

# **Convex Relaxations for Robust Identification of Hybrid Models**

A Dissertation Presented

by

**Necmiye Ozay**

to

The Department of Electrical and Computer Engineering

in partial fulfillment of the requirements  
for the degree of

**Doctor of Philosophy**

**in**

**Electrical Engineering**

Northeastern University

Boston, Massachusetts

July 2010

The dissertation of Necmiye Ozay was read and approved\* by the following:

Mario Sznaier

Professor of Electrical and Computer Engineering, Northeastern University

Dissertation Adviser

Chair of Committee

Dana H. Brooks

Professor of Electrical and Computer Engineering, Northeastern University

Dissertation Reader

Octavia I. Camps

Professor of Electrical and Computer Engineering, Northeastern University

Dissertation Reader

Constantino M. Lagoa

Professor of Electrical Engineering, The Pennsylvania State University

Dissertation Reader

Gilead Tadmor

Professor of Electrical and Computer Engineering, and Mathematics, Northeastern University

Dissertation Reader

Ali Abur

Professor of Electrical and Computer Engineering, Northeastern University

Department Head

\* Signatures are on file in the Graduate School.

# *Abstract*

Northeastern University

Department of Electrical and Computer Engineering

Doctor of Philosophy in Electrical Engineering

## **Convex Relaxations for Robust Identification of Hybrid Models**

by Necmiye Ozay

In order to extract useful information from a data set, it is necessary to understand the underlying model. This dissertation addresses two of the main challenges in identification of such models. First, the data is often generated by multiple, unknown number of sources; that is, the underlying model is usually a mixture model or hybrid model. This requires solving the identification and data association problems simultaneously. Second, the data is usually corrupted by noise necessitating robust identification schemes. Motivated by these challenges we consider the problem of robust identification of hybrid models that interpolate the data within a given noise bound. In particular, for static data we try to fit affine subspace arrangements or more generally a mixture of algebraic surfaces to the data; and for dynamic data we try to infer the underlying switched affine dynamical system. Clearly, as stated, these problems admit infinitely many solutions. For instance, one can always find a trivial hybrid model with as many submodels/subspaces as the number of data points (i.e. one submodel/subspace per data point). In order to regularize the problem, we define suitable a priori model sets and objective functions that seek "simple" models. Although this leads to generically NP-Hard problems, we develop computationally efficient algorithms based on convex relaxations.

Additionally, we discuss a related problem: robust model (in)validation for affine hybrid systems. Before a given system description, obtained either from first principles or an identification step, can be used to design controllers, it must be validated using additional experimental data. In this dissertation, we show that the invalidation problem for switched affine systems can be solved within a similar framework by exploiting a combination of elements from convex analysis and the classical theory of moments.

Finally, the effectiveness of the proposed methods are illustrated using both simulations and several non-trivial applications in computer vision such as video and dynamic texture segmentation, two-view motion segmentation and human activity analysis. In all cases the proposed methods significantly outperform existing approaches both in terms of accuracy and resilience to noise.

## *Acknowledgements*

First and foremost, I would like to thank my adviser, Professor Mario Sznaier for his support in all aspects of my graduate life. His guidance, encouragement and enthusiasm made this journey quite fun for me. I am especially thankful to him for giving me the independence to pursue my own research ideas but also being there with “crazy” ideas whenever I got stuck. His being a great teacher, technical expertise, attention to mathematical rigor and broad vision have been and will be a constant source of inspiration for me.

I would like to thank Professor Octavia Camps who introduced me to the fields of computer vision and pattern recognition. Her expertise and suggestions have been very useful in finding interesting computer vision applications for hybrid system identification algorithms developed in this dissertation. I would also like to express my gratitude to Professor Constantino Lagoa of Penn State. My research has benefited a lot from discussions and interactions with him.

I am very grateful to Professor Dana Brooks and Professor Gilead Tadmor for serving on my dissertation committee and for their insightful comments on my work. I would also like to thank Professor Brooks for letting me sit in his group meetings.

I feel very fortunate to meet with great friends during my graduate studies. I had a very pleasant start to my studies in the U.S. thanks to two special friends, Roberto Lubliner and Dimitris Zarpalas who have been a great source of support and encouragement. I would also like to thank my lab mates at the Robust Systems Laboratory at Northeastern for their help and support, and for all the fun times we spent together chatting about this and that (and sometimes about research). In particular, Mustafa Ayazoglu helped generating some of the simulation data and figures in this dissertation (he can even print videos on an eps file). My thanks also go to the CDSP gang and Ms. Joan Pratt for their friendship, the pizza parties and the wonderful times with the dragonboat team (go go go Huskies!!). I am also grateful to Sila Kurugol not only for being a very close friend but also for our collaboration on medical image segmentation.

I want to take this opportunity to thank my professors and teachers back in Turkey for providing me with a solid education and background. I would also like to thank my closest friends, particularly Bahar, Dilek and Gugu, for being next to me whenever I need, despite the physical distance.

Finally, my greatest appreciation goes to my family, especially to my parents Kafiye and Hami for their endless love and support, and for being on the other end of the webcam each and every day.

# Contents

|  |            |
|--|------------|
| <b>Abstract</b>  | <b>ii</b>  |
| <b>Acknowledgements</b>  | <b>iii</b> |
| <b>List of Figures</b>   | <b>vii</b> |
| <b>List of Tables</b>  | <b>ix</b>  |
| <b>Abbreviations</b>   | <b>x</b>   |
| <b>Symbols</b>   | <b>xi</b>  |
| <br>   |            |
| <b>1 Introduction</b>  | <b>1</b>   |
| 1.1 Contributions and Outline . . . . .                          | 3          |
| <br>   |            |
| <b>2 Convex Relaxations</b>                                      | <b>5</b>   |
| 2.1 Background Results on Sparsification . . . . .               | 6          |
| 2.1.1 Sparse Signal Recovery . . . . .                           | 6          |
| 2.1.2 Matrix Rank Minimization . . . . .                         | 8          |
| 2.2 The Problem of Moments and Polynomial Optimization . . . . . | 10         |
| 2.2.1 The Problem of Moments . . . . .                           | 10         |
| 2.2.1.1 One Dimensional Case . . . . .                           | 10         |
| 2.2.1.2 Multi-Dimensional Case . . . . .                         | 11         |
| 2.2.2 Polynomial Optimization via Moments . . . . .              | 14         |
| 2.2.2.1 Exploiting the Sparse Structure . . . . .                | 15         |
| <br>   |            |
| <b>3 Identification of a Class of Hybrid Dynamical Systems</b>   | <b>17</b>  |
| 3.1 Introduction and Motivation . . . . .                        | 17         |
| 3.2 Definitions . . . . .  | 19         |
| 3.3 Problem Statement . . . . .                                  | 20         |

|          |  |           |
|----------|--|-----------|
| 3.4      | A Sparsification Approach  | 21        |
| 3.4.1    | Main Results   | 21        |
| 3.4.1.1  | Identification with Minimum Number of Switches                           | 22        |
|          | A Greedy Algorithm for the $\ell_\infty$ Case:                           | 22        |
|          | Identifiability of the Switches and Convergence of the Greedy Algorithm: | 23        |
|          | The Case of General Convex Noise Descriptions:                           | 26        |
|          | Extension to Multi-input Multi-output Models:                            | 27        |
|          | Extension to Multidimensional Models:                                    | 28        |
| 3.4.1.2  | Identification with Minimum Number of Submodels:                         | 29        |
| 3.4.2    | Examples   | 31        |
| 3.4.3    | Applications: Segmentation of Video Sequences.                           | 40        |
|          | Video-Shot Segmentation:   | 41        |
|          | Dynamic Textures:  | 42        |
| 3.5      | A Moments-Based Convex Optimization Approach                             | 43        |
| 3.5.1    | Main Results   | 46        |
| 3.5.1.1  | A Moments Based Convex Relaxation:                                       | 46        |
| 3.5.2    | Illustrative Examples  | 50        |
| 3.5.2.1  | Academic Example:  | 50        |
| 3.5.2.2  | A Practical Example: Human Activity Segmentation:                        | 51        |
| <b>4</b> | <b>Model (In)validation for a Class of Hybrid Dynamical Systems</b>      | <b>56</b> |
| 4.1      | Introduction and Motivation  | 56        |
| 4.2      | (In)validating MIMO SARX Models  | 57        |
| 4.2.1    | Problem Statement  | 58        |
| 4.2.2    | A Convex Certificate for (In)validating MIMO SARX Models                 | 58        |
| 4.3      | Numerical Considerations   | 62        |
| 4.4      | Illustrative Examples  | 64        |
| 4.4.1    | Academic Examples  | 64        |
| 4.4.2    | A Practical Example: Activity Monitoring                                 | 66        |
| 4.5      | Conclusions  | 67        |
| <b>5</b> | <b>Clustering Data into Multiple Unknown Subspaces</b>                   | <b>69</b> |
| 5.1      | Sequential Sparsification for Change Detection                           | 70        |
| 5.1.1    | Introduction and Motivation  | 70        |
| 5.1.2    | Segmentation via Sparsification  | 71        |
| 5.1.3    | Applications   | 73        |
| 5.1.3.1  | Video Segmentation:  | 73        |
| 5.1.3.2  | Segmentation of Dynamic Textures:  | 74        |
| 5.1.4    | Experiments  | 74        |
| 5.1.4.1  | Video Segmentation:  | 74        |
| 5.1.4.2  | Temporal Segmentation of Dynamic Textures:                               | 76        |
| 5.2      | GPCA with Denoising: A Moments-Based Convex Approach                     | 80        |

---

|          |   |            |
|----------|---|------------|
| 5.2.1    | Introduction and Motivation . . . . .                 | 80         |
| 5.2.2    | Problem Statement . . . . .                           | 82         |
| 5.2.3    | Main Results . . . . .                                | 83         |
| 5.2.3.1  | A Convex Relaxation . . . . .                         | 85         |
| 5.2.3.2  | Extension to Quadratic Surfaces . . . . .             | 87         |
| 5.2.3.3  | Handling Outliers . . . . .                           | 88         |
| 5.2.4    | Experiments . . . . .                                 | 89         |
| 5.2.4.1  | Synthetic Data . . . . .                              | 89         |
| 5.2.4.2  | 2-D motion Estimation and Segmentation . . . . .      | 91         |
| 5.2.4.3  | Two View Perspective Motion Segmentation . . . . .    | 93         |
| <b>6</b> | <b>Conclusions and Future Work</b>                    | <b>96</b>  |
| <b>A</b> | <b>Sparsity Related Proofs</b>                        | <b>98</b>  |
| A.1      | Proof of Lemma 1 . . . . .                            | 98         |
| A.2      | Proof of Lemma 2 . . . . .                            | 99         |
| <b>B</b> | <b>Recovering the Parameters of the Model in GPCA</b> | <b>102</b> |
|          | <b>Bibliography</b>                                   | <b>104</b> |
|          | <b>Vita</b>   | <b>112</b> |
|          | <b>List of Publications</b>                           | <b>113</b> |

# List of Figures

|      |  |    |
|------|--|----|
| 3.1  | True and estimated parameter sequences for parameter $a_1(\sigma_t)$ for Example 3. . . . .  | 34 |
| 3.2  | Each histogram shows the frequency of estimated number of submodels for different noise levels. (a) $\epsilon = 0.05$ , (b) $\epsilon = 0.1$ , (c) $\epsilon = 0.25$ , (d) $\epsilon = 0.5$ . The true number of submodels is $s = 3$ . . . . .  | 35 |
| 3.3  | Median of parameter estimation error $\Delta_n$ versus noise level $\epsilon$ . Error bars indicate the median absolute deviation. . . . .   | 38 |
| 3.4  | Results for detecting switches in a 2-D system. Left: Original segmentation (i.e. $\sigma_{i,j}$ ). Middle: Output of the system in (3.28). Difference in frequency content of output values corresponding to the two different submodels can be inferred from the texture. Right: Resulting segmentation. . . . .     | 39 |
| 3.5  | Results for detecting switches (i.e. estimating $\hat{\sigma}_{i,j}$ ) in a texture image. Left: Original image. Middle: GPCA segmentation. Right: Segmentation via proposed method. . . .   | 40 |
| 3.6  | Video Segmentation Results. Left Column: Ground truth segmentation (jumps correspond to cuts and slanted lines correspond to gradual transitions). Right Column: Changes detected with different methods. Value 0 corresponds to frames within a segment and value 1 corresponds to the frames in transitions. . . . . | 42 |
| 3.7  | Sample dynamic texture patches. Top: smoke, Bottom: river . . . . .  | 43 |
| 3.8  | Results for detecting change in dynamics only. Top: Smoke sequence concatenated with transposed dynamics. Bottom: River sequence concatenated with reversed dynamics. . . . .  | 44 |
| 3.9  | Clustering via GPCA. . . . .   | 51 |
| 3.10 | Clustering via moments-based method. . . . .   | 52 |
| 3.11 | Absolute error for GPCA identification. . . . .  | 53 |
| 3.12 | Absolute error for moments-based identification. . . . .   | 54 |
| 3.13 | Sample frames from the video. . . . .  | 54 |
| 3.14 | Activity segmentation via GPCA. . . . .  | 55 |
| 3.15 | Activity segmentation via moments-based method. . . . .  | 55 |
| 4.1  | Problem Setup. The coefficient matrices of the submodels $G_i$ and a bound on the noise are known <i>a priori</i> . The experimental data consists of input/output measurements, $\mathbf{u}$ and $\tilde{\mathbf{y}}$ . The mode signal $\sigma_t$ and noise sequence $\eta$ are unknown. . . . .                     | 59 |
| 4.2  | Training sequence used in identification of the submodel ( $A_1$ ) for walking. . . . .  | 67 |



|      |  |    |
|------|--|----|
| 4.3  | Top row: Walk, wait, walk sequence (not invalidated). Second row: Running sequence (invalidated). Third row: Walk, jump sequence (invalidated). Last row: Jumping sequence (invalidated). . . . .  | 68 |
| 5.1  | Video Segmentation Results. First and third row: Ground truth segmentation. Second and last row: Changes detected with different methods. Value 0 corresponds to frames within a segment and value 1 corresponds to the frames in transitions. . . . .   | 77 |
| 5.2  | Sample dynamic texture patches. From left to right: water, flame, steam. . . . .   | 78 |
| 5.3  | Results for detecting change in dynamics only. Top: Smoke sequence concatenated with transposed dynamics. Bottom: River sequence concatenated with reversed dynamics. . . . .  | 79 |
| 5.4  | An image pair with 3 relocated objects with noisy correspondences superimposed. . . . .  | 80 |
| 5.5  | Sample segmentation results and mean misclassification rates. (a): GPCA segmentation (mean misclassification rate: 31.1%). (b): RGPCA segmentation (mean misclassification rate 14.9%). (c): Proposed algorithm (mean misclassification rate 3.7%). The image size is $600 \times 800$ and the noise level is $\pm 10$ pixels. . . . . | 81 |
| 5.6  | Example using synthetic data laying on two planes in $\mathbb{R}^3$ . Here the red stars and blue plus signs indicate the original (noisy) and the denoised data points, respectively. . . . .   | 91 |
| 5.7  | Fitting errors for GPCA, RGPCA and moments-based methods for the example in Fig. 5.6. . . . .  | 92 |
| 5.8  | Example using synthetic data lying on two lines (blue and green) and a plane (transparent red) in $\mathbb{R}^3$ . Here the red stars and blue plus signs indicate the original (noisy) and the denoised data points, respectively. . . . .  | 93 |
| 5.9  | Example using synthetic data lying on two planes in $\mathbb{R}^3$ . Here the red stars and blue plus signs indicate the original (noisy) and the denoised data points, respectively. Cyan circles indicate the points identified as outliers. . . . .   | 94 |
| 5.10 | Two perspective images of points on teapot surfaces. . . . .   | 95 |
| 5.11 | (a)-(b) First and second images with moments-based segmentation superimposed, 10 misclassified points. (c) Segmentation with the method in [1], 29 misclassified points. . . . .   | 95 |
| 5.12 | (a)-(b) First and second images with moments-based segmentation superimposed ( 9.06% misclassification rate). (c) Segmentation with the method in [1] (42.90% misclassification rate). . . . .   | 95 |

# List of Tables

|     |  |    |
|-----|--|----|
| 3.1 | Optimal Greedy Algorithm for Problem 2 . . . . .   | 22 |
| 3.2 | Algorithm for Problem 3 . . . . .  | 30 |
| 3.3 | Minimum number of submodel estimation error statistics for different noise levels. The results for sparsification is given both with formulation (3.23) and with formulation (3.24) in parantheses. . . . .  | 36 |
| 3.4 | Normalized parameter identification error statistics of minimum number of submodels problem with different noise level. The results for sparsification is given both with formulation (3.23) and with formulation (3.24) in parantheses. . . . .   | 37 |
| 3.5 | Rand Indices that show the quality of mode signal estimates. . . . .   | 37 |
| 3.6 | Error measure $\Delta_n$ that shows the quality of parameter estimates. . . . .  | 38 |
| 3.7 | Rand indices for video-shot segmentation . . . . .   | 41 |
| 3.8 | Estimated and true values of parameters . . . . .  | 51 |
| 4.1 | Invalidation results for example 2. The values of $\tilde{p}$ were respectively $-3.8441e - 008$ , $-8.2932e - 009$ , $0.8585$ , $-5.4026e - 008$ , $-1.5490e - 007$ and $0.7566$ . . . . .  | 65 |
| 4.2 | Invalidation results for example 3. The values of $\tilde{p}$ were respectively $0.0724$ , $0.0035$ , $-5.1810e - 007$ , $0.0737$ , $0.0034$ and $-1.4930e - 007$ . . . . .  | 65 |
| 4.3 | Invalidation results for example 4. The values of $\tilde{p}$ were ,respectively, $0.8963$ , $0.0997$ , $0.0080$ , $0.0308$ , $2.8638e - 004$ , $2.9069e - 006$ and $6.2061e - 006$ . . . . .  | 65 |
| 4.4 | Invalidation results for activity monitoring. The values of $\tilde{p}$ were, respectively, $-2.3303e - 008$ , $2.3707e - 005$ , $5.0293e - 007$ , and $1.5998e - 006$ . . . . .   | 67 |
| 5.1 | Rand indices . . . . .   | 76 |
| 5.2 | Results on Dynamic Texture Database . . . . .  | 78 |
| 5.3 | Synthetic Data Results. $D$ and $d_k$ denote the dimension of the ambient space and subspaces, respectively. $N$ shows the number of samples per subspace. $\epsilon$ denotes the true noise level. The last three columns show the mean and median (in parenthesis) fitting errors. . . . . | 90 |
| 5.4 | Misclassification rates for perspective motion segmentation examples. . . . .  | 94 |

# Abbreviations

|              |   |
|--------------|---|
| <b>ARX</b>   | Auto- <b>R</b> egressive Exogenous  |
| <b>SARX</b>  | Switched <b>A</b> ffine Auto- <b>R</b> egressive Exogenous                      |
| <b>LCCDE</b> | Linear <b>C</b> onstant <b>C</b> oefficient <b>D</b> ifference <b>E</b> quation |
| <b>ASCDE</b> | Affine <b>S</b> witched <b>C</b> oefficient <b>D</b> ifference <b>E</b> quation |
| <b>SISO</b>  | Single- <b>I</b> nput Single- <b>O</b> utput                                    |
| <b>MIMO</b>  | Multi- <b>I</b> nput Multi- <b>O</b> utput                                      |
| <b>LMI</b>   | Linear <b>M</b> atrix <b>I</b> nequality  |
| <b>SVD</b>   | Singular <b>V</b> alue <b>D</b> ecomposition                                    |
| <b>PCA</b>   | Principal <b>C</b> omponent <b>A</b> nalysis                                    |
| <b>GPCA</b>  | Generalized <b>P</b> incipal <b>C</b> omponent <b>A</b> nalysis                 |
| <b>RGPCA</b> | Robust <b>G</b> eneralized <b>P</b> incipal <b>C</b> omponent <b>A</b> nalysis  |
| <b>HQSA</b>  | Hybrid <b>Q</b> uadratic <b>S</b> urface <b>A</b> nalysis                       |
| <b>LP</b>    | Linear <b>P</b> rogramming  |
| <b>SDP</b>   | Semi- <b>d</b> efinite <b>P</b> rogramming                                      |
| <b>NP</b>    | Non-deterministic <b>P</b> olynomial-time                                       |
| <b>SOS</b>   | Sum of <b>S</b> quares  |

# Symbols

|   |   |
|---|---|
| $\mathbb{R}, \mathbb{Z}$                    | set of real numbers, integers   |
| $\mathbf{x}$                                | a vector in $\mathbb{R}^N$  |
| $\mathbf{M}$                                | a matrix in $\mathbb{R}^{n \times m}$   |
| $\ \mathbf{x}\ _p$                          | $p$ -norm in $\mathbb{R}^N$ , that is $\ \mathbf{x}\ _p \doteq \sqrt[p]{\sum_{i=1}^N x_i^p}$  |
| $\ \mathbf{x}\ _\infty$                     | $\infty$ -norm of the vector $\mathbf{x} \in \mathbb{R}^N$ , that is $\ \mathbf{x}\ _\infty \doteq \sup_i  x_i $  |
| $\{\mathbf{x}(t)\}_{t=1}^T, \{\mathbf{x}\}$ | a vector valued sequence of length $T$ where each $\mathbf{x}(t) \in \mathbb{R}^N$  |
| $\ \{\mathbf{x}\}\ _p$                      | $\ell_p$ norm of a vector valued sequence $\ \{\mathbf{x}\}\ _p \doteq \left(\sum_{i=1}^T \ \mathbf{x}(i)\ _p^p\right)^{1/p}$   |
| $\ \{\mathbf{x}\}\ _0$                      | $\ell_0$ -quasinorm $\doteq$ number of non-zero vectors in the sequence (i.e. cardinality of the set $\{t \in \mathbb{Z}   \mathbf{x}(t) \neq \mathbf{0}, t \in [1, T]\}$ ) |
| $\mathbf{I}$                                | identity matrix of appropriate dimension  |
| trace $\mathbf{M}$                          | trace of the matrix $\mathbf{M}$  |
| $\mathbf{M} \succeq \mathbf{N}$             | the matrix $\mathbf{M} - \mathbf{N}$ is positive semidefinite.  |
| $\mathbb{R}[x_1, \dots, x_n]$               | the ring of polynomials in $n$ variables over $\mathbb{R}$ . $\mathbb{R}[\mathbf{x}]$ may be used when $n$ is clear from the context.                                       |
| $\mathbb{N}_n$                              | positive integers up to $n$ , i.e. $\mathbb{N}_n \doteq \{1, \dots, n\}$  |
| $\wedge (\vee)$                             | logical AND (OR)  |
| $\langle \mathbf{M}, \mathbf{N} \rangle$    | $\text{trace}(\mathbf{M}^T \mathbf{N})$   |
| $\mathcal{P}_D^n$                           | set of $n^{\text{th}}$ degree multivariate polynomials in $D$ variables. $n$ and $D$ may be omitted when clear from the context.  |

*To my parents, Kafiye and Hami*

# Chapter 1

## Introduction

Hybrid models are abundant in a wide range of applications and processes. In dynamical systems community hybrid systems, systems characterized by the interaction of both continuous and discrete dynamics, have been the subject of considerable attention during the past decade. These systems arise naturally in many different contexts, e.g. biological systems ([2, 3], etc.), systems incorporating logical and continuous elements ([4, 5]), manufacturing ([6]), automotive ([7]), etc, and in addition, affine hybrid systems can be used to approximate nonlinear dynamics. Computer vision and pattern recognition communities have also extensively considered subspace arrangements ([8, 9, 10, 11]) and mixture models ([12, 13, 14]) in various segmentation/clustering applications such as affine motion segmentation, face clustering under varying illumination, image, video and dynamic texture segmentation.

In this dissertation, we consider the problem of parametric identification of hybrid models. In particular, for dynamic input/output data we try to infer the underlying switched affine dynamical system (i.e. a time-varying system that switches among a finite number of linear time invariant subsystems according to a discrete mode signal); and for static data we try to fit a union of affine subspaces (i.e. affine subspace arrangement) or more generally a mixture of algebraic surfaces to the data. The main challenge in both problems is that one needs to solve the identification and data association problems simultaneously. For switched affine dynamical systems, if we knew the mode signal (i.e. we knew which submodel was active at which time instants), then we can treat the problem as the identification of linear time-invariant systems. Conversely, if we knew the parameters of the submodels, then we could use model invalidation techniques to find which model is valid for each time instant. Similarly for static data, if we knew which data points came from the same subspace, then we could easily fit a subspace to those points. On the other hand if the subspaces were known, each data point could

be assigned to the subspace closest to it. Identification of hybrid models is a hard problem mainly due to this chicken-and-egg nature. Moreover, taking into account the fact that the data is usually corrupted by noise makes the problem even more challenging. We try to address these challenges with the proposed robust identification schemes.

Also, unless additional constraints are imposed, the identification problem admits infinitely many solutions. For instance, one can always find a trivial hybrid model with as many submodels/subspaces as the number of data points (i.e. one submodel/subspace per data point). In order to regularize the problem, we define suitable *a priori* model sets and objective functions that seek “simple” models. Hence, we recast the problem into an optimization form. Although this leads to generically non-convex NP-Hard problems, we develop computationally efficient algorithms based on convex relaxations. Convex optimization problems constitute a special class of mathematical optimization problems for which there are efficient polynomial time algorithms that are guaranteed to converge<sup>1</sup> to a global optimum [15]. Convex relaxations can be used to approximate (in some cases, even exactly solve for) the global optimum of a non-convex problem in a principled way. With this in mind, we try to convexify our problems and make use of the powerful tools developed for convex optimization.

We also consider a related problem: robust model (in)validation for affine hybrid systems. Before a given system description, obtained either from first principles or an identification step, can be used to design controllers, it must be validated using additional experimental data. Considering the inherent difficulties in hybrid system identification, an invalidation step becomes even more crucial for such systems. We show that the invalidation problem for switched affine systems can be solved within a similar framework by exploiting a combination of elements from convex analysis and the classical theory of moments.

In addition to developing hybrid identification and (in)validation schemes, we show that many interesting problems in computer vision can be robustly and efficiently solved within a hybrid identification/(in)validation framework. In particular, we apply the proposed methods in video and dynamic/static texture segmentation, two-view motion segmentation and human activity analysis. In all cases the proposed methods significantly outperform existing approaches both in terms of accuracy and resilience to noise.

---

<sup>1</sup>More precisely, one can get a solution arbitrarily close to the global optimum, in polynomial time.

## 1.1 Contributions and Outline

The main contributions of this dissertation are:

- Reformulation of the problems of identification of switched ARX systems with minimum number of switches and minimum number of submodels as sparsification problems, making a connection between sparse signal recovery and hybrid system identification. This reformulation enables efficient convex optimization based solution methods.
- An exact switch detection algorithm for SARX systems with  $\ell_\infty$  bounded noise. A notion of switch identifiability is presented together with necessary and sufficient conditions for identifiability of switches purely from input/output data.
- Derivation of the convex envelop for the function that counts the nonzeros vectors in a vector valued sequences which lead to a convex relaxation for sparsity problems involving vector valued sequences.
- Reformulation of identification of switched ARX systems with known number of submodels and  $\ell_\infty$  bounded noise as a matrix rank minimization problem where all the matrices involved are affine in decision variables. This problem can be solved by using efficient convex relaxations for rank minimization.
- An extension of moment-based polynomial optimization method for the case where the function to be minimized is not a polynomial function (i.e. rank). In hybrid system identification problem considered, keeping the rank objective instead of a more complicated equivalent polynomial objective function, it is possible to better utilize the problem structure to significantly reduce the computational complexity.
- Convex certificates for robust (in)validation of multi-input multi-output SARX models and recasting activity monitoring problem as a hybrid model invalidation problem.
- Application of hybrid system identification to several non-trivial problems in computer vision establishing the fact that hybrid dynamical models provide a compact representation for complex data streams that are generated by multiple sources or for which the underlying process varies with time. Also in all cases, the proposed approaches are shown to outperform the state-of-the-art techniques currently used in the computer vision field.



- A sparsification based method, which takes into account the order dependency, for clustering noise corrupted data that lie on a mixture of subspaces and its applications in video and dynamic texture segmentation.
- A moments-based convex method for segmentation of multiple algebraic surfaces from noisy sample data points. This method is applied to the problems of simultaneous 2-D two view motion segmentation and motion segmentation from two perspective views. Simulations and real examples illustrate that our formulation substantially reduces the noise sensitivity of existing approaches.

The rest of the dissertation is organized as follows. We start with a summary of convex relaxations used through out the dissertation in Chapter 2. The proposed methods are discussed in two main parts. First we discuss the dynamic case and then we present parallel results on the static case. Specifically, in Chapter 3, we consider the problem of identification of switched affine dynamical systems. A general introduction to this problem is presented in Section 3.1. In Sections 3.2 and 3.3, some relevant definitions are given and the problem is formally stated, respectively. We propose two different approaches for this problem. The sparsification-based approach is introduced in Section 3.4; and the moments-based approach is introduced in Section 3.5. The problem of model (in)validation for switched affine dynamical systems together with the proposed solution is presented in Chapter 4. Chapter 5 addresses the problem of segmentation of affine subspace arrangements or more generally mixture of algebraic surfaces. In Section 5.1, we present the sparsification method for clustering data into an unknown number of affine subspaces where the sequential nature of the data is relevant. The motivation for this problem is given in Section 5.1.1 and main results are presented in Section 5.1.2. In Section 5.2, we present a polynomial kernel denoising method and discuss its application in segmentation of noise corrupted data lying on a mixture of algebraic surfaces. Each chapter includes an example section illustrating the applications of the proposed methods and their advantages against existing algorithms. Finally, Chapter 6 concludes the dissertation with some remarks and directions for future research.

## Chapter 2

# Convex Relaxations

*Convex optimization* is a special class of mathematical optimization problems for which there are efficient polynomial time algorithms that are guaranteed to converge to a global minimum [15]. Least squares, linear programming, semidefinite programming, second order cone programming are all convex optimization problems. If a problem can be formulated as a convex problem, it can reliably and efficiently solved, for instance, using interior point methods. There are free optimization packages (i.e. solvers) such as SEDUMI [16] and SDPT3 [17] for solving convex programs. Moreover, user-friendly modeling languages such as CVX [18] and YALMIP [19] make these solvers accessible for a wide range of users.

On the contrary to convex problems, non-convex problems usually have lots of local minima where solution algorithms are likely to get trapped if not initialized properly. Convex relaxations can be used to approximate (in some cases, even exactly solve for) the global optimum of a non-convex problem in a principled way. In some cases, they can also be used to find bounds on the global optimum or to provide a good initial point for gradient search.

In this dissertation, we recast identification and invalidation problems into appropriate optimization problems. Unfortunately, these problems are usually non-convex and NP-Hard. Nevertheless, resorting to convex relaxations, we obtain efficient solutions. In what follows some important convex relaxations, which play a central role in deriving our main results, are summarized. It is important to note that this chapter is by no means a comprehensive treatment of the subject. Rather, it aims at giving the basic intuition behind each relaxation together with readily usable, relaxed, convex formulations that will help the reader understand the development in the proceeding chapters. We refer to several

relevant publications where one can find an in-depth treatment of each relaxation. More details are provided for the cases where modifying the existing theory was necessary to make it suitable for our problem.

## 2.1 Background Results on Sparsification

### 2.1.1 Sparse Signal Recovery

In this section, we present the background results on the problem of *sparse signal recovery* [20, 21, 22] that motivate the approach pursued in sections 3.4 and 5.1. We also provide some necessary modifications to the standard sparsification results so that they are applicable to the problems considered in this dissertation.

*Sparse signal recovery* problem can be stated as: given some linear measurements  $\mathbf{y} = A\mathbf{x}$  of a discrete signal  $\mathbf{x} \in \mathbb{R}^n$  where  $A \in \mathbb{R}^{m \times n}$ ,  $m \ll n$ , find the sparsest signal  $\mathbf{x}^*$  consistent with the measurements. In terms of the  $\ell_0$  quasinnorm (i.e.  $\|\cdot\|_0$  satisfies all of the norm axioms except homogeneity since  $\|c\mathbf{x}\|_0 = \|\mathbf{x}\|_0$  for all non-zero scalars  $c$ ), this problem can be recast into the following optimization form:

$$\min \|\mathbf{x}\|_0 \text{ subject to } : \mathbf{y} = A\mathbf{x} \quad (2.1)$$

It is well known that the problem above is at least generically NP-complete ([23, 24]). Two fundamental questions in sparse signal recovery are: (i) the uniqueness of the sparse solution, (ii) existence of efficient algorithms for finding such a signal. In the past few years it has been shown that if the matrix  $A$  satisfies the so-called *restricted isometry property* (RIP), the solution is unique and can be recovered efficiently by several algorithms. These algorithms fall into two main categories: greedy algorithms (e.g. orthogonal matching pursuit [25, 26, 27]) and  $\ell_1$ -based convex relaxation (also known as basis pursuit [20, 21, 22]). In this thesis we follow the latter approach which is based on replacing  $\|\mathbf{x}\|_0$  in the optimization above by  $\|\mathbf{x}\|_1$ . The idea behind this relaxation is the fact that the  $\ell_1$  norm is the *convex envelope* of the  $\ell_0$  norm, and thus, in a sense, minimizing the former yields the best convex relaxation to the (non-convex) problem of minimizing the latter. Moreover, as shown in [20, 21, 22], this relaxation is stable and robust to noise. That is, even when only noisy linear measurements are available, if RIP holds for  $A$ , which is true with high probability for random matrices, recovery of the correct support of the original signal and approximating the true value within a factor of the noise are

possible by solving:

$$\min \|x\|_1 \text{ subject to : } \|\mathbf{y} - A\mathbf{x}\| \leq \epsilon \quad (2.2)$$

where  $\epsilon$  is a bound on the norm of noise. This formulation arises naturally in many engineering applications such as magnetic resonance imaging, radar signal processing and image processing. Moreover, existence of efficient algorithms to solve this problem led to the compressed sensing framework which enabled speeding up signal acquisition considerably since the original sparse signal can be reconstructed using relatively few measurements. We refer the interested reader to the recent survey paper [28] for a comprehensive treatment of the subject.

In hybrid system identification, we will pursue a similar approach. However, we will work with sparsification problems in the space of vector valued finite<sup>1</sup> sequences

$$\mathcal{S} = \left\{ \{\mathbf{g}(t)\}_{t=t_0}^T \mid \mathbf{g}(t) \in \mathbb{R}^d \right\}$$

rather than with vectors  $\mathbf{x} \in \mathbb{R}^N$ . This change requires extending the theory behind the  $\ell_1$ -norm relaxation to the space  $\mathcal{S}$ . To this effect, begin by noting that the number of non-zero elements (i.e. vectors) in  $\{\mathbf{g}\} \in \mathcal{S}$  (i.e.  $\|\{\mathbf{g}\}\|_0$ ) is the same as in  $\|\bar{\mathbf{g}}\|_0$  where  $\bar{\mathbf{g}} = [\|\mathbf{g}(t_0)\|, \dots, \|\mathbf{g}(T)\|]^T \in \mathbb{R}^{T-t_0+1}$ . This suggests the use of  $\|\bar{\mathbf{g}}\|_1 = \sum_t \|\mathbf{g}(t)\|$  as a convex objective function with an appropriate choice of the norm  $\|\mathbf{g}(t)\|$ . In particular, we will use  $\|\mathbf{g}(t)\|_\infty$ . The theoretical support for this intuitive choice is provided next.

*Lemma 1.* The convex envelope of the  $\ell_0$ -norm of a vector valued sequence on  $\|\{\mathbf{g}\}\|_\infty \leq 1$  is given by

$$\|\{\mathbf{g}\}\|_{0,env} \triangleq \sum_t \|\mathbf{g}(t)\|_\infty. \quad (2.3)$$

*Proof.* Given in the Appendix A.1. □

A related line of results recently appeared in compressed sensing/sparse signal recovery community for structured sparsity (see for instance [29, 30, 31]).

Next, we provide a sufficient condition for exactness of the convex envelope based relaxation for the vector valued case. The original sparsity problem we would like to solve has the following form:

---

<sup>1</sup>Since the experimental data consists of only finite samples, we consider finite sequences. However, with appropriate modifications the discussions in this section can easily be extended to deal with infinite sequences.

$$\begin{aligned} \min_{\mathbf{g}(t)} \quad & \|\{\mathbf{g}(t)\}\|_0 \\ \text{s.t} \quad & \mathbf{A}\tilde{\mathbf{g}} = \mathbf{y} \end{aligned} \quad (2.4)$$

where  $\{\mathbf{g}(t)\}_{t=1}^T$  is a vector valued sequence with  $\mathbf{g}(t) \in R^d$ .  $\tilde{\mathbf{g}} \in R^{dT}$  is a vector formed by stacking the elements of  $\{\mathbf{g}(t)\}_{t=1}^T$ . Assume the minimum number of switches is  $k$  and the above problem (2.4) has a unique optimizer  $\{\mathbf{g}_o(t)\}_{t=1}^T$  with  $k$  non-zero elements. Next, consider the convex envelope based relaxation:

$$\begin{aligned} \min_{\mathbf{g}(t)} \quad & \sum_{t=1}^T \|\mathbf{g}(t)\|_\infty \\ \text{s.t} \quad & \mathbf{A}\tilde{\mathbf{g}} = \mathbf{y} \end{aligned} \quad (2.5)$$

Now, define the constants  $c_1, c_2$  such that

$$c_1 \|\tilde{\mathbf{g}}\|_2^2 \leq \|\mathbf{A}\tilde{\mathbf{g}}\|_2^2 \leq c_2 \|\tilde{\mathbf{g}}\|_2^2 \quad (2.6)$$

where  $c_1$  is the maximum and  $c_2$  is the minimum of all scalars such that these inequalities hold for all  $2k$ -sparse  $\{\mathbf{g}(t)\}_{t=1}^T$ .

*Lemma 2.* Let  $\tilde{\mathbf{g}}_o$  and  $\tilde{\mathbf{g}}_1$  be the arguments of the optimal solutions of problems (2.4) and (2.5), respectively. If for the matrix  $\mathbf{A}$ , the coefficients defined in (2.6) satisfy

$$\frac{c_1}{c_2} > \frac{T + dk^2}{T + dk^2 + 4k} \quad (2.7)$$

then, the problems (2.4) and (2.5) are equivalent (i.e.  $\tilde{\mathbf{g}}_o = \tilde{\mathbf{g}}_1$ ).

*Proof.* Given in the Appendix A.2. □

## 2.1.2 Matrix Rank Minimization

Through out this dissertation, we recast several problems into a matrix rank minimization form. This problem is closely related to sparsification in that  $\ell_0$ -norm minimization is a special case of matrix rank minimization where the matrix whose rank to be minimized is a diagonal matrix. Since rank minimization can be reduced to a sparsification problem which is NP-Hard, it follows that rank minimization as well is generically NP-Hard. However, as recalled next, efficient convex relaxations exist

when the constraint set is convex. Most of the material presented in this section is from [32, 33, 34]; and further details can be found therein.

The problem of rank minimization over convex sets can be formally stated as follows:

$$\begin{aligned} & \text{minimize} && \text{rank } \mathbf{X} \\ & \text{subject to} && \mathbf{X} \in \mathcal{C} \end{aligned} \quad (2.8)$$

where  $\mathbf{X} \in \mathbb{R}^{m \times n}$  is the optimization variable and  $\mathcal{C}$  is the convex constraint set.

A well-known convex relaxation for rank minimization is to replace rank objective with the *nuclear norm* objective. Nuclear norm is indeed the convex envelope of rank on the set of bounded matrices [32]; and is equal to sum of singular values. That is:

$$\|\mathbf{X}\|_{nuc} \doteq \sum_{i=1}^{\min(m,n)} \sigma_i(\mathbf{X}) \quad (2.9)$$

where  $\sigma_i(\mathbf{X})$  is the  $i^{\text{th}}$  largest singular value of  $\mathbf{X}$ .

*Remark 1.* A few interesting observations on nuclear norm are in order. First, as stated earlier, for diagonal matrices, rank minimization corresponds to  $\ell_0$ -norm minimization. Similarly, for this specific case, nuclear norm reduces to  $\ell_1$ -norm. Second, for positive semidefinite matrices, nuclear norm corresponds to the trace of the matrix. Hence, this relaxation is also known as *trace heuristic*.

By employing nuclear norm relaxation, (2.8) can be relaxed to the following convex program:

$$\begin{aligned} & \text{minimize} && \|\mathbf{X}\|_{nuc} \\ & \text{subject to} && \mathbf{X} \in \mathcal{C} \end{aligned} \quad (2.10)$$

Moreover, by exploiting the so called *semidefinite embedding lemma* ([33]), problem (2.10) can be converted to an equivalent semidefinite program as follows:

$$\begin{aligned} & \text{minimize} && \text{trace } \mathbf{Y} + \text{trace } \mathbf{Z} \\ & \text{subject to} && \begin{bmatrix} \mathbf{Y} & \mathbf{X} \\ \mathbf{X}^T & \mathbf{Z} \end{bmatrix} \succeq 0 \\ & && \mathbf{X} \in \mathcal{C} \end{aligned} \quad (2.11)$$

which can be solved using of-the-shelf semidefinite programming solvers.

A crucial property of this relaxation is that when the convex set  $\mathcal{C}$  is made up of a system of linear equalities (i.e.  $\mathcal{C} = \{\mathbf{X} | \mathcal{A}(\mathbf{X}) = \mathbf{b}\}$  where  $\mathcal{A}$  is a linear transformation), there exist some sufficient conditions for the exactness of the relaxation ([35]). In particular, as shown in [35], if the matrix representation of  $\mathcal{A}$  satisfies a version of RIP, which holds true with high probability for certain families of random matrices, the minimizers of (2.11) and (2.8) are the same.

## 2.2 The Problem of Moments and Polynomial Optimization

In this section, we recall some results from the classical theory of moments and their use in polynomial optimization. In particular, linear matrix inequality based characterization of moment sequences can be used to recast non-convex polynomial optimization problems as convex semidefinite programming problems (see the recent paper [36] for an excellent survey on this subject). These techniques play a key role in establishing the main results of sections 3.5, 4 and 5.2.

### 2.2.1 The Problem of Moments

#### 2.2.1.1 One Dimensional Case

Given a sequence of scalars  $\{m_i\}_{i=1}^n$ , the *problem of moments* is to determine whether there exist a probability measure in  $\mathbb{R}$  that has  $\{m_i\}$  as its first  $n$  moments (see references [37, 38, 39] for a historical review and details of the problem). In particular, in the section 3.5 we are interested in probability measures that are supported on bounded symmetric intervals of the real line. This problem is known as *truncated Hausdorff moment problem*. The following theorem provides necessary and sufficient conditions for the existence of such a measure.

*Theorem 1.* Given a sequence  $\{m_i : i = 1, 2, \dots, n\}$ , there exists a probability measure supported on  $[-\epsilon, \epsilon]$  such that

$$m_i = \mathbf{E}_\mu(x^i) = \int_{-\epsilon}^{\epsilon} x^i \mu(dx)$$

if and only if

- when  $n = 2k + 1$  (odd case), the following holds

$$\epsilon \mathbf{M}(0, 2k) \succeq \mathbf{M}(1, 2k + 1) \tag{2.12}$$

$$\mathbf{M}(1, 2k + 1) \succeq -\epsilon \mathbf{M}(0, 2k) \quad (2.13)$$

- when  $n = 2k$  (even case), the following holds

$$\mathbf{M}(0, 2k) \succeq 0 \quad (2.14)$$

$$\epsilon^2 \mathbf{M}(0, 2k - 2) \succeq \mathbf{M}(2, 2k) \quad (2.15)$$

where  $\mathbf{M}(i, i + 2j)$  is the  $(j + 1)$  by  $(j + 1)$  Hankel matrix formed from the moments, that is:

$$\mathbf{M}(i, i + 2j) \doteq \begin{bmatrix} m_i & m_{i+1} & \cdots & m_{i+j} \\ m_{i+1} & \ddots & \ddots & m_{i+j+1} \\ \vdots & \ddots & \ddots & \vdots \\ m_{i+j} & \cdots & \cdots & m_{i+2j} \end{bmatrix}, \quad (2.16)$$

and where  $m_0 = 1$ .

*Proof.* Direct application of Theorem III.2.3 and Theorem III.2.4 in [38].  $\square$

### 2.2.1.2 Multi-Dimensional Case

In this section, we consider probability measures supported in certain subsets of  $\mathbb{R}^D$ . There is an important difference in passing from one-dimensional distributions to multi-dimensional distributions. Note that in the one dimensional case, the truncated moment sequences can be exactly characterized with finite-sized linear matrix inequalities. However, in multi-dimensional case such a semidefinite characterization, in general, requires infinite LMIs. Nevertheless, it is possible to obtain arbitrarily good approximations by using a truncated version of the infinite LMIs.

Let  $\mathcal{K}$  be a closed subset of  $\mathbb{R}^D$  and let  $\alpha$  be a multi-index (i.e.  $\alpha \in \mathbb{N}^D$ ) representing the powers of a monomial in  $D$  variables. Given a sequence of scalars  $\{m_\alpha\}$ , the  $\mathcal{K}$ -moment problem is to determine whether there exists a probability measure  $\mu$  supported on  $\mathcal{K}$  such that it has each  $m_\alpha$  as its  $\alpha^{\text{th}}$  moment. That is:

$$m_\alpha = \mathbf{E}_\mu(\mathbf{x}^\alpha) = \int_{\mathcal{K}} \mathbf{x}^\alpha \mu(dx) \quad (2.17)$$

where  $\mathbf{x}^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_D^{\alpha_D}$ .



First, we are interested in the case where  $\mathcal{K}$  is the  $\epsilon$ -ball. The following theorem taken from [40] provides necessary and sufficient conditions for the existence of a probability measure that is supported on a ball of radius  $\epsilon$  centered at the origin.

*Theorem 2.* Let  $p = \sum_{\alpha} c_{\alpha} \mathbf{x}^{\alpha} \in \mathcal{P}$  denote a generic (multivariate) polynomial. Given a sequence  $\mathbf{m} \doteq \{m_{\alpha}\}$ , there exists a linear functional  $\mathbf{E}: \mathcal{P} \rightarrow \mathcal{R}$  such that

$$\mathbf{E}(p) = \sum_{\alpha} c_{\alpha} m_{\alpha} \quad (2.18)$$

and  $\{m_{\alpha}\}$  are the moments of a distribution supported on  $\|\mathbf{x}\|_2 \leq \epsilon$ , if and only if the following two conditions hold for all  $p \in \mathcal{P}$ :

$$\mathbf{E}(p^2) \geq 0 \quad (2.19)$$

$$\mathbf{E}((\epsilon^2 - (x_1^2 + \dots + x_D^2))p^2) \geq 0 \quad (2.20)$$

*Remark 2.* The conditions given in the above theorem consist of infinite semidefinite quadratic forms which can be converted into (infinite) linear matrix inequalities (LMIs) in the moment variables  $\{m_{\alpha}\}$ .

Next, we briefly discuss how to build a matrix representation of a given sequence  $\mathbf{m}$  that contains all the moments up to order  $2d$ . Although the order of the subsequence is immaterial, for the sake of clarity of presentation, we arrange the moments according to a graded reverse lexicographic order (grevlex) of the corresponding monomials so that we have  $\mathbf{0} = \alpha^{(1)} < \dots < \alpha^{(M_d)}$ , where  $M_d \doteq \binom{d+D}{D}$  is the number of moments in  $\mathbb{R}^D$  up to order  $d$ . Then, the moment conditions take the form:

$$\begin{aligned} \mathbf{L}^{(d)}(\mathbf{m}) \succeq 0 \\ \mathbf{K}^{(d)}(\epsilon, \mathbf{m}) \succeq 0 \end{aligned} \quad (2.21)$$

where

$$\begin{aligned} \mathbf{L}^{(d)}(i, j) &= m_{\alpha^{(i)} + \alpha^{(j)}} \text{ for all } i, j \leq M_d \\ \mathbf{K}^{(d)}(i, j) &= (\epsilon^2 m_{\alpha^{(i)} + \alpha^{(j)}} - m_{\alpha^{(i)} + \alpha^{(j)} + (2, 0, \dots, 0)} - \\ &\dots - m_{\alpha^{(i)} + \alpha^{(j)} + (0, \dots, 0, 2)}) \text{ for all } i, j \leq M_{d-1} \end{aligned}$$

It can be shown [41] that the linear matrix inequalities in equation (2.21) are necessary conditions for the existence of a measure  $\mu$  supported in the  $\epsilon$ -ball that has the sequence  $\mathbf{m}$  as its moments. Moreover, as  $d \uparrow \infty$ , (2.21) becomes equivalent to conditions (2.19)–(2.20) in Theorem 2; hence it is sufficient

as well. Thus, it is possible to get progressively better approximations to the infinite dimensional conditions (2.19)–(2.20) with finite dimensional LMIs by increasing the size of the moment matrices (i.e. by increasing  $d$ ) [41]. It is also worth noting that if as  $d$  increases, the rank of the moment matrices stops increasing, the so-called *flat extension* property ([42]) is satisfied. In this case, the finite dimensional conditions (2.21) corresponding to this value of  $d$  are necessary and sufficient for the existence of a measure supported in the  $\epsilon$  ball.

**Example 2.1.** *This simple example illustrates the structure of moment matrices when  $D = 2$ ,  $d = 2$ :*

$$\mathbf{L} = \begin{bmatrix} 1 & m_{(1,0)} & m_{(0,1)} & m_{(2,0)} & m_{(1,1)} & m_{(0,2)} \\ m_{(1,0)} & m_{(2,0)} & m_{(1,1)} & m_{(3,0)} & m_{(2,1)} & m_{(1,2)} \\ m_{(0,1)} & m_{(1,1)} & m_{(0,2)} & m_{(2,1)} & m_{(1,2)} & m_{(0,3)} \\ m_{(2,0)} & m_{(3,0)} & m_{(2,1)} & m_{(4,0)} & m_{(3,1)} & m_{(2,2)} \\ m_{(1,1)} & m_{(2,1)} & m_{(1,2)} & m_{(3,1)} & m_{(2,2)} & m_{(1,3)} \\ m_{(0,2)} & m_{(1,2)} & m_{(0,3)} & m_{(2,2)} & m_{(1,3)} & m_{(0,4)} \end{bmatrix} \quad (2.22)$$

$$\mathbf{K} = \epsilon^2 \mathbf{L}(1:3, 1:3) - \begin{bmatrix} m_{(2,0)} & m_{(3,0)} & m_{(2,1)} \\ m_{(3,0)} & m_{(4,0)} & m_{(3,1)} \\ m_{(2,1)} & m_{(3,1)} & m_{(2,2)} \end{bmatrix} - \begin{bmatrix} m_{(0,2)} & m_{(1,2)} & m_{(0,3)} \\ m_{(1,2)} & m_{(2,2)} & m_{(1,3)} \\ m_{(0,3)} & m_{(1,3)} & m_{(0,4)} \end{bmatrix} \quad (2.23)$$

Finally, we state conditions that moment sequences of probability measures supported in an arbitrary compact semialgebraic set  $\mathcal{K}$  should satisfy. For a historical review and details of this more general moment problem, we refer the interested reader to [42, 43] and references therein. As shown in [41, 42], the existence of such a measure can again be characterized by positive semidefiniteness of some infinite matrices, the so-called moment  $\mathbf{M}(m_\alpha)$  and localization matrices  $\mathbf{L}(g_k m_\alpha)$  where  $g_k(x) \geq 0$  are the polynomials defining  $\mathcal{K}$ .

Next, we briefly discuss how to build truncated versions of  $\mathbf{M}$  and  $\mathbf{L}$  of a given sequence  $\mathbf{m} \doteq \{m_\alpha\}$  that contains all the moments up to order  $2d$ . Although the order of the subsequence is immaterial, for the sake of clarity of presentation, we arrange the moments according to a graded reverse lexicographic order (grevlex) of the corresponding monomials so that we have  $\mathbf{0} = \alpha^{(1)} < \dots < \alpha^{(S_d)}$ , where  $S_d \doteq \binom{d+D}{D}$  is the number of moments in  $\mathbb{R}^D$  up to order  $d$ . The truncated version of  $\mathbf{M}$  is defined as follows:

$$\mathbf{M}_d(\mathbf{m})(i, j) = m_{\alpha^{(i)} + \alpha^{(j)}} \text{ for all } i, j \leq S_d. \quad (2.24)$$

Let  $g_k(x) = \sum_{\beta} g_{k,\beta^{(l)}} x^{\beta^{(l)}}$  be one of the defining polynomials of  $\mathcal{K}$  with coefficients  $g_{k,\beta^{(l)}}$  and degree  $\delta_k$ , then the corresponding truncated localization matrix is defined as:

$$\mathbf{L}_d(g_k \mathbf{m})(i, j) = \sum_{\beta} g_{k,\beta^{(l)}} m_{\beta^{(l)} + \alpha^{(i)} + \alpha^{(j)}} \quad (2.25)$$

for all  $i, j \leq S_{d - \lfloor \frac{\delta_k}{2} \rfloor}$

## 2.2.2 Polynomial Optimization via Moments

This section reviews some results from [41] that relate polynomial optimization to the problem of moments. Specifically, consider the problem of minimizing a real valued polynomial:

$$p_{\mathcal{K}}^* := \min_{x \in \mathcal{K}} p(x) \quad (\text{P1})$$

where  $\mathcal{K} \subset \mathbb{R}^{\mathbb{D}}$  is a compact set defined by polynomial inequalities. This problem is usually non-convex, hence hard to solve. Next, we consider a related problem:

$$\tilde{p}_{\mathcal{K}}^* := \min_{\mu \in \mathcal{P}(\mathcal{K})} \int p(x) \mu(dx) := \min_{\mu \in \mathcal{P}(\mathcal{K})} \mathbf{E}_{\mu} [p(x)] \quad (\text{P2})$$

where  $\mathcal{P}(\mathcal{K})$  is the space of finite Borel signed measures on  $\mathcal{K}$ . Although (P2) is an infinite dimensional problem, it is, in contrast to (P1), convex. The next result, taken from [41], establishes the relation between the two problems:

*Theorem 3.* Problems (P1) and (P2) are equivalent; that is:

- $\tilde{p}_{\mathcal{K}}^* = p_{\mathcal{K}}^*$ .
- If  $x^*$  is a global minimizer of (P1), then  $\mu^* = \delta_{x^*}$  (the Dirac at  $x^*$ ) is a global minimizer of (P2).
- For every optimal solution  $\mu^*$  of (P2),  $p(x) = \tilde{p}^*$ ,  $\mu^*$  almost everywhere.

*Proof.* See Proposition 2.1 in [41]. □

One direct consequence of this theorem is that when combined with the LMI-based characterizations of moments presented in Section 2.2.1, it is possible to convert the infinite dimensional problem (P2) in measures (or equivalently, the polynomial optimization problem (P1)) to a semidefinite programming

problem in the moments. When probability measures are supported on compact intervals (i.e. one dimensional case), (P2) can be reduced to a finite dimensional LMI optimization problem in the moments. In the more general multidimensional case, it is possible to obtain asymptotically convergent SDP relaxations. To this effect, define

$$\begin{aligned} p_d^* = \min_{\mathbf{m}} \quad & \sum_{\alpha} p_{\alpha} m_{\alpha} \\ \text{s.t.} \quad & \mathbf{M}_d(\mathbf{m}) \succeq 0, \\ & \mathbf{L}_d(g_k \mathbf{m}) \succeq 0, k = 1, \dots, n_g \end{aligned} \quad (2.26)$$

where moment  $\mathbf{M}_d(\mathbf{m})$  and localization  $\mathbf{L}_d(g_k \mathbf{m})$  matrices are as defined in equations (2.24) and (2.25).

*Theorem 4.* As  $d \rightarrow \infty$ ,  $p_d^* \uparrow p_{\mathcal{K}}^*$ .

*Proof.* See [41]. □

### 2.2.2.1 Exploiting the Sparse Structure

The next property will play a key role in reducing the computational complexity of problem (P1) by exploiting its structure.

*Definition 1.* Let  $\mathcal{K} \in \mathbb{R}^D$  be a semialgebraic set defined by  $n_g$  polynomials  $g_k$ . Let  $I_k \subset \{1, \dots, D\}$  be the set of indices of variables such that each  $g_k$  contains variables from some  $I_k$  and assume that the objective function  $p$  can be partitioned as  $p = p_1 + \dots + p_l$  where each  $p_j$  contains only variables from some  $I_k$ . If there exists a reordering  $I_{k'}$  of  $I_k$  such that for every  $k' = 1, \dots, d-1$ :

$$I_{k'+1} \cap \bigcup_{j=1}^{k'} I_j \subseteq I_s \text{ for some } s \leq k' \quad (2.27)$$

then the *running intersection property* is satisfied.

For the case of generic polynomials and constraints, solving problem (P1) using the method of moments requires considering moments and localization matrices containing  $O(D^{2d})$  variables. On the other hand, if the running intersection property holds, it can be shown [44, 45] that it is possible to define  $n_g$  sets of smaller sized matrices each containing only variables in  $I_k$  (i.e. number of variables is  $O(\kappa^{2d})$ , where  $\kappa$  is the number of elements in the maximum cardinality set among  $I_k$ 's). In

---

many practical applications, including the one considered in Chapter 4,  $\kappa \ll n$ . Hence, exploiting the sparse structure substantially reduces the number of variables in the optimization (and hence the computational complexity), while still providing convergent relaxations.

## Chapter 3

# Identification of a Class of Hybrid Dynamical Systems

### 3.1 Introduction and Motivation

Hybrid systems, systems characterized by the interaction of both continuous and discrete dynamics, have been the subject of considerable attention during the past decade. These systems arise naturally in many different contexts, e.g. biological systems, systems incorporating logical and continuous elements, manufacturing, etc, and in addition, can be used to approximate nonlinear dynamics. As a result of this research, an extensive body of results is now available addressing issues such as controllability/observability, stability analysis and control synthesis. However, applying these results requires using an explicit model of the system under consideration. While in some cases these models can be obtained from first principles, many practical applications require identifying the system from a combination of experimental data and some *a priori* information. This has prompted a substantial research effort devoted towards developing a framework for input/output identification of hybrid systems. As a result, several methods have been proposed addressing different aspects of the problem (see the excellent tutorial paper [46] for a summary of the main issues and recent developments in the field). While successful in many situations, a common feature of these methods is the computational complexity entailed in dealing with noisy measurements: in this case algebraic procedures [47] lead to nonconvex optimization problems, while optimization methods lead to generically NP-hard problems, either necessitating the use of relaxations [48] or restricted to small size problems [49].

Motivated by the computational complexity noted above, in the first portion of this dissertation we propose a new framework to the problem of set membership identification of a class of hybrid systems: switched affine models. Specifically, given noisy input/output data and some minimal *a priori* information about the set of admissible plants, our goal is to identify a suitable set of affine models along with a switching sequence that can explain the available experimental information, while optimizing a performance criteria (either minimum number of plants or minimum number of switches). Our main result shows that this problem can be reduced to a sparsification form, where the goal is to minimize the number of non-zero elements of a given vector sequence. Although in principle this leads to an NP-hard problem, efficient convex relaxations can be obtained by exploiting recent results on sparse signal recovery based on  $\ell_1$ -norm minimization [20, 21]. Then, we illustrate these results using two non-trivial problems arising in computer vision applications: segmentation of video sequences and of dynamic textures. As shown there, application of the proposed techniques outperforms existing state-of-the-art techniques.

In the second part, we consider the hybrid system identification problem when the number of subsystems is known. In this case we propose a moments-based convex optimization approach. The starting point is the algebraic geometric procedure due to Vidal *et al.* [47, 50]. In the case of noiseless measurements, the (unknown) parameters of each subsystem are recovered from the null space of a matrix  $\mathbf{V}(\mathbf{r})$  constructed from the input/output data  $\mathbf{r}$  via a nonlinear embedding (the Veronese map). In the case of noisy data, the entries of this matrix depend polynomially on the unknown noise terms. Thus, finding a model in the consistency set (e.g. a model that interpolates the data within the unknown noise level) is equivalent to finding an admissible noise sequence  $\eta$  that renders the matrix  $\mathbf{V}(\mathbf{r})$  rank deficient, and a vector  $\mathbf{c}$  in its null space. However, this is not a trivial problem, given the polynomial dependence noted above. The main result of this section shows that the problem of jointly finding  $\eta$  and  $\mathbf{c}$  is equivalent to minimizing the rank of a matrix whose entries are affine in the optimization variables, subject to a convex constraint imposing that these variables are the moments of an (unknown) probability distribution function with finite support. This result is achieved by using first an idea similar to that of [41] relating polynomial optimization and the problem of moments, to eliminate the polynomial dependence on the optimization variables, albeit at the price of introducing infinitely many constraints. The structure of the problem, and in particular the independence of the noise terms, can then be exploited to decouple this problem into several *finite dimensional* smaller ones, each involving only the moments of a one-dimensional distribution. Combining these ideas with a convex relaxation, similar to log-det heuristic of [51], that aims at dropping the rank of  $\mathbf{V}$  by one and estimating a vector in its nullspace, allows for recasting the original problem into a semidefinite optimization form that

can be solved efficiently. We illustrate the performance of the proposed method with one academic and one practical example.

## 3.2 Definitions

In this section we will consider switched autoregressive exogenous (SARX) hybrid affine models of the form:

$$y(t) = \sum_{i=1}^{n_a} a_i(\sigma_t)y(t-i) + \sum_{i=1}^{n_c} c_i(\sigma_t)u(t-i) + f(\sigma_t) + \eta(t) \quad (3.1)$$

where  $u$ ,  $y$  and  $\eta$  denote the input, output and noise, respectively, and where  $t \in [t_0, T]$ . The discrete variable  $\sigma_t \in \{1, \dots, s\}$ —the mode of the system—indicates which of the  $s$  submodels is active at time  $t$ . The time instants where the value of  $\sigma_t$  changes are called *discrete transitions* or *switches*. These switches partition the interval  $[t_0, T]$  into a *discrete hybrid time set* [52],  $\tau = \{I_i\}_{i=0}^k$ , such that  $\sigma_t$  is constant within each subinterval  $I_i = [\tau_i, \tau'_i]$  and different in consecutive intervals. In the sequel we denote by  $\tau_i$  and  $\tau'_i$  the beginning and ending times of the  $i^{\text{th}}$  interval, respectively. Clearly,  $\tau$  satisfies:

- $\tau_0 = t_0$ <sup>1</sup> and  $\tau'_k = T$ ,
- $\tau_i \leq \tau'_i = \tau_{i+1} - 1$ ,

and the number of switches is equal to  $k$ .

An equivalent representation of (3.1) is:

$$y(t) = \mathbf{p}(\sigma_t)^T \mathbf{r}(t) + \eta(t) \quad (3.2)$$

where  $\mathbf{r}(t) = [y(t-1), \dots, y(t-n_a), u(t-1), \dots, u(t-n_c), 1]^T$  is the regressor vector and  $\mathbf{p}(\sigma_t) = [a_1(\sigma_t), \dots, a_{n_a}(\sigma_t), c_1(\sigma_t), \dots, c_{n_c}(\sigma_t), f(\sigma_t)]^T$  is the unknown coefficient vector at time  $t$ .

<sup>1</sup>Since it is not possible to deduce information for  $t < \max(n_a, n_c)$  when the initial conditions are unknown, in the identification problem we take  $t_0 = \max(n_a, n_c)$ .



### 3.3 Problem Statement

In this section we consider the problem of identifying switched autoregressive exogenous (SARX) hybrid affine models from experimental measurements corrupted by noise. From a set-membership point of view, this problem can be formally stated as follows:

*Problem 1. [Consistency]* Given input/output data over the interval  $[t_0, T]$ , and a bound  $\epsilon$  on the  $\ell_p$  norm of the noise (i.e.  $\|\eta\| \leq \epsilon$ ), find a hybrid affine model of the form (3.1) that is consistent with the a priori information and experimental data.

It is clear that this problem, though ensuring consistency, is not well-posed and has infinitely many solutions. For instance, one can always find a trivial hybrid model with  $T - t_0 + 1$  submodels or one model with a large order that perfectly fits the data. This situation can be partially avoided by imposing upper bounds  $n_y$  and  $n_u$  on the order of each of the terms on the right hand side of (3.1), e.g.  $n_a \leq n_y$  and  $n_c \leq n_u$  for some known  $n_y, n_u$ . Still, even in this case the problem admits multiple solutions. More interesting problems can be posed by using the existing degrees of freedom to optimize suitable performance criteria.

One such criterion is to minimize the number of switches (i.e. minimum  $k$ ), subject to consistency. Practical situations where this problem is relevant arise for instance in segmentation problems in computer vision and medical image processing, where it is desired to maximize the size of regions (roughly equivalent to minimizing the number of boundaries), and in fault-detection, in cases where it is desired to minimize the number of false alarms.<sup>2</sup> This criterion may also be useful when the piecewise constant mode signal  $\sigma_t$  is known to satisfy a dwell-time constraint (i.e. the time between any consecutive switches is bounded below by a dwell-time) or an average dwell-time constraint (i.e. the number of switches in any given interval is bounded above by its length normalized by an average dwell-time, plus a chatter bound)<sup>3</sup>. The formal statement of the identification problem with this criterion is as follows:

*Problem 2. [Minimum Number of Switches]* Given input/output data over the interval  $[t_0, T]$ , and bounds  $\epsilon > \|\eta\|_p$ ,  $n_u \geq n_c$  and  $n_y \geq n_a$  on the  $\ell_p$  norm of the noise and the order of the regressors, respectively, find a hybrid affine model of the form (3.1) that is consistent with the a priori information and that can explain the experimental data with the minimum number of switches.

<sup>2</sup>A similar problem is considered in econometrics society [53] where a dynamic programming approach is developed for a fixed number of switches (i.e. when  $k$  is known).

<sup>3</sup>These are the discrete-time counterparts of some sets of mode signals defined in [54]. Detailed definitions of different sets of mode signals can be found therein.

An alternative is to try to find the minimum number of submodels (i.e. minimum  $s$ ) capable of explaining the data record. This criterion, used in [48], leads to the following identification problem:

*Problem 3. [Minimum Number of Submodels]* Given input/output data over the interval  $[t_0, T]$ , and bounds  $\epsilon$ ,  $n_y$ ,  $n_u$  on the norm of the noise and regressor orders, find a hybrid affine model of the form (3.1) with minimum number of submodels that is consistent with the a priori information and experimental data.

### 3.4 A Sparsification Approach

In this section, we develop identification schemes where we pose an optimization problem that is in the form of a sparse signal recovery problem. In Section 2.1, we present some background results on sparsification and their extensions that will be used to recast the identification problem into a convex optimization form.

#### 3.4.1 Main Results

In this section we show that both, Problems 2 and 3, can be converted into an equivalent *sparsification* form where the objective is to maximize the number of zero elements of a suitably defined vector valued sequence. While in principle maximizing sparsity is a generically non-convex, hard to solve problem, recent developments in sparse signal recovery reveal that efficient, computationally tractable relaxations can be obtained by exploiting elements from convex analysis. To this effect, we start by defining a time varying parameter vector  $\mathbf{p}(t) \in R^{n_y+n_u+1}$ . Replacing  $\mathbf{p}(\sigma_t)$  in (3.2) with  $\mathbf{p}(t)$ , allows for recasting the consistency problem into the following feasibility form:

$$\begin{aligned} & \text{find } \mathbf{p}(t) \\ & \text{s.t } y(t) - \mathbf{r}(t)^T \mathbf{p}(t) = \eta(t) \quad \forall t \\ & \quad \|\{\eta\}\|_* \leq \epsilon \end{aligned} \tag{3.3}$$

where  $\|\cdot\|_*$  denotes a suitable norm, specified according to the problem under consideration, and where  $\epsilon$  is an upper bound on the noise level. Thus, restricting problems 2 and 3 to the feasible set of (3.3) guarantees consistency.

### 3.4.1.1 Identification with Minimum Number of Switches

In this section we address Problem 2 and show that in the case of  $\ell_\infty$  bounded noise, it can be reduced to a convex optimization. For general convex noise descriptions, the problem can be converted into an equivalent *sparsification* form where the objective is to maximize the number of zero elements of a suitably defined vector valued sequence. The resulting (non-convex) optimization can then be solved using the relaxation proposed in Lemma 1 in the Appendix.

**A Greedy Algorithm for the  $\ell_\infty$  Case:** In the sequel we propose a computationally simple algorithm for solving Problem 2 in the case where the noise term is characterized in terms of its  $\ell_\infty$  norm. This solution is motivated by existing results in time series clustering showing that a greedy sliding window algorithm [55] is optimal. As we show below, similar ideas can be applied to problem 2, leading to an algorithm that entails solving a sequence of smaller linear programs in a greedy fashion.

---

#### Greedy Algorithm

---

$k = 0$

$t_0 = \max(n_y, n_u)$

$\tau_k = t_0$

FOR  $i = t_0 : T$

Solve the following feasibility problem in  $\mathbf{p}$ :

$$\mathcal{F} : \left\{ |y(t) - \mathbf{r}(t)^T \mathbf{p}| \leq \epsilon \quad \forall t \in [\tau_k, i] \right\}$$

IF  $\mathcal{F}$  is infeasible

Set  $I_k = [\tau_k, i - 1]$ ,  $k = k + 1$ , and  $\tau_k = i$

END IF

END FOR

Set  $I_k = [\tau_k, T]$  and  $\tau = \{I_j\}_{j=0}^k$

RETURN  $\tau$  and  $k$

---

TABLE 3.1: Optimal Greedy Algorithm for Problem 2

*Theorem 5.* Let  $k^*$  denote the number of switches in an optimal solution to Problem 2 when the noise is characterized in terms of an  $\ell_\infty$  bound:  $\|\{\eta\}\|_\infty \leq \epsilon$ . Then the value  $k$  returned by the greedy algorithm outlined in Table 3.1 coincides with the optimal  $k^*$ .

*Proof.* Assume  $\tau^* = \{I_i^*\}_{i=0}^{k^*}$  is the discrete hybrid time set corresponding to an optimal solution with  $k^*$  switches. Let  $\tau = \{I_i\}_{i=0}^k$  and  $k$  be the pair of values returned by the greedy algorithm. In order to establish that the proposition is true, it is enough to show that if  $\tau_i \in I_j^*$  then  $\tau'_i \geq \tau'_{j^*}$ . Then, an induction step shows that,  $\tau'_i \geq \tau'_{i^*} \forall i \in \{0, \dots, k^*\}$  implying  $k \leq k^*$ .

Since  $\tau^*$  is optimal (hence feasible),  $\mathbf{p}^*(t)$  is constant in each subinterval  $I^*$ . In particular, there exists  $\mathbf{p}_j$  such that for all  $t \in I_j^*$ ,  $\mathbf{p}^*(t) = \mathbf{p}_j$  and  $|y(t) - \mathbf{r}(t)^T \mathbf{p}_j| \leq \epsilon$ . When  $\tau_i \in I_j^*$ , the same  $\mathbf{p}_j$  is a feasible solution of  $\mathcal{F}$  in the  $(\tau_j^*)^{th}$  iteration of the greedy algorithm since  $\tau_i \in I_j^*$  implies  $[\tau_i, \tau'_{j^*}] \subseteq I_j^*$ . Therefore, the algorithm will continue to the next iteration without entering the if condition within the for loop, which implies  $\tau'_i \geq \tau'_{j^*}$ .

Next, we show by induction that for all  $i \leq k$ , there exists  $j \geq i$  such that  $\tau'_i \geq \tau'_{j^*}$ , hence  $\tau'_i \geq \tau'_{i^*}$ :

- For  $i = 0$ :  $\tau_0 = \tau_0^* \in I_0^* \Rightarrow \tau'_0 \geq \tau'_{0^*}$ .
- For  $i = m$ : Assume  $\exists j \geq m$  s.t.  $\tau'_m \geq \tau'_{j^*}$ .
- For  $i = m + 1$ : From the previous line and properties of hybrid time sets, we have that  $\tau_{m+1} = \tau'_m + 1 > \tau'_m \geq \tau'_{j^*} \Rightarrow \exists l > j$  (or equivalently  $\exists l \geq j + 1$ ) s.t.  $\tau_{m+1} \in I_l^* \Rightarrow \tau'_{m+1} \geq \tau'_{l^*} \geq \tau'_{j+1^*}$ . Since  $j \geq m$  implies  $j + 1 \geq m + 1$ , this proves the induction hypothesis.

Using the fact that  $T = \tau'_k = \tau'_{k^*}$  and the result of the induction particularly at  $i = k$  leads to  $\tau'_k \geq \tau'_{k^*} \Rightarrow \tau'_{k^*} \geq \tau'_{k^*} \Rightarrow k^* \geq k$ .

Since by construction the result of the greedy algorithm is feasible for problem 2 and  $k^*$  is the minimum solution of the problem,  $k^* \leq k$ . Therefore,  $k^* = k$ .  $\square$

*Remark 3.* By construction, the greedy algorithm pushes the end points of each interval forward in time as much as possible (i.e.  $\tau'_i$  is as large as possible). Similarly, running the algorithm backwards (i.e.  $i = T : t_0$ ) would push the start points of intervals (equivalently, end points of previous intervals) backward in time as much as possible. Therefore, running it once backwards and once forwards, it is possible to bracket the true locations of the switches.

**Identifiability of the Switches and Convergence of the Greedy Algorithm:** In this section, we address the issue of identifiability of the switches from input output data. We first present a necessary and sufficient condition under which the switches can be exactly identified in a noiseless setup. Later,

we show that when these identifiability conditions hold, the greedy algorithm given in Table 3.1 finds the exact switching times for sufficiently small noise levels (i.e., as  $\epsilon \rightarrow 0$ ).

*Definition 2.* Let  $\tau = \{I_i\}_{i=0}^k$  be a hybrid time set corresponding to a particular trajectory of a switched linear ARX system.  $\tau$  is said to be *causally identifiable* if whenever  $\sigma_{t-1} \neq \sigma_t$ , it is possible to detect the switch as soon as  $y(t)$  is observed.

*Definition 3.* Given the current regressor vector  $\mathbf{r}(t)$ , two submodels with parameter vectors  $\mathbf{p}_1$  and  $\mathbf{p}_2$  are *one-step indistinguishable* from  $\mathbf{r}(t)$  if  $\mathbf{r}(t)^T(\mathbf{p}_1 - \mathbf{p}_2) = 0$ .

The next result presents a necessary and sufficient condition for a switching sequence to be causally identifiable from input output data. Here, for notational simplicity, we define  $\mathbf{R}_{t_0, t_1} = [\mathbf{r}(t_0), \mathbf{r}(t_0 + 1), \dots, \mathbf{r}(t_1)]$ ,  $\mathbf{Y}_{t_0, t_1} = [y(t_0), y(t_0 + 1), \dots, y(t_1)]^T$  and  $\mathbf{N}_{t_0, t_1} = [\eta(t_0), \eta(t_0 + 1), \dots, \eta(t_1)]^T$ .

*Lemma 3.* If  $\mathbf{r}(\tau_{i+1}) \in \text{range}(\mathbf{R}_{\tau_i, \tau'_i})$  then  $\mathbf{r}^T(\tau_{i+1})[\mathbf{p}_{i+1} - \mathbf{p}_i] = \text{constant}$  for all pairs  $(\mathbf{p}_i, \mathbf{p}_{i+1})$  satisfying

$$\begin{aligned} \mathbf{Y}_{\tau_i, \tau'_i} &= \mathbf{R}_{\tau_i, \tau'_i}^T \mathbf{p}_i \\ y(\tau_{i+1}) &= \mathbf{r}^T(\tau_{i+1}) \mathbf{p}_{i+1} \end{aligned} \quad (3.4)$$

*Proof.* Since  $\mathbf{r}(\tau_{i+1}) \in \text{range}(\mathbf{R}_{\tau_i, \tau'_i})$  then  $\mathbf{r}^T(\tau_{i+1}) = \mathbf{v}^T \mathbf{R}_{\tau_i, \tau'_i}^T$  for some  $\mathbf{v} \neq 0$ . Consider now two pairs  $(\mathbf{p}_i, \mathbf{p}_{i+1})$  and  $(\hat{\mathbf{p}}_i, \hat{\mathbf{p}}_{i+1})$  satisfying (3.4). Then

$$\begin{aligned} \mathbf{r}^T(\tau_{i+1})[\mathbf{p}_{i+1} - \mathbf{p}_i] - \mathbf{r}^T(\tau_{i+1})[\hat{\mathbf{p}}_{i+1} - \hat{\mathbf{p}}_i] &= \mathbf{r}^T(\tau_{i+1})[\hat{\mathbf{p}}_i - \mathbf{p}_i] \\ &= \mathbf{v}^T [\mathbf{R}_{\tau_i, \tau'_i}^T \hat{\mathbf{p}}_i - \mathbf{R}_{\tau_i, \tau'_i}^T \mathbf{p}_i] = 0 \end{aligned}$$

where the last equality follows from the first equality in (3.4).  $\square$

*Theorem 6.* In the noise free case,  $\tau = \{I_i\}_{i=0}^k$  is causally identifiable from input/output data if and only if the following two conditions hold for all  $i$ :

$$\mathbf{r}^T(\tau_{i+1})[\mathbf{p}_{i+1} - \mathbf{p}_i] \neq 0 \quad (3.5)$$

$$\mathbf{r}(\tau_{i+1}) \in \text{range}(\mathbf{R}_{\tau_i, \tau'_i}) \quad (3.6)$$

*Proof. Necessity:* Clearly (3.5) is necessary for the switch to be detectable. To show that (3.6) is also necessary, assume that it fails. Then  $\mathbf{r}^T(\tau_{i+1})\mathbf{R}_{\tau_i, \tau'_i}^\perp \doteq \mathbf{v}^T \neq 0$ , where  $\mathbf{R}_{\tau_i, \tau'_i}^\perp$  denotes a basis for the orthogonal complement of  $\mathbf{R}_{\tau_i, \tau'_i}$ . Define:

$$\mathbf{p} \doteq \mathbf{p}_i + \frac{y(\tau_{i+1}) - \mathbf{r}^T(\tau_{i+1})\mathbf{p}_i}{\|\mathbf{v}\|^2} \mathbf{R}_{\tau_i, \tau'_i}^\perp \mathbf{v}$$

Simple algebra shows that  $\mathbf{p}$  satisfies  $\mathbf{Y}_{\tau_i, \tau'_i} = \mathbf{R}_{\tau_i, \tau'_i}^T \mathbf{p}$  and  $y(\tau_{i+1}) = \mathbf{r}^T(\tau_{i+1}) \mathbf{p}$ . It follows that the model

$$y(t) = \mathbf{r}^T(t) \mathbf{p} \quad (3.7)$$

can explain all the data in the interval  $[\tau_i, \tau_{i+1}]$ , and thus the switch is not causally detectable from the input/output data alone.

*Sufficiency:* Since  $\mathbf{r}^T(\tau_{i+1}) [\mathbf{p}_{i+1} - \mathbf{p}_i] \neq 0$ , it follows, from Lemma 3, that there does not exist a single  $\mathbf{p}$  such that (3.7) holds for all  $t \in [\tau_i, \tau_{i+1}]$ . Hence the switch is causally detectable from the input/output sequences  $\{\mathbf{u}, y\}$ .  $\square$

*Remark 4.* The results about formalizes the intuition that a switch is causally detectable if and only in the two modes involved are not one-step indistinguishable and no new modes of the present model have been excited at the last time step (condition (3.6)). In addition, it can be shown that conditions (3.5)–(3.6) are equivalent to

$$\text{rank}[\mathbf{Y}_{\tau_i, \tau_{i+1}} \mathbf{R}_{\tau_i, \tau_{i+1}}^T] > \text{rank}[\mathbf{R}_{\tau_i, \tau_{i+1}}^T] \quad (3.8)$$

However, the former are easier to generalize to the noisy case.

The next result shows that if the identifiability conditions (3.5)–(3.6) hold, the greedy algorithm finds the exact switches for sufficiently small noise levels.

*Theorem 7.* If a hybrid time set  $\tau$  is causally identifiable, then there exists a noise level  $\epsilon_0$  such that greedy algorithm finds the switches correctly whenever the noise level  $\epsilon$  is below  $\epsilon_0$ .

*Proof.* In order to show that the greedy algorithm correctly identifies the hybrid time set  $\tau$ , we need to show that

$$\|y(\tau_{i+1}) - \mathbf{r}^T(\tau_{i+1}) \mathbf{p}\| > \epsilon \quad (3.9)$$

for all  $\mathbf{p}$  such that

$$\mathbf{R}_{\tau_i, \tau'_i}^T \mathbf{p} = \mathbf{Y}_{\tau_i, \tau'_i} + \mathbf{N}_{\tau_i, \tau'_i} \quad (3.10)$$

or, equivalently  $\|\mathbf{r}(\tau_{i+1})^T \mathbf{p}_{i+1} - \mathbf{r}(\tau_{i+1})^T \mathbf{p}\| \geq 2\epsilon$  for all  $\mathbf{p}$  that satisfy (3.10). Since  $\tau$  is identifiable by hypothesis, it follows from Theorem 6 that for  $i \in \{\mathcal{I}\}$ ,  $\mathbf{r}(\tau_{i+1}) \in \text{range}(\mathbf{R}_{\tau_i, \tau'_i})$ . Hence,  $\mathbf{r}(\tau_{i+1}) = \mathbf{R}_{\tau_i, \tau'_i} \boldsymbol{\lambda}$  for some  $\boldsymbol{\lambda} \neq \mathbf{0}$ <sup>4</sup> and  $\|\mathbf{r}(\tau_{i+1})\|_2 \geq \underline{\sigma}_{R_i} \|\boldsymbol{\lambda}\|_2$ , where  $\underline{\sigma}_{R_i}$  denotes the smallest

<sup>4</sup>If  $\mathbf{R}_{\tau_i, \tau'_i}$  does not have full column rank, this representation is not unique. In this case, we choose  $\boldsymbol{\lambda}^*$  with minimum 2-norm.

(non-zero) singular value of  $\mathbf{R}_{\tau_i, \tau'_i}$ . It follows that, for all  $(\mathbf{p}, \hat{\mathbf{p}})$  that satisfy (3.10) we have

$$\mathbf{r}(\tau_{i+1})^T (\mathbf{p} - \hat{\mathbf{p}}) = \boldsymbol{\lambda}^T \mathbf{R}_{\tau_i, \tau'_i}^T (\mathbf{p} - \hat{\mathbf{p}}) = \boldsymbol{\lambda}^T (\mathbf{N}_{\tau_i, \tau'_i} - \hat{\mathbf{N}}_{\tau_i, \tau'_i})$$

Hence

$$|\mathbf{r}(\tau_{i+1})^T (\mathbf{p} - \hat{\mathbf{p}})| \leq \|\boldsymbol{\lambda}\|_2 \sqrt{(\tau_i - \tau_{i+1})} 2\epsilon \leq 2 \frac{\|\mathbf{r}(\tau_{i+1})\|_2 \sqrt{(\tau_i - \tau_{i+1})}}{\underline{\sigma}_R} \epsilon \doteq b(\epsilon) \quad (3.11)$$

In addition, identifiability of  $\tau$  implies that

$$\|\mathbf{r}(\tau_{i+1})^T \mathbf{p}_{i+1} - \mathbf{r}(\tau_{i+1})^T \mathbf{p}_i\| = \gamma_i > 0 \quad (3.12)$$

From (3.11) and (3.12), it follows that, if the noise level  $\epsilon$  satisfies

$$\epsilon < \min_{i \in \{\mathcal{I}\}} \frac{\underline{\sigma}_{R_i} \gamma_i}{2 \left( \|\mathbf{r}(\tau_{i+1})\|_2 \sqrt{(\tau_i - \tau_{i+1})} + \underline{\sigma}_{R_i} \right)} \quad (3.13)$$

then (3.9) holds for all  $i \in \{\mathcal{I}\}$  and hence all switches will be correctly detected by the greedy algorithm.

□

**The Case of General Convex Noise Descriptions:** In the case of general noise descriptions  $\eta \in \mathcal{N}$  all samples are coupled through the noise description. This requires the use of batch algorithms that consider all available data, as opposed to the greedy one used in the  $\ell_\infty$  case. As we show next, in this case the problem can be reduced to a sparsification form and efficiently reduced to a convex optimization using the tools described in the Appendix. The starting point is to consider the sequence of *first order differences* of the time varying parameters  $\mathbf{p}(t)$ , given by

$$\mathbf{g}(t) = \mathbf{p}(t) - \mathbf{p}(t+1) \quad (3.14)$$

Clearly, since a non-zero element of this sequence corresponds to a *switch*, the sequence should be sparse having only  $k$  non-zero elements out of  $T - t_0$ . Thus, with this definition, Problem 2 is equivalent to the following (non-convex) sparsification problem:

$$\begin{aligned} \min_{\mathbf{p}(t)} \quad & \|\{\mathbf{p}(t) - \mathbf{p}(t-1)\}\|_0 \\ \text{s.t.} \quad & y(t) - \mathbf{r}(t)^T \mathbf{p}(t) \in \mathcal{N} \quad \forall t \end{aligned} \quad (3.15)$$

From Lemma 1 in the Appendix, it follows that a convex relaxation can be obtained replacing  $\|\cdot\|_0$  by  $\|\cdot\|_1$ . A better heuristic can be obtained by adapting to this case the iterative weighted  $\ell_1$  heuristic proposed in ([51, 56, 57]). This requires solving, at each iteration, the following convex program:

$$\begin{aligned} \text{minimize}_{z,p} \quad & \sum_t w_t^{(k)} z_t \\ \text{subject to} \quad & \|\mathbf{p}(t) - \mathbf{p}(t-1)\|_\infty \leq z_t \quad \forall t \\ & y(t) - \mathbf{r}(t)^T \mathbf{p}(t) \in \mathcal{N} \quad \forall t \end{aligned} \quad (3.16)$$

where  $w_t^{(k)} = (z_t^{(k)} + \delta)^{-1}$  and where  $z_t^{(k)}$  denotes the optimal solution at the  $k^{\text{th}}$  iteration, with  $z^{(0)} = [1, 1, \dots, 1]^T$ , and  $\delta$  is a (small) regularization constant.

In the first iteration, this method solves the standard  $\ell_1$ -norm relaxation. Then at each subsequent iteration, it increases the weight  $w_t^{(k)}$  associated with the small  $z_t^{(k)}$ s, thus pushing these elements further towards zero.

*Remark 5.* Algorithm (3.16) requires solving  $m$  linear programs with  $(n_y + n_u + 2) \times (T - t_0 + 1)$  variables and  $2(n_y + n_u + 2) \times (T - t_0 + 1)$  inequality constraints, where  $m$  is the number of iterations required for convergence of the weighted  $\ell_1$ -norm relaxation, typically around 5. On the other hand, the greedy algorithm requires solving  $(T - t_0 + 1)$  linear programs with only  $(n_y + n_u + 1)$  variables and at most  $2(T - t_0 + 1)$  inequality constraints (the worst case scenario is when a single parameter value is feasible for the entire  $[t_0, T]$  interval). Thus, in cases where both algorithms are applicable (e.g. when the noise is characterized in terms of its  $\ell_\infty$  norm), the greedy algorithm is preferable from a computational complexity standpoint.

**Extension to Multi-input Multi-output Models:** It is straight forward to extend the sparsity based identification procedure with minimum number of switches criterion to multi-input multi-output (MIMO) models. Consider the MIMO switched ARX model with  $m_u$  inputs and  $m_y$  outputs:

$$y(t) = \sum_{i=1}^{n_a} A_i(\sigma_t) y(t-i) + \sum_{i=1}^{n_c} C_i(\sigma_t) u(t-i) + f(\sigma_t) + \eta(t) \quad (3.17)$$

where  $y \in \mathbb{R}^{m_y}$ ,  $u \in \mathbb{R}^{m_u}$  are outputs and inputs,  $A_i \in \mathbb{R}^{m_y \times m_y}$ ,  $C_i \in \mathbb{R}^{m_y \times m_u}$  and  $f \in \mathbb{R}^{m_y}$  are coefficient matrices, and  $\eta \in \mathbb{R}^{m_y}$  is the noise, respectively. It is possible to solve for coefficient



matrices in a similar manner as in (3.16). Particularly, the following modification steps are required: (i) define time varying coefficient matrices (i.e.  $A_i(t)$ ,  $C_i(t)$  and  $f(t)$ ) (ii) form  $\mathbf{p}(t) \in \mathbb{R}^{m_y^2 + m_y m_u + m_y}$  by stacking the elements of the coefficient matrices at time  $t$  into a column vector, and (iii) replace the regressor equation in (3.16) with the multivariate regressor corresponding to (3.17).

**Extension to Multidimensional Models:** In this section we consider the identification of hybrid multidimensional systems (i.e. hybrid systems where the process dynamics depend on more than one indeterminate). In particular, systems that are governed by *affine switched-coefficient difference equations* (ASCDEs) are considered. ASCDE is a generalization of the so-called *linear constant-coefficient difference equations* (LCCDEs) ([58]). An  $n$ -dimensional ASCDE has the following form:

$$\begin{aligned} y(t_1, \dots, t_n) = & \sum_{(k_1, \dots, k_n) \in \mathcal{R}_a} a_{k_1, \dots, k_n}(\sigma_{t_1, \dots, t_n}) y(t_1 - k_1, \dots, t_n - k_n) \\ & + \sum_{(k_1, \dots, k_n) \in \mathcal{R}_c} c_{k_1, \dots, k_n}(\sigma_{t_1, \dots, t_n}) u(t_1 - k_1, \dots, t_n - k_n) \\ & + f(\sigma_{t_1, \dots, t_n}) + \eta(t_1, \dots, t_n) \end{aligned} \quad (3.18)$$

where  $y$  is the output;  $u$  is the input;  $\eta$  is noise;  $\sigma_{t_1, \dots, t_n} \in \{1, \dots, s\}$  is the discrete mode signal as before.  $\mathcal{R}_a, \mathcal{R}_c \subset \mathbb{Z}^n$  are coefficient support regions; and  $a_{k_1, \dots, k_n}(\sigma_{t_1, \dots, t_n})$ ,  $c_{k_1, \dots, k_n}(\sigma_{t_1, \dots, t_n})$  and  $f_{k_1, \dots, k_n}(\sigma_{t_1, \dots, t_n})$  are the coefficients to be identified.

3-D models of this form in noise-free setup are considered in [59] for spatiotemporal segmentation. Such a model can be useful in approximating the behavior of a wave traveling in an inhomogeneous space or the images where different regions of the image are occupied by different textures.

As a shorthand notation, we denote the indeterminate in vector form, i.e.  $\mathbf{t} = [t_1, \dots, t_n] \in \mathbb{Z}^n$ . Let  $\mathcal{D} \subset \mathbb{Z}^n$  denote the domain over which experimental measurements are collected. The interior of the domain can be prescribed as:

$$\text{int}(\mathcal{D}) = \{\mathbf{t} \in \mathcal{D} \mid \mathbf{t} - \mathbf{k} \in \mathcal{D} \forall \mathbf{k} \in \mathcal{R}_a \forall \mathbf{k} \in \mathcal{R}_c\} \quad (3.19)$$

Now, we can define the set of neighboring indices as a set of unordered pairs:

$$\mathcal{I} = \{\{\mathbf{t}, \tilde{\mathbf{t}}\} \mid \|\mathbf{t} - \tilde{\mathbf{t}}\|_1 = 1, \text{ and } \mathbf{t}, \tilde{\mathbf{t}} \in \text{int}(\mathcal{D})\}. \quad (3.20)$$

A *switch* is defined between neighboring indices. That is, we say that there is a *switch* whenever  $\sigma_{\mathbf{t}} \neq \sigma_{\tilde{\mathbf{t}}}$  for  $\{\mathbf{t}, \tilde{\mathbf{t}}\} \in \mathcal{I}$  (analogous to 1D case where switches are defined between  $t$  and  $t + 1$ ). A

multidimensional hybrid “time” set is a partition  $\{P_i\}$  of  $\text{int}(\mathcal{D})$  such that within each part (where we call the elements of partition as parts or segments)  $\sigma_{\mathbf{t}}$  is constant and it is different between neighboring parts ( $P_i$  and  $P_j$  are called neighboring parts if there exists  $\{\mathbf{t}, \tilde{\mathbf{t}}\} \in \mathcal{I}$  such that  $\mathbf{t} \in P_i$  and  $\tilde{\mathbf{t}} \in P_j$ ).

As in the 1D case, identification of a system of the form (3.18) is ill-posed since for example one can choose a partition where each part consists of a single  $\mathbf{t}$ . We are interested in finding a partition with minimum number of switches (this corresponds to minimizing the boundary of the segments in image segmentation problem). In order to minimize the number of switches, one should consider sparsifying the following difference sequence:

$$\mathbf{g}(i) = \mathbf{p}(\mathbf{t}) - \mathbf{p}(\tilde{\mathbf{t}}), \quad \{\mathbf{t}, \tilde{\mathbf{t}}\} \in \mathcal{I} \quad (3.21)$$

where  $\mathbf{p}(\mathbf{t}) = [a_{\mathbf{k}_1}(\sigma_{\mathbf{t}}), \dots, a_{\mathbf{k}_{n_a}}(\sigma_{\mathbf{t}}), c_{\mathbf{k}_1}(\sigma_{\mathbf{t}}), \dots, c_{\mathbf{k}_{n_c}}(\sigma_{\mathbf{t}}), f(\sigma_{\mathbf{t}})]$ , and  $i = 1, \dots, |\mathcal{I}|$  is an index counting the elements of  $\mathcal{I}$ . Then, the identification problem can be written as:

$$\begin{aligned} \min_{\mathbf{p}(\mathbf{t})} \quad & \|\{\mathbf{g}(i)\}\|_0 \\ \text{s.t.} \quad & \text{Equation (3.18)} \\ & \|\{\eta\}\|_* \leq \epsilon \end{aligned} \quad (3.22)$$

which can be solved, exactly as in the 1D case, with the weighted  $\ell_1$  norm approximation.

### 3.4.1.2 Identification with Minimum Number of Submodels:

In this section, motivated by an idea used in [48], we present an iterative procedure for solving Problem 3. The main idea is to find one submodel at a time, along with the associated parameter vector  $\tilde{\mathbf{p}}$ , through the solution of a sparsification problem. This is accomplished by finding a parameter vector  $\tilde{\mathbf{p}}$  that makes  $|y(t) - \mathbf{r}(t)^T \tilde{\mathbf{p}}| \leq \epsilon$  feasible for as many time instants  $t$  as possible. Equivalently, defining  $\tilde{\mathbf{g}}(t) = \mathbf{p}(t) - \tilde{\mathbf{p}}$ , the goal is to maximize sparsity of  $\tilde{\mathbf{g}}(t)$  leading to the following optimization problem:

$$\begin{aligned} \min_{\mathbf{p}(t), \tilde{\mathbf{p}}} \quad & \|\{\mathbf{p}(t) - \tilde{\mathbf{p}}\}\|_0 \\ \text{s.t.} \quad & |y(t) - \mathbf{r}(t)^T \mathbf{p}(t)| \leq \epsilon \quad \forall t \end{aligned} \quad (3.23)$$

**Algorithm for Minimum # of Submodels**

$$t_0 = \max(n_y, n_u)$$

$$N_1 = \{t_0, \dots, T\}$$

$$l = 0$$

WHILE  $N_{l+1} \neq \emptyset$

Let  $l = l + 1$

Find  $\tilde{\mathbf{p}}_l$  by solving the re-weighted  $\ell_1$  optimization:

$$\begin{aligned} & \min_{z_t, \mathbf{p}(t), \tilde{\mathbf{p}}} \sum_t w_t^{(k)} z_t \\ & \text{subject to } \|\mathbf{p}(t) - \tilde{\mathbf{p}}\|_\infty \leq z_t \quad \forall t \in N_l \\ & \quad |y(t) - \mathbf{r}(t)^T \mathbf{p}(t)| \leq \epsilon \quad \forall t \in N_l \\ & \text{where } w_j^{(k)} = (z_j^{(k)} + \delta)^{-1} \text{ are weights with } z_j^{(k)} \text{ the arguments of the optimal solution in } k^{\text{th}} \text{ iteration and } \mathbf{z}^{(0)} = [1, 1, \dots, 1]^T; \text{ and } \delta \text{ is the regularization constant.} \end{aligned}$$

Let  $i = 1$

WHILE  $i < l$

$$\text{Let } K_{il} = \{t \in N_i : |y(t) - \mathbf{r}(t)^T \tilde{\mathbf{p}}_l| \leq \epsilon\}$$

IF  $\#K_{il} > \#K_i$

Let  $\tilde{\mathbf{p}}_i = \tilde{\mathbf{p}}_l$  and  $l = i$

END IF

Let  $i = i+1$

END WHILE

$$\text{Let } K_l = \{t \in N_l : |y(t) - \mathbf{r}(t)^T \tilde{\mathbf{p}}_l| \leq \epsilon\}$$

$$\text{Let } N_{l+1} = N_l \setminus K_l$$

END WHILE

RETURN  $s = l$  and  $K_i, i = 1, \dots, s$

TABLE 3.2: Algorithm for Problem 3

Then, we can eliminate the time instants  $t$  for which  $\tilde{\mathbf{g}}(t)$  is zero, and solve the same problem with the rest of the  $t$ 's up until all data points are clustered. The number of times (3.23) is solved gives an upper bound on the minimum number of submodels  $s$ . Combining this idea with a refinement step similar to the one proposed in [48] to re-estimate parameter values and reassign, if needed, data points, leads to the overall algorithm listed in Table 3.2, where minimization of  $\|\cdot\|_0$  is (approximately) accomplished through the use of the weighted  $\ell_1$  norm minimization relaxation.

Next, we briefly discuss an alternative formulation of the problem (3.23). The main idea is similar to that of above that is, to find a parameter vector  $\tilde{\mathbf{p}}$  that makes  $[y(t) - \mathbf{r}(t)^T \tilde{\mathbf{p}}] \in \mathcal{N}$  feasible for as

many time instants  $t$  as possible. Equivalently, if we define  $e(t) = y(t) - \mathbf{r}(t)^T \tilde{\mathbf{p}} + \eta(t)$ , the goal is to maximize sparsity of  $e(t)$  leading to the following optimization problem:

$$\begin{aligned} \min_{e(t), \tilde{\mathbf{p}}} \quad & \|\{e(t)\}\|_0 \\ \text{s.t.} \quad & [e(t) - y(t) + \mathbf{r}(t)^T \tilde{\mathbf{p}}] \in \mathcal{N} \quad \forall t \end{aligned} \quad (3.24)$$

We define,  $\mathbf{R}_{t_0, T} = [\mathbf{r}(t_0), \mathbf{r}(t_0+1), \dots, \mathbf{r}(T)]$ ,  $\mathbf{y} = [y(t_0), y(t_0+1), \dots, y(T)]^T$  and  $\mathbf{e} = [e(t_0), e(t_0+1), \dots, e(T)]^T$ . Assuming noise is characterized with a norm bound, we obtain the following sparsification problem:

$$\begin{aligned} \min_{\mathbf{e}} \quad & \|\mathbf{e}\|_0 \\ \text{s.t.} \quad & \left\| \begin{bmatrix} \mathbf{I} & \mathbf{R}_{t_0, T} \end{bmatrix} \begin{bmatrix} \mathbf{e} \\ \tilde{\mathbf{p}} \end{bmatrix} - \mathbf{y} \right\|_* \leq \epsilon \end{aligned} \quad (3.25)$$

The advantage of this formulation over (3.23) is that instead of having a time dependent parameter vector  $\mathbf{p}(t)$ , we just have the error sequence  $e(t)$  as unknowns. Hence, as long as number of outputs  $n_y$  is smaller than order of the regressor  $n_c + n_a$ , there are less variables in optimization. The other advantage is that problem (3.25) has the same form as standard sparse recovery problems for single output systems.

*Remark 6.* While consistent numerical experience shows that the idea of finding one system at a time works well in practice, counterexamples are available where it overestimates the number of systems. This is due to its greedy nature that tends to assign as many points as possible to the parameters found earlier, possibly resulting in the later need to use additional parameter values in order to explain unassigned data points. At this point the issues of existence of conditions under which the greedy algorithm is indeed optimal and bounds on its worst case performance are open research questions.

### 3.4.2 Examples

*Example 1:* This example illustrates the fact that dwell-time constraints are not necessary for identifiability of the switches. Consider three autonomous systems ( $\sigma_t \in \{1, 2, 3\}$ ) of the form:

$$y_t = a_1(\sigma_t)y_{t-1} + a_2(\sigma_t)y_{t-2} + a_3(\sigma_t)y_{t-3}$$

with

$$\begin{aligned} [a_1(1), a_2(1), a_3(1)] &= [-3, 2, 1] \\ [a_1(2), a_2(2), a_3(2)] &= \left[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right] \\ [a_1(3), a_2(3), a_3(3)] &= [2, -1, 1] \end{aligned}$$

and

$$\sigma_t = \begin{cases} 1, & t \in [1, 4] \\ 2, & t = 5 \\ 3, & t = 6 \end{cases}$$

The trajectory corresponding to the initial conditions  $y_0 = 0, y_{-1} = 7, y_{-2} = -12$ , is given by 2, 1, 1, 1, 1, 2. Thus, the rank condition (3.8) evaluated at  $t = 6$  yields

$$\text{rank} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 1 \end{bmatrix} = \text{rank} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} + 1$$

which implies that it is possible to detect the switch from  $t = 5$  to  $t = 6$  although the system remains in  $\sigma_t = 2$  for only one time instant.

*Example 2:* The goal of this example is to illustrate that noiseless switch identifiability does not imply that mode switches are identifiable under arbitrarily small noise. To this effect consider a system with 2 submodels: the first corresponds to  $\mathbf{p}_1 = \begin{bmatrix} 1/3 & 1/3 & 1/3 \end{bmatrix}$ , and is active for  $t=1,2$ . The second corresponds to  $\mathbf{p}_2 = \begin{bmatrix} 1 & -1 & 2 \end{bmatrix}$  and is active for  $t=3$ . The trajectory corresponding to the initial conditions  $\mathbf{r}(1) = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$  and no external input is given by  $y(1) = 1, y(2) = 1$  and  $y(3) = 2$ . In this case the associated matrices satisfy:

$$\text{rank} \left( \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 1 \end{bmatrix} \right) > \text{rank} \left( \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \right)$$

Hence the switch is causally identifiable. However, adding the noise sequence  $\eta(1) = \epsilon, \eta(2) = 0, \eta(3) = 0$  leads to the trajectory:  $y(1) = 1 + \epsilon, y(2) = 1 + \frac{\epsilon}{3}, y(3) = 2 - 2\frac{\epsilon}{3}$ . In this case the corresponding matrices satisfy:

$$\text{rank} \left( \begin{bmatrix} 1 + \epsilon & 1 & 1 & 1 \\ 1 + \frac{\epsilon}{3} & 1 + \epsilon & 1 & 1 \\ 2 - 2\frac{\epsilon}{3} & 1 + \frac{\epsilon}{3} & 1 + \epsilon & 1 \end{bmatrix} \right) = \text{rank} \left( \begin{bmatrix} 1 & 1 & 1 \\ 1 + \epsilon & 1 & 1 \\ 1 + \frac{\epsilon}{3} & 1 + \epsilon & 1 \end{bmatrix} \right) \quad \forall \epsilon > 0$$

Hence, the switch is not identifiable, regardless of how small  $\epsilon$  is. This is due to the fact that in this case condition (3.6) fails since  $\mathbf{r}(3) = \begin{bmatrix} 1 + \frac{\epsilon}{3} & 1 + \epsilon & 1 \end{bmatrix}^T \notin \text{range} \left( \begin{bmatrix} 1 & 1 & 1 \\ 1 + \epsilon & 1 & 1 \end{bmatrix}^T \right)$ .

*Example 3:* In this example, we considered input/output data generated by a hybrid system that switched among the following two ARX submodels:

$$y(t) = 0.2y(t-1) + 0.24y(t-2) + 2u(t-1); \quad t \in [1, 25] \cup [51, 75]$$

$$y(t) = -1.4y(t-1) - 0.53y(t-2) + u(t-1); \quad t \in [26, 50] \cup [76, 100]$$

with  $\|\eta\|_\infty = 0.5$ . The goal here was to identify a model that explained the experimental data record with the fewest possible number of switches. Figure 3.1 compares the performance of sparsification-based (both the  $\ell_1$ -based algorithm (3.15) and the greedy algorithm of Table 3.1) against the algebraic method and the bounded error method. As shown there, the sparsification based methods correctly estimated the parameters and number of switches, while the other two failed to do so. The running times for  $\ell_1$ -based, greedy, algebraic and bounded error methods are 5.9, 40.5, 1.1 and 17.9 seconds respectively. Additional examples illustrating the use of sparsification to find the minimum number of switches are given in section 3.4.3.

For this example,  $\epsilon_o$  in (3.13) was found to be 0.3136 which corresponds to 4.15% of the maximum absolute value of the output  $y(t)$ . Note that the analysis in section 3.4.1.1 is worst-case, meaning for all noise values below  $\epsilon_o = 0.3136$ , the greedy algorithm finds the correct switches. Even though the noise level in our example was 0.5 which was above  $\epsilon_o$ , greedy algorithm was still capable of correctly detecting the switches.

*Example 4:* This example considers the problem of estimating the minimum number of systems and investigates the effects of noise level on algorithm performance. The data used corresponded to the trajectories of 100 randomly generated SARX models of the form:

$$y(t) = a_1(\sigma_t)y(t-1) + a_2(\sigma_t)y(t-2) + c_1(\sigma_t)u(t-1) + \eta(t) \quad (3.26)$$

with

$$\sigma_t = \begin{cases} 1, & t \in [1, 60] \\ 2, & t \in [61, 120] \\ 3, & t \in [121, 180] \end{cases}$$

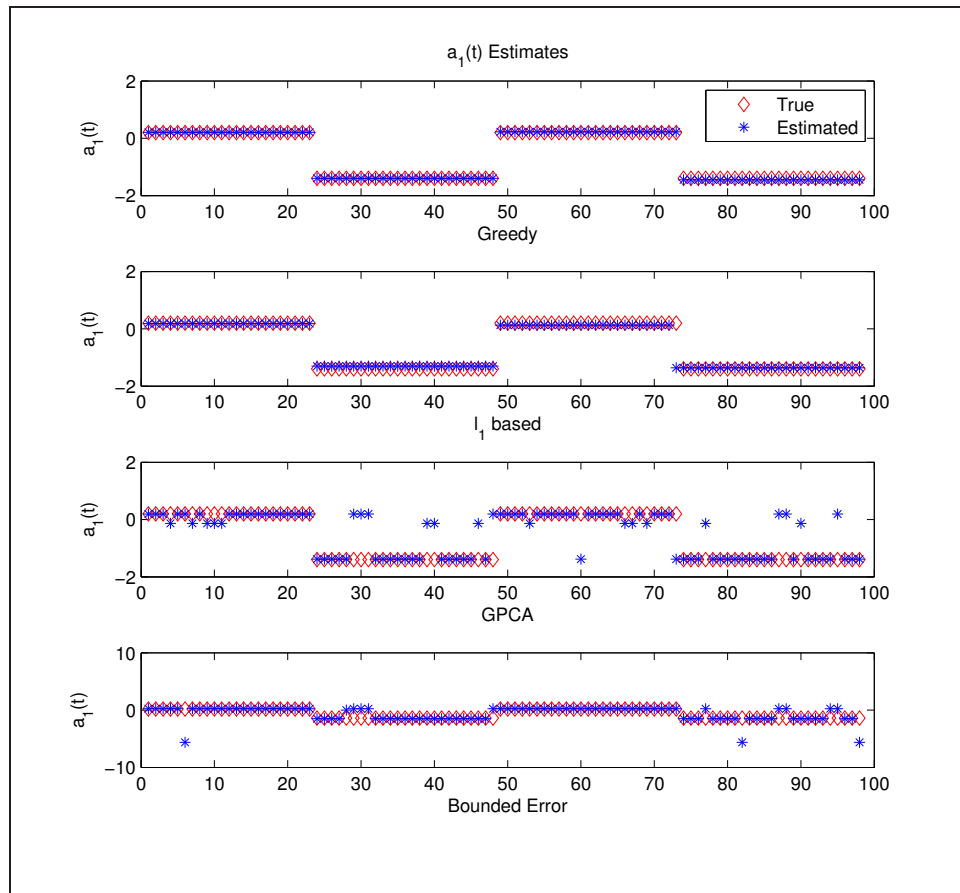


FIGURE 3.1: True and estimated parameter sequences for parameter  $a_1(\sigma_t)$  for Example 3.

where for all  $i \in \{1, 2, 3\}$ ,  $c_1(i)$  is a sample from a zero mean unit variance normal distribution,  $a_1(i)$  and  $a_2(i)$  are chosen such that the complex conjugate poles of the  $i^{\text{th}}$  submodel are distributed in  $0.5 \leq \|z\| \leq 1$  with uniform random phase and magnitude, and  $\eta(t)$  is an iid noise term uniformly distributed in  $[-\epsilon, \epsilon]$ . For each of these systems, the number of submodels was estimated by solving the minimum submodels problem with our method and the bounded-error method; and by approximating the rank of an appropriate matrix obtained from data as proposed in [50] for the algebraic method. The former two methods give upper bounds of true value  $s = 3$ , whereas the latter estimate depends on the threshold chosen to calculate the rank and could be lower than the true value. The same experiment was repeated for different noise levels. Results on these experiments are summarized in Figure 3.2 and Table 3.3.

Next we consider the parameter estimation accuracy for the same 100 random systems. To this end, the following normalized parameter identification error measure is defined:

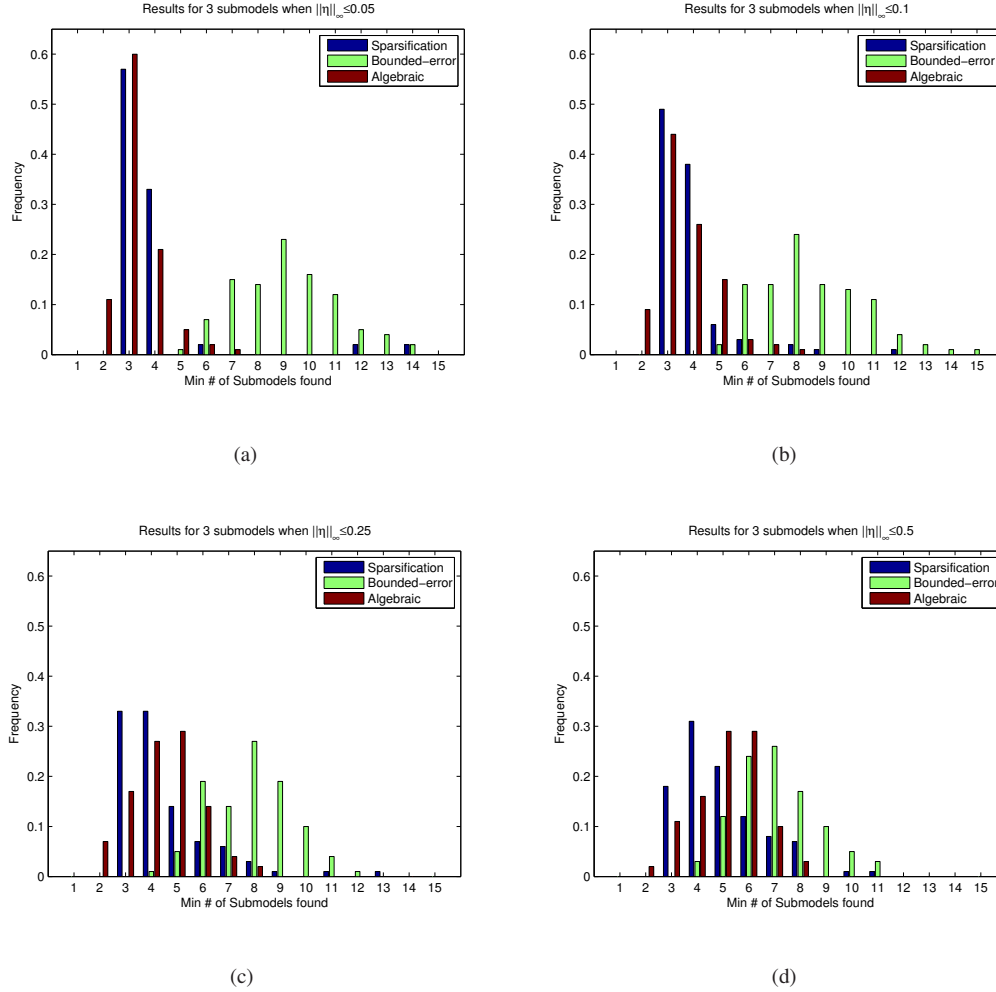


FIGURE 3.2: Each histogram shows the frequency of estimated number of submodels for different noise levels. (a)  $\epsilon = 0.05$ , (b)  $\epsilon = 0.1$ , (c)  $\epsilon = 0.25$ , (d)  $\epsilon = 0.5$ . The true number of submodels is  $s = 3$ .

$$\Delta_n = \frac{1}{T - t_0 + 1} \sum_{t=t_0}^T \frac{\|\mathbf{p}(\sigma_t) - \hat{\mathbf{p}}(\hat{\sigma}_t)\|_2}{\|\mathbf{p}(\sigma_t)\|_2} \quad (3.27)$$

The parameter estimation results are summarized in Figure 3.3 and Table 3.4. As shown there, the sparsification-based method outperformed both the bounded-error and algebraic procedures. While all methods proved considerably robust to noise in estimating the number of submodels, segmentation quality and parameter identification performance degraded significantly for the algebraic method as the noise level increased. On the other hand, sparsification was the most robust in terms of these



| Noise Level $\epsilon$ | Absolute Error            | Sparsification | Bounded-Error | Algebraic |
|------------------------|---------------------------|----------------|---------------|-----------|
| 0.05                   | Mean                      | 1.40 (0.99)    | 6.20          | 0.52      |
|                        | Standard deviation        | 3.48 (2.66)    | 2.09          | 0.77      |
|                        | Median                    | 0 (0)          | 6             | 0         |
|                        | Median absolute deviation | 0 (0)          | 1             | 0         |
| 0.1                    | Mean                      | 0.84 (1.04)    | 5.64          | 0.87      |
|                        | Standard deviation        | 1.36 (2.20)    | 2.03          | 1.02      |
|                        | Median                    | 1 (0.5)        | 5             | 1         |
|                        | Median absolute deviation | 1 (0.5)        | 1             | 1         |
| 0.25                   | Mean                      | 1.59 (1.95)    | 4.84          | 1.60      |
|                        | Standard deviation        | 2.17 (2.31)    | 1.61          | 1.16      |
|                        | Median                    | 1 (1)          | 5             | 1         |
|                        | Median absolute deviation | 1 (1)          | 1             | 1         |
| 0.5                    | Mean                      | 1.93 (2.39)    | 4.07          | 2.18      |
|                        | Standard deviation        | 1.65 (2.11)    | 1.58          | 1.25      |
|                        | Median                    | 2 (2)          | 4             | 2         |
|                        | Median absolute deviation | 1 (1)          | 1             | 1         |

TABLE 3.3: Minimum number of submodel estimation error statistics for different noise levels. The results for sparsification is given both with formulation (3.23) and with formulation (3.24) in parentheses.

performance criteria. The bounded-error method performed relatively poorly when estimating the number of submodels. Even though it clustered most of the data in the largest three submodels, it also generated superfluous submodels with parameter values far from the true values.

Among the two sparsification formulations (3.23) and (3.24) (results shown in parentheses in Tables 3.3 and 3.4 for the latter), we did not observe a significant difference in terms of these performance criteria, with (3.23) performing slightly better. Also, although problem (3.24) involves less variables, it required more iterations of the algorithm given in Table 3.2. Hence, the overall computation times were also similar.

*Example 5:* For the next example, we present more numerical results on a single system. In particular, we considered input/output data generated by an SARX system of the form (3.26) with the same mode signal  $\sigma_t$  and coefficients:

$$\begin{aligned}
 [a_1(1), a_2(1), c_1(1)] &= [-1.6758, -0.8292, 1.8106] \\
 [a_1(2), a_2(2), c_1(2)] &= [-0.8402, -0.6770, 0.2150] \\
 [a_1(3), a_2(3), c_1(3)] &= [1.0854, -0.9501, 0.6941].
 \end{aligned}$$

| Noise Level $\epsilon$ | $\Delta_n$                | Sparsification | Bounded-Error | Algebraic |
|------------------------|---------------------------|----------------|---------------|-----------|
| 0.05                   | Mean                      | 0.11 (0.08)    | 0.38          | 0.20      |
|                        | Standard deviation        | 0.23 (0.19)    | 0.77          | 0.27      |
|                        | Median                    | 0.04 (0.04)    | 0.20          | 0.08      |
|                        | Median absolute deviation | 0.02 (0.02)    | 0.10          | 0.05      |
| 0.1                    | Mean                      | 0.11 (0.15)    | 1.72          | 0.79      |
|                        | Standard deviation        | 0.18 (0.44)    | 10.90         | 4.30      |
|                        | Median                    | 0.06 (0.07)    | 0.25          | 0.15      |
|                        | Median absolute deviation | 0.02 (0.03)    | 0.10          | 0.08      |
| 0.25                   | Mean                      | 0.24 (0.26)    | 0.49          | 0.61      |
|                        | Standard deviation        | 0.21 (0.21)    | 0.87          | 0.82      |
|                        | Median                    | 0.17 (0.19)    | 0.34          | 0.37      |
|                        | Median absolute deviation | 0.08 (0.08)    | 0.12          | 0.17      |
| 0.5                    | Mean                      | 0.41 (0.42)    | 0.61          | 1.05      |
|                        | Standard deviation        | 0.27 (0.27)    | 0.50          | 1.43      |
|                        | Median                    | 0.35 (0.35)    | 0.49          | 0.70      |
|                        | Median absolute deviation | 0.15 (0.15)    | 0.12          | 0.28      |

TABLE 3.4: Normalized parameter identification error statistics of minimum number of submodels problem with different noise level. The results for sparsification is given both with formulation (3.23) and with formulation (3.24) in parantheses.

In this case we used two different criteria to assess the performance, for different noise levels, of the sparsification-based, bounded error, and algebraic algorithms, both in terms of quality of the segmentation and parameter identification error. Quality of the clustering was assessed using the Rand index [60] to compare the estimated mode signal  $\hat{\sigma}_t$  against the true  $\sigma_t$ <sup>5</sup>. Quality of the parameter estimation was evaluated using  $\Delta_n$  in (3.27).

For the noise level of  $\epsilon = 0.05$ , the sparsification, bounded error and algebraic methods found 4, 9 and 4 submodels, respectively. For the noise level of  $\epsilon = 0.5$ , the number of submodels found were 3, 9 and 4, respectively. The results of these experiments are summarized in Tables 3.5 and 3.6.

| Noise Level       | Sparsification | Bounded-error | Algebraic |
|-------------------|----------------|---------------|-----------|
| $\epsilon = 0.05$ | 0.9681         | 0.9157        | 0.9212    |
| $\epsilon = 0.5$  | 0.8436         | 0.7482        | 0.6849    |

TABLE 3.5: Rand Indices that show the quality of mode signal estimates.

In these last two examples the sparsification-based method outperformed both the bounded-error and algebraic procedures. While all methods proved considerably robust to noise in estimating the number of submodels, segmentation quality and parameter identification performance degraded significantly

<sup>5</sup>Recall that a Rand index of 1 corresponds to a perfect clustering.

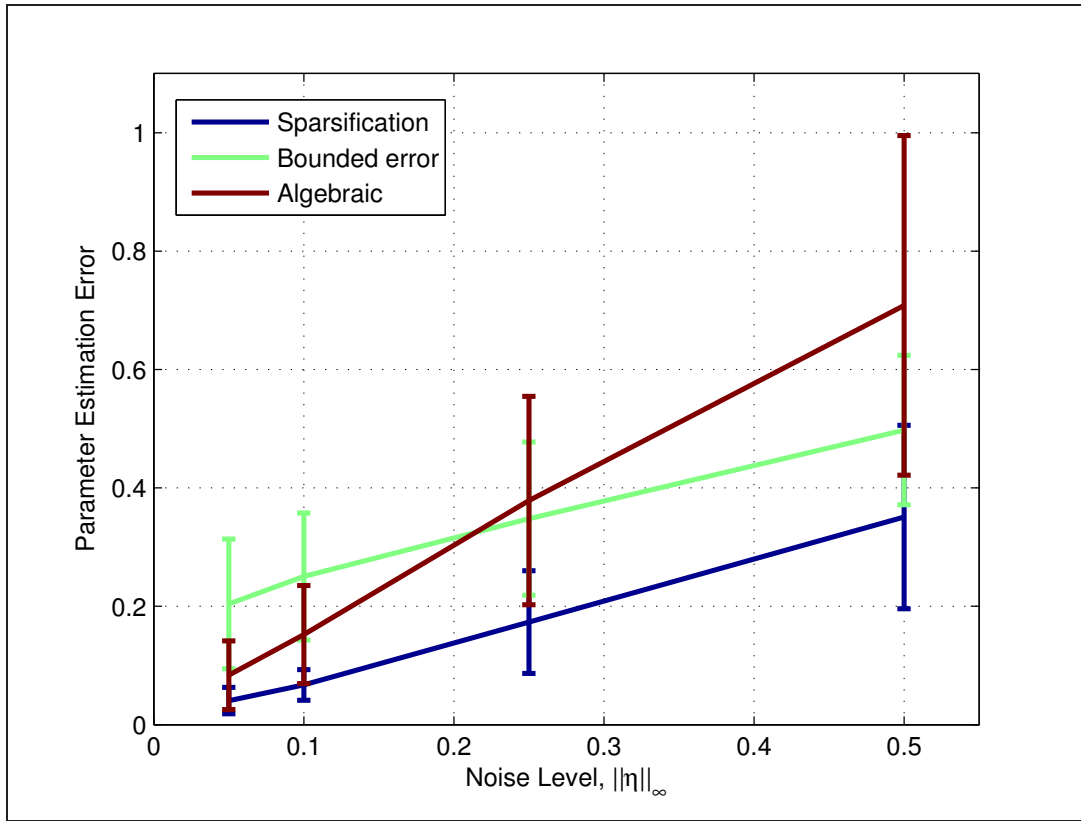


FIGURE 3.3: Median of parameter estimation error  $\Delta_n$  versus noise level  $\epsilon$ . Error bars indicate the median absolute deviation.

| Noise Level       | Sparsification | Bounded-error | Algebraic |
|-------------------|----------------|---------------|-----------|
| $\epsilon = 0.05$ | 0.0699         | 0.2883        | 0.0504    |
| $\epsilon = 0.5$  | 0.1459         | 0.8604        | 0.9245    |

TABLE 3.6: Error measure  $\Delta_n$  that shows the quality of parameter estimates.

for the algebraic method as the noise level increased. On the other hand, sparsification was the most robust in terms of these performance criteria. The bounded-error method performed relatively poorly when estimating the number of submodels. Even though it clustered most of the data in the largest three submodels, it also generated superfluous submodels with parameter values far from the true values.

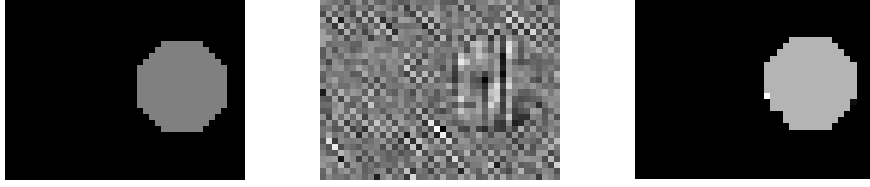


FIGURE 3.4: Results for detecting switches in a 2-D system. Left: Original segmentation (i.e.  $\sigma_{i,j}$ ). Middle: Output of the system in (3.28). Difference in frequency content of output values corresponding to the two different submodels can be inferred from the texture. Right: Resulting segmentation.

*Example 6:* In this example, we consider the identification of a 2-D system governed by the following linear switched-coefficient difference equation:

$$y(i, j) = a_{1,1}(\sigma_{i,j})y(i-1, j-1) + a_{0,1}(\sigma_{i,j})y(i, j-1) + a_{1,0}(\sigma_{i,j})y(i-1, j) + c_{0,0}(\sigma_{i,j})u(i, j) + \eta(i, j) \quad (3.28)$$

with

$$\sigma_{i,j} = \begin{cases} 1, & (i, j) \in \{(i, j) \in \mathbb{Z}^2 : (i-15)^2 + (j-30)^2 < 60\} \\ 2, & \text{otherwise} \end{cases}$$

where  $a_{1,1}(1) = 0.3$ ,  $a_{0,1}(1) = -0.3$ ,  $a_{1,0}(1) = -0.3$ ,  $a_{1,1}(2) = -0.5$ ,  $a_{0,1}(2) = 0.5$ ,  $a_{1,0}(2) = 0.7$  and  $c_{0,0}(1) = c_{0,0}(2) = 1$ . The boundary conditions and the input  $u$  are generated randomly from a zero-mean, unit-variance Gaussian distribution. Similarly, noise is uniformly randomly distributed in  $[-0.1, 0.1]$ . Input/output data on  $(i, j) \in [1, 30] \times [1, 40]$  is used for identification with minimum number of switches objective which in this case corresponds to shortest boundary between two different regions. Figure 3.4 shows the output of the system together with original and the resulting segmentation.

*Example 7:* This example illustrates the use of the proposed method to segment textured images. We generated two images, shown in Figure 3.5, by combining two different textures. The following 2D autonomous linear switched-coefficient difference equation was used to model the images:

$$I(x, y) = \sum_{(k_x, k_y) \in \mathcal{R}_a} a_{k_x, k_y}(\sigma_{x,y})I(x - k_x, y - k_y) + \eta(x, y) \quad (3.29)$$

where  $I(x, y)$  denotes the intensity at pixel location  $(x, y)$  and the support region  $\mathcal{R}_a$  was chosen according to fundamental period of the textures. The images together with the resulting segmentations are shown in figure 3.5. We used our algorithm to minimize the number of switches, which in this

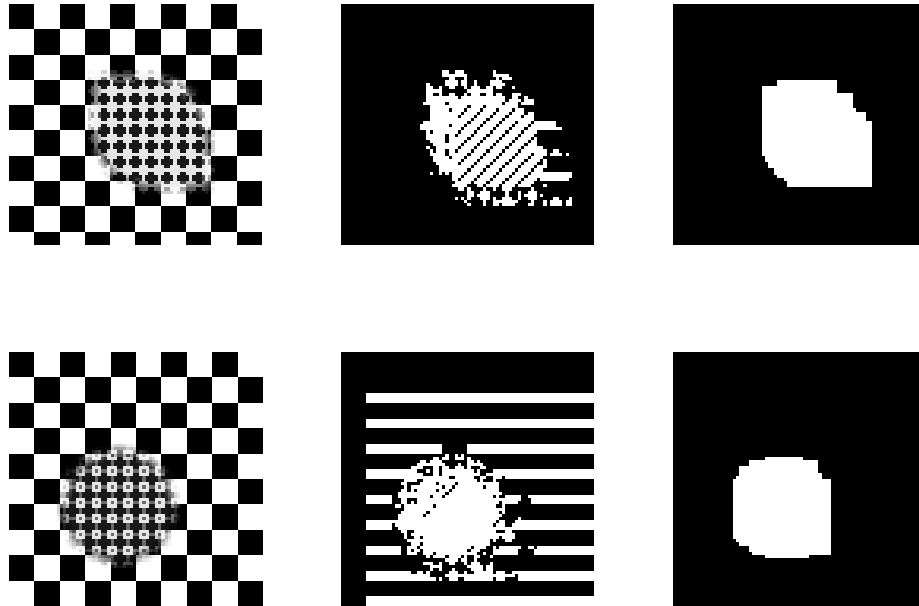


FIGURE 3.5: Results for detecting switches (i.e. estimating  $\hat{\sigma}_{i,j}$ ) in a texture image. Left: Original image. Middle: GPCA segmentation. Right: Segmentation via proposed method.

case corresponds to minimizing the length of the boundaries between regions; and compared the result against GPCA. Note that our model assumes continuity of the state through the region boundaries (i.e. there is no reset map) which is usually not the case with images. That's why the resulting segmentation does not have perfect boundaries. Nevertheless, the performance is better than that of GPCA as shown in figure 3.5.

### 3.4.3 Applications: Segmentation of Video Sequences.

In this section we illustrate the application of the proposed identification algorithm to two non-trivial problems arising in computer vision: segmentation of video-shots and dynamic textures. Here the goal is to detect changes, e.g. scenes or activities in the former, texture in the latter, in a sequence of frames. Given the high dimensionality of the data, the starting point is to perform a principal component analysis (PCA) compression [61] to obtain low dimensional feature vectors  $\mathbf{y}(t) \in \mathbb{R}^d$  representing each frame  $t$ . Specifically, we first convert each frame to gray scale. Next, we vectorize each frame of size  $N_x \times N_y$  and represent it with a vector  $\mathbf{f}(t) \in \mathbb{R}^{N_x N_y}$ . Then, we find the sample mean  $\mathbf{m} = \frac{1}{T-t_0+1} \sum_{t=t_0}^T \mathbf{f}(t)$  to form the mean subtracted data matrix  $\mathbf{F} = [\mathbf{f}(t_0) - \mathbf{m}, \dots, \mathbf{f}(T) - \mathbf{m}]$ .

|          | Sparsification | MPEG   | GPCA   | B2B    |
|----------|----------------|--------|--------|--------|
| mountain | 0.9965         | 0.9816 | 0.9263 | 0.5690 |
| family   | 0.9946         | 0.9480 | 0.8220 | 0.9078 |

TABLE 3.7: Rand indices for video-shot segmentation

Finally, performing a singular value decomposition on  $\mathbf{F} = \mathbf{U}\mathbf{D}\mathbf{V}^T$  and projecting the data using the first  $d$  columns of  $\mathbf{U}$  give the low dimensional representations  $[\mathbf{y}(t_0), \dots, \mathbf{y}(T)] = \mathbf{U}_{1:d}^T \mathbf{F}$ .

The next step is to assume, motivated by [8, 9, 13, 62], that each component  $y_j(\cdot)$  of the feature vector  $\mathbf{y}(t)$  evolves independently, according to an unknown model of the form<sup>6</sup>:

$$y_j(t) = \sum_{i=1}^{n_a} a_{i,j}(\sigma_t) y_j(t-1) + \eta(t), \quad \|\eta(t)\|_2 \leq \epsilon \quad (3.30)$$

Finally, defining  $\mathbf{g}(t) = [\mathbf{p}_1(t) - \mathbf{p}_1(t+1), \dots, \mathbf{p}_d(t) - \mathbf{p}_d(t+1)]$  allows to the (minimum number of switches) sparsification-based approach to segment a given sequence according to the non-zero elements in the corresponding sequence  $\|\mathbf{g}(\cdot)\|_\infty$ .

**Video-Shot Segmentation:** The goal here is to detect scene changes in video sequences. These changes can be categorized into two: (i) abrupt changes (cuts)<sup>7</sup>, and (ii) gradual transitions, e.g. various special effects that blend two consecutive scenes gradually. Figure 3.6 shows the ground truth and the segmentations obtained using the proposed method (using  $3^{rd}$  order models and  $d = 3$ ), GPCA [8], a histogram based method (bin to bin difference (B2B) with 256 bin histograms and window average thresholding [63]), and an MPEG-based method [64] for two sample sequences, *mountain.avi* and *family.avi*, available from <http://www.open-video.org>. Both the B2B and MPEG methods rely on user adjustable parameters (two in the B2B case, seven for MPEG). In our experiments we adjusted these parameters, by trial and error, to get the best possible results. Hence the resulting comparisons against the proposed sparsification method correspond to best-case scenarios for both MPEG and B2B. As shown in Table 3.7, the proposed method has slightly better performance than MPEG (the runner up), without the need to manually adjust seven parameters one of which, length of the transition, is very sensitive.

<sup>6</sup>This is a multioutput model as described in section 3.4.1.1 with diagonal coefficient matrices.

<sup>7</sup>Since we assume the continuity of the state, this might cause a few superfluous switches around cut frames. It is possible to overcome this problem by passing the resulting mode signal through a majority smoothing filter.

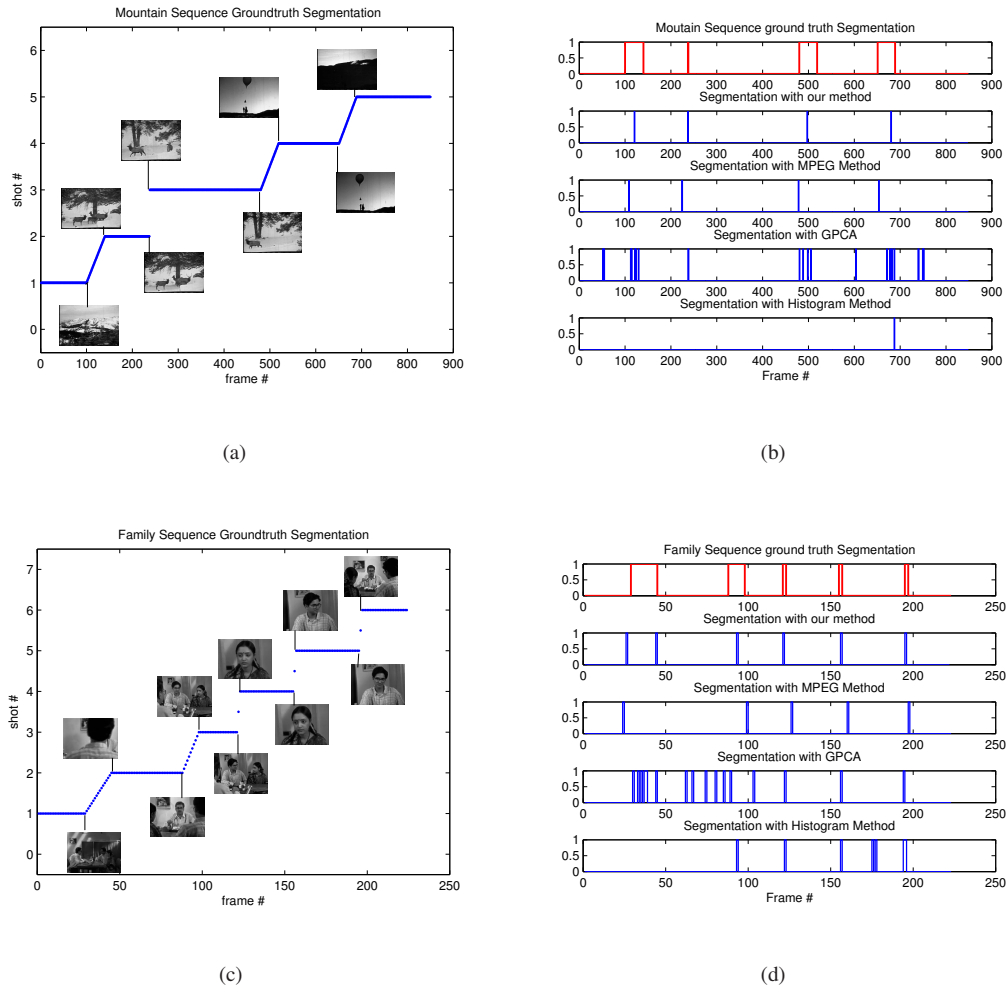


FIGURE 3.6: Video Segmentation Results. Left Column: Ground truth segmentation (jumps correspond to cuts and slanted lines correspond to gradual transitions). Right Column: Changes detected with different methods. Value 0 corresponds to frames within a segment and value 1 corresponds to the frames in transitions.

**Dynamic Textures:** Next, we consider two challenging sequences generated using the dynamic texture database <http://www.svcl.ucsd.edu/projects/motiondytex/synthdb/>. In the first one, we appended in time one patch from smoke to another patch from the same texture but transposed. Therefore, both sequences have the same photometric properties, but differ in the main motion direction: vertical in the first half and horizontal in the second half of the sequence. For the second example, we generated a sequence of river by sliding a window both in space and time (by going forward in time in the first half and by going backward in the second). Hence, the dynamics due to river flow are reversed. Sample frames from each sequence are shown in Figure 3.7. For these

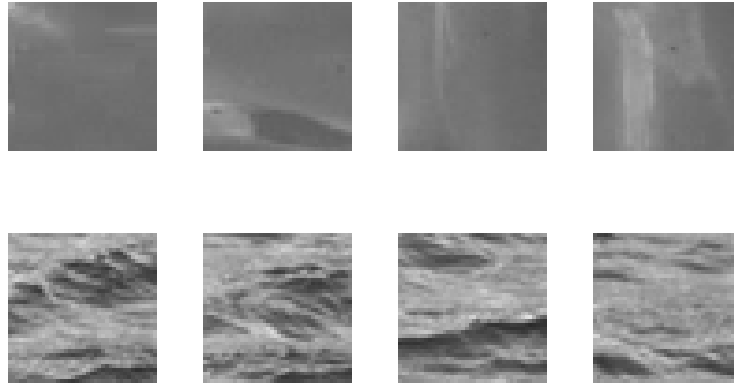


FIGURE 3.7: Sample dynamic texture patches. Top: smoke, Bottom: river

sequences both histogram and MPEG methods failed to detect the cut since the only change is in the dynamics. On the other hand, the proposed method (using 5<sup>th</sup> order models and  $d = 3$ ) correctly segmented both sequences. These results are summarized in Figure 3.8.

### 3.5 A Moments-Based Convex Optimization Approach

In this section we consider the problem of identifying hybrid systems from noisy input/output data when the number of subsystems is known. In the noise free case, Vidal *et al.* [47, 50] proposed a closed form solution to this problem. However, noise enters their formulation polynomially leading to a nonconvex problem. In this section, by using ideas from polynomial optimization we reformulate the problem in terms of probability distributions and moments. Our formulation, which is shown to be equivalent to the original non-convex problem, requires to solve a rank minimization subject to LMI constraints. Finally, we resort to a convex relaxation for dropping the rank to obtain a semidefinite programming problem that can be solved efficiently.

The problem of interest here can formally be stated as follows:

*Problem 4.* Given input/output data over the interval  $[1, T]$ , a bound on the  $\ell_\infty$  norm of the noise (i.e.  $\|\eta\|_\infty \leq \epsilon$ ), and the number of submodels  $s$ , find a hybrid linear model of the form (3.1) that is consistent with the a priori information and experimental data.



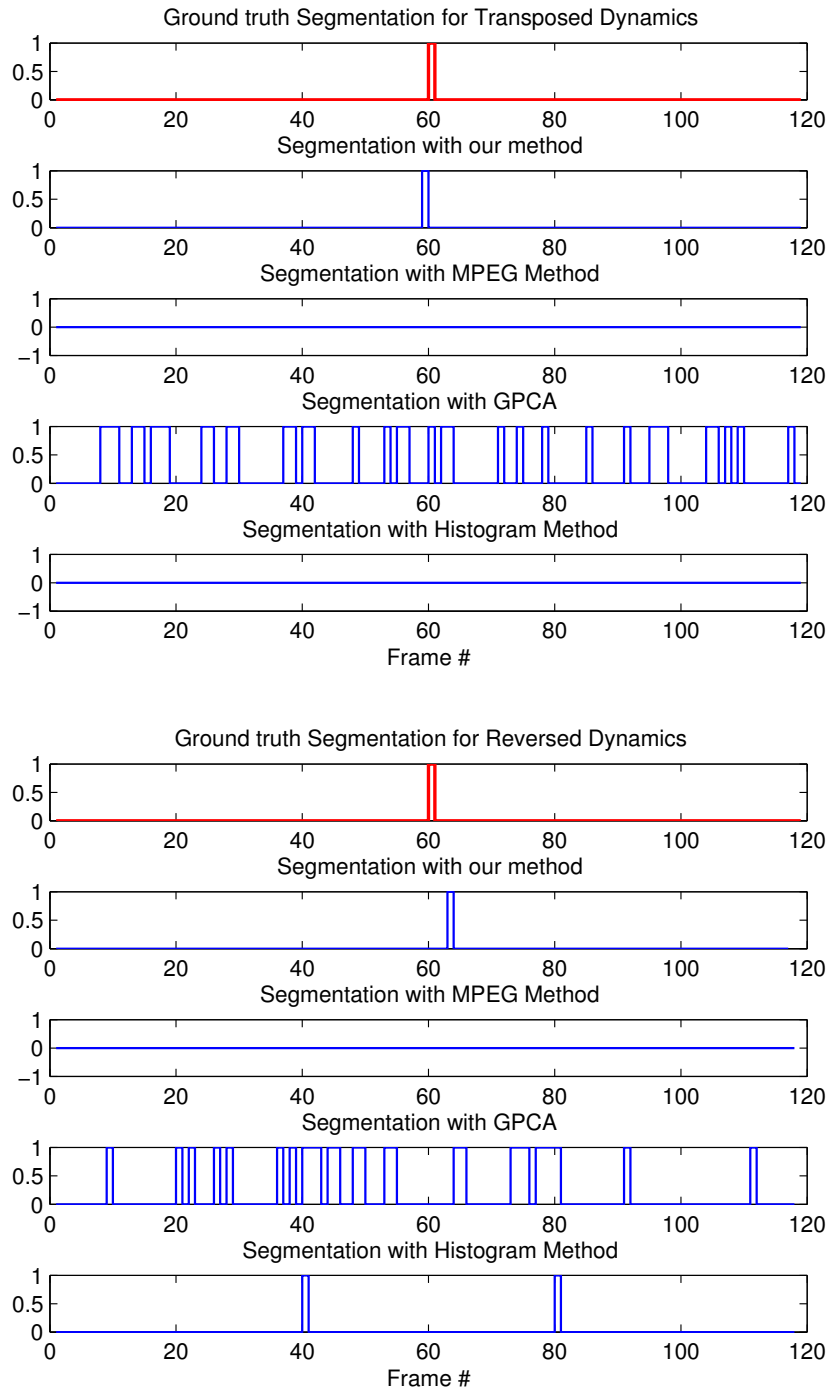


FIGURE 3.8: Results for detecting change in dynamics only. Top: Smoke sequence concatenated with transposed dynamics. Bottom: River sequence concatenated with reversed dynamics.

Practical situations where this problem is relevant arise for instance in segmentation problems in computer vision and medical image processing, where it is desired to segment an image or video clip into a given number of (not necessarily contiguous) regions, for instance corresponding to healthy versus diseased tissue, or a fixed number of activities.

In the noise free case (i.e.  $\eta_t = 0 \forall t$ ), the problem above can be elegantly solved using an algebraic procedure, Generalized Principal Component Analysis (GPCA), proposed by Vidal *et al.* [47, 50]. Note that in this case an equivalent representation of (3.1) is:

$$\mathbf{b}(\sigma_t)^T \mathbf{r}_t = 0 \quad (3.31)$$

where  $\mathbf{r}_t = [-y_t, y_{t-1}, \dots, y_{t-n_a}, u_{t-1}, \dots, u_{t-n_c}]^T$  and  $\mathbf{b}(\sigma_t) = [1, a_1(\sigma_t), \dots, a_{n_a}(\sigma_t), c_1(\sigma_t), \dots, c_{n_c}(\sigma_t)]^T$ , denote the regressor and (unknown) coefficients vectors at time  $t$ , respectively.

The idea behind the algebraic method is based on a polynomial constraint, the so-called *hybrid decoupling constraint*, that decouples the identification of model parameters from the identification of the discrete state and switching sequence. That is,

$$p_s(\mathbf{r}) = \prod_{i=1}^s (\mathbf{b}_i^T \mathbf{r}_t) = \mathbf{c}_s^T \nu_s(\mathbf{r}_t) = 0 \quad (3.32)$$

holds for all  $t$  independent of which of the  $s$  submodels is active at time  $t$ . In the above equality,  $\mathbf{b}_i \in \mathbb{R}^{n_a+n_c+1}$  is the parameter vector corresponding to the  $i^{th}$  submodel,  $\mathbf{r}_t$  is the known regressor vector at time  $t$ , and  $\nu_s(\cdot)$  is the Veronese map of degree  $s$ <sup>8</sup>. Collecting all data into a matrix form leads to:

$$\mathbf{V}_s \mathbf{c}_s \doteq \begin{bmatrix} \nu_s(\mathbf{r}_{t_0})^T \\ \vdots \\ \nu_s(\mathbf{r}_T)^T \end{bmatrix} \mathbf{c}_s = \mathbf{0} \quad (3.33)$$

Hence, one can solve for a vector  $\mathbf{c}_s$  in the nullspace of  $\mathbf{V}_s$ . Finally,  $\mathbf{b}_i$ , the parameters of the models can be computed via polynomial differentiation (see the Appendix B).

<sup>8</sup>Veronese map of degree  $s$ ,  $\nu_s: \mathbb{R}^n \rightarrow \mathbb{R}^m$ , is defined as:

$$\nu_s([x_1, \dots, x_n]^T) = [\dots, \xi^s, \dots]^T$$

where  $m = \binom{s+n-1}{s}$  and  $\xi^s \doteq x_1^{s_1} x_2^{s_2} \dots x_n^{s_n}$ ,  $\sum s_i = s$ , e.g. all possible monomials of order  $s$ , in lexicographical order.

### 3.5.1 Main Results

In the presence of noise, the approach outlined above breaks down, since conditions (3.32) and (3.33) no longer hold. Indeed, the noisy equivalent of (3.32) is given by:

$$p_s(\mathbf{r}, \eta) = \prod_{i=1}^s (\mathbf{b}_i^T \tilde{\mathbf{r}}_t) = \mathbf{c}_s^T \nu_s(\tilde{\mathbf{r}}_t) = 0 \quad (3.34)$$

where  $\tilde{\mathbf{r}}_t = [-y_t + \eta_t, y_{t-1}, \dots, y_{t-n_a}, u_{t-1}, \dots, u_{t-n_c}]^T$ . Proceeding as in the noiseless case, a “noisy” data matrix  $\mathbf{V}_s(\mathbf{r}, \eta)$  can be built. However, finding the coefficients of each subsystem entails now finding both an admissible noise sequence  $\eta^o$  and a vector  $\mathbf{c}^o$  in the nullspace of  $\mathbf{V}_s(\mathbf{r}, \eta^o)$  such that

$$\mathbf{V}_s(\mathbf{r}, \eta^o) \mathbf{c}^o = 0 \quad (3.35)$$

Since  $\mathbf{V}_s$  is a polynomial function of the unknown noise terms  $\eta(t)$ , this approach leads to a computationally very challenging nonlinear, nonconvex optimization problem. However, as we show in the sequel, by exploiting the method of the moments, (3.35) can be recast into a constrained rank minimization form which in turn can be relaxed to an efficient convex optimization. We recall some background results on polynomial optimization and problem of moments, which will be used to recast the identification problem into a convex optimization form, in Appendices 2.2.1 and 2.2.2.

#### 3.5.1.1 A Moments Based Convex Relaxation:

Consider the following rank minimization problem:

$$\begin{aligned} & \text{minimize}_{\eta_t} \quad \text{rank } \mathbf{V}_s(\mathbf{r}_t, \eta_t) \\ & \text{subject to} \quad \|\eta_t\|_\infty \leq \epsilon \end{aligned} \quad (3.36)$$

Clearly, Problem 4 is solvable if and only if (3.36) admits a rank deficient solution. Indeed, in the case where the order of each subsystem is precisely  $(n_a, n_c)$  and  $\mathbf{V}_s$  has generically full column rank (e.g. enough data points have been collected from each subsystem), then there exists a noise sequence  $\eta_t^o$  such that the right null space of  $\mathbf{V}_s(\mathbf{r}_t, \eta_t^o)$  has dimension 1. When some of the subsystems have order lower than  $(n_a, n_c)$  or the number of subsystems is overestimated, the dimension of the nullspace of  $\mathbf{V}_s$  is higher than 1 (see [47] for details). In that sense, minimizing the rank of  $\mathbf{V}_s$  amounts to searching for the simplest model that explains the data.

Since in this section we are interested in finding just one model consistent with the a-priori information (bounds on the  $\|\eta_t\|_\infty$ , the number of subsystems and their order), we will simply search for rank deficient solutions to (3.36). As we show next, this problem admits a computationally tractable relaxation.

Exploiting Theorem 1 and using the facts that (i)  $\eta_t$  and  $\eta_{\bar{t}}$  are independent for  $t \neq \bar{t}$ , and (ii)  $\eta_t$  only appears in the  $t^{\text{th}}$  row of  $\mathbf{V}_s$ , leads to the following moments optimization problem:

$$\begin{aligned} & \text{minimize}_{\mathbf{m}^{(t)}} \quad \text{rank } \tilde{\mathbf{V}}_s(\mathbf{r}_t, \mathbf{m}^{(t)}) \\ & \text{subject to} \quad (2.12) - (2.13) \quad \forall \mathbf{m}^{(t)} \quad \text{if } s \text{ is odd} \\ & \quad \quad \quad (2.14) - (2.15) \quad \forall \mathbf{m}^{(t)} \quad \text{if } s \text{ is even} \end{aligned} \quad (3.37)$$

where  $\mathbf{m}^{(t)} = [m_1^{(t)}, \dots, m_s^{(t)}]$  is the moment sequence corresponding to  $\eta_t$  and  $\tilde{\mathbf{V}}_s(\mathbf{r}_t, \mathbf{m}^{(t)})$  is a matrix linear in the moments, obtained by replacing each  $k^{\text{th}}$  degree monomial  $\eta_t^k$  in  $\mathbf{V}_s(\mathbf{r}_t, \eta_t)$  with the corresponding  $k^{\text{th}}$  order moment  $m_k^{(t)}$ .

*Example 1.* For instance when  $s = 2$  and  $(n_a, n_c) = (1, 1)$ , then  $\mathbf{r}_t = [-y_t, y_{t-1}, u_{t-1}]^T$  and the rows of  $\mathbf{V}_s(\mathbf{r}_t, \eta_t)$  depend on  $\mathbf{r}$  and  $\eta$  as follows:

$$\nu_2(\mathbf{r}_t, \eta_t)^T = \begin{bmatrix} y_t^2 - 2y_t\eta_t + \eta_t^2 \\ -y_t y_{t-1} + y_{t-1}\eta_t \\ -y_t u_{t-1} + u_{t-1}\eta_t \\ y_{t-1}^2 \\ y_{t-1}u_{t-1} \\ u_{t-1}^2 \end{bmatrix}^T. \quad (3.38)$$

The corresponding row of  $\tilde{\mathbf{V}}_s(\mathbf{r}_t, \mathbf{m}^{(t)})$  is given by:

$$\mathbf{E}_\mu [\nu_2(\mathbf{r}_t, \eta_t)^T] = \begin{bmatrix} y_t^2 - 2y_t m_1^{(t)} + m_2^{(t)} \\ -y_t y_{t-1} + y_{t-1} m_1^{(t)} \\ -y_t u_{t-1} + u_{t-1} m_1^{(t)} \\ y_{t-1}^2 \\ y_{t-1} u_{t-1} \\ u_{t-1}^2 \end{bmatrix}^T \quad (3.39)$$

Thus,  $\tilde{\mathbf{V}}_s(\mathbf{r}_t, \mathbf{m}^{(t)})$  is affine in the unknown moments.

Note that Theorem 3 cannot be directly applied to show the equivalence of (3.36) and (3.37) since rank is not a polynomial function. Nevertheless, as we show next, a result similar to Theorem 3 can be obtained:

*Theorem 8.* There exists a rank deficient solution to problem (3.36) if and only if there exists a rank deficient solution to problem (3.37). Moreover, if  $\mathbf{c}$  belongs to the nullspace of the solution of (3.37) then there exists a noise value  $\eta^*$  with  $\|\eta^*\|_\infty \leq \epsilon$  such that  $\mathbf{c}$  belongs to the nullspace of  $\mathbf{V}_s(\mathbf{r}_t, \eta^*)$ .

*Proof.* Assume that the minimum rank  $r_1$  in (3.36) is achieved by some sequence  $\eta_t^*$ . Then  $\tilde{\mathbf{V}}_s(\mathbf{r}_t, \mathbf{m}^{*(t)})$  with  $\mathbf{m}^{*(t)} = [\eta_t^*, (\eta_t^*)^2, \dots, (\eta_t^*)^s]$  (i.e. all distributions have point support) has rank  $r_1$  and  $\mathbf{m}^{*(t)}$  satisfies the LMI constraints. Hence the minimum rank obtained by solving (3.37) is less than or equal to the minimum rank obtained by solving (3.36).

Consider now an optimal solution  $\mathbf{m}^{*(t)}$  of (3.37). Note that, from Theorem 1, this guarantees the existence of  $T$  measures  $\mu^{*(t)}$ , each supported on  $[-\epsilon, \epsilon]$ . Let  $\mathbf{c}$  be in the nullspace of  $\tilde{\mathbf{V}}_s(\mathbf{r}_t, \mathbf{m}^{*(t)})$  (i.e.  $\tilde{\mathbf{V}}_s(\mathbf{r}_t, \mathbf{m}^{*(t)})\mathbf{c} = \mathbf{0}$ ). Thus, for each row of  $\mathbf{V}_s$ ,  $\mathbf{E}_{\mu^{*(t)}}[\nu_s(\mathbf{r}_t, \eta_t)]\mathbf{c} = \mathbf{E}_{\mu^{*(t)}}[\nu_s(\mathbf{r}_t, \eta_t)\mathbf{c}] = 0$ . By noting that  $\nu_s(\mathbf{r}_t, \eta_t)\mathbf{c}$  is a polynomial function of  $\eta_t$  (hence continuous) and  $\mu^{*(t)}$  is supported on  $[-\epsilon, \epsilon]$ , we can invoke the mean value theorem for integration to conclude that there exist  $\eta_t^* \in [-\epsilon, \epsilon]$  for all  $t$  such that  $\nu_s(\mathbf{r}_t, \eta_t^*)\mathbf{c} = 0$ .

Thus, whenever the nullspace of the solution of (3.36) is non-trivial, so is that of (3.37), which proves the theorem.  $\square$

An alternative way of obtaining an equivalent moment-based problem to (3.36) when  $\mathbf{V}_s(\mathbf{r}_t, \eta_t)$  is known to be rank deficient by one is to define the equivalent polynomial objective function  $\det[\mathbf{V}_s^T(\mathbf{r}_t, \eta_t)\mathbf{V}_s(\mathbf{r}_t, \eta_t)]$ . However, in this case one would need higher order (possibly infinite number of) moments of a multidimensional distribution since it is not clear how to exploit the independence of noise terms while keeping the problem linear in moments. Although the rank objective is non-convex, it has following advantages: (i) the equivalent moment based problem is finite dimensional (i.e. it requires only finite moment matrices); (ii) there are efficient convex relaxations for rank minimization; and (iii) extracting solutions requires only solving for the roots of a polynomial in one variable whereas for the case with multidimensional distributions this is a non-trivial task. We elaborate on the last two points next.

Although rank minimization is an NP-Hard problem, efficient convex relaxations are available. In particular, good approximate solutions can be obtained by using a log-det heuristic [51] that relaxes

rank minimization to a sequence of convex problems. Furthermore, since from a set membership point of view it suffices to find a rank deficient solution, we propose a modification of log–det heuristic that aims at dropping the rank by one. The algorithm, which is inspired by the adaptive step size defined for weighted  $\ell_1$  minimization in [65], is summarized next:

---

**Algorithm for dropping the rank**

---

$\mathbf{X} \in \mathbb{R}^{m \times n}$  and assuming wlog  $m \leq n$ , initialize:

$$k = 0$$

$$\mathbf{W}_y^{(0)} = \mathbf{I}_{m \times m}$$

$$\mathbf{W}_z^{(0)} = \mathbf{I}_{n \times n}$$

REPEAT

Solve

$$\begin{aligned} \min_{\mathbf{X}^{(k)}, \mathbf{Y}^{(k)}, \mathbf{Z}^{(k)}} \quad & \text{trace} \begin{bmatrix} \mathbf{W}_y^{(k)} \mathbf{Y}^{(k)} & 0 \\ 0 & \mathbf{W}_z^{(k)} \mathbf{Z}^{(k)} \end{bmatrix} \\ \text{subject to} \quad & \begin{bmatrix} \mathbf{Y}^{(k)} & \mathbf{X}^{(k)} \\ \mathbf{X}^{T(k)} & \mathbf{Z}^{(k)} \end{bmatrix} \succeq 0 \\ & \mathbf{X}^{(k)} \in \mathcal{C} \end{aligned}$$

Decompose  $\mathbf{X}^{(k)} = \mathbf{U} \mathbf{D} \mathbf{V}^T$  using SVD.

Set  $\epsilon = \mathbf{D}(m, m)$ .

Set  $\mathbf{W}_y^{(k+1)} = (\mathbf{Y}^{(k)} + \epsilon \mathbf{I})^{-1}$ .

Set  $\mathbf{W}_z^{(k+1)} = (\mathbf{Z}^{(k)} + \epsilon \mathbf{I})^{-1}$ .

Set  $k = k + 1$ .

UNTIL (a convergence criterion is reached)

RETURN  $\mathbf{X}^{(k)}$

---

Above, for the sake of notational simplicity, we used  $\mathbf{X} = \tilde{\mathbf{V}}_s(\mathbf{r}_t, \mathbf{m}^{(t)})$ ; and  $\mathbf{X}^{(k)} \in \mathcal{C}$  stands for convex constraints, that is,  $\mathbf{m}^{(t)}$  lies on a convex set  $\mathcal{C}$  defined by LMIs in (3.37).

Assuming a rank deficient  $\tilde{\mathbf{V}}_s(\mathbf{r}_t, \mathbf{m}^{(t)})$  is found, a vector  $\mathbf{c}$  in its nullspace can be found by simply performing a singular value decomposition. From Theorem 8, it follows that  $\mathbf{c}$  is also in the nullspace of  $\mathbf{V}_s(\mathbf{r}_t, \eta_t)$  (i.e.  $\mathbf{V}_s(\mathbf{r}_t, \eta_t)(\mathbf{c}) = 0$ ). Hence, for each row, we have  $\nu_s(\mathbf{r}_t, \eta_t^*) \mathbf{c} = 0$  which is a polynomial equation in one variable. One can solve for the noise values by finding the roots of this polynomial that lie in  $[-\epsilon, \epsilon]$  (which is guaranteed to exist again by Theorem 8). Once the noise values are estimated, the problem can be converted to the noise free case by plugging the noise estimates into  $\mathbf{V}_s(\mathbf{r}_t, \eta_t)$  and the system parameters can be computed using the GPCA procedure.

*Remark 7.* When the number of the submodels  $s$  is unknown, it is possible to search for minimum number of submodels that explains the data. This can be accomplished with a simple iteration on  $s$ ; starting with  $s = 1$  and increasing  $s$  up until a rank deficient solution to the problem (3.37) is found.

### 3.5.2 Illustrative Examples

In this section we use both an academic and a practical example to illustrate the effectiveness of the proposed method.

#### 3.5.2.1 Academic Example:

Consider a hybrid system that switches among the following three ARX subsystems

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

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

$$y_t = 1.7y_{t-1} - 0.72y_{t-2} + 0.5u_{t-1} + \eta_t \quad (\text{Submodel 3})$$

modeled as

$$y_t = p_1(\sigma_t)y_{t-1} + p_2(\sigma_t)y_{t-2} + p_3(\sigma_t)u_{t-1} + \eta_t. \quad (3.40)$$

where  $\sigma_t \in \{1, 2, 3\}$  depending on which model is active at time  $t$ . Experimental data was obtained by running a simulation for  $T = 96$  time steps with  $\|\eta\|_\infty = 0.25$  where  $\sigma_t = 1$  for  $t = [1, 32]$ ,  $\sigma_t = 2$  for  $t = [33, 64]$  and  $\sigma_t = 3$  for  $t = [65, 96]$ . The parameter values used for the simulation are shown in Table 3.8 together with the results obtained by our robust version of the algebraic method and the original algebraic method of [50]. Figures 3.9 and 3.10 show the clustering of data into different submodels. As seen there, the proposed method outperforms the method in [50]. Figures 3.11 and 3.12 show the absolute error given the identified model. The error values are quite large for the method in [50] whereas they mostly satisfy the prior bound of  $\|\eta\|_\infty = 0.25$  for the new method. Indeed, in the latter case, the error exceeds the bound only at a single time instant. This is due to the convex relaxation used to solve the original algebraic problem.

|            |       | True    | Moments-based | GPCA    |
|------------|-------|---------|---------------|---------|
| Submodel 1 | $p_1$ | 0.2000  | 0.1964        | 0.2248  |
|            | $p_2$ | 0.2400  | 0.2332        | 0.3764  |
|            | $p_3$ | 2.0000  | 1.9287        | 2.5907  |
| Submodel 2 | $p_1$ | -1.4000 | -1.2959       | -0.4491 |
|            | $p_2$ | -0.5300 | -0.4469       | 0.9188  |
|            | $p_3$ | 1.0000  | 1.0315        | 1.5262  |
| Submodel 3 | $p_1$ | 1.7000  | 1.6505        | 1.7213  |
|            | $p_2$ | -0.7200 | -0.6713       | -0.7103 |
|            | $p_3$ | 0.5000  | 0.5007        | 1.2194  |

TABLE 3.8: Estimated and true values of parameters

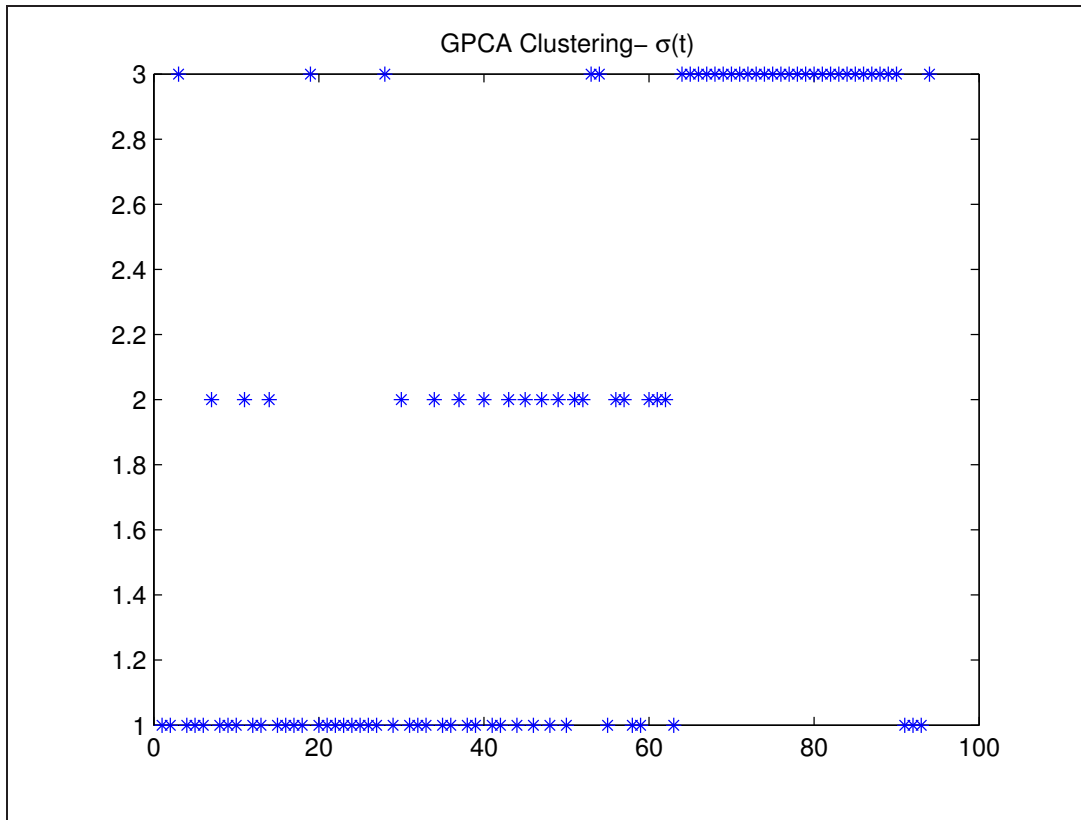


FIGURE 3.9: Clustering via GPCA.

### 3.5.2.2 A Practical Example: Human Activity Segmentation:

Next, we illustrate an application of the proposed method in a computer vision problem: human activity analysis. The input data consists of 55 frames extracted from a video sequence of a person walking and bending in front of the camera. Figure 3.13 shows some sample frames from the sequence. In the



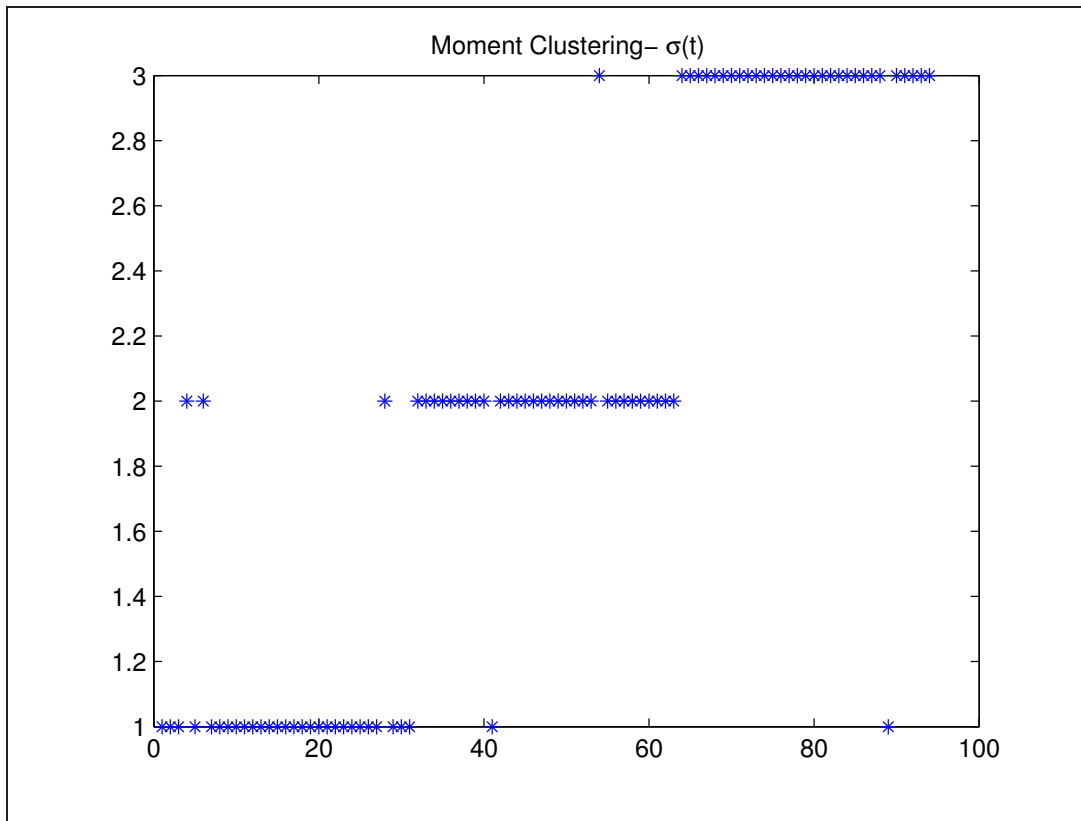


FIGURE 3.10: Clustering via moments-based method.

middle of the sequence the person bends down, then stands up and keeps walking. In order to recast the problem into the identification of a piecewise affine system, we first used simple background subtraction to estimate the location of the center of mass of the person in each frame. The horizontal<sup>9</sup> position of the center of mass was then modeled as the output of a first order switched affine autoregressive system:

$$x_t = a(\sigma_t)x_{t-1} + f(\sigma_t) + \eta_t \quad (3.41)$$

where  $a(\sigma_t)$  and  $f(\sigma_t)$  are unknown parameters. We set  $\|\eta_t\|_\infty = 3$ , allowing  $\pm 3$  pixels error in the position estimates.

Figure 3.15 shows that the proposed method is capable of segmenting normal activity (walking) from abnormal activity (bending). It also correctly classifies the activity at the initial and final portions of the video as the same. There is a single frame at the end of the sequence where the classification is incorrect. This is due to the fact that, as the person starts to leave the camera's field of view, the

<sup>9</sup>It may seem more natural to use the vertical position. However, this would have resulted in 3 segments, corresponding to roughly no vertical motion, downward and upward motion, while there are only two different activities involved.

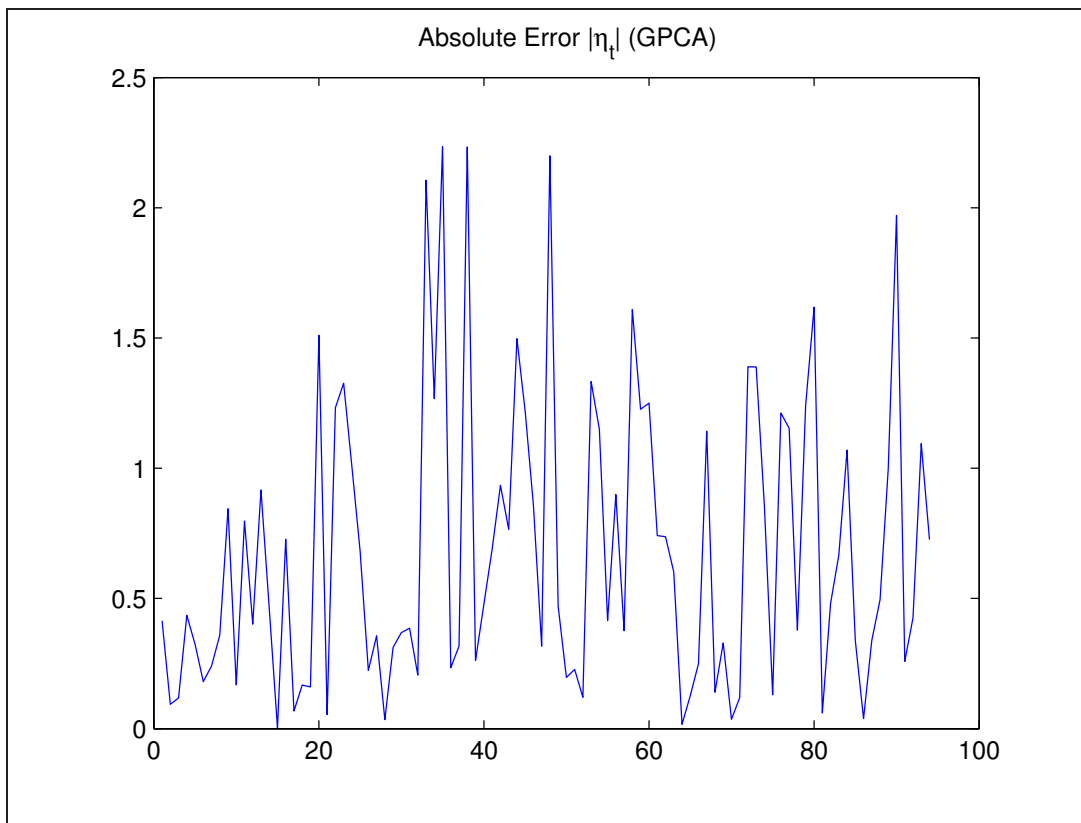


FIGURE 3.11: Absolute error for GPCA identification.

estimates of her center of mass become inaccurate. On the other hand, the classification with the original GPCA is less accurate as can be seen from figure 3.14.

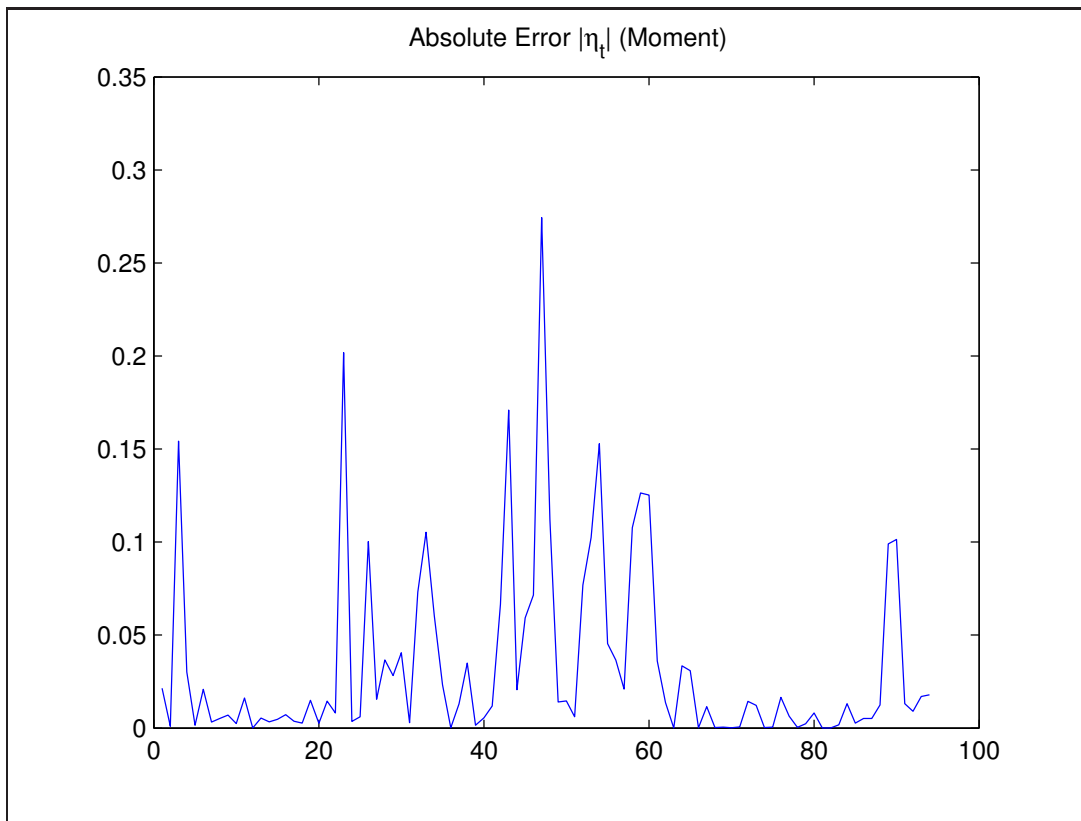


FIGURE 3.12: Absolute error for moments-based identification.

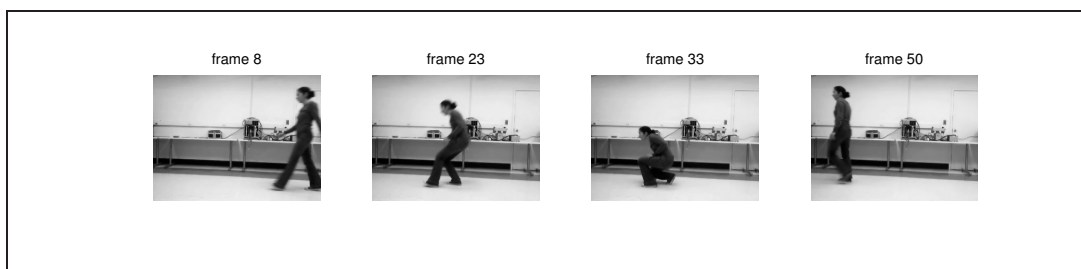


FIGURE 3.13: Sample frames from the video.

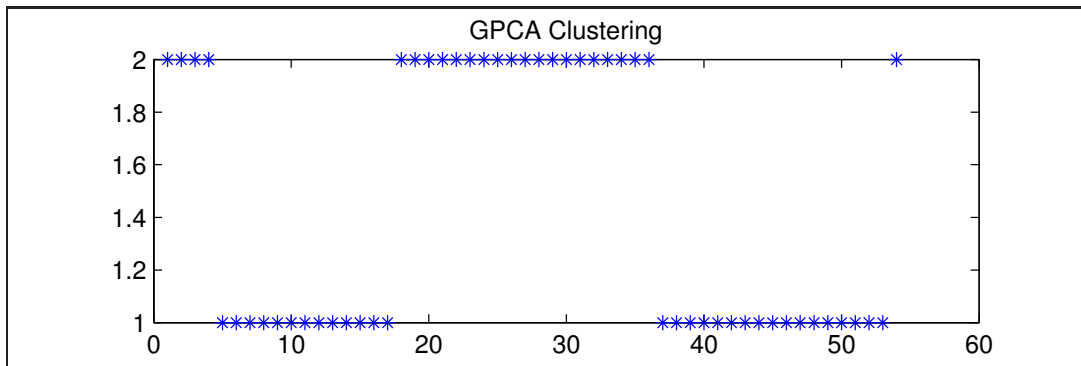


FIGURE 3.14: Activity segmentation via GPCA.

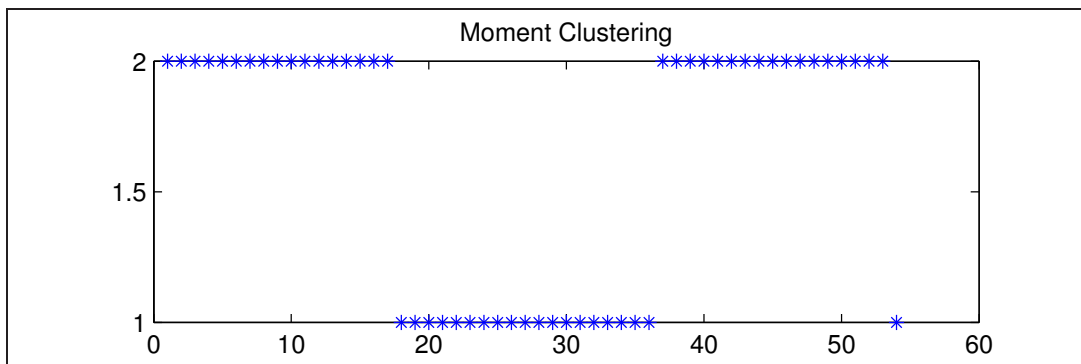


FIGURE 3.15: Activity segmentation via moments-based method.

## Chapter 4

# Model (In)validation for a Class of Hybrid Dynamical Systems

### 4.1 Introduction and Motivation

As mentioned in the previous Chapter, hybrid systems, dynamical systems where continuous and discrete states interact, are ubiquitous in many different contexts. Thus, during the past few years, a considerable research effort has been devoted to the problem of identifying hybrid systems, leading to several methods (see the excellent tutorial paper [46] for a summary of the main issues and recent developments in the field). Since the identification problem is generically NP-Hard, the majority of existing identification algorithms, including the ones presented in Chapter 3, are based on heuristics or relaxations ([48, 66, 67, 68, 69]). Hence, a crucial step before using the resulting models, is to check their validity against additional experimental data. Model (in)validation of Linear Time Invariant (LTI) systems has been extensively addressed in the past two decades and both time and frequency domain results are available in the literature (see for instance [70, 71, 72] and references therein). A related line of research is model (in)validation of Linear Parameter Varying (LPV) systems ([73, 74]) where it is assumed that parameter values are measurable during the experiment and used as part of *a posteriori* data during the (in)validation step. Finally, (in)validation of continuous-time nonlinear models was addressed in [75] using sum of squares methods and barrier functions. However, to the best of our knowledge, similar results for discrete-time switched linear systems with unknown switches has not been considered, with the main difficulty here being the combination of noisy measurements and an unknown mode signal.

The main result of this chapter is a necessary and sufficient condition for a multi-input multi-output switched affine autoregressive exogenous model to be (in)validated by the experimental data. Specifically, given a nominal model and experimental input/output data, we provide certificates for the existence/nonexistence of a switching sequence such that the resulting output sequence interpolates the given experimental data within a given noise bound. The starting point to obtain such certificates is to recast the (in)validation problem as one of checking whether a semialgebraic set is empty. By using a combination of recent results on moments-based sparse polynomial optimization and duality we show that emptiness of this set is equivalent to strict positivity of the solution of a related, convex optimization problem. In principle, checking this condition requires solving a sequence of convex optimization problems involving increasingly large matrices. However, as we show in this chapter, if in the process of solving these problems either a positive solution is found or the rank of certain matrices formed using the solution to the dual problem ceases to increase (the so-called flat extension property), then the process terminates with either an invalidation or a consistency certificate. A salient feature of the proposed approach is its ability to exploit the inherently sparse structure of the optimization problem to substantially reduce its computational complexity.

In the second portion of the chapter, these results are illustrated both with academic examples and a non-trivial problem arising in computer vision: activity monitoring. Typically, a visual surveillance system captures high volume data streams from multiple cameras. However, interesting (e.g. abnormal) activities are rare. Thus, it is important to be able to automatically eliminate the “normal” behavior and trigger an appropriate response when something potentially interesting or “abnormal” occurs. As we show in Section 4.4.2, this problem can be recast into a piecewise-affine model invalidation form and solved using the framework developed in this chapter.

## 4.2 (In)validating MIMO SARX Models

In this section we formally state the problem under consideration and show that it can be reduced to a polynomial optimization over a semialgebraic set. In turn, this allows for exploiting the results briefly discussed in section 2.2.1 to obtain computationally tractable (in)validation certificates.

### 4.2.1 Problem Statement

In this chapter, we consider multi-input, multi-output (MIMO) switched affine autoregressive exogenous (SARX) models of the form:

$$\begin{aligned} \mathbf{y}_t &= \sum_{k=1}^{n_a} \mathbf{A}_k(\sigma_t) \mathbf{y}_{t-k} \\ &\quad + \sum_{k=1}^{n_c} \mathbf{C}_k(\sigma_t) \mathbf{u}_{t-k} + \mathbf{f}(\sigma_t) \\ \tilde{\mathbf{y}}_t &= \mathbf{y}_t + \boldsymbol{\eta}_t \end{aligned} \quad (4.1)$$

where  $\mathbf{u}_t \in \mathbb{R}^{n_u}$  is the input,  $\tilde{\mathbf{y}}_t \in \mathbb{R}^{n_y}$  is the measured output corrupted by the noise  $\boldsymbol{\eta}_t \in \mathbb{R}^{n_y}$ , and  $\sigma_t \in \mathbb{N}_s$  is the discrete mode signal indicating which of the  $s$  submodels is active at time  $t$ . We do not make any dwell-time assumptions, hence the mode signal  $\sigma_t$  can switch arbitrarily among the  $s$  submodels  $G_i$ , each of which is associated with the set of its coefficient matrices  $\{\mathbf{A}_1(i), \dots, \mathbf{A}_{n_a}(i), \mathbf{C}_1(i), \dots, \mathbf{C}_{n_c}(i), \mathbf{f}(i)\}$ .

The model (in)validation problem for the setup described above and shown in Figure 4.1 can be formally stated as follows:

*Problem 5.* Given a nominal hybrid model of the form (4.1) together with its  $s$  submodels  $G_1, \dots, G_s$ , an *a priori* bound  $\epsilon$  on noise, and experimental data  $\{\mathbf{u}_t, \tilde{\mathbf{y}}_t\}_{t=t_0}^T$ , determine whether or not the *a priori* information and the *a posteriori* experimental data are consistent, i.e. whether the consistency set

$$\begin{aligned} \mathcal{T}(\boldsymbol{\eta}, \sigma) &= \{ \|\boldsymbol{\eta}_t\|_\infty \leq \epsilon, \sigma_t \in \mathbb{N}_s \\ &\quad \text{subject to (4.1)} \quad \forall t \in [t_0, T] \} \end{aligned}$$

is nonempty.

Since  $\mathcal{T}(\boldsymbol{\eta}, \sigma)$  contains all possible noise and mode signal sequences that can explain the observed data, clearly establishing that  $\mathcal{T}(\boldsymbol{\eta}, \sigma) = \emptyset$  is equivalent to invalidating the model.

### 4.2.2 A Convex Certificate for (In)validating MIMO SARX Models

Next, we present the main result of the chapter, showing that a necessary and sufficient condition for the model to be invalidated by the experimental data is strict positivity of the solution to a related convex optimization problem. We begin by constructing a semialgebraic set  $\mathcal{T}'(\boldsymbol{\eta})$  and showing its equivalence, in a sense to become clear later, to the consistency set  $\mathcal{T}(\boldsymbol{\eta}, \sigma)$ . To this effect, assume

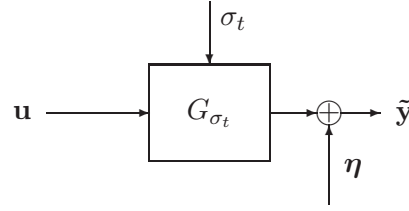


FIGURE 4.1: Problem Setup. The coefficient matrices of the submodels  $G_i$  and a bound on the noise are known *a priori*. The experimental data consists of input/output measurements,  $\mathbf{u}$  and  $\tilde{\mathbf{y}}$ . The mode signal  $\sigma_t$  and noise sequence  $\eta$  are unknown.

that the  $i^{th}$  submodel is active at time  $t$ . Rearranging the terms in Eq. (4.1) yields:

$$\begin{aligned} & \mathbf{A}_1(i)(\tilde{\mathbf{y}}_{t-1} - \boldsymbol{\eta}_{t-1}) + \dots + \mathbf{A}_{n_a}(i)(\tilde{\mathbf{y}}_{t-n_a} - \boldsymbol{\eta}_{t-n_a}) \\ & - (\tilde{\mathbf{y}}_t - \boldsymbol{\eta}_t) + \mathbf{C}_1(i)\mathbf{u}_{t-1} + \dots + \mathbf{C}_{n_c}(i)\mathbf{u}_{t-n_c} + \mathbf{f}(i) = \mathbf{0} \end{aligned} \quad (4.2)$$

which consists of  $n_y$  linear equations in  $n_a + 1$  unknown noise vectors  $\boldsymbol{\eta}_{t:t-n_a}$ . Denote the equation corresponding to  $j^{th}$  output by  $h_{t,i}^{(j)}(\boldsymbol{\eta}_{t:t-n_a})$ . Then Eq. (4.2) is equivalent to

$$[h_{t,i}^{(1)}(\boldsymbol{\eta}_{t:t-n_a}) = 0] \wedge \dots \wedge [h_{t,i}^{(n_y)}(\boldsymbol{\eta}_{t:t-n_a}) = 0] \quad (4.3)$$

or algebraically

$$g_{t,i}(\boldsymbol{\eta}_{t:t-n_a}) \doteq \sum_{j=1}^{n_y} [h_{t,i}^{(j)}(\boldsymbol{\eta}_{t:t-n_a})]^2 = 0. \quad (4.4)$$

Note that since the mode signal  $\sigma_t$  is unmeasurable, the actual subsystem  $G_i$  that is active at any given time  $t$  is not known. However, in order for the set of submodels given as part of *a priori* information not to be invalidated by the experimental data, Eq. (4.4) should hold true for some  $i \in \{1, \dots, s\}$ <sup>1</sup>. This condition can be expressed as

$$[g_{t,1}(\boldsymbol{\eta}_{t:t-n_a}) = 0] \vee \dots \vee [g_{t,s}(\boldsymbol{\eta}_{t:t-n_a}) = 0] \quad (4.5)$$

or algebraically

$$p_t(\boldsymbol{\eta}_{t:t-n_a}) \doteq \prod_{i=1}^s g_{t,i}(\boldsymbol{\eta}_{t:t-n_a}) = 0. \quad (4.6)$$

<sup>1</sup>This idea is similar to the hybrid decoupling constraint proposed in [47]



Next we use the equalities above to define a semialgebraic set and establish its relationship with the consistency set  $\mathcal{T}$ :

*Proposition 1.* Let

$$\mathcal{T}'(\boldsymbol{\eta}) \doteq \left\{ \boldsymbol{\eta} \mid f_{t,j}(\eta_t^{(j)}) \geq 0 \ \forall t \in [0, T], j \in \mathbb{N}_{n_y} \text{ and } p_t(\boldsymbol{\eta}_{t:t-n_a}) = 0 \ \forall t \in [n_a, T] \right\}.$$

where

$$f_{t,j}(\eta_t^{(j)}) \doteq \epsilon^2 - \left[ \eta_t^{(j)} \right]^2$$

Then,  $\mathcal{T}(\boldsymbol{\eta}, \sigma)$  is empty if and only if  $\mathcal{T}'(\boldsymbol{\eta})$  is empty.

*Proof.* Follows from the construction of  $\mathcal{T}'(\boldsymbol{\eta})$  and the fact that  $|\eta_t^{(j)}| \leq \epsilon \iff f_{t,j}(\eta_t^{(j)}) \geq 0 \quad \square$

At this point, one can use the *Positivstellensatz* and derive sum of squares certificates for the emptiness of  $\mathcal{T}'(\boldsymbol{\eta})$ , as proposed in [76]. However, as we show next, adopting a dual approach based on the theory of moments allows for exploiting the inherently sparse structure of the problem. In order to pursue this approach, start by considering the following optimization problem:

$$\begin{aligned} o^* &= \min_{\boldsymbol{\eta}} \sum_{t=n_a}^T p_t(\boldsymbol{\eta}_{t:t-n_a}) \\ &\text{s.t.} \\ &f_{t,j}(\eta_t^{(j)}) \geq 0 \ \forall t \in [0, T], j \in \mathbb{N}_{n_y}. \end{aligned} \quad (4.7)$$

Note that  $o^* \geq 0$ , since the objective function in (4.7) is a sum of squares polynomial<sup>2</sup>. Further, if  $\mathcal{T}'(\boldsymbol{\eta})$  is non-empty, then there exist a noise sequence  $\boldsymbol{\eta}^*$  for which (4.7) attains its minimum  $o^* = 0$ . Equivalently,  $o^* > 0 \iff \mathcal{T}'(\boldsymbol{\eta}) = \emptyset$ .

*Proposition 2.* Problem (4.7) above satisfies the running intersection property.

*Proof.* Consider the  $T - n_a + 1$  subsets  $I_k$  of the variables  $\boldsymbol{\eta}_{0:T}$  where each  $I_k$  contains only the variables  $\boldsymbol{\eta}_{k:k+n_a}$ . One can associate each  $f_{t,j}$  with  $I_0$  for  $t \leq n_a$  and with  $I_{t-n_a}$  for  $t > n_a$ . The collection  $I_{k,j}$  formed by repeating each  $I_k$ ,  $n_y$  times, satisfies (2.27) hence the running intersection property in Definition 1 holds.  $\square$

<sup>2</sup>since it is formed by multiplication and addition of SOS polynomials in (4.4), and the cone of SOS polynomials is closed under these operations.

Next, we use results from the theory of moments to obtain a convex problem where the objective and constraints have affine (rather than polynomial) dependence on the data and exploits sparsity. Consider the related moments-based optimization:

$$\begin{aligned}
d_N^* &= \min_{\mathbf{m}} \sum_{t=n_a}^T l_t(\mathbf{m}_{t-n_a:t}) \\
&\text{s.t.} \\
&\mathbf{M}_N(\mathbf{m}_{t-n_a:t}) \succeq 0 \quad \forall t \in [n_a, T] \\
&\mathbf{L}_N(f_{t,j} \mathbf{m}_{t-n_a:t}) \succeq 0 \quad \forall t \in [n_a + 1, T], j \in \mathbb{N}_{n_y} \\
&\mathbf{L}_N(f_{t,j} \mathbf{m}_{0:n_a}) \succeq 0 \quad \forall t \in [0, n_a], j \in \mathbb{N}_{n_y}
\end{aligned} \tag{4.8}$$

where each  $l_t$  is the linear functional of moments defined as  $l_t(\mathbf{m}_{t-n_a:t}) \doteq \mathbf{E} \{p_t(\boldsymbol{\eta}_{t:t-n_a})\}$ ,  $\mathbf{E}$  denotes expectation and where  $\mathbf{M}_N$  and  $\mathbf{L}_N$  are the moments and localization matrices associated with a truncated moments sequence containing terms up to order  $2N$  with  $N \geq s$ . Clearly  $d_N^* \geq 0$ , since the objective function in (4.7) is a sum of squares polynomial, and, from Theorem 4,  $d_N^* \uparrow o^*$  as  $N \rightarrow \infty$ . These observations lead to the following necessary and sufficient conditions for (in)validation:

*Proposition 3.* The following statements are equivalent:

- (i) The consistency set  $\mathcal{T}'(\boldsymbol{\eta})$  is empty
- (ii) There exists some finite  $N_o$  such that  $d_{N_o}^* > 0$
- (iii) The solution  $r^*$  to the following optimization problem is strictly greater than zero:

$$\begin{aligned}
r^* &= \min_{\mathbf{m}} \sum_{t=n_a}^T l_t(\mathbf{m}_{t-n_a:t}) \\
&\text{s.t.} \\
&\mathbf{M}_s(\mathbf{m}_{t-n_a:t}) \succeq 0 \quad \forall t \in [n_a, T] \\
&\text{rank} [\mathbf{M}_s(\mathbf{m}_{t-n_a:t})] = 1 \quad \forall t \in [n_a, T] \\
&\mathbf{L}_s(f_{t,j} \mathbf{m}_{t-n_a:t}) \succeq 0 \quad \forall t \in [n_a + 1, T], j \in \mathbb{N}_{n_y} \\
&\mathbf{L}_s(f_{t,j} \mathbf{m}_{0:n_a}) \succeq 0 \quad \forall t \in [0, n_a], j \in \mathbb{N}_{n_y}
\end{aligned} \tag{4.9}$$

where each of the  $T - n_a + 1$  moments sequences  $\mathbf{m}_{t-n_a:t}$ ,  $t \in [n_a, T]$ , contains moments up to order  $2s$  (i.e. two times number of submodels).

*Proof.* (i) $\Leftrightarrow$ (ii) Recall that  $\mathcal{T}'(\boldsymbol{\eta}) = \emptyset \iff o^* > 0$ . Since  $d_N^* \uparrow o^*$  as  $N \rightarrow \infty$ , if  $o^* > 0$ , there exist  $N_o$  such that  $d_{N_o}^* > 0$ . On the other hand, if  $d_{N_o}^* > 0$  then  $o^* > 0$  since  $d_{N_o}^* < o^*$ . Hence,  $\mathcal{T}'(\boldsymbol{\eta})$  is empty.

(i) $\Leftrightarrow$ (iii) To prove this equivalence, we show that  $r^*$  in (4.9) is equal to  $o^*$  in (4.7). Assume  $\eta^*$  is an optimizer of (4.7), then the moments of the distribution  $\mu^*$  with point support at  $\eta^*$  is feasible for (4.9) with the same objective value which implies  $r^* \leq o^*$ . On the other hand, if  $\mathbf{m}^*$  is a minimizer of (4.9), the rank condition implies that there is a corresponding measure  $\mu^*$  with point support, say at  $\eta^*$ . Since this value of  $\eta^*$  is a feasible point of (4.7),  $o^* \leq r^*$ . Therefore,  $r^* = o^*$  from which we conclude that  $r^* > 0$  is equivalent to (i).  $\square$

Note that by forming a single block diagonal matrix containing all LMI constraints in (4.8), it can be transformed into the standard dual form of semidefinite programs. That is:

$$\begin{aligned} d_N^* = \inf_{\mathbf{m}} \quad & \mathbf{b}^T \mathbf{m} + c_o \\ \text{s.t.} \quad & \sum_{\alpha \in \mathcal{I}} \mathbf{A}_\alpha m_\alpha \succeq \mathbf{C} \end{aligned} \quad (4.10)$$

where  $\mathcal{I}$  is the set of the multi-indexes of all moments that occur in (4.8) except the zeroth moment  $m_{\mathbf{0}} = 1$  which is used to form the constant terms  $\mathbf{C}$  and  $c_o$ .

*Remark 8.* It is important to highlight the reduction achieved by employing the running intersection property while forming the optimization problem (4.8). The conventional moment relaxation of order  $N$  in [41] would require  $O((Tn_y + n_y)^{2N})$  variables with a moment matrix of the size  $\begin{pmatrix} N + Tn_y + n_y \\ Tn_y + n_y \end{pmatrix}$ . On the other hand, (4.8) involves only  $O((n_a n_y + n_y)^{2N})$  variables with

$T - n_a + 1$  moment matrices of the size  $\begin{pmatrix} N + n_a n_y + n_y \\ n_a n_y + n_y \end{pmatrix}$  where, in general, the length of the experimental data  $T$  is substantially larger than the order of the regressor  $n_a$  (i.e.  $n_a \ll T$ ).

### 4.3 Numerical Considerations

From Proposition 3 it follows that (in)validation/validation certificates can be obtained, for instance, by solving a sequence of problems with increasing  $N$  until either a solution with  $d^* > 0$  is found (in which case the data invalidates the model) or condition (iii) fails, in which case the data record is consistent with the model and *a-priori* information. However, a potential numerical difficulty here stems from the fact that in practice  $d^* = 0$  is never reached since there is no closed form solution to the problem and numerical solvers terminate when the duality gap drops below a certain level. Thus, from a practical standpoint, more useful conditions can be obtained by examining the primal problem,

which can be written as

$$\begin{aligned}
p_N^* = \max_{\mathbf{X}} \quad & \langle \mathbf{C}, \mathbf{X} \rangle + c_o \\
\text{s.t.} \quad & \\
& \langle \mathbf{A}_\alpha, \mathbf{X} \rangle = b_\alpha \quad \forall \alpha \in \mathcal{I} \\
& \mathbf{X} \succeq 0
\end{aligned} \tag{4.11}$$

where, without loss of generality,  $\mathbf{X}$  can be chosen to have the same block-diagonal structure that  $\mathbf{A}_\alpha$ s and  $\mathbf{C}$ . Note that, from duality,  $p_N^* \leq d_N^*$ , and hence  $p_N^* > 0$  provides an invalidation certificate. In fact if problem (4.11) is feasible, strong duality holds [44]. Thus  $p_N^* > 0$  is strictly equivalent to  $d_N^* > 0$ , leading to the following result:

*Proposition 4.* The consistency set  $\mathcal{T}'(\boldsymbol{\eta})$  is empty if and only if  $p_{N_o}^* > 0$  for some finite  $N_o$ .

*Proof.* Follows directly from strong duality and Proposition 3. □

Let  $\tilde{d}_N$  and  $\tilde{p}_N$  be the numerical solutions to problems (4.8) and (4.11) respectively, obtained using a standard SDP solver. As noted before, in practice, one always has  $\tilde{d}_N > 0$ . Thus it can be hard to ascertain whether conditions (ii) or (iii) in Proposition 3 fail. On the other hand,  $\tilde{p} > 0$  is an invalidation certificate.

*Remark 9.* A certificate that the existing data record is consistent with the model and *a priori* assumptions can be obtained by resorting to a variation of the so-called flat extension property for sparse polynomial optimization stated in [44]. In particular, if for some  $N$ ,  $\tilde{p} \leq 0$  and the dual solution satisfies  $\text{rank} [\mathbf{M}_N(\mathbf{m}_{0:n_a}^*)] = 1$ ;  $\text{rank} [\mathbf{M}_N(\mathbf{m}_{t-n_a:t-1}^*)] = 1$  and  $\text{rank} [\mathbf{M}_N(\mathbf{m}_{t-n_a:t}^*)] = \text{rank} [\mathbf{M}_{N-2}(\mathbf{m}_{t-n_a:t}^*)] \quad \forall t \in [n_a, T]$  where  $\mathbf{m}^*$  denotes an optimal solution of (4.8), then this certifies that  $o^* = d_N^* = 0$ ; hence  $\mathcal{T}'(\boldsymbol{\eta})$  is not empty.

*Remark 10.* It is also worth pointing out that all of the results above (including the running intersection property) hold true in the presence of bounded parametric model uncertainty. For instance, if  $a_i(k, l)$  is an entry in one of the coefficient matrices of the  $i^{\text{th}}$  submodel, it is possible to use  $a_i(k, l) + \delta_i(k, l)$  with  $|\delta_i(k, l)| \leq \delta^*$  in the *a priori* model. This only requires additional variables in the optimization problem.

## 4.4 Illustrative Examples

In this section we illustrate the effectiveness of the proposed method both using academic examples and a computer vision application. In all cases, we used the moments relaxation corresponding to  $N = s$  and the resulting SDP problem was solved using SEDUMI [16].

### 4.4.1 Academic Examples

We consider the ARX submodels:

$$y_t = 0.2y_{t-1} + 0.24y_{t-2} + 2u_{t-1} \quad (G_1)$$

$$y_t = -1.4y_{t-1} - 0.53y_{t-2} + u_{t-1} \quad (G_2)$$

$$y_t = 1.7y_{t-1} - 0.72y_{t-2} + 0.5u_{t-1} \quad (G_3)$$

and the measurement equation:

$$\tilde{y}_t = y_t + \eta_t. \quad (4.12)$$

We ran different sets of simulations investigating different sources of model mismatch. In all cases, we collected input/output data  $\{u_t, \tilde{y}_t\}$  for  $t \in [0, 96]$  and tried to (in)validate the *a priori* model. In all experiments, when we used data inconsistent with the *a priori* information,  $\tilde{p}^*$  turned out to be positive. Hence, we correctly invalidated the model in each of such cases. On the other hand, whenever the *a priori* information was consistent with *a posteriori* data, we had  $\tilde{p}^* < 0$ .

*Example 2. (Submodel mismatch)* For the first set of experiments, we generated input/output data using different subsets of  $\{G_1, G_2, G_3\}$  with a random switching sequence  $\sigma_t$  and with uniform random noise  $\|\eta_t\| \leq 0.5$ . The noise used in the experiments was within the *a priori* noise bound. We used both correct and incorrect *a priori* submodel sets. Hence, the model should be invalid when a submodel that is not contained in the *a priori* submodel set is used in the actual experiment. The results are summarized in Table 4.1.

*Example 3. (Noise bound mismatch)* For this example, we generated input/output data using different subsets of  $\{G_1, G_2, G_3\}$  with a random switching sequence  $\sigma_t$  and with uniform random noise  $\|\eta_t\| \leq \epsilon$ . The *a priori* submodel set and the actual submodel set used in the experiment were the same. The source of invalidation was the actual noise level exceeding the *a priori* bound. The results of this set of experiments are summarized in Table 4.2.

| A priori        | Actual          | Result             |
|-----------------|-----------------|--------------------|
| $G_1, G_2, G_3$ | $G_1, G_2, G_3$ | not invalidated    |
| $G_1, G_2, G_3$ | $G_1, G_2$      | not invalidated    |
| $G_1, G_2$      | $G_1, G_2, G_3$ | <b>invalidated</b> |
| $G_1, G_2$      | $G_2$           | not invalidated    |
| $G_1, G_2, G_3$ | $G_1$           | not invalidated    |
| $G_1, G_2$      | $G_2, G_3$      | <b>invalidated</b> |

TABLE 4.1: Invalidation results for example 2. The values of  $\tilde{p}$  were respectively  $-3.8441e - 008$ ,  $-8.2932e - 009$ ,  $0.8585$ ,  $-5.4026e - 008$ ,  $-1.5490e - 007$  and  $0.7566$ .

| Submodels       | A priori $\epsilon$ | Actual $\epsilon$ | Result             |
|-----------------|---------------------|-------------------|--------------------|
| $G_1, G_2, G_3$ | 0.5                 | 1                 | <b>invalidated</b> |
| $G_1, G_2, G_3$ | 0.8                 | 1                 | <b>invalidated</b> |
| $G_1, G_2, G_3$ | 1.2                 | 1                 | not invalidated    |
| $G_1, G_2$      | 0.5                 | 1                 | <b>invalidated</b> |
| $G_1, G_2$      | 0.8                 | 1                 | <b>invalidated</b> |
| $G_1, G_2$      | 1.2                 | 1                 | not invalidated    |

TABLE 4.2: Invalidation results for example 3. The values of  $\tilde{p}$  were respectively  $0.0724$ ,  $0.0035$ ,  $-5.1810e - 007$ ,  $0.0737$ ,  $0.0034$  and  $-1.4930e - 007$ .

*Example 4. (Submodel perturbation)* For this set of experiments, we generated input/output data by perturbing the coefficients of *a priori* submodels  $\{G_1, G_2, G_3\}$ , and again using a random switching sequence  $\sigma_t$  and uniformly sampled random noise  $\|\eta_t\| \leq 0.5$ . We denote the submodel whose coefficient values are perturbed by  $e$  percent of their original values as  $G_i + \Delta_e$ . This case is more challenging than the one considered in Example 2 since the dynamics are similar. Nevertheless, we could invalidate in each of our trials. The results for this example are summarized in Table 4.3.

| A priori        | Actual   | Result             |
|-----------------|--|--------------------|
| $G_1, G_2, G_3$ | $G_1 + \Delta_5, G_2 + \Delta_5, G_3 + \Delta_5$ | <b>invalidated</b> |
| $G_1, G_2, G_3$ | $G_1 + \Delta_2, G_2 + \Delta_2, G_3 + \Delta_2$ | <b>invalidated</b> |
| $G_1, G_2, G_3$ | $G_1 + \Delta_1, G_2 + \Delta_1, G_3 + \Delta_1$ | <b>invalidated</b> |
| $G_1, G_2$      | $G_1 + \Delta_5, G_2 + \Delta_5$                 | <b>invalidated</b> |
| $G_1, G_2$      | $G_1 + \Delta_2, G_2 + \Delta_2$                 | <b>invalidated</b> |
| $G_1, G_2$      | $G_1 + \Delta_1, G_2 + \Delta_1$                 | <b>invalidated</b> |
| $G_1, G_2$      | $G_1 + \Delta_2$                                 | <b>invalidated</b> |

TABLE 4.3: Invalidation results for example 4. The values of  $\tilde{p}$  were ,respectively,  $0.8963$ ,  $0.0997$ ,  $0.0080$ ,  $0.0308$ ,  $2.8638e - 004$ ,  $2.9069e - 006$  and  $6.2061e - 006$ .

#### 4.4.2 A Practical Example: Activity Monitoring

In this section we illustrate the application of the proposed model invalidation framework to a non-trivial problem arising in computer vision: activity monitoring. Here we start with a set of dynamic models associated with “normal” behavior. If a person passing in front of the camera exhibits a combination of the normal activities (i.e. his/her behavior can be modeled with a hybrid system that has the normal activity dynamics as submodels), then the activity is considered not interesting. On the other hand, if he/she does something different than what has been encoded in the initial set of normal dynamics, this is an indication of an interesting event. In such cases the model should be invalidated.

We used a training video of walk shown in Fig. 4.2 to identify an autoregressive model for the dynamics of the center of mass of a person walking. By minimizing the  $\ell_\infty$  norm of process noise via linear programming, we obtained the following model for “walk”:

$$\begin{pmatrix} x_t \\ y_t \end{pmatrix} = \begin{pmatrix} 0.4747 & 0.0628 \\ -0.3424 & 1.2250 \end{pmatrix} \begin{pmatrix} x_{t-1} \\ y_{t-1} \end{pmatrix} + \begin{pmatrix} 0.5230 & -0.1144 \\ 0.3574 & -0.2513 \end{pmatrix} \begin{pmatrix} x_{t-2} \\ y_{t-2} \end{pmatrix} \quad (A_1)$$

where  $(x_t, y_t)$  is the normalized coordinate of the center of the person in the  $t^{\text{th}}$  frame. Another activity that we considered normal is “waiting” which can simply be modeled as:

$$\begin{pmatrix} x_t \\ y_t \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_{t-1} \\ y_{t-1} \end{pmatrix}. \quad (A_2)$$

We normalized the image coordinate system so that  $(x, y) \in [0, 1] \times [0, 1]$  and set the measurement noise level to  $\|\boldsymbol{\eta}_t\| \leq 0.04$ . This bound together with the submodels  $(A_1)$  and  $(A_2)$  for “normal” activities constitute the *a priori* information.

As for test purposes, we used four different video sequences. Background subtraction was used to locate the person; and the center of mass was estimated and tracked using the silhouettes. Then the center of mass trajectories were used for model (in)validation. In the first sequence, the person walks, waits and walks again; so the overall activity is normal. In the second sequence, the person runs for which our method found the certificate for invalidity by finding  $\tilde{p} > 0$ . In the third sequence, the person walks and then starts jumping. In the last sequence, the person passes in front of the camera by jumping. Again, our method flagged these abnormal activities by verifying the invalidity of the

FIGURE 4.2: Training sequence used in identification of the submodel ( $A_1$ ) for walking.

models. Sample frames from the sequences are shown in Fig. 4.3 and the results are summarized in Table 4.4.

| A priori   | Actual     | Result             |
|------------|------------|--------------------|
| walk, wait | walk, wait | not invalidated    |
| walk, wait | run        | <b>invalidated</b> |
| walk, wait | walk, jump | <b>invalidated</b> |
| walk, wait | jump       | <b>invalidated</b> |

TABLE 4.4: Invalidation results for activity monitoring. The values of  $\tilde{p}$  were, respectively,  $-2.3303e - 008$ ,  $2.3707e - 005$ ,  $5.0293e - 007$ , and  $1.5998e - 006$ .

## 4.5 Conclusions

In this chapter we considered the model (in)validation problem for switched ARX systems with unknown switches. Given a nominal model, a bound on the measurement noise and experimental input output data, we provided a necessary and sufficient condition that certifies the existence/nonexistence of admissible noise and switching sequences such that the resulting output sequence interpolates the given experimental data within the noise bound. In principle, computing these certificates entails solving a sequence of convex optimization problems involving increasingly large matrices. However, as we show here, if in the process of solving these problems either a positive solution is found or the so-called flat extension property holds, then the process terminates with a certificate that either the model has been invalidated (first case) or that the experimental data is indeed consistent with the model and *a-priori* information. By using duality, the proposed approach exploits the inherently sparse structure of the optimization problem to substantially reduce its computational complexity. The effectiveness of the proposed method was illustrated using both academic examples and a non-trivial problem arising in computer vision: activity monitoring.



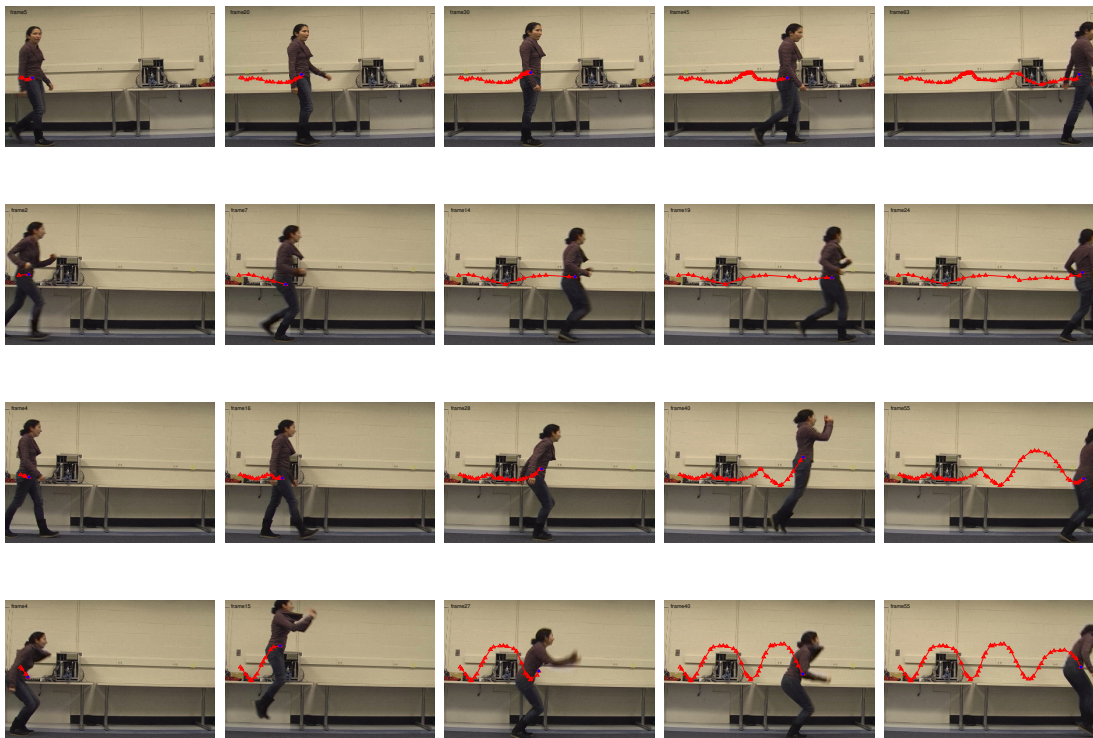


FIGURE 4.3: Top row: Walk, wait, walk sequence (not invalidated). Second row: Running sequence (invalidated). Third row: Walk, jump sequence (invalidated). Last row: Jumping sequence (invalidated).

## Chapter 5

# Clustering Data into Multiple Unknown Subspaces

In this chapter, we deal with static data and we try to cluster the data such that the data within each cluster lie on a subspace of the ambient space. This type of subspace clustering is relevant to many computer vision problems such as affine motion segmentation, face clustering under varying illumination, and image, video and dynamic texture segmentation. The subspace clustering problem and the problem of switched affine dynamical system identification have a quite similar structure. Hence, as we show next, methods developed for one can be easily modified to be used for the other. In particular, in section 5.1, we consider the case where temporal (spatial) ordering of the data points matters and show that utilizing the order dependence it is possible to develop more robust segmentation algorithms based on sparsification. In the second part (section 5.2), we consider the case where ordering does not matter. We build upon an algebraic method ([8]) which is quite sensitive to noise and show that it is possible to significantly reduce the noise resilience by combining elements from convex analysis and the problem of moments. Extensions to algebraic surfaces other than subspaces are also presented.

## 5.1 Sequential Sparsification for Change Detection

### 5.1.1 Introduction and Motivation

Change detection is a very general concept that is encountered in many areas of computer vision. From edge detection to video segmentation or image segmentation, a variety of computer vision tasks can be considered as change detection problems with different interpretations of *change*. Hence, we believe that a general purpose change detection method with only a few adjustable parameters will be valuable. This section takes a step in this direction by exploiting sparsification based optimization.

Under the assumption that there exists an underlying piecewise affine model (e.g. vectors are clustered in different subspaces), our main objective is to find when the model changes from one mode to another and, at the same time, learn the parameters of the model. Hybrid piecewise affine models [8, 9] and mixture models [12, 13, 14] have been the object of considerable attention in the past few years. Although some of the work (for instance [13]) assumes a fixed number of models, one of the main problems when working with hybrid models is that the number of models is usually unknown. [8] provides a closed form algebraic solution for the noise free case, but the estimation of the number of models usually fails when the data is noisy. As we show in this section, assuming a bound on the noise level, allows for recasting the problem into a robust optimization form where the objective is to find the minimum number of clusters (i.e. the simplest model to represent the data). A second point where our method departs from existing clustering techniques is that we make explicit use of the sequential nature of the data. For example, neighboring pixels in an image or consecutive frames in a video sequence are more likely to be within the same segment, and thus imposing continuity of the clusters leads to improved robustness.

Our main result shows that the robust segmentation problem can be recast into a change detection form, where the goal is to detect points where the underlying hybrid model switches modes, or, equivalently, to detect changes in the affine parameters describing the model. In principle, detecting these changes can be hard when the measurements are corrupted by noise. However, as we show, this can be *robustly* accomplished by searching for models that explain the observed data with the lowest possible number of switches (e.g. looking for segmentations that maximize the length of subsequences). This is equivalent to searching for descriptions that maximize the *sparsity* of the vector of first order temporal parameter differences, since each non-zero element of this vector corresponds to a switch. This allows us to proceed as in the dynamic case (see Section 3.4 for details). By exploiting the recently

developed results on signal sparsification, we obtain efficient, computationally tractable segmentation algorithms.

### 5.1.2 Segmentation via Sparsification

In this section we consider the problem of segmenting vector valued sequences  $\{\mathbf{x}(t)\}_{t=0}^T$  that are generated by an affine parametric hybrid model with unknown parameters. Specifically, the models that we consider have the following form:

$$\mathcal{H} : f\left(\mathbf{p}_{\sigma(t)}, \{\mathbf{x}(k)\}_{k=t-i}^{t+j}\right) = \mathbf{0} \quad (5.1)$$

where  $f$  is an affine function<sup>1</sup> of the parameter vector  $\mathbf{p}_{\sigma(t)}$  which takes values from a finite unknown set according to a piecewise constant function  $\sigma(t)$ . Here  $i$  and  $j$  are positive integers that account for the memory of the model (e.g.  $j = 0$  corresponds to a causal model, or  $i = j = 0$  corresponds to a memoryless model).

We say that there exists a *change* at time  $t$  if  $\sigma(t) \neq \sigma(t + 1)$ . Hence segmentation of a given sequence into subsequences is equivalent to finding how many times and when these changes occur. The segmentation problem can be formally stated as follows:

*Problem 6.* Given a sequence  $\{\mathbf{x}(t) \in \mathbb{R}^d\}_{t=1}^T$  generated by a hybrid parametric model  $\mathcal{H}$  of the form (5.1) find the minimum number of segments (i.e. subsequences)  $\{\mathcal{S}_i\}_{i=1}^N$  where on each  $\mathcal{S}_i = \{\mathbf{x}(t)\}_{t=T_i}^{T_{i+1}-1}$ ,  $\sigma(t)$  is constant and  $T_1 = 1, T_{N+1} - 1 = T$ .

This is a difficult problem, since neither the segmentation nor the parameters of the hybrid model are known. In order to overcome this difficulty, we consider the sequence of *first order differences* of the time varying parameters  $\mathbf{p}(t)$ , given by

$$\mathbf{g}(t) = \mathbf{p}(t) - \mathbf{p}(t + 1) \quad (5.2)$$

Clearly, since a non-zero element of this sequence corresponds to a *change*, the sequence should be sparse having only  $N - 1$  non-zero elements out of  $T$ . Next, in order to account for noise we introduce a noise term  $\eta(t)$ , satisfying  $\|\eta\|_* \leq \epsilon$ , where  $\|\cdot\|_*$  denotes a norm relevant to the specific problem

<sup>1</sup>That is:

$$f\left(\mathbf{p}_{\sigma(t)}, \{\mathbf{x}(k)\}_{k=t-i}^{t+j}\right) = A(\mathbf{x})\mathbf{p}_{\sigma(t)} + \mathbf{b}(\mathbf{x})$$

under consideration and  $\epsilon$  is an upper bound on the noise level. In this context, Problem 6 can be recast as an optimization problem as follows:

$$\begin{aligned} & \text{minimize}_{\mathbf{p}(t), \eta(t)} \quad \|\{\mathbf{g}\}\|_{l_0} \\ & \text{subject to} \quad f\left(\mathbf{p}(t), \{\mathbf{x}(k)\}_{k=t-i}^{t+j}\right) = \eta(t) \quad \forall t \\ & \quad \quad \quad \|\{\eta\}\|_* \leq \epsilon \end{aligned} \quad (5.3)$$

<sup>2</sup> Here  $l_0$  is a quasinorm that counts non-zero elements (i.e. minimizing  $l_0$  norm is the same as maximizing sparsity) and can be approximated by the  $l_1$  norm, leading to a linear cost function. When  $f$  is an affine function of  $\mathbf{p}(t)$ , (5.3) has a convex feasibility set  $\mathcal{F}$ . Thus, using the  $l_1$  norm leads to a convex, computationally tractable relaxation. Further, Fazel et al. proposed an iterative procedure in [51] and [56] to improve the solution obtained by the  $l_1$ -norm relaxation. In the sequel, we adopt this heuristic to solve Problem (5.3). This heuristic solves, at each iteration, the following weighted  $l_1$ -norm minimization on the convex feasible set  $\mathcal{F}$ :

$$\begin{aligned} & \text{minimize}_{z, g, p, \eta} \quad \sum_{t=1}^{T-1} w_t^{(k)} z_t \\ & \text{subject to} \quad \|\mathbf{g}(t)\|_\infty \leq z_t \quad \forall t \\ & \quad \quad \quad f\left(\mathbf{p}(t), \{\mathbf{x}(k)\}_{k=t-i}^{t+j}\right) = \eta(t) \quad \forall t \\ & \quad \quad \quad \|\{\eta\}\|_* \leq \epsilon \end{aligned} \quad (5.4)$$

where  $w_i^{(k)} = (z_i^{(k)} + \delta)^{-1}$  are weights with  $z_i^{(k)}$  being the arguments of the optimal solution at the  $k^{\text{th}}$  iteration and  $z^{(0)} = [1, 1, \dots, 1]^T$ ; and where  $\delta$  is a very small regularization constant that determines what should be considered zero.

The choice of  $*$ , the norm by which the noise is characterized, is very important and should be done according to the application in hand. For instance if finding anomalies is desired,  $l_\infty$ -norm performs well. The change detection algorithm highlights the outliers in the data when  $l_\infty$ -norm is used, since it looks for local errors. On the other hand, when a bound on the  $l_1$  or  $l_2$ -norm of the noise is used, the change detection algorithm is more robust to outliers and it favors the continuity of the segments (i.e. longer subsequences). Moreover, the noise level in different segments are not the same in most of the applications. Another advantage of using a global noise bound given by  $l_1$  or  $l_2$ -norm is that the optimization problem adjusts the noise distribution among the segments automatically.

---

<sup>2</sup>If  $f(\mathbf{0}, \cdot)$  is the zero function, (5.3) has a trivial solution  $\mathbf{p}(t) = \mathbf{0}$  for all  $t$ . To overcome this problem, we work with models where  $f(\mathbf{0}, \cdot)$  is not the zero function.

### 5.1.3 Applications

#### 5.1.3.1 Video Segmentation:

Segmenting and indexing video sequences have drawn a significant attention due to the increasing amounts of data in digital video databases. Systems that are capable of segmenting video and extracting key frames that could summarize the video content can substantially simplify browsing these databases over a network and retrieving important content. An analysis of the performances of early shot change detection algorithms is given in [63]. The methods analyzed in [63] can be categorized into two major groups: i) methods based on histogram distances, and ii) methods based on variations of MPEG coefficients. A recent comprehensive study is given in [77] where a formal framework for evaluation is also developed. Other methods include those where scene segmentation is based on image mosaicking [78, 79] or frames are segmented according to underlying subspace structure [80]. Formally, the video segmentation problem can be stated as the following instance of Problem 6:

*Problem 7.* Given the frames  $\{\mathcal{I}(t) \in \mathbb{R}^D\}_{t=1}^T$ , find  $N$  segments (i.e. subsequences)  $\{\mathcal{S}_i\}_{i=1}^N$  where  $N$  is unknown and  $\mathcal{S}_i = \{\mathcal{I}(t)\}_{t=T_i}^{T_{i+1}-1}$  with  $T_1 = 1, T_{N+1} - 1 = T$ , are generated by an underlying hybrid model.

Since the number of pixels  $D$  is usually much larger than the dimension of the subspace where the frames are embedded, it is reasonable to project the data to a lower dimensional space using PCA:

$$\mathcal{I}(t) \mapsto \mathbf{x}(t) \in \mathbb{R}^d.$$

Assuming that each  $\mathbf{x}(t)$  within the same segment lies on the same hyperplane not passing through the origin<sup>3</sup> leads to the following hybrid model:

$$\mathcal{H}_1 : f(\mathbf{p}_{\sigma(t)}, \mathbf{x}(t)) = \mathbf{p}_{\sigma(t)}^T \mathbf{x}(t) - 1 = 0 \quad (5.5)$$

Thus, in this context algorithm (5.4) can be directly used to robustly segment the video sequence. It is also worth stressing that as a by-product of our method we can also perform *key frame extraction* by selecting  $\mathcal{I}(t)$  corresponding to the minimum  $\|\eta(t)\|$  value in a segment (e.g. the frame with the smallest fitting error) as a good representative of the entire segment.

<sup>3</sup>Note that this always can be assumed without loss of generality due to the presence of noise in the data.

The content of a video sequence usually changes in a variety of different ways. For instance: the camera can switch between different scenes (e.g. shots); the activity within the scene can change over time; objects or people can enter or exit the scene, etc. There is a hierarchy in the level of segmentation one would require. The noise level  $\epsilon$  can be used as a tuning knob in this sense.

### 5.1.3.2 Segmentation of Dynamic Textures:

Modeling, recognition, synthesis and segmentation of dynamic textures have drawn a significant attention in recent years (e.g. see for instance [13, 62, 81, 82]). In the case of segmentation tasks, the most commonly used models are mixture models, which are consistent with our hybrid model framework.

In our sequential sparsification framework, the problem of temporal segmentation of dynamic textures reduces to the same mathematical problem as problem 7, with the difference that now the underlying hybrid model should take the dynamics into account. First, dimensionality reduction is performed via PCA ( $\mathcal{I}(t) \mapsto \mathbf{y}(t) \in \mathbb{R}^d$ ) and then the reduced-order data is assumed to satisfy a simple causal autoregressive model similar to the one in [62]. Specifically, the hybrid model we use is:

$$\mathcal{H}_2 : f(\mathbf{p}_{\sigma(t)}, \{\mathbf{y}(k)\}_{k=t-n}^t) = \mathbf{p}_{\sigma(t)}^T \begin{bmatrix} \mathbf{y}(t-n) \\ \vdots \\ \mathbf{y}(t) \end{bmatrix} - 1 = 0 \quad (5.6)$$

where  $n$  is the regressor order. This model, which can be considered as a step driven ARX model, was found to be effective experimentally<sup>4</sup>.

## 5.1.4 Experiments

### 5.1.4.1 Video Segmentation:

To evaluate the proposed method for video segmentation, we used three different video sequences (roadtrip.avi, mountain.avi, drama.avi and family.avi) available from <http://www.open-video.org>. The original mpeg files were decompressed, converted to grayscale and title frames were removed. Each sequence shows a different characteristic on the transition from one

<sup>4</sup>The independent term 1 here accounts for an exogenous driving signal. Normalizing the value of this signal to 1, essentially amounts to absorbing its dynamics into the coefficients  $\mathbf{p}$  of the model. This allows for detecting both changes in the coefficients of the model and in the statistics of the driving signal.

shot to the other. The camera is mostly non-stationary, either shaking or moving. We applied sequential subspace identification, GPCA, a histogram based method and an MPEG method for segmenting the sequences. For the first two methods, we preprocessed each frame by downsampling it by four and projecting to  $\mathbb{R}^3$  using principal component analysis (PCA). For histogram based method, we used bin to bin difference (B2B) with 256 bin histograms and window average thresholding as suggested in [63]. This method has two different parameters. MPEG method [64] is based on DC-difference images. This method requires seven different parameters, one of which is very sensitive to the length of the *gradual transitions*. In our experiments we adjusted the parameters of both methods, by trial and error, to get the best possible results. Hence the resulting comparisons against the proposed sequential-sparsification method correspond to best-case scenarios for both MPEG and B2B.

In the roadtrip sequence, the shot changes are in the form of *cuts*. The first three segments, captured in a moving car, have frames switching between the driver and views of country side through the car windows. The last segment, captured from outside the car, shows the car passing by and moving away so that there is an extreme change in the view angle. Figure 5.1(c) shows the results for this sequence.

The mountain sequence consists of five shots, connected via three *gradual transitions* and one *cut*. The transitions are in the form of approximately forty frames long dissolving effect. Figure 5.1(b) shows our groundtruth segmentation together with the initial and final frames of each shot. The results obtained by applying different methods are shown in 5.1(d).

In fact, drama sequence consists of a single shot. However, the semantic meaning of the sequence changes as the actors and actresses enter and exit the scene. Hence, it is still desirable to segment the video so that the whole story can be summarized by using just one frame from each segment. Figure 5.1(e) shows the groundtruth segmentation<sup>5</sup> together with some key frames. The sequence starts with an empty room, then the first actor enters the empty room during the first transition. The first actor leaves the scene between frames 234 and 273. After approximately 20 frames of empty room, the second actor, the actress and the first actor enter the scene. Hence, three people are in the room during segment 3. In the final transition the second actor exits leaving the first actor and the actress back in the room. The segmentation results for this sequence are also show in 5.1(g).

Family sequence consists of six shots, connected via *gradual transitions* of different lengths as shown in Figure 5.1(f). The segmentation results for this sequence are given in Figure 5.1(h).

Finally, Table 5.1 shows the Rand indices [60] corresponding to the clustering results obtained using the different methods, providing a quantitative criteria for comparison. Since Rand index does not

---

<sup>5</sup>Since the segments are not well defined in this case, the groundtruth segmentation is not unique.



|            | Roadtrip | Mountain | Drama  | Family |
|------------|----------|----------|--------|--------|
| Our Method | 0.9373   | 0.9629   | 0.9802 | 0.9638 |
| MPEG       | 1        | 0.9816   | 0.9133 | 0.9480 |
| GPCA       | 0.6965   | 0.9263   | 0.7968 | 0.8220 |
| Histogram  | 0.9615   | 0.5690   | 0.8809 | 0.9078 |

TABLE 5.1: Rand indices

handle dual memberships (e.g. in the ground truth there are data points in transitions that may either be considered in the previous or the latter segment), the frames in transition are neglected while calculating the indices. These results show that indeed the proposed method does well, with the worst relative performance being against MPEG and B2B in the sequence Roadtrip. This is mostly due to the fact that the parameters in both of these methods were adjusted by a lengthy trial and error process to yield optimal performance in this sequence. Indeed, in the case of MPEG based segmentation, which has seven adjustable parameters, the ones governing cut detection were adjusted to give optimal performance in the Roadtrip sequence, while the five gradual transition parameters were optimized for the Mountain sequence.

#### 5.1.4.2 Temporal Segmentation of Dynamic Textures:

For temporal segmentation of dynamic textures, we used the synthetic dynamic texture database (available from <http://www.svcl.ucsd.edu/projects/motiondytex/synthdb/>) to generate a dataset consisting of dynamic textures that change only temporally. We extracted patches of size  $35 \times 35 \times 60$  from each segment in the database and concatenated them in time. We applied our algorithm to find the frame number at which the video sequence switches from one texture to another one. Since the number of switches is unknown to our method, there were cases where the method found extra changes or missed an existing change. Table 5.2 shows the precision and recall rates over a hundred sequences for a fixed noise level. We used fourth order regressors. A change detected within a window of the size of the regressor order from the true frame of change is considered a correct detection. Some sample patches of dynamic textures are shown in figure 5.2.

Most of the false positives occurred in the sequences that contain flame. This is probably due to the fact that the variance of the stochastic process noise necessary to explain the dynamics of flame is a lot larger than the other textures. Since we used the same noise bound for all dynamic texture experiments, this resulted in extra segments in the sequences that contain flame.

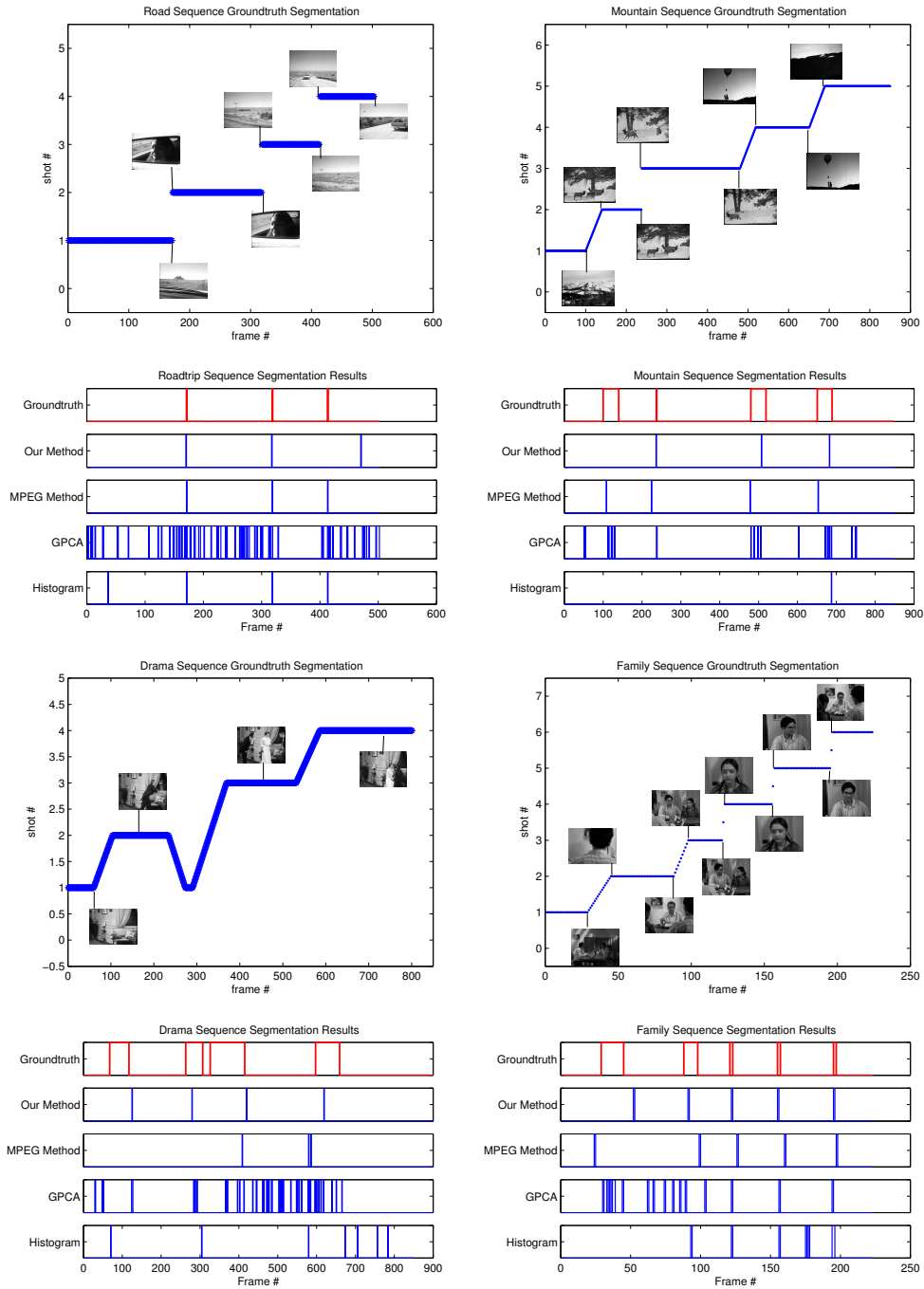


FIGURE 5.1: Video Segmentation Results. First and third row: Ground truth segmentation. Second and last row: Changes detected with different methods. Value 0 corresponds to frames within a segment and value 1 corresponds to the frames in transitions.

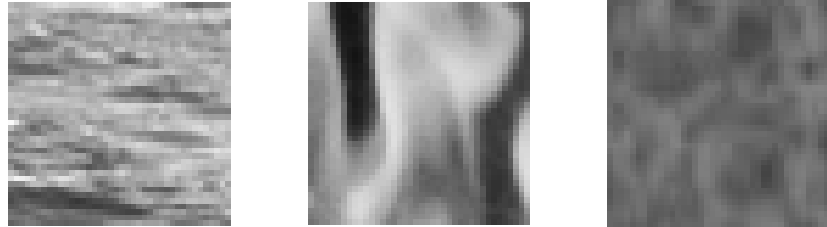


FIGURE 5.2: Sample dynamic texture patches. From left to right: water, flame, steam.

| Sequence Type            | Precision | Recall |
|--------------------------|-----------|--------|
| Two Different Textures   | 0.8384    | 0.9167 |
| Three Different Textures | 0.7362    | 0.6061 |

TABLE 5.2: Results on Dynamic Texture Database

Next, we consider two more challenging sequences. In the first one, we appended in time one patch from smoke to another patch from the same texture but transposed. Therefore, both sequences have the same photometric properties, but differ in the main motion direction: vertical in the first half and horizontal in the second half of the sequence. For the second example, we generated a sequence of river by sliding a window both in space and time (by going forward in time in the first half and by going backward in the second). Hence, dynamics due to river flow are reversed. For these sequences both histogram and MPEG methods fail to detect the cut since the only change is in the dynamics. On the other hand, the proposed method correctly segments both sequences. These results are summarized in Figure 5.3.

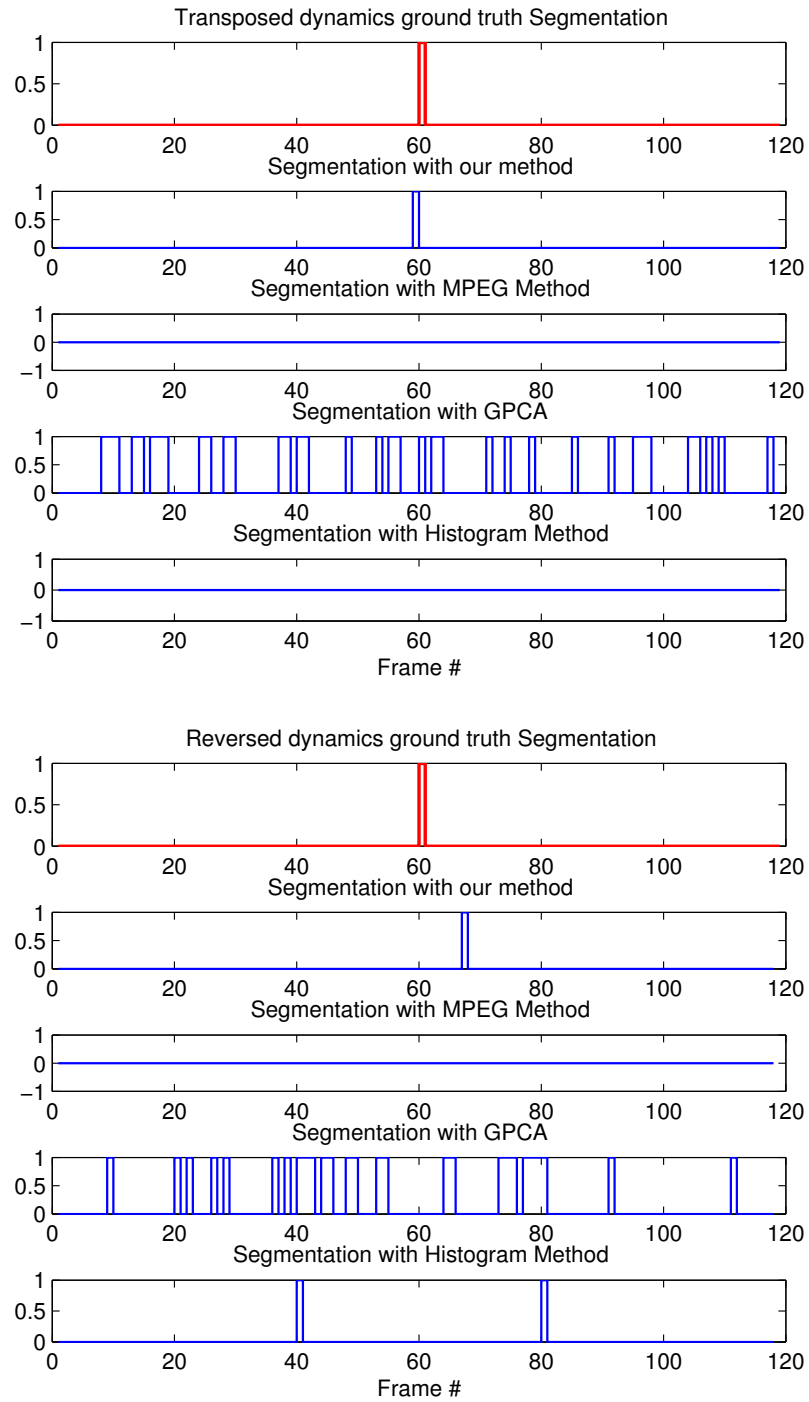


FIGURE 5.3: Results for detecting change in dynamics only. Top: Smoke sequence concatenated with transposed dynamics. Bottom: River sequence concatenated with reversed dynamics.



FIGURE 5.4: An image pair with 3 relocated objects with noisy correspondences superimposed.

## 5.2 GPCA with Denoising: A Moments-Based Convex Approach

### 5.2.1 Introduction and Motivation

Many problems of practical interest can be reduced to identifying a combination of an unknown number of subspaces and quadratic surfaces from sample points. Examples include among others image clustering/classification, segmentation of video sequences, motion segmentation under affine and perspective projections, and identification of piecewise affine systems [1, 8, 47, 80, 83]. In the ideal case of noiseless data, the problem can be elegantly solved using an algebraic approach, Generalized Principal Components Analysis (GPCA) [8], that only entails finding the null space of a matrix constructed from the data. In the case where the data is corrupted by noise, pursuing this approach requires first estimating the null space of a matrix whose entries depend polynomially on the noise, a non-trivial problem. If the noise is small, an approximate solution to this problem is given by the subspace associated with the smallest singular value of the matrix [8]. Recently, an improved method has been proposed [10] based on using a linearization analysis to reduce the problem to minimizing the Sampson distance from the data points to the zero set of a family of polynomials associated with the algebraic surfaces. In principle this approach leads to a constrained non-linear optimization problem. However, an approximate solution can be obtained by solving a generalized eigenvalue problem and improved using gradient descent methods. While this approach can successfully handle moderate noise levels, its performance degrades substantially as the noise level increases, as illustrated next.

Consider the image pair shown in Fig. 5.4, taken from [84], where each point coordinate has been corrupted by uniform random noise. The goal here is to estimate the motion and assign each point to a rigid object. As shown in [8] (see also Section 5.2.4) this problem can be reduced to that of assigning

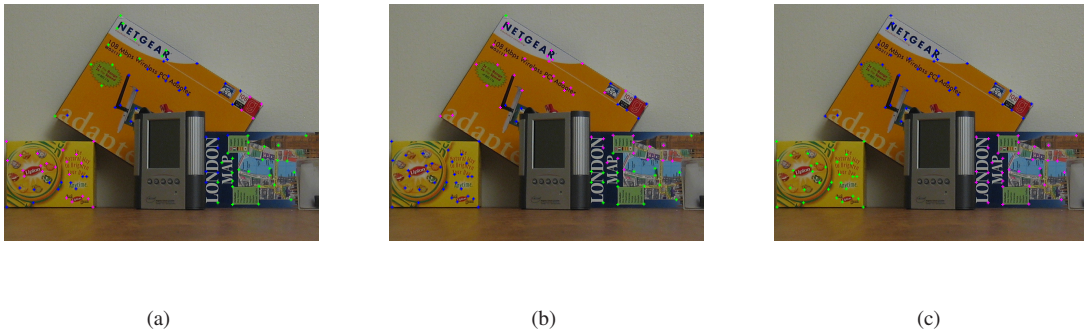


FIGURE 5.5: Sample segmentation results and mean misclassification rates. (a): GPCA segmentation (mean misclassification rate: 31.1%). (b): RGPCA segmentation (mean misclassification rate 14.9%). (c): Proposed algorithm (mean misclassification rate 3.7%). The image size is  $600 \times 800$  and the noise level is  $\pm 10$  pixels.

points to (an unknown number of) subspaces, and hence solved using GPCA. Figures 5.5 (a) and (b) show a typical outcome when applying the original GPCA method and its robust version (RGPCA) [10], and summarizes the results of 20 random runs with  $\pm 10$  pixels noise level. As expected, while robust GPCA outperforms GPCA, it still has a relatively large misclassification rate.

#### Contributions of this section:

This section is motivated by the example above. Our goal is to develop a computationally tractable algorithm for segmenting a mixture of subspaces and quadratic surfaces capable of handling not necessarily small amounts of noise with a-priori unknown statistical properties. Specifically, the main contributions of this section are:

- **Theoretical:** Our main theoretical contribution shows that the problem of estimating a multivariate polynomial  $Q(\mathbf{x})$ , with a-priori bounded order, from noisy measurements of its zero set can be reduced to a constrained rank minimization problem where all the matrices involved are affine in the decision variables and the constraints are convex. The significance of this result is that it allows for eliminating the polynomial dependency on the (unknown) noise terms that renders the problem of estimating  $Q$  (or equivalently the null space of the Veronese matrix in GPCA) difficult, at the price of adding additional (but convex) constraints.
- **Algorithmic:** The theoretical results outlined above allow for recasting the problem of segmenting a mixture of subspaces and quadratic surfaces into a constrained rank minimization

problem. Although in principle rank minimization problems are NP hard, this is a very active research area in optimization, and in the past few years a number of efficient convex relaxations have been proposed. Since all matrices involved are affine in the decision variables, the resulting overall problem is convex and can be efficiently solved, leading to a tractable segmentation algorithm. As we illustrate in this section, this algorithm performs well, even in the presence of substantial noise. An example of this situation is shown in Figure 5.5 (c).

### 5.2.2 Problem Statement

Let  $\mathcal{A} \doteq S_1 \cup S_2 \cup \dots \cup S_n$  denote an arrangement of subspaces  $S_k$  embedded in an ambient space of dimension  $D$ . To this arrangement of subspaces one can associate a set  $\mathcal{V}_{\mathcal{A}}^n$  of homogeneous multivariate polynomials  $Q_j(\mathbf{x})$  of degree  $n$  that has  $\mathcal{A}$  as its zero set, that is:

$$\mathcal{V}_{\mathcal{A}}^n \doteq \{Q_j(\mathbf{x}) \in \mathcal{P}^n : Q_j(\mathbf{x}) = 0 \iff \mathbf{x} \in \mathcal{A}\}.$$

Following [10], in the sequel we will refer to the set  $\mathcal{V}_{\mathcal{A}}^n$  as the set of vanishing polynomials of the arrangement  $\mathcal{A}$ . Note that each polynomial  $Q_j(\mathbf{x})$  can be written as  $Q_j = \nu_n(\mathbf{x})^T \mathbf{c}_j$ , where the vector  $\mathbf{c}_j \in \mathbb{R}^m$ ,  $m = \binom{n+D-1}{D-1}$ , contains the coefficients of the vanishing polynomial in appropriate order. It follows that given  $N_p$  noiseless samples  $\mathbf{x}_1, \dots, \mathbf{x}_{N_p} \in \mathcal{A}$ , the vectors  $\mathbf{c}_j$  span the null space  $\mathcal{N}$  of the matrix  $\mathbf{V}(\mathbf{x}) = \begin{bmatrix} \nu_n(\mathbf{x}_1) & \nu_n(\mathbf{x}_2) & \dots & \nu_n(\mathbf{x}_{N_p}) \end{bmatrix}^T$ . Thus these vectors (and hence a basis for  $\mathcal{V}_{\mathcal{A}}^n$ ) can be found via a simple singular value decomposition of  $\mathbf{V}(\mathbf{x})$ . On the other hand, in the case of noisy samples  $\mathbf{x}_i = \hat{\mathbf{x}}_i + \boldsymbol{\eta}_i$  the matrix  $\mathbf{V}$  depends polynomially on the noise  $\boldsymbol{\eta}_i$ . Thus, when  $\|\boldsymbol{\eta}\|$  is not small, the procedure outlined above no longer works, even when replacing the null space  $\mathcal{N}$  of  $\mathbf{V}$  with the subspace associated with its smallest singular values.

Our goal is to develop a computationally tractable algorithm (and the supporting theory) that allows for estimating both the set of vanishing polynomials and the subspaces from samples corrupted by (not necessarily small) noise. Specifically, we address the following problem:

*Problem 8.* [Polynomial estimation and subspace segmentation] Given a (sufficiently dense) set of noisy samples  $\mathbf{x}_i = \hat{\mathbf{x}}_i + \boldsymbol{\eta}_i$  of points  $\hat{\mathbf{x}}_i$  drawn from an arrangement of subspaces  $\mathcal{A}$ , and *a-priori* bounds on the number of subspaces,  $n$ , and the norm of noise  $\|\boldsymbol{\eta}_i\|_2 \leq \epsilon$ :

- 1.- Estimate a basis for  $\mathcal{V}_{\mathcal{A}}^n$ , the set of vanishing polynomials of  $\mathcal{A}$  of degree up to  $n$ .
- 2.- Estimate a subspace  $\mathcal{S}_k^\perp$  normal to each subspace  $S_k$ .

- 3.- For each noisy sample  $\mathbf{x}_i$  find a subspace  $\mathcal{S}_k$ , a point  $\hat{\mathbf{x}}_i \in \mathcal{S}_k$  and an admissible noise  $\boldsymbol{\eta}_i$ ,  $\|\boldsymbol{\eta}_i\| \leq \epsilon$  such that  $\mathbf{x}_i = \hat{\mathbf{x}}_i + \boldsymbol{\eta}_i$ .

Our main result establishes that the problem above can be reduced to a constrained rank minimization problem where all the matrices involved are affine in the optimization variables. Once this result is established, tractable algorithms can be obtained by appealing to recently introduced convex relaxations of rank minimization problems.

### 5.2.3 Main Results

In this section we present the main theoretical result of the section that allows for recasting Problem 8 into a constrained rank minimization form that can be further relaxed to a convex semi-definite optimization problem.

*Theorem 9.* Let  $\{Q_j(\mathbf{x})\}$  denote the set of polynomials of an arrangement of subspaces  $\mathcal{A}$  and denote by  $n_q$  its dimension. Consider a set of measurements corrupted by norm-bounded noise

$$\mathbf{x}_i = \hat{\mathbf{x}}_i + \boldsymbol{\eta}_i, \quad i = 1, 2, \dots, N_p \quad (5.7)$$

where  $\hat{\mathbf{x}}_i \in \mathcal{A}$  and  $\|\boldsymbol{\eta}_i\|_2 \leq \epsilon$ . Then:

1. If  $n_q > 1$ , there exists an admissible noise sequence  $\boldsymbol{\eta}_i$  and  $N_p$  points  $\hat{\mathbf{x}}_i$  satisfying (5.7) and such that  $Q_j(\hat{\mathbf{x}}_i) = 0$  if and only if there exist  $N_p$  sequences  $\mathbf{m}_i \doteq \{m_\alpha\}$  such that the following conditions hold:

$$\text{rank} \{ \mathbf{M}(\mathbf{x}, \mathbf{m}_1, \dots, \mathbf{m}_{N_p}) \} = h - n_q \quad (5.8)$$

$$\mathbf{L}_i(\mathbf{m}_i) \succeq 0, \quad i = 1, \dots, N_p \quad (5.9)$$

$$\mathbf{K}_i(\epsilon, \mathbf{m}_i) \succeq 0, \quad i = 1, \dots, N_p \quad (5.10)$$

$$\text{rank}(\mathbf{L}_i(\mathbf{m}_i)) = 1, \quad i = 1, \dots, N_p \quad (5.11)$$

where  $h = \binom{n+D-1}{D-1}$ ;  $\mathbf{M} \doteq \mathbf{E}(\mathbf{V})$ <sup>6</sup> and  $\mathbf{L}_i$  and  $\mathbf{K}_i$  are the moment matrices, defined in (2.21), associated with  $\boldsymbol{\eta}_i$  (i.e. the noise affecting the  $i^{\text{th}}$  sample point).

<sup>6</sup>Here the expectation operator  $\mathbf{E}$  acts elementwise on  $\mathbf{V}(\hat{\mathbf{x}}) = \mathbf{V}(\mathbf{x} - \boldsymbol{\eta})$ . That is,  $\mathbf{M}$  is constructed by replacing all the monomials in the noise terms  $\boldsymbol{\eta}$  in  $\mathbf{V}(\mathbf{x} - \boldsymbol{\eta})$  with the corresponding moments.



2. If  $n_q = 1$ , e.g. when all the subspaces  $S_k$  have dimension  $D - 1$ , then the rank constraint (5.11) is no longer required (e.g only (5.8)-(5.10) need to be enforced.)

*Proof.* First, if there exists a noise sequence  $\boldsymbol{\eta}_i^*$  such that

$$Q_j(\mathbf{x}_i - \boldsymbol{\eta}_i^*) = \nu_n(\mathbf{x}_i - \boldsymbol{\eta}_i^*)^T \mathbf{c}_j = 0 \text{ for all } i, j,$$

then, the moments  $\mathbf{m}_i$  of the atomic probability measures  $\mu_i$  with

$$\text{Prob}_{\mu_i}(\boldsymbol{\eta}_i = \boldsymbol{\eta}_i^*) = 1, \text{ Prob}_{\mu_i}(\boldsymbol{\eta}_i \neq \boldsymbol{\eta}_i^*) = 0.$$

trivially satisfy (5.8) through (5.11) and the vectors  $\mathbf{c}_j$  span the null space of the matrix  $\mathbf{M}(\mathbf{x}, \mathbf{m}_1, \dots, \mathbf{m}_{N_p})$ .

To prove the converse, let's first look at part 1. We start by noting that  $\text{rank}(\mathbf{L}_i) = 1$  is equivalent to the existence of a unique atomic measure  $\mu_i$  with one atom whose moments are equal to  $\mathbf{m}_i$  (see [42]) and, hence, there exist  $\boldsymbol{\eta}_i^*$  such that

$$\text{Prob}_{\mu_i}(\boldsymbol{\eta}_i = \boldsymbol{\eta}_i^*) = 1, \text{ Prob}_{\mu_i}(\boldsymbol{\eta}_i \neq \boldsymbol{\eta}_i^*) = 0.$$

Therefore, for this measure, one has

$$\mathbf{V}(\mathbf{x} - \boldsymbol{\eta}^*) = \mathbf{M}(\mathbf{x}, \mathbf{m}_1, \dots, \mathbf{m}_{N_p}).$$

Note that (5.10) implies that  $\|\boldsymbol{\eta}_i^*\|_2 \leq \epsilon$ . Given this, let  $\mathbf{c}_j$  be linearly independent vectors which span the null space of the matrix  $\mathbf{V}(\mathbf{x} - \boldsymbol{\eta}^*)$ . To conclude this part of the proof, define  $Q_j(\mathbf{x}_i) = \nu_n(\mathbf{x}_i)^T \mathbf{c}_j$  and  $\hat{\mathbf{x}}_i \doteq \mathbf{x}_i - \boldsymbol{\eta}_i^*$ . From the reasoning above it follows that

$$Q_j(\hat{\mathbf{x}}_i) = \nu_n(\hat{\mathbf{x}}_i)^T \mathbf{c}_j = \nu_n(\mathbf{x}_i - \boldsymbol{\eta}_i^*)^T \mathbf{c}_j = 0 \text{ for all } i, j.$$

We now turn our attention to part 2. Since condition (5.11) might not be satisfied, the measures  $\mu_i$  compatible with the moment sequences  $\mathbf{m}_i$  are not necessarily atomic measures. However, equation (5.8) implies that  $\mathcal{N}$ , the null space of the matrix  $\mathbf{M}(\mathbf{x}, \mathbf{m}_1, \dots, \mathbf{m}_{N_p})$ , has dimension one. Let  $\mathbf{c}$  span  $\mathcal{N}$ , i.e.,

$$\mathbf{M}(\mathbf{x}, \mathbf{m}_1, \dots, \mathbf{m}_{N_p})\mathbf{c} = 0.$$

Linearity of expectation implies that

$$\mathbf{E}_{\mu_i}[\nu_n(\mathbf{x}_i - \boldsymbol{\eta}_i)^T \mathbf{c}] = 0.$$

Therefore, there exist  $\boldsymbol{\eta}_i^+$  and  $\boldsymbol{\eta}_i^-$  within the noise bounds such that

$$\nu_n(\mathbf{x}_i - \boldsymbol{\eta}_i^+)^T \mathbf{c} \geq 0 \text{ and } \nu_n(\mathbf{x}_i - \boldsymbol{\eta}_i^-)^T \mathbf{c} \leq 0.$$

Since  $\nu_n(\mathbf{x}_i - \boldsymbol{\eta}_i)^T \mathbf{c}$  is a polynomial and, hence, a continuous function of  $\boldsymbol{\eta}_i$ , there exists a  $\boldsymbol{\eta}_i^*$  such that

$$\nu_n(\mathbf{x}_i - \boldsymbol{\eta}_i^*)^T \mathbf{c} = 0.$$

The proof is concluded by defining the unique (up to a multiplying constant) vanishing polynomial  $Q(\hat{\mathbf{x}}_i) = \nu_n(\hat{\mathbf{x}}_i)^T \mathbf{c}$  which, given the reasoning above, satisfies

$$Q(\hat{\mathbf{x}}_i) = \nu_n(\hat{\mathbf{x}}_i)^T \mathbf{c} = \nu_n(\mathbf{x}_i - \boldsymbol{\eta}_i^*)^T \mathbf{c} = 0 \text{ for all } i.$$

□

*Remark 11.* Note that, without the rank constraint (5.11), a finite order approximation in (5.9) and (5.10) only leads to necessary conditions for  $\mathbf{m}_i$  to be moments of a probability distribution. Hence, implementing the approach in part 2 with these truncated matrices only provides an approximation of the true  $Q(\cdot)$ . However, the approximation converges to the true solution as the moment approximation order increases.

### 5.2.3.1 A Convex Relaxation

Although there are a few methods to solve rank constrained semidefinite programs (see for instance [85]), these are usually computationally intensive and do not have convergence guarantees. For this reason, we proceed by relaxing the conditions in order to obtain a convex program that approximates the original problem and that can be efficiently solved with off-the-shelf solvers. To this end, we first consider the following problem:

$$\begin{aligned} & \text{minimize}_{\mathbf{m}_i} && \text{rank} \{ \mathbf{M}(\mathbf{x}, \mathbf{m}_1, \dots, \mathbf{m}_{N_p}) \} \\ & \text{subject to} && \mathbf{L}_i^{\lceil \frac{n}{2} \rceil}(\mathbf{m}_i) \succeq 0 \quad i = 1, \dots, N_p \\ & && \mathbf{K}_i^{\lceil \frac{n}{2} \rceil}(\epsilon, \mathbf{m}_i) \succeq 0 \quad i = 1, \dots, N_p \end{aligned} \quad (5.12)$$

where we truncate the moment matrices such that they only contain the moments up to order  $n$  ( $n+1$  if  $n$  is odd) which is the maximum order of the noise monomials appearing in the embedded data matrix  $\mathbf{V}$ .

Equation (5.12) is an affine matrix rank minimization problem subject to convex constraints. Although rank minimization is an NP–Hard problem, efficient convex relaxations are available. In particular, good approximate solutions can be obtained by using a log–det heuristic [51] that relaxes rank minimization to a sequence of convex problems<sup>7</sup>. Inspired by the adaptive step size defined for weighted  $\ell_1$  minimization in [65], the following problem is solved at each iteration:

$$\begin{aligned} \min_{\mathbf{m}_{1:N_p}, \mathbf{Y}, \mathbf{Z}} \quad & \text{trace}(\mathbf{W}_y^{(k)} \mathbf{Y}) + \text{trace}(\mathbf{W}_z^{(k)} \mathbf{Z}) \\ \text{subject to} \quad & \begin{bmatrix} \mathbf{Y} & \mathbf{M}(\mathbf{m}_{1:N_p}) \\ \mathbf{M}(\mathbf{m}_{1:N_p})^T & \mathbf{Z} \end{bmatrix} \succeq 0 \\ & \mathbf{L}_i^{\lceil \frac{n}{2} \rceil}(\mathbf{m}_i) \succeq 0, \mathbf{K}_i^{\lceil \frac{n}{2} \rceil}(\mathbf{m}_i) \succeq 0 \quad i = 1, \dots, N_p \end{aligned} \quad (5.13)$$

where  $\mathbf{W}_y^{(k+1)} = (\mathbf{Y}^{(k)} + \lambda_k \mathbf{I})^{-1}$ ,  $\mathbf{W}_z^{(k+1)} = (\mathbf{Z}^{(k)} + \lambda_k \mathbf{I})^{-1}$  are weights with  $\mathbf{Y}^{(k)}, \mathbf{Z}^{(k)}$  being the arguments of the optimal solution in the  $k^{\text{th}}$  iteration;  $\lambda_k$ , the regularization parameter, is set to the  $(h - n_q + 1)^{\text{th}}$  largest singular value of current optimal  $\mathbf{M}$  in iteration  $k$ ; and  $\mathbf{W}_y^{(0)}, \mathbf{W}_z^{(0)}$  are initialized with identity matrices<sup>8</sup>. Note that the matrices  $\mathbf{M}$ ,  $\mathbf{L}$  and  $\mathbf{K}$  are affine in the moment variables  $\mathbf{m}_i$  as defined in Theorem 9, and  $\mathbf{Y}$  and  $\mathbf{Z}$  are symmetric positive definite auxiliary variables. Hence, (5.13) is a convex semidefinite program.

*Remark 12.* Although when  $n_q > 1$ , the relaxation is less tight since we drop the rank–constraints on the moment matrices, in practice the moment matrices found by solving (5.13) are close to rank 1. Hence approximating them with rank 1 matrices using a singular value decomposition (SVD) gives satisfactory results (for example see Fig. 5.8).

After solving (5.13), it is possible to extract an admissible noise sequence from the moment matrix  $\mathbf{L}$  as described in [87]. In particular, when  $\mathbf{L}_i$  has rank 1, one can retrieve the noise from the elements of the first singular vector of  $\mathbf{L}_i$  corresponding to first order moments. Once an admissible noise sequence is obtained, we denoise the data and proceed with segmentation using polynomial differentiation as in [8].

<sup>7</sup>Although there are very recent faster algorithms for rank minimization (e.g. [86]), they currently cannot handle semidefinite constraints.

<sup>8</sup>The first iteration solves the nuclear norm heuristic. Then each iteration aims to reduce the rank further through the weighting scheme. In our experiments, the convergence is typically achieved within the first 10 iterations.

### 5.2.3.2 Extension to Quadratic Surfaces

In this section we briefly present a straightforward extension of our moments-based approach to the segmentation of a mixture of quadratic surfaces discussed in [1]. In particular, we are interested in the class of problems that arises in the context of motion segmentation from two perspective views, given the point correspondences [1, 88, 89]. The main idea is that point correspondences of a single rigid body satisfy the epipolar constraint  $\mathbf{x}_1^T \mathbf{F} \mathbf{x}_2 = 0$ <sup>9</sup> where  $\mathbf{F} \in \mathbb{R}^{3 \times 3}$  is the fundamental matrix and  $\mathbf{x}_j = (x_j, y_j, 1)^T$  for  $j = 1, 2$ , are the corresponding points in two views in homogeneous coordinates. In the case of  $n$  rigid objects, all point correspondences satisfy

$$\prod_{i=1}^n \mathbf{x}_1^T \mathbf{F}_i \mathbf{x}_2 = 0. \quad (5.14)$$

Let  $\mathbf{z} = (x_1, y_1, x_2, y_2)^T$  be a joint vector of corresponding pairs and define the *perspective embedding* of degree  $2n$ ,  $\pi_{2n}(\mathbf{z}) : \mathbb{R}^4 \rightarrow \mathbb{R}^m$ ,  $m = \binom{2+n}{n}^2$  as

$$\pi_{2n}(\mathbf{z}) \doteq \begin{bmatrix} x_1^{\alpha_1^{(1)}} y_1^{\alpha_2^{(1)}} x_2^{\alpha_3^{(1)}} y_2^{\alpha_4^{(1)}} \\ \vdots \\ x_1^{\alpha_1^{(m)}} y_1^{\alpha_2^{(m)}} x_2^{\alpha_3^{(m)}} y_2^{\alpha_4^{(m)}} \end{bmatrix}$$

where the exponents of the monomials satisfy  $\alpha_1^{(k)} + \alpha_2^{(k)} \leq n$ ,  $\alpha_3^{(k)} + \alpha_4^{(k)} \leq n$  for all  $k = 1, \dots, m$ .

As in section 5.2.2, we can associate with Eq. (5.14) a polynomial  $Q_j(\mathbf{z}) = \mathbf{c}_j^T \pi_{2n}(\mathbf{z})$  where  $\mathbf{c}_j$  is the coefficient vector. It follows that, given  $N_p$  noiseless corresponding pairs  $\mathbf{z}_1, \dots, \mathbf{z}_{N_p}$ , the vectors  $\mathbf{c}_j$  span the null space  $\mathcal{N}$  of the embedded data matrix  $\mathbf{P}(\mathbf{z}) = \begin{bmatrix} \pi_{2n}(\mathbf{z}_1) & \pi_{2n}(\mathbf{z}_2) & \dots & \pi_{2n}(\mathbf{z}_{N_p}) \end{bmatrix}^T$ . In the noisy case,  $\mathbf{P}(\mathbf{z})$  depends polynomially on the noise  $\boldsymbol{\eta}$ . However, as before, we can resort to moments-based rank minimization of  $\mathbf{E}(\mathbf{P})$  to find a noise sequence  $\boldsymbol{\eta}^*$  that renders  $\mathbf{P}$  rank deficient and obtain its null space (and hence  $Q_j$ ). Once the data is denoised using  $\boldsymbol{\eta}^*$ , we proceed as in [1]. First, we form the *mutual contraction subspaces* between different pairs  $(\mathbf{z}_k, \mathbf{z}_1)$  from the derivatives and Hessians of polynomials  $Q_j$ . Applying spectral clustering to the similarity matrix built from the subspace angles between mutual contraction subspaces leads then to the desired segmentation.

<sup>9</sup>As shown in [90], this equation can be written as a quadratic form in the joint image space.

### 5.2.3.3 Handling Outliers

Upto this point, we assume that the data is only corrupted by norm bounded noise. However, this assumption could be quite conservative in many real-world problems due to the existence of outliers that do not belong to any of the algebraic surfaces. One way of eliminating outliers is proposed in [10] where authors use methods from robust statistics such as influence functions and multivariate trimming. In this section, we propose a different approach. Namely, we take a step towards handling outliers in a principled way within our optimization framework. For simplicity of exposition, we consider the setup in Problem 8 with the addition of outlying data points. The extension to more general algebraic surfaces follows similar lines.

We call the sample point  $\mathbf{x}_j$  an outlier if the Euclidean distance of  $\mathbf{x}_j$  to the subspace arrangement  $\mathcal{A}$  is more than the noise bound  $\epsilon$ , and an inlier otherwise. The index sets corresponding to inliers and outliers are denoted by  $\mathcal{I}_{in} \subseteq \{1, \dots, N_p\}$  and  $\mathcal{I}_{out} = \{1, \dots, N_p\} \setminus \mathcal{I}_{in}$ , respectively. Let  $\mathbf{V}_{in}$  be the matrix whose rows are  $\nu_n(\hat{\mathbf{x}}_i)^T$  for all  $i \in \mathcal{I}_{in}$ . Then, for all  $j \in \mathcal{I}_{out}$  and for any noise value  $\boldsymbol{\eta}_j$  with  $\|\boldsymbol{\eta}_j\|_2 \leq \epsilon$ ,  $\nu_n(\mathbf{x}_j - \boldsymbol{\eta}_j)^T$  will not be in the row space of  $\mathbf{V}_{in}$ . Thus, if we form  $\mathbf{M}$  as in Theorem 9,  $\mathbf{M}$  will have some ‘‘outlying’’ rows corresponding to outliers. In order to have a rank deficient  $\mathbf{M}$ , we should compensate for the outliers. With this observation in mind, we consider the following optimization problem:

$$\begin{aligned}
& \min_{\mathbf{m}_{1:N_p}, \mathbf{E}} \quad \|\mathbf{E}\|_{rs} \\
& \text{subject to} \quad \text{rank} \{ \mathbf{M}(\mathbf{x}, \mathbf{m}_1, \dots, \mathbf{m}_{N_p}) + \mathbf{E} \} = h - n_q \\
& \quad \mathbf{L}_i(\mathbf{m}_i) \succeq 0, \quad i = 1, \dots, N_p \\
& \quad \mathbf{K}_i(\epsilon, \mathbf{m}_i) \succeq 0, \quad i = 1, \dots, N_p \\
& \quad \text{rank}(\mathbf{L}_i(\mathbf{m}_i)) = 1, \quad i = 1, \dots, N_p
\end{aligned} \tag{5.15}$$

where  $\|\mathbf{E}\|_{rs}$  counts the number of non-zero rows in  $\mathbf{E}$  (i.e. row sparsity) and the rest is the same as in Equations (5.8)-(5.11). Essentially, non-zero rows of  $\mathbf{E}$  correspond to outlying data points, hence in a sense (5.15) looks for a compatible model with minimum number of outliers.

As before, we start by relaxing the rank constraints. We drop the rank constraints on  $\mathbf{L}_i$ 's and move the constraint on  $\mathbf{M} + \mathbf{E}$  to the objective with a Lagrange multiplier:

$$\begin{aligned}
& \text{minimize}_{\mathbf{m}_i, \mathbf{E}} \quad \text{rank} \{ \mathbf{M}(\mathbf{x}, \mathbf{m}_1, \dots, \mathbf{m}_{N_p}) + \mathbf{E} \} + \gamma \|\mathbf{E}\|_{rs} \\
& \text{subject to} \quad \mathbf{L}_i^{\lceil \frac{n}{2} \rceil}(\mathbf{m}_i) \succeq 0 \quad i = 1, \dots, N_p \\
& \quad \mathbf{K}_i^{\lceil \frac{n}{2} \rceil}(\epsilon, \mathbf{m}_i) \succeq 0 \quad i = 1, \dots, N_p
\end{aligned} \tag{5.16}$$

Row sparsity is a non-convex function; it is indeed equivalent to sparsity in vector valued sequences if we treat each row  $\mathbf{E}_i$  of the matrix  $\mathbf{E}$  as a vector in the sequence. Hence, the relaxation developed in Lemma 1 can be applied in this case as well. As for the rank, nuclear norm relaxation can be used. On top of these relaxations, we adopt the weighting scheme to obtain following convex program:

$$\begin{aligned} \min_{\mathbf{m}_{1:N_p}, \mathbf{Y}, \mathbf{Z}} \quad & \text{trace}(\mathbf{W}_y^{(k)} \mathbf{Y}) + \text{trace}(\mathbf{W}_z^{(k)} \mathbf{Z}) + \gamma^{(k)} \sum_{i=1}^{N_p} w_i^{(k)} \|\mathbf{E}_i\|_\infty \\ \text{subject to} \quad & \begin{bmatrix} \mathbf{Y} & \mathbf{M}(\mathbf{m}_{1:N_p}) + \mathbf{E} \\ \mathbf{M}(\mathbf{m}_{1:N_p})^T + \mathbf{E}^T & \mathbf{Z} \end{bmatrix} \succeq 0 \\ & \mathbf{L}_i^{\lceil \frac{n}{2} \rceil}(\mathbf{m}_i) \succeq 0, \mathbf{K}_i^{\lceil \frac{n}{2} \rceil}(\mathbf{m}_i) \succeq 0 \quad i = 1, \dots, N_p \end{aligned} \quad (5.17)$$

where  $\mathbf{W}_y^{(k+1)} = (\mathbf{Y}^{(k)} + \lambda_k \mathbf{I})^{-1}$ ,  $\mathbf{W}_z^{(k+1)} = (\mathbf{Z}^{(k)} + \lambda_k \mathbf{I})^{-1}$ ,  $w_i^{(k+1)} = \frac{1}{\|\mathbf{E}_i^{(k)}\|_\infty + \delta}$  are weights with  $\mathbf{Y}^{(k)}$ ,  $\mathbf{Z}^{(k)}$  and  $\mathbf{E}^{(k)}$  being the arguments of the optimal solution in the  $k^{\text{th}}$  iteration; the regularization parameter  $\lambda_k$  is set to the  $(h - n_q + 1)^{\text{th}}$  largest singular value of current optimal  $\mathbf{M} + \mathbf{E}$  in iteration  $k$ ; and  $\delta$  is another small regularization constant.  $\mathbf{W}_y^{(0)}$ ,  $\mathbf{W}_z^{(0)}$  are initialized with identity matrices; and  $w_i^{(0)} = 1/\sqrt{N_p}$  for all  $i$ . Lagrange multiplier  $\gamma^{(0)}$  is a tuning parameter that regulates the amount of outliers. A small value of  $\gamma^{(0)}$  would allow lots of data points to be classified as outliers, whereas a large value might result in too few outliers preventing the algorithm to find a rank deficient solution. Once  $\gamma^{(0)}$  is appropriately initialized,  $\gamma^{(k)}$  is set to  $\gamma^{(k)} = \gamma^{(0)} \frac{\|\mathbf{W}_y^{(k)}\|_2 + \|\mathbf{W}_z^{(k)}\|_2}{2\|[w_1^{(k)}, \dots, w_{N_p}^{(k)}]\|_2}$  so that the effect of rank minimization term and row sparsity term are approximately constant all through the iterations.

## 5.2.4 Experiments

In this section we illustrate the ability of the proposed method to deal with relatively large noise using both synthetic and real data. The data we use here is taken from the literature and thus publicly available.<sup>10</sup>

### 5.2.4.1 Synthetic Data

*Example 1:* First, we use synthetic data to investigate the performance of the moments-based method for different number of subspaces, different subspace dimensions and different noise levels. For each set of examples, we generated 20 random instances of the problem data with data points lying on a subspace arrangement corrupted by random noise in the direction of subspace normals with uniform

<sup>10</sup>A reference implementation is provided at <http://www.coe.neu.edu/~necmiye/cvpr10.htm>

random magnitude in  $[0.8\epsilon, \epsilon]$ . All data points were sampled within the unit hypercube of the ambient space, so that the noise level,  $\epsilon$ , corresponds roughly to the percent noise level. The segmentation was performed using the convex relaxation described in section 5.2.3.1, implemented in Matlab using CVX toolbox [18], and performance was evaluated in terms of the average, over all runs, of the worst case fitting error,

$$\text{err}_f = \max_{i \in [1, N_p]} \min_{k \in [1, n]} \mathbf{b}_k^T \mathbf{x}_i$$

where  $\mathbf{b}_k$ 's are the subspace normals found by each algorithm. The results are summarized in Table 5.3, showing that in all cases the moments-based method outperforms both GPCA and RGPCA.

| $D$ | $d_k$     | $N$          | $\epsilon$ | Moments $\text{err}_f$ | GPCA $\text{err}_f$ | RGPCA $\text{err}_f$ |
|-----|-----------|--------------|------------|------------------------|---------------------|----------------------|
| 4   | [3, 3]    | [50, 50]     | 0.15       | 0.250 (0.183)          | 0.488 (0.477)       | 0.253 (0.234)        |
| 3   | [2, 2]    | [50, 50]     | 0.10       | 0.101 (0.100)          | 0.393 (0.334)       | 0.192 (0.134)        |
| 3   | [2, 2]    | [50, 50]     | 0.15       | 0.154 (0.151)          | 0.488 (0.443)       | 0.289 (0.225)        |
| 3   | [2, 2]    | [50, 50]     | 0.20       | 0.227 (0.205)          | 0.543 (0.539)       | 0.370 (0.329)        |
| 3   | [2, 2, 2] | [40, 40, 40] | 0.15       | 0.259 (0.206)          | 0.499 (0.477)       | 0.421 (0.398)        |

TABLE 5.3: Synthetic Data Results.  $D$  and  $d_k$  denote the dimension of the ambient space and subspaces, respectively.  $N$  shows the number of samples per subspace.  $\epsilon$  denotes the true noise level. The last three columns show the mean and median (in parenthesis) fitting errors.

Figure 5.6 shows a typical noisy data set and the denoised version obtained by substituting the noise estimates found via moments. As illustrated there, it is possible to align the noisy data points on subspaces before the clustering stage. Hence applying GPCA clustering to these almost noiseless data points avoids the difficulties entailed in using polynomial differentiation in the presence of noise.

*Example 2:* This example illustrates the case where subspaces have different dimensions. In particular, we considered a subspace arrangement consisting of two lines and a plane in  $\mathbb{R}^3$ . The set-up is similar to the simulations in Example 1 with  $d_k = [2, 1, 1]$ ,  $N = [40, 40, 40]$  and  $\epsilon = 0.1$ . Figure 5.8 shows the result where it can be seen that moments-based method successfully denoised the data. Quantitatively, the fitting error for our algorithm is  $\text{err}_f = 0.09$ , whereas the fitting error for GPCA is 0.2575.

*Example 3:* This example illustrates the outlier handling algorithm. We considered a subspace arrangement consisting of two planes in  $\mathbb{R}^3$ . We sampled 50 data points within the unit cube from each plane. We corrupted the data with noise along the normal direction of the subspaces with uniform random magnitude in  $[0.12, 0.15]$ . Additionally, we contaminated the data with 10 outliers sampled from a multivariate Gaussian distribution with zero-mean and identity variance. We solved the problem using algorithm (5.17) with  $\gamma^{(0)} = 17$ . The data points corresponding to non-zero rows of the  $\mathbf{E}$

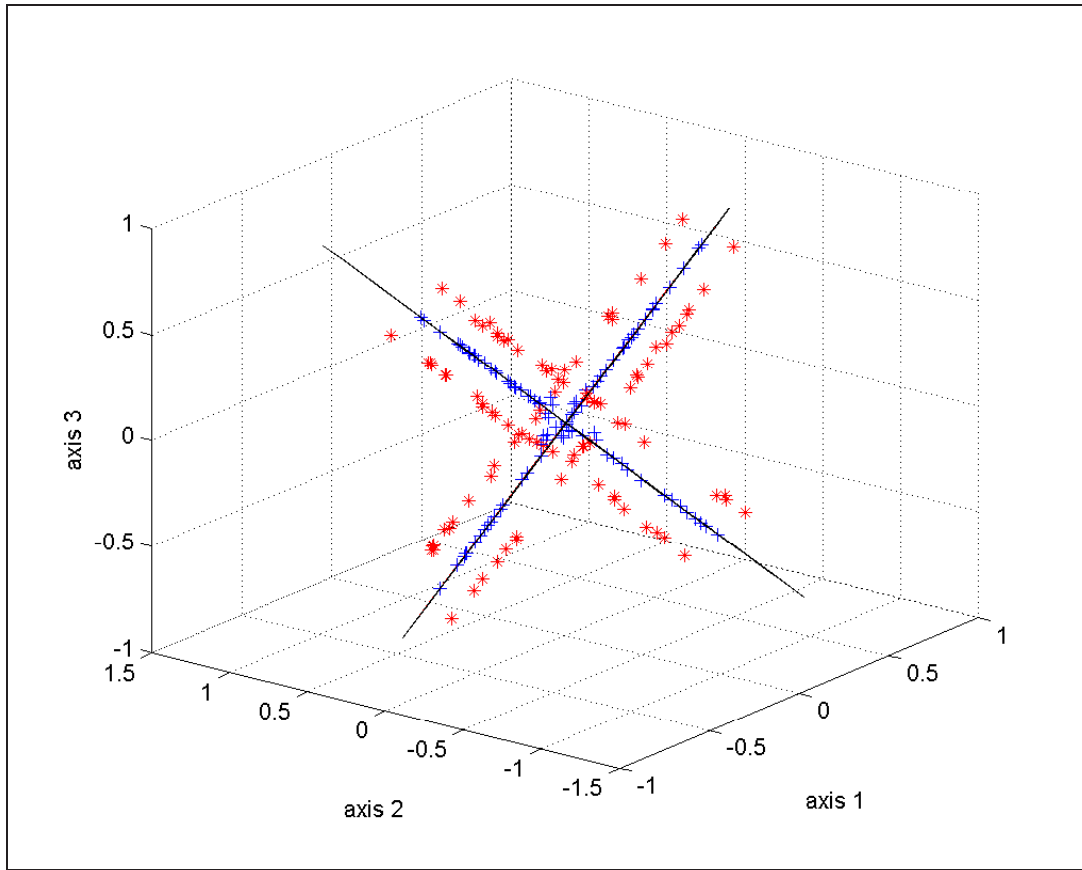


FIGURE 5.6: Example using synthetic data laying on two planes in  $\mathbb{R}^3$ . Here the red stars and blue plus signs indicate the original (noisy) and the denoised data points, respectively.

matrix were classified as outliers. We denoised the rest of the data points (i.e. inliers) with the noise estimates obtained with moments-based optimization. The results are shown in Fig. 5.9.

#### 5.2.4.2 2-D motion Estimation and Segmentation

Next, we consider the problem of simultaneous multiple 2-D motion estimation and clustering from two images. Let  $(\mathbf{x}_i^{(n)}, \mathbf{y}_i^{(n)})$  be the coordinates of the  $n^{\text{th}}$  feature point in the  $i^{\text{th}}$  image. The coordinates of this point across frames are related through

$$\begin{bmatrix} \mathbf{x}_1^{(n)} \\ \mathbf{y}_1^{(n)} \end{bmatrix} = \mathbf{R}_j \begin{bmatrix} \mathbf{x}_2^{(n)} \\ \mathbf{y}_2^{(n)} \end{bmatrix} + \mathbf{T}_j$$



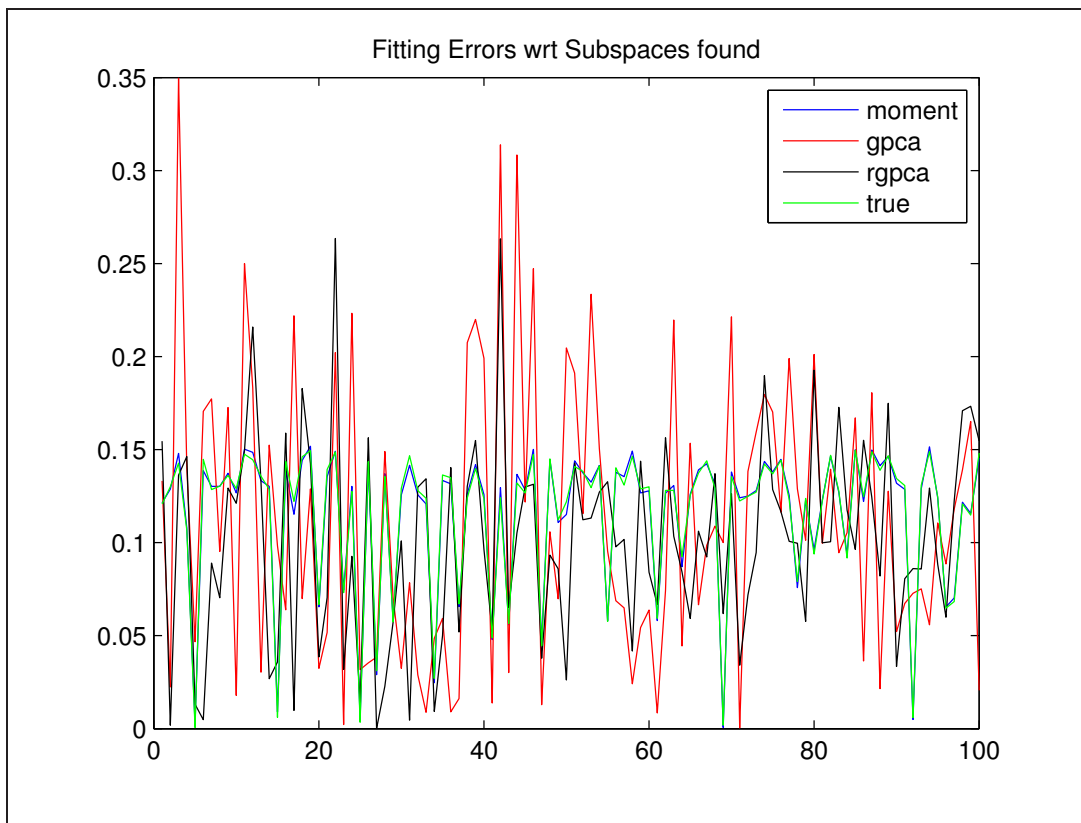


FIGURE 5.7: Fitting errors for GPCA, RGPCA and moments-based methods for the example in Fig. 5.6.

where  $(\mathbf{R}_j, \mathbf{T}_j)$  are the rotation and translation matrices of the motion of the  $j^{\text{th}}$  object. Rearranging shows that the vectors,  $\mathbf{f}^{(n)} = [\mathbf{x}_1^{(n)}, \mathbf{y}_1^{(n)}, \mathbf{x}_2^{(n)}, \mathbf{y}_2^{(n)}, 1]^T$  corresponding to points belonging to the same object  $j$  lay on a 3 dimensional subspace in  $\mathbb{R}^5$ . If multiple objects with different motions are present, each lays in a different subspace and thus, in the noiseless case, can be segmented using GPCA [8].

The image pair shown in Fig. 5.4 is taken from [84]. We manually marked the feature correspondences and then added uniform random noise to each feature point coordinate in both images. We formed the vectors  $\mathbf{f}^{(n)}$ , and projected them to  $\mathbb{R}^4$  using a  $5 \times 4$  matrix with orthonormal columns. Note that such a projection preserves the subspace clusters with probability 1. Specifically, we performed an SVD of the matrix  $\mathbf{F} = [\mathbf{f}^{(1)}, \dots, \mathbf{f}^{(N)}] = \mathbf{U}\mathbf{D}\mathbf{V}^T$  and projected the data using the first 4 columns of  $\mathbf{U}$  (i.e.  $\mathbf{X} = \mathbf{U}_{1:4}^T \mathbf{F}$ ). Finally, we applied our method, GPCA and RGPCA to cluster these 4-D vectors in  $\mathbf{X}$ . The results, along with a typical segmentation, are summarized in Fig. 5.5, showing that the moments-based algorithm yields a substantially lower misclassification rate.

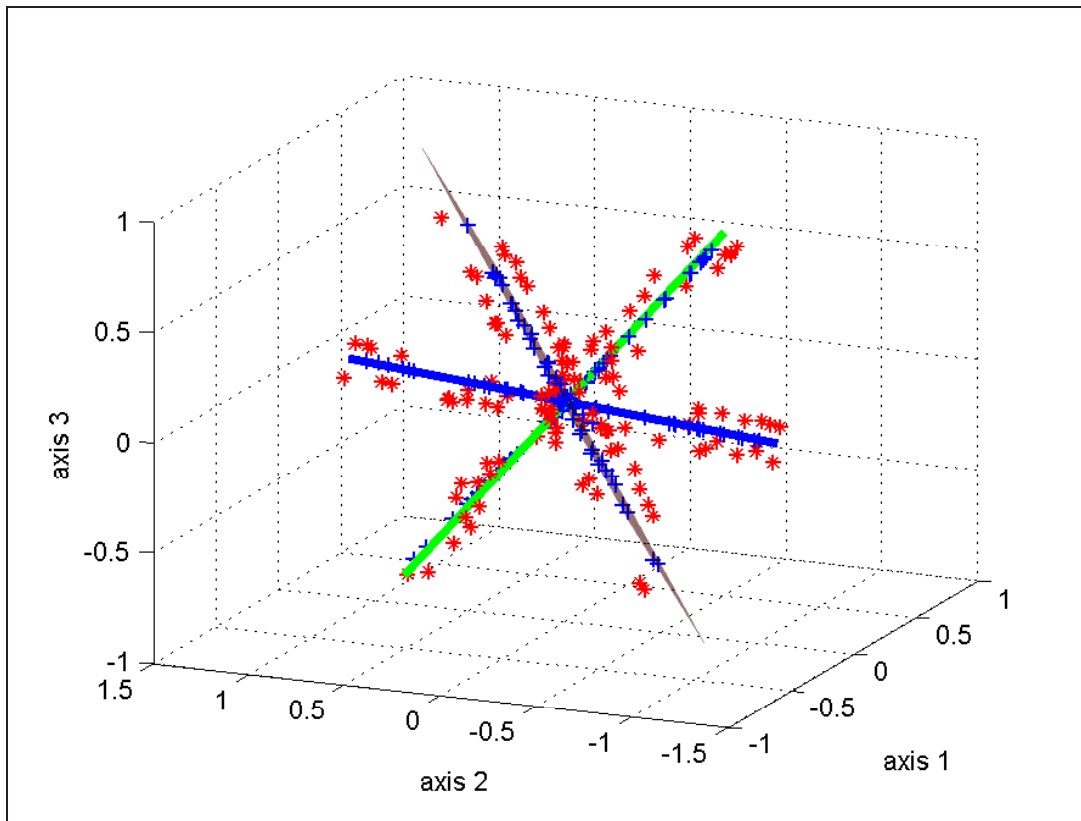


FIGURE 5.8: Example using synthetic data lying on two lines (blue and green) and a plane (transparent red) in  $\mathbb{R}^3$ . Here the red stars and blue plus signs indicate the original (noisy) and the denoised data points, respectively.

### 5.2.4.3 Two View Perspective Motion Segmentation

In this section, we demonstrate the performance of our method in the problem of motion segmentation from two perspective views and compare against hybrid quadratic surface analysis (HQSA) [1], using both synthetic and real data.

First, we artificially generated two teapots and projected 95 points from each teapot surface to the image plane using a perspective camera model. Then we moved both teapots to different locations and generated a second image proceeding in the same way. The two views used are shown in Fig. 5.10. Then, we ran 20 random trials by adding iid zero mean Gaussian noise with variance 0.15 to each image point in both images. The mean misclassification error for moments-based method is around 2% whereas the mean misclassification error for HQSA is 23%.

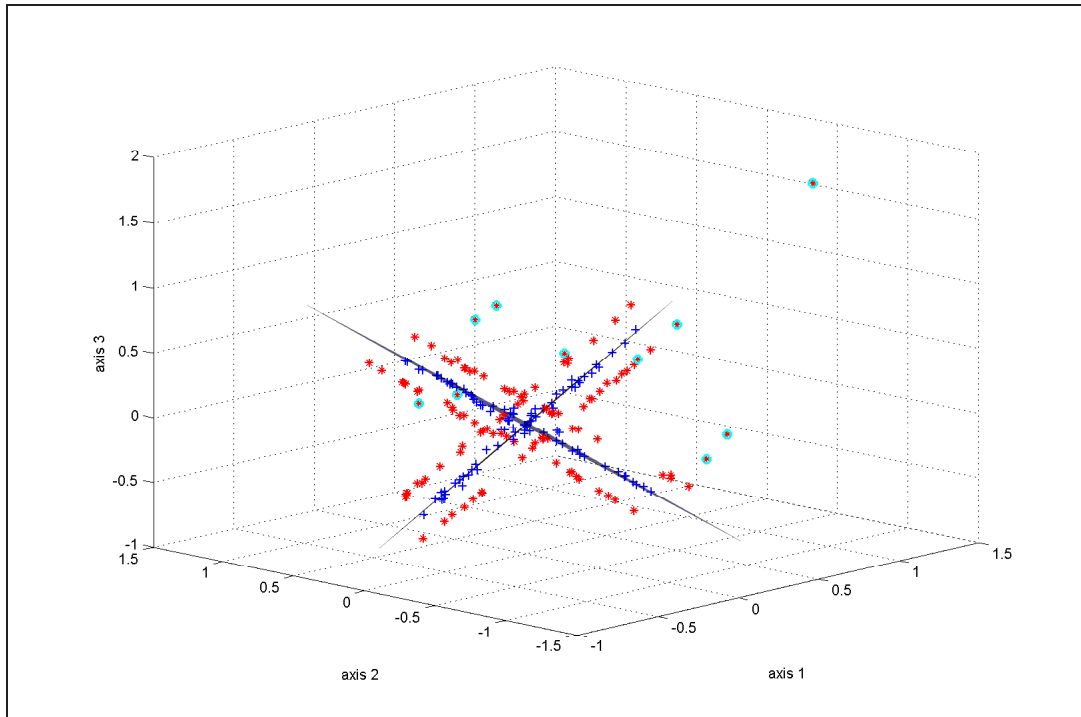


FIGURE 5.9: Example using synthetic data lying on two planes in  $\mathbb{R}^3$ . Here the red stars and blue plus signs indicate the original (noisy) and the denoised data points, respectively. Cyan circles indicate the points identified as outliers.

Next we consider two real examples. Figure 5.11 shows two piles of boxes that are moved in two different directions. This example is taken from [91]. We used the first and ninth frames of the sequence as our two perspective views. The missing tracks were ignored. In the resulting segmentation HQSA misclassifies 29 points whereas the moments-based approach misclassifies only 10 points. Since our model takes into account only additive noise and this is not the only source of error, the final classification is not perfect. However, the moments-based approach still outperforms HQSA. For the next example, we used as perspective views the first and last frames of the “truck2” sequence from the Hopkins 155 dataset [92]. In this case, the misclassification rate for the moments-based approach is 9.06% (it only misclassifies the left rear wheel as background). On the other hand, HQSA has a misclassification rate of 42.90%, clustering half of the points on the truck as background.

| Sequence | Moments-based | HQSA   |
|----------|---------------|--------|
| teapot   | 2.1%          | 23.42% |
| boxes    | 4.05%         | 11.74% |
| truck2   | 9.06%         | 42.90% |

TABLE 5.4: Misclassification rates for perspective motion segmentation examples.

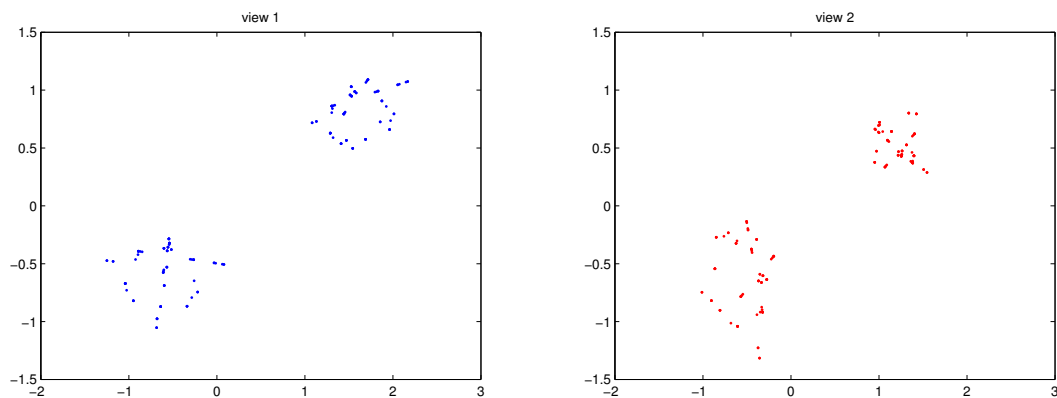


FIGURE 5.10: Two perspective images of points on teapot surfaces.

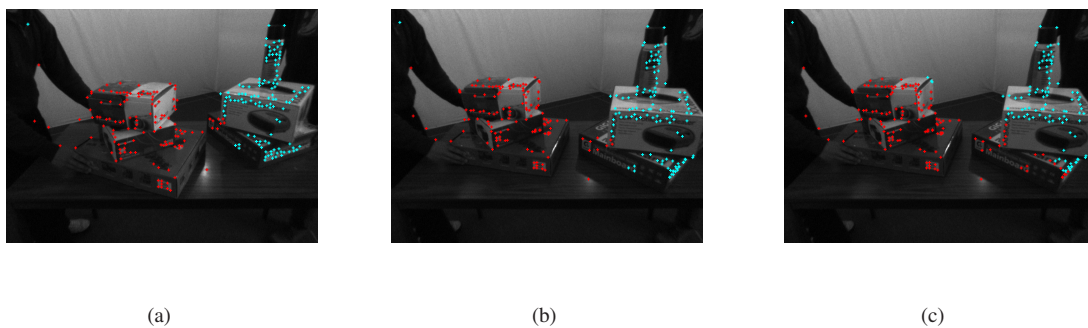


FIGURE 5.11: (a)-(b) First and second images with moments-based segmentation superimposed, 10 misclassified points. (c) Segmentation with the method in [1], 29 misclassified points.



FIGURE 5.12: (a)-(b) First and second images with moments-based segmentation superimposed ( 9.06% misclassification rate). (c) Segmentation with the method in [1] (42.90% misclassification rate).

## Chapter 6

# Conclusions and Future Work

We consider the problem of robustly identifying hybrid models from data and minimal *a priori* assumptions on the model sets and noise level. While this problem is known to be generically NP-hard, we show that efficient convex relaxations can be obtained by recasting it into an optimization form. Our main results show that, when an explanation with the minimum number of switches is sought (a problem relevant for instance in the context of segmentation), the problem can be recast into a sparsification form and efficiently solved using recently introduced relaxations. A similar idea can be also used when minimizing the number of submodels. However, in this case, while usually working well in practice, the sparsification approach is suboptimal. We then consider a related problem when the number of submodels is known. For this case we develop a moments-based convex optimization approach. We present parallel results for segmentation of subspace arrangements and more generally algebraic surfaces. The problem of model (in)validation for switched affine systems is also formulated as an optimization problem from which it is possible to get convex certificates. The advantages of the proposed techniques over existing methods are illustrated using both simulation examples and non-trivial problems arising in computer vision. As shown there, while most existing methods perform well in noiseless scenarios, sparsification-based or moments-based techniques are more robust to noise.

There are some open problems and possible extensions of the proposed methods that have not been addressed in this dissertation. Future work will include:

- *Convergence Analysis*: It is important to analyze the convergence properties of proposed identification schemes. We are currently trying to answer the following questions: “Under what conditions, does a given algorithm find the true system or segmentation?”; “Is there a noise

level, below which, the algorithms are guaranteed to converge?”. We have some preliminary convergence results for the greedy algorithm for minimum number of switches problem. Future work will include extending these results to other problems and algorithms.

- *Robust Control of Switched Linear Systems:* The ultimate goal in system identification is usually to obtain a model so that one can design controllers. There are some very recent results on control of switching linear systems [93] and  $\mathcal{L}_2$  induced norm characterization [94] of such systems. Another direction for future research is to exploit these results for robust control of switched linear systems.
- *Segmentation of 2-D Images and Arbitrary Graphs:* Similar to multidimensional extension in dynamical case, our results in static case can be extended to perform unsupervised segmentation of natural or medical images. The idea is to form an uncertain graph with nodes on each pixel and potential edges connecting each neighboring pixels. Each node can be associated with a value, which is an invariant of the segments (e.g. this could be the intensity if we presume that the intensity is almost constant within each segment, or a statistics derived from some local neighborhood of the pixel) plus an uncertainty. Then each potential edge can be associated with a weight that is the difference of the values of the two nodes connected by that edge. Sparsifying the weights of this graph will result in a segmentation with least complex boundary. We have some preliminary results in this direction where we used the intensity values of the low frequency components of the images as invariants. The future work will include extending these result to different frequency scales by identifying relevant invariants and also developing methods to reduce computational complexity so that it can be applied to larger images/graphs efficiently.
- *Improving Computational Efficiency:* Although there are efficient algorithms to solve generic LPs or SDPs, it is possible to achieve further speed-ups or solve larger-scale problems by using specially designed optimization algorithms that take into account the specific problem structure. Such algorithms start appearing both for  $\ell_1$ -norm minimization (sparsification) and nuclear norm minimization (rank minimization). In this thesis, we just used of-the-shelf convex programming solvers. In the future, we will incorporate the recently developed fast sparsification and rank minimization algorithms (with modifications if necessary) with the methods proposed in this thesis to further improve computational efficiency.

# Appendix A

## Sparsity Related Proofs

### A.1 Proof of Lemma 1

In order to prove the lemma, we need some preliminary results from convex analysis. For a function  $f : \mathcal{C} \rightarrow \mathbb{R}$ , where  $\mathcal{C} \subseteq \mathbb{R}^n$ , the conjugate  $f^*$  is defined as

$$f^*(y) = \sup_{x \in \mathcal{C}} (\langle x, y \rangle - f(x))$$

Under some technical conditions (see [95] Theorem 1.3.5), which are met here, the conjugate of the conjugate (i.e.  $f^{**}$ ) gives the convex envelope of the function  $f$ .

The proof proceeds now along the lines of that of the Theorem 1 in [32], by computing  $\|x\|_0^{**}$ ,  $x \in \mathcal{S}$ . The isomorphism  $\mathcal{I}$  from  $\mathcal{S}$  to  $\mathbb{R}^{d(T-t_o+1)}$ , which simply stacks the elements of the sequence into a column vector, naturally induces an inner product on  $\mathcal{S}$  as  $\langle x, y \rangle = \langle \mathcal{I}(x), \mathcal{I}(y) \rangle = \sum_{t=1}^T x^T(t)y(t)$ . For  $f : \mathcal{S} \rightarrow \mathbb{R}$ ,  $f(x) = \|x\|_0$ , the conjugate function in  $\mathcal{C} \doteq \|x\|_\infty \leq 1$  is:

$$\begin{aligned} f^*(y) &= \sup_{\|x\|_\infty \leq 1} \{\langle x, y \rangle - f(x)\} \\ &= \sum_{i \in \lambda} \|y(i)\|_1 - |\lambda| \end{aligned} \tag{A.1}$$

where  $\lambda = \{j : \|y(j)\|_1 > 1, j \in \{1, 2, \dots, T\}\}$  is an index set and  $|\lambda|$  is its cardinality.

$$\begin{aligned}
f^{**}(z) &= \sup_{y \in \mathcal{S}} \{\langle y, z \rangle - f^*(y)\} \\
&= \sup_{y \in \mathcal{S}} \left\{ \sum_{i \in \lambda} y(i)^T z(i) \right. \\
&\quad \left. + \sum_{i \notin \lambda} y(i)^T z(i) - \sum_{i \in \lambda} \|y(i)\|_1 + |\lambda| \right\} \\
&= \sup_{y \in \mathcal{S}} \left\{ \sum_{i \in \lambda} y(i)^T [z(i) - \text{sign}(y(i))] \right. \\
&\quad \left. + \sum_{i \notin \lambda} y(i)^T z(i) + |\lambda| \right\}
\end{aligned} \tag{A.2}$$

Here we consider two cases:

1) If  $\|z\|_\infty > 1$ , it is possible to choose  $y$  such that the first term in (A.2) grows unboundedly and  $f^{**}(z) \rightarrow \infty$ . So the domain of  $f^{**}$  is  $\|z\|_\infty \leq 1$ .

2) If  $\|z\|_\infty \leq 1$ , the first term in the last line of (A.2) is nonpositive. So to maximize the first term,  $y(i)$  values should be chosen small in absolute value for  $i \in \lambda$ . Keeping in mind the bounds imposed on  $y(i)$  values by  $\lambda$ , the maximum value of the second term is  $\sum_{i \notin \lambda} \|z(i)\|_\infty$ . Similarly,  $\sup_y \left\{ \sum_{i \in \lambda} y(i)^T [z(i) - \text{sign}(y(i))] + |\lambda| \right\} = \sum_{i \in \lambda} [\|z(i)\|_\infty - 1] + |\lambda| = \sum_{i \in \lambda} \|z(i)\|_\infty$ . Hence,

$$f^{**}(z) = \sum_{i=1}^T \|z(i)\|_\infty. \tag{A.3}$$

## A.2 Proof of Lemma 2

Let  $\tilde{\mathbf{g}}_1 = \tilde{\mathbf{g}}_o + \tilde{\mathbf{h}}$  be the solution of the relaxed problem (2.5) and let  $\mathcal{I}_o = \{t | \mathbf{g}_o(t) \neq 0\}$  be the indices of non-zeros elements. From the optimality of  $\tilde{\mathbf{g}}_1$ :

$$\sum_{t=1}^T \|\mathbf{g}_1(t)\|_\infty = \sum_{t=1}^T \|\mathbf{g}_o(t) + \mathbf{h}(t)\|_\infty = \sum_{t \in \mathcal{I}_o} \|\mathbf{g}_o(t) + \mathbf{h}(t)\|_\infty + \sum_{t \notin \mathcal{I}_o} \|\mathbf{h}(t)\|_\infty \leq \sum_{t=1}^T \|\mathbf{g}_o(t)\|_\infty \tag{A.4}$$

From the last inequality:

$$\begin{aligned}
\sum_{t \in \mathcal{I}_o} \|\mathbf{g}_o(t)\|_\infty - \sum_{t \in \mathcal{I}_o} \|\mathbf{h}(t)\|_\infty + \sum_{t \notin \mathcal{I}_o} \|\mathbf{h}(t)\|_\infty &\leq \sum_{t=1}^T \|\mathbf{g}_o(t)\|_\infty \\
\sum_{t \notin \mathcal{I}_o} \|\mathbf{h}(t)\|_\infty &\leq \sum_{t \in \mathcal{I}_o} \|\mathbf{h}(t)\|_\infty
\end{aligned} \tag{A.5}$$



Since both  $\tilde{\mathbf{g}}_1$  and  $\tilde{\mathbf{g}}_o$  are feasible,  $\mathbf{A}\tilde{\mathbf{h}} = 0$  which implies

$$\mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o} = -\mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o^c} \quad (\text{A.6})$$

where  $\tilde{\mathbf{h}}_{\mathcal{I}_o(\mathcal{I}_o^c)}$  is the restriction of  $\tilde{\mathbf{h}}$  to  $\mathcal{I}_o(\mathcal{I}_o^c)$ , complement of  $\mathcal{I}_o$ ). Consider a partition of  $\mathcal{I}_o^c$  in to disjoint sets of cardinality  $k$ ,  $\mathcal{I}_o^c = \cup_{j=1}^l \mathcal{I}_{o_j}^c$  where  $l \leq \lceil (T-k)/k \rceil$ . From Eq. (2.6), for all  $j$ , we have:

$$c_1 \|\tilde{\mathbf{h}}_{\mathcal{I}_o} \pm \tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2 \leq \|\mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o} \pm \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2 \leq c_2 \|\tilde{\mathbf{h}}_{\mathcal{I}_o} \pm \tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2 \quad (\text{A.7})$$

From parallelogram identity:

$$\begin{aligned} \langle \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o}, \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c} \rangle &= \frac{\|\mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o} + \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2 - \|\mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o} - \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2}{4} \\ &\leq \frac{c_2 \|\tilde{\mathbf{h}}_{\mathcal{I}_o} + \tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2 - c_1 \|\tilde{\mathbf{h}}_{\mathcal{I}_o} - \tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2}{4} \end{aligned} \quad (\text{A.8})$$

and

$$\begin{aligned} -\langle \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o}, \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c} \rangle &= \frac{\|\mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o} - \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2 - \|\mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o} + \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2}{4} \\ &\leq \frac{c_2 \|\tilde{\mathbf{h}}_{\mathcal{I}_o} - \tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2 - c_1 \|\tilde{\mathbf{h}}_{\mathcal{I}_o} + \tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2}{4} \\ &= \frac{(c_2 - c_1)(\|\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 + \|\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2)}{4} \end{aligned} \quad (\text{A.9})$$

From Eq. (A.6):

$$\begin{aligned} \|\mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 = -\langle \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o}, \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o^c} \rangle &= -\langle \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o}, \mathbf{A} \sum_j \tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c} \rangle \\ &= -\sum_{j=1}^l \langle \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o}, \mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c} \rangle \\ &\leq \sum_{j=1}^l \frac{(c_2 - c_1)(\|\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 + \|\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2)}{4} \\ &= \frac{c_2 - c_1}{4} (l \|\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 + \sum_{j=1}^l \|\tilde{\mathbf{h}}_{\mathcal{I}_{o_j}^c}\|_2^2) \\ &= \frac{c_2 - c_1}{4} (l \|\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 + \|\tilde{\mathbf{h}}_{\mathcal{I}_o^c}\|_2^2) \end{aligned} \quad (\text{A.10})$$

Now using the norm equivalence identities and eq. (A.5), we have

$$\begin{aligned} \|\tilde{\mathbf{h}}_{\mathcal{I}_o^c}\|_2^2 &= \sum_{t \in \mathcal{I}_o^c} \|\mathbf{h}(t)\|_2^2 \\ &\leq \sum_{t \in \mathcal{I}_o^c} d \|\mathbf{h}(t)\|_\infty^2 \\ &\leq d \left( \sum_{t \in \mathcal{I}_o^c} \|\mathbf{h}(t)\|_\infty \right)^2 \\ &\leq d \left( \sum_{t \in \mathcal{I}_o} \|\mathbf{h}(t)\|_\infty \right)^2 \\ &\leq d \left( \sum_{t \in \mathcal{I}_o} \|\mathbf{h}(t)\|_2 \right)^2 \\ &\leq dk \|\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 \end{aligned} \quad (\text{A.11})$$

Following from eq. (A.10):

$$\begin{aligned} c_1 \|\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 \leq \|\mathbf{A}\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 &\leq \frac{(c_2 - c_1)(l + dk)}{4} \|\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 \\ &\leq \frac{(c_2 - c_1)(\lceil \frac{T-k}{k} \rceil + dk)}{4} \|\tilde{\mathbf{h}}_{\mathcal{I}_o}\|_2^2 \end{aligned} \quad (\text{A.12})$$

If  $c_1 > \frac{(c_2 - c_1)(\lceil \frac{T-k}{k} \rceil + dk)}{4}$ , then we can conclude  $\tilde{\mathbf{h}}_{\mathcal{I}_o} = 0$  which implies  $\tilde{\mathbf{h}}_{\mathcal{I}_c} = 0$  and  $\tilde{\mathbf{g}}_1 = \tilde{\mathbf{g}}_o$ . Therefore, a sufficient condition for exactness is:

$$\frac{c_1}{c_2} > \frac{T + dk^2}{T + dk^2 + 4k} \quad (\text{A.13})$$

## Appendix B

# Recovering the Parameters of the Model in GPCA

Here we recall, for ease of reference, the polynomial differentiation based procedure proposed in [50] to recover the parameters of the model once  $\mathbf{c}_s$  is computed. The derivative of  $p_s(\mathbf{r})$  at a point  $\mathbf{r}$  is given by

$$Dp_s(\mathbf{r}) = \frac{\delta p_s(\mathbf{r})}{\delta \mathbf{r}} = \sum_{i=1}^s \prod_{j \neq i} (\mathbf{b}_j^T \mathbf{r}) \mathbf{b}_i \quad (\text{B.1})$$

Since  $\mathbf{b}_i^T \mathbf{r} = 0$  when  $\mathbf{r}$  is generated by the  $i^{\text{th}}$  submodel (i.e.  $\sigma_t(\mathbf{r}) = i$ ), it follows from (B.1) that the parameter vector is given by:

$$\mathbf{b}_i = \frac{Dp_s(\mathbf{r})}{\mathbf{e}^T Dp_s(\mathbf{r})} \Big|_{\sigma_t(\mathbf{r})=i} \quad (\text{B.2})$$

where  $\mathbf{e}^T = [1, 0, \dots, 0]$ .

Since, in general, the association of data points with submodels  $\sigma_t(\mathbf{r})$  is unknown, one can use the following heuristic function, suggested in [8], to choose one point from each submodel  $\{\mathbf{r}_{t_i}\}_{i=1}^s$ :

$$\mathbf{r}_{t_{i-1}} = \underset{\mathbf{r}_t: Dp_s(\mathbf{r}_t) \neq 0}{\operatorname{argmin}} \frac{\frac{|p_s(\mathbf{r}_t)|}{\|Dp_s(\mathbf{r}_t)\|} + \delta}{|(\mathbf{b}_1^T \mathbf{r}_t) \cdots (\mathbf{b}_s^T \mathbf{r}_t)| + \delta} \quad (\text{B.3})$$

where  $\delta > 0$  is a small number to avoid division by zero in noise free case.

Finally, given the parameter vectors  $\{\mathbf{b}_i\}_{i=1}^s$ , the mode signal can be computed as follows:

$$\sigma_t = \operatorname{argmin}_{i=1,\dots,s} (\mathbf{b}_i^T \mathbf{r}_t)^2. \quad (\text{B.4})$$

# Bibliography

- [1] S. Rao, A. Y. Yang, A. Wagner, and Y. Ma, “Segmentation of hybrid motions via hybrid quadratic surface analysis,” in *ICCV*, pp. 2–9, 2005.
- [2] R. Alur, C. Belta, F. Ivanicic, V. Kumar, M. Mintz, G. J. Pappas, H. Rubin, and J. Schug, “Hybrid modeling and simulation of biomolecular networks,” in *HSCC*, pp. 19–32, 2001.
- [3] H. de Jong, J.-L. Gouzé, C. Hernandez, M. Page, S. Tewfik, and J. Geiselmann, “Hybrid modeling and simulation of genetic regulatory networks: A qualitative approach,” in *HSCC*, pp. 267–282, 2003.
- [4] A. Bemporad, “Efficient conversion of mixed logical dynamical systems into an equivalent piecewise affine form,” *Automatic Control, IEEE Transactions on*, vol. 49, pp. 832–838, May 2004.
- [5] A. Bemporad and M. Morari, “Control of systems integrating logic, dynamics, and constraints,” *Automatica*, vol. 35, pp. 407–427, 1999.
- [6] D. Pepyne and C. Cassandras, “Optimal control of hybrid systems in manufacturing,” *Proceedings of the IEEE*, vol. 88, pp. 1108–1123, Jul 2000.
- [7] A. Balluchi, L. Benvenuti, M. di Benedetto, C. Pinello, and A. Sangiovanni-Vincentelli, “Automotive engine control and hybrid systems: challenges and opportunities,” *Proceedings of the IEEE*, vol. 88, pp. 888–912, Jul 2000.
- [8] R. Vidal, Y. Ma, and S. Sastry, “Generalized principal component analysis (gpca),” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 1945–1959, December 2005.
- [9] W. Hong, J. Wright, K. Huang, and Y. Ma, “Multiscale hybrid linear models for lossy image representation,” *IEEE Transactions on Image Processing*, vol. 15, pp. 3655–3671, December 2006.

- [10] Y. Ma, A. Yang, D. H., and F. R., “Estimation of subspace arrangements with applications in modeling and segmenting mixed data,” *SIAM Review*, vol. 50, pp. 413–458, 2008.
- [11] J. Ho, M.-H. Yang, J. Lim, K.-C. Lee, and D. J. Kriegman, “Clustering appearances of objects under varying illumination conditions,” in *CVPR (1)*, pp. 11–18, 2003.
- [12] M. Tipping and C. Bishop, “Mixtures of probabilistic principal component analysers,” *Neural Computation*, vol. 11, pp. 443–482, 1999.
- [13] A. B. Chan and N. Vasconcelos, “Mixtures of dynamic textures,” in *IEEE International Conference on Computer Vision*, vol. 1, pp. 641–647, 2005.
- [14] O. Rotem, H. Greenspan, and J. Goldberger, “Combining region and edge cues for image segmentation in a probabilistic gaussian mixture framework,” in *CVPR07*, pp. 1–8, 2007.
- [15] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge Univ. Press, 2004.
- [16] J. F. Sturm, “Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones,” 1999.
- [17] R. H. Tutuncu, K. C. Toh, and M. J. Todd, “Solving semidefinite-quadratic-linear programs using sdpt3,” *Mathematical Programming Ser. B*, vol. 95, pp. 189–217, 2003.
- [18] M. Grant and S. Boyd, “CVX: Matlab software for disciplined convex programming.” (web page and software), June 2009. <http://stanford.edu/~boyd/cvx>.
- [19] J. Löfberg, “Yalmip : A toolbox for modeling and optimization in MATLAB,” in *Proceedings of the CACSD Conference*, 2004.
- [20] D. L. Donoho, M. Elad, and V. N. Temlyakov, “Stable recovery of sparse overcomplete representations in the presence of noise,” *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, 2006.
- [21] J. A. Tropp, “Just relax: convex programming methods for identifying sparse signals in noise,” *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1030–1051, 2006.
- [22] E. J. Candès, J. K. Romberg, and T. Tao, “Stable signal recovery from incomplete and inaccurate measurements,” *Communications on Pure and Applied Mathematics*, vol. 59, pp. 1207–1223, August 2006.
- [23] B. K. Natarajan, “Sparse approximate solutions to linear systems,” *SIAM Journal on Computing*, vol. 24, no. 2, pp. 227–234, 1995.

- 
- [24] E. Amaldi and V. Kann, “On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems,” *Theoretical Computer Science*, vol. 209, no. 1–2, pp. 237–260, 1998.
- [25] J. A. Tropp, “Greed is good: algorithmic results for sparse approximation,” *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.
- [26] J. A. Tropp and A. C. Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit,” *IEEE Trans. Inform. Theory*, vol. 53, pp. 4655–4666, 2007.
- [27] D. Needell and R. Vershynin, “Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit,” *Foundations of Computational Mathematics*, vol. 9, no. 3, pp. 317–334, 2009.
- [28] A. M. Bruckstein, D. L. Donoho, and M. Elad, “From sparse solutions of systems of equations to sparse modeling of signals and images,” *SIAM Review*, vol. 51, no. 1, pp. 34 – 81, 2009.
- [29] Y. Zhang, “On theory of compressive sensing via  $\ell_1$ -minimization: Simple derivations and extensions,” tech. rep., Rice University, 2008.
- [30] Y. C. Eldar and M. Mishali, “Robust recovery of signals from a structured union of subspaces,” *Information Theory, IEEE Transactions on*, vol. 55, pp. 5302 –5316, nov. 2009.
- [31] M. Stojnic, F. Parvaresh, and B. Hassibi, “On the reconstruction of block-sparse signals with an optimal number of measurements,” *Signal Processing, IEEE Transactions on*, vol. 57, pp. 3075 –3085, aug. 2009.
- [32] M. Fazel, H. Hindi, and S. Boyd, “A rank minimization heuristic with application to minimum order system approximation,” in *American Control Conference*, 2001.
- [33] M. Fazel, *Matrix Rank Minimization with Applications*. PhD thesis, Stanford University, 2002.
- [34] M. Fazel, H. Hindi, and S. Boyd, “Rank minimization and applications in system theory,” in *American Control Conference*, pp. 3273–3278, 2004.
- [35] B. Recht, M. Fazel, and P. A. Parrilo, “Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization.” preprint.
- [36] M. Laurent, *Emerging Applications of Algebraic Geometry*, ch. Sums of squares, moment matrices and optimization over polynomials, pp. 157–270. Springer, 2009.

- [37] J. A. Shohat and J. D. Tamarkin, *The Problem of Moments*. Math. Surveys I, Providence, RI: American Mathematical Society, 1943.
- [38] M. G. Krein and A. A. Nudelman, *The Markov Moment Problem and Extremal Problems*, vol. 50 of *Translations of Mathematical Monographs*. Providence, RI: American Mathematical Society, 1977.
- [39] R. E. Curto and L. A. Fialkow, “Recursiveness, positivity, and truncated moment problems,” *Houston J. Math*, vol. 17, pp. 603–635, 1991.
- [40] J. L. McGregor, “Solvability criteria for certain n-dimensional moment problems,” *Journal of Approximation Theory*, vol. 30, pp. 315–333, 1980.
- [41] J. Lasserre, “Global optimization with polynomials and the problem of moments,” *SIAM J. Optimization*, vol. 11, pp. 796–817, 2001.
- [42] R. E. Curto and L. A. Fialkow, “Truncated k-moment problems in several variables,” *Journal of Operator Theory*, vol. 54, no. 1, pp. 189–226, 2005.
- [43] K. Schmüdgen, “The k-moment problem for compact semi-algebraic sets,” *Mathematische Annalen*, vol. 289, no. 1, pp. 203–206, 1991.
- [44] J. Lasserre, “Convergent sdp-relaxations in polynomial optimization with sparsity,” *SIAM J. Optimization*, vol. 17, no. 3, pp. 822–843, 2006.
- [45] H. Waki, S. Kim, M. Kojima, and M. Muramatsu, “Sums of squares and semidefinite program relaxations for polynomial optimization problems with structured sparsity,” *SIAM J. on Optimization*, vol. 17, no. 1, pp. 218–242, 2006.
- [46] S. Paoletti, A. Juloski, G. Ferrari-Trecate, and R. Vidal, “Identification of hybrid systems: A tutorial,” *European Journal of Control*, vol. 13, no. 2, pp. 242–260, 2007.
- [47] Y. Ma and R. Vidal, “A closed form solution to the identification of hybrid arx models via the identification of algebraic varieties,” in *Hybrid Systems Computation and Control*, pp. 449–465, March 2005.
- [48] A. Bemporad, A. Garulli, S. Paoletti, and A. Vicino, “A bounded-error approach to piecewise affine system identification,” *IEEE Transactions on Automatic Control*, vol. 50, no. 10, pp. 1567–1580, 2005.



- [49] J. Roll, A. Bemporad, and L. Ljung, "Identification of pieewise affine systems via mixed-integer programming," *Automatica*, vol. 40, pp. 37–50, 2004.
- [50] R. Vidal, S. Soatto, Y. Ma, and S. Sastry, "An algebraic geometric approach to the identification of linear hybrid systems," in *IEEE Conference on Decision and Control*, pp. 167–172, Dec. 2003.
- [51] M. Fazel, H. Hindi, and S. Boyd, "Log-det heuristic for matrix rank minimization with applications to hankel and euclidean distance matrices," in *American Control Conference*, 2003.
- [52] J. Lygeros, K. H. Johansson, S. N. Simic, J. Zhang, and S. S. Sastry, "Dynamical properties of hybrid automata," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 2–17, 2003.
- [53] Z. Qu and P. Perron, "Estimating and testing structural changes in multivariate regressions," *Econometrica*, vol. 75, pp. 459–502, 2007.
- [54] J. P. Hespanha, "Uniform stability of switched linear systems: Extensions of lasalle's invariance principle," *IEEE Transactions on Automatic Control*, vol. 49, no. 4, pp. 470–482, 2004.
- [55] A. Gionis and H. Mannila, "Segmentation algorithms for time series and sequence data," in *SIAM International Conference on Data Mining*, 2005. Tutorial.
- [56] M. Lobo, M. Fazel, and S. Boyd, "Portfolio optimization with linear and fixed transaction costs," *Annals of Operations Research*, vol. 152, no. 1, pp. 376–394, 2007.
- [57] E. J. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted l1 minimization," tech. rep., California Institute of Technology, 2007.
- [58] J. W. Woods, *Multidimensional Signal, Image and Video Processing and Coding*. Academic Press, 2006.
- [59] R. Vidal, "Identification of spatial-temporal switched arx systems," in *IEEE Conference on Decision and Control*, pp. 4675–4680, Dec. 2007.
- [60] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, vol. 66, pp. 846–850, 1971.
- [61] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley & Sons, Inc., second ed.
- [62] L. Cooper, J. Liu, and K. Huang, "Spatial segmentation of temporal texture using mixture linear models," in *Workshop on Dynamical Vision*, pp. 142–150, 2005.

- [63] U. Gargi, R. Kasturi, and S. H. Strayer, "Performance characterization of video-shot-change detection methods," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 1, pp. 1–13, 2000.
- [64] B.-L. Y. and B. Liu, "A unified approach to temporal segmentation of motion jpeg and mpeg compressed video," in *International Conference on Multimedia Computing and Systems*, pp. 81–88, May 1995.
- [65] E. J. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted  $l_1$  minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, 2008.
- [66] N. Ozay, M. Sznaier, C. Lagoa, and O. Camps, "A sparsification approach to set membership identification of a class of affine hybrid systems," in *Proc. 47<sup>th</sup> IEEE Conf. Dec. Control*, pp. 123–130, 2008.
- [67] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari, "A clustering technique for the identification of piecewise affine systems," *Automatica*, vol. 39, pp. 205–217, 2003.
- [68] N. Ozay, C. Lagoa, and M. Sznaier, "Robust identification of switched affine systems via moments-based convex optimization," in *Proc. 48<sup>th</sup> IEEE Conf. Dec. Control*, pp. 4686–4691, 2009.
- [69] C. Feng, C. Lagoa, and M. Sznaier, "Hybrid system identification via sparse polynomial optimization," in *American Control Conference*, 2010. (to appear).
- [70] K. Poola, P. Khargonekar, A. Tikku, J. Krause, and K. Nagpal, "A time domain approach to model validation," *IEEE Transactions on Automatic Control*, vol. 39, no. 5, pp. 951–959, 1994.
- [71] J. Chen, "Frequency-domain tests for validation of linear fractional uncertain models," *IEEE Transactions on Automatic Control*, vol. 42, pp. 748–760, June 1997.
- [72] R. Sánchez Peña and M. Sznaier, *Robust Systems Theory and Applications*. Wiley & Sons, Inc., 1998.
- [73] M. Sznaier and M. C. Mazzaro, "An lmi approach to control-oriented identification and model (in) validation of lpv systems," *Automatic Control, IEEE Transactions on*, vol. 48, pp. 1619–1624, sept. 2003.
- [74] F. Bianchi and R. Sánchez Peña, "Robust identification/invalidation in an lpv framework," *International Journal of Robust and Nonlinear Control*, vol. 20, pp. 301–312, Mar. 2009.

- [75] S. Prajna, “Barrier certificates for nonlinear model validation,” *Automatica*, vol. 42, no. 1, pp. 117–126, 2006.
- [76] P. A. Parrilo, “Semidefinite programming relaxations for semialgebraic problems,” *Mathematical Programming Ser. B*, vol. 96, no. 2, pp. 293–320, 2003.
- [77] J. Yuan, H. Wang, L. Xiao, W. Zheng, J. Li, F. Lin, and B. Zhang, “A formal study of shot boundary detection,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, pp. 168–186, February 2007.
- [78] M. Osian and L. Van Gool, “Video shot characterization,” *Machine Vision and Applications*, vol. 15, pp. 172–177, July 2004.
- [79] N. Petrovic, A. Ivanovic, and N. Jojic, “Recursive estimation of generative models of video,” in *CVPR06*, pp. I: 79–86, 2006.
- [80] L. Lu and R. Vidal, “Combined central and subspace clustering for computer vision applications,” in *International Conference on Machine Learning*, pp. 593–600, 2006.
- [81] G. Doretto, A. Chiuso, Y. Wu, and S. Soatto, “Dynamic textures,” *IJCV*, vol. 51, pp. 91–109, February 2003.
- [82] A. Ghoreyshi and R. Vidal, “Segmenting dynamic textures with ising descriptors, arx models and level sets,” in *WDV06*, pp. 127–141, 2006.
- [83] J. Ho, M.-H. Yang, J. Lim, K.-C. Lee, and D. J. Kriegman, “Clustering appearances of objects under varying illumination conditions,” in *CVPR (1)*, pp. 11–18, 2003.
- [84] O. Tuzel, R. Subbarao, and P. Meer, “Simultaneous multiple 3d motion estimation via mode finding on lie groups,” in *ICCV*, pp. 18–25, 2005.
- [85] R. Orsi, U. Helmke, and J. B. Moore, “A newton-like method for solving rank constrained linear matrix inequalities,” *Automatica*, vol. 42, pp. 1875–1882, 2006.
- [86] J. F. Cai, E. J. Candes, and Z. Shen, “A singular value thresholding algorithm for matrix completion.” preprint.
- [87] D. Henrion and J.-B. Lasserre, *Positive Polynomials in Control*, ch. Detecting global optimality and extracting solutions in GloptiPoly, pp. 293–310. Springer, 2005.

- 
- [88] R. Vidal, Y. Ma, S. Soatto, and S. Sastry, “Two-view multibody structure from motion,” *IJCV*, vol. 68, no. 1, pp. 7–25, 2006.
- [89] K. Schindler and D. Suter, “Two-view multibody structure-and-motion with outliers through model selection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 6, pp. 983–995, 2006.
- [90] P. Anandan and S. Avidan, “Integrating local affine into global projective images in the joint image space,” in *ECCV (1)*, pp. 907–921, 2000.
- [91] K. Schindler, D. Suter, and H. Wang, “A model selection framework for multibody structure-and-motion of image sequences,” *IJCV*, vol. 79, no. 2, pp. 159–177, 2008.
- [92] R. Tron and R. Vidal, “A benchmark for the comparison of 3-d motion segmentation algorithms,” in *CVPR*, 2007.
- [93] F. Blanchini, S. Miani, and F. Mesquine, “A separation principle for linear switching systems and parametrization of all stabilizing controllers,” *Automatic Control, IEEE Transactions on*, vol. 54, pp. 279–292, Feb. 2009.
- [94] K. Hirata and J. P. Hespanha, “ $\mathcal{L}_2$ -induced gain analysis for a class of switched systems.” Submitted to conference publication, Mar. 2009.
- [95] J.-B. Hiriart-Urruty and C. Lemarechal, *Convex Analysis and Minimization Algorithms II: Advanced Theory and Bundle Methods*, vol. 306 of *Grundlehrer der mathematischen Wissenschaften*. Springer-Verlag, 1993.

# Vita

Necmiye Ozay received the B.S. degree in Electrical and Electronics Engineering from Bogazici University, Istanbul in 2004 and the M.S. degree in Electrical Engineering from the Pennsylvania State University, University Park, PA in 2006. Currently she is a Ph.D. candidate at Electrical and Computer Engineering Department at Northeastern University, Boston, MA. In summer 2008, she was a research intern at GE Global Research, Niskayuna, NY. She has also held short term visiting positions at Sabanci University, Istanbul in 2005 and Polytechnic University of Catalunya, Barcelona in 2008. Her research interests lie at the broad interface of system identification, convex optimization, control theory, and computer vision. She received the IEEE Control Systems Society Conference on Decision and Control Best Student Paper Award in 2008 and the IEEE Computer Society Biometrics Workshop Best Paper Honorable Mention Award in 2009. She is a student member of the IEEE and SIAM.

# List of Publications

The following list includes all the papers published or submitted for publication by the author during her graduate studies. Papers denoted by \* are directly related to the line of research presented in this dissertation.

\*N. Ozay, M. Sznaier, and C. M. Lagoa, “Model (In)validation of Switched ARX Systems with Unknown Switches and its Application to Activity Monitoring”, Submitted for conference publication, March 2010.

\*C. Feng, C. M. Lagoa, N. Ozay, and M. Sznaier, “Hybrid System Identification: An SDP Approach”, Submitted for conference publication, March 2010.

S. Kurugol, N. Ozay, J. G. Dy, G. C. Sharp, and D. Brooks, “Locally Deformable Shape Model to Improve 3D Level Set based Esophagus Segmentation”, *Proc. 20th International Conference on Pattern Recognition (ICPR)*, Istanbul, August 2010. (to appear)

\*N. Ozay, M. Sznaier, C. M. Lagoa, and O. Camps, “GPCA with Denoising: A Moments-Based Convex Approach”, *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, CA, June 2010. (to appear)

\*N. Ozay, M. Sznaier, C. M. Lagoa, and O. Camps, “A Sparsification Approach to Set Membership Identification of Switched Affine Systems”, Submitted for journal publication, May 2009.

\*M. Sznaier, O. Camps, N. Ozay, T. Ding, G. Tadmor and D. Brooks, “The role of dynamics in extracting information sparsely encoded in high dimensional data streams”, in *Dynamics of Information Systems: Theory and Applications* (eds. M.J. Hirsch, P.M. Pardalos and R. Murphey), Springer 2010.

\*N. Ozay, C. M. Lagoa, and M. Sznaier, "Robust identification of switched affine systems via moments-based convex optimization", *Proc. 48th IEEE Conference on Decision and Control (CDC)*, Shanghai, P.R. China, December 2009.

N. Ozay, Y. Tong, F. W. Wheeler, and X. Liu, "Improving Face Recognition with a Quality-based Probabilistic Framework", *Proc. IEEE Computer Society Workshop on Biometrics* (in conjunction with CVPR 2009), pages 134-141, June 2009.

\*N. Ozay, M. Sznaier, C. M. Lagoa, and O. Camps, "A Sparsification Approach to Set Membership Identification of a Class of Affine Hybrid Systems", *Proc. 47th IEEE Conference on Decision and Control (CDC)*, pages 123-130, December 2008.

\*N. Ozay, M. Sznaier, and O. Camps, "Sequential Sparsification for Change Detection", *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, Alaska, June 2008.

N. Ozay, and M. Sznaier, "A Pessimistic Approach to Frequency Domain Model (In)Validation", *Proc. 46th IEEE Conference on Decision and Control (CDC)*, pages 4895-4900, December 2007.

M. Sznaier, C. M. Lagoa, and N. Ozay, "Risk-adjusted output feedback receding horizon control of constrained linear parameter varying systems", *International Journal of Robust and Nonlinear Control* (Special issue on Nonlinear Model Predictive Control), 17(17):1614-1633, 2007.

M. Sznaier, C. M. Lagoa, and N. Ozay, "Risk-adjusted output feedback receding horizon control of constrained linear parameter varying systems", *Proc. European Control Conference (ECC)*, Kos, Greece, July 2007.

R. Lubliner, N. Ozay, D. Zarpalas, and O. Camps, "Activity Recognition from Silhouettes Using Linear Systems and Model (In)validation Techniques", *Proc. 18th International Conference on Pattern Recognition (ICPR)*, pages 347-350, August 2006.