule

(b) Comparison of the estimation error with switching rule v.s. without switching rule

Figure 4.6: Performance comparison between estimation with switching rule and without switching rule

subsystem #2.

\[
\begin{align*}
\#1 : & \quad y_t = 0.2y_{t-1} + 0.24y_{t-2} + 2u_t \\
\#2 : & \quad y_t = -1.4y_{t-1} - 0.53y_{t-2} + u_t \\
\#3 : & \quad y_t = 1.7y_{t-1} - 0.72y_{t-2} + 0.5u_t \\
\end{align*}
\]

\[\hat{y}_t = y_t + \eta_t\]

The system output of length \(T \approx 60\) is corrupted by noise \(\|\eta\|_\infty = 0.354\), leading to the Signal-to-Noise ratio of 16.8\(dB\). We implement the proposed filter with and without using the prior information of switching rule. The worst-case optimal filters are implemented using the same parameter setting: memory \(r_0 = 10\), and moment matrix order equals \(N = 3\).

From Figure 4.6a we can tell that, even under the same parameter setting (same estimator memory and moment matrix order), adopting prior information of switch rule helps decreasing the uncertainty of the feasible range, which is the worst-case estimation error. Therefore, by handling the prior information of switching rule, the worst-case optimality of proposed filter can be further improved. It’s worth mentioning that decreasing the worst-case estimation error is not guaranteed to decrease the estimation error of each time instance. However, as shown in Figure 4.6b, we can see that the estimation error is indeed improved with high probability.
Chapter 5

Sparse Structured Output Feedback
Controller Design

5.1 Previous Works

In this chapter, we consider the controller design under sparse information structure, which is known as decentralized control. Unlike the centralized control where the sensors and actuators are fully connected, the design of decentralized controller is generally known to be difficult and even NP-hard[13, 14]. This prompted a large research effort devoted to identifying conditions that lead to tractable problems and to develop relaxations for cases where these conditions do not hold.

Regarding scenarios, leading to convex problems, decentralized control with a temporal delayed information structure was considered in [70, 71]. Special plant and controller structures that lead to a convex problem when combined with the Youla-Kucera parameterization were discussed in [72]. A convex semidefinite approach to synthesizing distributed rather than decentralized controller was proposed in [73] for structured plant. [74, 75, 76] showed that a necessary condition for the existence of a convex parameterization of sparse structured controllers is the so called Quadratic Invariance property, and analyzed conditions under which this property indeed guarantees the existence of a sparse controller. Along these lines, [76] fully characterized stabilization for all QI problems and [77] proposed a regularized design method based upon atomic norm minimization. Lampersky and Lessard [78] considered the case of systems having partially nested information constraints and proposed a dynamic programming based solution. Shah and Parrilo[79], addressed the case of poset-causal systems.
For cases where the conditions above do not hold, non-convex optimization techniques have been used to handle the structural constraints [80, 81]. In these approaches, the sparsity constraints are incorporated directly into the problem formulation, and the resulting (non-convex) problem is solved using an Augmented Lagrangian type first order method. However, since the problem is non-convex, there is no guarantee of global optimality of the resulting solutions. Alternatively, suboptimal solutions (in the sense that are based only on sufficient conditions) have been proposed in [82, 83, 84], where the desired sparse structure is achieved by imposing a suitable structure on the design variables, typically matrix factors. However, these approaches are restricted in the type of information structures that can be imposed, and some require additional structure on the open loop plant, such as a full row rank output matrix. Further, since these methods rely on sufficient conditions, they may fail to find a solution, even if a desirable controller exists.

Finally, rather than directly attempt to solve the non-convex optimization problem, several approaches seek to solve the convex relaxations, in some cases with optimality certificates. [85] initially considered a finite-horizon scenario and cast decentralized control as a rank-constrained semi-definite programming problem. While these results have been extended to the infinite horizon case in [86], the computational complexity of this relaxation is high, due to the semi-definite constraints arising from the use of quadratic Lyapunov functions. Finally, [87] introduced a convex surrogate of the original non-convex problem, also for finite horizon cases.

In the following part of this chapter, we propose two alternative solutions to solve the sparse structured controller design problem under scenario of static and dynamic control, respectively. The effectiveness of the proposed methods is illustrated using numerical examples.

5.2 Sparse Structured Static Output Feedback Controller Design

In the first half of this chapter, we introduce a polynomial optimization approach to sparsity-constrained controller design, which falls into a rank minimization problem with efficient convex optimization methods applicable. Consider a discrete-time LTI system

\[
 x_{k+1} = Ax_k + Bu_k \\
 y_k = Cx_k
\]

(5.1)

the problem can be stated as:
Problem 3. Given system (5.1) and a sparse pattern, design a static output-feedback controller $F$ such that

$$u_k = F y_k, \quad F \in S$$

(5.2)

stabilizing the given plant. Here $S$ represents a subspace of real matrices following the given sparse pattern.

5.3 Proposed Solution

Let’s first consider the stability of a simple linear system

$$x_{k+1} = Ax_k$$

(5.3)

Definition 6. [88] Let $P$ denotes a full column rank matrix. The function

$$V(x) = \|Px\|_\infty$$

(5.4)

is said to be polyhedral Lyapunov function for (5.3) if

$$V(x_{k+1}) - V(x_k) < 0$$

(5.5)

The advantage of adopting polyhedral Lyapunov functions is that, as shown in [88], this class of Lyapunov functions is universal in the sense that the origin is an asymptotically stable equilibrium point of (5.3) if and only if there exists such a matrix $P$ satisfying (5.4) and (5.5).

Lemma 4. A matrix $A \in \mathbb{R}^{n \times n}$ is said to be Hurwitz if and only if there exists a full column rank matrix $P \in \mathbb{R}^{p \times n}$ and a matrix $H \in \mathbb{R}^{p \times p}$ with $\|H\|_\infty < 1$ such that

$$PA = HP$$

(5.6)

Proof. From [88] we have that $A$ is Hurwitz if and only if there exists a matrix $P$ of full column rank such that, for all $x$

$$\|PAx\|_\infty - \|Px\|_\infty < 0$$

(5.7)

thus, the set $\{x \mid \|Px\|_\infty \leq 1\}$ is positive invariant (in fact contractive), for the system (5.1). The proof follows now from a slight extension of Proposition 2 in [89].

Corollary 1. A static output-feedback controller $F$ is stabilizing if and only if, there exists a positive scalar $\sigma < 1$, a matrix $P$ of full column rank and a matrix $H$ such that

$$P(A + BFC) = HP, \quad \|H\|_\infty \leq \sigma$$

(5.8)
Therefore, Problem 3 is equivalent to finding matrices $F$, $P$ and $H$ satisfying the below conditions

\[
P(A + BFC) = HP
\]

\[
\|H\|_\infty \leq \sigma, \text{ for } 0 < \sigma < 1
\]

$P$ is full column rank

\[
F \in S
\]

Note that the full column rank constraint is difficult to deal with due to the apparent non-convexity. Next, we present a Theorem leading to reformulation and avoiding the rank constraint in $P$.

**Theorem 5.** Given a matrix $A$ and a positive scalar $\sigma < 1$, such that there is a matrix $P$ of full column rank satisfying $PA = HP$ with $\|H\|_\infty \leq \sigma$. There always exists $P' = \begin{bmatrix} I \\ P^* \end{bmatrix}$, such that $P'A = H'P'$ with $\|H'\|_\infty \leq \sigma$.

**Proof.** Let

\[
H' = \begin{bmatrix} H_1 & H_2 \\ 0 & H \end{bmatrix}
\]

(5.10)

with this partition, we have

\[
P'A = \begin{bmatrix} A \\ P^*A \end{bmatrix}
\]

\[
H'P' = \begin{bmatrix} H_1 & H_2 \\ 0 & H \end{bmatrix} \begin{bmatrix} I \\ P^* \end{bmatrix}
\]

(5.11)

Therefore, satisfying $P'A = H'P'$ with $\|H'\|_\infty \leq \sigma$ is equivalent to satisfy

\[
A = \begin{bmatrix} H_1 & H_2 \end{bmatrix} \begin{bmatrix} I \\ P^* \end{bmatrix}, \text{ with } \| \begin{bmatrix} H_1 & H_2 \end{bmatrix} \|_\infty \leq \sigma
\]

(5.12)

\[
P^*A = HP^*, \text{ with } \|H\|_\infty \leq \sigma
\]

(5.13)

To show that (5.12) is feasible, choose $H_1 = \alpha A$ for some $\alpha$ such that $\|H_1\|_\infty \leq \gamma \leq \sigma$, and $H_2$ satisfying

\[
H_2P = A - H_1 = (1 - \alpha)A
\]
Since by hypothesis $P$ has full column rank, therefore we always can find a $\tilde{H}_2$ satisfying above equality. Let’s say $\|\tilde{H}_2\|_\infty = \beta$, then we define $H_2 = \frac{\sigma - \gamma}{\sigma - \gamma} \tilde{H}_2$, then

$$(1 - \alpha) A = \tilde{H}_2 P$$

$$= H_2 \frac{\beta}{\sigma - \gamma} P$$

Let $P^* = \frac{\beta}{\sigma - \gamma} P$, we find

$$A = \begin{bmatrix} H_1 & H_2 \\ I & P^* \end{bmatrix}$$

with $\|H_1\|_\infty \leq \gamma$ and $\|H_2\|_\infty = (\sigma - \gamma)$. Hence, $\left\| \begin{bmatrix} H_1 & H_2 \end{bmatrix} \right\|_\infty \leq \sigma$. To complete the proof, note that (5.13) follows immediately from the hypothesis, by simply rescaling both sides.

Using Theorem 5, we can rewrite (5.9) into constraints with bilinear formation

$$\begin{bmatrix} I \\ P \end{bmatrix} (A + BFC) = H \begin{bmatrix} I \\ P \end{bmatrix}$$

$$\left\| H \right\|_\infty \leq \sigma$$

$$(F)_{r,c} = 0, \ \forall (r,c) \in C$$

where $0 < \sigma < 1$. The bilinear constraint is of polynomial form, therefore it can be handled in moments-based method. In order to involve moments-based method, we have to introduce additional variables to form the equivalent polynomial form of constraint $\|H\|_\infty \leq \sigma$,

$$H = H^+ - H^-$$

$$(H^+)_{i,j} \geq 0, \ \forall (i,j)$$

$$(H^-)_{i,j} \geq 0, \ \forall (i,j)$$

$$(H^+)_{i,j} \times (H^-)_{i,j} = 0 \ \forall (i,j)$$

$$\sum_j (H^+)^j + (H^-)^j \leq \sigma$$

Substitute (5.15) into $\|H\|_\infty \leq \sigma$ of (5.14), we result in constraints of polynomial form. Therefore, stabilizing controllers with the specified sparse pattern can be found by solving the
5.3. PROPOSED SOLUTION

following (semi-algebraic) feasibility problem:

\[
\begin{bmatrix}
I \\
P
\end{bmatrix} (A + BFC) = H \begin{bmatrix}
I \\
P
\end{bmatrix}
\]

\[H = H^+ - H^-\]

\[(H^+)_{i,j} \geq 0, \forall (i,j)\]

\[(H^-)_{i,j} \geq 0, \forall (i,j)\]

\[(H^+)_{i,j} \times (H^-)_{i,j} = 0, \forall (i,j)\]

\[\sum_j [(H^+)_{ij} + (H^-)_{ij}] \leq \sigma < 1\]

\[(F)_{r,c} = 0, \forall (r,c) \in C\]

or, equivalently, the optimization problem

\[
\min \sigma
\]

\[
\text{s.t.} \quad \begin{bmatrix}
I \\
P
\end{bmatrix} (A + BFC) = H \begin{bmatrix}
I \\
P
\end{bmatrix}
\]

\[H = H^+ - H^-\]

\[(H^+)_{i,j} \geq 0, \forall (i,j)\]

\[(H^-)_{i,j} \geq 0, \forall (i,j)\]

\[(H^+)_{i,j} \times (H^-)_{i,j} = 0, \forall (i,j)\]

\[\sum_j [(H^+)_{ij} + (H^-)_{ij}] \leq \sigma\]

\[(F)_{r,c} = 0, \forall (r,c) \in C\]

From Theorem 5, we can tell that the existence of solution satisfying \(\sigma < 1\) is a necessary and sufficient condition for the existence of a stabilizing output feedback controller following the given sparse pattern. Moreover, from the results introduced in section 2.1 it follows that, for a fixed \(p \geq 1\), suitable matrices \(P \in \mathbb{R}^{p \times n}, H_+, H_- \in \mathbb{R}^{(p+n) \times (p+n)}\) and \(F\) can be found (or shown not exist) by solving a sequence of convex problems in the variables \(m\) of the form

\[
\min m_{1,\sigma}
\]

\[
\text{s.t.} \quad \begin{bmatrix}
M_T(m) \\
L_T(g_k(\cdot) \geq 0) \\
L_T(h_l(\cdot) = 0)
\end{bmatrix} \geq 0
\]

\[(5.18)\]

where \(m_{1,\sigma}\) denotes the expected value of \(\sigma\), \(M_T(m)\) represents the truncated moment matrix, containing terms of order up to \(2T\) associated with the sequence \(\{m\}\). \(L_T(g_k(\cdot) \geq 0)\) and \(L_T(h_l(\cdot) = 0)\)
denote the truncated localizing matrices corresponding to the inequality and equality constraints in (5.17), respectively. Also, the objective function value converges to $\sigma_{opt}$ from below as $T \to \infty$. Hence, if for a finite value of $T$, the objective function value is not less than 1, then (5.16) is infeasible and a larger value of $p$ should be chosen.

**Remark 3.** It has been shown in [89] that, the necessary and sufficient condition for a system in (5.3) to have a polyhedral Lyapunov function of the form (5.4) with $P \in \mathbb{R}^{p \times n}$ is that all eigenvalues of the matrix $A$ must be contained in the (open) diamond $|Re(\lambda) + Im(\lambda)| < 1$. Thus, it follows that if a sparse static gain that places all closed loop poles in this region exists, then only matrices $P \in \mathbb{R}^{n \times n}$ need to be considered when solving (5.17).

### 5.3.1 Computational Complexity Considerations and Low Order Relaxations

In this section, we are going to present a low order relaxation algorithm, giving consideration to computational complexity.

Note that a salient feature of formation (5.16) and (5.17) is that, the polynomials only involve monomials of degree up to 2. Therefore, moment matrix with order 1 will contains all variables appearing in the problems. It’s worth mentioning that this is the main motivation in adopting polyhedral Lyapunov functions instead of quadratic Lyapunov function. Consider a quadratic Lyapunov function of form $V(x) = x^TPx$, which requires $P \succ 0$, the equivalent polynomial constraints will be setting determinant of all leading minors to be positive. This will results in polynomials of degree up to the dimension of $x$ (say $n$), and further on moment matrix of order up to $n/2$. As the number of moments involved in relaxation of order $T$ with $m$ variables is $\binom{T+m}{m}$, it is obvious that using polyhedral Lyapunov functions will lead to substantial complexity reduction.

However, there is no guarantee that order 1 relaxation will be exact. Giving consideration of this question, by exploiting the fact that moment matrix associated with atomic measures having a single atom has rank 1, we finally result in the below constraints that enforcing rank 1 solution based on the moments-based feasible set

$$
\begin{align*}
M_1(m) &\succeq 0 \\
L_1(g_k(\cdot)) &\succeq 0 \\
L_1(h_l(\cdot)) &= 0 \\
rank[M_1(m)] &= 1
\end{align*}
$$

(5.19)

Since the above constraints involve rank constraint, it is non-convex. However, by iteratively minimizing re-weighted nuclear norm of $M_1$ (discuss of convergence property can be
found in [90]), a convex relaxation that seeks to obtain low rank solution, we can achieve a convex optimization procedure, as shown in Algorithm 4.

**Algorithm 4** Design Stabilizing Gain given Specific Sparse Pattern

```
Initialize: \( \text{iter} = 0, W^{(0)} = I \)

repeat
    Solve
        \[
        \min_m \text{Trace}(W^{(\text{iter})} M_1) \quad \text{s.t.} \quad M_1(m) \succeq 0 \\
        L_1(g_k(\cdot) \succeq 0) \succeq 0 \\
        L_1(h_l(\cdot) = 0) = 0
        \]  
        (5.20)

    Update
        \[
        W^{(\text{iter}+1)} = \left( M_1^{(\text{iter})} + \sigma_2(M_1^{(\text{iter})}) \right)^{-1} \\
        \text{iter} = \text{iter} + 1
        \]

until rank\( [M_1] = 1 \).
```

5.3.2 **Extension to Unknown Sparse Pattern**

In this section, we consider the situation that exact sparse pattern is unknown. In this case, we present a group lasso type scheme with the aim at finding a static output-feedback controller with relatively sparse pattern.

From (5.19), we observe that, for the equality constraints, say \( h_l(\cdot) = 0 \), one way of doing this in moments-based method is forming the localizing matrix for \( h_l(\cdot) \) and instead of letting it be positive semi-definite, making it equal to 0. Simply speaking, one equality constraint corresponding to a vector of moments equals to zero. Therefore, we form the vectors of moments corresponding to localizing matrix of elements in \( F \), and minimize them using group lasso during similar procedure as Algorithm 4. We result in our maximal sparse output-feedback controller design algorithm, shown in Algorithm 5.
**Algorithm 5** Design Stabilizing Gain with Maximal Sparse Pattern

**Initialize:** \( \text{iter} = 0, W^{(0)} = I, W_F^{(0)} = 1 \)

repeat

Solve

\[
\min_m \text{Trace}(W^{(\text{iter})} M_1) + \lambda \sum_{i,j} (W_F^{(\text{iter})})_{i,j} \| \text{vect}[L_1(F(i,j))] \|_2
\]

s.t. \( M_1(m) \succeq 0 \)
\( L_1(g_k(\cdot) \geq 0) \succeq 0 \)
\( L_1(h_l(\cdot) = 0) = 0 \) (5.21)

Update

\[
W^{(\text{iter}+1)} = \left( M_1^{(\text{iter})} + \sigma_2(M_1^{(\text{iter})}) \right)^{-1}
\]
\[
(W_F^{(\text{iter}+1)})_{i,j} = \left( \| \text{vect}[L_1(F(i,j))] \|_2 + \sigma_2(M_1^{(\text{iter})}) \right)^{-1}
\]

\( \text{iter} = \text{iter} + 1 \)

until \( \text{rank}[M_1] = 1 \).

### 5.3.3 Extension to Continuous-time Case

In order to generalize our algorithms into continuous-time case,

\[
\begin{align*}
    \dot{x}(t) &= Ax(t) + Bu(t) \\
    y(t) &= Cx(t)
\end{align*}
\]

we recall that the Euler Approximation System (EAS) of it is

\[
x(t+1) = \left[ I + \tau A \right]x(t) + \tau Bu(t)
\]

(5.23)

It has been shown that the stability of EAS implies the stability of the continuous-time system, and any positively invariant set of EAS is also positively invariant for it’s continuous-time counterpart[91]. The stability of (5.22) conversely implies the existence of some \( \tau > 0 \) such that its EAS is stable. Hence, given a continuous-time plant, the algorithms we proposed can be adopted according to the corresponding Euler approximation with a small enough \( \tau \).
5.4 Results and Experiments

In this section, we illustrate the proposed approach with several numerical examples, and compare to some existing approaches, which are known to be designing sparse controllers, static or dynamic. The goal of this section is to show that, our method is competent in more general case.

5.4.1 Design Sparse Gain with Non-Quadratic Invariant Pattern

5.4.1.1 Discrete-time Case

Consider a discrete-time output-feedback case and specific sparse pattern, which doesn’t satisfy the Quadratic Invariant Property under the plant.

\[
A = \begin{bmatrix}
0 & 1.0000 & 0 & 0 \\
0 & 0 & 1.0000 & 0 \\
0 & 0 & 0 & 1.0000 \\
0.0556 & -0.3477 & -1.8320 & -2.4300 \\
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
0.8026 & 0.9874 & 0.3856 \\
0.5635 & 0.4275 & -0.3341 \\
0.3888 & -0.7953 & 0.8435 \\
0.6525 & -0.1054 & 0.2650 \\
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
0 & 1.0000 & 0 & 0.8644 \\
0 & 0 & 1.0000 & -0.1482 \\
0 & 0 & 0 & -0.3990 \\
\end{bmatrix}
\]

\[
Pattern(S) = \begin{bmatrix}
1 & 1 & 0 \\
1 & 1 & 1 \\
0 & 1 & 1 \\
\end{bmatrix}
\]

using Algorithm 4, we have

\[
F = \begin{bmatrix}
-0.0954 & -0.8741 & 0 \\
-0.5670 & -0.1186 & -0.4140 \\
0 & 2.3228 & -3.1990 \\
\end{bmatrix} \in S
\]

(5.25)

with \(eig(A + BFC) = \{-0.5210, 0.0737, -0.1275 - 0.1116i, -0.1275 + 0.1116i\}\), and impulse response is shown in Fig 5.1.
It can be easily shown that the Quadratic Invariance property does not hold here, therefore a convex parametrization of all controllers does not exist. On the other hand, since that a block-diagonal quadratic Lyapunov function cannot be found and also the specific sparse pattern is not row or column sparse, both methods in [82, 84] fail to produce a solution here.
5.4. RESULTS AND EXPERIMENTS

5.4.1.2 Continuous-time Case

In this example, we consider a continuous-time state-feedback case given a sparse pattern that doesn’t satisfy the Quadratic Invariant Property under the plant. Consider a system

\[
A = \begin{bmatrix}
-0.0634 & -0.0976 & 0.1394 \\
-0.0726 & -0.2701 & -0.0427 \\
0.2266 & -0.0935 & -0.1666 \\
\end{bmatrix},
\]

\[
B = \begin{bmatrix}
0.6001 & -0.6363 & -0.7279 \\
-0.1372 & -0.4724 & 0.7386 \\
0.8213 & -0.7089 & 0.1594 \\
\end{bmatrix},
\]

\[
C = \begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\end{bmatrix},
\]

\[
Pattern(S) = \begin{bmatrix}
1 & 1 & 0 \\
1 & 1 & 1 \\
0 & 1 & 1 \\
\end{bmatrix}
\]

using Algorithm 4, we have

\[
F = \begin{bmatrix}
-1.8179 & 2.9661 & 0 \\
2.2409 & 3.7234 & 4.4081 \\
0 & -2.5720 & -1.7594 \\
\end{bmatrix} \in \mathcal{S}
\]

with \(\text{eig}(A + BFC) = \{ -0.3552, -5.0663 - 1.1373i, -5.0663 + 1.1373i \}\), and impulse response is shown in Fig 5.2.

Again, it can be easily shown that the Quadratic Invariance property does not hold here. In this example, we also adopt the code [81] used to compute \(H_2\) optimal state-feedback gain given a structure, the resulting gain stabilizes the plant but fails to satisfy the specific sparse pattern. We also use the sparsity-promoting algorithm to search for sparse gains, however, among all the sparse gains the algorithm found, no one satisfies the specific sparse pattern. We omit showing every single result their algorithm returned due to space consideration.
5.4.2 Design Sparse Output-Feedback Gain with Unknown Sparse Pattern

Here, we consider the discrete-time output-feedback case, with unknown sparse pattern.

\[
A = \begin{bmatrix}
-0.0791 & 0.8733 & -0.6679 \\
1.0092 & 0.5231 & 0.4420 \\
0.1902 & 0.2181 & -1.1440
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
0.7558 & 2.3445 & 0.4887 \\
0.9657 & 2.8938 & 2.5699 \\
1.3785 & 1.6142 & 0.4805
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
-0.7583 & -0.5812 & 0.2294 \\
0.7254 & 0.1046 & -0.2752 \\
0.0314 & 0.2598 & -0.9009
\end{bmatrix}
\]  

using Algorithm 5, we arrive at a sparse static controller as

\[
F = \begin{bmatrix}
0 & 0 & -0.7048 \\
0.0667 & 0 & -0.2203 \\
0.3719 & 0 & 0.7614
\end{bmatrix}
\]

with \(\text{eig}(A + BK) = \{-0.4212, 0.0808, -0.1649\}\), and impulse response is shown in Fig 5.3.
5.5. **SPARSE STRUCTURED DYNAMIC OUTPUT FEEDBACK CONTROLLER DESIGN**

It is interesting to note that our results unveil an interesting feature of the example: the fact that although $C$ is full rank, the second sensor can be ignored when designing an output feedback stabilizing controller (hence the column of zeros in $F$).

### 5.5 Sparse Structured Dynamic Output Feedback Controller Design

A convex approach designing a stabilizing static controller with sparse structure is proposed in previous section. Although static feedback controller has broad application domain and gets extensive research attention in control analysis, the dynamic controller still has advantage due to its general modeling and ability of using richer information from the past.

In the following section, we propose a convex approach to the problem of designing dynamic output feedback controllers subject to information constraints. The main idea is to recast the problem into an estimation form and exploit recent results [45] that allow for exactly solving this latter problem via convex optimization. These results are illustrated with an example showing the ability of the proposed algorithm to find suitable controllers, even in cases where quadratic invariance does not hold and thus the set of sparsity constrained controllers does not admit a convex characterization.
5.6 Statement of the Problem

Consider an LTI discrete time system of the form shown in Figure 5.4, where \( w \in R^{n_w} \) and \( u \in R^{n_u} \) represent disturbance and control inputs, respectively, \( \zeta \in R^{n_c} \) is a performance output and \( y \in R^{n_y} \) represents measurements available to the controller. In this context, the problem of interest can be formalized as:

**Problem 4.** Given a desired sparsity pattern \( S \), design an internally stabilizing controller \( C_s(z) \) such that \( C_s(z) \in S \), possibly subject to additional performance specifications on the closed-loop transfer matrix \( T_{\zeta w} \).

As mentioned before, it is well known that unless the plant \( P \) satisfies the so-called quadratic invariance property ([74, 92, 75]). Problem 4 is a very challenging, non-convex optimization problem. In the sequel, in order to obtain a tractable convex relaxation, we will make the following assumption:

**Assumption 1.** The plant \( P \) is strongly stabilizable, that is it can be stabilized with an open-loop stable controller ([93]).

5.7 Sparse Controller Synthesis via Convex Optimization

In this section we present the main result: a convex optimization based algorithm to find dynamic output feedback controllers subject to structural sparsity constraints. The main idea of the proposed method is to recast the problem into an estimation form and then exploit the results in [45] on synthesis of sparsity constrained optimal filters.
5.7. SPARSE CONTROLLER SYNTHESIS VIA CONVEX OPTIMIZATION

5.7.1 Finding Sparse Approximations of a Given Controller

In order to obtain tractable relaxations of Problem 4, we begin by considering the following related problem:

**Problem 5.** Given a strongly stabilizable plant \( P(z) \), an internally stabilizing open-loop stable controller \( C(z) \) and a sparsity pattern \( S \), find the best (in a sense to be precisely defined later) sparse approximation \( C_s(z) \in S \) to \( C(z) \).

\[ C(z) \]
\[ \rightarrow \]
\[ e \]
\[ \rightarrow \]
\[ C \]
\[ \rightarrow \]
\[ u \]
\[ \rightarrow \]
\[ P \]
\[ \rightarrow \]
\[ y \]

Figure 5.5: Example where seeking sparse approximations by minimizing the distance from the controller to the desired sparsity set does not necessarily yield the best performing sparse approximation.

In principle, one could just seek for solutions to the problem above by recasting it into a (convex) minimization problem of the form

\[
\min_{C_s(z) \in \mathcal{RH}_\infty \cap S} \| C(z) - C_s(z) \|_\alpha
\]

where \( \| . \|_\alpha \) denotes a suitable norm (e.g. \( \alpha = 2 \) or \( \alpha = \infty \)). However, note that this amounts to an open-loop approximation of the controller, which could lead to unnecessarily conservative results.

Consider for instance the topology shown in Figure 5.5, and assume that a controller \( C(z) \) has been found to achieve a desired performance level, \( \| T_{yw}(C) \|_\alpha \), where \( T_{yw}(C) = PC(I + PC)^{-1} \) denotes the closed-loop transfer function from the input \( w \) to the output \( y \) achieved by the controller \( C \). Clearly, having small \( \| C - C_s \|_\alpha \) does not guarantee that \( \| T_{yw}(C) - T_{yw}(C_s) \|_\alpha \) is small. On the other hand, this analysis suggests a minimization problem of the form:

\[
\min_{C_s(z) \in \mathcal{RH}_\infty \cap S} \| (C(I + PC)^{-1} - C_s(I + PC_s)^{-1}) \|_\alpha
\]

\[ = \min_{C_s(z) \in \mathcal{RH}_\infty \cap S} \| T_{uw}(C) - T_{uw}(C_s) \|_\alpha \] (5.30)

That is, minimizing the distance between closed-loop control actions, rather than open-loop controllers. Although in principle it seems that (5.30) above leads to a challenging non-convex problem, the main result of this paper shows that indeed this problem can be recast into a convex form through a suitable re-parameterization.
5.7.2 Sparse Controller Approximation as a Filtering Problem

Motivated by the discussion above, we propose to seek solutions to Problem 5, by recasting it into the filtering problem shown in Figure 5.6, where the goal is to obtain an optimal estimate \( \hat{u} \) of the actual control action \( u \), using \( y \) as input. This setup leads to the following problem:

**Problem 6.** Given \( P \), an internally stabilizing, open–loop stable controller \( C \), and a sparsity pattern \( S \), find a stable filter \( F(z) \in \mathcal{RH}_{\infty} \cap S \) such that:

1. In the absence of disturbances, \( \lim_{k \to \infty} \| u_k - \hat{u}_k \| = 0 \), for any initial condition of the plant \( P \).

2. It minimizes the worst case error, over all possible disturbances, between the estimated and actual control actions, \( \hat{u} = Fy \) and \( u \), respectively. That is:

\[
F(z) = \arg\min_{F(z) \in \mathcal{RH}_{\infty} \cap S} \left\{ \sup_{w \in \mathcal{B}_{2}} \| u - Fy \|_{\ell^2} \right\} \tag{5.31}
\]

The advantage of this approach over the direct minimization (5.30), is that, contrary to the control synthesis case, as recently shown in [45], the problem of synthesizing optimal filters subject to sparsity constraints is always convex. Specifically, we have the following result:

**Theorem 6.** Problem 6 is equivalent to the following convex optimization problem:

\[
J = \min_{Q(z) \in \mathcal{RH}_{\infty} \cap S} \| T_{uw}(C) - QT_{yw}(C) \|_{\infty} \tag{5.32}
\]

where \( T_{uw}(C) \equiv C(I - P_{22}C)^{-1}P_{21} \) and \( T_{yw}(C) \equiv (I - P_{22}C)^{-1}P_{21} \) denote the closed–loop transfer functions from the disturbance input \( w \) to the control \( u \) and measured output \( y \), respectively, obtained when the loop is closed with the controller \( C \).
5.7. SPARSE CONTROLLER SYNTHESIS VIA CONVEX OPTIMIZATION

Proof. Assume that the closed-loop system corresponding to the controller $C$, shown in Figure 5.6, has a state space realization

$$
\begin{bmatrix}
A_{cl} & B_{cl} \\
C_y & D_y \\
C_u & D_u
\end{bmatrix}
$$

From Theorem 2.1 in [43], it follows that all solutions to Problem 6 are of the form $F = F_1(z) + QF_2(z)$, where $Q \in \mathcal{RH}_\infty$ is an arbitrary stable transfer matrix, and where $F_1$ and $F_2$ have state-space realizations:

$$
F_1 = \begin{bmatrix} A_{cl} - LC_y & L \\
C_u & 0 \end{bmatrix}
$$

$$
F_2 = \begin{bmatrix} A_{cl} - LC_y & L \\
-RC_{cl}^{-1/2}C_y & RC_{cl}^{-1/2} \end{bmatrix}
$$

(5.33)

where $L$ is any matrix such that $A_{cl} - LC_y$ is stable and $R_c > 0$ is a scaling matrix. In our case, since $A_{cl}$ is stable, choosing $L = 0$ and $R_c = I$ yields $F_1 = 0$ and $F_2 = I$. With this choice, $F$ simply reduces to $Q$. Finally, using the equivalence between the $\|\cdot\|_{\ell_2} \to \ell_2$ and $\|\cdot\|_{\mathcal{H}_\infty}$ in (5.31) leads to the desired result.

5.7.3 Stability Analysis

![Figure 5.7: Setup for stability analysis when using the sparsity constrained controller.](image)

Next, we address the issues of closed-loop stability when using the sparse controller $C_s(z)$, and of how to select the nominal controller $C(z)$.

Theorem 7. Consider the topology shown in Fig 5.7. In this case, if the optimal value objective function in (5.32) satisfies $J < 1$ then the closed-loop obtained when using the sparse controller $C_s(z)$ is internally stable.
Proof. Recall (Lemma 3.1 in [93]) that, if the plant $P$ has a minimal realization

$$
\begin{bmatrix}
A & B_1 & B_2 \\
C_1 & D_{11} & D_{12} \\
C_2 & D_{21} & D_{22}
\end{bmatrix}
$$

then a controller $C_s$ internally stabilizes $P$ if and only if the transfer matrices $(I + P_{22}C_s)^{-1}$, $(I + P_{22}C_s)^{-1}P_{22}$ and $C_s(I + P_{22}C_s)^{-1}$ are stable. For ease of notation, define $T = (C_s - C)P_{22}(I + CP_{22})^{-1}$ and note that

$$
det[I + P_{22}C_s] = det[I + C_sP_{22}] = \\
det[I + CP_{22} + (C_s - C)P_{22}] = \\
det[(I + T)(I + CP_{22})] = \\
det[I + T]det[I + CP_{22}]
$$

(5.34)

Since $C$ is by assumption internally stabilizing, the second factor in the equation above does not have any roots outside the open unit disk. Thus, it follows that if $\|T\|_{\mathcal{H}_\infty} < 1$, then $(I + P_{22}C_s)^{-1}$ is stable. Since in this topology $P_{22} = P_{21}$, this last condition is equivalent to $J < 1$. Further, this condition is also sufficient for stability of $C_s(I + P_{22}C_s)^{-1}$, since by construction $C_s$ is open-loop stable. To complete the proof, we need to show stability of $(I + P_{22}C_s)^{-1}P_{22}$. To this effect, note that:

$$
(I + P_{22}C_s)^{-1}P_{22} = P_{22}(I + C_sP_{22})^{-1} \\
= [P_{22}(I + CP_{22})^{-1}] \times (I + T)^{-1}
$$

(5.35)

As shown earlier in the proof, $J = \|T\|_{\mathcal{H}_\infty} < 1$ guarantees stability of the second factor in (5.35). Stability of $(I + P_{22}C_s)^{-1}P_{22}$ follows now from the fact that the first factor in (5.35) is also stable, since, by hypothesis $C$ internally stabilizes $P_{22}$. \qed

Remark 4. Note that in the topology shown in Fig. 5.7, $T = (C_s - C)T_{yw}(C)$. Hence, a sufficient condition for stability\(^1\) is $\|C - C_s\|_{\mathcal{H}_\infty} < \|T_{yw}\|_{\mathcal{H}_\infty}^{-1}$, which suggest selecting $C_s$ to simply minimize $\|C - C_s\|_{\mathcal{H}_\infty}$. Note, however, that this approach is always more conservative than selecting $C_s$ to minimize $\|T\|_{\mathcal{H}_\infty}$, as proposed in (5.32). Nevertheless, this reasoning suggests that, in the absence of additional performance constraints, the controller $C(z)$ should be chosen to minimize $\|T_{yw}(C)\|_{\mathcal{H}_\infty}$. Such a controller can be found using for instance the algorithms proposed in [94, 95, 96, 97, 98].

\(^1\)A similar conclusion can be obtained from a simple small gain type argument.
5.7.4 Proposed Algorithm

Based on the considerations above, we propose the following sparse control synthesis algorithm:

1) Given a plant $P(z)$, design a stable $\mathcal{H}_\infty$ controller $C(z)$ such that $\|T_{yw}(C)\|_\infty \leq \gamma$.

2) Denote by $G(z) = \mathcal{F}_G(P, C)$ the closed-loop system, with a state space realization given by:

\[
\begin{bmatrix}
 y \\
 u
\end{bmatrix} =
\begin{bmatrix}
 A_{cl} & B_{cl} \\
 C_y & D_y \\
 C_u & D_u
\end{bmatrix} w
\]  

(5.36)

3) Form $T_{uw}(C)$ and $T_{yw}(C)$ as

\[
T_{uw}(C) = \begin{bmatrix}
 A_{cl} & B_{cl} \\
 C_u & 0
\end{bmatrix}
\]

\[
T_{yw}(C) = \begin{bmatrix}
 A_{cl} & B_{cl} \\
 C_y & D_y
\end{bmatrix}
\]

4) Obtain $C_s(z)$ by solving

\[
C_s = \arg\min_{C_s(z) \in \mathcal{R}\mathcal{H}_\infty \cap \mathcal{S}} \|T_{uw}(C) - C_sT_{yw}(C)\|_\infty
\]

5.8 Illustrative Example

In this section we illustrate the effectiveness of the proposed approach with an example from the static controller design section. Note that in this case, the LMI-based static sparse gain design approaches \([82, 84]\) fail to produce a solution. Further, since the pair (plant, given sparsity pattern) violates the Quadratic Invariance property, the set of stabilizing controllers with the desired sparsity pattern does not admit a convex characterization, and hence the techniques proposed for instance in \([92, 75, 99]\) cannot be used.

We consider a discrete-time system with state space realization:

\[
P(z) = \begin{bmatrix}
 A & B \\
 C & 0
\end{bmatrix}
\]
CHAPTER 5. SPARSE STRUCTURED OUTPUT FEEDBACK CONTROLLER DESIGN

with

\[
A = \begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
0.0556 & -0.3477 & -1.8320 & -2.4300
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
0.8026 & 0.9874 & 0.3856 \\
0.5635 & 0.4275 & -0.3341 \\
0.3888 & -0.7953 & 0.8435 \\
0.6525 & -0.1054 & 0.2650
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
0 & 1 & 0 & 0.8644 \\
0 & 0 & 1 & -0.1482 \\
0 & 0 & 0 & -0.3990
\end{bmatrix}
\]

and a desired sparsity pattern \( S \) defined by the binary matrix:

\[
S = \begin{bmatrix}
1 & 1 & 0 \\
1 & 1 & 1 \\
0 & 1 & 1
\end{bmatrix}
\]

Since the traditional \( \mathcal{H}_\infty \) synthesis method produces an unstable controller, we use [96] to synthesize a fifth order stable centralized controller \( C \) with \( \| \mathcal{F}_1(P, C) \|_\infty = 2.3051 \). Using this controller, we apply the proposed sparse controller design algorithm with different order choices for \( C_s \in S \) and plot the corresponding objective function values of \( \| T_{uw}(C) - C_s T_{yw}(C) \|_\infty \) in Figure 5.8. As shown there, using a controller order higher than 4, leads to values of \( \| T_{uw}(C) - C_s T_{yw}(C) \|_\infty \) smaller than 1. Hence, Theorem 7 guarantees stability of the closed-loop system.

In order to illustrate the advantages of the proposed approach, in Figure 5.9, we compare the performance achieved by the sparse controllers, as a function of their order, against that of the static controller obtained using the approach proposed in static sparse structured controller design. As shown there, as the controller order increases, the closed-loop \( \mathcal{H}_\infty \) norm decreases. For completeness, we also show the closed-loop \( \mathcal{H}_\infty \) achieved using the stable centralized controller \( C \), illustrating the fact that imposing sparsity constraints on the controller leads to a gap in performance. It is also worth noting that a sparse controller obtained by simply projecting this centralized \( \mathcal{H}_\infty \) optimal controller onto \( S \) fails to stabilize the system.
5.8. ILLUSTRATIVE EXAMPLE

Figure 5.8: Values of $\|T_{uw}(C) - C_s T_{yw}(C)\|_\infty$ for different sparse controller orders.

Figure 5.9: Closed loop $\mathcal{H}_\infty$ norm achieved by the different controllers.
Chapter 6

Identifying Sparse Dynamical Graphical Models via Super-Atomic Norm Minimization Approach

6.1 Motivation and Previous Works

During the past few years, considerable attentions have been devoted to the problem of identifying dynamical graphical models, which can be represented by a graph structure $G = \{V, E\}$. Here, the vertices $V$ are associated with time series and the edges $E$ relate the values of these series at different time instants.

The identification problem is generally ill posed in the absence of additional priors, since potentially an infinite number of topologies can explain a given set of finite, noisy observations. Thus, a “sparsity” prior is added to regularize the problem typically, encapsulating the fact that, amongst all possible solutions, the one with the fewest number of links is usually the correct one. An example of this situation is when graphs are used to encapsulate causal relationships between agents and predict future behavior, exploiting the concept of Granger causality [100]. Several approaches have been proposed to solve the resulting problem. A cycling descent algorithm was adopted in [101] that directly attempted to enforce sparsity and used causal Laguerre basis functions to model the connections between two time series. Since attempting to directly enforce sparsity leads to non-convex, hard to solve problems, a large portion of the existing literature uses the $\ell_1$ norm as a convex surrogate for sparsity, leading to a number of convex optimization based algorithms...
Valdés-Sosa et al. [102] discussed the identification of a sparse interconnection structure arising in the context of brain function by posing the links between time series as first order Vector Auto-Regressive (VAR) process. This was extended to higher order models in [103]. Identifying chemical reaction network and genetic network has been discussed in [104] and [105], respectively. Finally, [106] provided conditions for compressive sensing tools to recover the exact topology in the case of noise free data.

\( \ell_1 \) regularization based approaches tend to enforce sparsity in terms of the total number of non-zero coefficients involved in representing the graph. Thus, they do not, typically, lead to sparse topologies, since the latter requires enforcing \textit{block-sparsity}, that is, all coefficients of the model associated with a given edge should be zero simultaneously. This observation motivated the use of group lasso based approaches [107, 108, 109]. While these methods usually work well, in some cases they fail to produce the sparsest topology, motivating the introduction of re-weighted iterative algorithms [110, 111, 112]. These algorithms work well in practice, but the use of a sum-of-\( \ell_2 \) norms objective function leads to quadratic programs, whose complexity is larger than \( n^3 \).

An alternative to the approaches above is given by orthogonal matching pursuit type algorithms. Cycling Orthogonal Least Squares (COLS) [111] seeks to find sparse solutions using a modified Orthogonal Least Squares algorithm. A Block Orthogonal Matching Pursuit algorithm has also been investigated [113], using a notion of coherence analog to the one proposed in the context of compressive sensing [114], in this case applied to the network-derived sensing matrix. A further extension to cases where the blocks have different sizes was presented in [115]. At the present time, these approaches cannot handle cases, often arising in practice, where the network is subject to unknown, but sparse, external inputs.

Finally, a Bayesian approach has been proposed to obtain sparse topologies in [116], where the problem was posed as sparse input selection for MISO LTI systems. A sparse plus low rank criterion has been discussed in [117], in which a two layer structure (manifest and latent) was assumed. Typically, Bayesian approaches require strong prior information about the system to be identified and the resulting algorithms have relatively high computational cost.

As an alternative to the approaches above, inspired by recent progress in obtaining sparse representations by exploiting the geometry of the problem, in this chapter we propose a new super-atomic norm based approach. This proposed method aims at solving a similar problem as group-lasso does. However, as we show next, due to the introducing of super-atoms, we achieve substantial improvement in computational efficiency. The motivation of this work is based on the following points:
1. Extension of the original atomic norm framework proposed in \([48]\) for obtaining sparse solutions to sets of linear equations to the *block-sparse* case. This is accomplished by introducing the concept of super-atoms and its associated super-atomic norm, and showing that minimizing this norm indeed minimizes the convex envelope (e.g. the tightest convex approximation) of cardinality of a set of vectors.

2. Showing that the approach above leads to very efficient algorithms for minimizing functions subject to block-sparsity constraints that only require performing inner products and thus can comfortably handle very large data sets.

3. Recasting the network identification framework into a constrained super-atomic norm minimization framework, which allows for directly using these algorithms

These results are illustrated with several examples where the proposed approach compares favorably against existing ones both in terms of recovery of the underlying network and computational complexity.

### 6.2 Problem Statement

As indicated at the beginning of this chapter, we consider models represented by a directed graph \(G = \{V, E\}\) structure, where each node \(V\) corresponds to a given time series, and the edges \(E\) connecting these are linear shift invariant operators. The corresponding equations are given by

\[
x_j(t) = \sum_{i=1}^{n} \sum_{k=1}^{\ell} c_{j,i}(k)x_i(t-k) + \eta_j(t),
\]

\[
t \in [r+1, T], \ j = 1, \ldots, n
\]

where \(x_{j}(\cdot)\) denotes the time series at the \(j^{th}\) node, \(c_{j,i}(\cdot)\) are the coefficients of an ARX model relating the present value of the time series at node \(j\) to the past values measured at node \(i\), and \(\eta_{j}(t)\)
represents measurement noise. For ease of notation, define

\[ x_j = [x_j(T), \ldots, x_j(r+1)]^T, \]

\[ \eta_j = [\eta_j(T), \ldots, \eta_j(r+1)]^T, \]

\[ c_{j,i} = [c_{j,i}(1), \ldots, c_{j,i}(r)]^T, \]

\[ c_j = [c_{j,1}, \ldots, c_{j,n}]^T, \]

\[ C = [c_1, \ldots, c_n]^T, \]

\[ X = [x_1, \ldots, x_n]^T, \]

\[ H_i = \begin{bmatrix} x_i(T-1) & x_i(T-2) & \cdots & x_i(T-r) \\ x_i(T-2) & x_i(T-3) & \cdots & \vdots \\ \vdots & \cdots & \cdots & \vdots \\ x_i(r) & \cdots & \cdots & x_i(1) \end{bmatrix}, \]

\[ H = [H_1 \ldots H_n], \]

\[ \Xi = [\eta_1 \ldots \eta_n]^T. \]

With this notation, the equations describing the complete model can be written in compact form as:

\[ X = HC + \Xi \quad (6.2) \]

Our goal is to identify models of the form (6.2) from experimental data and a-priori information about the noise and the order of the edge systems. As noted in the previous section, due to the finite data record and the presence of noise, this problem is ill-posed, admitting infinite solutions. However, in the absence of additional information, amongst all these solutions, the sparsest one, in the sense of having the smallest number of edges, is often the most desirable. Thus, we will add a “sparsity” prior leading to the following regularized problem:

**Problem 7.** Given \( T \) measurements of \( n \) time series \( x_i(t), \ i = 1, \ldots, n, \ t \in [1, T] \), and upper bounds \( \epsilon \) and \( r \) on the noise level and edge model order, respectively, solve:

\[ \min \sum_j \sum_i \|c_{j,i}\|_0 \quad (6.3) \]

\[ \text{s.t.} \quad (6.2) \text{ and } \|\eta_j\|_2 \leq \epsilon, \forall j = 1, \ldots, n \]

where, \( c_{j,i} \in \mathbb{R}^r \). Thus, the objective function in this problem is precisely \( |E| \), the number of edges in the graph.
Note that due to its structure, the problem above decouples into \( n \) subproblems of the form:

**Subproblem 1.**

\[
\min \| \{ c_i \} \|_0 \\
\text{s.t.} \quad x_j = \sum_i H_i c_i + \eta_j \\
\| \eta_j \|_2 \leq \epsilon
\]  

(6.4)

where, by a slight abuse of notation, we have defined \( \| \{ c_i \} \|_0 \) as the number of non-zero vectors in the set \( \{ c_{i,j} \} \) given \( j \).

### 6.3 Super Atoms and Block Sparsity

The class of problems considered in this chapter require enforcing *block-sparsity*, rather than sparsity. This will be accomplished by considering *super-atoms* and the associated super-atomic norm, rather than the traditional atomic norms. Assume that the set \( \mathcal{A} \) can be partitioned into \( N \) centrally symmetric subsets \( \mathcal{A}_i \) such that \( \mathcal{A} = \bigcup_i \mathcal{A}_i \) and \( \mathcal{A}_i \cap \mathcal{A}_j = \emptyset, \ i \neq j \). In the sequel, we will refer to the sets \( \mathcal{A}_i \) as *super-atoms*. Further, to each super-atom \( \mathcal{A}_i = \{ a_{i,1}, \ldots a_{i,n_i} \} \) we will associate the matrix \( \mathbf{A}_i \) having as its \( j \)-th column \( a_{i,j} \), the coordinates of the atom \( a_{i,j} \) in a suitable basis in \( X \).

**Definition 7.** Given a set of super-atoms \( \{ \mathcal{A}_i \} \) and a point \( x \in X \), its super-atomic norm is defined as:

\[
\| x \|_{sA} = \inf \left\{ \tau > 0 : \ x = \sum_i (\tau \mathbf{A}_i) c_i \text{ and } \sum_i \| c_i \|_\infty = 1 \right\}
\]  

(6.5)

Note that the definition above reduces to the usual atomic norm definition when \( \mathcal{A}_i = \{ a_i \} \).

This connection and the connection with block-sparsity is highlighted by the following easily shown result:

**Lemma 5.**

\[
\| x \|_{sA} = \min_c \sum_{i=1}^N \| c_i \|_\infty \\
\text{s.t.} \quad x = \sum_i \mathbf{A}_i c_i
\]  

(6.6)

**Remark 5.** Recall that, given a vector sequence \( \{ c \} \), \( \| c \|_\infty \leq 1 \), the convex envelope (e.g. the tightest convex relaxation) of its cardinality is given by [118]:

\[
\| \{ c \} \|_{o,env} = \sum_i \| c_i \|_\infty
\]
Thus, from Lemma 5, it follows that, minimizing the super-atomic norm is a good surrogate to block-sparse representations, a key property that we will exploit in this chapter.

### 6.4 Sparse Network Identification via Super-Atomic Norm Minimization

From the results in the previous section, it follows that Problem 7 can be recast into a collection of $n$ super-atomic norm minimizations of the form

$$
\begin{align*}
\min_{x} & \quad \|z\|_{s,A} \\
\text{s.t.} & \quad \|z - x_j\|_2 \leq \epsilon
\end{align*}
$$

(6.7)

by simply defining each super-atom as the collection of columns from the matrices $H_i$, (e.g a collection of vectors, each containing delayed measurement of the respective time-series):

$$A_i = \{H_i(:, t)\}, \; t = 1, \ldots r$$

The problem above is convex and thus can be solved for instance using interior point methods. However, while these methods work well for moderate size problems, their poor scaling properties renders them impractical as the size of the data grows. Thus, in this chapter, rather than solving (6.7), we will solve the related problem

$$
\begin{align*}
\min_{x} & \quad \|z - x_j\|_2 \\
\text{s.t.} & \quad \|z\|_{s,A} \leq \tau
\end{align*}
$$

(6.8)

that is, we will impose soft, rather than hard constraints on the fitting error. The advantage of the formulation (6.8) is that it is amenable to be solved by the following extension of Algorithm 1:

Steps 4–6 in the algorithm above correspond to Step 3 in Algorithm 1. The first step selects the super-atom whose elements yield the largest decrease trend in the cost function and Steps 5 and 6 select the best linear combination of elements in this super-atom. Thus, the combination of Steps 4-6 guarantees that at each step, both the objective function will improve, unless already at the optimum, that only elements from a single super-atom will be added to the solution, and that, at all times, $\|z^{(k)}\|_{s,A} \leq \tau$ reveals the projection-free property of the proposed algorithm. Proceeding as in [119] it can be shown that, as long as the objective function $f(\cdot)$ is convex and smooth, the algorithm above is guaranteed to converge to the optimum, with a convergence rate of $O(\frac{1}{k})$. Further, as shown below, all the steps in Algorithm 6 admit an explicit solution once being applied to solve problem (6.8).
Algorithm 6 Convex Minimization subject to Super-atomic Norm Constraints

1: Data: set of super-atoms \( \mathcal{A} = \{ \mathcal{A}_1, \ldots, \mathcal{A}_i, \ldots \} \)
2: Initialize \( z^{(0)} \leftarrow \tau a \) for some arbitrary \( a \in \mathcal{A} \)
3: for \( k = 0, 1, 2, 3, \ldots, k_{\text{max}} \) do
4: \( L \leftarrow \arg \min_m \left\{ \min_{||c||_\infty \leq 1} \langle \partial f(z^{(k)}), \sum_i a_{i,m} c_i \rangle \text{ s.t. } a_{i,m} \in \mathcal{A}_m \right\} \)
5: \( c \leftarrow \arg \min_{||c||_\infty \leq 1} \langle \partial f(z^{(k)}), \sum a_{i,L} c_i \rangle \text{ s.t. } a_{i,L} \in \mathcal{A}_L \)
6: \( a \leftarrow \sum_i a_{i,L} c_i \)
7: \( \alpha_k \leftarrow \arg\min_{|c| \in [0,1]} f(z^{(k)} + \alpha [\tau a - z^{(k)}]) \)
8: \( z^{(k+1)} \leftarrow z^{(k)} + \alpha_k [\tau a - z^{(k)}] \)
9: end for

Lemma 6. Let \( \mathcal{A}_l \) denote the matrix having as columns the coordinates of \( a_{i,i} \), the elements of the super-atom \( \mathcal{A}_i \), and assume that the super-atoms are centrally symmetric, that is, \( a \in \mathcal{A}_i \Rightarrow -a \in \mathcal{A}_i \). Then, explicit solutions to Steps 4-6 of Algorithm 6 are given by

(i) Step 4: \( L \leftarrow \arg \max_i \{ ||[\partial f(z^{(k)})]^T \mathcal{A}_i||_1 \} \)

(ii) Step 5: \( c = -\text{sign}([\partial f(z^{(k)})]^T \mathcal{A}_L) \)

(iii) Step 6: \( a \leftarrow \mathcal{A}_L c \)

Further, for the case where \( f(z) = \frac{1}{2} \| x_j - z \|_2^2 \), the explicit solution to Step 7 is given by \( \alpha_k = \text{max}\{\min\{\alpha_o, 1\}, 0\} \) where

\[
\alpha_o = \frac{[\tau a - z^{(k)}]^T [x_j - z^{(k)}]}{\| \tau a - z^{(k)} \|^2_2}
\] (6.9)

Proof. We have that, if \( |c_i| \leq 1 \) then

\[
\langle \partial f(z^{(k)}), \sum_i a_{i,m} c_i \rangle \geq -\sum_i |\langle \partial f(z^{(k)}), a_{i,m} \rangle| = -\| \partial f(z^{(k)})^T \mathcal{A}_m \|_1
\]

where we used the fact that the super-atoms are centrally symmetric. Thus, the minimum in Step 4 is achieved by the super-atom that maximizes \( ||\partial f(z^{(k)})^T \mathcal{A}_m||_1 \) with the corresponding linear combination of atoms given by \( a = \mathcal{A}_L c \) with \( c_i = \text{sign} \langle \partial f(z^{(k)}), a_{i,L} \rangle \). The proof of properties (i) through (iii) follows now from the fact that the sets \( \mathcal{A}_i \) are centrally symmetric. Finally, using the explicit expression for \( f(.) \) to compute \( \partial f/\partial a \) yields:

\[
\partial f/\partial a = \alpha || \tau a - z^{(k)} ||^2_2 - [\tau a - z^{(k)}]^T [x_j - z^{(k)}]
\]
Hence \( f(z^{(k)} + \alpha [\tau \mathbf{a} - z^{(k)}]) \) is decreasing in \( \alpha < \alpha_o \). The proof follows now from convexity of \( f(\cdot) \).

The results above lead to the following Frank-Wolfe type algorithm for the specific case of network identification:

**Algorithm 7** Topology Identification via Super-Atomic Norm Minimization

1. Define \( A = \{A_1, \ldots, A_i, \ldots \} \) and \( \phi = \{c_1, \ldots, c_i, \ldots \} \). Denote the \( l \)-th element as \( A_l \) and \( \phi_l \), respectively.
2. Initialize \( z^{(0)} = 0 \) and \( \phi_l = 0, \forall l = 1, 2, \ldots \)
3. for \( k = 0, 1, 2, 3, \ldots, k_{\text{max}} \) do
   4. \( L \leftarrow \arg \max \{||\partial f(z^{(k)})||^T A_l||_1 \} \)
   5. \( e = -\text{sign}((\partial f(z^{(k)}))^T A_L) \)
   6. \( a \leftarrow A_L e \)
   7. \( \alpha_k \leftarrow \max\{\min\{\alpha_o, 1\}, 0\} \) where \( \alpha_o \) is defined in (6.9)
   8. \( z^{(k+1)} \leftarrow z^{(k)} + \alpha_k [\tau \mathbf{a} - z^{(k)}] \)
   9. \( \phi_l = (1 - \alpha_k) \phi_l, \forall l \)
   10. \( \phi_L = \phi_L + (\alpha_k \tau) e \)
4. end for

Note that this algorithm requires computing only inner products and sorting a vector and thus can comfortably handle very large data sets.

### 6.5 Extensions

In this section we cover several extensions of the basic algorithm needed to handle practical scenarios.

#### 6.5.1 External inputs

Many practical situations require taking into account relatively rate external events. Following [110], we will model these interactions by adding at each node, a piecewise constant signal \( u_j(\cdot) \), with a sparse derivative,

\[
x_j(t) = \sum_{i=1}^{n} \sum_{k=1}^{r} c_{j,i}(k)x_i(t-k) + u_j(t) + \eta_j(t),
\]

\( t \in [r+1, T], \ j = 1, \ldots, n \) (6.10)
This extension fits naturally the proposed framework by modifying the objective in (6.3) to

$$
\min \| \{ c_i \} \|_0 + \lambda \| \{ \Delta u_j \} \|_0
$$

(6.11)

where $$\Delta u_j = [u_j(2) - u_j(1) \ldots u_j(t) - u_j(t-1) \ldots]$$ and the parameter $$\lambda$$ allows for trading-off graph versus input sparsity. The problem above can be reformulated in terms of super-atomic norm minimization, by simply adding the following super-atoms to the set $$\mathcal{A}$$:

$$
\mathcal{A}_u = \{ u_1, \ldots, u_T \}
$$

(6.12)

where $$u_t$$ is defined as the $$t$$-th column of a lower triangular matrix with $$\{0, 1\}$$ elements. Thus, Algorithm 7 can be easily extended to handle external inputs by simply adding $$\mathcal{A}_u$$ into $$\mathcal{A}$$ and the corresponding coefficients to $$\phi$$.

**Remark 6.** To understand the effect of $$\lambda$$ on the super-atoms, let’s consider a simple case. Without loss of generality, let’s assume the next convex optimization problem has unique solution:

$$
\tau^* = \min \sum_i w_i \| c_i \|_{\infty}
$$

s.t. $$\| x_j - z \|_2 \leq \epsilon^*$$

$$
z = \sum_i \mathcal{A}_i c_i
$$

(6.13)

where, $$w_i$$ denotes weight which is positive number. We can rewrite the above problem as

$$
\epsilon^* = \min \| x_j - z \|_2
$$

s.t. $$\sum_i w_i \| c_i \|_{\infty} \leq \tau^*$$

$$
z = \sum_i \frac{\mathcal{A}_i}{w_i} c_i
$$

(6.14)

Let’s define new super-atoms $$\tilde{\mathcal{A}}_i = \frac{\mathcal{A}_i}{w_i}$$, and corresponding coefficient $$\tilde{c}_i = \frac{w_i}{\mathcal{A}_i} c_i$$, we can further rewrite the problem as

$$
\epsilon^* = \min \| x_j - z \|_2
$$

s.t. $$\sum_i \| \tilde{c}_i \|_{\infty} \leq \tau^*$$

$$
z = \sum_i \tilde{\mathcal{A}}_i \tilde{c}_i
$$

(6.15)

The above problem is a traditional atomic norm optimization problem and equivalent to the original weighted convex optimization problem.
6.6. REWEIGHTED HEURISTIC SUPER-ATOMIC NORM MINIMIZATION

6.5.2 Missing data

Consider now a situation where some of the data is missing, due for instance to sensor outages, or in the case of video-based applications, occlusion. This scenario can be handled by noticing that from (6.2) it follows that

\[ M = \begin{bmatrix} X - \Xi & H \end{bmatrix} \]

does not have full column rank. Thus, missing data can be recovered by minimizing the rank of \( M \) with respect to the missing data and noise sequence, for instance by solving a regularized nuclear norm minimization problem of the form:

\[
\min_{\hat{m}, \hat{X}, \hat{H}} \| \begin{bmatrix} \hat{X} & \hat{H} \end{bmatrix} \|_* + \mu \| \hat{m} - m \|_\infty
\]  

(6.16)

where, \( \begin{bmatrix} \hat{X} & \hat{H} \end{bmatrix} \) denotes the low rank estimation of \( M \), \( \hat{m} \) and \( m \) denote the elements of \( \begin{bmatrix} \hat{X} & \hat{H} \end{bmatrix} \) and \( \begin{bmatrix} X & H \end{bmatrix} \), respectively, at the positions where data is available.

6.6 Reweighted Heuristic Super-Atomic Norm Minimization

In this section, we propose a reweighted heuristic algorithm based on Algorithm 7 with the goal of further improving sparsity. The idea of reweighted heuristic procedure was first introduced in [120]. This algorithm is used in the computer vision experiments in section 6.7. In each iteration, our algorithm solves problem as

\[
\min_{c_i, p_t} \| x_j - z \|_2
\]

s.t. \( \| z \|_{s,A} \leq \tau \)

(6.17)

where, \( z \) is defined by two type of super atoms, \( A_a \) from the video data and \( A_u \) for the piecewise constant input with the corresponding coefficients denoted as \( c_i \)'s and \( p_t \)'s, respectively. In order to consider the effect of each agent equally at the beginning, we initialized the weights based on the 2 norm of corresponding super atoms. The algorithm is proposed in Algorithm 8.

6.7 Examples

In this section, we present four examples consisting of two synthetic examples, one example using videos from computer vision data set, and one using real world video from Youtube.
Algorithm 8 Reweighted Causality Identification in Computer Vision Application

1: Initialize $w^a_i = \|A_a(i)\|_2/\text{mean}(\{\|A_a(i)\|_2, \forall i\}), \forall i; w^u_t = 1, \forall t; s_i = 1, \forall i \neq j$ and $s_j > 1$

2: while not converge do
3: \hspace{1em} $w^a_i \leftarrow s_i w^a_i$, $w^u_t \leftarrow \lambda w^u_t$
4: \hspace{1em} $A_a \leftarrow \frac{1}{w^a_i} \circ A_a$, $A_u \leftarrow \frac{1}{w^u_t} \circ A_u$
5: \hspace{1em} Solve (6.17) using Algorithm 7
6: \hspace{1em} $c_t \leftarrow \frac{1}{w^a_i} c_t$, $p_t \leftarrow \frac{1}{w^u_t} p_t$
7: \hspace{1em} $w^a_i \leftarrow 1/(\|c_t\|_\infty + \delta)$
8: \hspace{1em} $w^u_t \leftarrow 1/(|p_t| + \delta)$

9: end while

In these examples, we compare the proposed methods against Cycling Orthogonal Least Squares (COLS)[111], Group Lasso (GL)[108] and its variant modified to handle external inputs (GLEI) [112].

6.7.1 Synthetic Example without External Input

In this example, we first generated $N$ nodes, and for each node a signal of length 1000 was drawn from a Normal distribution $\mathcal{N}(0, I)$. Then, in order to generate the time series observed at the output node, we randomly chose $n_a$ nodes and generated random ARX models of order $r$, with coefficients uniformly distributed on the open interval $(0, 1)$. Finally, the resulting output signal was then corrupted using Gaussian noise drawn from $\mathcal{N}(0, 0.05 I)$, achieving a Signal-to-Noise ratio around $25dB$.

We considered three different scenarios obtained by fixing two parameters from the set $r$, $n_a$ and $N$ and varying the third. For each parameter setting, we ran 10 experiments and compared the results against those obtained using Group Lasso and COLS. All three approaches successfully identified the underlying system. However, as shown in Tables 6.1, 6.2 and 6.3 the proposed Frank-Wolfe based algorithm (Algorithm 7) outperforms the others in terms of computational complexity as the size of the problem grows. It’s worth noting that we implemented Group Lasso using the highly optimized commercial solver Gurobi [121].
### 6.7. EXAMPLES

#### Table 6.1: Mean Computing Time as a Function of System Order ($N = 500, n_a = 10$)

<table>
<thead>
<tr>
<th>$r$</th>
<th>2</th>
<th>6</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed</td>
<td>0.8232</td>
<td>5.4491</td>
<td>9.6660</td>
</tr>
<tr>
<td>Group Lasso</td>
<td>3.0599</td>
<td>76.0957</td>
<td>352.1598</td>
</tr>
<tr>
<td>COLS</td>
<td>3.8695</td>
<td>11.1641</td>
<td>19.7986</td>
</tr>
</tbody>
</table>

#### Table 6.2: Mean Computing Time as a Function of Number of Links ($N = 500, r = 6$)

<table>
<thead>
<tr>
<th>$n_a$</th>
<th>5</th>
<th>10</th>
<th>20</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed</td>
<td>5.5009</td>
<td>5.4970</td>
<td>5.4772</td>
<td>5.6117</td>
</tr>
<tr>
<td>Group Lasso</td>
<td>52.3320</td>
<td>61.0487</td>
<td>53.2604</td>
<td>55.9573</td>
</tr>
<tr>
<td>COLS</td>
<td>2.7994</td>
<td>11.2746</td>
<td>47.1583</td>
<td>389.2545</td>
</tr>
</tbody>
</table>

#### Table 6.3: Mean Computing Time as a Function of the Total Number of Nodes ($r = 6, n_a = 10$)

<table>
<thead>
<tr>
<th>$N$</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>400</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed</td>
<td>0.4373</td>
<td>1.2237</td>
<td>2.6294</td>
<td>4.0679</td>
<td>5.4672</td>
</tr>
<tr>
<td>Group Lasso</td>
<td>0.5539</td>
<td>3.7758</td>
<td>12.7853</td>
<td>29.3695</td>
<td>51.5379</td>
</tr>
<tr>
<td>COLS</td>
<td>2.0871</td>
<td>4.2519</td>
<td>6.4819</td>
<td>8.7866</td>
<td>11.1526</td>
</tr>
</tbody>
</table>
6.7.2 Synthetic Example with External Input

In this example, the output node was generated using $N = 200$, $n_a = 10$ and $r = 6$ with an external input which switches 5 times at random location. We compared the Frank-Wolfe based algorithm against COLS and GLEI. While the proposed method and GLEI successfully identified the correct links, COLS failed to find 8 out of a total $n_a$ links, due to the fact that it doesn’t consider external inputs. The recovered input signal is shown in Figure 6.1. Although GLEI successfully identified the correct links, it took around 35 seconds to solve the problem using Gurobi, while the proposed Frank-Wolfe based algorithm only took 7 seconds.

6.7.3 Human Interaction

The goal here is to identify causal relationships from video data. For this experiment, we took the first example from [112], which considers two video sequences from UT Human Interaction Data Set [122]. In both video sequences, we use as data the position of each agent’s head in image coordinates, normalized to the interval $[-1, 1]$. Note that in both sequences, there are missing data due to either occlusion or targets leaving the image. Due to these missing data, none of the algorithms was directly applicable. Therefore, we first recovered those missing data via nuclear norm minimization, then ran the proposed reweighted algorithm (Algorithm 8), reweighted GLEI
and COLS on the recovered data using system order $r = 2$. For the proposed method, we set the self-loop penalty equal to 10, $\lambda = 0.05$ and $\tau = 5$. For the reweighted GLEI, we adopt the parameter setting reported in [112]. Since COLS needs information on number of links, we used 1 for each agent which is intuitively consistent with the ground truth, that is the fact that the video consists mostly of pair-wise interactions.

For both sequence 6 and 16, a clip of around 100 frames was used. The causal correlations between agents identified by all three methods are shown in Figure 6.2 and 6.3, respectively. As in example of section 6.7.2, COLS fails to identify the correct relationships, since it does not take into account external inputs\(^1\). On the other hand, both the proposed method and reweighted GLEI correctly identified the causally interacting people. However, the proposed method required less computing time per iteration than reweighted GLEI. For sequences 6 and 16, the proposed method took 0.0643 and 0.1115 seconds per iteration, respectively, while reweighted GLEI required 0.4867 and 0.3583 seconds per iteration. In this experiment, both the proposed method and reweighted GLEI converged in about the same number of iterations.

### 6.7.4 Youtube Tennis Game

In this example, the goal is to identify causally interacting agents in a clip of 210 frames taken from a mixed double tennis match at the London 2012 Olympics. The position of the centroid of each player was recorded, and normalized to interval $[-1, 1]$. In this experiment, we also set the system order for each edge to $r = 2$. For both proposed method and reweighted GLEI, we set self-loop penalty to 10 and $\lambda = 0.5$. In proposed method, we used $\tau = 20$. For reweighted GLEI, we chose $\epsilon = 0.1$. For COLS, we set the link number on each player as 2 which is consistent to the fact that each team consisted of two people. The causal correlations identified by three methods are shown in Figure 6.4.

In this case, although the ground truth is not available, intuitively it is expected that players will react primarily to their opponents. Therefore, we would expect links between player and his/her opponents. The graph identified by COLS doesn’t match the intuition, while the ones obtained using the proposed method and reweighted GLEI match. For this experiment, Algorithm 8 and reweighted GLEI also took about the same number of iterations to converge. However, each iteration of the proposed method required 0.3802 seconds versus 2.13 for GLEI.

\(^1\)In this applications, these inputs account for interactions between an agent and its environment.
Figure 6.2: Sample Frames of UT Sequence 6 showing the causally interacting groups identified using different methods. Top: Proposed Method. Center: reweighted GLEI. Bottom: COLS.
Figure 6.3: Sample Frames of UT Sequence 16 showing the causally interacting groups identified using different methods. Top: Proposed Method. Center: reweighted GLEI. Bottom: COLS. The red circle denotes the agent position recovered by solving (6.16)
Figure 6.4: Causally interacting groups in Double Tennis. Top: Proposed Method. Center: reweighted GLEI. Bottom: COLS.
6.8 Chapter Summary

Many problems of practical interest require identifying a dynamical graphical model from input/output data. As shown in this chapter, this can be efficiently done by recasting the problem into an expanded atomic-norm minimization framework that promotes block-sparsity and allows for exploiting computationally efficient Frank-Wolfe type algorithms. Further, the proposed framework can be easily expanded to accommodate unknown exogenous inputs and missing data. These results were illustrated with several examples drawn from video-analytics, showing that the proposed method outperforms existing ones, specially as the size of the problem increases.
Chapter 7

Self Scaled Regularized Robust Regression

7.1 Introduction and Motivation

Regression analysis seeks to find the relationship between one or more independent variables and a dependent variable. The application of this model can be found in areas ranging from statistics, signal processing, machine learning to computer vision. Examples, illustrated in Fig. 7.1, include line extraction from 2D images, planar surface fitting in range images, and classification using linear discriminant analysis (LDA), among others. When all the available data are inliers, least squares regression (LSR) provides good fitting regression parameters [123]. However, it is well known that in the presence of outlier data points, i.e. data points that do not fit the sought model, LSR can result in very poor fitting models [124].

The goal of robust regression is to find good fitting models in spite of the presence of
7.1. INTRODUCTION AND MOTIVATION

outliers. Robust algorithms for linear regression include least median squares regression (LMedS) [125] and random sample consensus type methods (RANSAC) [126]. While these methods perform well, they are inherently combinatorial.

Alternative approaches exploit recent advances in compressive sensing [127, 128] by reformulating robust linear regression as an optimization problem with sparse regularizations [15, 16, 17, 18]. The advantage of these methods is that they admit convex relaxations which can be solved using polynomial-time algorithms. In [18], Mitra et al. derived conditions under which these relaxations solve the robust regression problem, which depend on the smallest principal angle between the regressor subspace and all outlier subspaces, in the case of noiseless inliers.

A drawback of the sparsity-based approaches is that the presence of a few gross outliers, outliers which are very far from the inlier data, can poison the optimization problem and lead to ill fitting models. Another limitation of the current sparsity-based methods is that they cannot accommodate a-priori semi-supervised knowledge such as co-occurrence information when it is known that a subset of points should have a single label – i.e. they are all inliers or all outliers. Thus, to address the above limitations, we propose a new formulation for robust linear regression which can handle gross outliers and a priori information.

The following points motivate the proposed work:

- We provide a new sparsity-based formulation to maximize the number of inlier data points.
- We show that this new approach is equivalent to a “self-scaled” $\ell_1$ regularized robust regression problem, where the cost function is automatically scaled and the scalings capture a-priori information. Hence, we have called the proposed method a “Self-Scaled Regularized Robust Regression” ($S^2R^3$) algorithm.
- We show that the self-scaling property of the proposed approach yields smaller fitting errors in the presence of gross outliers.
- We can incorporate a priori information by adding simple constraints to the problem.

The following part is organized as follows. Section 7.2 gives an overview of related work. In section 7.3 a new sparsity-based formulation to the robust linear regression problem is introduced. Section 7.4 presents the tightest, tractable convex relaxation of the proposed objective and investigates the relationship between the new and existing approaches. Section 7.5 describes how to incorporate prior information through the use of additional constraints in the optimization
problem. Section 7.6 summarizes the proposed algorithm and section 7.7 illustrates its performance with several application examples using synthetic and real data. Finally, section 7.8 concludes this chapter.

7.2 Literature Review

Most current robust linear regression methods can be classified into one of four major classes: median-based, M-estimators, RANSAC, and sparsity-based convex methods.

Median-based approaches try to overcome the limitations of least squares regression by using, instead of the mean, the median of the fitting errors since it is more robust to outliers. For example, LMedS \[125\] seeks to minimize the median of the squared residues using a random sampling algorithm. However, this algorithm is combinatorial on the number of regression parameters and hence is not suitable for high dimensional regression problems.

An alternative for making least squares regression robust is to use M-estimators \[124\]. In this approach, the residual error in the maximum likelihood estimate is replaced by a non-linear function of the residuals that penalizes residuals from inliers almost linearly but saturates for residual errors due to outliers. A disadvantage of this approach is that the resulting optimization problem is non-convex. Solving this problem with iterative steepest descent methods is not guaranteed to converge to the true optimum. It has also been proposed to solve this problem using random sampling methods \[129\]. However, as mentioned above, this approach suffers from its combinatorial complexity.

Perhaps the most commonly used robust regression algorithms belong to the RANSAC family \[126, 130, 131, 132, 133\]. The main idea behind these approaches is to try to minimize the number of outliers. However, these techniques, like LMedS, rely on random sampling of the data to separate the inliers from the outliers, based on fitting error, and hence are also inherently combinatorial.

More recently, inspired by the success of compressive sensing, it has been proposed to use sparsity ideas to separate outliers from inliers \[15, 16, 17, 18\]. These methods reformulate the regression problem by minimizing the least square error for the inlier data while enforcing the natural assumption that there are more inliers than outliers data points. While these new formulations are non-convex, they can be relaxed to convex optimization problems which can be solved in polynomial time. The issue of whether the solutions obtained using these relaxations are also solutions to the original regression problem was addressed in \[18\]. There, the authors computed, for the case
when there is no inlier noise, a lower bound of the maximum number of outliers that minimizing the number of outliers (such that the inliers fit the model) can handle. This bound is given by the smallest principal angle $\theta_k$ between the regressor subspace and all the $k$-dimensional outlier subspaces. However, the quality of the solutions obtained by these formulations suffers when the size of some of the outlier errors is very large.

In this chapter, a \textit{Self-Scaled Regularized Robust Regression} ($S^2R^3$) algorithm is proposed. The $S^2R^3$ algorithm belongs to the last category, as it maximizes the number of inliers through a convex relaxation. However, the main advantage of the proposed formulation is that it is not sensitive to the scale of the outlier errors. Indeed, as shown in section 7.4 the proposed formulation is equivalent to a properly scaled $\ell_1$ regularized regression. While the $S^2R^3$ method has the desirable property that the data scaling is done automatically, it also suggests that previous methods would benefit from proper scaling of the data, as well. In addition, in contrast with previous approaches, the $S^2R^3$ method can easily handle prior information such as co-occurrence labeling.

### 7.3 Statement of the problem

Given $N$ data points $x_i \in \mathbb{R}^d$, corresponding scalars $y_i$, $i = 1, \ldots, N$, a vector $r \in \mathbb{R}^d$, and a noise bound $\epsilon$, define the set of inliers as:

$$S_i(r) = \{ x_i : |y_i - x_i^T r| \leq \epsilon \} \quad (7.1)$$

The robust regression problem consists on determining a vector $r$ such that the number of inliers is maximized, that is:

$$r^* = \arg \max_r |S_i(r)| \quad (7.2)$$

By introducing additional variables $r_i \in \mathbb{R}^d$ the problem above can be reformulated as:

$$r^* = \arg \min_{r, r_i} \| \{ r - r_i \} \|_o \quad \text{s.t.} \quad |y_i - x_i^T r_i| \leq \epsilon, \ i = 1, \ldots, N \quad (7.3)$$

where $\| \{ r - r_i \} \|_o$ denotes the number of non-zero elements of the vector sequence $\{ r - r_i \}_{i=1}^N$.

**Lemma 7.** Problems (7.2) and (7.3) are equivalent.

**Proof.** Given $r$, define

$$J(r) = \min_{r_i \in \mathbb{R}^d} \| \{ r - r_i \} \|_o \quad \text{s.t.} \quad |y_i - x_i^T r_i| \leq \epsilon, \ i = 1, \ldots, N \quad (7.4)$$
Since $\|r - r_i\|_0 = 0 \iff r = r_i \iff x_i \in S_i(r)$, it follows that $\|\{r - r_i\}\|_0 =$ number of outliers, or equivalently, $|S_i(r)| = N - J(r)$. Thus $r^*$ maximizes $|S_i(r)|$, and hence it is a solution to (7.2), if and only if it is a minimizer of $J(r)$.

Note that the solution to problem (7.3) may not be unique. Conditions guaranteeing uniqueness and exact recovery are discussed below.

**Theorem 8.** Let $X(i, :) = x_i^T$ and denote by $I_k$ the index set of subsets $S_k \subseteq S \doteq \{1, \ldots, N\}$, with $|S_k| = k$. Then, in the noiseless case, if the matrix $[X I]$ has full column rank for all $I \in I_k$ and Problem (7.3) admits a solution with $\|\{r - r_i\}\|_0 < \frac{k}{2}$, the model $r$ is unique.

**Proof.** Define $s_i = x_i^T (r_i - r)$ and consider the following related problem:

$$\min_{r, s} \|s\|_0 \quad \text{s.t.} \quad y_i = x_i^T r + s_i, \ i = 1, \ldots, N \quad (7.5)$$

Note that in the noiseless case, (7.5) and (7.4) have the same constraint set. Since $\|s\|_0 \leq \|\{r - r_i\}\|_0$, it follows from the hypothesis and Proposition II.1 in [18], that if (7.4) admits a $m$-sparse solution, with $m < \frac{k}{2}$ then the solution to (7.5) is unique. To finish the proof, assume by contradiction that (7.4) admits multiple $m$-sparse solutions with different $r$. Then, the corresponding vectors $r$ and $s$ solve (7.5), contradicting the fact that this problem has a unique solution.

### 7.4 Main Results

While the results in section 7.3 guarantee exact recovery of the model under some conditions, they require solving problem (7.3), which can be easily shown to be generically NP-hard. In this section we present a tractable convex relaxation and investigate its relationship with existing approaches.

#### 7.4.1 Self-Scaled Regularized Robust Regression

Recall that the convex envelope (that is the tightest convex approximation) of the cardinality of a vector sequence $\{v_i\}$ [118] is given by:

$$\|\{v\}\|_{0, env} = \sum_i \|v_i\|_\infty \quad (7.6)$$
It follows that replacing $\|\{r - r_i\}\|_0$ by $\sum_{i=1}^{N} \|r - r_i\|_\infty$ provides the tightest convex approximation to the objective function, motivating the following convex relaxation of (7.3)

$$\min_{r,r_i} \sum_{i=1}^{N} \|r - r_i\|_\infty$$

s.t. $|y_i - x_i^T r_i| \leq \epsilon, \ i = 1, \ldots, N$ \hspace{1cm} (7.7)

As we show next, in the absence of additional constraints, the problem above is equivalent to a suitable scaled traditional $\ell_1$-regularized robust regression. Thus, in the sequel we will refer to problem (7.7) as the self-scaled regularized robust regression problem ($S^2R^3$).

**Theorem 9.** Problem (7.7) is equivalent to the following optimization problem:

$$\min_{r,\eta} \sum_{i=1}^{N} \frac{|y_i - x_i^T r + \eta_i|}{\|x_i\|_1}$$

s.t. $|\eta_i| \leq \epsilon, \ i = 1, \ldots, N$ \hspace{1cm} (7.8)

**Proof.** Rewriting the constraints in (7.7) as $y_i = x_i^T r_i + \eta_i$ for some $|\eta_i| \leq \epsilon$ leads to

$$y_i = x_i^T (r_i - r) + x_i^T r + \eta_i$$

Thus

$$|x_i^T (r - r_i)| = |x_i^T r + \eta_i - y_i|$$ \hspace{1cm} (7.9)

Since the $\ell_1$ and $\ell_\infty$ norms are dual [134], from the equation above it follows that

$$\|x_i\|_1 \|r - r_i\|_\infty \geq |x_i^T r + \eta_i - y_i| \Rightarrow \|r - r_i\|_\infty \geq \frac{|x_i^T r + \eta_i - y_i|}{\|x_i\|_1}$$ \hspace{1cm} (7.10)

(with equality holding when $x$ and $r - r_i$ are aligned). For fixed $y_i, r, \eta_i$, consider now the following minimization problem:

$$\min_{r_i} \|r - r_i\|_\infty$$

s.t. $x_i^T r_i + \eta_i - y_i = 0$ \hspace{1cm} (7.11)

We claim that the solution to this problem is given (component-wise) by

$$\tilde{r}_i^{(j)} = r^{(j)} - \frac{x_i^T r - y_i + \eta_i}{\|x_i\|_1} sign(x_i^{(j)})$$ \hspace{1cm} (7.12)

To show this, note that $\tilde{r}_i$ is a feasible solution of (7.11), and such that each component of the difference vector $r - \tilde{r}_i$ satisfies:

$$|r^{(j)} - \tilde{r}_i^{(j)}| = \frac{|x_i^T r - y_i + \eta_i|}{\|x_i\|_1}, \ j = 1, \ldots, d$$
and hence
\[ \| \mathbf{r} - \tilde{r}_i \|_\infty = \frac{|x_i^T \mathbf{r} - y_i + \eta_i|}{\| x_i \|_1}, \quad j = 1, \ldots, d \] (7.13)
since, from (7.10), this is the lowest possible value of the objective, optimality of \( \tilde{r}_i \) follows.

Replacing each term in the objective function in (7.7) by its optimal value leads to:
\[
\{ \min_{\mathbf{r}, \eta_i} \sum_{i=1}^{N} \| \mathbf{r} - \mathbf{r}_i \|_\infty \mid y_i = x_i^T \mathbf{r}_i + \eta_i, \quad |\eta_i| \leq \epsilon \} = \{ \min_{\mathbf{r}} \sum_{i=1}^{N} \frac{|x_i^T \mathbf{r} + \eta_i - y_i|}{\| x_i \|_1} \mid |\eta_i| \leq \epsilon \} \quad (7.14)
\]

### 7.4.2 Connections with regularized \( \ell_1 \) robust regression

By introducing an outlier error vector \( s \), problem (7.2) can be reformulated as:
\[
\min_{\mathbf{r}, \eta, s} \| s \|_o \\
\text{s.t.} \quad \mathbf{y} = \mathbf{X} \mathbf{r} + \eta + \mathbf{s} \\
\| \eta \|_\infty \leq \epsilon
\] (7.15)
where \( \mathbf{X}(i,:) = x_i^T \). Since this problem is known to be NP hard, a convex relaxation can be obtained by using the \( \ell_1 \) norm as surrogate for cardinality, leading to the \( \ell_1 \) regularized robust regression problem introduced in [18].
\[
\min_{\mathbf{r}, \eta} \| s \|_1 \\
\text{s.t.} \quad \mathbf{y} = \mathbf{X} \mathbf{r} + \eta + \mathbf{s} \\
\| \eta \|_\infty \leq \epsilon
\] (7.16)
From (7.8), (7.16) and Theorem 9, it follows that, in the unconstrained case, (7.7) can be considered as a scaled version of (7.16), where each data point is automatically scaled by its \( \ell_1 \) norm. As we will illustrate with several examples, this scaling prevents small groups of outliers, far from the inlier manifold, from “poisoning” the optimization, hence leading to better fitting.

### 7.4.3 Exact Recovery Conditions and Bounds on the Estimation Error

From Theorem 9, it follows that the results in [18] can be directly applied to establish bounds on the norm of the difference between the solutions to (7.3) and its convex relaxation (7.7). To this effect, begin by defining the normalized data matrix \( \mathbf{X}_n \), with rows given by \( \mathbf{X}_n(i,:) = \frac{x_i^T}{\| x_i \|_1} \).

Next, perform a reduced QR decomposition \( \mathbf{X}_n = \mathbf{Q} \mathbf{R} \), where \( \mathbf{Q} \) is orthonormal and \( \mathbf{R} \) is upper diagonal, and define \( \mathbf{z} = \mathbf{R} \mathbf{r} \). Proceeding as in [18], we will first find the estimation error \( \Delta \mathbf{z} \) and...
7.5. INCORPORATING PRIORS

In many situations of practical interest, additional a-priori information is available that can be exploited to improve performance. In the sequel, we illustrate the ability of the proposed algorithm to exploit priors, using two commonly occurring scenarios.

7.5.1 Using co-occurrence information

Consider the case where it is known that certain sets of points should have the same label. An example of this situation arises for instance in motion-based segmentation problems, where often
it is known that a group of points belongs to either the target or the background. As we show in the sequel, this information can be easily incorporated as additional constraints in the formulation (7.7). On the other hand, traditional $\ell_1$ regularized regression cannot exploit this information, since the problem is formulated in terms of error indicator variables $s_i$, rather than candidate model parameters $r_i$. Specifically, let $I$ denote the set of indices of points $x_i$ that should have the same label and denote by $X_I$ and $y_I$ the sub matrix of $X$ formed by considering only the rows indexed by elements of $I$, and the vector formed by the corresponding elements of $y$, respectively. Consider first the noiseless case and assume that $y_I \in \text{span-col}(X_I)$ and $\text{rank}(X_I) \leq d^1$. Under these conditions, there exist at least one $r^*$ such that $y_I = X_I r^*$. Thus adding the constraints $r_i = r_I \forall i \in I$ (enforced by simply using the same variable $r_I$ in all terms in (7.7) involving elements of $I$, does not change the optimal solution. This follows from the fact that $r_I$ can be set to $r$ if the points indexed by $I$ are inliers, or to $r^*$ if they are outliers, without changing the value of the objective. In the case of noisy data, the same reasoning can be applied as long as there exists some vector $\eta_I$, with $\|\eta_I\|_\infty \leq \epsilon$ such that $y_I - \eta_I \in \text{span-col}(X_I)$. As before, this condition holds trivially as long as $|I| \leq d$.

### 7.5.2 Non-full rank data

Conventional robust regression typically considers the case where $X$ is full rank. However, this assumption does not always hold in practice. Indeed, several practical problems involve considering the case where $y_i = 0$, and hence, if non-trivial solutions to (7.7) exist, they are not unique. An example of this situation is the problem of estimating the fundamental matrix \cite{135}, where the solution is unique up to a scaling factor. In these cases, in order to avoid ambiguities, it is of interest to impose additional constraints on $r$. One such class of constraints is of the form $u^T r = 1$, for some suitable chosen $u$. For instance, the choice $u = 1$ leads to the constraint $\sum r(j) = 1$, while $u = e_i$ corresponds to $r(i) = 1$, both used in the context of fundamental matrix estimation.

In this case, it can be shown, by computing sub gradients, that the optimal solutions to (7.7) for the case $\sum r_i(j) = 1$ is:

$$
\tilde{r}_i(j) = r(j) - (x_i^T r + \eta_i - y_i) \frac{\text{sign}(x_i(j)x_j - \text{median})}{\|x_i - \text{median}\|_1}
$$

with associated cost

$$
\|\tilde{r}_i - r\|_\infty = \frac{|x_i^T r + \eta_i - y_i|}{\|x_i - \text{median}\|_1} \quad (7.19)
$$

\(^1\text{This situation holds trivially when } |I| \leq d\).
leading again to a modified regularized $\ell_1$ regression, where each term is now scaled by the factor $\|x_i - \bar{x}_{\text{median}}\|_1$.

### 7.6 Self Scaled Regularized Regression Algorithm

Theorem 8 provides sufficient conditions guaranteeing that solving (7.7) will lead to the sparsest $\{r - r_i\}$ sequence and hence result in a model that maximizes the number of inliers. However, in many practical situations these conditions may not hold. In these cases, sparse solutions can be obtained by using a reweighted heuristic [136], leading to the following algorithm\(^2\).

#### Algorithm 9 Reweighted Self Scaled Regression

1. **Initialize:** $w^0 \leftarrow \{1, \ldots, 1\}$, $\tau = 10^{-2}$
2. **repeat**
3. **Solve**
   \[
   \{r^k, r^k_i\} = \arg \min_{r, r_i} \sum_{i=1}^{N} w^k(i) \|r - r_i\|_\infty \quad \text{s.t.} \quad |y_i - x_i^T r_i| \leq \epsilon, \quad i = 1, \ldots, N
   \]
4. **Update** $w^{k+1}(i) = (\|r^k - r^k_i\|_\infty + \tau)^{-1}$.
5. **until** convergence

### 7.7 Experimental Results

In this section we describe two sets of experiments to evaluate the performance of the proposed algorithm. The first set of experiments uses synthetic data to fit a hyperplane while the second set uses real data to reconstruct corrupted face images from the Yale face dataset. In all cases, performance is compared against 8 existing regression methods that range from classic techniques using random approaches to state of the art convex formulations.

**Randomized Algorithms.** The methods using random approaches we compared against are: RANSAC, MSAC, MLEASAC and LMEDS. To ensure that comparisons are fair, i.e. all the methods are solving a robust regression problem, these techniques were used to solve the problem

---

\(^2\)A similar algorithm was proposed in [118] in the context of systems identification, but without an analysis of its recovery properties or relationship to traditional $\ell^1$ regularized robust regression.
The most critical parameters for randomized algorithms are the inlier noise bound and the number of iterations. We set the inlier noise bound for all randomized algorithms equal to the inlier noise bound of convex formulations. In other words, all algorithms shared same inlier noise bound. The number of iterations was set to 500 for all noise levels and for all algorithms. We used the implementation from GML Toolbox from [137].

**M-estimator.** M-estimator is a standard robust regression method. We used the MATLAB implementation with “huber” weighting function, which is the common setup for it. We found the best parameter for M-estimator by line search in one dataset and use it for the corresponding experiment.

**Constrained RPCA:** Robust PCA is a recent robust regression method proposed by [138]. We modified the original formulation inline with [139]. In our formulation we find the smallest penalty parameter that gives a rank deficient data matrix. This step removes both inliers and outliers of the data and makes sure it has a null space of dimension 1. Then, we use the null space vector as the final model.

**RR [17]:** This formulation is similar to RPCA formulations with the inclusion of the model term in the optimization function. Their extended formulation is bilinear and solved with ALM. We implemented our own version following their supplementary material. This algorithm has 2 parameters which are difficult to tune. We used grid search to find the best setup given a dataset from each experiment.

**BPRR and BSRR [18]:** These are the most recent formulations for robust regression and they are the closest works to ours. We implemented them using CVX Toolbox [140]. The only parameter these formulations require is the inlier noise bound (same as ours).

### 7.7.1 Synthetic Data Experiments

This set of experiments attempts to recover hyperplanes from data corrupted by outliers. The data was generated as follows. First, a vector \( \mathbf{r} \) was drawn using a Normal distribution \( N(0, I) \). Then, the input samples \( \mathbf{x}_i \) were uniformly sampled from \([0, 1]^{m-1}\), where \( m = 5 \) is the dimension of the data. Next, the outputs \( y_i \) were computed as \( y_i = \mathbf{x}_i \mathbf{r} + e_i \), with \( e_i \) uniformly distributed from \([-\epsilon, \epsilon]\), where \( \epsilon = 0.1 \). Finally, the outliers were seeded by randomly sampling \( y_i \) and \( \mathbf{x}_i \) from \( N(0, 15) \) and \( N(0, 1) \), respectively.
In all the experiments, the inlier noise bound was set to the value used to generate the data. The number of outliers was varied from 10% to 90%, in increments of 10%. The algorithm was run 100 times for each level of outliers. Performance was compared using two performance scores: geometric mean of precision and recall, and the regression recovery error.

The results of this set of experiments can be seen in Figures 7.2 and 7.3 and the running times are given in Table 7.1. Note that our algorithm performs the best, both with low and high percentages of outliers. On the other hand, randomized algorithms show a significant performance drop when the percentage of outliers is 70% and above, showing the advantage of our formulation. Furthermore, it should be noted that not all the convex formulations have similar robustness under heavy outlier noise. In particular, the early failure of the BPRR algorithm illustrates the importance of the self-scaling property of the proposed approach.

**Using Priors.** To evaluate the impact of using priors we proceeded as follows. After a run without priors was done, no more than half of the false positive points were paired randomly with a true negative point, and no more than half of the false negative points were paired randomly to a true point. As seen in Figure 7.2 and 7.3, the ability to incorporate the additional co-occurrence information can boost the performance of the proposed algorithm between 5 to 10 percent under heavy outlier noise.
CHAPTER 7. SELF SCALED REGULARIZED ROBUST REGRESSION

Figure 7.3: Synthetic Data Experiments: Fitting a 5-dimensional hyperplane. The plots show the Model Error for all the evaluated algorithms.

<table>
<thead>
<tr>
<th>Method</th>
<th>Implementation</th>
<th>Times</th>
</tr>
</thead>
<tbody>
<tr>
<td>proposed</td>
<td>Gurobi(LP)</td>
<td>0.0266</td>
</tr>
<tr>
<td>RANSAC</td>
<td>MATLAB</td>
<td>0.0491</td>
</tr>
<tr>
<td>MSAC</td>
<td>MATLAB</td>
<td>0.0495</td>
</tr>
<tr>
<td>MLESAC</td>
<td>MATLAB</td>
<td>0.1282</td>
</tr>
<tr>
<td>LMeds</td>
<td>MATLAB</td>
<td>0.0580</td>
</tr>
<tr>
<td>M-estimator</td>
<td>MATLAB</td>
<td>0.0080</td>
</tr>
<tr>
<td>cRPCA</td>
<td>MATLAB(ADMM)</td>
<td>2.5346</td>
</tr>
<tr>
<td>BPRR</td>
<td>CVX</td>
<td>0.9859</td>
</tr>
<tr>
<td>RR</td>
<td>MATLAB(ADMM)</td>
<td>1.9388</td>
</tr>
<tr>
<td>BSRR</td>
<td>MATLAB</td>
<td>0.7307</td>
</tr>
</tbody>
</table>

Table 7.1: Running times for the experiments with synthetic data.

7.7.2 Real Data Experiments

This set of experiments attempts to reconstruct face images that have been corrupted with heavy occlusion, where the occluding pixels constitute the outliers. The data used for these
7.7. EXPERIMENTAL RESULTS

experiments is from the CroppedYale Dataset [141]. The dataset contains 38 subjects. We choose 8 face images per person, taken under mild illumination conditions and computed an eigenface set with 20 eigenfaces. Then, the goal of these experiments was: given a corrupted face image of a subject in the database (this (uncorrupted) image was not used to compute the eigenspace), get the best reconstruction/approximation of the true face image.

We reconstructed one image per person. Occlusion was simulated by randomly placing 10 blocks of size $30 \times 30$. To increase the difficulty of the problem and reduce the dimensionality, data was randomly sampled (400 pixels from the image and the basis). The performance of the algorithms was evaluated using the Root Mean Square metric (Table 7.2),

$$RMS(I, \hat{I}) = \sqrt{\frac{\|I - \hat{I}\|_F^2}{N_{\text{pixels}}}}$$

where $I$ is the original image without occlusion and $\hat{I}$ is the reconstructed image. A visual comparison for one instance of recovery using all the evaluated methods is shown in Figure 7.4. We normalized all images to $[0, 1]$ range to remove scaling effects of the pixel values on the RMS metric. We also computed a best possible reconstruction of the original face image by using the 20 eigenfaces. We used the model of this step as the ground truth model and computed the model recovery error as in the synthetic experiments (Table 7.3). The experiments show that the mean RMS and the model error are the best for our method and that the recovered images are visually closer to the un-occluded original image.

<table>
<thead>
<tr>
<th></th>
<th>proposed</th>
<th>BPRR</th>
<th>BSRR</th>
<th>M-est.</th>
<th>RR</th>
<th>cRPCA</th>
<th>MLESAC</th>
<th>MSAC</th>
<th>RANSAC</th>
<th>LMedS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean RMS</td>
<td>0.1320</td>
<td>0.1397</td>
<td>0.1378</td>
<td>0.1345</td>
<td>0.1844</td>
<td>0.1854</td>
<td>0.1751</td>
<td>0.1773</td>
<td>0.1690</td>
<td>0.1835</td>
</tr>
<tr>
<td>stdev</td>
<td>0.0074</td>
<td>0.0052</td>
<td>0.0081</td>
<td>0.0074</td>
<td>0.0054</td>
<td>0.0071</td>
<td>0.0082</td>
<td>0.0067</td>
<td>0.0064</td>
<td>0.0085</td>
</tr>
</tbody>
</table>

Table 7.2: Fitting to original image error.

Finally, we ran another set of experiments where we gave all the *SAC algorithms (RANSAC, MSAC, MLESAC, LMedS) some extra time. For these experiments, we set the number of iterations so that these algorithms could use as much time or longer than the time used by the

<table>
<thead>
<tr>
<th></th>
<th>proposed</th>
<th>BPRR</th>
<th>BSRR</th>
<th>M-est.</th>
<th>RR</th>
<th>cRPCA</th>
<th>MLESAC</th>
<th>MSAC</th>
<th>RANSAC</th>
<th>LMedS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean RMS</td>
<td>0.7105</td>
<td>0.9092</td>
<td>0.7428</td>
<td>0.7232</td>
<td>1.0663</td>
<td>1.0761</td>
<td>1.0682</td>
<td>1.0917</td>
<td>1.1013</td>
<td>1.0528</td>
</tr>
<tr>
<td>stdev</td>
<td>0.0382</td>
<td>0.0472</td>
<td>0.0533</td>
<td>0.0435</td>
<td>0.0441</td>
<td>0.0506</td>
<td>0.0329</td>
<td>0.0337</td>
<td>0.0338</td>
<td>0.0353</td>
</tr>
</tbody>
</table>

Table 7.3: Model estimation error.
CHAPTER 7. SELF SCALED REGULARIZED ROBUST REGRESSION

Table 7.4: Fitting to original image error (allowing extra time to the *SAC algorithms).

<table>
<thead>
<tr>
<th></th>
<th>proposed</th>
<th>BPRR</th>
<th>BSRR</th>
<th>M-est.</th>
<th>RR</th>
<th>cRPCA</th>
<th>MLESAC</th>
<th>MSAC</th>
<th>RANSAC</th>
<th>LMedS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean RMS</td>
<td>0.1320</td>
<td>0.1397</td>
<td>0.1378</td>
<td>0.1345</td>
<td>0.1844</td>
<td>0.1854</td>
<td>0.1704</td>
<td>0.1545</td>
<td>0.1588</td>
<td>0.1661</td>
</tr>
<tr>
<td>stdev</td>
<td>0.0074</td>
<td>0.0052</td>
<td>0.0081</td>
<td>0.0074</td>
<td>0.0054</td>
<td>0.0071</td>
<td>0.0069</td>
<td>0.0064</td>
<td>0.0074</td>
<td>0.0082</td>
</tr>
<tr>
<td>run time</td>
<td>1.5088</td>
<td>1.6553</td>
<td>51.1901</td>
<td>0.0343</td>
<td>19.5540</td>
<td>0.3533</td>
<td>3.3083</td>
<td>1.5997</td>
<td>1.5864</td>
<td>1.7923</td>
</tr>
</tbody>
</table>

Table 7.5: Model estimation error (allowing extra time to the *SAC algorithms).

<table>
<thead>
<tr>
<th></th>
<th>proposed</th>
<th>BPRR</th>
<th>BSRR</th>
<th>M-est.</th>
<th>RR</th>
<th>cRPCA</th>
<th>MLESAC</th>
<th>MSAC</th>
<th>RANSAC</th>
<th>LMedS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean RMS</td>
<td>0.7105</td>
<td>0.9092</td>
<td>0.7428</td>
<td>0.7232</td>
<td>1.0663</td>
<td>1.0761</td>
<td>0.8719</td>
<td>0.9098</td>
<td>0.9045</td>
<td>0.9183</td>
</tr>
<tr>
<td>stdev</td>
<td>0.0382</td>
<td>0.0472</td>
<td>0.0533</td>
<td>0.0435</td>
<td>0.0441</td>
<td>0.0506</td>
<td>0.0392</td>
<td>0.0366</td>
<td>0.0305</td>
<td>0.0369</td>
</tr>
<tr>
<td>run time</td>
<td>1.5088</td>
<td>1.6553</td>
<td>51.1901</td>
<td>0.0343</td>
<td>19.5540</td>
<td>0.3533</td>
<td>3.3083</td>
<td>1.5997</td>
<td>1.5864</td>
<td>1.7923</td>
</tr>
</tbody>
</table>

proposed algorithm. While the extra time improved the performance of the *SAC algorithms it was not enough to achieve the best performance, as summarized in Tables 7.4 and 7.5.

7.8 Chapter Summary

Robust regression is at the core of a large number of machine learning and computer vision problems ranging from recovering 3D geometry, to classification and image reconstruction. While this problem has been the object of a very large research effort, it remains challenging in scenarios characterized by noisy correspondences and high percentage of gross outliers. The main result of this chapter is a computationally tractable regression algorithm specifically tailored to this situation. Contrary to other sparsification based approaches, the proposed algorithm seeks to directly sparsify the set of models that explain the data, rather than the set of outlier errors. The intuition behind this approach is that this set of models can be normalized so that all its elements have comparable magnitude, a fact that prevents gross outliers from skewing the results. Surprisingly, as shown in the paper, the proposed approach is equivalent to a self-scaled robust regression, where the data points are automatically scaled by a problem dependent quantity, providing an alternative explanation of the reason behind its improved performance in the presence of gross outliers. In addition, working directly with models (rather than outlier errors) allows for exploiting existing a-priori information about co-occurrences to improve the resulting model, a feature hitherto beyond the ability of existing regression techniques. As shown in the experiments, the combination of self-scaling and the ability to exploit priors allows the proposed algorithm to consistently outperform
Figure 7.4: Face recovery results: In order from left to right, top to bottom: original image, occluded image, best possible recovery with given basis, proposed, BPRR, cRPCA, LMedS, Mestimator, MLESAC, MSAC, RANSAC, and RR.

existing techniques, regardless of the percentage of outliers.
Chapter 8

Geometry-aware GARCH Model
Identification and Optimal Filtering on
the Manifold of Positive Definite
Matrices

Figure 8.1: Two examples where covariance features are used to describe a target. On the left, the appearance of the target car has roughly constant covariance. On the right, the covariance of the appearance of the spinning ball changes over time.

8.1 Previous Works and Motivation

Covariance matrices, as a way of capturing stochastic information from large number of data samples, are ubiquitous in computer vision, in problems ranging from tracking [19, 20, 21, 22, 23, 24, 25, 26, 27] to object detection [28, 29], person re-identification [30], activity recognition
8.1. PREVIOUS WORKS AND MOTIVATION

[31], face recognition[32] and Diffusion Tensor Imaging (DTI) [33, 34]. Applications outside the computer vision field include economics [35], fault detection [36] and power systems [37].

Most of these applications require estimating the present value of a covariance matrix from a combination of noisy measurements and past historical data, with the main difficulty here arising from the need to account for the fact that these matrices evolve on a Riemannian manifold. For example, [19] proposed to use as covariance estimate the Karcher mean of the measurements, as a counterpart to the use of the arithmetic mean update in Euclidean space. However, this approach does not take into consideration measurement noise. Explicitly accounting for this noise leads to recursive filtering methods. In this context, [20] considered linear systems evolving on a Riemannian manifold and proposed a Kalman recursive scheme where a matrix log mapping (given a so called base point), is used to flatten the PD manifold prior to computing the predicted and corrected states. However, it is known that flattening the manifold often leads to less accurate distance calculation, resulting in poor prediction/estimation. An intrinsic extension of recursive filtering has been introduced in [27], where the on-manifold distance is considered. A limitation of this approach is that it assumes that the present value of the covariance evolves according to a known first order model (that is, the present value of the covariance depends only on its most immediate past value). However, these assumptions do not necessarily hold in many practical scenarios where covariances evolve according to more complex dynamics that are not a-priori known (see Figure 8.1).

To address these limitations, in this chapter we propose a new framework for recursive filtering on the PD manifold using Generalized Autoregressive Conditional Heteroskedasticity (GARCH) models for propagating past measurements, combined with a maximum likelihood estimator based on minimizing the Jensen Bregman LogDet divergence[26]. Specifically, the contributions of this work are as follows:

1. We propose a new GARCH model to propagate a sequence of positive definite matrices and show that the parameters of this model can be identified from experimental data by solving a tractable convex optimization problem.

2. We introduce a new probabilistic dynamic model for recursive filtering on the PD manifold based on a generalized Gaussian distribution. As shown in the chapter, under suitable conditions, the generalized Gaussian conjugate prior can indeed be expressed in terms of the JBLD distance between the observed and predicted data. This key observation leads to a filter that admits a closed-form solution and compares favorably, both in terms of the estimation error and computational time, against existing approaches.
CHAPTER 8. GEOMETRY-AWARE OPTIMAL FILTER ON PD MANIFOLD

8.2 Preliminaries

8.2.1 Metrics in $\mathbb{S}^n_{+++}$

The positive definite matrices form a convex cone in Euclidean space. However, it has been shown that the metrics that do not take into account the geometry of the $\mathbb{S}^n_{+++}$ manifold have poor accuracy in practical applications\[142, 143]. As mentioned in [144], symmetric matrices with nonpositive eigenvalues are at finite distance from any PD matrix. Moreover, the Euclidean averaging of PD matrices often leads to a swelling effect, i.e. the determinant of the Euclidean mean can be strictly larger than the original determinants. This is not acceptable when considering the (PD) covariance matrices, which is the case in this chapter.

In order to take into account the non-flat manifold geometry of $\mathbb{S}^n_{+++}$, an approximation to the geodesic distance can be obtained by using the matrix log operator to project PD matrices into a plane tangent to the Riemannian manifold. Then, the Frobenius norm of the difference between projections can be used as a surrogate for the geodesic distance, leading to the Log-Euclidean Riemannian Metric (LERM) \[142\]. As noted above, a potential pitfall of this approach is that flattening the PD manifold often leads to less accurate distance computation, which results in poor predictions. A full blown, popularly used manifold metric is the the Affine Invariant Riemannian Metric (AIRM) \[33, 145\]. This approach uses the geodesic length along the manifold curvature leading to the distance measure:

$$J_R(X, Y) = \| \log \left( X^{-\frac{1}{2}} Y X^{-\frac{1}{2}} \right) \|_F$$

The main disadvantages of this metric are the high computational burden entailed in computing it, and for the specific application in this paper, the lack of convexity\(^1\). As an alternative, during the past few years, a family of metrics originating in the Jeffrey’s Kullback-Leibler divergence, which measures the distance between distributions, has been investigated. The idea behind this approach is to consider PD matrices as covariances of zero-mean Gaussian distributions.

Recently, a computationally efficient and empirically effective metric, the Jensen-Bregman LogDet Divergence, was proposed in [26]

$$J_{ld}(X, Y) \triangleq \log \left| \frac{X + Y}{2} \right| - \frac{1}{2} \log |XY|$$

\(^1\)Convexity in Euclidean sense, which gives access to efficient convex optimization tools with well-developed theoretical support, i.e. ADMM.
As noted in [146] the JBLD is \textit{geometry aware} and has been proven to be \textit{non-flat} [147, 148]. Furthermore, JBLD and AIRM are closely related in how they measure geodesic length (see Theorem 1 in [146]).

From the standpoint of this work, \( J_{ld} \) offers several advantages over the more traditional \( J_{R} \), as noted below.

1. - \( J_{ld}(X, Y)^{\frac{1}{2}} \) is a metric[149].

2. - If \( 0 < \sigma I \leq X, Y \leq \sigma I \) then \( J_{ld}(X, Y) \leq J_{R}^{2}(X, Y) \leq (2 \log \frac{\sigma}{\sigma}) (J_{ld}(X, Y) + n \log 2) \) [26].

3. - For a fixed \( Y \in \mathbb{S}_{++}^{n} \), \( J_{ld}(X, Y) \) is convex in the region \( \{ X \in \mathbb{S}_{++}^{n} : X \leq (1 + \sqrt{2})Y \} \) [26].

\textbf{Remark 8.} The properties above, along with the empirically observed fact that \( \sqrt{J_{ld}} \) is a good proxy for \( J_{R} \) [149], motivate its use in this work.

### 8.2.2 Inverse Wishart Distributions and GARCH models

The models that we use in this work originate in the conjugate prior distribution of multivariate Gaussian sampling. Recall (see for instance [150]) that the likelihood function of the covariance matrix \( P \) of \( n \) independent observations \( x_{i} \sim \mathcal{N}(\mu, P) \) is given by the inverse Wishart distribution, that is:

\[
L(P, Q) \propto |P^{-1}|^{\frac{n-1}{2}} e^{-\frac{\text{tr}(P^{-1}Q)}{2}}
\]

(8.2)

where \( Q \) denotes the empirical covariance, e.g. \( Q = \sum_{i=1}^{n} (x_{i} - \bar{x})(x_{i} - \bar{x})^{T} \) and \( \bar{x} = \frac{\sum_{i=1}^{n} x_{i}}{n} \).

In the problems of interest in this work, we also need to model the evolution of \( P \) as a function of time. Motivated by the models commonly used to propagate the parameters of Wishart distributions [35], we propose a GARCH model of the form:

\[
p(P_{t}|P_{t-1}, \ldots, P_{t-r}) \propto e^{-\frac{J_{ld}(P_{t}, \sum_{i=1}^{r} S_{t-i} A_{i}, S_{t-i})}{2 \omega^{2}}}
\]

(8.3)

where \( S_{t-i} = P_{t-i}^{\frac{3}{2}} \), \( r \) denotes the system order and where \( A_{i} \geq 0 \) are the parameters that define the autoregressive model. Intuitively, the probability of obtaining a given covariance at time \( t \) decays exponentially with its distance, measured in the \( J_{ld} \) sense, from the predictions of the model \( \sum_{i=1}^{r} S_{t-i} A_{i} S_{t-i} \). The effectiveness of this model in capturing the dynamics governing the evolution of matrices in \( \mathbb{S}_{++} \) will be demonstrated in Section 8.5 with several examples.
Remark 9. Note that (8.3) can be indeed considered a generalization of multivariate stochastic volatility models, commonly used in econometrics to propagate covariances, to the case where the present value of the covariance depends on several past values. Specifically, under suitable conditions (see Theorem 11 in section 8.4.1), $P_t$ in (8.3) has an Inverse Wishart distribution with parameter $Q = \sum_{i=1}^{r} S_{t-i} A_i S_{t-i}$, which in the case $r = 1$ coincides with the WIC model proposed in [35].

Remark 10. The proposed model includes as a special case the simpler scalar model when $A_i = a_i I$ and $a_i$ is a scalar. On the other hand, allowing the use of matrices allows modeling more complex sequences as illustrated by the next simple example. Consider a (periodic) covariance sequence,

$$
\begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}, \begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix}, \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}, \begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix}, \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}, \begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix}, \ldots
$$

The corresponding GARCH model is given by:

$$
P_t = S_{t-2}^T \begin{bmatrix}
1 & 0 \\
0 & 0
\end{bmatrix} S_{t-2} + S_{t-3}^T \begin{bmatrix}
0 & 0 \\
0 & 1
\end{bmatrix} S_{t-3}
$$

which cannot be expressed as a scalar linear combination of $P_{t-1}$, $P_{t-2}$ and $P_{t-3}$.

8.2.3 Problem Statement

In the context of the discussion in Sections 8.2.1 and 8.2.2 the problem of interest in this chapter can be stated as:

Problem 8. Given a noisy observation $Q_t$ of a covariance matrix $P_t$ and past historical data $P_{t-r}$, find the JBLD-based Maximum Likelihood estimation of the parameters $P_t$.

We propose to solve this problem by splitting it into two subproblems, (i) estimating the propagation model parameters from training data and (ii) finding a maximum likelihood estimator of $P_t$ assuming that the propagation model is known. Formally, this leads to the following two problems:

Problem 9. Given a sequence of training data $\{P_t\}_{t=1}^{T} \in S_{++}^n$, find the JBLD-based Maximum Likelihood estimation of the parameters $A_i$, such that the dynamic model is stable.

Problem 10. Given a noisy observation $Q_t$, find the JBLD-based Maximum Likelihood estimator of $P_t$ assuming a known propagation model of the form (8.3).
8.3 Estimation of the GARCH model

Since the right hand side of (8.3) does not define a positive definite kernel for all \( \omega \) (or equivalently, the JBLD cannot be isometrically embedded in a Hilbert space unless restricted only to commuting matrices [147]), it follows that the problem of estimating the propagation model for \( P_t \) cannot be solved by mapping the data to a Hilbert space and using classical, Euclidean geometry based techniques there. Nevertheless, as we show in this section, Problem 9 can be reduced to a convex optimization and efficiently solved by exploiting the properties of the JBLD.

Given a sequence of training data \( \{ P_t \}_{t=1}^T \), estimating the model parameters in (8.3) is equivalent to solving the following Maximum Likelihood problem:

\[
\max_{A_i} \prod_{t=r}^T p(P_t|P_{t-1}, \ldots, P_{t-r}) \\
\text{s.t. } A_i \succ 0, \quad \forall i = 1, \ldots, r \\
\quad \sum_{i=1}^r S_{t-i} A_i S_{t-i} \preceq (1 + \sqrt{2}) P_t \\
\quad \| \sum_i A_i \|_2 \leq 1
\]  

(8.4)

where the second constraint enforces that the prediction should be not too far from the training data, and where the last constraint has been added to enforce stability of the resulting model. Using (8.3) this problem reduces to:

\[
\min_{A_i} \sum_{t=r}^T J_{ld}(P_t, \sum_{i=1}^r S_{t-i} A_i S_{t-i}) \\
\text{s.t. } A_i \succ 0, \quad \forall i = 1, \ldots, r \\
\quad \sum_{i=1}^r S_{t-i} A_i S_{t-i} \preceq (1 + \sqrt{2}) P_t \\
\quad \| \sum_i A_i \|_2 \leq 1
\]  

(8.5)

Since \( J_{ld}(X, Y) \) is convex with respect to \( X \) in the region \( X \preceq (1 + \sqrt{2}) Y \) [26] it follows that, as long as the problem is feasible, then it is convex and can be solved using for instance first order, ADMM type methods. Further, by using a splitting-variable type argument, it can be shown that in this case all intermediate steps in the ADMM method admit a closed-form solution and hence the overall algorithm is computationally very efficient.
As we show next, this problem can be efficiently solved using ADMM based methods. To this effect, the first step is to use a splitting-variable argument to rewrite (8.5) as:

\[
\min_{A_i, B_i, \hat{P}_t} \sum_t J_{ld}(P_t, \hat{P}_t) \\
\text{s.t.} \quad \hat{P}_t = \sum_{i=1}^r S_{t-i} A_i S_{t-i} \\
B_i = A_i, \forall i = 1, \cdots, r \\
B_i \succeq 0, \forall i = 1, \cdots, r \\
\|\sum_{i=1}^r B_i\|_2 \leq 1
\]

(8.6)

The corresponding augmented Lagrangian is:

\[
\mathcal{L}(\hat{P}_t, B_t, A_i) = \sum_t \left( J_{ld}(P_t, \hat{P}_t) + \frac{\mu}{2} \|\hat{P}_t - \sum_{i=1}^r S_{t-i} A_i S_{t-i} + \beta_t\|^2 \right) + \sum_{i=1}^r \left( [B_i]_{+2} + \frac{\mu}{2} \|A_i - B_i + \gamma_i\|^2 \right)
\]

(8.7)

where \(\mu\) here is a penalty parameter and we define an extended real-valued function \([\cdot]_{+2}\) on the convex set \(\{B_i \succeq 0, \|\sum_i B_i\|_2 \leq 1\}\):

\[
[B_i]_{+2} = \begin{cases} 
0, & \text{if } B_i \succeq 0, \text{ and } \|\sum_i B_i\|_2 \leq 1 \\
+\infty, & \text{otherwise}
\end{cases}
\]

(8.8)

and the last term enforces \(A_i \succeq B_i\).

Recall that the ADMM method proceeds iteratively, minimizing the Lagrangian with respect to one set of variables while holding the rest constant, and then updating the the scaled dual variables \(\beta_t\) and \(\gamma_i\) via a gradient ascent step. As shown next, in our case, this minimization can either be done explicitly, (albeit in the case of \(P\) this entails an approximation motivated by concave-convex procedure (CCCP)[151] ideas).

In the following part, we will introduce the two primal update steps in the ADMM, which is

\[
\text{update: } \{\hat{P}_t, B_i\} \leftarrow \arg \min_{\hat{P}_t, B_i} \mathcal{L}(\hat{P}_t, B_i, A_i)
\]

\[
\text{update: } \{A_i\} \leftarrow \arg \min_{A_i} \mathcal{L}(\hat{P}_t, B_i, A_i)
\]

(8.9)

This splitting allows for having intermediate solutions that violate the PSD constraint, thus substantially improving the convergence properties of the algorithm.
8.3. ESTIMATION OF THE GARCH MODEL

Notice that, for the \( \{ \hat{P}_t, B_t \} \) update, since the variables \( \hat{P}_t \) and \( B_t \) relates to \( A_i \) individually, we can establish this step by update them individually. For instance, we first update \( \hat{P}_t \), then update \( B_t \) as follows.

1. **\( \hat{P}_t \) Update**: requires solving an optimization problem for each \( t \) (\( t \) is omitted in order to simplify the notation):

\[
\min_{\hat{P}} J_{ld}(P, \hat{P}) + \frac{\mu}{2} \| \hat{P} - Z \|^2
\]

Although the problem above does not admit a closed form solution, since in the region \( \{ \hat{P} \in S_{++}^{n+} : \hat{P} \preceq (1 + \sqrt{2})P \} \), \( J_{ld} \) is convex, it can be solved using for instance gradient based methods. Alternatively, a computationally attractive approximation that leads to closed form solutions can be obtained exploiting ideas from CCCP as follows. Note that:

\[
\min_{\hat{P}} J_{ld}(P, \hat{P}) + \frac{\mu}{2} \| \hat{P} - Z \|^2
= \min_{\hat{P}} -\frac{1}{2} \log |\hat{P}P| + \frac{\mu}{2} \| \hat{P} - Z \|^2 + \log \left| \frac{\hat{P} + P}{2} \right|
\]

(8.11)

where \( F_{vex}(\hat{P}) = -\frac{1}{2} \log |\hat{P}P| + \frac{\mu}{2} \| \hat{P} - Z \|^2 \) is a convex function with respect to \( \hat{P} \), and \( F_{cave}(\hat{P}) = \log \left| \frac{\hat{P} + P}{2} \right| \) is a concave function w.r.t \( \hat{P} \). Motivated by the CCCP method, we will minimize the upper bound of this function obtained by replacing the concave term with its first order Taylor expansion around the current best estimation \( \hat{P}^{(k)} \). This approximation leads to the following expression for \( \hat{P}^{(k+1)} \):

\[
\hat{P}^{(k+1)} = \arg \min_{\hat{P}} F_{vex}(\hat{P}) + \langle \hat{P}, \partial_{\hat{P}} F_{cave}(\hat{P}^{(k)}) \rangle
\]

(8.12)

Setting the gradient with respect to \( \hat{P} \) to zero, we have

\[
-\frac{1}{2}(\hat{P}^{(k+1)})^{-1} + \mu(\hat{P}^{(k+1)} - Z) = -\frac{1}{2} \left( \frac{\hat{P}^{(k)} + P}{2} \right)^{-1}
\]

(8.13)

Defining \( Q = \left( \frac{\hat{P}^{(k)} + P}{2} \right)^{-1} - 2\mu Z \), we have

\[
(\hat{P}^{(k+1)})^{-1} - 2\mu \hat{P}^{(k+1)} = Q
\]

(8.14)

Finally, considering the eigenvalue decomposition of \( Q = V \Sigma V^T \), and using the fact that \( \hat{P} \succeq 0 \), leads to the following closed form expression for the solution to (8.14):

\[
\hat{P}^{(k+1)} = V \Sigma V^T
\]

(8.15)
with $\Sigma(i, i) = \sqrt{\frac{D(i,i)^2 + 8\mu - D(i,i)}{4\mu}}$.

2. $B_i$ Update is given by:

$$\{B_i\}_{i=1}^r = \arg\min_{\{B_i\}} \sum_{i=1}^r \left( [B_i]_{+2} + \frac{\mu}{2} \|B_i - (\mathbf{A}_i + \gamma_i)\|^2 \right)$$

(8.16)

The solution for this problem is first projecting $\mathbf{A}_i + \gamma_i$ onto the set $S_+$, given by $B_i = \text{Proj}_+(\mathbf{A}_i + \gamma_i)$, then scaling $B_i = \frac{B_i}{\|\sum_i B_i\|^2}$ if $\|\sum_i B_i\|^2 > 1$.

3. $A_i$ Update. As shown below the update for each $A_i$ simply reduces to a Least-Square problem of the form:

$$A_i = \arg\min_{A_i} \sum_{i} \frac{\mu}{2} \| \mathbf{P}_i - \sum_{i=1}^r S_{i-1} A_i S_{i-1} + \beta_i \|^2$$

$$+ \frac{\mu}{2} \|A_i - (B_i - \gamma_i)\|^2$$

$$= \arg\min_{A_i} \| M x - h \|^2 = \text{mat}(M^{-1}h)$$

(8.17)

where $\text{mat}(\cdot)$ reshapes vector to matrix of suitable size,

$$M = \begin{bmatrix}
S_r^T \otimes S_r & \cdots & S_1^T \otimes S_1 \\
\vdots & \ddots & \vdots \\
S_{T-1}^T \otimes S_{T-1} & \cdots & S_{T-r}^T \otimes S_{T-r} \\
I
\end{bmatrix}$$

$$x = \begin{bmatrix}
\text{vec}(A_1) \\
\vdots \\
\text{vec}(A_r) \\
\text{vec}(\mathbf{P}_{r+1} + \beta_{r+1}) \\
\vdots \\
\text{vec}(\mathbf{P}_r + \beta_r) \\
\text{vec}(B_1 - \gamma_1) \\
\vdots \\
\text{vec}(B_r - \gamma_r)
\end{bmatrix}$$

(8.18)

Here, we have exploited the fact that $\text{vec}(\mathbf{A}\mathbf{B}\mathbf{C}) = (\mathbf{C}^T \otimes \mathbf{A}) \text{vec}(\mathbf{B})$. 


Next, we present the theorem regarding the convergence of the proposed ADMM procedure. Based on the proof, we discuss the $\mu$ strategy applied in the proposed ADMM type algorithm.

**Theorem 10.** Assume that the given experimental data can be explained by a model of the form

$$\hat{P}_t = \sum_{i=1}^{r} S_{t-i} A_i S_{t-i}; \quad S_{t-i} = P_{t-i}^{\frac{1}{2}},$$

such that the interpolation error matrix $\Pi_t \triangleq \hat{P}_t - P_t \preceq \sqrt{2} P_t$. Then, Algorithm 11 converges to the optimal solution of Problem 9, provided that the trajectories $\hat{P}_t^{(k)}$ remain inside the region $\hat{P}_t^{(k)} \preceq (1 + \sqrt{2}) P_t$

**Proof.** Since the trajectories remain in the region $\hat{P}_t^{(k)} \preceq (1 + \sqrt{2}) P_t$, where $J_{id}(\hat{P}, P)$ is convex, then, when using the exact update rule

$$\hat{P}_t^{(k)} = \arg \min_P J_{id}(\hat{P}, P) + \frac{\mu}{2}\|\hat{P} - Z\|^2$$

leads to a standard ADMM algorithm and the convergence proof in [152] holds. If one uses instead the approximate update in (8.12), From Theorem 2 in [151], since we are working in the region where $J_{id}$ is convex (and hence has no local maxima), the update guarantees that

$$J_{id}(\hat{P}_t^{(k+1)}, P) + \frac{\mu}{2}\|\hat{P}_t^{(k+1)} - Z\|^2 < J_{id}(\hat{P}_t^{(k)}, P) + \frac{\mu}{2}\|\hat{P}_t^{(k)} - Z\|^2$$

unless the algorithm had converged to the (unique) minimum in the region of interest. From here on, the proof proceed along the lines of the proof in [152].

4. $\mu$ Update. From the above proof, we can tell that in order to keep the ADMM in a convex region, we would prefer $\mu$ to be small. Following this idea, instead of using the most famous $\mu$ strategy: $\mu \to +\infty$, we use the adaptive strategy introduced in [152]. The idea of this strategy is to adaptively find a fixed optimal $\mu$ for ADMM. We can easily see that in the ADMM updates, $\hat{P}_t$ update and $B_i$ update don’t affect each other while they are both related to $A_i$. As we mentioned in (8.9), we can think the variables $\hat{P}_t$ and $B_i$ as a single variable, and $A_i$ as the other group, which will admit an equality constraint

$$Ix - Mz - c = 0$$
where, $I$ denotes identity matrix, $c = 0$ and

$$
\begin{bmatrix}
\text{vec}(\hat{P}_{r+1}) \\
\vdots \\
\text{vec}(B_1) \\
\vdots \\
\text{vec}(B_r)
\end{bmatrix},
\quad
z =
\begin{bmatrix}
\text{vec}(A_1) \\
\vdots \\
\text{vec}(A_r)
\end{bmatrix}
$$

then following the analysis in [152], we can have that the residue for the dual feasibility condition

$$s^{\text{iter}+1} = -\mu M(z^{\text{iter}+1} - z^{\text{iter}})$$

and the primal residual

$$r^{\text{iter}+1} = x^{\text{iter}+1} - Mz^{\text{iter}+1}$$

**Algorithm 10 $\mu$ Strategy**

```plaintext
if $\|r^{\text{iter}}\|_2 \geq 10\|s^{\text{iter}}\|_2$ then
    $\mu = 2\mu, \ \beta_i = \beta_i/2, \ \gamma_i = \gamma_i/2.$
end if

if $\|s^{\text{iter}}\|_2 \geq 10\|r^{\text{iter}}\|_2$ then
    $\mu = \mu/2, \ \beta_i = 2\beta_i, \ \gamma_i = 2\gamma_i.$
end if
```

**5. Stop Criteria.** For stop condition, we use the one introduced in [152], and customize it according to our ADMM formulations,

$$
\|r^{\text{iter}}\|_2 \leq \sqrt{n}e^{\text{abs}} + e^{\text{rel}} \max\{\|x^{\text{iter}}\|_2, \|MZ^{\text{iter}}\|_2\}
$$

$$
\|s^{\text{iter}}\|_2 \leq \sqrt{n}e^{\text{abs}} + e^{\text{rel}} \|y^{\text{iter}}\|_2
$$

where, $y = [\mu\beta_{r+1}^T \cdots \mu\beta_r^T \mu\gamma_1^T \cdots \mu\gamma_r^T]^T$.

Collecting the results above leads to the ADMM based identification procedure outlined in Algorithm 11.
Algorithm 11 Solving Problem (8.5) by ADMM

Inputs: \( \{ P_t \}_{t=1}^{T}, S_t = P_t^{1/2} \).
Initialize: \( \hat{P}_t = 0, A_i = B_i = 0, \beta_t = 0, \gamma_i = 0, \mu = 1, e^{obs} = e^{rel} = 10^{-4}, N = 100. \)

while stop criteria is not satisfied do

1. fix the others and update \( \hat{P}_t \) by
   Initializing \( \hat{P}_t^{(0)} \) as positive-definite matrices and letting
   \[
   Z_t = \sum_{i=1}^{r} S_{t-i} A_i S_{t-i} - \beta_t
   \]
   while not converged do
     solve (8.15) using \( Q_t = \left( \frac{\hat{P}_t^{(k)} + P_t}{2} \right)^{-1} - 2\mu Z_t. \)
   end while

2. fix the others and update each \( B_i \) by
   \[
   B_i = \text{Proj}_+(A_i + \gamma_i)
   \]
   then, \( B_i = \frac{B_i}{\| \sum_i B_i \|_2} \) if \( \| \sum_i B_i \|_2 > 1. \)

3. fix the others and update \( A_i \) by forming \( M \) and \( h \) as (8.18) and take
   \[
   \begin{bmatrix}
   \text{vec}(A_1) \\
   \vdots \\
   \text{vec}(A_r)
   \end{bmatrix} = M^{-1} h
   \]

4. update the scaled multipliers
   \[
   \beta_t = \beta_t + (\hat{P}_t - \sum_i S_{t-i} A_i S_{t-i})
   \]
   \[
   \gamma_i = \gamma_i + (A_i - B_i)
   \]

5. update the parameter \( \mu \) using Algorithm 10 in the first \( N \) iterations.
end while
8.4 A JBLD Maximum Likelihood Estimator

In this section, we show that a suitably modified version of Problem 10 admits a closed form solution. To this effect, we begin by re-examining the conjugate prior of the multivariate Gaussian distribution.

8.4.1 A Generalized Gaussian Conjugate Prior

Combining (8.2) and (8.3) it follows that the likelihood function of $P_t$, the present value of the covariance given an observation $Q_t$ and past values $P_{t-r}^{i-1}$ satisfies:

$$L(P_t, Q_t, P_{t-r}^{i-1}) \propto |P_t^{-1}|^{-\frac{n-1}{2}} e^{-\frac{tr(P_t^{-1}Q_t)}{2}} \times e^{-\frac{J_{ld}(P_t, \sum_{i=1}^{r} s_{t-i} A_i s_{t-i})}{2|\lambda|}} \quad (8.26)$$

In principle, this expression can be used to find a maximum likelihood estimate of $P_t$. However, the resulting optimization problem is not amenable to closed form solutions. In addition, the first factor in (8.26) does not take into account the manifold geometry. As we show next, surprisingly, if $QP^{-1} \approx I$, that is the prediction and observation are roughly aligned and with the same magnitude, then (8.26) can be expressed in terms of the JBLD, leading to closed form solutions.

**Theorem 11.** Let $X \triangleq P^{-\frac{1}{2}} Q P^{-\frac{1}{2}}$ and denote by $\lambda_i$ the eigenvalues of $\Delta \triangleq X^{-1} - I$. Then,

$$e^{-\frac{1}{2\sigma^2} J_{ld}(P, Q)} \propto |P^{-1}|^{-\frac{n-1}{2}} e^{-\frac{tr(P^{-1}Q)}{2}} + O(\lambda_i) \quad (8.27)$$

where, for notational simplicity we defined $\frac{1}{2}\sigma^2 \triangleq n$ and $\hat{Q} \triangleq nQ$.

**Proof.** From the explicit expression of $J_{ld}$ it follows that

$$e^{-\frac{1}{2\sigma^2} J_{ld}(P, Q)} = |X|^{-\frac{1}{2\sigma^2}} |I + \frac{X - I}{2}|^{-\frac{1}{2\sigma^2}} = |X|^{-\frac{1}{2\sigma^2}} \prod_{i=1}^{d} (1 + \lambda_i)^{-\frac{n}{2}} \quad (8.28)$$

Next, note that

$$\prod_{i=1}^{d} (1 + \lambda_i)^{-\frac{n}{2}} = (1 - n \sum_{i=1}^{d} \lambda_i + O(\lambda_i^2)) = e^{-\frac{ntr(X - I)}{2}} + O(\lambda_i^2) \quad (8.29)$$

Replacing (8.29) in (8.28) and using the fact that

$$|X|^{-\frac{n}{2}} = |X|^{-\frac{n-1}{2}} |X|^{-\frac{1}{2}} = |X|^{-\frac{n-1}{2}} |I + \Delta + O(\lambda_i^2)| = |X|^{-\frac{n-1}{2}} (1 + O(\lambda_i)) \quad (8.30)$$
yields:

\[ e^{-\frac{1}{2\sigma^2} J_{ld}(P, Q)} = e^{\frac{d}{2\sigma^2} \left| X \right|^2} e^{-\frac{u(P^{-1}P^Q)}{2}} + \mathcal{O}(\lambda_i) \]

(8.31)

Remark 11. The Theorem above shows that the likelihood function of a Wishart distribution can be approximated by a kernel using the JBLD. In the following, we will show that this is the key to obtaining a tractable, fast solution in terms of the Stein mean.

8.4.2 An explicit MLE

From Theorem 11 and (8.3) it follows that Problem 10 can be solved using a likelihood function of the form:

\[
p(P_t | Q_t, P_{t-1}, \ldots, P_{t-r}) = \frac{1}{Z_s} e^{-\frac{J_{ld}(Q_t, P_t)}{2\sigma^2}} \times e^{-\frac{J_{ld}(P_t, \sum_{i=1}^{r} S_{t-i}A_iS_{t-i})}{2\sigma^2}}
\]

(8.32)

where \( Q_t \) denotes the noisy observation and \( Z_s \) is a normalization factor. In this context, the MLE of \( P_t \) is given by:

\[
P_t^* = \arg \max_{P_t} p(P_t | P_{t-1}, \ldots, P_{t-r}) p(Q_t | P_t)
\]

(8.33)

or, equivalently,

\[
P_t^* = \arg \min_{P_t} (1 - \lambda) J_{ld}(P_t, \sum_{i=1}^{r} S_{t-i}A_iS_{t-i}) + \lambda J_{ld}(Q_t, P_t)
\]

(8.34)

where, \( \lambda = \frac{\omega^2}{\sigma^2 + \phi^2} \). The solution to this optimization is a weighted Stein Mean, which admits the following closed form solution[153]:

\[
P_t^* = \hat{P}_t \left[ \sqrt{P_t^{-1}Q_t + \frac{(2\lambda - 1)^2}{4}(I - \hat{P}_t^{-1}Q_t)^2} \right.
\]

(8.35)

\[
- \frac{2\lambda - 1}{2} (I - \hat{P}_t^{-1}Q_t)
\]

where \( \hat{P}_t = \sum_{i=1}^{r} S_{t-i}A_iS_{t-i} \), leading to the JBLD recursive filter algorithm outlined in Algorithm 12.
Algorithm 12 JBLD Recursive Filter (JBRF)

**Inputs:** past estimations \( \{\hat{P}_{t-1}, \ldots, \hat{P}_{t-r}\} \), \( S_{t-i} = \hat{P}_{t-i}^{-\frac{3}{2}} \), observation \( Q_t \) and \( \lambda = \frac{\omega^2}{\omega^2 + \phi^2} \).

**Prediction:**

\[
\tilde{P}_t = \sum_{i=1}^{r} S_{t-i} A_i S_{t-i}
\]

**Correction:**

\[
\hat{P}_t = \tilde{P}_t \left[ \sqrt{\tilde{P}_t^{-1} Q_t + \frac{(2\lambda - 1)^2}{4} (I - \tilde{P}_t^{-1} Q_t)^2} - \frac{2\lambda - 1}{2} (I - \tilde{P}_t^{-1} Q_t) \right]
\]

**Outputs:** \( \hat{P}_t = \text{JBRF}(\hat{P}_{t-1}, \ldots, \hat{P}_{t-r}, Q_t) \).

### 8.5 Experiments

In this section, we illustrate the advantages of the proposed JBLD recursive filter (JBRF) by comparing its performance using both synthetic data and real data, against the following three state-of-the-art methods:

**Manifold Mean.** [19] proposed using the Karcher mean of past observations as the estimator for the present value of the covariance. Note that the Karcher mean is based on using the Affine Invariant Riemannian metric. Thus, for consistency, we modified this method to use the Stein, rather than the Karcher, mean, since the former is the manifold mean under the JBLD metric used in this work. In the experiments involving synthetic and video clips downloaded from Youtube, we set the memory length of this method to 20, which allows it to use a larger number of past observations compared to JBRF, IRF and LRF.

**LRF.** The recursive filter for linear system on PD manifold introduced in [20] obtained using the Euclidean distance computed using the matrix \( \log \) and \( \exp \) operator to flatten the manifold.

**IRF.** The intrinsic recursive filter on PD manifold proposed in [27].

#### 8.5.1 Synthetic Data Experiments

The goal here is to compare all methods in a simple scenario: estimation of a constant covariance matrix in \( S^3_{++} \). Thus, a time sequence of corrupted observations was randomly sampled by adding Gaussian noise to an identity matrix \( I_3 \). First, a vector \( w \in \mathbb{R}^6 \) was sampled from a
8.5. EXPERIMENTS

(a) $\sigma^2 = 0.1$

(b) $\sigma^2 = 1$

(c) $\sigma^2 = 2$

Figure 8.2: Mean estimation error from 20 trials for the synthetic data experiment.

Gaussian distribution $\mathcal{N}(0, \sigma^2 I_6)$, and used to form a matrix $W \in S^3$. Then the noise $W$ was added to $I_3$ using the manifold exponential operator $\exp_X(v)$

$$
\exp_X(v) = X^{1/2} \exp_m(X^{-1/2} v X^{-1/2}) X^{1/2}
$$

(8.36)

Note that the manifold exponential operator maps the tangent vector $v$ to the location on the manifold reached in a unit time by the geodesic starting at $X$ in the tangent direction.

We chose $\sigma^2 = \{0.1, 1, 2\}$, and for each value we generated 20 sequences of length 1000, which can be viewed as random measurements of the identity matrix. Our recursive filter was applied as an estimator of the sequence, as well as the Manifold Mean method, IRF and LRF. The estimation error was computed using the JBLD between the estimations and the ground truth $I_3$. For each value of $\sigma^2$, we took the mean of estimation error over the corresponding 20 sequences.

The value of the tuning parameters for IRF and LRF was chosen according to the corresponding set-ups. For IRF, the parameters were set to $\phi^2/\omega^2 = 200$, as reported in [27]. For LRF, we set $\Omega = \omega I_6$ with $\omega = 0.0001$ and $\Phi = \sigma^2 I_6$ as reported in [20]. For the base point of LRF, we used the first observation of each sequence, which is the best information available about the sequence before filtering. The justification for this setting is given in [27]. For our filter, we set the parameters as $\phi^2/\omega^2 = 50$.

The mean estimation errors from 20 trials for each different noise level are shown in Figure 8.2a, 8.2b and 8.2c. It can be observed that both Manifold Mean and LRF converge faster than JBRF in terms of the number of iterations. However, as the noise level increases, the estimation error of LRF gets larger, which leads to the worst performance compared to JBRF and IRF. On the other hand, the performance of Manifold Mean is constant, and worse than JBRF and IRF but better than LRF for larger noise level. The reason of this poor performance of Manifold Mean on synthetic data is that a memory length of 20 is still not enough to eliminate the noise effects. However, even for
a memory length of 20, the Stein Mean computation is already the slowest in terms of computing
time. Both IRF and our method show robust performance with respect to different noise levels. In
terms of running time, the proposed method is the fastest, with an average time of 0.11 seconds for
each sequence, running on an iMAC with a 4 GHz CPU. This is about 6 times faster than LRF (0.66
seconds), around 20 times faster than IRF (2.43 seconds) and almost one order of magnitude faster
than calculating the Stein mean (on average 10 seconds).

![Figure 8.3: Tracking under occlusion. Top: sample training data. Bottom: tracking results using
different filters.](image)

### 8.5.2 Tracking Under Occlusion and Clutter

This lab experiment was specifically designed to provide a very challenging environment,
shown in Fig 8.3. The goal here is to track a multicolored spinning ball in the presence of occlusion
(frames 16 – 19) and clutter whose color covariance descriptors are, in some frames, similar to
those of the target. Note that, due to the spinning, the appearances of the target as it enters and
emerges from the occlusion are different, thus necessitating a data-driven framework capable of
accurately predicting this change. We first used the information from frames 1 to 14 to identify an
11th order model of the form (8.3) that captures the evolution of the covariance feature obtained
from the coordinates and color of the target:

\[
f(x, y) = \begin{bmatrix} x & y & R(x, y) & G(x, y) & B(x, y) \end{bmatrix}
\]

(8.37)
Next, we used the different filters to estimate the covariance feature starting from frame 15, which is the last unoccluded frame, based on the data from frames 1 – 14. To this effect, we first used the dynamical model to predict the covariance feature in the next frame. Next, we searched for the best match (in the JBLD or, in the case of IRF, AIRM, sense) by comparing against the covariance features obtained using a sliding window (with size equal to that of the bounding box in frame 14), generated every 3 pixels. The best match was chosen as the target in the frame and used as observation to perform the correction step in all filtering methods. During the occlusion, no correction step was performed. Again, for LRF and IRF, the parameters were chosen as reported in [20] and [27]. For JBRF, we set $\omega = 0.01$, and $\phi = 0.01$.

As shown in Figure 8.3, only the proposed method is capable of sustained tracking. This is due to the fact that the LRF and IRF methods use only information from at most the previous two frames, which in this case cannot accurately predict the evolution of the covariance in the absence of correction steps. On the other hand, the Stein Mean method, even using information from more than the past 11 frames, fails due to the fact that the updating methodology does not reflect the dynamic evolution of the target.

### 8.5.3 Youtube Video Experiments

In this experiment, we evaluate the proposed filter using several Youtube videos with more than 1000 frames in total. The videos contain a spinning multicolored ball and fish schooling behavior. We divided each sequence into two parts: training data (around 60%) and testing data (around 40%). For each sequence, we first extracted RGB covariance features from the object (the spinning ball or the entire fish school) and used the training data to estimate the model parameters for JBRF. The system order was determined empirically, by searching for the best fit. The data was corrupted with Gaussian noise $\mathcal{N}(0,0.01)$ prior to extracting the covariance features. These corrupted covariance sequences were then processed using the different estimation methods. The tuning parameters for this experiment were set as follows. For JBRF and IRF, we first calculated the fitting error of the state transition, in the corresponding non-Euclidean metric, using the training sequence and associated system model. The parameter $\omega$ for JBRF and IRF was then set to the unbiased estimation of the standard deviation using these fitting errors. For the parameter $\phi$ which controls the variance of the observation noise, we performed a grid search with values $1e\{-3,-2,-1,0,1,2,3\}$ and used the one giving minimum estimation error. For LRF, we set $\omega = 0.0001$ as proposed in [20], and performed a grid search for $\phi$ with values $1e\{-3,-2,-1,0,1,2,3\}$. The results reported correspond
CHAPTER 8. GEOMETRY-AWARE OPTIMAL FILTER ON PD MANIFOLD

Table 8.1: Mean Estimation Error and Running Time for the Experiments Using Youtube Video Clips (number in parathesis denotes the system order of JBRF model)

<table>
<thead>
<tr>
<th>Error</th>
<th>Methods</th>
<th>Spinning Ball</th>
<th>Fish #1</th>
<th>Fish #2</th>
<th>Fish #3</th>
<th>Fish #4</th>
<th>Fish #5</th>
<th>Fish #6</th>
</tr>
</thead>
<tbody>
<tr>
<td>JBRF</td>
<td>0.3565</td>
<td>1.2901</td>
<td>0.9664</td>
<td>0.5819</td>
<td>1.5692</td>
<td>1.6730</td>
<td>1.5302</td>
<td></td>
</tr>
<tr>
<td>IRF</td>
<td>0.4937</td>
<td>1.5266</td>
<td>1.7863</td>
<td>0.6691</td>
<td>1.9112</td>
<td>1.8701</td>
<td>1.8741</td>
<td></td>
</tr>
<tr>
<td>LRF</td>
<td>0.5792</td>
<td>1.5294</td>
<td>1.7900</td>
<td>0.6726</td>
<td>1.9116</td>
<td>1.8700</td>
<td>1.8746</td>
<td></td>
</tr>
<tr>
<td>Stein</td>
<td>0.6037</td>
<td>1.5350</td>
<td>1.8114</td>
<td>0.6744</td>
<td>1.9117</td>
<td>1.8719</td>
<td>1.8750</td>
<td></td>
</tr>
<tr>
<td>Mean baseline</td>
<td>0.4936</td>
<td>1.5266</td>
<td>1.8126</td>
<td>0.6829</td>
<td>1.9112</td>
<td>1.8716</td>
<td>1.8741</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Running Time (seconds/100 frames)</th>
<th>JBRF</th>
<th>IRF</th>
<th>LRF</th>
<th>Stein Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>JBRF</td>
<td>0.0691</td>
<td>0.2519</td>
<td>0.0930</td>
<td>0.7345</td>
</tr>
<tr>
<td>IRF</td>
<td>0.0320</td>
<td>0.2628</td>
<td>0.0989</td>
<td>0.7440</td>
</tr>
<tr>
<td>LRF</td>
<td>0.0288</td>
<td>0.2484</td>
<td>0.1008</td>
<td>0.7444</td>
</tr>
<tr>
<td>Stein</td>
<td>0.0245</td>
<td>0.2575</td>
<td>0.0976</td>
<td>0.7371</td>
</tr>
<tr>
<td>Mean</td>
<td>0.0241</td>
<td>0.2613</td>
<td>0.0959</td>
<td>0.7293</td>
</tr>
<tr>
<td>baseline</td>
<td>0.0245</td>
<td>0.2629</td>
<td>0.0953</td>
<td>0.7361</td>
</tr>
<tr>
<td>baseline</td>
<td>0.0245</td>
<td>0.2700</td>
<td>0.1000</td>
<td>0.7601</td>
</tr>
</tbody>
</table>

The estimation error was again computed using JBLD between the estimations and the ground truth (extracted from frames before corruption). The mean estimation errors and average run time to filter 100 frames are shown in the Table 8.1. Sample frames from several sequences are shown in Figure 8.4 along with their noise corrupted counterparts. Table 8.1 shows that indeed JBRF achieves the minimum estimation error among all methods, while, at the same time being 60% faster than the closest competitor. It is also worth emphasizing that the performance improvement is not just due to the fact that the JBLR can use higher order models. As shown in the last five columns of the table, using a data driven model leads to substantial performance improvement, even when the order of this model is comparable to the one used by competing methods.

8.6 Chapter Summary

In this chapter, we proposed a framework for obtaining maximum likelihood estimates of both the dynamic propagation model and of the present value of the matrix. The main advantages
of the proposed approach, compared against existing techniques are (i) the ability to identify the propagation model and to exploit it to obtain better predictions while taking into account the non-Euclidean geometry\(^3\) of the problem, and (ii) the use of a generalized Gaussian approximation to the Jensen Bregman LogDet Divergence that leads to closed form maximum likelihood estimates. As illustrated both with synthetic and video data, the use of the identified manifold dynamics combined with the JBLD metric leads to filters that compare favorably against existing techniques both in terms of the estimation error and the computational time required to compute the estimates.

\(^3\)Note that these results cannot be obtained by embedding the data in a Hilbert space and using Euclidean geometry based filtering there, since, as shown in [147, 148] the JBLD can be isometrically embedded in a Hilbert space only when working with commuting matrices.
Chapter 9

Conclusion and Future Work

In practice, large scaled network and data collection has become ubiquitous. These scenarios generally involve complex data exchange and/or large data volume, which introduce new difficulties to data modeling, estimation and feedback control. In order to handle these new challenges, this dissertation discusses three categories of system design problem considering the sparsity and structural constraints.

The first category is the information structure constrained estimator and controller design. Low complexity of model structure, as a typical idea in practice, can be implemented as sparsity structure on the information channels in a large-scaled network systems. While the problems involving a sparse information structure are generically NP-hard, several convex relaxations are introduced in this category. In the aspect of estimator design, we consider the worst-case optimal estimator following the concept of information based complexity. Two cases are studied: the estimator design based on sensor selection, and the estimator design for switched ARX system. NP-hardness of these problems comes from the binary property of selecting either sensing channels or the switching modes. In the proposed approach, we illustrate that these binary variables can be easily handled using polynomial constraints, which leads to convex SDP approaches by applying moments-based relaxation techniques. Moreover, we show that the relaxation can be exact with SDP of finite size. Regarding the second set of challenges, a general form of sparse structured controller design is discussed in both static and dynamic control scenarios. We first present a semi-algebraic optimization based convex relaxation for moderately large problem, which is guaranteed to find a controller if one exists. Then, in order to solve large scale problem, we further proposed an optimal filtering based convex approach, with sufficient certificate for the existence of a desired controller. Finally, regarding the third set of challenges, we present a new super-atomic norm formulation estimating
the dynamical graphical network with sparse structure. By further exploiting a Frank-Wolfe type method, a computationally efficient algorithm is proposed. This algorithm shows good scalability, and outperforms existing methods in both accuracy and speed.

Sensor fault is an unavoidable problem in big data collection, which motivates the second class of problems addressed in this dissertation. In order to handle the corruptions from fault sensors, it is necessary to reject these measured data while estimating the model, which leads to the use of robust regression. In this topic, we present an improved robust regression formulation, called the self scaled regularized robust regression ($S^2R^3$). As shown in both theoretical and experimental aspects, the proposed method achieves better performance in both the robustness to gross outliers and the accuracy of model estimation. It is worth mentioning that due to the LP formulation of $S^2R^3$, the proposed method is also very competitive in terms of computational time.

In order to handle the large volume of data collection, compressive data representations are also considered to capture the information with formulations of smaller or fixed size, such as sampling algorithms, compressive sensing, covariance feature, etc. This leads to the third class of problems addressed in the dissertation: identification of models for propagating covariance features, whose size is independent from the number of data samples. Although the covariance feature belongs to the convex positive definite cone in the Euclidean space, many pioneer works have shown that substantial improvement can be achieved by considering the PD manifold structure instead of Euclidean geometry. In the proposed approach, a maximum likelihood filter is introduced by combining a geometry-aware metric and an on-manifold model. Also, a systematic method is introduced to identify the dynamical model from experimental data, which is, to the best of our knowledge, the first attempt in the approaches based on optimal filtering.

There are several possible extensions of the proposed methods that have not been addressed in this dissertation. Further work will include:

1. **Improving computational efficiency.** Although the moment-theory based SDP relaxation is a powerful technique achieving global optimality in special problems, the computation complexity of solving an SDP problem is still high, $O(n^3)$. Therefore, in order to generalize our algorithms into scenarios involving large data sets, such as very large networks, it’s reasonable to consider customizing an efficient optimization scheme for large size moments-based SDP by utilizing some problem-based properties. It can be observed that the moments-based SDP often admits a low-rank solution when the flat extension property is satisfied. Also given the idea that the rank of moment matrix represents how many peaks in the potential probability
density function generating the moment variables. And potentially, the peaks correspond to the locations of globally optimal solutions. It is reasonable to pursue low rank solution while solving moments-based SDP (even before the flat extension property can be satisfied). Given the efficiency of Frank-Wolfe algorithm in finding sparse and low rank solutions (see [47] for a detailed review), it is reasonable to seek to develop Frank-Wolfe methods based on atomic-norm formulation. However, due to the special structure of moment matrix, the atoms here will not be simply rank 1 positive semi-definite matrices as in general SDP.

2. Connection to graph theory. In [154], an interesting and systematic insight of the exactness of moments-based SDP relaxation is discussed. The authors build sufficient and necessary conditions through the tree width of a generalized weighted graph relating to the (hidden) structure of the polynomial optimization. A direction for future research is to build a connection between the proposed methods in this dissertation and the graph theory.
Bibliography


