Rank Constraints and Error Quantification in Restricted Complexity Problems

Ramón A. Delgado Pulgar

Ingeniero Civil Electrónico
M.Sc. in Electronics Engineering

A thesis submitted in fulfilment
of the requirements for the degree of
Doctor of Philosophy

November, 2014

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Declaration

The thesis contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. I give consent to the final version of my thesis being made available worldwide when deposited in the University's Digital Repository, subject to the provisions of the Copyright Act 1968.

I hereby certify that the work embodied in this thesis contains published papers of which I am a joint author. I have included as part of the thesis a written statement, endorsed by my supervisor, attesting to my contribution to the joint publications.

Ramón A. Delgado Pulgar

April, 2014

Acknowledgements

First and foremost, I would like to thank the unconditional love and support of my wife, Rocío, and my daughters, Antonella and Laura. Specially, I would like to thank Rocío for making me want to be a better person, someone who deserves such a wonderful wife.

Secondly, I would like to thank Prof. Graham Goodwin, for his guidance, encouragement and patience. The experience of being under your supervision has been invaluable on both an academic and a personal level. Thank you for the support throughout all these years. I would also like to thank Dr. Juan C. Agüero for his advice, understanding and dedicated involvement throughout the development of my academic career.

During my PhD I had the opportunity to interact with several visitors. I would like to thank Prof. Arie Feuer for getting involved in the early stage of our work. I would also like to thank Daniel Dolz and Dr. Dimitrios Katselis for their good humour and for helping me to improve my work through their comments and suggestions.

I am very grateful of Diego, Ricardo and Kathy for all the great moments, good advice and support. I would also like to thank my friends Damian and Bea, Mauricio and Carmen, Boris and Bec, Eduardo and Hope, for all the good memories and support.

I greatly appreciate the support provided by Prof. Mario Salgado, and Dr. Juan Yuz for offering me the opportunity to start my PhD in Newcastle. I am grateful to Prof. Rick Middleton for his support towards the end of my PhD. I would also like to thank Dianne and Jayne for their good disposition, help and friendly support.

Many thanks to my friends and their families for all the support provided. Thank you Mauricio and Cinthya, Denis and Isabella, Shane and Elizabeth, Lewis and Gia, Katherine, Edwin and Brenda, Rolando and Georgina, Juan Pablo and Carolina, Isaac and Diana, David and Sandra, Alfonso and Cecilia.

I would like to thank the many friends I made during my stay in Newcastle. Thank you Daniel and Karen, Alejandro and Phoebe, Pierre and Elisa, Aurelio and Vi, Felipe and Cintia, Rodrigo

and Vylma, Sonja and Edwin, Steffi and Florian, Steven Sandi, Alain Yetendje, He Kong, Robert Palma, Fernando López-Caamal, Andrés López, Katrina Lau, Koen Van der Mijle, Jay Cobe, Juan Alvarez and Esteban Osella.

Back in Chile, I would like to express my deep gratitude to my parents, Ramón and Lidia, for their unconditional love and support. I would also like to extend huge, warm thanks to my brothers Daniel, Mauricio, Juan Pablo and Esteban for their support and for encouraging me with their best wishes. I want to express my gratitude to my parents-in-law, Sergio and Angélica, and to my sister-in-law, Javiera, for welcoming me into their family. I would like to thank my aunts, uncles and cousins for all their support and encouragement from afar.

I would also like to thank my friends from San Felipe and Valparaíso for all their support and friendship on the days before our move to Australia. Thank you Gretel, Mónica, Nikole, Gonzalo, Daniel, Cristian, Paula, Stjpe, Francisco, Marcela and Nelson, Valeria and Pedro, Virginia, Jaime, Sebastian Pulgar, Sebastian Novoa, Luna, Patricio and Daniela, Andrés.

I greatly appreciate the advice and the good memories from Eduardo Silva, rest in peace.

Finally, I would like to acknowledge the financial support received from the University of Newcastle.

To Rocío, Antonella and Laura.

Abstract

This thesis addresses two issues that arise in restricted complexity estimation problems: The first is estimation subject to rank constraints. The second corresponds to uncertainty quantification when the amount of data available is small relative to the number of variables to be estimated.

In many practical problems one wishes to choose a simple solution from a set of possible solutions. The reasons for this can be many fold. For example, in design problems, one may know that a simple solution is possible. However, one does not know how to obtain such a simple solution from a large set of possible alternatives. In estimation problems, one may deliberately restrict the set of possible solutions to avoid over-fitting of noisy data. We term the class of problems having simple solutions *restricted complexity problems*.

The first part of the thesis address restricted problems where the restriction on complexity can be related to constraining the rank of a particular matrix. This leads us to address rank-constrained optimization problems.

The second part of the thesis focuses on quantification of estimation-error. It is well known that, when the amount of data available for estimation is small, the variance error could be significantly large. In these circumstances it is beneficial to, not only, have an estimated value for the parameters but also to be able to quantify the associated error. However, most of the existing methods for error quantification rely upon asymptotic results with large data. We focus on parametric uncertainty quantification for finite data estimation, with an extension to the related problem of moving horizon estimation.

The third part of the thesis, focuses on quantification of estimation errors when the complexity of the model is deliberately chosen to be smaller than the complexity of the "true" model. This has motivated a novel approach, commonly known in the literature by the generic title "model error modelling", to uncertainty quantification. This has been a central theme in several areas including statistics, time series analysis, econometrics and system identification. In the third part of the thesis we address the problem of model error modelling for dynamic system identification.

Contents

1

Introduction

This thesis addresses two issues that commonly arise in restricted complexity estimation problems. The first issue is estimation subject to rank constraints. The second issue corresponds to uncertainty quantification. Both issues are connected to the bias-variance trade off problem.

In any estimation problem, there exists a bias-variance tradeoff. The bias-variance tradeoff has two competing aspects: (i) When the nominal model is less complex than the true model, then systematic errors arise since the nominal model cannot reproduce all of the dynamics of the true system. These errors are usually called "bias errors". (ii) On the other hand, choosing a more complex model may cause the estimated parameters to follow the characteristics of the particular noise realization. This leads to a variance error.

Choosing the model complexity that provides the right balance between Bias and Variance errors is a difficult task. In past literature, several tools has been developed to deal with the problem of choosing the right complexity. For example the, well known, Akaike information Criterion and the Bayesian Information Criterion address this issue [4, 97]. These criteria are aimed at finding the model complexity that provides the right balance between bias and variance errors. In recent years, the research focus has changed from finding the right model complexity to that of managing the balance between model complexity and estimation error [48]. In this context, regularization techniques are powerful tools to manage the bias-variance tradeoff. Regularization techniques deal with the complexity issue by adding a penalty term to the cost function. This extra term penalizes model complexity. Optimization in the presence of these extra terms, produces a tendency to choose models having reduced complexity.

Once the model complexity has been chosen, including prior information of the chosen complexity in the estimation problem is also not straightforward. In most applications, the complexity of the nominal model is managed by using the specific structure of the problem. However, there are many

associated difficulties. In many applications the notion of model complexity can be related to the rank of a particular matrix. In the first part of the thesis, we focus on estimation problems that can be formulated as rank-constrained optimization problems. We develop novel algorithms and analyze their behaviour.

Rank-constrained optimization finds application in many areas including data modelling, systems and control, computer algebra, signal processing, psychometrics, machine learning, computer vision among others [71]. In many of these applications, the rank of a matrix is related to the complexity of a model. For example, in system identification, the order of a rational system is equal to the rank of an infinite dimensional Hankel matrix. In factor analysis, the number of latent factors is equal to the rank of a covariance matrix. These and other areas of application will be studied in the thesis.

The second part of the thesis focuses on quantification of estimation-error. It is well known that, when the amount of data available for estimation is small, variance errors can be significantly large. Thus, knowledge of the parametric uncertainty becomes a significant issue. However, most of the existing methods for error quantification rely upon asymptotic results. Moreover, for state estimation methods, there exist several methods that provide estimates without any accuracy information at all. Thus, in the second part of the thesis, we focus on parametric uncertainty quantification for finite data estimation. We also extend these ideas to the related problem of moving horizon estimation. Moving horizon estimation transforms filtering, smoothing and prediction problems into a set of standard constrained optimization problems over a finite horizon. In order to limit the size of the problem, this technique requires that the range of data used for estimation be small. In turn, when new data arrives, the oldest data is summarized by a, so called, "arrival cost". We will use the approach described earlier for error quantification on finite data problems, to provide an arrival cost for the moving horizon estimation problem.

The third part of the thesis, focuses on quantification of estimation errors when the complexity of the model is deliberately chosen to be smaller than the complexity of the "true" model. The modelling of the model uncertainty has been a central theme in many areas such as statistics, time series analysis, econometrics and system identification. We focus on modelling the model uncertainty in dynamic models. In particular, we model the uncertainty as a realization of a stochastic process. In the third part of the thesis we study the problem of modelling the model uncertainty for dynamic system identification and develop a novel scheme for simultaneously estimating a nominal model and quantifying the model uncertainty.

1.1 Overview of Thesis Content

An outline of the remaining chapters of the thesis is given below.

Part I Model Complexity Management

- **Chapter 2** addresses the problem of optimization subject to rank constraints. We propose a general approach to rank-constrained optimization. This approach provides enough flexibility to handle additional convex constraints.
- **Chapter 3** applies the method outlined in Chapter 2 to Factor Analysis. We propose a novel approach to Factor Analysis which allows one to consider cross-correlated idiosyncratic errors.
- **Chapter 4** extends the results of Chapter 2 to general matrices, i.e. not necessarily positive semi-definite matrices.
- **Chapter 5** applies the method described in Chapter 4 to the problem of impulse response estimation. The efficacy of the proposed approach is studied via a numerical example.
- **Chapter 6** extends the results in Chapter 4 to the problem of cardinality-constrained optimization. We explore the ability to include group constraints into the cardinality-constrained optimization problem. We also develop an algorithm to solve cardinality-constrained optimization problems. This algorithm is based on the general framework for constrained optimization called Augmented Lagrangian methods.
- **Chapter 7** applies the method outlined in Chapter 6 to Model Predictive Control. We propose the design of quadratic model predictive control controllers subject to cardinality constraints on each control horizon.

Part II Estimation-Error Quantification

- **Chapter 8** proposes a novel method for estimation-error quantification for finite data estimation. The proposed method combines Maximum A Posteriori and Bayesian estimation techniques. The method provides a solution to the error quantification problem.
- **Chapter 9** applies the method developed in Chapter 8 to Moving Horizon Estimation. The proposed method provides a possible solution to the problem of computation of an entry cost.

Part III Modelling Model Uncertainty

- **Chapter 10** proposes a novel method for the problem of modelling model uncertainty. In this chapter, we propose a systematic methodology to describe a broad class of model uncertainties. This approach allows one to simultaneously estimate the nominal model and the associated uncertainty.

- **Chapter 7** draws conclusions and describes possible future research directions.

1.2 Publications by the Author

1. **Referred journals**

   (a) Directly related to the content of the thesis:

      i. R.A. Delgado (40%), J.C. Agüero (40%) and G.C. Goodwin (20%), " On Rank-Constrained Optimization," Submitted to Automatica.

      ii. R.A. Delgado (60%) and G.C. Goodwin (40%), "A Combined MAP and Bayesian Scheme for Finite Data and/or Moving Horizon Estimation" Automatica, Vol 50(4):1116-1121, April 2014

   (b) Other journal publications by the author having indirect relationship to the thesis:

      i. J.C. Agüero, W. Tang, J.I. Yuz, R.A. Delgado and G.C. Goodwin. " Dual time-frequency domain system identification" Automatica, Vol 48(12):3031-3041, Dec 2012.

      ii. J.C. Agüero, J.I. Yuz, G.C. Goodwin, and R.A. Delgado. "On the equivalence of time and frequency domain maximum likelihood estimation" Automatica, Vol 46(2):260-270, Feb 2010.

2. **Refereed conferences**

   (a) Directly related to the content of the thesis

      i. R.A. Delgado (35%), J.C. Agüero (35%) and G.C. Goodwin (30%) "A Rank-Constrained Optimization Approach: Application to Factor Analysis." In 19th IFAC World Congress, Cape Town, South Africa, 2014

      ii. R.P. Aguilera (25%), R.A. Delgado (25%), D. Dolz (25%) and J.C. Agüero (25%). "Quadratic MPC with ℓ_0-input constraint," In 19th IFAC World Congress, Cape Town, South Africa, 2014

iii. R.A. Delgado (35%), G.C. Goodwin (25%), R. Carvajal (20%) and J.C. Agüero (20%). "A Novel Approach to Model Error Modelling using the Expectation-Maximization Algorithm" In 51st IEEE Conference on Decision and Control (CDC), Maui, USA, 2012

(b) Other conference publications by the author having indirect relationship to the thesis

i. R. Carvajal, R.A. Delgado, J.C. Agüero and G.C. Goodwin. "An Identification Method for Errors-In-Variables Systems Using Incomplete Data," In 16th IFAC Symposium on System Identification, Brussels, Belgium, 2012

ii. R.A. Delgado, G.C. Goodwin and A. Feuer. "An on-line MUSIC algorithm with application to sparse signal reconstruction," In 20th International Symposium on Mathematical Theory of Networks and Systems (MTNS), Melbourne, Australia, 2012

iii. R.A. Delgado, J.C. Agüero, G.C. Goodwin and J.I. Yuz. "Two-degree-of-freedom anti-aliasing technique for wide-band networked control." In 18th IFAC World Congress, Milan, Italy, 2011

I hereby certify that the percentages given above for the papers directly related to the thesis accurately reflect the relative contribution of each author.

Prof. Graham C. Goodwin
Supervisor

Part I

Model Complexity Management

2

Rank-Constrained Optimization

2.1 Introduction

In this chapter we focus on methods to manage the complexity of a nominal model for a certain class of design and estimation problems. The complexity of the nominal model can be handled by several strategies. In most applications the complexity of the nominal model is managed by using the specific structure of the problem. We focus on applications where the complexity of the nominal model can be related to the rank of a particular matrix. More specifically, we focus on solving estimation problems where the estimation problem can be formulated as a rank-constrained optimization problem. We propose a general approach to rank-constrained optimization. This approach provides enough flexibility to handle additional convex constraints.

Rank-constrained optimization finds application in many areas including data modelling, systems and control, computer algebra, signal processing, psychometrics, machine learning, computer vision among others [71]. In many applications, the rank of a matrix is related to the complexity of a model. For example, in system identification, the order of a rational system is equal to the rank of an infinite dimensional Hankel matrix. In factor analysis the number of latent factors is equal to the rank of a covariance matrix. These and other areas of application will be studied in later chapters.

Rank-constrained optimization problems are known to be difficult, since they are inherently combinatorial in nature. Thus, there is no general procedure for solving them [69]. On the other hand, there exists a huge range of publications aimed at solving specific rank-constrained problems. In most problems, the rank-constraint is handled by taking advantage of the structure of the specific problem. For example, one approach to handle rank-constraints is based on Riemannian manifold optimization. This idea has been applied to linear regression [76], and to solving

Lyapunov equations [109]. Other methods use a greedy selection approach such as GECO [98] and ADMiRA [62]. Newton-like algorithms have also been developed aimed at solving Linear Matrix Inequalities subject to rank constraints [81]. Also, variable projection-type algorithms have been applied to structured low-rank approximation problems [72].

Rank-Constrained optimization problems are closely related to Rank-Minimization problems. Rank-Minimization problems have received increasing attention over the past few decades. The focus has centred on various approximations such as trace, nuclear norm and log-det heuristics (see e.g. [37–39]). There also exist some approaches that solve a Rank-Minimization problem exactly, see e.g [28], but the computational complexity is formidable even for small-size problems. Most heuristics developed for Rank-Minimization problems can be applied to Rank-Constrained problems. However, in these heuristics, the condition on the rank is not considered as a hard constraint.

A novel approach to dealing with rank constraints has been proposed in [28, 29, 59, 104]. The main idea is to exploit the fact that, for an $n \times n$ positive semidefinite matrix A, imposing the rank constraint $\operatorname{rank}\{A\} \leq r$ is equivalent to imposing the constraint that the sum of the $n - r$ smallest eigenvalues of A is equal to zero. This approach is an effective mechanism for imposing rank constraints. However, the idea has not been widely adopted. The methodology developed in the current chapter builds on this circle of ideas. In particular, we propose an approach to solve convex optimization problems subject to rank constraints. A key contribution of the work presented here is the fact that we consider rank constraints for general (i.e. not necessarily positive semidefinite) matrices. Our approach allows one to constrain the rank of a matrix whilst minimizing a cost function. The proposed approach is also applicable to the related problem of cardinality-constrained optimization.

The layout of the remainder of the chapter is as follows: In Section 2.2, we formulate the problem of interest. Section 2.3 revisits existing results. In Section 2.4, the proposed approach is presented. Finally, conclusions are drawn in Section 2.5.

Notation and basic definitions: $\operatorname{rank}\{A\}$ denotes the rank of a matrix A. $\lambda_i(A)$ denotes the i-th largest eigenvalue of a matrix A, $A \circ B$ denotes the Hadamard product of A and B, $A \succeq 0$ denotes that A is positive semidefinite, and $A \succeq B$ denotes that $A - B \succeq 0$. We represent the transpose of a given matrix A as $A^\top$. $\mathbb{S}^n$ denotes the set of symmetric matrices of size $n \times n$, and $\mathbb{S}^n_+$ the set of symmetric positive semidefinite matrices, i.e. $\mathbb{S}^n_+ := \{A \in \mathbb{S}^n | A \succeq 0\}$. $\mathbf{1}$ denotes a vector with only ones as entries. $\|\|_F$ denotes the Frobenius norm. $E^n(i,j)$ denotes to a $n \times n$ matrix with zeroes in all its entries except in entry (i,j) which has value $E^n_{ij} = 1$.

2.2 Problem Description

Consider the following rank-constrained optimization problem

$$\begin{aligned} \mathcal{P}_{rco}: \quad & \min_{\theta \in \mathbb{R}^p} \ f(\theta) \\ & \text{subject to } \ \theta \in \Omega \\ & \qquad \text{rank}\left\{G(\theta)\right\} \leq r \\ & \qquad G(\theta) \in \mathbb{S}_+^n \end{aligned}$$

where $\Omega \subset \mathbb{R}^p$ is a convex set, $f(\theta) : \mathbb{R}^p \to \mathbb{R}$ and $G(\theta) : \mathbb{R}^p \to \mathbb{S}_+^n$ are such that θ belongs to a convex set.

The rank-constrained optimization problem $\mathcal{P}_{rco}$ is known to be difficult, since it is combinatorial in nature. Also, notice that $\mathcal{P}_{rco}$ also covers some optimization problems subject to cardinality constraints. This is achieved, by considering $G(\theta)$ to have a diagonal structure. This latter aspect will be explained in detail in chapter 6.

The condition $G(\theta) \in \mathbb{S}_+^n$ can be relaxed in order to consider general non-square real matrices, $G(\theta)$, as shown later in chapter 4. However, for the sake of simplicity this generalization is not considered in the current chapter.

There is a large number of optimization problems that can be described by problem $\mathcal{P}_{rco}$. Examples include, Factor Analysis and several System Identification problems. These specific cases will be addressed later in the thesis.

2.3 Existing Results

Although problem $\mathcal{P}_{rco}$ is, in general, difficult there are several special cases where a solution can be efficiently obtained. One such case is the unconstrained low-rank approximation problem, where the Eckart-Young Theorem [36] provides a closed form solution

Theorem 2.3.1. *[36] Given a matrix $X \in \mathbb{R}^{m \times n}$, the goal is to find*

$$\hat{X} = \arg\min_{Z} \|X - Z\|_F \ s.t \ \text{rank}\left\{Z\right\} \leq r$$

The solution is given by the truncated Singular Value Decomposition, SVD, i.e. if $X = USV^\top$ is a SVD of X, then the minimizer is given by $\hat{X} = U_{1:r}S_{1:r,1:r}V_{1:r}^\top$ where $U_{1:r}$ denotes the first r columns of U, and $S_{1:r,1:r}$ denotes the submatrix composed of the first r rows and the first r

columns of S, i.e. the submatrix containing the r-largest singular values of X. The minimiser $\hat{X}$ is unique if and only if $\sigma_{r+1} \neq \sigma_r$.

The Eckart-Young Theorem has become the cornerstone of most existing approaches to rank constrained optimization. One of the main reasons for this choice is that an SVD can be computed in an efficient and numerically robust way. However, for other problems such as structured low-rank approximation, the SVD-based methods can be seen as a relaxation of the original rank-constrained problem [69].

Another approach to impose rank constraints has been followed by several authors [29, 59, 104]. These ideas are also related to the method first presented in [28]. The underlying idea of these methods is that, for a matrix $G \in \mathbb{S}^n_+$, imposing the rank constraint $\text{rank}\{G\} \leq r$ is equivalent to imposing the constraint that the sum of the $n-r$ smallest eigenvalues of G is equal to zero.

In order to describe the approach in more detail, consider the following definition

$$\Phi_{n,r} = \{W \in \mathbb{S}^n,\ 0 \preceq W \preceq I, \text{trace}(W) = n - r\} \tag{2.1}$$

This set corresponds to the convex hull of the rank-(n-r) projection matrices [29].

The set $\Phi_{n,r}$ can be used to compute the sum of the $(n-r)$-smallest eigenvalues. The following lemma establishes the connection.

Lemma 2.3.1. *Consider $G \in \mathbb{S}^n$ whose eigenvalues are $\lambda_1(G) \geq \cdots \geq \lambda_n(G)$, and $r \in \{1, 2, \ldots, n\}$ then*

$$\sum_{i=r+1}^{n} \lambda_i(G) = \min_{W \in \Phi_{n,r}} \text{trace}(W^\top G)$$

Proof. Direct from [82, Theorem 3.4] and by considering that

$$\sum_{i=r+1}^{n} \lambda_i(G) = \text{trace}(G) - \sum_{i=1}^{r} \lambda_i(G)$$

. ■

The above result leads to the following optimization problem for finding the $(n-r)$ smallest eigenvalues:

$$\mathcal{P}_1 : \quad \min_{W \in \Phi_{n,r}} \text{trace}(W^\top G)$$

Problem $\mathcal{P}_1$ has a closed-form solution. In fact, let us consider the diagonalisation $G = Q\Lambda Q^\top$, then the direction matrix $W = UU^\top$ is optimal, where U corresponds to the directions of Q corresponding to the $n - r$ smallest entries of the diagonal matrix Λ.

The conditions under which the solution to $\mathcal{P}_1$ is zero are interesting. Specifically, the equivalence between imposing the constraint $\operatorname{rank}\{G\} \leq r$ and requiring that $\sum_{i=r+1}^{n} \lambda_i(G) = 0$, is given by the following result:

Lemma 2.3.2. *The rank of a matrix $G \in \mathbb{S}_+^n$ is less than r, if and only if, there exists a $W \in \Phi_{n,r}$, such that*

$$\operatorname{trace}(W^\top G) = 0 \tag{2.2}$$

Proof. From Lemma 2.3.1 we have that

$$\sum_{i=r+1}^{n} \lambda_i(G) \leq \operatorname{trace}(W^\top G) \tag{2.3}$$

Since $G \in \mathbb{S}_+^n$, we have that $\sum_{i=r+1}^{n} \lambda_i(G) \geq 0$. Then, (2.2) provides a sufficient condition for $\operatorname{rank}\{G\} \leq r$. The necessity of (2.2) follows from the equality between the sum of the (n-r) smallest eigenvalues of G and the optimal value of the cost function in problem $\mathcal{P}_1$ established in Lemma 2.3.1. ■

In [29] Lemma 2.3.1 and Lemma 2.3.2 have been used to solve the following rank-constrained feasibility problem

$$\begin{aligned} \mathcal{P}_{feas}: \quad & \underset{G \in \mathbb{S}_+^n}{\text{find}} \quad G \\ & \text{subject to} \quad G \in \Omega \\ & \qquad\qquad \operatorname{rank}\{G\} \leq r, \end{aligned}$$

where Ω is a convex set. One of the contributions in [29] is to reformulate the above rank-constrained feasibility problem $\mathcal{P}_{feas}$, as the following minimization problem.

$$\begin{aligned} \mathcal{P}_2: \quad & \min_{G \in \mathbb{S}_+^n} \min_{W \in \Phi_{n,r}} \quad \operatorname{trace}(W^\top G) \\ & \text{subject to} \quad G \in \Omega \end{aligned}$$

Problem $\mathcal{P}_2$ can be (locally) solved by iteratively alternating minimization[1] between G and W. Specifically, given a current estimate $\hat{G}^m$ of G, at iteration m, then the optimization update is as

[1] This alternating minimization scheme, is also known as Alternate Convex Search, see e.g. [51].

follows:

$$\hat{W}^{m+1} = \arg\left\{ \min_{W \in \Phi_{n,r}} \operatorname{trace}(W^\top \hat{G}^m) \right\} \tag{2.4}$$

$$\hat{G}^{m+1} = \arg\left\{ \min_{G \in \mathbb{S}^n_+} \operatorname{trace}((\hat{W}^{m+1})^\top G) \quad s.t\ G \in \Omega \right\} \tag{2.5}$$

If condition (2.2) is reached then this indicates that the feasibility problem $\mathcal{P}_{feas}$ has been solved. However, the term $\operatorname{trace}(W^\top G)$ is multimodal. As a consequence, it follows that the alternating minimization (2.4)-(2.5) cannot be guaranteed to converge, in general, to the global minimum of $\mathcal{P}_2$, see e.g [51, §4]. Hence, the inability to obtain a solution via alternating minimization to (2.4)-(2.5) satisfying (2.2) does not imply the non-existence of such a solution.

It is worth mentioning that iterations of alternating minimization (2.4)-(2.5) can be easily implemented using standard optimization software such as CVX [52] or YALMIP [67].

2.4 An alternative representation for Rank-Constrained Optimization

In this section we describe an equivalent representation for the rank-constrained optimization problem $\mathcal{P}_{rco}$. Also, we analyze several strategies to solve this alternative representation of $\mathcal{P}_{rco}$.

The optimization problem $\mathcal{P}_{rco}$ is equivalent to the following optimization problem with bilinear constraint.

$$\begin{aligned} \mathcal{P}_{bi}: \quad & \min_{\theta \in \mathbb{R}^p} \min_{W \in \mathbb{S}^n} \ f(\theta) \\ & \text{subject to} \ \ \theta \in \Omega \\ & \qquad\qquad \operatorname{trace}(G(\theta)^\top W) = 0 \\ & \qquad\qquad G(\theta) \in \mathbb{S}^n_+ \\ & \qquad\qquad W \in \Phi_{n,r} \end{aligned}$$

A difficulty is that Problem $\mathcal{P}_{bi}$ is non-convex due to the presence of the bilinear constraint $\operatorname{trace}(G(\theta)^\top W) = 0$. However, this reformulation allows one to use several optimization tools, including local and global optimization methods.

In the following sections we explore some local and global optimization techniques to solve $\mathcal{P}_{bi}$.

2.4.1 Local Optimization

In this section, we describe a local optimization algorithm to solve the Rank-Constrained Optimization problem $\mathcal{P}_{rco}$. In this approach, we reformulate $\mathcal{P}_{rco}$ as the optimization problem $\mathcal{P}_{bi}$. Problem $\mathcal{P}_{bi}$ can be solved by using an iterative procedure that, in each iteration, solves a feasibility problem. In more detail, let $\theta^m \in \Omega$ be the current estimate for $\theta \in \Omega$, at iteration m. Then, problem $\mathcal{P}_{bi}$ can be solved, by iteratively solving the following related feasibility problem

$$\begin{aligned} \mathcal{P}_{inner}: \quad \min_{\theta\in\mathbb{R}^p} \min_{W\in\mathbb{S}^n} \quad & \text{trace}(W^\top G(\theta)) \\ \text{subject to} \quad & f(\theta) \leq f(\theta^{m-1})(1-\eta_m) \\ & \theta \in \Omega \\ & G(\theta) \in \mathbb{S}^n_+ \\ & W \in \Phi_{n,r} \end{aligned}$$

where $0 < \eta_m < 0.5$ is a parameter chosen by the user. If the feasibility problem $\mathcal{P}_{inner}$ cannot be solved, we set $\eta_{m+1} = 0.5\eta_m$ and proceed with the algorithm, until problem $\mathcal{P}_{inner}$ cannot be solved for any η_m above some specified lower bound.

In the method described above, the speed of convergence is controlled by the value of $\eta_m > 0$. Note that, choosing an $\eta_m > 0$ in $\mathcal{P}_{inner}$ is a mechanism to reduce $f(\theta)$ by a non-negligible amount at each step. An algorithm based on this idea to solve problem $\mathcal{P}_{rco}$, is described in Algorithm 2.1 (see Table).

Minimization of the objective function $\text{trace}(W^\top G(\theta))$ is difficult, due to its multimodal nature. The presence of a bilinear term in the objective function has been well-studied in the literature, (see e.g. [5, 41, 51]). It is well-known, for example, that an alternating minimization scheme does not necessarily converge to the global optimum of such an objective function [51, §4]. However, in many areas, the provably sub-optimal solution provided by alternating minimization is good enough to satisfy the requirements of the problem. For example, in chapter 3, a problem in the form of $\mathcal{P}_{inner}$ will be solved using an alternating minimization scheme. The results obtained by that procedure are competitive with the results obtained by other state-of-the-art algorithms.

2.4.2 Global Optimization

In this section, we investigate global approaches to solve problem $\mathcal{P}_{inner}$. Several general optimization methods can be applied such as, multi-start local optimization, simulated annealing,

Algorithm 2.1 Rank-constrained optimization algorithm.

Input: $0 < \eta_0 \leq 0.5$

Input: $\theta^0 \in \Omega$ such that rank $\left\{G(\theta^0)\right\} \leq r$

$m \leftarrow 0$

while $\eta_m \geq$ Tolerance **do**

Solve problem $\mathcal{P}_{inner}$.

if in $\mathcal{P}_{inner}$ $(\text{trace}(W^\top G(\theta)) \leq \epsilon_{feasible})$ **then**

Denote the solution as θ^{m+1}

$\eta_{m+1} \leftarrow \eta_m$

else

$\theta^{m+1} \leftarrow \theta^m$

$\eta_{m+1} \leftarrow 0.5\eta_m$

end if

$m \leftarrow m + 1$

end while

return $\theta = \theta^m$

among others. However, these randomized methods do not ensure (deterministic) convergence to a global optimum. On the other hand, deterministic global optimization methods, such as branch-and-bound methods do guarantee convergence to the global optimum. Thus, we propose a global optimization procedure to solve a variant of $\mathcal{P}_{inner}$. The procedure is a branch-and-bound algorithm that solves the following optimization problem. Given $\underline{G}, \overline{G}, \underline{W}, \overline{W} \in \mathbb{S}^n$, solve

$$
\begin{aligned}
\mathcal{P}_4: \quad \min_{\theta \in \mathbb{R}^p} \min_{W \in \mathbb{S}^n} \quad & \text{trace}(W^\top G(\theta)) \\
\text{subject to} \quad & f(\theta) \leq f(\theta^{m-1})(1 - \eta_m) \\
& \theta \in \Omega \\
& W \in \Phi_{n,r} \\
& G(\theta) \in \mathcal{S}^n \\
& \underline{W}_{i,j} \leq W_{i,j} \leq \overline{W}_{i,j} \\
& \underline{G}_{i,j} \leq G(\theta)_{i,j} \leq \overline{G}_{i,j} \\
& i, j = 1, \ldots, n
\end{aligned}
$$

Notice that the main difference between $\mathcal{P}_4$ and $\mathcal{P}_{inner}$ is that $\mathcal{P}_4$ is constrained to the hyper-rectangle

$$\begin{aligned}\mathcal{Q} :=& \{\theta \in \mathbb{R}^p, W \in \mathbb{S}^n | G(\theta) \in \mathbb{S}^n, \\ & \underline{W}_{i,j} \leq W_{i,j} \leq \overline{W}_{i,j}, \underline{G}_{i,j} \leq G(\theta)_{i,j} \leq \overline{G}_{i,j} \\ & , \forall i, j \in \{1, \ldots, n\}\}\end{aligned} \tag{2.6}$$

The additional upper and lower bounds on $G(\theta)$ and W, allow one to derive a lower bound for the term $\text{trace}(W^\top G(\theta))$, required for the development of a branch-and-bound algorithm. Note that the use of branch-and-bound algorithms to minimize bilinear objective function is not new, see for example [5, 41].

Remark 2.4.1. *Existing methods that exploit (similar results to) Lemma 2.3.1 and Lemma 2.3.2 [29, 59, 104] do not address the fact that the term* $\text{trace}(W^\top G)$ *requires a global optmization procedure. In [28] the issue of global optimization is addressed for the related problem of rank-minimization. The global optimization method used in the latter work is based on a Sum-of Squares approach. However, it is reported in [28] that the computational complexity of the algorithm is excessively large even for small size problems.*

In the following section we describe a branch-and-bound implementation to solve $\mathcal{P}_4$.

Branch-and-Bound Algorithm

Branch-and-Bound is a general procedure for solving global optimization problems. Branch-and-bound has been applied in several areas, but is particularly important in mixed integer programming and in other problems of a combinatorial nature; see [17, 26] for a general description and [27] for a historical review.

Branch-and-Bound algorithms find the global minimum of a function $h : \mathbb{R}^m \to \mathbb{R}$ over a hyper-rectangle $\mathcal{Q}_0$. Let $\mathcal{Q} \subseteq \mathcal{Q}_0$, and consider

$$\varphi(\mathcal{Q}) = \min_{\theta \in \mathcal{Q}} h(\theta)$$

The branch-and-bound algorithm then computes $\varphi(\mathcal{Q}_0)$, by using two functions $\varphi_{lb}(\mathcal{Q})$ and $\varphi_{ub}(\mathcal{Q})$ that provide a lower and upper bound of $\varphi(\mathcal{Q})$, i.e.

$$\varphi_{lb}(\mathcal{Q}) \leq \varphi(\mathcal{Q}) \leq \varphi_{ub}(\mathcal{Q})$$

The two bounding functions, $\varphi_{ub}(\mathcal{Q})$ and $\varphi_{lb}(\mathcal{Q})$, must satisfy $(\varphi_{ub}(\mathcal{Q}) - \varphi_{lb}(\mathcal{Q})) \to 0$ as the size of $\mathcal{Q}$ goes to zero. A branch-and-bound algorithm begins by partitioning the initial hyper-rectangle

$\mathcal{Q}_0$ into the union of p_0 hyper-rectangles, i.e. $\mathcal{Q}_0 = \cup_{q=1}^{p_0} \mathcal{Q}_q$. This partition of $\mathcal{Q}_0$ is subsequently refined in a sequence of stages. At stage k, the initial hyper-rectangle has been partitioned as $\mathcal{Q}_0 = \cup_{q=1}^{p_k} \mathcal{Q}_q$, where

$$\min_{1\leq q\leq p_k} \varphi_{lb}(\mathcal{Q}_q) \leq \varphi(\mathcal{Q}_0) \leq \min_{1\leq q\leq p_k} \varphi_{ub}(\mathcal{Q}_q)$$

The upper and lower bounds are improved in subsequent stages. The branch-and-bound algorithm stops when the difference between the upper and lower bound is less than or equal to a given tolerance constant δ, i.e.

$$\min_{1\leq q\leq p_k} \varphi_{ub}(\mathcal{Q}_q) - \min_{1\leq q\leq p_k} \varphi_{lb}(\mathcal{Q}_q) \leq \delta$$

The choice of the bounding functions $\varphi_{ub}(\mathcal{Q})$ and $\varphi_{lb}(\mathcal{Q})$ has a significant impact on the performance of the branch-and-bound algorithm. Typically, the upper bound $\varphi_{ub}(\mathcal{Q})$ is obtained via an easy-to-compute suboptimal solution for $\varphi(\mathcal{Q})$. However, finding an appropriate lower bound function, $\varphi_{lb}(\mathcal{Q})$, is a difficult task. Another important step in developing a branch-and-bound algorithm, is choosing a hyper-rectangle, $\mathcal{Q}_j$, over which the partition will be refined at each stage. A commonly used criterion is to choose the hyper-rectangle having lowest value of $\varphi_{lb}(\mathcal{Q}_j)$. Thus, the branch-and-bound algorithm focuses on refining the partition on the most promising parts of $\mathcal{Q}_0$. Under this criterion, no superfluous bound calculations take place after the optimal solution has been found [26], i.e. when $\varphi_{ub}(\mathcal{Q}_j) = \varphi(\mathcal{Q}_0)$ for a $Q_j \subset Q_0$.

To apply this idea to Problem $\mathcal{P}_4$ we need to find a suitable lower bound for the term $\text{trace}(W^\top G(\theta))$. To do this we utilize the ideas presented in [73] to construct convex under-estimators. In particular, we will use the following result

Theorem 2.4.1. *[5]*
Let $\Xi := \{\underline{x} \leq x \leq \overline{x}, \underline{y} \leq y \leq \overline{y}\} \subset \mathbb{R}^2$.
Then

$$x\,y \geq x\,\underline{y} + \underline{x}\,y - \underline{x}\,\underline{y} \tag{2.7}$$

$$x\,y \geq x\,\overline{y} + \overline{x}\,y - \overline{x}\,\overline{y} \tag{2.8}$$

Proof. See [5]. ∎

Then, a lower bound for the term $\text{trace}(W^\top G(\theta))$ can be found by considering the fact that $\text{trace}(G^\top W) = \mathbf{1}^\top (G \circ W)\mathbf{1}$, and that each entry of G and W is bounded from above and below, i.e. $\underline{G}_{ij} \leq G_{ij} \leq \overline{G}_{ij}$ and $\underline{W}_{ij} \leq W_{ij} \leq \overline{W}_{ij}$.

The above argument leads to the following bounding function over $\mathcal{Q}$

$$\begin{aligned}
\varphi_{lb}(\mathcal{Q}) \coloneqq \min_{\theta\in\mathbb{R}^p}\min_{W,Z\in\mathbb{S}^n} \quad & \mathbf{1}^\top Z\mathbf{1} \\
\text{subject to} \quad & f(\theta) \leq \eta_m \\
& \theta \in \Omega \\
& W \in \Phi_{n,r} \\
& (\theta, W) \in \mathcal{Q} \\
& (G(\theta))_{ij}\overline{W}_{ij} + \overline{G}_{ij}W_{ij} - \overline{G}_{ij}\overline{W}_{ij} \leq Z_{ij} \\
& (G(\theta))_{ij}\underline{W}_{ij} + \underline{G}_{ij}W_{ij} - \underline{G}_{ij}\underline{W}_{ij} \leq Z_{ij} \\
& \forall i,j \in \{1,\ldots,n\}
\end{aligned}$$

$$\varphi_{ub}(\mathcal{Q}) \coloneqq \operatorname{trace}(W^\top G(\theta)) \mid (\theta, W) = \arg\varphi_{lb}(\mathcal{Q}) \tag{2.9}$$

One can then use any suitable branch-and-bound algorithm.

The final branch-and-bound method used to solve problem $\mathcal{P}_4$ is described in Algorithm 2.2 (see Table).

In Algorithm 2.2, the partitioning of $\mathcal{Q}$ into $\mathcal{Q}_a$, $\mathcal{Q}_b$, $\mathcal{Q}_c$ and $\mathcal{Q}_d$, is achieved by splitting by half along the longest "G-width" $d_g = \max_{i,j}(\overline{G}_{ij} - \underline{G}_{ij})$, with optimal arguments (i_g, j_g). Also, by splitting by half along the longest "W-width" $d_w = \max_{i,j}(\overline{W}_{ij} - \underline{W}_{ij})$, with optimal arguments (i_w, j_w).

We denote by $\underline{G}, \overline{G}, \underline{W}, \overline{W}$ the bounds that determine $\mathcal{Q}$, and denote by $\underline{G}^a, \overline{G}^a, \underline{W}^a, \overline{W}^a$ the bounds that determine $\mathcal{Q}_a$. In an analogous fashion we define bounds for $\mathcal{Q}_b$, $\mathcal{Q}_c$ and $\mathcal{Q}_d$. Then, we set the upper and lower bound on each hyper-rectangle as

$$\begin{array}{llcll}
\underline{G}^a & = \underline{G} + \frac{d_g}{2}E^n(i_g, j_g) & ; & \overline{G}^a & = \overline{G} \\
\underline{W}^a & = \underline{W} & ; & \overline{W}^a & = \overline{W} - \frac{d_w}{2}E^n(i_w, j_w) \\
\hline
\underline{G}^b & = \underline{G} + \frac{d_g}{2}E^n(i_g, j_g) & ; & \overline{G}^b & = \overline{G} \\
\underline{W}^b & = \underline{W} + \frac{d_w}{2}E^n(i_w, j_w) & ; & \overline{W}^b & = \overline{W} \\
\hline
\underline{G}^c & = \underline{G} & ; & \overline{G}^c & = \overline{G} - \frac{d_g}{2}E^n(i_g, j_g) \\
\underline{W}^c & = \underline{W} & ; & \overline{W}^c & = \overline{W} - \frac{d_w}{2}E^n(i_w, j_w) \\
\hline
\underline{G}^d & = \underline{G} & ; & \overline{G}^d & = \overline{G} - \frac{d_g}{2}E^n(i_g, j_g) \\
\underline{W}^d & = \underline{W} + \frac{d_w}{2}E^n(i_w, j_w) & ; & \overline{W}^d & = \overline{W}
\end{array}$$

Algorithm 2.2 Branch and Bound algorithm to solve $\mathcal{P}_4$.

Input: $\mathcal{Q}_0$: Initial hyper-rectangle

$\mathcal{L} \leftarrow \{\mathcal{Q}_0\}$

$UB \leftarrow \varphi_{ub}(\mathcal{Q}_0)$

$LB \leftarrow \varphi_{lb}(\mathcal{Q}_0)$

while $UB > \epsilon_{feasible}$ and $\mathcal{L} \neq \emptyset$ **do**

Select $\mathcal{Q} \in \mathcal{L}$ with the such that $\varphi_{lb}(\mathcal{Q}) = LB$.

Split $\mathcal{Q}$ into Q_a,Q_b,Q_c and Q_d, along the longest edges.

$\mathcal{L} \leftarrow \{\mathcal{L} - \mathcal{Q}\} \cup \{\mathcal{Q}_a, \mathcal{Q}_b, \mathcal{Q}_c, \mathcal{Q}_d\}$

$LB \leftarrow \min_{\mathcal{Q} \in \mathcal{L}} \varphi_{lb}(\mathcal{Q})$

$UB \leftarrow \min_{\mathcal{Q} \in \mathcal{L}} \varphi_{ub}(\mathcal{Q})$

$\mathcal{Q}_{opt} \leftarrow \{\mathcal{Q} \in \mathcal{L} | \varphi_{ub}(\mathcal{Q}) = UB\}$

Eliminate from $\mathcal{L}$ all $\{\mathcal{Q} \in \mathcal{L} | \varphi_{lb}(\mathcal{Q}) > \epsilon_{feasible}\}$

end while

if $\mathcal{L} = \emptyset$ **then**

$\mathcal{Q}_{opt} = \emptyset$

end if

return $\mathcal{Q}_{opt}$

Improving the lower bound function

The branch-and-bound algorithm described above requires an upper and a lower bound for each entry in W and $G(\theta)$. These bounds are used to compute a lower bound function $\varphi_{lb}(\mathcal{Q})$. In rank-constrained optimization problems, constraints on the eigenvalues naturally appear, such as, $0 \preceq W \preceq I$. In problems such as Factor Analysis, this kind of constraint also appears on $G(\theta)$. In this chapter, the general method presented in [73] is used to construct a convex under-estimator of $\text{trace}(G^\top W)$, based on constraints on the positive semidefinite cone.

We first revisit the following known results.

Fact 2.4.1. *[12, Fact 8.21.12] Let $G, W \in \mathbb{S}^N_+$. Then $(G \circ W) \in \mathbb{S}^N_+$*

Fact 2.4.2. *[12, Fact 7.6.9] Let $G, W \in \mathbb{R}^{m\times n}$. Then* $\text{trace}(G^\top W) = \mathbf{1}^\top (G \circ W)\mathbf{1}$

The following result is useful in providing a tight lower bound function $\varphi_{lb}(\mathcal{Q})$.

Theorem 2.4.2. *Let $G_L, G_U, G, W_L, W_U, W \in \mathbb{S}^n$ be such that*

$$G - G_L \in \mathbb{S}^n_+ \qquad W - W_L \in \mathbb{S}^n_+ \tag{2.10}$$

$$G_U - G \in \mathbb{S}^n_+ \qquad W_U - W \in \mathbb{S}^n_+ \tag{2.11}$$

Then

$$\text{trace}(G^\top W) \geq \mathbf{1}^T (G \circ W_L + G_L \circ W - G_L \circ W_L)\mathbf{1} \tag{2.12}$$

$$\text{trace}(G^\top W) \geq \mathbf{1}^T (G \circ W_U + G_U \circ W - G_U \circ W_U)\mathbf{1} \tag{2.13}$$

Proof. From Fact 2.4.1, and (2.10) we have that

$$(G - G_L) \circ (W - W_L) \in \mathbb{S}^n_+ \tag{2.14}$$

which, after expanding terms, gives

$$G \circ W - (G \circ W_L + G_L \circ W - G_L \circ W_L) \in \mathbb{S}^n_+ \tag{2.15}$$

The same procedure can be used with (2.11) to obtain

$$G \circ W - (G \circ W_U + G_U \circ W - G_U \circ W_U) \in \mathbb{S}^n_+ \tag{2.16}$$

from the definition of a positive semidefinite matrix, $A \succeq 0$ if and only if $x^\top Ax \geq 0$, $\forall x$. Consider $x = \mathbf{1}$, then from (2.15) we have that

$$\mathbf{1}^\top (G \circ W - (G \circ W_U + G_U \circ W - G_U \circ W_U))\mathbf{1} \geq 0 \tag{2.17}$$

by rearranging and considering Fact 2.4.2 we obtain (2.12). The same procedure can be used with (2.16) to obtain (2.13). ■

Thus, if constraints (2.10)-(2.11) are present in the problem $\mathcal{P}_4$, the following constraints can be included when computing $\varphi_{lb}(\mathcal{Q})$

$$G \circ W_U + G_U \circ W - G_U \circ W_U \preceq Z \tag{2.18}$$

$$G \circ W_L + G_L \circ W - G_L \circ W_L \preceq Z. \tag{2.19}$$

2.5 Chapter Summary

We have proposed a general approach to rank-constrained optimization. The approach is based on the minimisation of the sum of the smallest (n-r) eigenvalues of a positive semidefinite matrix. We have developed local and global methods for this problem. Notice that the optimization method can be extended to consider non-convex functions $f(\theta)$ in problem $\mathcal{P}_{rco}$ by using the general optimization framework of Majorization-Minimization algorithms [57,61]. Areas for future research include: missing data problems; and using the proposed approach in binary and integer programming.

3

Factor Analysis

3.1 Introduction

In the previous chapter, we have presented a general method to solve rank-constrained optimization problems. One problem that fits into this category is that of Factor Analysis (FA). In this chapter we apply the method presented in chapter 2 to the problem of Factor Analysis.

In Factor Analysis a collection of N random variables are measured. It is assumed that these variables can be decomposed in two parts: a common part describing the co-movement between the random variables, and an idiosyncratic part describing the individual movement of each random variable. The common part is modelled as a linear combination of r random variables, called *factors*.

Factor Analysis is used in several areas including econometrics, psychometrics, psychology, among others. One of the main issues in the existing literature on Factor Analysis is that it is generally assumed that the idiosyncratic noises are uncorrelated. This is considered a very restrictive assumption [34, 103]. In this chapter we propose a novel approach that allows one to relax this assumption on the idiosyncratic noise. This approach is based on the method presented in chapter 2. The results obtained by the proposed method are shown to be competitive with the results obtained by other state-of-the-art algorithms.

As mentioned above, classical FA is based on the strict assumption that idiosyncratic movements must be uncorrelated, i.e. the idiosyncratic covariance must be diagonal [22]. There has been a renewed interest in stating a more general problem that, among others things, relaxes the assumption of diagonal idiosyncratic covariance. One approach is the "approximate factor model" [10, 103], where (by introducing the assumption that $N \to \infty$) "weak" correlation within the idiosyncratic covariance is allowed. An advantage of this approach is that it provides a theoretical framework for

use of standard FA tools when "weak" idiosyncratic correlation is present. This idea has also been considered when the factors have dynamics, see e.g. [30]. Another approach arises by exploiting the relationship with Principal Components Analysis (PCA). In PCA the aim is to describe most of the variability in the output by a few variables called *Principal Components*. Several tools developed for PCA can be applied to FA by simply normalizing the measurements before applying the method. In recent years, *Robust* PCA (RPCA) has received increasing attention. This approach, when applied to FA, allows decomposition between the common and idiosyncratic parts, under the assumption that the idiosyncratic covariance is "sparse".

RPCA does not require prior information about the number of factors or about the structure of the idiosyncratic covariance. Instead, a regularization parameter is included that manages the tradeoff between having a low-rank description for the common part (i.e. small number of factors) and having a small idiosyncratic part (i.e. small noise covariance, usually in terms of the ℓ_1 norm). In the FA framework, a big disadvantage of the existing RPCA methods is that there is no guarantee that the estimated idiosyncratic covariance matrix will be positive semidefinite. Another issue associated with the existing RPCA methods is that prior information about the number of factors cannot be included. This is seen as a disadvantage, since in FA, specialized methods have been developed to choose the number of factors.

In chapter 2 we have developed a general optimization method to handle rank-constraints. We denote this method as RCO, standing for Rank-Constrained Optimization.

In this chapter we propose an approach to Factor Analysis based on RCO. The method ensures that: (i) the number of factors is less than or equal to a pre-specified bound; (ii) the idiosyncratic covariance matrix is positive semidefinite; and (iii) it allows us to relax the assumption of diagonal idiosyncratic covariance, by assuming instead that this matrix is sparse.

The layout of the remainder of the chapter is as follows: In Section 3.2 we describe the Factor Analysis problem. Section 3.3 reviews existing methods for FA. The proposed approach is described in Section 3.4. In Section 3.5 we present a numerical example. Conclusions are drawn in Section 3.6.

3.2 Problem description

Here, and in the sequel, we use the following notation: $\lambda_i(A)$ denotes the i-th largest eigenvalue of a matrix A, $A \succeq 0$ denotes that the matrix A it is positive semidefinite. $\mathbb{S}^N$ denotes the set of

symmetric positive semidefinite matrices of size $N \times N$. Consider a measured output $y_k \in \mathbb{R}^N$, factors $f_k \in \mathbb{R}^r$, idiosyncratic noise $v_k \in \mathbb{R}^N$, and a model:

$$y_k = Af_k + v_k \tag{3.1}$$

where $A \in \mathbb{R}^{N \times n}$ is the matrix of factor loadings. We assume that f_k and v_k are mutually uncorrelated i.i.d. zero-mean Gaussian processes, with covariances Φ and Ψ, respectively. Thus, the measured output y_k is an i.i.d. zero-mean Gaussian process with covariance

$$\Sigma = A\Phi A^\top + \Psi \tag{3.2}$$

There are two important issues that are shared by (classical) FA and PCA: (i) any rotation of the factors will produce the same output characteristics, (ii) the components of the idiosyncratic noise v_k must be mutually uncorrelated, i.e. Ψ must be a diagonal matrix.

Remark 3.2.1. *As mentioned in the introduction to this chapter, the assumption that Ψ is diagonal is considered very restrictive [103]. Approximate factor models relax this assumption by considering that $N \to \infty$, $T \to \infty$. Under this assumption estimators (such as principal components and quasi maximum likelihood [34, 103]) can be shown to be consistent in the presence of "weak" idiosyncratic cross-correlation. However, it has been pointed out that when N is small, the presence of cross-correlation could severally deteriorate the performance of such estimators [16].*

3.3 Existing Methods

In this section we briefly review the more relevant existing methods for Factor Analysis.

3.3.1 Principal Components Analysis

A *principal components* (PC) estimator minimizes the residual sum of squares

$$\sum_{k=1}^{T} (y_k - Af_k)^\top (y_k - Af_k) \text{ subject to } A^\top A = I_r$$

The PC estimate $\hat{A}$ can be computed as the eigenvectors corresponding to the r-largest eigenvalues of $S = \sum_{k=1}^{T} y_k y_k^\top$. Then $\hat{f}_k = (\hat{A}^\top \hat{A})^{-1} \hat{A}^\top y_k$. This allows one to obtain the PC estimates in a computationally efficient way. However, there is a drawback, for fixed N and when $T \to \infty$, the PC estimator is inconsistent unless $\Psi = \sigma^2 I_N$.

As mentioned in the introduction, the PC estimator can be applied to FA provided the variables are a-priori normalized, i.e. such that the variables have unitary covariance. In fact, for the special

case when y_k is normally distributed, and $\Psi = \sigma^2 I$ the PC estimator is the maximum likelihood estimator. Also, it has been shown that when $N \to \infty$ and $T \to \infty$, the PC estimator is consistent even when there is "weak" cross-correlation in Ψ. For technical details see [10].

Recently, interest has turned to *Robust* PCA (RPCA), where the focus is to discover the structure of Ψ by minimizing the ℓ_1 norm. An advantage of this approach is that it allows non-diagonal Ψ. However, existing RPCA algorithms do not ensure $\Psi \succeq 0$. Thus, they are unsuitable for FA. Existing RPCA algorithms, such as [64], are based on the trace heuristic to induce the rank constraint on $A\Phi A^\top$. This means that the rank of the solution cannot be specified a priori.

3.3.2 Expectation-Maximization

Expectation-Maximization (EM) algorithms [32] have been successfully applied in several areas such as System Identification [1,31], Channel Estimation [21] and the computation of PC estimates [95].

EM algorithms are two-step iterative procedures designed to compute the maximum likelihood estimate. EM algorithms introduce the concept of complete data. The complete data is assumed to be composed of a set of measured variables, $\mathcal{Y}$, and a set of unmeasured variables known as the *hidden variables*, $\mathcal{H}$. A more detailed explanation of the EM algorithm is given in Appendix A.

In FA, the *factors* are a natural choice for hidden variables. An EM algorithm for FA can be described as follows: Given a current estimate of the parameters $(\hat{A}, \hat{\Psi})$ and setting $\Phi = I_r$, then the associated EM iteration is given by, see e.g. [118]:

E-step Compute

$$\mu_{f_k|\mathcal{Y}} = \hat{A}^\top (\hat{A}\hat{A}^\top + \hat{\Psi})^{-1} y_k \tag{3.3}$$

$$\Sigma_{f_k|\mathcal{Y}} = I_r - \hat{A}^\top (\hat{A}\hat{A}^\top + \hat{\Psi})^{-1} \hat{A} \tag{3.4}$$

where $\mu_{f_k|\mathcal{Y}}$ and $\Sigma_{f_k|\mathcal{Y}}$ are the mean and covariance of the factors, conditioned on $\mathcal{Y}$ assuming the estimates $(\hat{A}, \hat{\Psi})$ are available.

M-step

$$\hat{A} = \left(\sum_{k=1}^{N} y_k \mu_{f_k|\mathcal{Y}}^\top \right) \left(\sum_{k=1}^{N} \Sigma_{f_k|\mathcal{Y}} + \mu_{f_k|\mathcal{Y}} \mu_{f_k|\mathcal{Y}}^\top \right)^{-1} \tag{3.5}$$

$$\begin{aligned} \hat{\Psi} = \operatorname{diag}^* \{ 1/N \sum_{k=1}^{N} (y_k y_k^\top - y_k \mu_{f_k|\mathcal{Y}}^\top \hat{A}^\top - \hat{A} \mu_{f_k|\mathcal{Y}} y_k^\top \\ + \hat{A} (\Sigma_{f_k|\mathcal{Y}} + \mu_{f_k|\mathcal{Y}} \mu_{f_k|\mathcal{Y}}^\top) \hat{A}^\top) \} \end{aligned} \tag{3.6}$$

where the operator $\operatorname{diag}^*\{M\}$ returns a diagonal matrix with values equal to the values on the diagonal of M (i.e. if $L = \operatorname{diag}^*\{M\}$), thus $L_{ii} = M_{ii}$ and $L_{ij} = 0$ for $i \neq j$).

It is known that, under fairly general conditions, EM algorithms converge to a stationary point of the likelihood function [32]. The proof of convergence of EM algorithms is based, loosely speaking, on the construction of a surrogate function, such that whenever that surrogate function is improved, the likelihood function is also improved.

3.3.3 Minimum Rank Factor Analysis

Minimum Rank Factor Analysis (MRFA) was originally proposed in [104]. This approach minimizes the sum of the smallest $N - r$ eigenvalues of $\Sigma - \Psi$. This can be interpreted as minimizing the unexplained co-movement. An advantage of MRFA is that it includes the constraints $\Sigma - \Psi \succeq 0$ and $\Psi \succeq 0$. MRFA can be formulated as follows:

$$\begin{aligned} \mathcal{P}_{MRFA}: \quad & \underset{\Psi \in \mathbb{S}^N}{\text{minimize}} \;\; h(\Psi) = \sum_{i=r+1}^{N} \lambda_i(\Sigma - \Psi) \\ & \text{subject to} \;\; \Sigma - \Psi \succeq 0 \\ & \Psi \succeq 0 \\ & \Psi \text{ diagonal} \end{aligned}$$

The minimum of $h(\Psi)$ coincides with the minimum of the function

$$g(\Psi, W) = \operatorname{trace}(W^\top (\Sigma - \Psi)) \tag{3.7}$$

where $W = XX^\top$ and X is an $N \times (N - r)$ column-wise orthonormal matrix [104]. Then (3.7) can be monotonically minimized by alternating between optimizing Ψ and W [99, 104].

3.4 Factor Analysis with Correlated Errors

In Classical FA, the idiosyncratic covariance matrix Ψ must be diagonal. As mentioned earlier, this is considered very restrictive [34]. An alternative weaker assumption is to impose a sparsity assumption such that most of the entries in Ψ are zero. In recent years there has been increasing attention paid to finding sparse solutions. In this framework, ℓ_1-norm regularization has received a lot of attention due to its ability to deliver sparse solutions subject to certain restrictions [105]. In the related problem of PCA, ℓ_1 regularization has been applied to relax the diagonal assumption on Ψ. However, such methods are unsuitable for FA since they do not ensure that $\Sigma - \Psi \succeq 0$, $\Psi \succeq 0$. Also prior information about the number of factors cannot be included.

In this section we propose to drop the assumption of diagonal Ψ. Instead we assume that Ψ is "sparse". The problem then falls into the general class of rank-constrained optimization problems of the type described in chapter 2. This problem can then be solved by using the RCO method by choosing[1] $f(\Psi) = \|\Psi\|_1$, i.e.

$$\begin{aligned} \mathcal{P}_3 : \qquad & \min_{\Psi \in \mathbb{S}^N} \ \|\Psi\|_1 \\ \text{subject to } & \ \Sigma - \Psi \succeq 0 \\ & \ \Psi \succeq 0 \\ & \ \operatorname{rank}\{\Sigma - \Psi\} \leq r \end{aligned}$$

The proposed approach ensures both that $\Psi \succeq 0$, and $\Sigma - \Psi \succeq 0$.

3.5 Examples

In this section we present a simulation study illustrating the above ideas. We solve problem $\mathcal{P}_3$ using the method developed in chapter 2. In the first example we show the efficacy of the proposed local approach which solves $\mathcal{P}_{inner}$ using an alternating minimization scheme. In the second example, we show the efficacy of the proposed global optimization method in which we solve $\mathcal{P}_4$ instead of $\mathcal{P}_{inner}$, using a branch-and-bound algorithm.

3.5.1 Local Optimization Example

Consider a model as in (3.1)-(3.2), where $r = 3$, $N = 20$, $T = 100$, $\Phi = I_r$, the matrix of factor loadings, A, is constructed as a random matrix in which each matrix entry is independent and

[1] Let $A \in \mathbb{R}^{m \times n}$, then the matrix norm $\|A\|_1$ is defined as $\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^{m} |a_{ij}|$.

Method	$d(\cdot)$	Self-Time [s]	Total Time [s]
PCA	0.1992	0.0136	0.0136
RCO	0.1002	274.6596	274.6732
EM	0.1330	0.0062	0.0198
RPCA	0.2260	0.3803	0.3803

Table 3.1: Mean value over $N_{mc} = 100$ Monte Carlo simulations of the performance index $d(\cdot)$ and the execution time (Self-time and time to get initial estimate plus self-time).

normal distributed, i.e $A_{ij} \sim N(0,1)$. The covariance matrix Ψ is constructed such that the matrix entries satisfy $\Psi_{ij} = \tau^{|i-j|}$ with $\tau = 0.7$. In order to adjust the signal to noise ratio we fix the ratio $\|AA^\top\|_F/\|\Psi\|_F = 2$.

We run $N_{mc} = 100$ Monte Carlo simulations, with different realizations of f_k and v_k, and different factor loadings, A.

We solve problem $\mathcal{P}_3$ using the proposed local approach (see Algorithm 2.1), denoted as RCO. We compare the proposed methods against PCA, the EM algorithm described in section 3.3.2, and a RPCA method [64]. In RPCA we choose a regularization parameter such that the rank constraint is satisfied. Table 3.1 shows the mean value, over the Monte Carlo simulations, of the execution time and the performance index

$$d(P_m) = 1 - \frac{\text{trace}(AP_mA^\top)}{\text{trace}(AA^\top)}$$

where P_m corresponds to the orthogonal projection based on $\hat{A}$ for the method m. We can see that the proposed method delivers better results, but at the expense of larger computational effort.

Figure 3.1 compares the mean magnitude, denoted by $\bar{\Psi}$, of $\hat{\Psi} = \Sigma - \hat{A}\hat{A}^\top$ for all methods over the Monte Carlo simulations. The magnitudes are shown on a logarithmic scale. Figure 3.1b shows $\bar{\Psi}$ for PCA, which produces a $\bar{\Psi}$ matrix consistent with the assumption of diagonal Ψ. Note, however, that the true $\bar{\Psi}$ is not diagonal, see Figure 3.1a. Figures 3.1c shows the resultant $\bar{\Psi}$ for the proposed method, RCO. We see that non-diagonal entries appear more frequently than in other methods. Indeed, Figure 3.1c is close in appearance to the true result in Figure 3.1a. Figure 3.1d shows $\bar{\Psi}$ for EM. The results seem to be consistent with the assumption that matrix Ψ is diagonal. Again this is inconsistent with the true results in Figure 3.1a. Figure 3.1e shows $\bar{\Psi}$ for RPCA, that on average "discovers" a $\bar{\Psi}$ diagonal, but on average does not properly recover the non-diagonal entries.

Table 3.2: Factor Analysis: Performance index and cost function value for the local and global algorithms. The optimal value for the corresponding norm is in **bold**.

	$d(\cdot)$	$\|\widehat{\Psi}\|_1$	time [s]
RCO	0.066	4.46	23.83
RCO-G	0.066	**4.46**	310.19

3.5.2 Global Optimization Example

In this section we apply the proposed global optimization method to the problem of Factor Analysis. We denote as RCO-G the proposed approach in which we solve $\mathcal{P}_4$ instead of $\mathcal{P}_{inner}$, using a branch-and-bound algorithm, i.e. a global optimization method. Also, in RCO-G we use Theorem 2.4.2 by including contraints (2.18)-(2.19) when solving $\mathcal{P}_4$.

Consider a model of the form of (3.1)-(3.2), where $r = 1$, $N = 3$, $T = 100$, $\Phi = I_r$. The data is generated as in Example 3.5.1.

We compare the proposed local and global algorithms, RCO and RCO-G. In these methods, the cost function to optimize is $f(\Psi) = \|\Psi\|_1$. Table 3.2 shows the results. The global optimisation procedure reaches the global optimum with optimum cost 4.46. However, we notice that for this case, the local optimization method RCO also achieves the global optimum. Moreover, the execution time for RCO is less than the execution time for RCO-G.

In order show a case where the local optimization method achieves only a local optimum, we consider a variant of the problem where the cost function to be minimized is changed to[2] $f(\Psi) = \|\operatorname{vec}(\Psi)\|_1$. Table 3.3 shows the results. The global optimization algorithm RCO-G achieves the global optimum with optimum value 9.17. However, the local algorithm RCO now does not achieve the global optimum. Notice that the running time for RCO-G is much higher than the running time for RCO.

In Table 3.2 and Table 3.3, the RCO-G method makes use of Theorem 2.4.2 to improve the lower bound function $\varphi_{lb}(\mathcal{Q})$. However, when Theorem 2.4.2 is not used in RCO-G the total computation time for the results in Table 3.2 increases to $796.97[s]$ and the algorithm is terminated after $37361[s]$, for the results in Table 3.3. Hence we conclude that Theorem 2.4.2 improves the lower bound thereby significantly reducing the execution time for the RCO-G method.

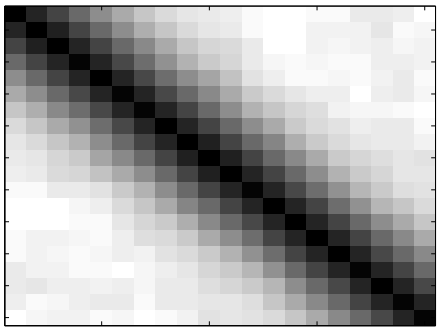
[2]Let $A \in \mathbb{R}^{m\times n}$, then the matrix norm $\|\operatorname{vec}(A)\|_1$ is given by $\|\operatorname{vec}(A)\|_1 = \sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|$.

Table 3.3: Factor Analysis: Performance index and cost function value for the local and global algorithms. The optimal value for the corresponding norm is in **bold**.

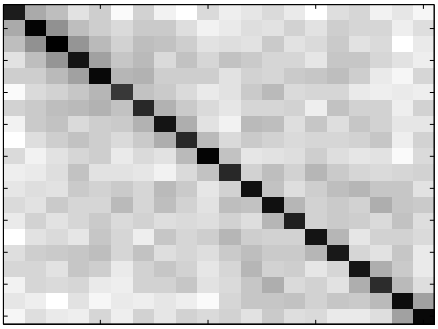
	$d(\cdot)$	$\|vec(\widehat{\Psi})\|_1$	time [s]
RCO	0.165	11.12	43.20
RCO-G	0.218	**9.17**	1768.31

3.6 Chapter Summary

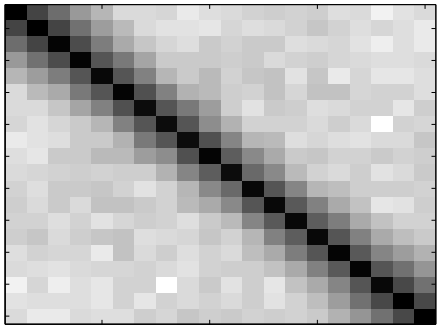
We have presented a novel Factor Analysis approach that ensures that: (i) the number of factors is less or equal to a pre-specified bound, and (ii) the idiosyncratic covariance matrix is positive semidefinite. The method allows one to relax the common assumption in classical FA that the idiosyncratic covariance, Ψ, is diagonal. We assume instead that Ψ is sparse and we estimate Ψ by minimizing its ℓ_1 norm. The numerical examples have shown the efficacy of the approach. We have shown that, in many cases, it outperforms existing methods.



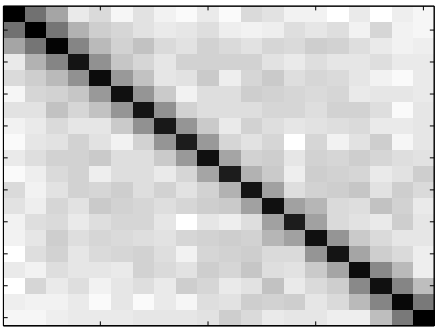
(a) True



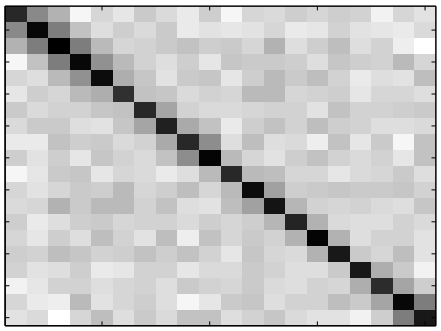
(b) PCA



(c) RCO



(d) EM



(e) RPCA

Figure 3.1: Mean value over $N_{mc} = 100$ Monte Carlo simulations on the magnitude of each entry of $\bar{\Psi}$ for each method

4

Rank Constraints for general real matrices

4.1 Introduction

In this chapter we describe one of the key results of this thesis, namely the extension of Lemma 2.3.2 to general real matrices (i.e not necessarily positive semi-definite matrices).

4.2 The key result

Lemma 2.3.2 provides a convenient way to impose rank constraints for positive semi-definite matrices. We next extend Lemma 2.3.2 to general matrices $G \in \mathbb{R}^{m\times n}$. This is described in the following Theorem

Theorem 4.2.1. *Let $G \in \mathbb{R}^{m\times n}$ then the following expressions are equivalent*

(i) $\text{rank}\,\{G\} \leq r$

(ii) $\exists W_R \in \Phi_{n,r}$, *such that* $GW_R = 0_{m\times n}$

(iii) $\exists W_L \in \Phi_{m,r}$, *such that* $W_L G = 0_{m\times n}$.

where

$$\Phi_{n,r} = \{W \in \mathbb{S}^n,\ 0 \preceq W \preceq I_n, \text{trace}(W) = n - r\} \tag{4.1}$$

Proof. Here we provide a sketch of the proof. A more detailed proof is given in Appendix B.

First we prove **(i)** $\Longrightarrow$ **(ii)**. Let $\operatorname{rank}\{G\} \leq r$ then there exist at least $n-r$ linearly independent vectors $u_i \in \mathbb{R}^n$ such that $Gu_i = 0$. Define $U = [u_1, \ldots, u_{n-r}] \in \mathbb{R}^{n\times r}$ having full column rank. Then we can construct a orthogonal projector[1], $W_R = UU^\dagger$ which satisfies the condition $\operatorname{rank}\{W_R\} = n-r$ and is such that $GW_R = 0$. Since W_R is an orthogonal projector it also satisfies $W_R \in \mathbb{S}^n$, $0 \preceq W_R \preceq I_n$ and $\operatorname{rank}\{W_R\} = \operatorname{trace}(W_R) = n-r$, i.e. $W_R \in \Phi_{n,r}$.

The procedure to prove **(i)** $\Longrightarrow$ **(iii)** is similar to the proof **(i)** $\Longrightarrow$ **(ii)**.

Next, we prove **(ii)** $\Longrightarrow$ **(i)**. For all $W_R \in \mathbb{S}^n$ such that $0 \preceq W_R \preceq I_n$, it is true that

$$\operatorname{trace}(W_R) \leq \operatorname{rank}\{W_R\} \tag{4.2}$$

On the other hand, by using *Sylvester's Inequality* (see e.g. [12, Proposition 2.5.9]), we have that

$$\operatorname{rank}\{G\} + \operatorname{rank}\{W_R\} \leq n + \operatorname{rank}\{GW_R\} \tag{4.3}$$

Then, by using (4.2), we have

$$\operatorname{rank}\{G\} + \operatorname{trace}(W_R) \leq n + \operatorname{rank}\{GW_R\} \tag{4.4}$$

Then by using the fact that $\operatorname{rank}\{GW_R\} = \operatorname{rank}\{0_{m\times n}\} = 0$ we obtain that

$$\operatorname{rank}\{G\} \leq n - \operatorname{trace}(W_R) \tag{4.5}$$

Since $W_R \in \Phi_{n,r}$, we have that $\operatorname{trace}(W_R) = n-r$. Then

$$\operatorname{rank}\{G\} \leq r \tag{4.6}$$

This completes the proof that **(ii)** $\Longrightarrow$ **(i)**. The procedure to prove **(iii)** $\Longrightarrow$ **(i)** is similar to the proof **(ii)** $\Longrightarrow$ **(i)**. ■

Theorem 4.2.1 is believed to be novel. As mentioned earlier, the approaches taken in [28,29] establish equivalence for the rank constraint only for semidefinite positive matrices. Because Theorem 4.2.1 treats general matrices it has the potential to impact broad classes of optimization problems. The most closely related approach on rank-constrained optimization, is the method described in [71,72] where the rank-nullity theorem is used to establish that, for a matrix $G \in \mathbb{R}^{m\times n}$,

$$\operatorname{rank}\{G\} \leq r \iff \text{there exist a full row rank matrix } U \in \mathbb{R}^{(m-r)\times m} \text{ such that } UG = 0 \tag{4.7}$$

However, requiring that U is full row rank is restrictive. For example, it may lead to the necessity of including additional non-convex constraints, such as $UU^\top = I_{m-r}$.

[1] Here † denotes the Moore-Penrose pseudoinverse, see e.g. [12, §6.1].

Notice that one of the key steps in proving Theorem 4.2.1 is that for all $W \in \mathbb{S}^n$ such that $0 \preceq W \preceq I$, it is true that $\text{trace}(W) \leq \text{rank}\,\{W\}$. This fact is a consequence of a stronger result that says that in the set of interest, $\{W \in \mathbb{S}^n | 0 \preceq W \preceq I\}$, the trace function is the largest convex function that is less than or equal to the rank function. This latter result is one of the key underlying ingredients in the development of the nuclear norm heuristic [37, 38], for the related problem of rank minimization.

We shall show in the sequel that Theorem 4.2.1 has a significant impact on methods for including rank constraints into optimization problems.

Theorem 4.2.1 allows us to relax the assumption that $G(\theta) \in \mathbb{S}^n_+$ by allowing us to consider general real matrices $G(\theta) \in \mathbb{R}^{m \times n}$. The following corollary establishes the fact that Lemma 2.3.2 is an special case of Theorem 4.2.1.

Corollary 4.2.1. *Let $G \in \mathbb{S}^n_+$ and $W \in \mathbb{S}^n_+$, then*

$$\text{trace}(GW) = 0 \iff WG = 0 \tag{4.8}$$

Proof. Since G and W are symmetric and positive semidefinite, then by *Cholesky decomposition*, see e.g. [12, Fact 8.9.37], there exist matrices $P \in \mathbb{R}^{n \times n}$ and $Q \in \mathbb{R}^{n \times n}$ such that

$$G = PP^\top \tag{4.9}$$

$$W = QQ^\top \tag{4.10}$$

we have that

$$\begin{aligned} \text{trace}(GW) &= \text{trace}(PP^\top QQ^\top) && (4.11)\\ &= \text{trace}(Q^\top PP^\top Q) && (4.12) \end{aligned}$$

Next, we recall that for $A \in \mathbb{R}^{m \times n}$ the *Frobenius norm* is defined by $\|A\|_F = \sqrt{\text{trace}(A^\top A)}$, see e.g. [12, page 547]. Then, we have

$$\text{trace}(Q^\top PP^\top Q) = \|P^\top Q\|_F^2 \tag{4.13}$$

and from the definition of a norm we have that $\|A\| = 0$ if and only if $A = 0$, see e.g. [12, Definition 9.2.1.]. Then we have that

$$\text{trace}(GW) = \|P^\top Q\|_F^2 = 0 \implies GW = 0 \tag{4.14}$$

This concludes the proof for $\text{trace}(GW) = 0 \implies GW = 0$. The proof for $GW = 0 \implies \text{trace}(GW) = 0$ is straightforward. ■

The generality of Theorem 4.2.1 will be used in the next chapters where we apply the proposed approach to the problems of impulse response estimation and cardinality-constrained optimization.

4.3 Chapter Summary

This Chapter has presented a novel approach to imposing rank constraints on general (i.e. not necessarily positive semi-definite) matrices. Application of these ideas will be studied in the next three chapters. However, it is believed that the method has wide applicability. Future research is aimed at exploring other potential areas of application.

5

Impulse Response Estimation

In chapter 4 we have presented a general method for solving rank-constrained optimization problems when the rank constraint applies to general (i.e. not necessarily positive semi-definite) matrices. In this chapter we apply the method to the problem of impulse response estimation.

Impulse response estimation finds application in many areas including systems and control, econometrics and acoustic and audio applications. For example, for dynamic systems, the impulse response can be used to model a linear dynamical system. Impulse responses are also used in acoustic and audio applications. In these applications, the impulse response is typically used to describe the acoustic characteristic of a location, such as a concert hall, see e.g. [75].

In system identification, the complexity of the impulse response can be expressed by the McMillan degree of the system. Moreover, the McMillan degree of the system can be related to the rank of a particular Hankel matrix. This approach has been explored by several authors, see e.g. [8, 53, 56]. All these methods use the nuclear-norm heuristic to impose the rank constraint. However, the nuclear-norm heuristic considers the rank-constraint as a soft constraint. In this chapter, we handle the rank-constraint as a hard constraint. Then, the impulse response estimation problem can be written as a rank-constrained optimization problem.

The layout of the remainder of the chapter is as follows: In Section 5.1 we describe the Impulse Response Estimation problem. In Section 5.2 we present a numerical example. Conclusions are drawn in Section 5.3.

5.1 Problem description

Consider the following input-output dynamical system

$$y_t = \sum_{k=1}^{\infty} g_k u_{t-k} + v_t \tag{5.1}$$

where v_t is white noise. The McMillan degree of the system is given by the rank of the Hankel matrix

$$\mathcal{H}_s(g) = \begin{bmatrix} g_1 & g_2 & g_3 & \cdots & g_{L-s+1} \\ g_2 & g_3 & g_4 & \cdots & g_{L-s+2} \\ g_3 & g_4 & g_5 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & g_{L-1} \\ g_s & g_{s+1} & \cdots & g_{L-1} & g_L \end{bmatrix} \tag{5.2}$$

We assume that the McMillan degree is known to be less than or equal to n. Hence to obtain an estimate, $\hat{g}_k$ for $k = 1, 2, \ldots, L$, we solve the following rank-constrained optimisation problem

$$\begin{aligned} \text{IRE}: \quad & \underset{\theta}{\text{minimize}} \ \|Y - \Phi\theta\| \\ & \text{subject to} \ \ \operatorname{rank} \mathcal{H}_s(\theta) \leq n \end{aligned}$$

where

$$Y = \begin{bmatrix} y_t & y_{t-1} & y_{t-2} & \cdots & y_M \end{bmatrix}^T \tag{5.3}$$

$$\Phi = \begin{bmatrix} u_{t-1}^T & u_{t-2}^T & \cdots & u_{t-L}^T \\ u_{t-2}^T & u_{t-3}^T & \cdots & u_{t-L-1}^T \\ \vdots & \vdots & \vdots & \vdots \\ u_{t-M}^T & u_{T-M-1}^T & \cdots & u_{t-M-L}^T \end{bmatrix} \tag{5.4}$$

$$\theta = \begin{bmatrix} g_1 & g_2 & \cdots & g_L \end{bmatrix}^T \tag{5.5}$$

We note that IRE is a rank-constrained optimization problem where the rank constraint is applied to a general real matrix. The problem can be solved by the methods developed in Chapter 4.

5.2 Numerical example

In this section, we present a simple example to show the efficacy of the proposed approach. The data was generated using the following third order model,

$$G_0(z) = \frac{0.18(z - 0.95)}{(z - 0.7)(z - 0.8)(z - 0.85)}$$

Table 5.1: Impulse Response Estimation: norm of the error and Singular values of $\mathcal{H}_s(\theta)$ for the true impulse response and for the estimates given by LS, RCO and NN methods.

	True	LS	RCO	NN
Error	1.4642	1.1938	1.2211	1.3675
σ_1	1.0922	1.0857	1.0846	0.9789
σ_2	0.3339	0.3914	0.3880	0.2963
σ_3	0.0120	0.0507	0.0195	0.0015
σ_4	0	0.0455	0	$2.40 \cdot 10^{-8}$
σ_5	0	0.0254	0	0
σ_6	0	0.0085	0	0

Obviously the associated impulse-response has infinitely many terms, but the McMillan degree is 3.

We are given $N = 40$ samples contaminated with additive white noise, $v_t \sim N(0, 0.1^2)$. Within the framework of problem IRE, we consider $n = 3$, $L = 12$, $M = N - L$, $\Phi \in \mathbb{R}^{M \times L}$, $s = 6$ and thus $\mathcal{H}_s(\theta) \in \mathbb{R}^{s \times (L-s)+1}$.

We obtain an estimate of $\hat{\theta}$ using the proposed approach, RCO, and compare it with the Least Squares (LS) estimate. Also we include in the comparison the Nuclear-Norm (NN) heuristic [37], using a regularization parameter $\mu = 4.5/\sqrt{L}$, which approximately imposes the rank constraint.

Table 5.1 shows the error norm, $\|Y - \Phi\theta\|$, and the singular values of $\mathcal{H}_s(\theta)$ for all estimates. The lowest error norm is, of course, achieved by the LS method since it is unconstrained. Note, however, that the resultant model does not satisfy the known McMillan degree constraint. Also, the resultant model may not be the best for other purposes e.g. prediction. Notice that the proposed RCO method achieves a error norm that is less than the error norm for the NN method. The singular values of values of $\mathcal{H}_s(\theta)$, σ_i, are shown as zero if they are smaller than 10^{-9}. Note that the method RCO achieves the required rank constraint, whereas the rank constraint is only approximated by the LS method. Notice that the NN method impose the rank constraint. However, there is seem to be a loss of performance when imposing the rank constraint via nuclear-norm heuristic. This loss of performance has been previously reported in [70].

5.3 Chapter summary

In this brief chapter we have applied the method described in chapter 2 to the problem of impulse response estimation. The numerical example shows the efficacy of the proposed method, and that it is competitive with state-of-the-art methods, in particular, the one based on the nuclear-norm heuristic.

6

Cardinality-Constrained Optimization

6.1 Introduction

In this chapter we study cardinality-constrained optimization problems. Cardinality[1] constraints are shown to be a particular case of rank constraints. In particular, we apply the general results presented in chapter 4 to the problem of cardinality-constrained optimization.

Cardinality-constrained optimization problems are one of the possible approaches to *sparsity* gender problems. There exist a large number of problems that can be formulated as cardinality-constrained optimization problems. For example, matching pursuit and several compressive sensing problems, see e.g. [107]. Moreover, problems such as matching pursuit with unconstrained dictionary, have been proven to be NP-hard [107]. A traditional approach to solve cardinality-constrained problems is via Greedy algorithms. Greedy algorithms are, in general, computationally inexpensive and, under certain conditions, various guarantees can be established regarding the distance between the suboptimal and optimal solution, see e.g. [107].

Within the framework of *sparsity* gender problems, cardinality-minimization problems have received increasing attention , see e.g. [20], where the ℓ_1-norm heuristic is used to promote sparsity.

In this chapter we present an alternative form to represent cardinality constraints. We also present a novel method to solve cardinality-constrained optimization problems.

The layout of the remainder of the chapter is as follows: In Section 6.2 we describe the proposed

[1] We use *cardinality* to refer to the number of non-zero elements of a vector. This is also known in the literature as the ℓ_0 quasi-norm of a vector, see e.g. [20, 107].

approach to cardinality-constrained optimization. Section 6.4 describes how to include group constraints into the cardinality-constrained optimization problem. Conclusions are drawn in Section 6.5.

6.2 An alternative representation of cardinality constraints

In this section we present an alternative representation of cardinality constraints, which is derived from the results presented in chapter 4. Then, we apply this representation to express the cardinality-constrained optimization as an optimization problem with a bilinear constraint.

Cardinality constrained optimization problems thus can be seen as a particular case of rank-constrained optimization problems. This is done by considering that the matrix to be rank-constrained is square and has a diagonal structure. Then, we use the general result from Theorem 4.2.1 to impose the resultant cardinality constraint.

Corollary 6.2.1. *Let $x \in \mathbb{R}^n$, then the following expressions are equivalent*

(i) $\operatorname{card} x \leq r$

(ii) *there exists a* $w \in \{w \in \mathbb{R}^n |\ 0 \leq w_i \leq 1, i = 1, \ldots, n; \sum_{i=1}^{n} w_i = n - r\}$, *such that* $x_i w_i = 0$ $i = 1, \ldots, n$.

Proof. Consider the following definition $G = \operatorname{diag}\{x\} \in \mathbb{R}^{n \times n}$, i.e.

$$G = \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ 0 & x_2 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & x_{n-1} & 0 \\ 0 & \cdots & 0 & 0 & x_n \end{bmatrix} \tag{6.1}$$

Notice that from construction $\operatorname{rank}\{G\} = \operatorname{card} x$. From Theorem 4.2.1, we have that $\operatorname{rank}\{G\} \leq r$, if and only if, there exist a $W \in \{W \in \mathbb{S}^n |\ 0 \preceq W \preceq I_n; \operatorname{trace}(W) = n - r\}$ such that $GW = 0$. Since G is diagonal, without loss of generality, then we can consider that $W = \operatorname{diag}\{w\}$. This can be easily seen by defining $C = GW$ and considering that W is symmetric. Note that, since G is diagonal, $C_{ij} = G_{ii}W_{ij}$ and $C_{ji} = G_{jj}W_{ji}$. If $G_{ii} = G_{jj} = 0$ for $i \neq j$ then $W_{ij} = W_{ji}$ can take any value, including zero, and still satisfy $C_{ij} = C_{ji} = 0$. If $G_{ii} \neq 0$ then $W_{ij} = W_{ji} = 0$ in order to satisfy that $C_{ij} = 0$. Finally, conditions on w are directly derived from conditions on W. ■

We note in passing that this representation of cardinality constraints is related to the results reported in [28, 84] for cardinality minimization problems.

Next, we consider the following cardinality constrained optimization problem

$$\begin{aligned} \mathcal{P}_{card}: \quad & \underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; f(x) \\ & \text{subject to} \;\; \operatorname{card} x \leq r \end{aligned}$$

We note from the above discussion that this can be formulated as a rank-constrained optimization problem. Moreover, from corollary 6.2.1, the problem can then be formulated as a optimization problem subject to bilinear constraints as follows

$$\begin{aligned} \mathcal{P}_{cardequiv}: \quad & \underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; f(x) \\ & \text{subject to} \;\; x_i w_i = 0; \quad i = 1,\ldots,n \\ & \qquad\qquad\;\; 0 \leq w_i \leq 1; \quad i = 1,\ldots,n \\ & \qquad\qquad\;\; \mathbf{1}^\top w = n - r \end{aligned}$$

6.3 Numerical Example

In this section we present a numerical example to show the efficacy of the proposed method to impose cardinality constraints.

6.3.1 Formulation

Assume we are given a set of noisy measurements $y \in \mathbb{R}^n$, and a matrix $A \in \mathbb{R}^{n\times m}$. We assume the data satisfies the following model

$$y = Ax + v \tag{6.2}$$

where v is a zero-mean Gaussian white noise with variance $\sigma^2 I_n$. Our goal is to estimate the vector $x \in \mathbb{R}^m$. Say that we know that the cardinality of x is less than r, but do not know which entries are zero. The problem is of the "sparsity" gendre, see e.g. [20]. The problem can be reformulated as the following optimisation problem

$$\begin{aligned} \text{CARDI}: \quad & \underset{x}{\text{minimize}} \;\; \|y - Ax\|_2 \\ & \text{subject to} \;\; \operatorname{card} x \leq r \end{aligned}$$

where $r < m < n$. Notice that A is a tall matrix. We recognise this as exactly the problem discussed in section 6.2. Moreover, we know that it is a special case of the problem $\mathcal{P}_{card}$. Thus,

we rewrite problem CARDI as the following optimization problem

$$\begin{aligned} \text{CARDIeq}: \quad \underset{x\in\mathbb{R}^m, w\in\mathbb{R}^m}{\text{minimize}} \quad & \|y-Ax\|_2^2 \\ \text{subject to} \quad & x_i w_i = 0 \quad i=1,\cdots,m \\ & 0 \le w_i \le 1 \quad i=1,\cdots,m \\ & \mathbf{1}^\top w = m-r \end{aligned}$$

There are many methods that could be used to solve the problem CARDIeq including the techniques utilised in Chapter 3 and 5. To illustrate the power of the general methodology, as opposed to the particular solution method, we will utilize an alternative method as described below.

6.3.2 Augmented Lagrangian method

Problem $\mathcal{P}_{cardequiv}$ can be solved using a range of non-convex optimization methods. To illustrate, we consider the constrained optimization framework of Augmented Lagrangian methods [55,86,94]. In particular, we use the Augmented Lagrangian algorithm described in [14]. We solve problem $\mathcal{P}_{cardequiv}$ by using the Augmented Lagrangian method described in Algorithm 6.1. We denote this method as CCO-AL, that stands for Cardinality-Constrained Optimization via Augmented Lagrangian method.

We next briefly describe Augmented Lagrangian methods. Consider the following optimization problem

$$\begin{aligned} \mathcal{P}_{al}: \quad \underset{z\in\mathbb{R}^n}{\text{minimize}} \quad & f(z) \\ \text{subject to} \quad & h(z)=0 \\ & g(z)\le 0 \\ & z\in\Omega \end{aligned} \tag{6.3}$$

where $f:\mathbb{R}^n\to\mathbb{R}$, $h:\mathbb{R}^n\to\mathbb{R}^{n_e}$, $g:\mathbb{R}^n\to\mathbb{R}^{n_i}$ are continuous and $\Omega\subset\mathbb{R}^n$ is compact.

The Augmented Lagrangian function will be defined by, see e.g. [13,14]

$$L_\rho(z,\lambda,\mu) = f(z) + \frac{\rho}{2}\left(\sum_{i=1}^{n_e}\left[h_i(z)+\frac{\lambda_i}{\rho}\right]^2 + \sum_{i=1}^{n_i}\left[\max(0, g_i(z)+\frac{\mu_i}{\rho})\right]^2\right) \tag{6.4}$$

for all $z\in\Omega$, $\rho>0$, $\lambda\in\mathbb{R}^{n_e}$, $\mu\in\mathbb{R}^{n_i}_+$.

The Augmented Lagrangian method is an iterative procedure that, at each iteration, given a ρ_k, $\lambda^k$and μ^k optimizes $L_{\rho_k}(z,\lambda^k,\mu^k)$ over z. The quantities ρ_k, $\lambda^k$and μ^k are then updated using

simple rules. The Augmented Lagrangian methodology is guaranteed to converge to a stationary point satisfying the Karush-Kuhn-Tucker conditions under quite general conditions, see e.g. [9].

Algorithm 6.1 shows an implementation of the Augmented Lagrangian algorithm.

Algorithm 6.1 Augmented Lagrangian algorithm [13, 14].

Initialize Let $\lambda_{min} \leq \lambda_{max}$, $\mu_{max} > 0$, $\gamma > 1$, $0 < \tau < 1$. Let $\lambda_i^1 \in [\lambda_{min}, \lambda_{max}]$ for $i = 1, \ldots, n_e$, $\mu_i^1 \in [0, \mu_{max}], i = 1, \ldots, n_i$ and $\rho_1 > 0$. Initialize $k = 1$

Step1 Solve the subproblem

$$\min_{z \in \Omega} L_{\rho_k}(z, \lambda^k, \mu^k) \tag{6.5}$$

Step2 Test feasibility and Optimality

if

$$\|h(z^k)\| + \|\max\{g(z^k), 0\}\| \leq \epsilon_{feas} \tag{6.6}$$

stop algorithm declaring *Solution Found*.

Step3 Define

$$V_i^k = \max\left(g_i(z^k), -\frac{\mu_i^k}{\rho_k}\right), \qquad , i = 1, \ldots, n_i \tag{6.7}$$

If $k = 1$ or

$$\max\{\|h(z^k)\|_\infty, \|V^k\|_\infty\} \leq \tau \max\{\|h(z^{k-1})\|_\infty, \|V^{k-1}\|_\infty\}$$

define $\rho_{k+1} = \rho_k$. Otherwise $\rho_{k+1} = \gamma \rho_k$

Step4 Compute for $i = 1, \ldots, n_e$,$j = 1, \ldots, n_i$

$$\lambda_i^{k+1} = \min\{\max\{\lambda_{min}, \lambda_i^k + \rho_{k+1} h_i(z^k)\}, \lambda_{max}\} \tag{6.8}$$

$$\mu_j^{k+1} = \min\{\max\{0, \mu_j^k + \rho_{k+1} g_j(z^k)\}, \mu_{max}\} \tag{6.9}$$

set $k = k + 1$ and go to **Step1**.

6.3.3 Numerical results

Consider the model (6.2), where $n = 40$, and $\sigma^2 = 0.1$. The matrix A is generated at random. The vector to be estimated is generated such that the cardinality of x is less than or equal to r. We

	case 1		case 2	
Method	$\|y - Ax\|_2$	time [s]	$\|y - Ax\|_2$	time [s]
True	1.9399	-	1.9078	-
ℓ_1	2.1156	0.22	2.1368	0.24
CCO-AL	1.9165	3.16	1.83	4.93

Table 6.1: Constraining cardinality: Norm of the error of the estimates obtained by l_1 minimization and RCO method.

consider two cases: case 1 in which $m = 8$ and $r = 2$; and case 2 in which $m = 16$ and $r = 4$.

We consider two methods: (i) We solve problem CARDI using the ℓ_1-norm heuristic which is commonly used to promote sparsity, see [20]. This is implemented using the CVX parser [52]. For the ℓ_1-norm heuristic, we use as regularisation parameter $\mu = 1/\sqrt{n}$. (ii) We solve problem CARDIeq using the CCO-AL method described in the previous section.

Table 6.1 shows the error $\|y - Ax\|_2$, and the execution time for all methods. Note that the proposed method delivers better results than those obtained via the ℓ_1-norm heuristic. The non-zero elements obtained by both methods coincide with the non-zero elements of the "true vector" used to generate the data, save that, in case 1, the ℓ_1-norm heuristic delivers an estimate that has cardinality 3, where the non-zero elements include the non-zero elements of the "true vector".

6.4 Group Constraints

The idea of group-handling in sparse representations has received increasing attention in the last decade, see e.g. [60, 115]. These methods are based on the ℓ_1-norm heuristic. In this section we explore the possibility of including group constraints into cardinality-constrained optimization problems.

In problem CARDIeq, including the auxiliary variable w to manage the cardinality of the vector x opens a new paradigm in cardinality constrained optimization problems. At the optimum, the variable w is a binary variable having value $w_i = 1$ in those elements corresponding to $x_i = 0$. Then, constraints over w can be included in the optimization problem to manage how the zero and non-zero elements of x interact. For example, we formulate the problem by seeking to minimize $\|y - Ax\|_2$ subject to $\operatorname{card} x \leq r$, with $x \in \mathbb{R}^m$, with m and r being even numbers. Moreover, we split the vector x in two groups: Group $G1$ consisting of the first $m/2$ elements of x, and group

	ℓ_1	CCO-AL	Group-RCO
$\|y - Ax\|_2$	5.1799	3.5734	4.7612

Table 6.2: Group constraints: Norm of the error of the estimates obtained by ℓ_1-norm, CCO-AL and Group-RCO methods.

$G2$ consisting of the last $m/2$ elements of x. Our assumption is that, if the ith-element of $G1$ is non-zero, i.e. $x_i \neq 0$, then the ith-element of $G2$ must be zero, i.e $x_{m/2+i} = 0$, and vice versa. The associated optimization problem can be formulated as

$$\begin{aligned} \underset{x\in\mathbb{R}^m, w\in\mathbb{R}^m}{\text{minimize}} \quad & \|y - Ax\|_2^2 \\ \text{subject to} \quad & x_i w_i = 0 \qquad i = 1, \cdots, m \\ & 0 \leq w_i \leq 1 \qquad i = 1, \cdots, m \\ & \mathbf{1}^\top w = m - r \\ & w_i + w_j \geq 1; \qquad i = 1, \ldots, m/2, j = m/2 + i \end{aligned}$$

We denote the proposed approach as Group-RCO.

6.4.1 Numerical Example

Consider the model (6.2), where $n = 30$, $m = 20$, $r = 4$, $\sigma^2 = 0.5$ and matrix A is generated at random. We consider the CCO-AL method, the Group-RCO method described above, and the ℓ_1-norm heuristic with regularization parameter $\mu = 10/\sqrt{n}$. The resulting ℓ_1-norm estimate have cardinality $r = 4$. Table 6.2 shows the error $\|y - Ax\|_2$ for all methods. Notice that the CCO-AL, and Group-RCO methods deliver better results than the ℓ_1-norm heuristic. Table 6.3 shows the estimates obtained by all methods. Notice that the results obtained by CCO-AL are not required to satisfy the group constraints. This is clear by noticing that the 10th and 20th-elements of the estimate have non-zero values. However, Group-RCO satisfies the group constraints, and the resulting estimate have cardinality $2 \leq r = 4$.

6.5 Chapter Summary

In this chapter we have extended the results of chapter 2 to the problem of cardinality-constrained optimization. We have developed an algorithm to solve cardinality-constrained optimization problems. This algorithm is based on the general framework for constrained optimization of Augmented Lagrangian methods. We have also explored the ability to include group constraints into

element index	ℓ_1	CCO-AL	Group-RCO
1	0	0	0
2	1.1688	1.6562	1.5501
3	0	0	0
4	0	0	0
5	0	0	0
6	0	-0.2208	0
7	0	0	0
8	0	0	0
9	0	0	0
10	0.8980	1.3535	1.2382
11	0	0	0
12	-0.0797	0	0
13	0	0	0
14	0	0	0
15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0
19	0	0	0
20	0.0443	0.5739	0

Table 6.3: Group constraints: Estimates obtained by ℓ_1-norm, CCO-AL and Group-RCO methods.

cardinality-constrained optimization problems.

7

Model Predictive Control with sparse-input constraints

In chapter 6 we have presented a general method for solving cardinality-constrained optimization problems. In this chapter we apply the method described in chapter 6 to Model Predictive Control (MPC).

Controlling a process using a reduced number of active inputs has several advantages. For example, it can be used to reduce the energy consumption of self-powered devices due to transmission, or to reduce network bandwidth usage. In this chapter we propose the design of quadratic MPC controllers subject to cardinality constraints on each control horizon.

A stability analysis of the resultant solution is beyond the scope of the current thesis. The complete problem formulation and an initial stability analysis can be found in [3].

The layout of the remainder of the chapter is as follows: In Section 7.1 we describe the problem. In Section 7.2 we present a simulation study. Conclusions are drawn in Section 7.3.

7.1 Problem description

Consider the following discrete-time linear time-invariant system:

$$x_{k+1} = Ax_k + Bu_k, \tag{7.1}$$

where $x_k \in \mathbb{R}^n$ is the system state , $u_k \in \mathbb{R}^m$ is the control input vector. The pair (A, B) is assumed to be stabilizable where the matrix A is not necessarily Schur stable.

We are seeking to control the system (7.1) with a reduced number of active inputs, $\gamma \leq m$. To this

end, one needs to design a controller which can provide the best possible actuation considering only γ active inputs while the remaining $\gamma - m$ inactive inputs take a null value.

We use $\sigma \in \mathbb{R}^m$ to denote a binary vector which indicates the active and inactive inputs, i.e., the $i-th$ component of σ is given by:

$$\sigma_i = \begin{cases} 1 & \text{if } u_i \text{ is active,} \\ 0 & \text{otherwise } (u_i = 0), \end{cases} \tag{7.2}$$

for all $i \in \{1, \ldots, m\}$. Thus, the number of non-zero elements of vector σ is $|\sigma|_0 = \gamma$.

To formulate the MPC optimal problem, we consider the following quadratic cost function

$$V_N(x, \vec{u}) = \hat{x}_N^\top P \hat{x}_N + \sum_{j=0}^{N-1} \hat{x}_j^\top Q \hat{x}_j + \hat{u}_j^\top R \hat{u}_j \tag{7.3}$$

where $\hat{x}$ and $\hat{u}$ stand for the predicted values of the system state and input respectively, and N is the prediction horizon. The matrices Q, R, and P are assumed to be positive definite. The vector $\vec{u}$ contains the tentative control action over the prediction horizon, i.e.,

$$\vec{u} = \left[\hat{u}_0^\top, \ldots, \hat{u}_{N-1}^\top\right]^\top \in \mathbb{R}^{Nm},$$

The optimization problem of interest for the current state, $x_k = x$, is given as

$$\mathcal{P}_{MPC}: \qquad V_N^{op}(x) = \underset{\vec{u} \in \mathbb{R}^{Nm}}{\text{minimize}} \ \{V_N(x, \vec{u})\} \tag{7.4}$$

$$\text{subject to } \ \hat{x}_{j+1} = A\hat{x}_j + B\hat{u}_j, \tag{7.5}$$

$$\text{card}\, \hat{u}_j \leq \gamma, \tag{7.6}$$

$$\hat{x}_N \in \mathcal{X}_f \tag{7.7}$$

for all $j \in \{0, \ldots, N-1\}$, where $\hat{x}_0 = x_k$ and $\gamma \leq m$. Note that constraint (7.6) encompasses the cardinality constraint along the prediction horizon, while (7.7) is a terminal constraint, with a terminal region given by

$$\mathcal{X}_f = \{x \in \mathbb{R}^n : |x| \leq b\}, \tag{7.8}$$

with $b \in \mathbb{R}_{\geq 0}$.

Consequently, the optimal input sequence, $\vec{u}^{op}(x)$, is the one which minimizes the cost function,

$$\vec{u}^{op}(x) = \arg\left\{\min_{\vec{u} \in \mathcal{U}(x)} V_N(x, \vec{u})\right\}. \tag{7.9}$$

where the set $\mathcal{U}(x) \in \mathbb{R}^{Nm}$ contains all possible input vector sequences, $\vec{u}$, which satisfy the constraints (7.6)-(7.7).

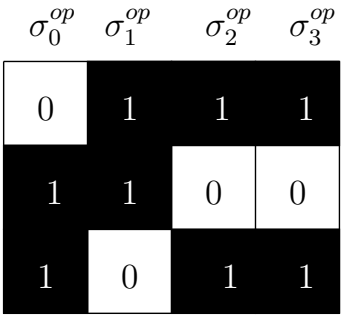


Figure 7.1: Vector of active inputs in the horizon.

The resulting optimal solution is the input control sequence

$$\vec{u}^{op}(x) = \left[(\hat{u}_0^{op})^\top, \ldots, (\hat{u}_{N-1}^{op})^\top\right]^\top, \tag{7.10}$$

while the resulting optimal state sequence is

$$\vec{x}^{op}(x) = \left[x^\top, (\hat{x}_1^{op})^\top, \ldots, (\hat{x}_N^{op})^\top\right]^\top.$$

Additionally, for this particular problem, we also obtain the resulting optimal active input sequence, given by

$$\vec{\sigma}^{op}(x) = \left[(\sigma_0^{op})^\top, \ldots, (\sigma_{N-1}^{op})^\top\right]^\top. \tag{7.11}$$

Notice that the elements of $\sigma^{op}(x)$ may differ from each other. However, $\operatorname{card}\sigma_j^{op} \leq \gamma$ for all $j \in \{0, \ldots, N-1\}$. For example, if $N = 4$, $m = 3$, and $\gamma = 2$, a possible $\vec{\sigma}^{op}(x)$ is shown in Figure 7.1.

We also denote the domain of the cost function V_N via

$$\mathcal{X}_N = \{x \in \mathbb{R}^n : \mathcal{U}(x) \neq 0\}. \tag{7.12}$$

Finally, we use a *receding horizon* technique i.e. only the first element of $\vec{u}^{op}(x)$ is applied to the system at each sampling instant. The solution of the optimal problem, $\mathcal{P}_{MPC}(x)$ in (7.4), yields the MPC control law, $\kappa_N(\cdot) : X_N \to \mathbb{R}^m$,

$$\kappa_N(x) = \hat{u}_0^{op}. \tag{7.13}$$

The resulting MPC loop can be represented via

$$x_{k+1} = Ax_k + B\kappa_N(x_k). \tag{7.14}$$

The proposed method described in chapter 6 is used to impose the cardinality constraint. Denote as $\hat{u}_{ji}$ to the tentative control action at the j-th instant in the horizon, for the i-th input, similarly we define the auxiliary variables w_{ji} with $j \in \{0,\dots,N-1\}$, and $i \in \{1,\dots,m\}$. Then, problem $\mathcal{P}_{MPC}$ can be reformulated as follows

$$
\begin{aligned}
\mathcal{P}_{MPCEQ}: \qquad V_N^{op}(x) = \underset{\vec{u}\in\mathbb{R}^{Nm}}{\text{minimize}} \;& \{V_N(x,\vec{u})\} \\
\text{subject to } & \hat{x}_{j+1} = A\hat{x}_j + B\hat{u}_j, \\
& \hat{x}_N \in \mathcal{X}_f \\
& \hat{u}_{ji} w_{ji} = 0, \\
& 0 \le w_{ji} \le 1, \\
& \sum_{\ell=1}^{m} w_{j\ell} = m - \gamma,
\end{aligned}
$$

for all $j \in \{0,\dots,N-1\}$ and for all $i \in \{1,\dots,m\}$. Notice that constraints $\sum_{\ell=1}^{m} w_{j\ell} = m-\gamma$ for all $j \in \{0,\dots,N-1\}$, correspond to cardinality constraints imposed on groups. This is another example of group constraints discussed in section 6.4.

7.2 Numerical Example

In this section we verify the performance of the proposed approach via a simulation study.

We consider a linear system (7.1) such that $x \in \mathbb{R}^4$, $u \in \mathbb{R}^3$, and

$$
A = \begin{bmatrix} 0.6122 & 0.2349 & -0.0021 & 0.1362 \\ -0.0366 & 0.7871 & 0.2047 & -0.1814 \\ -0.1941 & -0.1420 & 1.1499 & -0.2657 \\ -0.1864 & 0.0280 & 0.2942 & 1.2742 \end{bmatrix}, \tag{7.15}
$$

$$
B = \begin{bmatrix} -0.0151 & 0.2338 & 0.2710 \\ -0.3032 & -0.1504 & 0.0087 \\ 0.8390 & -0.0009 & -0.3200 \\ -0.0878 & -0.4431 & 0.0016 \end{bmatrix}. \tag{7.16}
$$

Notice that matrix A has 2 unstable eigenvalues.

To design the quadratic cost function, $V_N(x,\vec{u})$, we choose $Q = I_{4\times4}$, $R = diag\{1,1,5\}$, and a

prediction horizon of $N = 4$. Then, the matrix P is obtained by solving a Riccati equation, yielding

$$P = \begin{bmatrix} 1.6324 & 0.2733 & -0.4011 & -0.3720 \\ 0.2733 & 2.5520 & 0.7596 & -1.1004 \\ -0.4011 & 0.7596 & 3.2467 & 0.3548 \\ -0.3720 & -1.1004 & 0.3548 & 7.0326 \end{bmatrix}. \tag{7.17}$$

By prior experimentation we know that it is possible to stabilize the system using only $\gamma = 2$ inputs, provided we choose $\sigma = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}^\top$.

We are going to analyze the implementation of two MPC controllers. The first one computes the optimal control actions using a time-invariant cardinality constraint, $\gamma = 2$ (which ensures stability of the control loop). For the second controller, we let γ be time variant, i.e., $\gamma(k)$ can change with k. We then compute the optimal controller that guarantee stability with the minimum number of active inputs.

Figure 7.2 shows that the controller using $\gamma = 2$ stabilizes the system quicker than the one with $\gamma(k)$ at the expense of using more active inputs at each instant (see Figure 7.3 and Figure 7.4). This fact reveals the trade-off between control performances and number of active control inputs: the control performance gets worse when the number of active control inputs is lower and vice versa. Note that, at some instants, the system can be stabilized even with $\operatorname{card} u(k) = 0$.

Although the controller with $\gamma(k)$ uses less active control inputs, in general, the computational effort (and hence the calculation time) might be higher than for the case with a fixed γ because more combinations have to be explored. This makes the choice of the time-invariant γ a good trade-off between performance, usage of non-zero control inputs and on-line computational effort.

7.3 Chapter Summary

In this brief chapter, we have applied the approach developed in chapter 6 to model predictive control. The formulated model predictive control algorithm takes advantage of two of the main features of the method developed in chapter 6, namely, the ability to handle cardinality constraints, and the ability to handle group constraints.

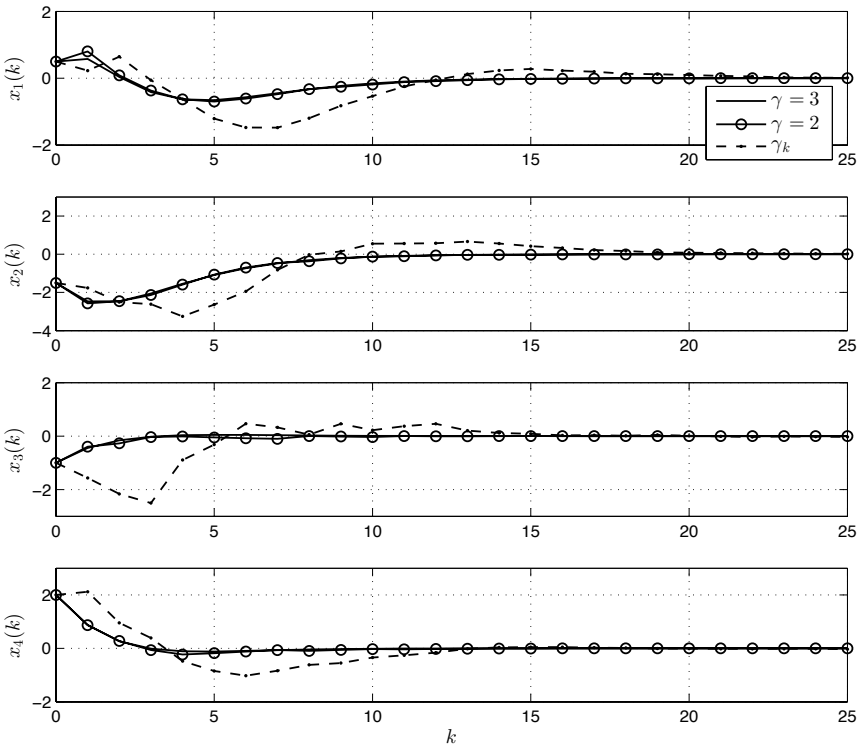


Figure 7.2: State stabilization $x(k)$ for $\gamma = 2$ and variable $\gamma(k)$.

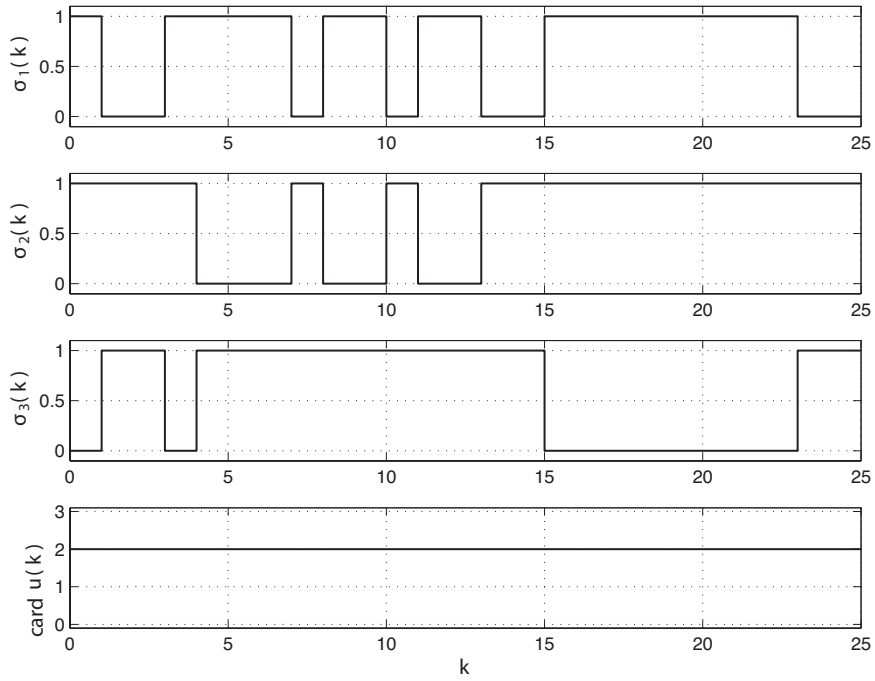


Figure 7.3: $\sigma(k)$ and $\mathrm{card}\{u(k)\}$ for $\gamma = 2$.

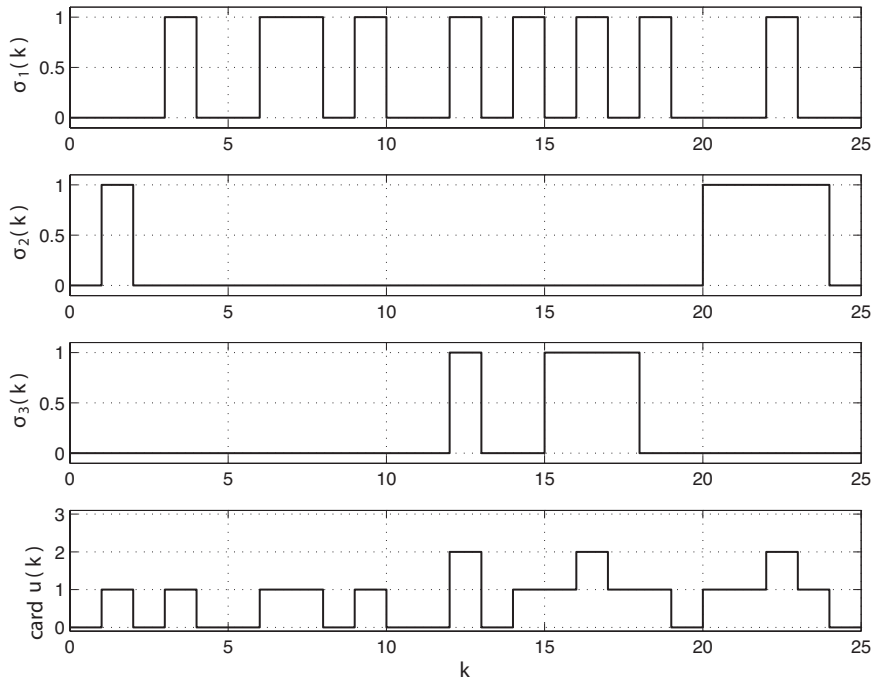


Figure 7.4: $\sigma(k)$ and $\mathrm{card}\{u(k)\}$ for minimum $\gamma(k)$.

Part II

Estimation-Error Quantification

8

Finite Data Estimation

8.1 Introduction

In this chapter we deal with the problem of estimation-error quantification for finite data. Finite data estimation arises in many practical problems. A well-known example is parameter estimation when only a small amount of data is available. Most of the methods used in practice for this problem provide error quantification. However, such error quantification typically depends upon asymptotic results. In this chapter, we propose an alternative approach to finite data estimation. The new approach provides a solution to the error quantification problem.

In some cases Prediction Error Methods (PEM) can be used to solve finite data parameter estimation [66]. These methods perform well but suffer from conceptual problems. For example, the usual quantification of the accuracy in PEM depends upon asymptotic results. This has motivated several authors to develop alternative schemes for parameter estimation with finite data [18, 113]. Of course, full Bayesian methods also provide a solution to the finite data problem but these suffer from other difficulties as we will discuss below.

The estimation procedure can be given two interpretations. If one adopts probabilistic models, then the estimation procedure can be interpreted as computing the Maximum A Posteriori (MAP) estimate. Alternatively, one can interpret the estimation procedure as simply a procedure for comparing a measured trajectory with a model trajectory via a suitable cost function. Whichever interpretation one uses, the estimation procedure requires that one use optimization for its solution. A fundamental issue of relevance to the current chapter is that only a point estimate is obtained. In the sequel we adopt the probabilistic interpretation.

By way of contrast, Bayesian estimation computes the complete a-posteriori distribution. However, Bayesian estimation also suffers from disadvantages. In particular, Bayesian estimation is generally

computationally expensive. Moreover, the size of the problem typically grows exponentially with the number of data points. Hence some form of simplification is usually required. In practice, this is achieved by using approximate schemes e.g. deterministic gridding algorithms, particle filtering or other resampling methods [25].

Here, we propose an approach to finite data estimation that combines MAP and Bayesian techniques. Importantly, it also provides a possible solution to the error quantification problem.

The layout of the remainder of the chapter is as follows: In section 8.2, we present the problem formulation. In section 8.3 we outline the combined MAP-Bayesian scheme for finite data problems. In section 8.4, we present a numerical example. Conclusions are presented in section 8.5.

8.2 Problem formulation

Consider a nonlinear system described by a state space model of the form

$$x_{t+1} = f(x_t) + w_t \tag{8.1}$$

$$y_t = h(x_t) + v_t \tag{8.2}$$

where $x_t \in \mathbb{R}^{n_x}$, $y_t \in \mathbb{R}^{n_y}$. For simplicity[1] we assume that

$$\begin{bmatrix} w_t \\ v_t \end{bmatrix} \sim N\left(\begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \right) \tag{8.3}$$

Our goal is to estimate the states $x_0, \ldots, x_N$, given observations $y_1, \ldots, y_N$. We also assume that a prior distribution is available for x_0.

Two general approaches for solving this problem are MAP and Bayesian estimation. These two approaches are based on the common element of the a-posteriori distribution. An expression for the a-posteriori distribution is given in Lemma 8.2.1 below:.

Lemma 8.2.1. *For the system (8.1)-(8.2), the a-posteriori distribution for the states $x_0, \ldots, x_N$, given the observations $y_1, \ldots, y_N$ is*

$$\begin{aligned} &p(x_0, x_1, \ldots, x_N | y_1, \ldots, y_N) \\ &\qquad \propto \prod_{i=1}^{N} p(x_i|x_{i-1}) p(y_i|x_i) p(x_0) \end{aligned} \tag{8.4}$$

where $\propto$ denotes "modulo a normalizing constant".

[1] The extension to more general models and noise distributions presents no additional conceptual difficulties.

Proof. From Bayes rule,

$$\begin{aligned} &p(x_0, x_1, \ldots, x_N | y_0, \ldots, y_{N-1}) \\ &\quad \propto p(y_1, \ldots, y_N | x_0, \ldots, x_N) p(x_0, \ldots, x_N) \end{aligned} \tag{8.5}$$

The results then follows by using the Markov property inherent in (8.1),(8.2). ■

MAP and Bayesian estimation can then be described as follows:

Maximum A Posteriori (MAP) estimation provides a point estimate corresponding to the maximum of the a-posteriori distribution, i.e.

$$\hat{x}_0, \ldots, \hat{x}_N = \arg \max_{x_0, \ldots, x_N} p(x_0, x_1, \ldots, x_N | y_1, \ldots, y_N) \tag{8.6}$$

Note that the associated algorithm only explores the a-posteriori distribution in so far as is necessary to reach the maximum.

On the other hand, Bayesian Estimation is aimed at computing (at least approximately) the complete a-posteriori distribution as in (8.4). From this distribution, one can extract any desired point estimate (e.g. mean, MAP, etc.). Information about the accuracy of any particular estimate is also automatically available.

Unfortunately, the computation of the complete a-posteriori distribution is, in general, intractable. However there are very special cases, such as unconstrained linear Gaussian problems, where the Kalman Filter provides an exact representation of the a-posteriori distribution. Hence, for most problems, approximate methods are typically used in practice. For example, the Extended Kalman Filter (EKF), see e.g. [58], linearizes the non linear system, then applies standard Kalman Filter to propagate the mean and covariance of the estimates. Alternatively, one could use a deterministic grid on the state space. A related approach is Minimum Distortion Filtering (MDF) [45], which uses a grid for the a-posteriori distribution that is focussed on the most likely parts of the state space. Another commonly used method is Particle Filtering (PF) [50]. This method draws a set of random samples from the disturbance distribution.

We see from the above discussion that MAP and Bayesian methods each have strengths and weaknesses. Here we propose a strategy which combines MAP and Bayesian methods so as to capitalize on the strength of each method. The core idea is explained in the next section.

8.3 Combined MAP and Bayesian estimation

We begin by describing the algorithm in the context of finite data estimation.

Initialization: We assume that we are given the prior distribution $p(x_0)$ and data $y_1, \ldots, y_N$. Also, we assume that $p(x_0)$ is well approximated by a point distribution of the form

$$p(x_0) = \sum_{s=1}^{N_x} p_s^0 \delta(x_0 - \bar{x}_0(s)) \tag{8.7}$$

where $p_1^0, \ldots, p_{N_x}^0$ denote point probability masses at $\bar{x}_0(1), \ldots, \bar{x}_0(N_x)$ respectively, and N_x is the number of points in the point distribution.

We also assume we are given a point distribution $M(x)$ with N_x points,

$$M(x) = \sum_{l=1}^{N_x} q_l \delta(x - u(l)) \tag{8.8}$$

that approximates a multivariate standard Gaussian distribution in $\mathbb{R}^{n_x}$, i.e zero mean and diagonal unitary variance I_{n_x}.

Our proposal has 4 components.

(i) MAP estimation: In combined MAP and Bayesian estimation, we first use MAP estimation to obtain a single trajectory estimate $\hat{x}_1, \ldots, \hat{x}_N$. Note that we do not produce a MAP estimate for x_0, instead we marginalize over the point distribution form of $p(x_0)$, i.e.

$$\begin{aligned} &\hat{x}_1, \ldots, \hat{x}_N \\ &= \arg \max_{x_1, \ldots, x_N} \sum_{s=1}^{N_x} p(\bar{x}_0(s), x_1, \ldots, x_N | y_1, \ldots, y_N). \end{aligned} \tag{8.9}$$

The estimated trajectory $\hat{x}_1, \ldots, \hat{x}_N$ is the "most likely" trajectory in the state space. However, there is no information about the accuracy of the estimate.

(ii) EKF for breadth estimation: The EKF provides a computationally efficient way to obtain a measure of the covariance of the estimates. The EKF uses the standard Kalman Filter covariance update given by

$$\Pi_{k|k-1} = A_{k-1} \Pi_{k-1|k-1} A_{k-1}^T + Q \tag{8.10}$$

$$L_k = \Pi_{k|k-1} C_k^T (C_k \Pi_{k|k-1} C_k^T + R)^{-1} \tag{8.11}$$

$$\Pi_{k|k} = \Pi_{k|k-1} - L_k C_k \Pi_{k|k-1} \tag{8.12}$$

where, in our approach, we linearize the nonlinear system about the MAP estimate $\hat{x}_1, \ldots, \hat{x}_N$, i.e. we use

$$A_k = \frac{\partial f(x)}{\partial x^T}\bigg|_{x=\hat{x}_k} \qquad C_k = \frac{\partial h(x)}{\partial x^T}\bigg|_{x=\hat{x}_k} \tag{8.13}$$

where $\Pi_{0|0}$ is computed as

$$\Pi_{0|0} = \sum_{s=1}^{N_x} p_s^0 (\bar{x}_0(s) - \hat{x}_0)(\bar{x}_0(s) - \hat{x}_0)^T \tag{8.14}$$

$$\hat{x}_0 = \sum_{s=1}^{N_x} p_s^0 \bar{x}_0(s) \tag{8.15}$$

(iii) Gridding: (Here we shift and scale the point distribution (8.8)). Given the MAP estimates $\hat{x}_1, \ldots, \hat{x}_N$ in (8.9) and the covariances $\Pi_{1|1}, \ldots, \Pi_{N|N}$ computed using (8.12), we have a Gaussian approximation for the distribution of x_k, $k = 1, \ldots, N$. Next, for each $k = 1, \ldots, N$ we allocate a set of N_x points by using the following affine transformation of $M(x)$, i.e. for $l = 1, \ldots, N_x$, we define

$$\bar{x}_k(l) = \hat{x}_k + \Pi_{k|k}^{1/2} u(l). \tag{8.16}$$

where $M(x)$ and the corresponding points $u(l)$ are defined by (8.8), and $\hat{x}_k$ as in (8.9).

Remark 8.3.1. *Note, that the probability masses q_l, $l = 1, \ldots, N_x$ of $M(x)$ are not used in the method, since the probability masses are computed in step **(iv)**.* ▽▽▽

(iv) Approximate Bayesian update: The previous step provides a grid of points $\bar{x}_k(l)$, $k = 1, \ldots, N$, $l = 1, \ldots, N_x$ that is allocated in a part of the state space where the a-posteriori distribution is concentrated. In this step, the corresponding probability masses p_l^k are computed. To compute p_l^k we carry out a full recursive Bayesian computation over the grid. For $l = 1, \ldots, N_x$ beginning with $k = 1$ compute

$$\begin{aligned} p_l^k &= p(\bar{x}_k(l)|y_1, \ldots, y_k) \\ &= c \sum_{j=1}^{N_x} p_j^{k-1} p(\bar{x}_k(l)|\bar{x}_{k-1}(j)) \cdot p(y_k|\bar{x}_k(l)) \end{aligned} \tag{8.17}$$

where c is a normalizing constant, when k is updated then p_l^k is computed for $l = 1, \ldots, N_x$ using (8.17). The procedure is repeated until p_N^l is computed for $l = 1, \ldots, N_x$.

The proposed method is summarized in Algorithm 8.1 (see Table).

Algorithm 8.1 Combined MAP and Bayesian estimation algorithm

- Initialize:
  - ⋆ Provide a point distribution for the initial state as in (8.7).
  - ⋆ Provide a point distribution $M(x)$ that approximates a multivariate standard Gaussian distribution, as in (8.8).
- Obtain a trajectory $\hat{x}_1, \ldots, \hat{x}_N$ by solving (8.9).
- Compute the covariance of x_0, $\Pi_{0|0}$, using (8.14), and then compute $\Pi_{1|1}, \ldots, \Pi_{N|N}$ using (8.10)-(8.13).
- Allocate a grid of points, $\bar{x}_k(l)$, $k = 1, \ldots, N$; $l = 1, \ldots, N_x$ on the state space by using (8.16).
- Perform full Bayesian update over the grid of points, i.e. for $k = 1, \ldots, N$
  - ⋆ for $l = 1 : N_x$, compute the point masses p_l^k by using (8.17)
  - ⋆ $k = k + 1$
- End Result: The pair $(\bar{x}_k(l), p_l^k)$ for $k = 1, \ldots, N$; $l = 1, \ldots, N_x$ that approximates the a-posteriori distribution (8.4).

Remark 8.3.2. *If the prior distribution is continuous, then one can obtain* $p(x_0)$ *as in* (8.7) *by random sampling or by use of vector quantization methods (see e.g. [43]). This step is computationally expensive but will be performed* off-line *before running the algorithm. The same techniques can also be used to obtain the point distribution* $M(x)$. ▽▽▽

Remark 8.3.3. *The idea behind the proposed approach is to provide additional information about the estimate obtained by MAP estimation. However the proposed approach can also be used to improve the accuracy information provided by other methods, such as the Extended Kalman Filter, or the Unscented Kalman Filter, among others.* ▽▽▽

Remark 8.3.4. *The current exposition of the combined MAP-Bayesian estimation, suggests that the single trajectory* $\hat{x}_1, \ldots, \hat{x}_N$ *is obtained by using MAP estimation. Nevertheless, the proposed approach can also be applied when the single trajectory is obtained using other optimization based methods.* ▽▽▽

8.4 Numerical example

In this section we present a numerical example. In the example, we consider a two state batch reactor.

Consider a well-mixed semibatch chemical reactor where the material balances for the two components are

$$\dot{c}_A = -2\,\kappa\, c_A^2 + \frac{Q_f}{V} c_{Af} \tag{8.18}$$

$$\dot{c}_B = \kappa\, c_A^2 + \frac{Q_f}{V} c_{Bf} \tag{8.19}$$

with parameter values

$$\frac{Q_f}{V} = 0.4$$

$$\kappa = 0.16$$

$$c_{Af} = 1$$

$$c_{Bf} = 0$$

The initial conditions are $c_A(0) = 3$ and $c_B(0) = 1$. The measurements correspond to the total pressure, which is the sum of the two states. The sample period is $\Delta = 0.1$. State estimation is carried out assuming that x_0 is normal distributed with mean $[3\ 1]^T$ and variance I_2. We consider that, in (8.18)-(8.19), a disturbance is introduced, which is zero-mean Gaussian distributed with variance $Q \cdot \Delta$, where $Q = 0.01 I_2$.

We are given $N = 100$ measurements, that are contaminated with additive zero-mean Gaussian white noise, with variance $\Delta \cdot R$, with $R = 0.01$.

We compare the proposed approach (MAP-Bayes) with $N_x = 13$ against a Particle Filter (PF) [50] with $N_{pf} = 100$ particles, the Extended Kalman Filter (EKF) (see e.g. [58]) and an approximation to Full Bayesian Filtering via a fine deterministic grid (FB). Also we consider a variant of our algorithm, denoted as EKF-Bayes, where instead of using MAP to obtain the single trajectory, we use EKF.

Figures 8.2a-8.2b show the estimated mean value obtained by each method. All the estimated mean values are close to the mean obtained by FB, save for the mean value provided by EKF which is far from the FB's estimate. Note that the proposed EKF-Bayes approach provides satisfactory results, even when EKF fails. Thus the combined strategy remedies the failure of EKF even though EKF is used as part of the combined algorithm. The failure of EKF in the batch reactor example has been previously reported in [54].

Figure 8.1 shows the tradeoff in the MAP-Bayes approach between performance and computational load. This tradeoff is managed by choosing the number of points to be allocated, N_x. The performance is measured by using the Hellinger divergence (see e.g. [63]), where we use the distribution provided by FB as a reference. The computational load is measured by the average execution time per sample.

8.5 Chapter Summary

This chapter has described a finite data estimation scheme which allows one to combine MAP and Bayesian strategies. The scheme resolves a difficulty that is inherent in the usual MAP based estimation schemes, namely how to quantify the accuracy of the resulting estimates. The performance of the scheme has been compared with other commonly used schemes via a simulation example. The example shows that the new scheme gives good performance at reasonable computational cost. In practice, the trade-off in computational effort depends on the relative effort directed at finding the maximum of a function and exploring the full a posteriori distribution. The appropriate balance is problem dependent. We thus encourage others to use the ideas presented here. We provide access to preliminary software, see http://db.tt/b459SW2e.

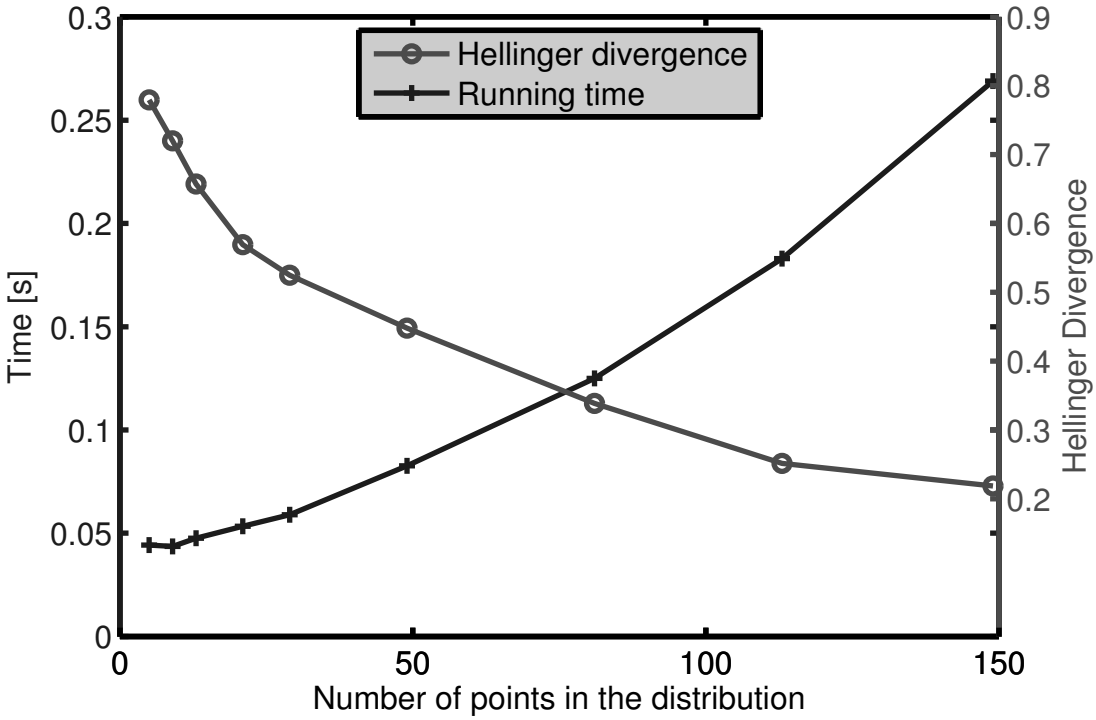
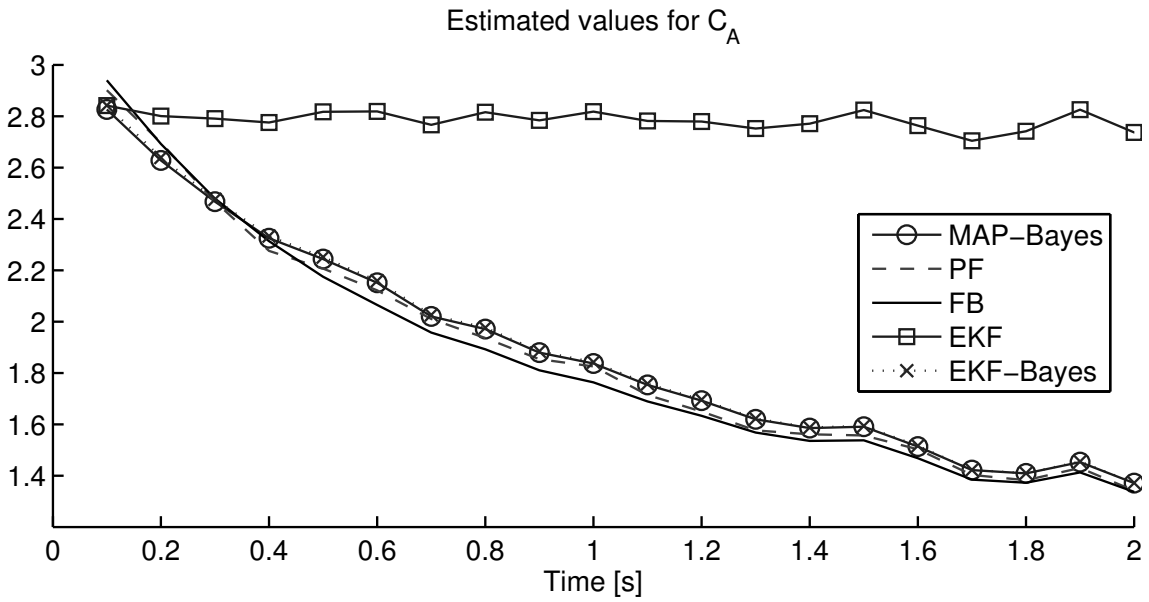
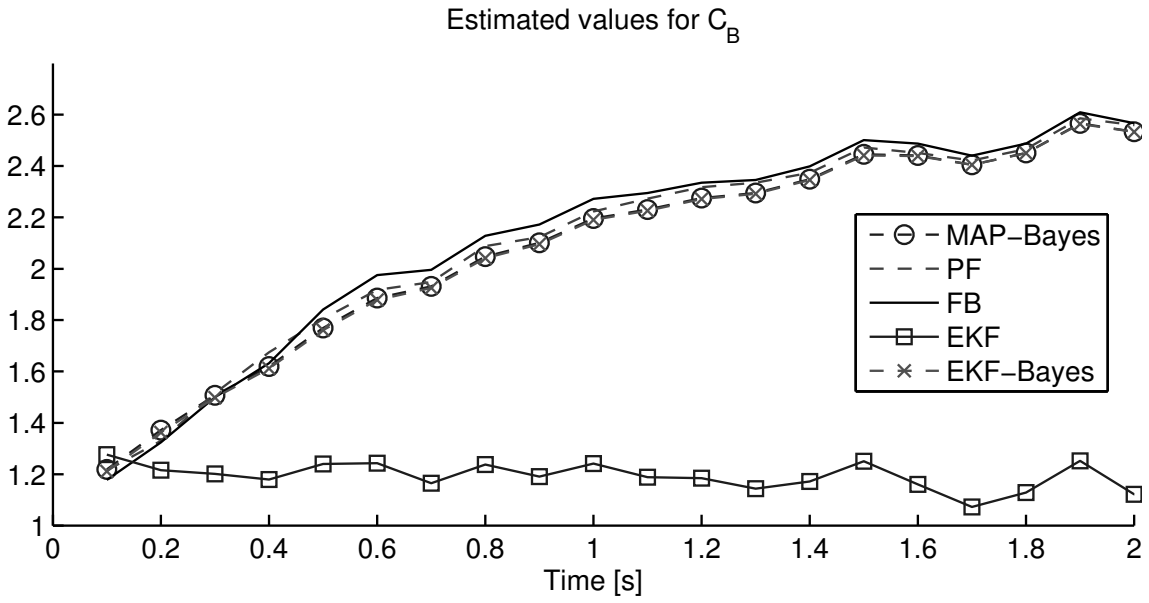


Figure 8.1: Trade off between computational load and performance in MAP-Bayes method.



(a) C_A (Concentration of A)



(b) C_B (Concentration of B)

Figure 8.2: Mean value of the distributions provided by each method.

9

Moving Horizon Estimation

9.1 Introduction

In this chapter we apply the methods described in chapter 8 to the problem of moving horizon estimation. Moving Horizon Estimation (MHE) is closely related to finite data estimation, since it combines a sequence of finite data problems. MHE transforms filtering, smoothing and prediction problems into a standard constrained optimisation problem over a finite horizon. In order to limit the size of the problem, MHE requires that the range of data used for estimation be small. This means that, when new data arrives, the oldest data is required to be summarized by a, so called, "arrival cost". However, the formulation of a statistically well posed arrival cost remains an open problem. In this chapter, we will use the approach described in Chapter 8 for error quantification for finite data problems, to provide an arrival cost for the MHE problem.

MHE has received increasing attention over the last decade [6, 11, 87–90, 111]. MHE has several advantages compared with other schemes. These advantages arise due to the transformation of the problem into a standard optimisation problem. One such advantage, is that it allows one to incorporate constraints, for example, on the states of the system (e.g a tank cannot be more than full or less than empty). Also, standard tools developed for Model Predictive Control can be applied to MHE (see e.g. [33, 91]).

On the other hand, there are difficulties associated with the usual MHE scheme. For example, the impact of past data needs to be summarized in the form of an a-priori distribution. This is typically achieved by adding an arrival cost [11, 89, 111]. However, the formulation of a statistically well posed arrival cost remains an open problem [54]. To address this problem various approximate arrival cost strategies have been proposed, see e.g [6, 87, 88, 108, 117]. One such strategy expresses the arrival cost as a simple quadratic function of the difference between the current initial state,

and the propagation of the initial state estimate from the previous horizon (see e.g. [6, 7]).

Here, we use the method developed in Chapter 8 to provide a possible solution to the entry cost problem.

The layout of the remainder of the chapter is as follows: In section 9.2 we describe our approach to Moving Horizon Estimation. In section 9.3, we present a numerical example. Conclusions are presented in section 9.4.

9.2 Moving Horizon Estimation

In MHE, the estimation is carried out, online, by successively solving a sequence of finite data problems. A new issue that arises in the MHE framework is how to incorporate the effect of the past data (outside the current estimation interval). In the combined MAP-Bayesian framework described in chapter 8, the a-posteriori distribution for the last state can be stored and reused at a later time as the prior distribution.

There are several versions of the moving horizon algorithm depending upon whether estimates are required only after each block of N samples or whether estimates are needed after each data point is received. We refer to these cases as Block estimation and Sample-by-Sample estimation, respectively.

(i) Block estimation. Here one simply uses the a-posteriori distribution for the final state obtained in the previous block as prior distribution for the next block (see e.g [91, page 353])

(ii) Sample-by-Sample estimation with filtered update. Here one needs to place the a-posteriori distribution for the final state obtained from the current block in a "rotating store". This scheme can be seen as N Block estimators running in parallel.

(ii) Sample-by-Sample estimation with smoothed update: in this update scheme the second value of the previous solution is used as the initial condition.

An analysis of the different Sample-by-Sample update schemes can be found in [40]. Here, we focus on the Block estimation scheme. However, there is no conceptual difficulties to use the proposed approach with any of the MHE schemes described above.

Regarding the stability and convergence of the approach, the proposed MHE scheme described here, can be seen as a special case of the algorithm described in [89]. Hence, the stability and

convergence results presented in [89] can also be adapted to the method proposed here.

9.3 Numerical example

In this section we present a numerical example. We consider a single state water tank, with two different radii as a function of the height.

Thus, consider a water tank having two areas: at some height, h_c, the transversal area increases by a factor of 10. We consider that the water level is measured with noise, and we are interested in estimating the volume of water in the tank.

The volume of water in the tank is governed by

$$\frac{dV}{dt} = V_{in} - V_{out} \tag{9.1}$$

where

$$V_{out} = \begin{cases} \pi r_0^2 \sqrt{2g\frac{V}{\pi r_0^2}} & \text{if } V/(\pi r_0^2) \leq h_c \\ \\ \pi r_0^2 \sqrt{2g(\frac{V}{\pi r_1^2} + h_c(1 - \frac{r_0^2}{r_1^2}))} & \text{if } V/(\pi r_0^2) > h_c \end{cases} \tag{9.2}$$

The input water volume is given by $V_{in} = q_{in} + w$, with q_{in} constant and w an i.i.d. Gaussian distributed process, with zero mean and variance $\Delta \cdot Q$. The sampling period is $\Delta = 0.1[s]$. We measure the water level, that is related to the volume by

$$h = \begin{cases} \frac{V}{\pi r_0^2} & \text{if } h \leq h_c \\ \\ \frac{V}{\pi r_1^2} + h_c(1 - \frac{r_0^2}{r_1^2}) & \text{if } h > h_c \end{cases} \tag{9.3}$$

and the measurements are contaminated with an additive zero-mean Gaussian white noise, with variance $\Delta \cdot R$. The parameter values are chosen to be

$$r_0 = 1/\pi; \qquad r_1 = \sqrt{10}/\pi \qquad h_c = 1/(\pi r_0^2);$$
$$g = 9.80665; \qquad q_{in} = 0.447$$

We are given $N = 100$ measurements. We assume that the initial condition for the water volume is distributed uniformly on the interval $[0, 2]$. We take $R = 0.01$ and $Q = 0.1$.

We compare the proposed approach (MAP-Bayes) against the Particle Filter (PF) [50], the Extended Kalman Filter (EKF) (see e.g. [58]) and an approximation to Full Bayesian Filtering via a

fine deterministic grid (FB). Also we consider a variant of our algorithm, where instead of using MAP to obtain the single trajectory, we use EKF.

For the MAP-Bayes method we use rolling horizon length $N_h = 1$, and use $N_x = 19$ points to approximate the standard Normal distribution. In PF, we use $N_{pf} = 100$ particles. For the EKF we approximate the initial distribution as Normal distributed with mean 1 and variance 1/3.

Table 9.1 shows the average running time per sample. We see that the proposed MAP-Bayes and EKF-Bayes have the same order of magnitud running time. For this example EKF requires the least computational effort.

Table 9.2 shows the Mean Square Error (MSE) (between the "true value" as computed by FB and the given method) of the mean, mode, variance and skewness estimated for each method. For the estimated mean, variance and skewness PF provides the best estimates, but produce a poor estimate for the mode value. Next, EKF-Bayes produces mean, variance and skewness estimates with the same order of magnitude error as PF. Nevertheless, the estimated mode value is much better than that provided by PF. The MAP-Bayes method produces slightly worse estimates than EKF-Bayes, however this could be problem dependant, since in MAP-Bayes the points are allocated around the MAP estimate, and in the EKF-Bayes method the points are allocated around the mean value. EKF has the poorest performance over all the methods.

	MAP-Bayes	PF	FB	EKF	EKF-Bayes
Time	0.046	0.003	3.660	0.002	0.016

Table 9.1: Average running time per sample for each method.

Method	mean	mode	variance	skewness
MAP-Bayes	0.0245	0.0092	0.0029	0.5433
PF	0.0014	0.1341	0.0003	0.1545
EKF	0.1701	0.0856	0.0127	0.6341
EKF-Bayes	0.0030	0.0039	0.0004	0.3013

Table 9.2: Mean Square Error of the mean, variance, mode and skewness of each method, compared to that obtained by Full Bayesian filtering.

9.4 Chapter Summary

This chapter has extended the methods described in chapter 8 to Moving Horizon state estimation. The method provides a possible solution to two difficulties that are inherent in the usual (MAP based) Moving Horizon state estimation schemes, namely (i) how to obtain a meaningful "arrival cost" and (ii) how to quantify the accuracy of the resulting estimates. The example shows that the new scheme gives good performance at reasonable computational cost. We provide access to preliminary software, see http://db.tt/b459SW2e.

Part III

Modelling Model Uncertainty

10

Stochastic Embedding Revisited

10.1 Introduction

Chapter 8 addressed the problem of estimation-error quantification when the system belonged to the model set. We focused on the case when only limited data was available and hence asymptotic error quantification was inappropriate. A novel combined MAP-Bayesian strategy was developed. In this chapter, we go a step further and address the case where the model structure is artificially restricted so as to achieve a better bias-variance trade-off. This raises a key issue of how to describe the inherent errors between the system and the restricted model.

The idea of uncertainty modelling has been a central theme in statistics, time series analysis, econometrics and system identification (see e.g. [23, 79, 80] and the reference therein). In practice, physical systems are more complex than models that describe the system's behaviour (see e.g. [23, 46]). This fact is well recognized in the areas of system and control, where robustness to model inadequacy has been a key focus.

In this chapter, we model the uncertainty as a realization of a stochastic process. We then solve the estimation problem using the EM algorithm. The novel feature of our algorithm is that allows us to estimate simultaneously the nominal model and associated uncertainty. The estimation is carried out in the maximum likelihood framework.

The layout of the remainder of the chapter is as follows: In section 10.2 we review some existing approaches to uncertainty modelling. In section 10.3 we present a description of the model of interest and describe the proposed estimation algorithm. In section 10.4 we present numerical examples. Finally, we draw conclusions in section 10.5.

10.2 Traditional Approaches to uncertainty modelling

10.2.1 General Overview

The topic of uncertainty modelling has attracted significant attention in the past 20 years, where methods such as Set Membership, Stochastic Embedding and Model Error Modelling has been developed.

The Set membership (SM) approach (see e.g. [77, 78]), deals with the problem of "bounded but unknown errors" in a deterministic framework. SM delivers a set of possible solutions.

On the other hand, a probabilistic framework was considered in [18, 19, 110, 112], where a set of possible solutions can be found based on finite sample properties. Moreover, confidence sets were found using Sub-sampling, Resampling and/or Bootstrapping techniques (see e.g. [15, 35, 42, 106]).

In [24] it was shown that the prior information of the smoothness in the impulse response can be used to estimate bounds on the set of possible solutions.

The methods mentioned above make few assumptions about the model error. On the other hand, some approaches assign a model to the model error. For example, in the *Model Error Modelling* (MEM) method [65], the model error description is obtained from the residuals, $\varepsilon = y - \hat{y}$. In MEM, the model error description is considered as the sum of two terms. The first term is a function of the input signal, and the second term is a function of a random process.

Another method that assigns a model to the model error is the *Stochastic Embedding* (SE) approach (see e.g. [47–49]). In this approach the nominal model structure is embedded into a larger class of models. In this sense, a characterization of the undermodelling is included as a realization of a stochastic process. Moreover, in the existing literature [47–49], the estimation is carried out by an ad-hoc two step procedure: (i) obtain a non parametric estimation of the frequency response, then (ii) obtain a parametric model. In SE the nominal model is obtained by using Least Squares. This limits the complexity of the models that can be handled. For a comparison between SE, SM and MEM see [93].

The key difficulty in this area is finding a satisfactory, broad, and systematic method to describe the model error. In this chapter we propose a systematic methodology to describe a broad class of model uncertainties. The idea is based on the SE approach, which treats the model uncertainty as a realization of a stochastic process. The main novelty of the work presented here is that we estimate the nominal and uncertainty models simultaneously via a non ad-hoc Maximum Likelihood strategy

using the Expectation-Maximization algorithm.

In the following two subsections we give further details of two existing methods for uncertainty modelling.

10.2.2 Modelling the Residuals

In this approach [65], known in the literature as Model Error Modelling (MEM), the dynamic system $G(\beta, q)$ is first assumed to belong to a model class parametrized by $\beta \in \Omega$, where Ω is a constraint set in the parameter space, and q is the forward shift operator. The system output is represented by

$$y_t = G(\beta, q)u_t + v_t, \tag{10.1}$$

where u_t is the input signal, v_t is additive noise and y_t the measured output, $t = 0, 1, \ldots, N-1$. By minimizing a cost function, such as that utilized in the Prediction Error Method (PEM) [66], it is possible to obtain an estimate of the system parameters, denoted by $\hat{\beta}$. This estimate is then utilized to obtain the model errors (residues) of the estimates (see e.g. [65])

$$\varepsilon_t = y_t - G(\hat{\beta}, q)u_t. \tag{10.2}$$

Next, a model structure is assigned to the residues, such as the following

$$\varepsilon_t = F(\gamma, q)u_t + H(\gamma, q)w_t, \tag{10.3}$$

where $F(\gamma, q)$, and $H(\gamma, q)$ are rational transfer function in the operator q, and parametrized by $\gamma \in \Gamma$. The parameters of the error model in (10.3) are then estimated using an estimation procedure such as PEM.

This procedure provides a model error description that captures the reliability of the nominal model. This model error description can be used, for example, to generate a confidence set that can be used for model validation or for other specific applications, such as robust control. Note, however, that the procedure is ad-hoc since one first fits the nominal model and then one fits the model to the residuals.

10.2.3 Stochastic Embedding

The SE [47, 49] approach treats both noise and undermodelling as stochastic process.

In [49] the SE approach considers additive undermodelling. However, in [47] the undermodelling

is described as a multiplicative error in the form $G = G_o(1 + G_\Delta)$ where G_o is the nominal model and G_Δ is the model uncertainty.

We focus on the case when the undermodelling is considered as multiplicative. The estimation procedure described in [47] for this case is given as follows. Consider the input signal as a finite sum of cosine functions [47]

$$u_t = \sum_{r=1}^{m} a_k cos(\omega_r t), \tag{10.4}$$

where m is the number of elements in the summation, ω_r is the r-th normalized discrete frequency given by $\omega_r = \frac{2\pi k_r}{N}$, $k_r \in \{1, \ldots, N\}$, and $m \leq N$. The periodic input, u_t is used to obtain a point (non-parametric) estimate of the transfer function of the system, $g(j\omega_r)$, $\forall r$, and its associated measurement errors at each ω_r of the (not necessarily equally spaced) frequencies set $\{\omega_1, \omega_2, \ldots, \omega_m\}$. Then, these frequency domain estimates are used to obtain a parametric model of $g(j\omega_r)$ by estimating the parameters that define it, and the parameters of the embedded stochastic process that represent the undermodelling.

First, assuming that the measurements are obtained under steady state conditions, the sampled output response is expressed as

$$y_t = \sum_{r=1}^{m} a_r g^{\mathbf{r}}(\omega_r) \cos(\omega_r t) - \sum_{r=1}^{m} a_r g^{\mathbf{i}}(\omega_r) \sin(\omega_r t) + v_t, \tag{10.5}$$

where $g^{\mathbf{r}}(\omega_r)$ and $g^{\mathbf{i}}(\omega_r)$ are the real and imaginary parts of $g(j\omega_r)$, i.e. $g(j\omega_r) = g^{\mathbf{r}}(\omega_r) + jg^{\mathbf{i}}(\omega_r)$. Also, v_t is additive white noise with variance σ_v^2. Next, we rewrite (10.5) in vector form as

$$Y = \Phi G + V, \tag{10.6}$$

where

$$Y = \begin{bmatrix} y_1 & y_2 & \ldots & y_N \end{bmatrix}^T, \tag{10.7}$$

$$G = \begin{bmatrix} g^{\mathbf{r}}(\omega_1) & g^{\mathbf{i}}(\omega_1) & \ldots & g^{\mathbf{r}}(\omega_m) & g^{\mathbf{i}}(\omega_m) \end{bmatrix}^T, \tag{10.8}$$

$$\Phi = \begin{bmatrix} \cos(\omega_1) & \sin(\omega_1) & \ldots & \sin(\omega_m) \\ \cos(\omega_1 2) & \sin(\omega_1 2) & \ldots & \sin(\omega_m 2) \\ \vdots & \vdots & \ddots & \vdots \\ \cos(\omega_1 N) & \sin(\omega_1 N) & \ldots & \sin(\omega_m N) \end{bmatrix}, \tag{10.9}$$

$$\times \operatorname{diag}[a_1, -a1, a2, -a2, \ldots, a_m, -a_m], \tag{10.10}$$

$$V = \begin{bmatrix} v_1 & v_2 & \ldots & v_N \end{bmatrix}^T. \tag{10.11}$$

Then, an unbiased estimate of G is given by

$$\widehat{G} = \begin{bmatrix} \hat{g}^{\mathbf{r}}(\omega_1) \\ \hat{g}^{\mathbf{i}}(\omega_1) \\ \vdots \\ \hat{g}^{\mathbf{r}}(\omega_m) \\ \hat{g}^{\mathbf{i}}(\omega_m) \end{bmatrix} = (\Phi^T\Phi)^{-1}\Phi^T Y. \tag{10.12}$$

The estimate $\hat{g}(j\omega)$ is represented in terms of a given set of basis functions $b_1(j\omega), \ldots, b_p(j\omega)$. We define the vectors

$$\mathbf{B}^{\mathbf{r}}(\omega) = [b_1^{\mathbf{r}}(\omega), \ldots, b_p^{\mathbf{r}}(\omega)]$$
$$\mathbf{B}^{\mathbf{i}}(\omega) = [b_1^{\mathbf{i}}(\omega), \ldots, b_p^{\mathbf{i}}(\omega)]$$

containing the real and imaginary parts of the basis function. We denote by θ the vector containing the coefficient of the basis functions that describe the nominal model, and by $\bar{\theta}$ the vector of coefficients of the basis functions that describe the undermodelling.

Thus, we can represent the system as

$$\widehat{G} = \mathcal{B}\theta + \Lambda\mathcal{B}\bar{\theta} + \widetilde{G}, \tag{10.13}$$

where

$$\bar{\theta} = \theta \tag{10.14}$$
$$\mathcal{B} = \begin{bmatrix} \mathbf{B}^{\mathbf{r}}(\omega_1) & \mathbf{B}^{\mathbf{i}}(\omega_1) & \ldots & \mathbf{B}^{\mathbf{r}}(\omega_m) & \mathbf{B}^{\mathbf{i}}(\omega_m) \end{bmatrix}^T \tag{10.15}$$
$$\widetilde{G} = \begin{bmatrix} \tilde{g}^{\mathbf{r}}(\omega_1) & \tilde{g}^{\mathbf{i}}(\omega_1) & \ldots & \tilde{g}^{\mathbf{r}}(\omega_m) & \tilde{g}^{\mathbf{i}}(\omega_m) \end{bmatrix}^T, \tag{10.16}$$
$$\Lambda = \mathrm{diag}[\lambda^{\mathbf{r}}(\omega_1), \lambda^{\mathbf{i}}(\omega_1), \ldots, \lambda^{\mathbf{r}}(\omega_m), \lambda^{\mathbf{i}}(\omega_m)]. \tag{10.17}$$

The noise signals $\{\tilde{g}^{\mathbf{r}}\}$ and $\{\tilde{g}^{\mathbf{i}}\}$ in (10.16), are assumed to be uncorrelated white noises having variance $2\sigma_v^2/(a_r^2 N)$. The random process $\{\lambda^{\mathbf{r}}\}$ and $\{\lambda^{\mathbf{i}}\}$ in (10.17) are two independent random walk process. A different model can be considered assuming that $\{\lambda^{\mathbf{r}}\}$ and $\{\lambda^{\mathbf{i}}\}$ are two independent integrated random walks. For more details see [47].

Remark 10.2.1. *In order to incorporate the undermodelling as a multiplicative error, $\theta = \bar{\theta}$ must be satisfied in (10.13). However, if this equality is satisfied, then the estimation problem is difficult, because the matrix of regressors $\Lambda\mathcal{B}$ is a function of the undermodelling. For this reason, the algorithm described in [47] was carried out in two steps. First, an estimate $\hat{\theta}$ of θ was obtained. Next, the estimate of the undermodelling variance was obtained assuming that $\bar{\theta} = \hat{\theta}$. This step is clearly ad-hoc although the results in [47] suggest that useful results are obtained using this idea.* ▽▽▽

10.3 Simultaneous Estimation of the Nominal and the Uncertainty Models

The methods described in sections 10.2.2 and 10.2.3, both contain ad-hoc elements. In this section we describe a procedure aimed at addressing the ad-hoc features of the two algorithms described above.

We take the Stochastic Embedding approach [49] to describe uncertainty. Next, we develop an estimation procedure that is based on the Expectation-Maximization algorithm. The two main features of our proposed algorithm are: (i) it estimates both the nominal model and the uncertainty simultaneously; and (ii) the estimation is carried out in the maximum likelihood framework.

10.3.1 Model Description

In this section we present the model description within the framework of Stochastic Embedding. Consider the following model for the system

$$G(j\omega_k) = G_o(j\omega_k)(1 + G_\Delta(j\omega_k)), \tag{10.18}$$

and the corresponding data generating system

$$Y(j\omega_k) = G(j\omega_k)U(j\omega_k) + V(j\omega_k), \tag{10.19}$$

where $\omega_k = \frac{2\pi k}{N}$, $Y(j\omega_k)$ and $U(j\omega_k)$ are the Discrete Fourier Transform of the measurement and the input signal, respectively. We assume that $V(j\omega_k)$ and $G_\Delta(j\omega_k)$ are realizations of random process, with probability distributions $p(G_\Delta(j\omega_k))$ and $p(V(j\omega_k))$, respectively.

This representation of $G_\Delta(j\omega_k)$ as a realization of a random process allows the representation of a class of possible models characterized by $G_o(j\omega_k)$ and $p(G_\Delta(j\omega_k))$.

Remark 10.3.1. *The assumption that $G_\Delta(j\omega_k)$ is a realization of a stochastic process suggests that, to obtain representative estimates of the uncertainty, the data should reflect multiple realizations of uncertainty, i.e. the data should include different scenarios associated with different uncertainty realizations.* ▽▽▽

In order to simplify the notation, we denote $G^{(p)}_{\Delta,k} \triangleq G_\Delta(j\omega_k)^{(p)}$, $G^{(p)}_\Delta = \{G_{\Delta,k}\}_{k=0}^{L}$, $Y^{(p)}_k \triangleq Y(j\omega_k)^{(p)}$ and $U^{(p)}_k \triangleq U(j\omega_k)^{(p)}$.

10.3.2 EM-based estimation

In this section, we provide a description of the estimation algorithm that is based on the Expectation-Maximization (EM) algorithm [32].

The EM algorithm has been used to identify different classes of dynamic systems, such as, continuous time systems using sampled-data [116], finite impulse response systems using quantized data [68], state-space systems using incomplete data [1], channel estimation in telecommunications [21], bilinear state-space systems [44], and non-linear state-space systems [96].

The EM algorithm is an iterative two step procedure [32] where the concept of complete data is introduced. The complete data is composed of the measured data, $\mathcal{Y}$, and also an unmeasured data set known as the *hidden data*, $\mathcal{X}$. Then, (loosely speaking) one estimates the hidden data based on the current parameter estimate (in the Expectation step (E-step)) and then updates the parameters by maximizing a function that depends on the joint probability density function (pdf) of the hidden data and the measurements evaluated at the estimated hidden data (in the Maximization step (M-step)). A more detailed description of the EM algorithm is given in Appendix A.

Given a current estimate $\hat{\theta}_i \in \Omega$, where Ω is the constraint set in the parameter space, an iteration of the EM algorithm is defined by:

E-step

$$\mathcal{Q}(\theta, \hat{\theta}_i) = \mathbf{E}\{\log p(\mathcal{Y}, \mathcal{X}|\theta)|\mathcal{Y}, \hat{\theta}_i\}. \tag{10.20}$$

M-step

$$\hat{\theta}_{i+1} = \arg\max_{\theta \in \Omega} \mathcal{Q}(\theta, \hat{\theta}_i). \tag{10.21}$$

where $p(\mathcal{Y}, \mathcal{X}|\theta)$ is the joint probability density function (pdf) of $\mathcal{Y}$ and $\mathcal{X}$ given θ.

Remark 10.3.2. *Note that recent work related to the ideas presented below has been carried out by by Ljung, Goodwin and Agüero (submitted for publication).* ▽▽▽

For the development of the algorithm we consider $G_\Delta(j\omega_k)$ as the hidden variable.

Lemma 10.3.1. *Consider the system given by (10.18)-(10.19), and that $G_\Delta \sim p(G_\Delta)$ is the hidden variable in the EM algorithm. Also consider that the data is collected from N_{exp} independent experiments, i.e. $\{G_\Delta^{(p)}, \mathcal{Y}^{(p)}\}_{p=1}^{N_{exp}}$. Then the auxiliary function $\mathcal{Q}(\theta, \hat{\theta}_i)$ in the EM algorithm is*

given by

$$\mathcal{Q}(\theta,\hat{\theta}_i)=\sum_{p=1}^{N_{exp}}\mathbf{E}\left\{\log p(\mathcal{Y}^{(p)}|G_{\Delta}^{(p)},\theta)|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}+\sum_{p=1}^{N_{exp}}\mathbf{E}\left\{\log p(G_{\Delta}^{(p)}|\theta)|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}$$

Proof. Following the ideas presented in [2], we rewrite (10.20) based on multiple experiments as

$$\mathcal{Q}(\theta,\hat{\theta}_i)=\mathbf{E}\left\{\log p\left(\{\mathcal{Y}^{(p)},G_{\Delta}^{(p)}\}_{p=1}^{N_{exp}}|\theta\right)\Big|\{\mathcal{Y}^{(p)}\}_{p=1}^{N_{exp}},\hat{\theta}_i\right\}. \tag{10.22}$$

Assuming that the experiments are independent, then using Bayes' theorem, we have that

$$\mathcal{Q}(\theta,\hat{\theta}_i)=\sum_{p=1}^{N_{exp}}\mathbf{E}\left\{\log p\left(\mathcal{Y}^{(p)},G_{\Delta}^{(p)}|\theta\right)\Big|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}, \tag{10.23}$$

and using Bayes' theorem again we express $\log p(\mathcal{Y}^{(p)},G_{\Delta}^{(p)}\Big|\theta)$ as

$$\log p(\mathcal{Y}^{(p)},G_{\Delta}^{(p)}|\theta)=\log p(\mathcal{Y}^{(p)}|G_{\Delta}^{(p)}|\theta)+\log p(G_{\Delta}^{(p)}|\theta). \tag{10.24}$$

Finally, substituting (10.24) into (10.23), the result (10.22) follows. ■

In particular, when $G_{\Delta,k+1}^{(p)}$ and $Y_k^{(p)}$ are independent, proper and Gaussian distributed [83], then, the joint pdf of $\mathcal{Y}^{(p)}$ and $G_{\Delta}^{(p)}$ is given by

$$p(\mathcal{Y}^{(p)},G_{\Delta}^{(p)}|\theta)=\prod_{k=0}^{L-1}p\left(G_{\Delta,k+1}^{(p)},Y_k^{(p)}\Big|G_{\Delta,k}^{(p)},\theta\right)p(G_{\Delta,0}^{(p)}), \tag{10.25}$$

and the joint pdf of $G_{\Delta,k+1}^{(p)}$ and $Y_k^{(p)}$ is given by

$$p\left(G_{\Delta,k+1}^{(p)},Y_k^{(p)}|G_{\Delta,k}^{(p)},\theta\right)\sim N_p\left(\begin{bmatrix}\mu_{\Delta,k}^{(p)}\\ \mu_{y,k}^{(p)}\end{bmatrix};\begin{bmatrix}\sigma_{\Delta,k}^2 & 0\\ 0 & \sigma_{y,k}^2\end{bmatrix}\right) \tag{10.26}$$

where $N_p(\cdot,\cdot)$ represents a proper Gaussian distribution, and

$$G_{\Delta,0}^{(p)}\sim N_p(\mu_{\Delta,-1}^{(p)};\sigma_{\Delta,-1}^2). \tag{10.27}$$

Corollary 10.3.1. *Consider that the pdf of the complete data is given by* (10.25)-(10.26)*, and* $G_{\Delta,0}^{(p)}$ *is given by* (10.27)*. Then, the auxiliary function* $\mathcal{Q}(\theta,\hat{\theta}_i)$ *of the EM algorithm is given by (excluding constant terms)*

$$\begin{aligned}\mathcal{Q}(\theta,\hat{\theta}_i) = &- N_{exp}\left(\sum_{k=0}^{L-1}\log\sigma^2_{\Delta,k}+\sum_{k=0}^{L-1}\log\sigma^2_{y,k}+\log\sigma^2_{\Delta,-1}\right)\\ &-\sum_{p=1}^{N_{exp}}\sum_{k=0}^{L-1}\left[\frac{1}{\sigma^2_{\Delta,k}}\cdot\mathbf{E}\left\{(G^{(p)}_{\Delta,k+1}-\mu^{(p)}_{\Delta,k})(G^{(p)}_{\Delta,k}-\mu^{(p)}_{\Delta,k})^*\Big|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}\right]\\ &-\sum_{p=1}^{N_{exp}}\sum_{k=0}^{L-1}\frac{1}{\sigma^2_{y,k}}\mathbf{E}\left\{(Y^{(p)}_k-\mu^{(p)}_{y,k})(Y^{(p)}_k-\mu^{(p)}_{y,k})^*\Big|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}\\ &-\sum_{p=1}^{N_{exp}}\frac{1}{\sigma^2_{\Delta,-1}}\mathbf{E}\left\{(G^{(p)}_{\Delta,0}-\mu^{(p)}_{\Delta,-1})(G^{(p)}_{\Delta,0}-\mu^{(p)}_{\Delta,-1})^*\Big|\mathcal{Y}^{(p)},\hat{\theta}_i\right\},\end{aligned}\tag{10.28}$$

where * denotes the complex conjugate.

Proof. From (10.20) and (10.25) we have that

$$\mathcal{Q}(\theta,\hat{\theta}_i)=\sum_{p=1}^{N_{exp}}\mathbf{E}\left\{\sum_{k=0}^{L-1}\log p\left(G^{(p)}_{\Delta,k+1},Y^{(p)}_k|G^{(p)}_{\Delta,k}\right)\Big|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}+\sum_{p=1}^{N_{exp}}\mathbf{E}\left\{\log p(G^{(p)}_{\Delta,0})|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}\tag{10.29}$$

From (10.26) we deduce the following

$$\begin{aligned}\log p\left(G^{(p)}_{\Delta,k+1},Y^{(p)}_k|G^{(p)}_{\Delta,k},\theta\right) &= -\log\pi^2-\log\det\left(\begin{bmatrix}\sigma^2_{\Delta,k}&0\\0&\sigma^2_{y,k}\end{bmatrix}\right)\\ &-\left(\begin{bmatrix}G^{(p)}_{\Delta,k+1}\\Y^{(p)}_k\end{bmatrix}-\begin{bmatrix}\mu^{(p)}_{\Delta,k+1}\\\mu^{(p)}_{y,k}\end{bmatrix}\right)^H\begin{bmatrix}\sigma^2_{\Delta,k}&0\\0&\sigma^2_{y,k}\end{bmatrix}^{-1}\left(\begin{bmatrix}G^{(p)}_{\Delta,k+1}\\Y^{(p)}_k\end{bmatrix}-\begin{bmatrix}\mu^{(p)}_{\Delta,k+1}\\\mu^{(p)}_{y,k}\end{bmatrix}\right),\\ &= -\log\pi^2-\log\sigma^2_{\Delta,k}-\log\det\sigma^2_{y,k}-\frac{1}{\sigma^2_{\Delta,k}}(G^{(p)}_{\Delta,k+1}-\mu^{(p)}_{\Delta,k})^*(G^{(p)}_{\Delta,k+1}-\mu^{(p)}_{\Delta,k})\\ &-\frac{1}{\sigma^2_{y,k}}(Y^{(p)}_k-\mu^{(p)}_{y,k})^*(Y^{(p)}_k-\mu^{(p)}_{y,k}).\end{aligned}\tag{10.30}$$

In a similar way, for $G^{(p)}_{\Delta,0}$, we have:

$$\log p(G^{(p)}_{\Delta,0}) = -\log\pi-\log\det\sigma^2_{\Delta,-1}-\frac{1}{\sigma^2_{\Delta,-1}}(G^{(p)}_{\Delta,0}-\mu^{(p)}_{\Delta,-1})^*(G^{(p)}_{\Delta,0}-\mu^{(p)}_{\Delta,-1}),\tag{10.31}$$

then substituting (10.30) and (10.31) into (10.29) we obtain (10.28). ■

Once $\mathcal{Q}(\theta,\hat{\theta}_i)$ has been calculated, we can maximize it using standard optimization tools (e.g. the Newton-Raphson method).

Remark 10.3.3. *The conditional mean and covariances in the above expression (given $\mathcal{Y}$) is available, can be obtained by using the Kalman smoother algorithm (for more details, see e.g [1, 44, 58, 100]).* ▽▽▽

Case Study

In practice, due to physical properties of real systems, typical error bounds roughly grow with the frequency [119]. To capture this idea, the stochastic process $G_{\Delta,k}$ can be modelled as a random walk in the frequency domain as was done in [47]. In this section, we explore this idea, and consider that $\widetilde{G}_\Delta$ is a random walk in the frequency domain, with $G_\Delta(j\omega_0) = 0$. Then (10.18) and (10.19) can be rewritten as

$$G^{(p)}_{\Delta,k+1} = G^{(p)}_{\Delta,k} + W^{(p)}_k, \tag{10.32}$$

$$Y^{(p)}_k = G^{(p)}_k(\beta)(1 + G^{(p)}_{\Delta,k})U^{(p)}_k + V^{(p)}_k, \tag{10.33}$$

where $\beta \in \Omega$ contains the parameters of the nominal model, and $W^{(p)}_k$ and $V^{(p)}_k$ are two independent circularly symmetric white Gaussian noise sequences[1], with mean and variance given by:

$$\begin{bmatrix} W^{(p)}_k \\ V^{(p)}_k \end{bmatrix} \sim \mathcal{N}_p \left(0; \begin{bmatrix} \sigma^2_w & 0 \\ 0 & \sigma^2_v \end{bmatrix} \right). \tag{10.34}$$

We are interested in simultaneously estimating the nominal model parameters β, and the covariances $\gamma = [\ \sigma^2_w \ \ \sigma^2_v\ ]^\top$. We denote the set of parameters to be estimated as $\theta = [\ \beta^T \ \ \gamma^T\ ]^\top$.

Lemma 10.3.2. *Consider the system given by (10.32)-(10.33), where the random noise sequences are given by (10.34). The auxiliary function $\mathcal{Q}^{(p)}(\theta, \hat{\theta}_i)$, for the p-th experiment, of the EM algorithm is given by,*

[1]Note that circularly symmetric Gaussian random variables have the following properties: (i) the real and imaginary parts which are independent, and (ii) the real and imaginary parts have the same covariance matrix.

$$
\begin{aligned}
\mathcal{Q}^{(p)}(\theta, \hat{\theta}_i) = & -L \log \sigma_w^2 - L \log \sigma_v^2 \\
& - \frac{1}{\sigma_w^2} \sum_{k=0}^{L-1} \Big[\mathbf{E} \left\{ G_{\Delta,k+1}^{(p)} G_{\Delta,k+1}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} \\
& - 2\mathbf{Re} \left\{ \mathbf{E} \left\{ G_{\Delta,k+1}^{(p)} G_{\Delta,k}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} \right\} \\
& + \mathbf{E} \left\{ G_{\Delta,k}^{(p)} G_{\Delta,k}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} \Big] \\
& + \frac{1}{\sigma_v^2} \sum_{k=0}^{L} \Big[|Y_k^{(p)}|^2 - 2\mathbf{Re} \left\{ Y_k^{(p)} U_k^{(p)*} G_k(\beta)^* \right\} \\
& - 2\mathbf{Re} \left\{ Y_k^{(p)} \mathbf{E} \left\{ G_{\Delta,k}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} G_k(\beta)^* U_k^{(p)*} \right\} \\
& + |G_k(\beta) U_k^{(p)}|^2 \Big(1 + 2\mathbf{Re} \left\{ \mathbf{E} \left\{ G_{\Delta,k}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} \right\} \\
& + \mathbf{E} \left\{ G_{\Delta,k}^{(p)} G_{\Delta,k}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} \Big) \Big],
\end{aligned}
\tag{10.35}
$$

where $|\cdot|$ *and* $\mathbf{Re}\{\cdot\}$ *denote the magnitude and the real part of a complex number, respectively.*

Proof. In (10.28) consider that the term inside of the first summation is

$$
\begin{aligned}
\mu_{\Delta,k}^{(p)} &= G_{\Delta,k}^{(p)}, \\
\sigma_{\Delta,k}^2 &= \sigma_w^2, \\
\mu_{y,k}^{(p)} &= G_k(\beta)(1 + G_{\Delta,k}^{(p)}) U_k^{(p)}, \\
\sigma_{y,k}^2 &= \sigma_v^2,
\end{aligned}
$$

and $G_{\Delta,0}^{(p)} = 0$.

We then have that

$$
\begin{aligned}
\sum_{k=0}^{L-1} \log \sigma_{\Delta,k}^2 &= L \log \sigma_w^2, \\
\sum_{k=0}^{L-1} \log \sigma_{y,k}^2 &= L \log \sigma_v^2.
\end{aligned}
$$

The term in the first summation in (10.28) is, then, given by

$$
\begin{aligned}
\mathbf{E} \left\{ (G_{\Delta,k+1}^{(p)} - \mu_{\Delta,k}^{(p)})(G_{\Delta,k}^{(p)} - \mu_{\Delta,k})^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} = & \mathbf{E} \left\{ G_{\Delta,k+1}^{(p)} G_{\Delta,k+1}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} \\
& - \mathbf{E} \left\{ G_{\Delta,k+1}^{(p)} G_{\Delta,k}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} \\
& - \mathbf{E} \left\{ G_{\Delta,k}^{(p)} G_{\Delta,k+1}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\} \\
& + \mathbf{E} \left\{ G_{\Delta,k}^{(p)} G_{\Delta,k}^{(p)*} | \mathcal{Y}^{(p)}, \hat{\theta}_i \right\},
\end{aligned}
$$

which is equivalent to the term inside of the first summation in (10.35). On the other hand, the term in the second summation in (10.28) can be expanded as

$$\begin{aligned}\mathbf{E}\left\{(Y_k^{(p)}-\mu_{y,k}^{(p)})(Y_k^{(p)}-\mu_{y,k}^{(p)})^*|\mathcal{Y}^{(p)},\hat{\theta}_i\right\} &= |Y_k^{(p)}|^2 - Y_k^{(p)}U_k^{(p)*}G_k(\beta)^* - G_k(\beta)U_k^{(p)}Y_k^{(p)*}\\ &- Y_k^{(p)}\mathbf{E}\left\{G_{\Delta,k}^{(p)*}|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}G_k(\beta)^*U_k^{(p)*}\\ &- G_k(\beta)\mathbf{E}\left\{G_{\Delta,k}^{(p)}|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}U_k^{(p)}Y_k^{(p)*}\\ &+ |G_k(\beta)U_k^{(p)}|^2\Big(1+2\mathbf{Re}\left\{\mathbf{E}\left\{G_{\Delta,k}^{(p)*}|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}\right\}\\ &+ \mathbf{E}\left\{G_{\Delta,k}^{(p)}G_{\Delta,k}^{(p)*}|\mathcal{Y}^{(p)},\hat{\theta}_i\right\}\Big).\end{aligned}$$

Which is equivalent to the term in the second summation in (10.35). Moreover, given that $G_{\Delta,0}^{(p)}$ is deterministic, the last term in (10.28) is not included in (10.35). ■

The computation of $\mathbf{E}\left\{G_{\Delta,k}G_{\Delta,k}^*|\mathcal{Y},\hat{\theta}_i\right\}$, $\mathbf{E}\left\{G_{\Delta,k+1}G_{\Delta,k}^*|\mathcal{Y},\hat{\theta}_i\right\}$ and $\mathbf{E}\left\{G_{\Delta,k}|\mathcal{Y},\hat{\theta}_i\right\}$ can be carried out using the Kalman smoother (see section 10.3.3).

Remark 10.3.4. *The extension of Lemma 10.3.2, for $V(j\omega_k)$ in (10.33) being coloured noise, can be done by following the ideas in [85].* ▽▽▽

10.3.3 Kalman Smoother

In this section, we describe the Kalman smoother algorithm, which is necessary for the computation of $\mathbf{E}\left\{G_{\Delta,k}G_{\Delta,k}^*|\mathcal{Y},\hat{\theta}_i\right\}$, $\mathbf{E}\left\{G_{\Delta,k+1}G_{\Delta,k}^*|\mathcal{Y},\hat{\theta}_i\right\}$ and $\mathbf{E}\left\{G_{\Delta,k}|\mathcal{Y},\hat{\theta}_i\right\}$.

Consider the following system

$$x_t = Ax_{t-1} + B_t + w_t, \tag{10.36}$$

$$y_t = C_t x_t + D_t + v_t, \tag{10.37}$$

where measurements of y_t are available for $t = 1,\ldots,n$. The Kalman smoother provides estimates for x_t based on the entire data sample $\{y_t\}_{t=1}^n$, for $t \le n$. The Kalman smoother equations are based on the quantities computed by the Kalman Filter. We first describe the Kalman filter, and later we present the Kalman smoother.

We denote by

$$x_{t|s} \triangleq \mathbf{E}\{x_t|y_s, y_{s-1},\ldots,y_1\}$$

and by

$$P_{t|s} = \mathbf{E}\left\{(x_t - x_{t|s})(x_t - x_{t|s})^H|y_s, y_{s-1},\ldots,y_1\right\}$$

We assume that the initial condition $x_{0|0}$ has mean μ and variance P_0. Then the Kalman Filter for $t = 1, \ldots, n$ is given by (see e.g. [101])

$$x_{t|t-1} = Ax_{t-1|t-1} + B_t, \tag{10.38}$$

$$P_{t|t-1} = AP_{t-1|t-1}A^T + \Sigma_w, \tag{10.39}$$

with

$$x_{t|t} = x_{t|t-1} + K_t(y_t - C_t x_{t|t-1} - D_t), \tag{10.40}$$

$$P_{t|t} = [I - K_t C_t]P_{t|t-1}, \tag{10.41}$$

where

$$K_t = P_{t|t-1}C_t^H[C_t P_{t|t-1}C_t^H + \Sigma_v]^{-1}. \tag{10.42}$$

Once the quantities of the Kalman filter has been computed. The Kalman smoother can be computed for $t = n, n-1, \ldots, 1$ by (see e.g. [101])

$$x_{t-1|n} = x_{t-1|t-1} + J_t(x_{t|n} - x_{t|t-1}), \tag{10.43}$$

$$P_{t-1|n} = P_{t-1|t-1} + J_{t-1}(P_{t|n} - P_{t|t-1})J_{t-1}^H, \tag{10.44}$$

where

$$J_{t-1} = P_{t-1|t-1}A^T[P_{t|t-1}]^{-1}. \tag{10.45}$$

In our case, we need to compute

$$M_{t|n} = \mathbf{E}\left\{(x_t - x_{t|n})(x_{t-1} - x_{t-1|n})^H | y_n, y_{n-1}, \ldots, y_1\right\}.$$

This expectation can be obtained using the quantities in the Kalman smoothing, by defining

$$M_{n|n} = [I - K_n C_n]AP_{n-1|n-1}, \tag{10.46}$$

and for $t = n, n-1, \ldots, 2$

$$M_{t-1,|n} = P_{t-1|t-1}J_{t-2}^H + J_{t-1}(M_{t|n} - AP_{t-1|t-1})J_{t-2}^H. \tag{10.47}$$

For the system (10.32)-(10.33) we have: $A = 1$, $B_t = 0$, $C_t = G_t(\beta)U_t$, and $D_t = G_t(\beta)U_t$.

10.4 Numerical examples

In this section we present two numerical examples that illustrate the performance of the proposed method.

10.4.1 Example 1

Consider the uncertain system defined in (10.32) and (10.33) where the nominal model is given by

$$G(\theta, z) = \frac{0.12}{z^2 - 1.3z + 0.42}, \tag{10.48}$$

and where noise in the model uncertainty description has variance$\sigma_w^2 = 4 \cdot 10^{-4}$, and the measurement noise has variance $\sigma_v^2 = 0.1$. The input signal consists of $L = 50$ complex-valued samples, $\{U_k\}_{k=0}^{L-1}$, generated as a proper Gaussian sequence with zero mean and variance $\sigma_u^2 = 4$.

We run $N_{exp} = 20$ different experiments each with $L = 50$ samples, which correspond to N_{exp} realizations of the noise and the uncertainty.

Figure 10.1 shows the magnitude and phase of the frequency response of different realizations of undermodelling, $G_{\Delta,k}^{(i)}$, and the frequency response of the estimated nominal model $G_k(\hat{\beta})$. In Figure 10.1a the magnitude of G_k and $G_k(\hat{\beta})$ is shown. We can see that the estimated model follows the characteristics of the realizations of the true system (10.48). The estimated model is given by $G(\hat{\beta}, z) = 0.116/(z^2 - 1.307z + 0.423)$, $\hat{\sigma}_w^2 = 0.0033$, and $\hat{\sigma}_v^2 = 0.0997$.

10.4.2 Example 2

In this example we compare the proposed approach, denoted as SE-EM, with two existing methods, namely: Set Membership (SM) estimation (see e.g. [78]), and the Model Error Modelling (MEM) approach of [65]. To perform this comparison we use the *i4c: Identification for Control Package* [92].

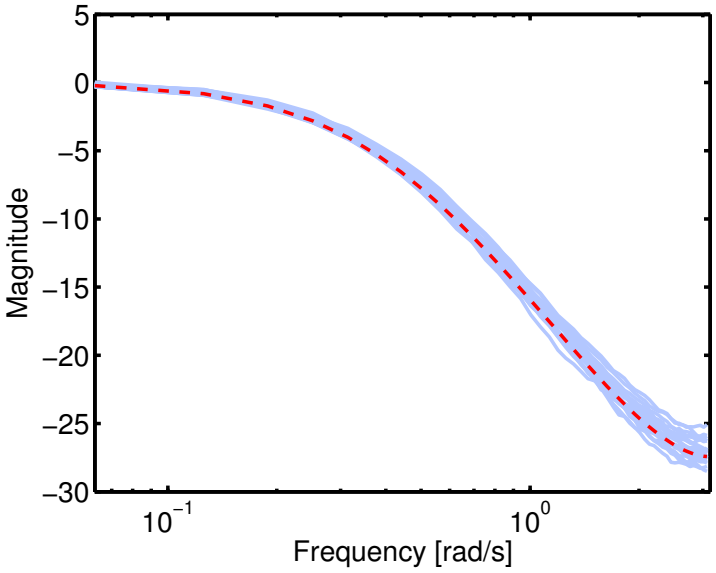
Consider the following true data generating system

$$y_t = \frac{q^{-1} + 0.5q^{-2}}{1 - 2.2q^{-1} + 2.42q^{-2} - 1.87q^{-3} + 0.7225q^{-4}} u_t + w_t \tag{10.49}$$

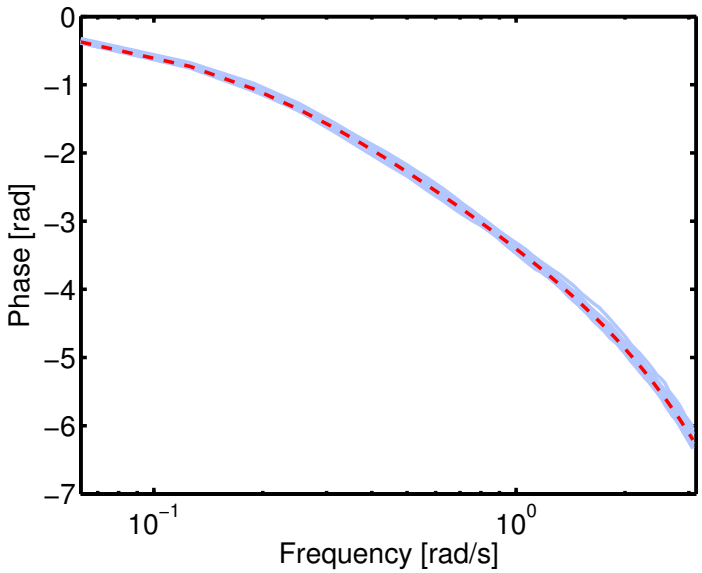
with the input u_t being a PseudoRandom Binary Signal (PRBS) with clock period 5. The disturbance w_t is chosen to be the signal registered at the Charles F. Richter Seismological Laboratory in October, 1989 during the Loma Prieta earthquake (This signal has been made available by The MathWorks Inc.). We assume that the signal u_t is independent of the signal w_t. Figure 10.2 shows the noise-free part of the output and the disturbance signal w_t.

We describe the data generating system in (10.49) by a nominal second order Output Error model, using MEM, SM, and SE-EM.

In MEM the model error is given by a Box-Jenkins model with a Finite Impulse Response with 20

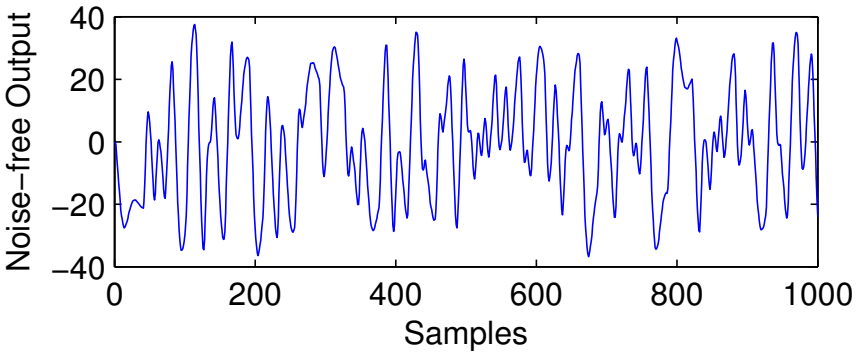


(a) Magnitude [dB]

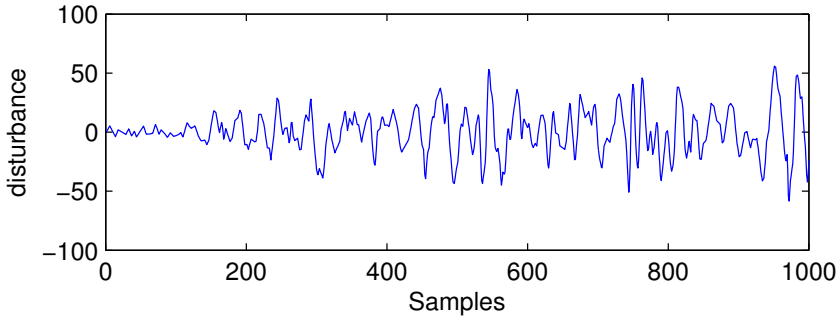


(b) Phase [rad]

Figure 10.1: Magnitude and Phase of the frequency response of different realizations of undermodelling (light blue and continuous line), and the frequency response of the estimated nominal model (red-dashed line).



(a) noise-free part of the output



(b) noise term, w_t

Figure 10.2: Noise-free component and noise component of the output.

parameters for the input and the disturbance model is a fifth order rational transfer function.

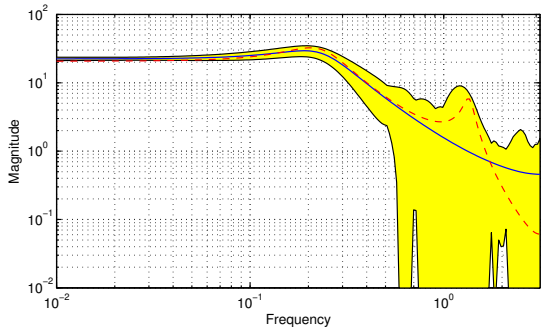
For SM we use the upper bound $\delta = 59.098$. This upper bound was found by incrementing δ by 10% each step until the initial set becomes feasible.

For SE-EM we consider two uncertainty models: (i) An uncertainty model that considers a random walk (RW) in the frequency domain. For this uncertainty model we divide the data into $N_{exp} = 25$ parts. It is assumed the data corresponds to 25 independent experiments. (ii) The second uncertainty model corresponds to zero-mean proper Gaussian noise in the frequency domain.

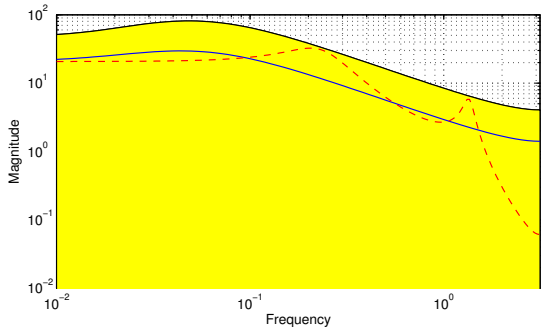
Figures 10.3a-10.3d show the estimated models with the corresponding 99% confidence uncertainty regions (except for SM where the region corresponds to 100% certainty since this method assumes bounded errors). Figure 10.3a shows the uncertainty description obtained by MEM. Notice that the bounds covers the true system response. Figure 10.3b shows the uncertainty region obtained by SM estimation. Notice that the upper bound covers the true system response. Note however, that no lower bound is provided. Figure 10.3c shows the uncertainty region obtained by SE-EM when the uncertainty model has constant variance over all the frequencies. The bound again cover the true system response. However, the region designated for the true system is not as tight as for MEM. Figure 10.3d shows the uncertainty region obtained by SE-EM when the uncertainty model is a random walk in the frequency domain. In this case the bound again covers the true system response. The uncertainty quantification has a less complex shape than for MEM.

Notice that the nominal models obtained in MEM (by using PEM) and SE-EM differ slightly from each other. Also notice that SM and SE-EM deliver nominal models that are affected by the shape of the uncertainty model.

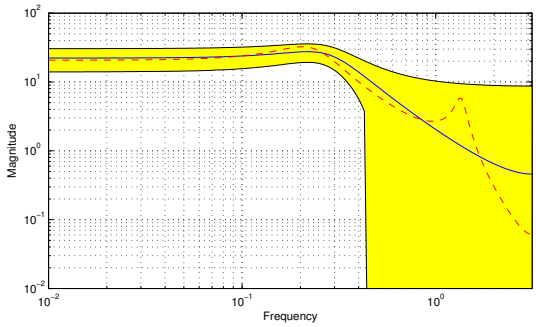
Although it is dangerous to make general conclusions from a single experiment, we do note that, based in this example, MEM and SE-EM (with RW undermodelling) provide tightest description of undermodelling. Also, SE-EM has the potential advantage over MEM that the final undermodelling region is simpler. The latter may be useful in some applications e.g. subsequent robust control design.



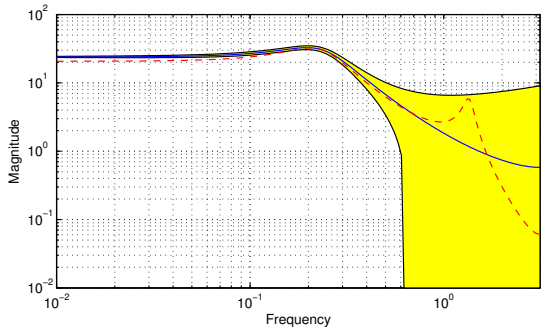
(a) MEM



(b) SM



(c) SE-EM with constant variance undermodelling



(d) SE-EM (with RW undermodelling)

Figure 10.3: Estimated model (blue-continuous line), corresponding uncertainty 99% confidence region (100% for SM), and the true system (red-discontinuous line) for several methods.

10.5 Chapter Summary

In this chapter we have presented an approach to uncertainty modelling based on Stochastic Embedding. Within this framework, we have developed a new method to estimate the model uncertainty based on the EM algorithm. The novel feature of our algorithm is that allows us to estimate the nominal and the uncertainty models simultaneously. We have presented some numerical examples that illustrate the benefits of the proposed approach for system identification.

11

Conclusions

11.1 Overview

In the first part of the thesis, we have seen that including rank constraints in an optimization problem is a difficult task. We have presented a novel approach to rank-constrained optimization. We also have described a novel form to represent rank constraints for non-square real matrices. Moreover, we have extended the results to the related problem of cardinality-constrained optimization. We have shown that the proposed approach may have significant impact in a large number of applications.

In the second part of the thesis we have studied the problem of uncertainty quantification when the system lies in the model set but the amount of data available is small. We have developed a novel approach to deal with parametric uncertainty under these conditions. Also, we have applied the results to moving horizon estimation problem.

In the third part of the thesis, we have studied the problem of modelling the model uncertainty. Our approach uses EM and allows one to estimate the nominal and the uncertainty models, simultaneously.

11.2 Summary of contributions by chapter

In the first part of the thesis we have addressed the problem of managing model complexity by solving rank-constrained optimization problems.

In chapter 2 we have presented an equivalent representation for rank-constrained optimization problems. Using this representation, we have developed a novel algorithm that satisfactorily imposes rank-constraints in the optimization problem. This algorithm, based on alternating convex

optimization, provides a (provably) suboptimal solution to the rank-constrained optimization problem. We have also developed a second algorithm, based on a global optimization method, which allows one to overcome the sub-optimality issues of the earlier approach.

In chapter 3 we have analysed the problem of factor analysis. We have proposed a novel approach that relaxes the assumption that the idiosyncratic noises are uncorrelated. We have assumed instead that the noise covariance matrix is sparse. The proposed method is based on the methods developed earlier in chapter 2 and ensures that the idiosyncratic covariance matrix is positive semidefinite. A numerical example was used to show the efficacy of the approach. It has been shown that, in many cases, the new algorithm outperforms existing methods.

In chapter 4, we developed a new approach to describe a rank constraints for general non-square real matrices.

In chapter 5 we have applied the method developed in chapter 4 to the problem of impulse response estimation subject to prior knowledge of the McMillan degree of the system. The contribution of this chapter is to develop a method that can consider the minimization of the predicted output errors subject to a rank-constraint on the Hankel matrix.

In chapter 6 we have extended the results in chapter 4 to the problem of cardinality-constrained optimization. We have developed a novel approach to handle cardinality constraints. The main contributions of this method are that it can incorporate cardinality constraints as hard constraints. We have extended this idea to handle group constraints. We have used this ideas to develop an algorithm to solve cardinality-constrained optimization problems.

In chapter 7 we have applied the method developed in chapter 6 to model predictive control. The contribution is to develop a method that can handle cardinality constraints on *each control horizon*.

In the second part of the thesis we focus on error quantification issues when the system lies in the model set and the available amount of data is small.

In chapter 8 we have studied the problem of estimation-error quantification for finite data estimation. We have developed a novel method for quantification of estimation errors. The proposed approach allows one to quantify the accuracy of the estimates obtained by Maximum a Posteriori methods.

In chapter 9 we have applied the method developed in chapter 8 to the problem of moving horizon estimation. The contribution of the proposed method is that it offers a possible solution to two

known difficulties. One is the provision of a meaningful arrival cost, and the other is the quantification of the accuracy of the estimates. A numerical example shows that the proposed method gives good performance at reasonable computational cost.

The third part of the thesis focusses on the modelling of model uncertainty when the system does not belong to the model set.

In chapter 10 we have addressed the problem of modelling model uncertainty in dynamic models. We have proposed a systematic methodology using EM to simultaneously estimate the nominal and the uncertainty models. The main advantage of the proposed method over existing approaches is that it can handle a fairly general class of nominal models whilst allowing simultaneous estimation of the nominal and the uncertainty models. This last issue is important since the estimated nominal model is affected by the assumptions made regarding the uncertainty model.

11.3 Future Research

The results of the thesis have the potential to be extended in several directions.

Rank-Constrained Optimization for general non-square real matrices

One of the main results in the thesis, is Theorem 4.2.1, which provides a more general representation of rank constraints. Future research could include the use of this result to incorporate rank constraints for more general rank-constrained optimization problems.

Non-Convex Objective Functions

The global optimization method developed in the first part of the thesis could be extended to handle non-convex objective functions. This can be done by using the general optimization framework of Majorization-Minimization algorithms [57, 61]. Moreover, a similar objective could be pursued by using a global optimization method based on Augmented Lagrangian methods, see for example [14].

Estimation-Error Quantification

Within the method developed in chapter 8 for estimation error quantification for finite data problems, there is no conceptual difficulties to consider distributions other than Gaussian. This idea

could be further explored.

Identifiability for Several Uncertainty Models

One of the main contributions of chapter 10 is to develop a method that allows one to simultaneously estimate the nominal and uncertainty models. This leads to a related question, namely when is it possible to uniquely recover the nominal and the uncertainty models. Future research could include the study of structural identifiability to address this question.

A

EM ALGORITHM

The purpose of this section is to describe one of the main tools used in estimation problems. This tool has been utilized several times in the body of the thesis. The tool is the Expectation-Maximization (EM) algorithm [32]. We provide a description of the EM algorithm and its underlying principles.

The EM algorithm has been used to identify different classes of dynamic systems, such as, continuous time systems using sampled-data [116], finite impulse response systems using quantized systems [68], state-space systems using incomplete data [1], channel estimation in telecommunications [21], bilinear state-space systems [44], and non-linear state-space systems [96].

The EM algorithm aimed at finding the Maximum Likelihood estimate $\hat{\theta}_{ML}$ of a parameter vector θ based on the measured data, $\mathcal{Y}$. Thus, the EM algorithm estimate

$$\hat{\theta}_{ML} = \arg\max_{\theta \in \Theta} p(\mathcal{Y}|\theta) \tag{A.1}$$

where $p(\cdot|\theta)$ denotes a probability density function which is parametrized by a vector $\theta \in \mathbb{R}^d$, and $\Theta \subset \mathbb{R}^d$ denotes the feasible set.

The traditional approach to solve problem (A.1) is to exploit the fact that the cost function is smooth enough for a gradient-based optimization algorithm [66, 102]. By way of contrast, instead of exploiting any smoothness of $p(\mathcal{Y}|\theta)$, the EM algorithm uses the characteristic that $p(\mathcal{Y}|\theta)$ is a probability density function.

The EM algorithm introduces the concept of complete data. It is assumed that, the complete data is composed of the measured data, $\mathcal{Y}$, and also of an unmeasured component known as the "hidden variables", $\mathcal{X}$.

The principles underlying the EM algorithm depend on the elementary definition of conditional

probability $p(\mathcal{Y}|\theta)p(\mathcal{X}|\mathcal{Y},\theta) = p(\mathcal{X},\mathcal{Y}|\theta)$. Thus,

$$\log p(\mathcal{Y}|\theta) = \log p(\mathcal{X},\mathcal{Y}|\theta) - \log p_\theta(\mathcal{X}|\mathcal{Y},\theta). \tag{A.2}$$

The EM algorithm approximates $\log p(\mathcal{X},\mathcal{Y}|\theta)$ by an auxiliary function $\mathcal{Q}(\theta,\theta_i)$ as follows

$$\log p(\mathcal{X},\mathcal{Y}|\theta) \approx \mathcal{Q}(\theta,\theta_i) \triangleq \mathbf{E}\{\log p(\mathcal{X},\mathcal{Y}|\theta)|\mathcal{Y},\theta_i\}. \tag{A.3}$$

where $\mathbf{E}\{\cdot|\theta_i\}$ denotes the expected value with respect to an underlying probability density function defined by an assumption of the parameter vector being θ_i. We denote $L(\theta) \triangleq \log p(\mathcal{Y}|\theta)$. Notice that $L(\theta)$ can be written as

$$\begin{aligned} L(\theta) \triangleq \log p(\mathcal{Y}|\theta) &= \mathbf{E}\{\log p(\mathcal{Y}|\theta)|\mathcal{Y},\theta_i\} && \text{(A.4)} \\ &= \mathbf{E}\{\log p(\mathcal{X},\mathcal{Y}|\theta)|\mathcal{Y},\theta_i\} - \mathbf{E}\{\log p(\mathcal{X}|\mathcal{Y},\theta)|\mathcal{Y},\theta_i\} \\ &= \mathcal{Q}(\theta,\theta_i) - \mathcal{H}(\theta,\theta_i) && \text{(A.5)} \end{aligned}$$

where

$$\mathcal{H}(\theta,\theta_i) \triangleq \mathbf{E}\{\log p(\mathcal{X}|\mathcal{Y},\theta)|\mathcal{Y},\theta_i\} \tag{A.6}$$

Moreover, using Jensen inequality it is possible to establish that $\mathcal{H}(\theta_i,\theta_i) \geq \mathcal{H}(\theta,\theta_i)$ [32]. Then, using (A.5) we can establish that

$$\mathcal{Q}(\theta,\theta_i) > \mathcal{Q}(\theta_i,\theta_i) \Rightarrow L(\theta) > L(\theta_i) \tag{A.7}$$

Hence, any new θ that improves $\mathcal{Q}(\theta,\theta_i)$ must also improve $L(\theta)$.

Based on the above, the EM algorithm is an iterative two step procedure. In the, so called, Expectation step (E-step) one calculates $\mathcal{Q}(\theta,\theta_i)$. Then in the, so called, Maximization step (M-step), one updates the parameters by maximizing $\mathcal{Q}(\theta,\theta_i)$.

In detail, given a current estimate $\hat{\theta}_i \in \Theta$, one iteration of the EM algorithm is given by:

1. *E-step:*

$$\text{Calculate: } \mathcal{Q}(\theta,\hat{\theta}_i) \tag{A.8}$$

2. *M-step:*

$$\text{Estimate: } \hat{\theta}_{i+1} = \arg\max_{\theta\in\Theta} \mathcal{Q}(\theta,\hat{\theta}_i) \tag{A.9}$$

Under quite general conditions the EM algorithm can be proven to converge to a stationary point of the likelihood function [32,114]. In many practical applications this will be a local maximum of the likelihood function [74].

B

Detailed proof of Theorem 4.2.1

In this section we provide a detailed proof for Theorem 4.2.1. This detailed proof is based on the use of Singular Values Decomposition (SVD).

B.1 Preliminaries

We first consider *Sylvester's inequality*, for real matrices, which provides a lower bound for the rank of the product of two matrices.

Proposition B.1.1. *[12, Proposition 2.5.9] Let $A \in \mathbb{R}^{n\times m}$ and $B \in \mathbb{R}^{m\times l}$. Then,*

$$\operatorname{rank}\{A\} + \operatorname{rank}\{B\} \leq m + \operatorname{rank}\{AB\}. \tag{B.1}$$

Next, we review some existing results on Singular Values Decomposition.

Definition B.1.1. *[12, Definition 5.6.1.] Let $A \in \mathbb{R}^{m\times n}$. Then the* singular values *of A are the $\min\{m,n\}$ nonnegative numbers $\sigma_1(A), \ldots, \sigma_{\min\{m,n\}}(A)$, where, for all $i = 1, \ldots, \min\{m,n\}$,*

$$\sigma_i(A) \coloneqq \lambda_i^{1/2}(AA^\top) = \lambda_i^{1/2}(A^\top A). \tag{B.2}$$

Hence,

$$\sigma_1(A) \geq \cdots \geq \sigma_{\min\{m,n\}}(A) \geq 0. \tag{B.3}$$

Note that if $A \in \mathbb{S}^n$ and is positive semidefinite, then $\sigma_i(A) = \lambda_i(A)$ for $i = 1, \ldots, n$.

Definition B.1.2. *[12, Definition 3.1.1. xxii] Let $A \in \mathbb{R}^{n\times n}$. A is orthogonal if $A^\top A = I_n$*

Theorem B.1.1. *[12, Theorem 5.6.4.] Let $A \in \mathbb{R}^{m\times n}$, assume that A is nonzero, let $c \coloneqq \operatorname{rank}\{A\}$, and define $B \coloneqq \operatorname{diag}\{\sigma_1(A), \ldots, \sigma_c(A)\}$. Then, there exists orthogonal matrices $U \in$*

$\mathbb{R}^{m\times m}$ *and* $V \in \mathbb{R}^{n\times n}$ *such that*

$$A = U \begin{bmatrix} B & 0_{c\times(n-c)} \\ 0_{(m-c)\times c} & 0_{(m-c)\times(n-c)} \end{bmatrix} V^\top. \tag{B.4}$$

Let $r \in \mathbb{N}$ such that $c \leq r \leq \min\{m, n\}$ and consider the following block partition

$$A = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \left[\begin{array}{cc|c} B & 0_{c\times(c-r)} & 0_{c\times(n-r)} \\ 0_{(r-c)\times(c-r)} & 0_{(r-c)\times(c-r)} & 0_{(r-c)\times(n-r)} \\ \hline 0_{(m-r)\times(c-r)} & 0_{(m-r)\times(c-r)} & 0_{(m-r)\times(n-r)} \end{array}\right] \begin{bmatrix} V_1^\top \\ V_2^\top \end{bmatrix}. \tag{B.5}$$

where $U_1 \in \mathbb{R}^{m\times r}$, $U_2 \in \mathbb{R}^{m\times(m-r)}$, $V_1 \in \mathbb{R}^{n\times r}$, and $V_2 \in \mathbb{R}^{n\times(n-r)}$.

Based on the block partition (B.5) of the SVD, we establish the following two Lemmas.

Lemma B.1.1. *Let* $A \in \mathbb{R}^{m\times n}$ *with SVD given by* (B.5)*, and let* $W_R \coloneqq V_2V_2^\top$*. Then,* $W_R \in \mathbb{R}^{n\times n}$ *satisfies that*

(i) $W_R \succeq 0$.

(ii) $W_R = W_R^\top$.

(iii) $W_R \preceq I_n$.

(iv) $\operatorname{trace}(W_R) = n - r$.

(v) $AW_R = 0_{m\times n}$.

Proof. From construction, **(i)** is a well known result, see e.g. [12, Fact 3.7.25]. The proof of **(ii)** is straightforward $V_2V_2^\top = (V_2V_2^\top)^\top$. To prove **(iii)** , consider in (B.5) that V is an orthogonal matrix, then $V_1^\top V_1 = I_r$, $V_2^\top V_2 = I_{n-r}$, $V_1V_2^\top = 0_{n\times n}$ and

$$V_1V_1^\top + V_2V_2^\top = I_n \tag{B.6}$$

where $V_1V_1^\top$ is positive semidefinite. Then $I_n - W_R = V_1V_1^\top \succeq 0$. Next we prove **(iv)**, by using the trace operator in (B.6), we have that

$$\begin{aligned} \operatorname{trace}(W_R) &= \operatorname{trace}(I_n) - \operatorname{trace}(V_1V_1^\top) && \text{(B.7)} \\ &= n - \operatorname{trace}(V_1^\top V_1) && \text{(B.8)} \\ &= n - \operatorname{trace}(I_r) = n - r. && \text{(B.9)} \end{aligned}$$

Finally, to prove **(v)**, we note that by using (B.5) we have that

$$A\,W_R = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \left[\begin{array}{cc|c} B & 0_{c\times(c-r)} & 0_{c\times(n-r)} \\ 0_{(r-c)\times(c-r)} & 0_{(r-c)\times(c-r)} & 0_{(r-c)\times(n-r)} \\ \hline 0_{(m-r)\times(c-r)} & 0_{(m-r)\times(c-r)} & 0_{(m-r)\times(n-r)} \end{array}\right] \begin{bmatrix} V_1^\top \\ V_2^\top \end{bmatrix} V_2 V_2^\top \tag{B.10}$$

$$= \begin{bmatrix} U_1 & U_2 \end{bmatrix} \left[\begin{array}{cc|c} B & 0_{c\times(c-r)} & 0_{c\times(n-r)} \\ 0_{(r-c)\times(c-r)} & 0_{(r-c)\times(c-r)} & 0_{(r-c)\times(n-r)} \\ \hline 0_{(m-r)\times(c-r)} & 0_{(m-r)\times(c-r)} & 0_{(m-r)\times(n-r)} \end{array}\right] \begin{bmatrix} V_1^\top V_2 \\ V_2^\top V_2 \end{bmatrix} V_2^\top \tag{B.11}$$

$$= \begin{bmatrix} U_1 & U_2 \end{bmatrix} \left[\begin{array}{cc|c} B & 0_{c\times(c-r)} & 0_{c\times(n-r)} \\ 0_{(r-c)\times(c-r)} & 0_{(r-c)\times(c-r)} & 0_{(r-c)\times(n-r)} \\ \hline 0_{(m-r)\times(c-r)} & 0_{(m-r)\times(c-r)} & 0_{(m-r)\times(n-r)} \end{array}\right] \begin{bmatrix} 0_{r\times(n-r)} \\ I_{n-r} \end{bmatrix} V_2^\top \tag{B.12}$$

$$= \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} 0_{r\times(n-r)} \\ 0_{(m-r)\times(n-r)} \end{bmatrix} V_2^\top \tag{B.13}$$

$$= 0_{m\times n} \tag{B.14}$$

■

Lemma B.1.2. *Let $A \in \mathbb{R}^{m\times n}$ with SVD given by (B.5), and let $W_L \coloneqq U_2 U_2^\top$. Then, $W_L \in \mathbb{R}^{m\times m}$ satisfies that*

(i) $W_L \succeq 0$

(ii) $W_L \preceq I_m$

(iii) $W_L = W_L^\top$

(iv) $\operatorname{trace}(W_L) = m - r$

(v) $W_L A = 0_{m\times n}$

Proof. The proof is similar to the proof of Lemma B.1.1. ■

The following Lemma is one of the key ingredients to prove Theorem 4.2.1.

Lemma B.1.3. *Let $W \in \mathbb{S}^n$, such that $0 \preceq W \preceq I_n$, then*

$$\operatorname{trace}(W) \leq \operatorname{rank}\{W\} \tag{B.15}$$

Proof. Let $c \coloneqq \operatorname{rank}\{W\}$, and define $B \coloneqq \operatorname{diag}\{\sigma_1(AW, \ldots, \sigma_c(AW\}$ and consider the SVD

$$W = U \begin{bmatrix} B & 0_{c\times(n-c)} \\ 0_{(m-c)\times c} & 0_{(m-c)\times(n-c)} \end{bmatrix} U^\top. \tag{B.16}$$

then

$$\operatorname{trace}(W) = \sum_{k=1}^{c} \sigma_i(W) \tag{B.17}$$

since $W \in \mathbb{S}^n$ and $W \succeq 0$, we have that $\sigma_i(W) = \lambda_i(W)$ for all $i = 1, \ldots, n$. Furthermore, since $W \preceq I_n$ we have that $1 \geq \lambda_1(W)$, see [12, Lemma 8.4.1 iii], then $1 \geq \lambda_1(W) \geq \cdots \geq \lambda_c(W) \geq \cdots \geq \lambda_n(W)$. Finally, we have that

$$\operatorname{trace}(W) = \sum_{k=1}^{c} \lambda_i(W) \leq c = \operatorname{rank}\{W\} \tag{B.18}$$

■

B.2 Proof of Theorem 4.2.1

Finally, we establish Theorem 4.2.1.

Proof. Lemma B.1.1 establishes that **(i)** $\Longrightarrow$ **(ii)**. Lemma B.1.2 proves that **(i)** $\Longrightarrow$ **(iii)**

Next, we prove **(ii)** $\Longrightarrow$ **(i)**. From Lemma B.1.3 we have that

$$\operatorname{trace}(W_R) \leq \operatorname{rank}\{W_R\} \tag{B.19}$$

On the other hand, by using *Sylvester's Inequality*, we have that

$$\operatorname{rank}\{G\} + \operatorname{rank}\{W_R\} \leq n + \operatorname{rank}\{GW_R\} \tag{B.20}$$

Then, by using (B.19), we have

$$\operatorname{rank}\{G\} + \operatorname{trace}(W_R) \leq n + \operatorname{rank}\{GW_R\} \tag{B.21}$$

Then by using the fact that $\operatorname{rank}\{GW_R\} = \operatorname{rank}\{0_{m\times n}\} = 0$ we obtain

$$\operatorname{rank}\{G\} \leq n - \operatorname{trace}(W_R) \tag{B.22}$$

Since $W_R \in \Phi_{n,r}$, we have that $\operatorname{trace}(W_R) = n - r$. Then

$$\operatorname{rank}\{G\} \leq r. \tag{B.23}$$

This completes the proof that **(ii)** $\Longrightarrow$ **(i)**. The prove **(iii)** $\Longrightarrow$ **(i)** is similar to the proof of **(ii)** $\Longrightarrow$ **(i)**. ■

Bibliography

[1] J. C. Agüero, W. Tang, J. I. Yuz, R. D., and G. C. Goodwin. Dual time–frequency domain system identification. *Automatica*, 48(12):3031 – 3041, 2012.

[2] J. C. Agüero, J. I. Yuz, and G. C. Goodwin. Frequency domain identification of MIMO state space models using the EM algorithm. In *European Control Conference - ECC'07*, Kos, Greece, 2-5 July 2007.

[3] R. P. Aguilera, R. A. Delgado, D. Dolz, and J. C. Agüero. Quadratic MPC with ℓ_0-input constraint. In *19th IFAC World Congress*, Cape Town, South Africa, 2014.

[4] H. Akaike. A new look at the statistical model identification. *Automatic Control, IEEE Transactions on*, 19(6):716–723, Dec 1974.

[5] F. A. Al-Khayyal and J. E. Falk. Jointly constrained biconvex programming. *Mathematics of Operations Research*, 8(2):273–286, 1983.

[6] A. Alessandri, M. Baglietto, and G. Battistelli. Moving-horizon state estimation for nonlinear discrete-time systems: New stability results and approximation schemes. *Automatica*, 44(7):1753–1765, 2008.

[7] A. Alessandri, M. Baglietto, G. Battistelli, and V. Zavala. Advances in moving horizon estimation for nonlinear systems. In *49th IEEE Conference on Decision and Control (CDC)*, pages 5681–5688, 2010.

[8] K. Aljanaideh, B. J. Coffer, and D. S. Bernstein. Closed-loop identification of unstable systems using noncausal FIR models. In *American Control Conference (ACC), 2013*, pages 1669–1674. IEEE, 2013.

[9] R. Andreani, J. Martínez, and M. Schuverdt. On the relation between constant positive linear dependence condition and quasinormality constraint qualification. *Journal of Optimization Theory and Applications*, 125(2):473–483, 2005.

[10] J. Bai. Inferential theory of factor models of large dimensions. *Econometrica*, 71:135–172, 2003.

[11] T. Başar and P. Bernhard. *H_∞ Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*. Birkhäuser Boston, Boston, MA, 2008.

[12] D. S. Bernstein. *Matrix mathematics: theory, facts, and formulas*. Princeton University Press, 2009.

[13] E. Birgin, J. Martínez, and L. Prudente. Augmented lagrangians with possible infeasibility and finite termination for global nonlinear programming. *Journal of Global Optimization*, 58(2):207–242, 2014.

[14] E. G. Birgin, C. Floudas, and J. M. Martínez. Global minimization using an augmented lagrangian method with variable lower-level constraints. *Mathematical Programming*, 125(1):139–162, 2010.

[15] S. Bittanti and M. Lovera. Bootstrap-based estimates of uncertainty in subspace identification methods. *Automatica*, 36:1605–1615, 2000.

[16] J. Boivin and S. Ng. Are more data always better for factor analysis? *Journal of Econometrics*, 132(1):169–194, 2006.

[17] S. P. Boyd and J. Mattingley. Branch and bound methods. *Notes for EE364b, Stanford University*, pages 2006–07, 2007.

[18] M. C. Campi and E. Weyer. Finite sample properties of system identification methods. *IEEE Transactions on Automatic Control*, 47:1329–1334, 2002.

[19] M. C. Campi and E. Weyer. Guaranteed non asymptotic confidence regions in system identification. *Automatica*, 41:1751–1764, 2005.

[20] E. J. Candes, J. K. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Info. Th.*, 52(2):489–509, February 2006.

[21] R. Carvajal, J. C. Agüero, B. I. Godoy, and G. C. Goodwin. EM-based channel estimation in OFDM systems with phase noise. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1–5. IEEE, 2011.

[22] G. Chamberlain and M. Rotchschild. Arbitrage factor structure, and mean-variance analysis of large asset markets. *Econometrika*, 51(1281–1304), 1983.

[23] C. Chatfield. Model uncertainty, data mining and statistical inference. *Journal of the Royal Statistical Society, Series A*, 158(3):419–466, 1995.

[24] T. Chen, H. Ohlsson, G. C. Goodwin, and L. Ljung. Kernel selection in linear system identification part ii: A classical perspective. In *50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, Orlando, FL, USA, December 2011.

[25] Z. Chen. Bayesian filtering: From Kalman filters to particle filters, and beyond. *Adaptive Systems Lab., McMaster University., Hamilton, Ontario, Canada.*, 2003.

[26] J. Clausen. Branch and bound algorithms-principles and examples. *Department of Computer Science, University of Copenhagen*, pages 1–30, 1999.

[27] W. Cook. Markowitz and Manne+ Eastman+ Land and Doig= Branch and Bound. *Documenta Mathematica: Optimization Stories*, pages 227–238, 2012.

[28] A. d'Aspremont. A semidefinite representation for some minimum cardinality problems. In *42nd IEEE Conference on Decision and Control, 2003*, volume 5, pages 4985–4990, 2003.

[29] J. Dattorro. *Convex optimization and Euclidean distance geometry*. Meboo Publishing, USA, 2005.

[30] M. Deistler, B. Anderson, A. Filler, C. Zinner, and W. Chen. Generalized linear dynamic factor models: An approach via singular autoregressions. *European Journal of Control*, 16(3):211–224, 2010.

[31] R. A. Delgado, G. C. Goodwin, R. Carvajal, and J. C. Agüero. A novel approach to model error modelling using the expectation-maximization algorithm. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 7327–7332. IEEE, 2012.

[32] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[33] M. Diehl, H. Ferreau, and N. Haverbeke. Efficient Numerical Methods for Nonlinear MPC and Moving Horizon Estimation. In L. Magni, D. Raimondo, and F. Allgöwer, editors, *Nonlinear Model Predictive Control*, volume 384 of *Lecture Notes in Control and Information Sciences*, pages 391–417. Springer Berlin / Heidelberg, 2009.

[34] C. Doz, D. Giannone, and L. Reichlin. A quasi–maximum likelihood approach for large, approximate dynamic factor models. *Review of economics and statistics*, 94(4):1014–1024, 2012.

[35] W. J. Dunstan and R. R. Bitmead. Empirical estimation of parameter distributions in system identification. In *Proc. of the 13th IFAC Symposium on system Identification*, Rotterdam, The Netherlands, 2003.

[36] G. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1:211–218, 1936.

[37] M. Fazel. A rank minimization heuristic with application to minimum order system approximation. In *Proceedings of the American Control Conference*, pages 4734–4739. IEEE, 2001.

[38] M. Fazel. *Matrix Rank Minimization with Applications*. PhD thesis, Standford University, Standford, CA, 2002.

[39] M. Fazel, H. Hindi, and S. P. Boyd. Log-det heuristic for matrix rank minimization with applications to Hankel and euclidean distance matrices. In *American Control Conference*, volume 3, pages 2156–2162, 2003.

[40] P. Findeisen. Moving horizon state estimation of discrete time systems. Master's thesis, University of Wisconsin-Madison, 1997.

[41] M. Fukuda and M. Kojima. Branch-and-cut algorithms for the bilinear matrix inequality eigenvalue problem. *Computational Optimization and Applications*, 19(1):79–105, 2001.

[42] S. Garatti and R. R. Bitmead. On resampling and uncertainty estimation in linear system identification. *Automatica*, 46(5):785 – 795, 2010.

[43] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression*. Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, 1992.

[44] S. Gibson, A. Wills, and B. M. Ninness. Maximum-likelihood parameter estimation of bilinear systems. *Automatic Control, IEEE Transactions on*, 50(10):1581 – 1596, oct. 2005.

[45] G. Goodwin, A. Feuer, and C. Müller. Sequential bayesian filtering via minimum distortion filtering. *Control: Three Decades of Progress: Dedicated to Chris Byrnes and Anders Lindquist*, 2010.

[46] G. C. Goodwin. Some observations on robust estimation and control. In *7th IFAC Symposium on Identification and System Parameter Estimation*, York, UK, 1985.

[47] G. C. Goodwin, J. H. Braslavsky, and M. M. Seron. Non-stationary stochastic embedding for transfer function estimation. *Automatica*, 38(1):47 – 62, 2002.

[48] G. C. Goodwin, M. Gevers, and B. Ninness. Quantifying the error in estimated transfer functions with application to model order selection. *Automatic Control, IEEE Transactions on*, 37(7):913–928, 1992.

[49] G. C. Goodwin and M. E. Salgado. Stochastic embedding approach for quantifying uncertainty in the estimation of restricted complexity models. *International Journal of Adaptive Control and Signal Processing*, 3(4):333–356, 1989.

[50] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEEE Proceedings F*, 140(2):107–113, 1993.

[51] J. Gorski, F. Pfeuffer, and K. Klamroth. Biconvex sets and optimization with biconvex functions: a survey and extensions. *Mathematical Methods of Operations Research*, 66(3):373–407, 2007.

[52] M. C. Grant, S. P. Boyd, and Y. Ye. CVX: Matlab software for disciplined convex programming (web page and software) v2.0. *Available at http://cvxr.com/cvx*, April 2011.

[53] C. Grossmann, C. Jones, and M. Morari. System identification with missing data via nuclear norm regularization. In *Presented at: European Control Conference*, volume 23, page 26, 2009.

[54] E. L. Haseltine and J. B. Rawlings. Critical evaluation of extended Kalman filtering and moving-horizon estimation. *Industrial & engineering chemistry research*, 44(8):2451–2460, 2005.

[55] M. R. Hestenes. Multiplier and gradient methods. *Journal of optimization theory and applications*, 4(5):303–320, 1969.

[56] H. Hjalmarsson, J. S. Welsh, and C. R. Rojas. Identification of box-jenkins models using structured ARX models and nuclear norm relaxation. In *Proceedings of the 16th IFAC Symposium on System Identification (SYSID 2012)(accepted for publication)*, 2012.

[57] D. R. Hunter and K. Lange. A tutorial on MM algorithms. *The American Statistician*, 58(1):30–37, 2004.

[58] A. H. Jazwinski. *Stochastic processes and filtering theory*. Number v. 63 in Mathematics in science and engineering. Academic Press, 1970.

[59] S.-J. Kim and Y.-H. Moon. Structurally constrained H_2 and H_∞ control: A rank-constrained LMI approach. *Automatica*, 42(9):1583 – 1588, 2006.

[60] Y. Kim, J. Kim, and Y. Kim. Blockwise sparse regression. *Statistica Sinica*, 16(2):375, 2006.

[61] K. Lange, D. R. Hunter, and I. Yang. Optimization transfer using surrogate objective functions. *Journal of Computational and Graphical Statistics*, 9(1):1–20, 2000.

[62] K. Lee and Y. Bresler. ADMiRA: Atomic decomposition for minimum rank approximation. *IEEE Transactions on Information Theory*, 56(9):4402–4416, 2010.

[63] F. Liese and I. Vajda. On divergences and informations in statistics and information theory. *Information Theory, IEEE Transactions on*, 52(10):4394–4412, 2006.

[64] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Y. Ma. Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 61, 2009.

[65] L. Ljung. Model validation and model error modelling. In *The Åström symposium on control*, Lund, Sweden, 1999.

[66] L. Ljung. *System Identification: Theory for the user*. Prentice Hall, 2nd edition, 1999.

[67] J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[68] D. Marelli, B. Godoy, and G. Goodwin. A scenario-based approach to parameter estimation in state-space models having quantized output data. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 2011 –2016, dec. 2010.

[69] I. Markovsky. Structured low-rank approximation and its applications. *Automatica*, 44(4):891–909, 2008.

[70] I. Markovsky. How effective is the nuclear norm heuristic in solving data approximation problems? In *Proc. of the 16th IFAC Symposium on System Identification*, pages 316–321, Brussels, 2012.

[71] I. Markovsky. *Low Rank Approximation: Algorithms, Implementation, Applications*. Communications and Control Engineering. Springer, 2012.

[72] I. Markovsky. Recent progress on variable projection methods for structured low-rank approximation. *Signal Processing*, 96PB:406–419, 2014.

[73] G. P. McCormick. Computability of global solutions to factorable nonconvex programs: Part i—convex underestimating problems. *Mathematical programming*, 10(1):147–175, 1976.

[74] G. J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley, 1997.

[75] J. Merimaa, T. Peltonen, and T. Lokki. Concert hall impulse responses-Pori, Finland. *Report and data available at: http://www. acoustics. hut. fi/projects/poririrs*, 2005.

[76] G. Meyer. *Geometric optimization algorithms for linear regression on fixed-rank matrices*. PhD thesis, University of Liège, Belgium, 2011.

[77] M. Milanese, J. P. Norton, H. Piet-Lahanier, and E. Walter, editors. *Bounding approaches to system identification*. Plenum Press, 1996.

[78] M. Milanese and A. Vicino. Optimal estimation theory for dynamic system with set membership uncertainty: an overview. *Automatica*, 27:997–1009, 1991.

[79] B. Ninness and G. C. Goodwin. Estimation of model quality. *Automatica*, 31(12):1771–1797, 1995.

[80] A. Onatski and N. Williams. Modelling model uncertainty. *Journal of European Economic Association*, 1(5):1087–1122, Sept. 2003.

[81] R. Orsi, U. Helmke, and J. B. Moore. A newton-like method for solving rank constrained linear matrix inequalities. *Automatica*, 42(11):1875 – 1882, 2006.

[82] M. Overton and R. Womersley. Optimality conditions and duality theory for minimizing sums of the largest eigenvalues of symmetric matrices. *Mathematical Programming*, 62(1–3):321–357, 1993.

[83] B. Picinbono. *Random signals and systems*. Prentice Hall, Englewood Cliffs (NJ), 1993.

[84] D. Piga and R. Tóth. An SDP approach for ℓ_0-minimization: Application to ARX model segmentation. *Automatica*, 49(12):3646–3653, 2013.

[85] R. Pintelon and J. Schoukens. *System Identification: A Frequency Domain Approach*. Wiley, 2012.

[86] M. J. D. Powell. A method for nonlinear constraints in minimization problems. In R. Fletcher, editor, *Optimization*, pages 283–298. Academic Press, New York, 1969.

[87] C. Rao. *Moving Horizon Strategies for the Constrained Monitoring and Control of Nonlinear Discrete-Time Systems*. PhD thesis, University of Wisconsin-Madison, Madison, WI, 2000.

[88] C. Rao, J. Rawlings, and J. H. Lee. Constrained linear state estimation – a moving horizon approach. *Automatica*, 37:1619–1628, 2001.

[89] C. Rao, J. Rawlings, and D. Mayne. Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. *Automatic Control, IEEE Transactions on*, 48(2):246–258, 2003.

[90] J. Rawlings and B. Bakshi. Particle filtering and moving horizon estimation. *Computers & chemical engineering*, 30(10-12):1529–1541, 2006.

[91] J. Rawlings and D. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009.

[92] W. Reinelt. i4c: Identification for control package (version 1.1b5). Linköping University, Linköping, Sweden, Sept. 2000. http://www.wolfgang-reinelt.de/i4c/.

[93] W. Reinelt, A. Garulli, and L. Ljung. Comparing different approaches to model error modeling in robust identification. *Automatica*, 38(5):787 – 803, 2002.

[94] R. T. Rockafellar. Augmented lagrange multiplier functions and duality in nonconvex programming. *SIAM Journal on Control*, 12(2):268–285, 1974.

[95] S. Roweis. EM algorithms for PCA and SPCA. *Advances in neural information processing systems*, pages 626–632, 1998.

[96] T. B. Schön, A. Wills, and B. Ninness. System identification of nonlinear state-space models. *Automatica*, 47(1):39–49, Jan. 2011.

[97] G. Schwarz. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

[98] S. Shalev-Shwartz, A. Gonen, and O. Shamir. Large-scale convex minimization with a low-rank constraint. *arXiv preprint arXiv:1106.1622*, 2011.

[99] A. Shapiro and J. M. F. Ten Berge. Statistical inference of minimum rank factor analysis. *Psychometrika*, 67(1):79–94, 2002.

[100] R. H. Shumway and D. S. Stoffer. An approach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis*, 3(4):253–264, 1982.

[101] R. H. Shumway and D. S. Stoffer. *Time Series Analysis and Its Applications*. Springer-Verlag, 2000.

[102] T. Söderström and P. Stoica. *System Identification*. Prentice-Hall International, 1989.

[103] J. H. Stock and M. W. Watson. Forecasting using principal components from a large number of predictors. *Journal of the American Statistical Association*, 97:1167–1179, 2002.

[104] J. M. ten Berge and H. A. Kiers. A numerical approach to approximate and exact minimum rank of a covariance matrix. *Psychometrika*, 56(2):309–315, 1991.

[105] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[106] F. Tjärnström and L. Ljung. Using the Bootstrap to estimate the variance in case of undermodelling. *IEEE Transactions on Automatic Control*, 47:395–398, 2002.

[107] J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. *Information Theory, IEEE Transactions on*, 50(10):2231–2242, 2004.

[108] S. Ungarala. Computing arrival cost parameters in moving horizon estimation using sampling based filters. *Journal of Process Control*, 19(9):1576 – 1588, 2009.

[109] B. Vandereycken. *Riemannian and multilevel optimization for rank-constrained matrix problems*. PhD thesis, Department of Computer Science, KU Leuven, 2010.

[110] S. R. Venkatesh and M. A. Dahleh. On system identification of complex systems from finite data. *IEEE Transactions on Automatic Control*, 46:235–257, 2001.

[111] S. Verdu and H. V. Poor. Abstract dynamic programming models under commutativity conditions. *SIAM J. Control Optim.*, 25:990, 1006 1987.

[112] M. Vidyasagar and R. L. Karandikar. A learning theory approach to system identification and stochastic adaptive control. *Journal of Process Control*, 18(3–4):421 – 430, 2008.

[113] E. Weyer and M. Campi. Non-asymptotic confidence ellipsoids for the least-squares estimate. *Automatica*, 38(9):1539 – 1547, 2002.

[114] C. F. J. Wu. On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.

[115] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

[116] J. Yuz, J. Alfaro, J. Agüero, and G. Goodwin. Identification of continuous-time state space models from nonuniform fast-sampled data. *IET Control Theory and Applications*, 5(7):842–855, 2011.

[117] V. Zavala. Stability analysis of an approximate scheme for moving horizon estimation. *Computers & Chemical Engineering*, 34(10):1662–1670, 2010.

[118] J.-H. Zhao, L. Philip, and Q. Jiang. ML estimation for factor analysis: EM or non-EM? *Statistics and computing*, 18(2):109–123, 2008.

[119] K. Zhou, J. Doyle, and K. Glover. *Robust and optimal control*. Prentice Hall, 1996.