Nonparametric identification of linear time-varying systems

Péter Zoltán Csurcsia

Engineers and scientists want a reliable mathematical model of the observed phenomenon for understanding, design and control. System identification is a tool which allows the user to build models of dynamic systems from experimental noisy data. This is an interdisciplinary science which connects the world of control theory, data acquisition, signal processing, statistics, time series analysis and many other areas.

In modeling and measurement techniques it is commonly assumed that the observed systems are linear time-invariant. This point of view is acceptable as long as the time variations of the systems are negligible. However, in some cases, this assumption is not satisfied and it leads to a very low accuracy of the estimates. In those cases, advanced modelling is needed taking into account the time-varying behavior of the model. In this thesis a very important class of systems, namely, the linear time varying systems are considered.

The importance of these systems can be seen through some application examples. A good example from the electrical field is, for example a non-compensated transistor (in an operational amplifier) with a shifting offset-voltage: the higher the temperature, the higher the offset drift. The offset variations influence the system parameters and result in a time-varying behavior. The changing bio-impedance in the heart is also a good example from biomedical sciences. In chemistry, an interesting example can be the impedance changing due to the pitting corrosion in metals.

It is already shown that LTV systems can be described by a two dimensional impulse response function. The challenge is that the time-varying two dimensional impulse response functions are not uniquely determined from a single set of input and output signals – like in the case of linear time invariant systems. Due to this non-uniqueness, the number of possible solutions is growing quadratically with the number of samples.

To decrease the degrees of freedom, user-defined adjustable constraints will be imposed. This will be implemented by using two different approaches. First, a special two dimensional regularization technique is applied. The second implementation technique uses generalized two dimensional smoothing B-splines. Using the proposed methods high quality models can be built.

This thesis involves the theoretical and implementational questions of the time-varying system identification.
Nonparametric identification of linear time-varying systems

Thesis submitted in fulfilment of the requirements for the degree of Doctor in Engineering (Doctor in de Ingenieurswetenschappen) and for the PhD degree of the Budapest University of Technology and Economics by

Péter Zoltán Csurcsia

Public defense: 27th October 2015  Private defense: 17th September 2015

Advisors:  Prof. dr. ir. Johan Schoukens
           Vrije Universiteit Brussel

           Prof. dr. ir. István Kollár
           Budapest University of Technology and Economics

Jury:  Prof. dr. ir. István Vajk
       Budapest University of Technology and Economics

       Prof. dr. ir. Keith R. Godfrey
       University of Warwick

       Prof. dr. ir. Jérôme Antoni
       University of Lyon

       Prof. dr. ir. Steve Vanlanduit president
       University of Antwerp, Vrije Universiteit Brussel

       Prof. dr. ir. Johan Deconinck vice-president
       Vrije Universiteit Brussel

       Dr. ir. John Lataire secretary
       Vrije Universiteit Brussel
ACKNOWLEDGMENTS

First of all I must thank my supervisors Johan Schoukens and István Kollár. They always found some time to have a good conversation and supporting me with some nice ideas.

Special thanks go to John Lataire, Georgios Birpoutsoukis and Ivan Markovsky. We had many nice discussions regarding to my (sometimes quite strange and weird) ideas.

I thank the (former and current) colleagues – who became my friends as well – for the warm and inspiring atmosphere at the department that contributed to a pleasant research.

I also would like to thank József Kohut, my college mentor, and László Nagy (†), Valéria Molnár (†), my secondary school mentors. They supported and guided me to be a scientist from an average student.

I would like to thank the members of the jury for their precious time reading my thesis and for their constructive comments given before, during and after the private defense.

This doctoral research was made possible thanks to the financial support from the Flemish government via the Methusalem project.
I started my PhD research in the fall of 2010 with Prof. István Kollár at the Budapest University of Technology and Economics, Department of Measurement and Information Systems.

In 2011, during the Spring Doctoral School on Identification of Nonlinear Dynamic Systems organized by the Department of Fundamental Electricity and Instrumentation (ELEC) at the Vrije Universiteit Brussel, I got a great opportunity to collaborate with ELEC as a PhD researcher under the supervision of Prof. Johan Schoukens.

From 2011 on I focused on different smoothing techniques. The basic idea was to use them for the identification of linear time-varying systems. As the result of this research, I developed a modified B-spline technique, which can be used to estimate time-varying systems in the time domain.

In order to follow the fashion in the system identification, from 2013 on I studied and analyzed the regularization technique (as a special viewpoint of the Bayesian statistical framework). In one and a half years I was able to develop a complex methodology to estimate time-varying systems. As a surprising result, the newly developed technique beats the B-spline technique in terms of performance. For that reason, this thesis discusses first the regularization approach, then the B-spline approach.
The reader is expected to have a background in engineering and to know the basics of signal processing, systems and signals, linear algebra and statistics.

To support the reader, the most important and relevant notions related to the systems and signals including the basics of the linear time-varying systems and a brief introduction to system identification are provided in the first part “Preliminaries”.

The reader will learn the basics of time-varying systems, the regularization techniques and that B-splines can be used for system identification purposes as well.

To guide the reader, simple and straightforward steps will lead to the proposed estimation methods starting from simple models and basic assumptions. The long and complicated derivations and proofs can be found in the appendices.

This thesis involves questions of theory and of implementation of time-varying system identification and intended to provide ready-to-use solutions for the practitioners as well. Using the proposed methods high quality models can be built.

I hope You, the reader of my thesis will enjoy my work.

Brussels, July 3rd, 2015

Péter
# Table of Contents

Acknowledgments ........................................................................................................ i  
Preface ......................................................................................................................... ii  
Table of Contents ......................................................................................................... iv  
Operators ....................................................................................................................... xi  
Symbols ........................................................................................................................ xii  
Abbreviations ................................................................................................................. xv  

Preliminaries .................................................................................................................. 1  

| Chapter 1 | Introduction | ............................................... | 3  
|------------|--------------|---------------------------------|-----  
| 1.1        | Objectives   | ............................................... | 5  
| 1.2        | Outline      | ............................................... | 6  

| Chapter 2 | Signals and systems | ................................................. | 9  
|------------|---------------------|---------------------------------|-----  
| 2.1        | Signals             | ............................................... | 9  
| 2.1.1      | Continuous and discrete time signals | ................................................ | 10  
| 2.1.2      | Deterministic and stochastic signals | ................................................ | 10  
| 2.2        | Systems             | ............................................... | 11  
| 2.2.1      | Linearity           | ............................................... | 11  
| 2.2.2      | Time-invariant systems | ........................................... | 11  
| 2.2.3      | Time-variant systems | ........................................... | 12  
| 2.2.4      | Smooth systems      | ............................................... | 16  
| 2.3        | The scope of the thesis | ................................................ | 17  

| Chapter 3 | An introduction to system identification | ............................................... | 19  

iv
Nonparametric identification using B-splines .............................................. 87

Chapter 8  
An introduction to B-splines ................................................................. 89

8.1 One dimensional B-Splines .............................................................. 90

8.2 Two dimensional B-splines .............................................................. 95

Chapter 9  
Time domain nonparametric estimate of linear time-varying systems using B-splines ................................................................. 97

9.1 The model ...................................................................................... 98

9.2 The choice of the cost function .................................................. 100

9.2.1 The cost function ...................................................................... 100

9.2.2 Observations .......................................................................... 101

9.3 Tuning of the model complexity ................................................ 101

9.4 An example ............................................................................... 103

9.4.1 Simulation Setup ...................................................................... 103

9.4.2 Results of the estimation ....................................................... 104

9.5 Convert the B-spline LTV kernel to the two dimensional impulse response form ............................................................................................. 106

9.5.1 The conversion ........................................................................ 106

9.5.2 A simulation example .......................................................... 107

9.6 Transient elimination .................................................................... 109

9.6.1 The proposed transient elimination technique ..................... 109

9.6.2 An example ........................................................................... 110

9.7 Conclusions .............................................................................. 113

Chapter 10  
Frozen nonparametric estimate of linear time-varying systems using B-splines ........................................................................................ 115

10.1 A spectral description of linear time-varying systems ........... 116

10.2 The proposed frozen methods .................................................. 119

10.3 The frequency domain approach ................................................ 120

10.3.1 The model ........................................................................... 120

10.3.2 Obtain the frozen transfer function ................................... 122

10.3.3 Post-processing ................................................................. 125

10.4 Time domain approach ............................................................ 130
10.4.1 The model ............................................................... 130
10.4.2 Obtain the frozen impulse response function .......... 130
10.4.3 Post-processing ..................................................... 132
10.5 Example .................................................................. 133
10.6 Conclusions ............................................................ 137

Chapter 11 An experimental comparison of the proposed methods based on the measurement of a time-varying system .......................................................... 139
11.1 The measurement setup .............................................. 140
11.1.1 The underlying system ......................................... 140
11.1.2 Instrumentation .................................................... 141
11.1.3 Excitation and scheduling signals ......................... 141
11.2 The estimation procedure .......................................... 142
11.3 Results and conclusions ........................................... 143

Conclusions ................................................................... 147

Chapter 12 Conclusions .................................................... 149
12.1 Main contributions of this thesis and scientific statements ...... 150
12.1.1 Regularization technique for LTV Systems ................ 150
12.1.2 B-spline technique used for Linear Time-variant Systems ... 151
12.1.3 A comparison between different LTV approaches .......... 153
12.2 Contributions presented elsewhere but not included in this thesis.. ......................................................... 154
12.2.1 Nonlinearities ....................................................... 154
12.2.2 B-spline based LTI identification ............................ 154
12.2.3 Research activities not related to system identification .... 156
12.3 Future research ........................................................ 157
12.3.1 Extend the methods to linear parameter-varying systems.... 157
12.3.2 Formulate the problem in the frequency domain .......... 158
12.3.3 Development of a non-equidistant B-spline algorithm ...... 158
12.3.4 New kernels for regularization .................................. 158
12.3.5 Speeding up the regularization .................................. 159
12.3.6 Closed-loop identification method............................ 159
12.3.7 Extension to periodicity .................................................... 159
12.3.8 Regularization of B-spline kernels ..................................... 159
12.4 List of publications ................................................................. 160

Appendices ............................................................................................................ 165

Appendix A. ........................................................................................................... 167
A.1 Derivation of mean square error of model fitting ...................... 167
A.2 Derivation of the regularized time-varying least squares estimation from the cost function ................................................................. 168
A.3 The regularized estimate and the maximum a posteriori estimation ... ............................................................................................................. 169
A.3.1 Equivalence ........................................................................ 169
A.3.2 The optimal choice of the covariance hypermatrix ............ 170
A.3.3 Tuning the hyperparameters via a Bayesian view ............... 173
A.4 Computational complexity and memory needs ...................... 174
A.4.1 Computational complexities ........................................... 174
A.4.2 Memory needs ................................................................. 175
A.5 Degrees of freedom ................................................................. 176
A.5.1 General case ................................................................. 176
A.5.2 Realistic case ................................................................. 177
A.5.3 Special case ................................................................. 178
A.5.4 Regularized case ............................................................. 178
A.6 Computer configuration used for calculating the estimations...... 181

Appendix B. ........................................................................................................... 183
B.1 Derivation of the B-splines time-varying least squares estimation from the cost function ................................................................. 183
B.2 Derivation of the B-spline surface fitting cost function (complex case) ................................................................................................. 184
B.3 Statistical properties of the proposed two dimensional non-frozen B-spline LTV estimator ................................................................. 186
B.4 Computational complexity and memory needs ...................... 188
B.4.1 Computational complexities ........................................... 188
B.4.2 Memory needs ................................................................. 188
B.5 Degrees of freedom .......................................................... 189
B.6 Penalized B-splines ............................................................ 190
  B.6.1 Introduction ............................................................... 190
  B.6.2 The cost function from the viewpoint of identification .... 190
  B.6.3 Bias and variance trade-off ........................................... 191
  B.6.4 Example .................................................................... 192

Bibliography ........................................................................... 195
## Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>\cdot</td>
</tr>
<tr>
<td>${\cdot}^2$</td>
<td>Componentwise matrix squaring</td>
</tr>
<tr>
<td>$|\cdot|_2$</td>
<td>L2 (Frobenius) norm</td>
</tr>
<tr>
<td>$\mathcal{F}_D{\cdot}$</td>
<td>Discrete Fourier transform</td>
</tr>
<tr>
<td>$\mathcal{F}_D^{-1}{\cdot}$</td>
<td>Inverse discrete Fourier transform</td>
</tr>
<tr>
<td>$\bar{x}$</td>
<td>Complex conjugate of $x$</td>
</tr>
<tr>
<td>$x^\dagger$</td>
<td>Moore-Penrose pseudo inverse of matrix $x$</td>
</tr>
<tr>
<td>$x^T$</td>
<td>Transpose of a vector/matrix $x$</td>
</tr>
<tr>
<td>$x^H$</td>
<td>Hermitian transpose of a vector/matrix $x$</td>
</tr>
<tr>
<td>$</td>
<td>\cdot</td>
</tr>
<tr>
<td>$\Im{\cdot}$</td>
<td>Imaginary part of a complex number</td>
</tr>
<tr>
<td>$\mathcal{L}{\cdot}$</td>
<td>Laplace transform</td>
</tr>
<tr>
<td>$\Re{\cdot}$</td>
<td>Real part of a complex number</td>
</tr>
<tr>
<td>$\text{COV}{\cdot,\cdot}$</td>
<td>Covariance function</td>
</tr>
<tr>
<td>$\text{CR}(\cdot)$</td>
<td>Crest factor value</td>
</tr>
<tr>
<td>$\text{det}(\cdot)$</td>
<td>Determinant</td>
</tr>
<tr>
<td>$\text{MSE}{\cdot}$</td>
<td>Mean square error</td>
</tr>
<tr>
<td>$\text{rms}(\cdot)$</td>
<td>Root mean square value</td>
</tr>
<tr>
<td>$\text{SVD}{\cdot}$</td>
<td>Singular value decomposition</td>
</tr>
</tbody>
</table>
**SYMBOLS**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B_i^{d_T} / B_j^{d_T}$</td>
<td>A matrix containing the B-spline basis functions</td>
</tr>
<tr>
<td>$\mathbb{C}$</td>
<td>Complex numbers</td>
</tr>
<tr>
<td>$\hat{C}_{yu}$</td>
<td>Cross-covariance estimate</td>
</tr>
<tr>
<td>$C_{LTV,BS}$</td>
<td>Vectorial form of the B-spline estimate of a two dimensional LTV impulse response function</td>
</tr>
<tr>
<td>$C_{ij}$</td>
<td>Vectorial form of the B-spline control points</td>
</tr>
<tr>
<td>$d_t / d_\tau$</td>
<td>Degrees of B-splines</td>
</tr>
<tr>
<td>$\hat{h}_{LTV}[t, \tau]$</td>
<td>B-spline smoothed two dimensional impulse response function of a linear time-varying system</td>
</tr>
<tr>
<td>$h_{NL\times1,RLS}$</td>
<td>Vectorial form of the regularized estimate of the two dimensional LTV impulse response function</td>
</tr>
<tr>
<td>$h_{LTV}$</td>
<td>Vectorial form of the two dimensional impulse response function of a linear time-varying system</td>
</tr>
<tr>
<td>$h_{L}$</td>
<td>Impulse response function with a length of $L$</td>
</tr>
<tr>
<td>$h_{NL\times1}$</td>
<td>Vectorial form of the two dimensional LTV impulse response function</td>
</tr>
<tr>
<td>$h_{\text{transient}}$</td>
<td>Impulse response function of a transient term (vectorial form)</td>
</tr>
<tr>
<td>$h$</td>
<td>LTI impulse response function (vectorial form)</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>$h[t, \tau]$</td>
<td>Two dimensional impulse response function of a linear time-varying system</td>
</tr>
<tr>
<td>$e$</td>
<td>Output observation noise with the following property $e \sim \mathcal{N}(0, \sigma^2)$ (vectorial form)</td>
</tr>
<tr>
<td>$f_s$</td>
<td>Sampling frequency</td>
</tr>
<tr>
<td>$K$</td>
<td>Observation matrix (used for LTI FIR estimation)</td>
</tr>
<tr>
<td>$K'_{BS}$</td>
<td>Observation matrix (used for 2D B-splines to eliminate transient)</td>
</tr>
<tr>
<td>$K_{LTV,BS}$</td>
<td>Observation matrix (used for 2D B-splines)</td>
</tr>
<tr>
<td>$K'_{LTV}$</td>
<td>Observation hypermatrix (used for 2D regularization)</td>
</tr>
<tr>
<td>$K''_{LTV}$</td>
<td>Extended observation hypermatrix (used for 2D regularization to eliminate transient)</td>
</tr>
<tr>
<td>$m$</td>
<td>Discrete frequency index</td>
</tr>
<tr>
<td>$\mathbb{N}^+$</td>
<td>Positive natural numbers</td>
</tr>
<tr>
<td>$\mathbb{N}^0$</td>
<td>Natural numbers including zero</td>
</tr>
<tr>
<td>$N$</td>
<td>Total number of data samples of an observation</td>
</tr>
<tr>
<td>$N_{transient}$</td>
<td>The length of the transient term</td>
</tr>
<tr>
<td>$n_t/n_\tau$</td>
<td>The number of B-spline control points used for an LTV system</td>
</tr>
<tr>
<td>$u$</td>
<td>Excitation signal (vectorial form)</td>
</tr>
<tr>
<td>$L$</td>
<td>The length of (the longest considered) impulse response</td>
</tr>
<tr>
<td>$P$</td>
<td>Covariance matrix (used for LTI regularization)</td>
</tr>
<tr>
<td>$P_{LTV}$</td>
<td>Covariance hypermatrix (used for 2D regularization)</td>
</tr>
<tr>
<td>$P'_{LTV}$</td>
<td>Extended covariance hypermatrix (used for 2D regularization to eliminate transient)</td>
</tr>
<tr>
<td>$p_h$</td>
<td>Hyperparameters (used for regularization)</td>
</tr>
<tr>
<td>$p_{BS}$</td>
<td>Hyperparameters (used for B-splines)</td>
</tr>
<tr>
<td>$\hat{R}_{yu}$</td>
<td>Cross-correlation estimate</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td>$V_{\text{LTV,BS}}$</td>
<td>B-spline cost function of an LTV system</td>
</tr>
<tr>
<td>$V_{\text{LTV,LS}}$</td>
<td>Linear LS cost function of an LTV system</td>
</tr>
<tr>
<td>$V_{\text{LTV}}$</td>
<td>Regularized cost function of an LTV system</td>
</tr>
<tr>
<td>$\hat{y}$</td>
<td>Smoothed modeled output</td>
</tr>
<tr>
<td>$y_m$</td>
<td>Measured output (vectorial form)</td>
</tr>
<tr>
<td>$\angle$</td>
<td>Phase</td>
</tr>
<tr>
<td>$\delta$</td>
<td>Kronecker delta functions (vectorial form)</td>
</tr>
<tr>
<td>$\omega$</td>
<td>Angular frequency</td>
</tr>
</tbody>
</table>
## Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIC</td>
<td>Akaike Information Criterion</td>
</tr>
<tr>
<td>AR</td>
<td>AutoRegressive model</td>
</tr>
<tr>
<td>BIBO</td>
<td>Bounded Input, Bounded Output (stability)</td>
</tr>
<tr>
<td>BLTI</td>
<td>Best Linear Time-Invariant (approximation)</td>
</tr>
<tr>
<td>CPSD</td>
<td>Cross Power Spectral Density</td>
</tr>
<tr>
<td>Cr</td>
<td>Crest factor</td>
</tr>
<tr>
<td>CV</td>
<td>Cross-Validation</td>
</tr>
<tr>
<td>DC</td>
<td>Diagonal Correlated kernel</td>
</tr>
<tr>
<td>DFT</td>
<td>Discrete Fourier Transform</td>
</tr>
<tr>
<td>DI</td>
<td>Diagonal kernel function</td>
</tr>
<tr>
<td>DoF</td>
<td>Degrees of Freedom</td>
</tr>
<tr>
<td>DSP</td>
<td>Digital Signal Processing</td>
</tr>
<tr>
<td>FDIDENT</td>
<td>Frequency Domain System Identification Toolbox</td>
</tr>
<tr>
<td>FIR</td>
<td>Finite Impulse Response</td>
</tr>
<tr>
<td>FIRF</td>
<td>Frozen Impulse Response Function</td>
</tr>
<tr>
<td>FRF</td>
<td>Frequency Response Function</td>
</tr>
<tr>
<td>FTF</td>
<td>Frozen Transfer Function</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>i.i.d.</td>
<td>Independently and identically distributed</td>
</tr>
<tr>
<td>ident</td>
<td>Identification toolbox</td>
</tr>
<tr>
<td>IRF</td>
<td>Impulse Response Function</td>
</tr>
<tr>
<td>ITF</td>
<td>Instantaneous Transfer Function</td>
</tr>
<tr>
<td>LOWESS</td>
<td>LOcally WEighted Scatterplot Smoothing</td>
</tr>
<tr>
<td>LPM</td>
<td>Local Polynomial Method</td>
</tr>
<tr>
<td>LS</td>
<td>Least Squares</td>
</tr>
<tr>
<td>LsTV</td>
<td>Linear slowly Time-Varying</td>
</tr>
<tr>
<td>LTI</td>
<td>Linear Time Invariant</td>
</tr>
<tr>
<td>LTV</td>
<td>Linear Time-Varying</td>
</tr>
<tr>
<td>MAP</td>
<td>Maximum A Posteriori (estimation)</td>
</tr>
<tr>
<td>ML</td>
<td>Maximum Likelihood (estimation)</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Square Error</td>
</tr>
<tr>
<td>pdf</td>
<td>Probability density function</td>
</tr>
<tr>
<td>pmf</td>
<td>Probability mass function</td>
</tr>
<tr>
<td>PRBS</td>
<td>PseudoRandom Binary Sequence (signal)</td>
</tr>
<tr>
<td>RBF</td>
<td>Radial Basis (kernel) Function</td>
</tr>
<tr>
<td>rms</td>
<td>Root mean square</td>
</tr>
<tr>
<td>rpms</td>
<td>Random phase multisine (signal)</td>
</tr>
<tr>
<td>rmse</td>
<td>Relative rms error</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>SS</td>
<td>Stable Spline (kernel function)</td>
</tr>
<tr>
<td>std</td>
<td>Standard deviation</td>
</tr>
<tr>
<td>SVD</td>
<td>Singular Value Decomposition</td>
</tr>
<tr>
<td>TC</td>
<td>Tuned-Correlated (kernel function)</td>
</tr>
<tr>
<td>var</td>
<td>Variance</td>
</tr>
</tbody>
</table>
Preliminaries
Chapter 1
Introduction

Engineers and scientists look for a reliable mathematical model of the observed phenomenon for understanding, design and control. System identification is a tool which allows them to build high quality models of dynamic systems starting from experimental noisy data.

In modeling and measurement techniques it is commonly assumed that the observed systems are linear time-invariant. This point of view is acceptable as long as the time variations and the nonlinearities of the systems are negligible. However, in some cases this assumption is not satisfied and it leads to a low accuracy of the estimates. In this thesis linear time-varying systems are considered.

The importance of time-varying systems can be seen through some application examples. To motivate the reader, various application fields can be mentioned:

- **Electrical engineering**
  
  A good example in electrical engineering is a (non-compensated) transistor in an operational amplifier with a shifting offset-voltage: the higher the temperature, the higher the offset drift. The offset voltage also changes as time passes (due to aging) [1], [2]. The offset variations influence the system parameters and result in a time-varying behavior. An example for a non-compensated operating amplifier can be for instance the Texas Instrument μA741C [3].

- **Aerospace engineering**

  In the case of an airplane, the time-varying behavior originates from the decreasing weight due to the fuel consumption, and from different surface configurations during take-off, cruise and landing [4]. Moreover, the resonance frequency and damping of most vibrating parts (for instance the wings) of a plane vary as a function of the flight speed and height [5], [6].
Chemistry

In chemistry, an interesting example can be the pitting corrosion in metals. The impedance changing in a metal is a function of the progress of the underlying chemical reaction. As the metal is being corroded, small holes are formed which grow with time, and passivate. This chemical reaction changes the value of the impedance [7], [8].

Biology

Aging and mortification in biological systems can also be a good example. The human adapts its stiffness of muscles to environmental conditions. The varying bio-impedance in the heart is also an interesting example from biomedical sciences [9]. The human vocal tract is a time-variant system, with its transfer function at any given time dependent on the shape of the vocal organs [10], [11]. The human hearing is also a well-known time-varying system, just think of the Fletcher–Munson curves [12].

Mechatronics and civil engineering

A robot arm which is a non-linear system can be seen as a time-varying system, when it is linearized around continuously evolving set points. The arm of a tower crane – with a heavy load – is also a time-varying system: the longer the cable, the lower the resonance frequency.

Acoustics and vibration engineering

Many acoustical, vibrational and noise processes have time-varying behavior. Using the techniques of acoustical radiation, the sound pressure or the sound intensity radiated from a vibration structure can indicate the status or the (upcoming) problems of the observed system [13], [14]. An interesting application example is for instance, the noise analysis of the internal combustion engines [15], [16].

Economics

After human aging the most well-known examples can be found in economics. All the economical processes are time-varying and they are still subject of many ongoing economical researches. Some interesting studies can be found in [17], [18], where they try to model the time-variations on the market with similar ideas presented in this thesis.

In those cases, advanced modelling is needed taking into account the time-varying behavior of the system.
1.1 Objectives

The time-varying systems are split into two classes. The first class consists of systems which are inherently time-varying. It means that the time-variations are the natural part of the observed phenomenon and in many cases they cannot be (significantly) controlled. A well-known example for this class of systems is aging.

In the second class, time variations depend on one or more special external variables (in most cases they are the scheduling variables). A good example can be for instance a tower crane, where the cable length can vary at any time resulting in a time varying behavior. The length of the cable is set by the operator of the machine.

This thesis mainly focuses on the first class of the systems, but under some conditions the provided methods can be applied for the second class as well.

A further distinction can be made between the cases, where the time variations follow a periodic behavior or there is no periodicity. The systems with no periodic time-varying behavior are the arbitrary time-varying systems. In this thesis the – general – arbitrary time-varying situation is studied.

The common problem in the above-mentioned application examples is that the system dynamics can change during the measurements. Think of the tower crane in real operating mode: the cable length (and even the weight of the load) can change several times during a measurement. The challenge is to build accurate models which can track the varying dynamics of these systems, while using as few experiments as possible.

In this thesis, nonparametric models are considered. It is already shown that the linear time-varying systems can be nonparametrically described in the time domain with a two dimensional impulse response function. However, due to the high number of parameters and the underdetermined system of linear equations, it is barely used in practice. Let us take a simple example: a measurement of a time-varying system contains \( N \) samples, which are (in time) equidistantly collected. But during the measurement – at these sample times – the system can have \( N \) different dynamics (in time domain they can be represented by impulse response functions). If we assume that the length of each instantaneous impulse response function is \( L \), then we have \( N L \) different parameters to be estimated. On the other hand, we have only \( N \) equations (measured samples). Using nonparametric modeling, these equations will have very high degrees of freedom. This means that we have infinitely many solutions, which are equally possible.

As a consequence, time-varying systems cannot be uniquely determined from a single set of input and output signals – unlike in the general case of linear time
invariant systems. Due to this fact, the number of possible solutions grows quadratically with the number of samples.

To decrease the degrees of freedom, some user-defined adjustable constraints will be imposed. These will be implemented by using two different approaches. First, a special two dimensional regularization technique is applied. The second implementation technique uses generalized two dimensional smoothing B-splines. In addition to the beneficial effects on the degrees of freedom, the effect of the disturbing noise can be decreased and a possible transient elimination technique will be shown.

Using the proposed methods, high quality models can be built.

### 1.2 Outline

This thesis consists of four parts and twelve interrelated chapters, and structured as follows:

**Preliminaries**

In the first part the basic concepts related to systems, signals as well as an intuitive introduction to system identification are given as follows:

- The aim of Chapter 2 is to give an overview about the basic principles of signals and systems. A definition of a linear time-varying system is given together with a brief overview about the different techniques of time-varying estimation techniques.

- In Chapter 3 a general overview about system identification is provided, where the most relevant nonparametric linear time-invariant impulse response identification techniques are explained. In addition to that, in this chapter many useful signal processing notions are defined such as the autocorrelation or the periodogram. These will be referred to later on.

- The proper choice of the excitation signal plays a dominant role in system identification. This topic is the experiment design. Some basic choices and important notions are provided in Chapter 4.
Nonparametric identification using regularization techniques

This part covers the basic concepts of linear time-invariant regularization techniques, the application possibilities and the problem formulation together with the proposed estimation technique. To support the reader, several illustrative examples are shown in this chapter.

- In Chapter 5 the key idea of the linear time-invariant regularization technique is discussed from the viewpoint of system identification.

- Chapter 6 deals with the non-uniqueness issues of the linear time-varying impulse response estimation. This non-uniqueness issue originates from the high degrees of freedom of systems of linear equations used to describe the impulse responses of the time-varying system. To decrease this freedom, a special two dimensional regularization method is provided. This chapter is based on the earlier works [19], [20].

- Chapter 7 provides an extension of the proposed regularization technique. First, a cross-validation technique is discussed, which can be used in real experimental conditions. Secondly, it gives a method to eliminate the undesired effect of the transient term. Last but not least, a possible reduction technique of the computational complexities and memory needs is provided.

Nonparametric identification using regularization B-splines

In this part the definitions of the one and two dimensional B-spline smoothing technique are given. Relying on the basic concepts of the previous part, the proposed time domain B-spline time-varying estimation method is presented. In addition to that, a surface smoothing and parameter reducing technique will be shown, which can be used for the “frozen” sliding window estimation technique. To support the theory, a measurement example is provided comparing the regularization and the B-spline techniques.

- The aim of Chapter 7 is to give an introduction to the basics of the one and two dimensional B-spline technique. This chapter is based on [21], [22].

- Chapter 8 shows the two dimensional time-domain linear time-varying impulse response estimation B-spline technique. Using B-splines a unique solution can be obtained by reducing the number of parameters to be estimated from the noisy observation. Further, a possible transient
elimination technique is shown here as well. Chapter 8 is based on the earlier works [22], [23].

- In Chapter 9 the state-of-the art nonparametric “frozen” frequency domain method is discussed. Based on this idea, a simplified frequency and time domain methodology is presented to estimate slowly time-varying systems. This chapter is based on [21].

- An experimental comparison of the B-spline and the regularization approaches is shown in Chapter 10. The experiment is based on the measurement of a linear time-varying system.

Conclusions

In this part the main contributions of my entire PhD research are provided. The scientific statements – theses – can be found also here. Last but not least, the list of publications is provided here.

Appendices

- The theoretical aspects of the proposed regularization technique are discussed in the Appendix A. Apart from some long derivations, the link between the regularization technique and the Bayesian statistical framework is presented. The statistical properties of the two dimensional regularized estimation are given here as well. Finally, a proof is given to show that the regularization technique can decrease the degrees of freedom to zero, which results in a unique solution of the estimate.

- In Appendix B some derivations and the statistical properties of the two dimensional frozen and non-frozen B-spline techniques are provided. In this appendix a proof is given to show that the non-frozen B-spline technique decreases the degrees of freedom of system of linear equations to zero. It also explains the differences between the P-splines and the regression B-splines.
Chapter 2
Signals and systems

Signals play a key role in understanding the observed phenomenon and in designing experiments. From the viewpoint of engineering, this observation is basically a measurement process consisting of two steps. In the first step information is collected by interchanging signals. In the second step the acquired information is analyzed and processed. The information related to the observed phenomenon is delivered by signals. This phenomenon can be described by the interactions of signals. When observing, we – directly or indirectly – interact with the observed object which is further referred to as the system.

In this chapter the class of observed systems and signals are defined. A detailed description about the signals, systems ([24], [25]) and processing techniques ([26], [27]) are beyond the scope of this thesis.

2.1 Signals

The notion of “signal” can be defined in many different ways such as in [28], where an engineering definition is given.

**Definition 2.1** A signal is a measurable quantity which provides information on the status of the observed phenomenon (system) or influences the properties of a system.

There are several possibilities to describe signals. In this work the main description is done by statistical and probability properties. Unless otherwise stated, signals (and systems) are described mainly in the time or alternatively in the frequency domain [29]. In this section a time domain based description is given.

Next, some important definitions and assumptions will be introduced.
2.1.1 Continuous and discrete time signals

**Definition 2.2** The signal $x$ is continuous when it is defined as a function of the independent – time – variable $t$ and for this $x(t)$ relationship a continuum of values of $t$ is assigned [30], [31].

In other words, $x(t)$ can be specified at any arbitrary instant of $t$. Analog signals are always continuous time signals.

**Definition 2.3** When the independent time variable $t$ can take only countable values, then we are talking about discrete time signals [31].

To distinguish this from the continuous case here a different notation is used: $x[t]$. Digital signals are discrete time signals.

In this thesis only discrete signals are considered. The questions regarding to the conversion between continuous and discrete time signals are out of the scope of this thesis.

2.1.2 Deterministic and stochastic signals

**Definition 2.4** A signal is deterministic when its values can be predicted for any given time or in other words, its values can be specified exactly [24].

**Definition 2.5** If a signal is not deterministic, then the signal is said to be stochastic [25].

**Definition 2.6** A deterministic signal is periodic, if its values repeat at equal shift of time, i.e. $x[t + T] = x[t]$, $\forall t$ and $T$ is the period of time. Quasi-periodic signals have different values over the equal time shift, but they contain some periodic components [24].

When a deterministic signal has no periodic component, in general – in this thesis – it is assumed that it tends to zero (for $|t| \to \infty$). This kind of – absolutely integrable [32] – signal is the transient signal.

If the above-mentioned assumption on the deterministic signals is not satisfied, then the signal is stochastic – or random. Consequently, it means that their values cannot be predicted exactly and therefore to describe them, statistical properties and probabilities must be used [24].

**Definition 2.7** When the statistical properties are invariant i.e., their values do not change over time or over different realizations, then these signals are stationary signals [24].

This assumption is typically made for the central moments [33]. Usually it is enough to consider the first two central moments (mean and variance). If the
assumption is only satisfied for these moments, then it is called weak stationarity [34]. This shall hold for every realization of the signal.

When the observation is not repeatable or only one realization is available, it is still possible to tell something about the stationarity properties. This notion is the ergodicity [35].

**Definition 2.8** If a signal has the same behavior averaged over time as averaged over different realizations and for these the stationarity holds, then the class of signals are ergodic.

A wider overview about some selected signals will be given in Chapter 4

### 2.2 Systems

In this context the observed object (phenomenon) is a system. When any kind of mathematical description is assigned to the system, then we are talking about a (mathematical) model. The most relevant systems, their models and properties are explained here.

#### 2.2.1 Linearity

A system is linear, when the principle of superposition holds: the system output to a linear combination of two – or more – signals is the same linear combination of what the outputs would have been when the signals would have been passed through individually [36].

**Definition 2.9** Assume that there is a function $G$ which – fully – describes the linear system. In this case $G$ is a model of the observed system. If the inputs – excitations – are denoted by $u_1$, $u_2$, the output is denoted by $y$, and $c_1, c_2$ are arbitrary constants, then superposition means:

$$
\begin{align*}
    y_1 &= G\{u_1\} \\
    y_2 &= G\{u_2\} \\
    y &= G\{c_1u_1 + c_2u_2\} = G\{c_1u_1\} + G\{c_2u_2\} = c_1y_1 + c_2y_2
\end{align*}
$$

(2.1)

Of course this can be extended to any number of inputs. If the system does not satisfy the assumptions above, the behavior of the observed system (modeled by $G$) is nonlinear.

#### 2.2.2 Time-invariant systems

In order to have a satisfactory model of a system, an exact input-output relationship must be defined.
**Definition 2.10** A linear system is said to be static, if the system output at any instant of time depends only on the input at the same time. If this assumption does not hold, then the linear system is dynamic [37].

**Definition 2.11** A linear dynamic (and static) system – denoted by $G$ – can be described by an impulse response function (IRF) in steady-state as follows:

$$y[t] = G[u] = \sum_{t=-\infty}^{\infty} h[\tau]u[t-\tau] = \sum_{t=-\infty}^{\infty} h[t-\tau]u[\tau]$$  \hspace{1cm} (2.2)

where $h[\tau]$ is the impulse response of the observed system at time instant $\tau$ [38].

The above equation is the convolution theorem for discrete time systems. When the signal $u[t]$ is a Kronecker delta functions $\delta[t]$ [24] (Dirac delta function), then the convolution gives the impulse response function to the output. When the experiment on a linear system is repeated at any time and it gives the same output – the IRF remains the same – then it is called a linear time-invariant (LTI) system. The most important properties of an LTI system are causality and stability.

**Definition 2.12** A discrete-time LTI system is causal, when the actual value of the output depends only on the actual and the past values of the input, or in other word the system’s output cannot react to the future excitation, i.e. $h[t] = 0$ when $t < 0$ [38].

To define the stability, the most used bounded input, bounded output (BIBO) criterion can be given [24].

**Definition 2.13** An LTI system is stable if, for every bounded input, the output is bounded finite.

### 2.2.3 Time-variant systems

The focus of this thesis is on the identification of systems based on the concept of Linear Time-varying (LTV) models. Unlike in the LTI case, here the impulse response function – hence the dynamics – of a linear time-varying system can be changed at any time.

Many authors refer to LTV systems as a special class of Linear Parameter-Varying (LPV) systems [39], [40], [41].

**Definition 2.14** A discrete, linear parameter-varying system can be defined by its two dimensional impulse response function (denoted by $h_{LPV}[t, p_s]$) as follows:

$$y[t] = \sum_{t=-\infty}^{+\infty} h_{LPV}[\tau, p_s]u[t-\tau] = \sum_{t=-\infty}^{+\infty} h_{LPV}[t-\tau, p_s]u[t]$$  \hspace{1cm} (2.3)

where the parameter $p_s$ is the scheduling variable.
It is obvious when the variable $p_s$ represents the time, then the LPV model will depend on the time as well, which leads us to the definition of the LTV systems [42].

**Definition 2.15** The $G_{LTV}$ discrete, linear time dependent system can be defined by its two dimensional impulse response function (denoted by $h_{LTV}[t, \tau]$) as follows:

$$G_{LTV}\{u[t]\} = y[t] = \sum_{\tau=-\infty}^{t} h_{LTV}[t, \tau] u[t - \tau]$$

where the parameter $t$ is the global time and $\tau$ is the system time [22].

It is also important to remark that some authors refer to $h_{LTV}[t, \tau]$ as the time-varying kernel. The main concept of this thesis is based on this impulse response notion.

This means that the response of the system to an impulse depends on the time on at which the excitation is applied. An illustration is shown in Figure 2.1. Observe that the responses to two impulses applied at different time instants are different. In this case, $h_{LTV}[t, \tau]$ can be seen as snapshots of the instantaneous dynamic behavior at these two time instants of the impulses. Next, the linearity and stability will be defined.

**Definition 2.16** $G_{LTV}$ is linear because the superposition holds, i.e. with $c_1$, $c_2$ arbitrary constant values:

$$y = G_{LTV}\{c_1 u_1[t] + c_2 u_2[t]\} = c_1 G_{LTV}\{u_1[t]\} + c_2 G_{LTV}\{u_2[t]\}$$

**Definition 2.17** $G_{LTV}$ discrete, linear time dependent system is causal when the following is true:

$$h_{LTV}[t, \tau] = 0, \text{ when } \tau < 0$$

Note that this definition differs from the LTI case where the constraint is on the – global – time variable $t$. In Chapter 6 this will be explained in details. Last but not least, the stability can be defined in the same way as for the LTI case [43].

**Definition 2.18** An LTV system is stable if, for every bounded input, the output is bounded and finite at any global time of $t$ [44].

![Figure 2.1: An example of an LTV system.](image-url)
Before we go any further, a brief overview about different approaches used for – different types of – LTV systems will be given here.

Although LTV systems can be described – in theory – uniquely in the time domain with a two dimensional impulse response function, the related description in the frequency domain is not so trivial. It sounds contradictory but the – classical – transfer function does not exist for LTV systems. A transfer function describes the amplitude and phase relation between input and output sinusoidal signals of an LTI system [24]. However in this case this definition does not hold because the response of an LTV system to a sinusoidal is not sinusoidal anymore. It is due to the fact that the poles and zeros of an LTV system are not fixed and they can move continuously [43].

Therefore a generalization of the transfer function is needed. One possibility is proposed in [42], [44] and applied to practical problems for instance in [45]. This is the system function, also known as instantaneous transfer function (ITF). This is well studied and discussed in [40], [41], [45], [46], [47]. It is possible to interpret this notion as a snapshot of the instantaneous dynamic behavior – it is analogous to $h_{LTV}[t, \tau]$ in the time domain.

**Definition 2.19** For a causal LTV system, the system function is defined at any time instant $t$ with the $\mathcal{L}$ transform (Laplace transform) as follows:

$$H_{LTV}(t, s) = \mathcal{L}\{h_{LTV}(t, \tau)\} = \int_{t=0}^{+\infty} h_{LTV}(t, \tau)e^{-st}d\tau \quad (2.7)$$

It can be proven that the system function has similar properties as the (LTI) transfer function [44], [48], [49], i.e. in steady-state (zero initial conditions):

i. the output of the system can be computed as $y(t) = \mathcal{L}^{-1}\{H_{LTV}(t, s)U(s)\}$
   
   where $U(s) = \mathcal{L}\{u(t)\}$ and $\mathcal{L}^{-1}$ is the inverse Laplace transform,

ii. the response to a sine wave excitation $u(t) = \sin(\omega t)$ is given by the modulated output as follows: $y(t) = |H_{LTV}(t, j\omega t)|\sin(\omega t + \angle H_{LTV}(t, \omega t))$.

The discrete system function can be obtained by using the $\mathcal{Z}$-transform as follows:

**Definition 2.20** For a causal discrete LTV system, the discrete system function is defined at time instant $t$ with the $\mathcal{Z}$ transform (discrete Laplace transform) as follows:

$$H_{LTV}[t, z] = \mathcal{Z}\{h_{LTV}[t, \tau]\} = \sum_{\tau=0}^{+\infty} h_{LTV}[t, \tau]z^{-\tau} \quad (2.8)$$

In this thesis, the proposed time domain methods will be compared to a special implementation of the above-mentioned concept [40].
There is another similar approach to define somehow a transfer function. When the time variations are slow, during a short observation time the system can be well approximated by a time invariant model. Then it is a common practice to describe the system as a series of LTI systems. In these techniques, at each measurement time a “frozen” LTI model is built. These LTI systems are called the frozen instantaneous systems. These models can describe the time-varying behavior quite well [39], [44] [50], [51], [52].

The drawback of these methods is that there can be a quite significant time variation during a single experiment, such that an LTI model is not sufficient to describe the system’s behavior.

**Definition 2.21** The transfer functions obtained by using “frozen” coefficients are the frozen transfer function (FTF) [41].

A longer description about the instantaneous and frozen approaches can be found in Chapter 10.

There is also a recently published similar approach in [53]. This method uses a – one dimensional – regularization technique (see Chapter 5). In the referred work a sliding window is used over different moments. The estimation is done by an extended kernel function (see Chapter 5). This is quite similar to the above-mentioned frozen LTI approaches.

It is also possible to use different recursion techniques to track the changes of the parameters such as in [54], [55], [56]. These techniques typically use a kind of sliding window with the assumption that the system is time invariant inside that window. The most common techniques use time-varying ARX, ARMAX [57] parametric models.

Related to the parametric representations, some authors expand the time-varying coefficient onto a finite set of basis sequences, wavelets [58], [59]. There are some interesting wavelets techniques which provide (directly) a good estimation of the impulse responses [60], [61].

There are some distinguished methods where they build a model from the complete measured time window and considered frequency band at once using difference or differential equations such as in [62], [63], [64], [65]. The basic idea of this nonparametric estimation (see Section 3.1.2) is that the parameters need to be estimated at once which is similar to the main concept of this thesis. Based on this concept two proposed methods (using regularization and B-splines) will be shown in this thesis.

There are some alternatives in control and automation, where they prefer to use a state space representation instead. These studies basically describe the effect of the
varying subspace parameters as a function of $t$. The structural properties of such an LTV system are analyzed using the solutions of the system of differential (difference) equations [65], [66], [67]. Some other authors try to identify state space models using wavelets [68], [69]. This kind of representation is out of the scope of the thesis. Last but not least, there is an important subclass of LTV systems: the periodically time-varying systems (LTP) which are studied for instance in [41], [70]. In this case, to estimate model parameters more advanced statistical methods can be used.

Related to the instantaneous and frozen as well as to the periodically varying systems approaches, there is a very important the concept that needs to be mentioned. It is the best linear time-invariant (BLTI) approximation of an LTP system [71]. It tells us what we would obtain, if we would simply use an LTI model instead an LTV model.

### 2.2.4 Smooth systems

In this thesis only smooth systems are considered. Therefore for the smoothness property a qualitative definition will be given [22]. An intuitive example can be found in Figure 2.2.

**Assumption 2.1** A discrete LTI system is smooth, if the absolute value of the finite difference of $h[t]$ is relatively small (compared to the peak-to-peak value of the impulse response function) [32].

**Assumption 2.2** A discrete stable LTV system is smooth, if the absolute value of the finite difference over adjacent points of $h_{LTV}[t, \tau]$ in both $t, \tau$ directions is relatively small (compared to the peak-to-peak value of the impulse response function).

![Figure 2.2: An example of smooth and non-smooth LTI impulse responses.](image-url)
2.3 The scope of the thesis

The main contributions in this thesis are related to the class of smooth linear time-variant systems, but some achievements with smooth linear time-invariant systems will be shown as well.

In this work, the considered signals are restricted to discrete deterministic or ergodic signals.

In this thesis the following novelties are presented based on the different concepts of time-varying systems:

- a proposed frequency domain B-spline based FTF estimation (“frozen windowing”) will be shown with an assumption on the smoothness of IRFs (see Section 2.2.4). This is discussed in Chapter 10,
- two different “non-frozen” time domain methods will be presented based on B-splines (Chapter 9) and a regularization technique (Chapter 6–7) with an assumption on the smoothness (and on the stability).
Chapter 3
An introduction to system identification

Engineers and scientists look for a reliable mathematical model of the observed system for understanding, design and control. Modeling can be very complicated and its accuracy will depend – among others – on the observed phenomenon, the environment of the experiment, the preliminary knowledge on the phenomenon and the processing of these data. To handle this complexity a general viewpoint is needed. This field of science is system identification. It allows the user to model dynamic systems from experimental noisy data. This is an interdisciplinary science which connects the world of control theory, data acquisition, signal processing, statistics, time series analysis and many other various areas.

In this Chapter a general overview about system identification is given together with the most relevant nonparametric techniques. A detailed description can be found in the classic reference textbooks such as [37], [57], [72] and in the available toolboxes such as [73], [74].

3.1 Introduction

System identification is a powerful technique for building high quality models of systems from noisy observations. Basically, the system identification process consists of four interrelated steps:

- experiment design and data acquisition,
- selection of the model type,
- estimation of the model parameters, and
- validation of the estimated model.
In the following sections, each of these steps will be explained.

3.1.1 Experiment design and data acquisition

The experiment design plays an important role in the identification procedure. It makes it possible to collect valuable information (data) about the system. This measurement allows the user to build a model.

If the experiment is not well designed, then it cannot be guaranteed that all the required information can be extracted from the measurement. Therefore it is important to pay enough attention to this step.

The user has to select an excitation signal that is as close as possible to the real experimental conditions, covering the full (frequency) band of interest.

If the excitation signal is carefully chosen – persistent – then the measurements based on this signal provide sufficient information on the observed system to identify it.

The experiment design has many aspects such as the selection of measurement devices, questions regarding to the environment and signal design. The latter will be detailed in Chapter 4.

3.1.2 Selection of the model type

When the data (information) are collected from the observations, a precise model type and its structure need to be chosen. This model is supposed to describe the observed system quite well. Although at first glance it seems to be easy, although there is no doubt that it is the most difficult step [38]. Here some important model choices follow.

3.1.2.1 White, black and gray box modeling

Sometimes it is possible to use prior information about the system and about its internal structure. In this case we are talking about white box modeling – or physical modeling. These white box models – typically – rely either on the laws of applied sciences (physics, chemistry, engineering, etc.) or on the known physical structure of the system. The main disadvantage of this approach is the lack of flexibility: the model building process needs to be done for every new problem and it can lead to complicated structures.

When no prior information is available or – it is not taken into account – then we are talking about black box modeling. In this case the model is strictly built from the observations – such as the input and output measurements. Black box models are – in general – more flexible than white box models and they can be used to identify
various kinds of systems. The main issue with black box models is that the number of necessary parameters can grow dramatically, resulting in a higher computational load and storage capacity.

There is an intermediate step between white and black box modeling. When some but not all preliminary knowledge is available or used, then we are talking about grey box modeling. The built model is based on both prior information and experimental data. Grey box modeling is also known as semi-physical modeling [75].

3.1.2.2 Linear and nonlinear models

Almost every system is nonlinear in real life. The difficulty with these systems is that there is no unique solution to describe them. It is due to the many different types of nonlinear systems with different behaviors. Consequently, it means that the modeling must be extensively involved and – unfortunately – universally usable design tools are not yet available.

For these reasons, nonlinear systems are often approximated by the models of linear systems, because this is often a reasonable approximation, and LTI theory is well understood. This model is usually closer to real-world phenomena, and it simplifies calculations. In most of the cases, it is reasonable to assume this because in many cases the linearities are dominating – and the nonlinearities are negligible [72].

3.1.2.3 Parametric and nonparametric models

When a system is described with a model which has a (very) limited number of terms, the model is called parametric model. For instance, a parametric model is used when a system is described by its poles and zeros.

In case of a nonparametric representation, the system is described by measurements of a system function with high number of samples – in theory with infinite number of samples [72]. Such kind of nonparametric model is, for instance, the impulse response function or the frequency domain equivalent, the frequency response function (FRF).

In the second part of this chapter a brief overview will be given to the estimation of IRF and FRF using classical system identification methods.

3.1.2.4 Estimation of the model parameters

Once the type of model is chosen, the actual values of the parameters have to be determined with respect to the collected (available) data. In order to assess the model quality, an objective criterion (function) is used which is a measure of the goodness
of the fit. For this performance check-up there are many well-known statistical methods accessible such as the maximum likelihood (ML) framework.

In the ML framework, the probability density function (pdf) of the observation noise [37] and the excitation signal is assumed to be known exactly [76]. In the particular case when the disturbing (observation) noise has white Gaussian distribution [30] with zero mean and a certain variance, the ML estimation method boils down to a least squares (LS) problem [77].

3.1.2.5 Validation of the estimation

Once the model parameters are estimated, the evaluated model must undergo a validation test. In this phase the model must be able to predict the behavior of the system well under new conditions. When the model cannot predict it, then there are some modeling errors left. Several techniques are available to perform this check-up [57], [78].

The most used method is the cross-validation technique. In this case the whole dataset is split into two subsets: estimation and validation sets. The estimation set is used to estimate the model and the validation set is used to verify whether the model predicts well the behavior of the system. When the modeling error is lower than a certain value, it is needed to step back to a previous stage.

3.2 An overview about estimation of nonparametric models

In this section only linear time-invariant estimation methods are taken into account. An overview about time-varying system identification will be given later on.

3.2.1 Transient analysis

Due to its simplicity, in industrial practice it is still one of the most widely known identification method ([79], [80]). In this case the excitation signal is strictly limited to the typical unit step function or to the unit impulse function (see Chapter 4). The output observation constitutes the model [38]. This is a simple continuous time model that describes the main time constants, the static gain, the delay and the system dynamics.

With this method typically – a parametric transfer function of – process models can be estimated for designing controllers [81]. An example of a first-order plus time delay process model is shown here [80] which has the following form:
Chapter 3- An overview about estimation of nonparametric models

\[ G(s) = \frac{K_p}{1+sT_1} e^{-sT_d} \, . \] Using Figure 3.1 the parameters can be read directly \((K_p \approx 1.5, T_d \approx 2, T_1 \approx T_1' - T_d = 2.2)\).

![Graph showing input and output signals with time [sec] on the x-axis and amplitude [V] on the y-axis.](image)

**Figure 3.1:** An example of transient analysis using step excitation

The drawback is that the obtained information is somewhat limited (see Section 4.2.2).

### 3.2.2 Correlation analysis

In this technique the cross-correlation function \([33]\) between the output and input signals is used to estimate the IRF with the following – causal and stable – model:

\[ y_m[t] = y[t] + e[t] = \sum_{\tau=0}^{\infty} h[\tau] u[t - \tau] + e[t] \tag{3.1} \]

where \(e[t]\) is the observation noise assumed to have a normal distribution with zero mean and variance \(\sigma^2\), i.e.: \(e \sim \mathcal{N}(0, \sigma^2)\).

In this particular case the input signal \(u[t]\) is a stationary, stochastic signal (see Chapter 2). When the excitation signal is a stationary Gaussian white noise (see Chapter 4), i.e., \(u \sim \mathcal{N}(0, \sigma_u^2)\), the estimation reduces to the following – simple – equation \([38]\):

\[ \hat{h}[t] = \frac{R_{yu}[t]}{\sigma_u^2} \tag{3.2} \]

where \(\hat{R}_{yu}\) is the estimate of the output-input cross-correlation function \(R_{yu}\).

**Definition 3.1:** The output-input cross-correlation function \(R_{yu}\) is defined as follows:

\[ R_{yu}[\tau] = \mathbb{E}\left\{ y(t)u^T(t - \tau) \right\} \tag{3.3} \]

If the measurement consists of \(N\) pair of samples (\(N\) for the input and \(N\) for the output) then the output-input cross-correlation function \(R_{yu}\) can be estimated at time lag \(t_{\text{lag}}\) as

\[ \hat{R}_{yu}[t_{\text{lag}}] = \sum_{\tau=0}^{N-|t_{\text{lag}}|} y[\tau]u[|t_{\text{lag}}| - \tau] = \sum_{\tau=0}^{N-|t_{\text{lag}}|} y[|t_{\text{lag}}| + \tau]u[\tau] \tag{3.4} \]
The biased cross-correlation function estimate is $\tilde{R}_{yu} \left[ t_{\text{lag}} \right]/N$, and the unbiased cross-correlation function estimate is $R_{yu} \left[ t_{\text{lag}} \right]/(N - |t_{\text{lag}}|)$. When $u[t] = y[t]$ then the cross-correlation (estimate) is called the autocorrelation (estimate).

**Definition 3.2** The output-input cross-covariance function $C_{yu}$ is defined as follows:

$$C_{yu}[\tau] = \mathbb{E} \{ (y(t) - \mathbb{E}[y])(u[t+\tau] - \mathbb{E}[u]) \}$$  \hspace{1cm} (3.5)$$

The output-input cross-covariance function $C_{yu}$ can be estimated at time lag $t_{\text{lag}}$ as

$$\hat{C}_{yu} \left[ t_{\text{lag}} \right] = \sum_{\tau=-N}^{N-1} \left(y[\tau] - \bar{y}\right) \left(u[\tau + t_{\text{lag}}] - \bar{u}\right)$$ \hspace{1cm} (3.6)$$

where $\mathbb{E}\{\bar{u}\} = \frac{1}{N}\sum_{t=0}^{N-1} u[t]$ and $\mathbb{E}\{\bar{y}\} = \frac{1}{N}\sum_{t=0}^{N-1} y[t]$ are the sample means of $u[t]$ and $y[t]$. The biased cross-covariance function estimate is $\hat{C}_{yu} \left[ t_{\text{lag}} \right]/N$, and the unbiased cross-covariance function estimate is $\tilde{C}_{yu} \left[ t_{\text{lag}} \right]/(N - |t_{\text{lag}}|)$. When $u[t] = y[t]$ then the cross-covariance (estimate) is called auto-covariance (estimate).

It also implies that the summation in Eq. (3.1) goes to $N - 1$ instead to infinity (see Eq. (3.10)). Note that many textbooks such as [37], [38] and [78] define the cross-correlation and cross-covariance in different ways using different concepts.

With prewhitening [38] the signals, this technique may work with many other types of excitations. The main drawback is that, it assumes that the input is uncorrelated with the disturbing noise measured at the output. It will not work properly when the data are collected from the system under output feedback.

The previously given technique can be used in the frequency domain as well. A special kind of correlation analysis is the well-known cross power spectral density (CPSD) estimation based on the Wiener–Khinchin theorem.

**Definition 3.3** CPSD estimate of the LTI frequency response function is the distribution of power per unit frequency and is defined as

$$\tilde{H}[m] = \frac{\mathcal{F}_D \{ \hat{R}_{yu}[\tau] \}}{\mathcal{F}_D \{ \hat{R}_{uu}[\tau] \}} = \frac{\hat{S}_{yu}[m]}{\hat{S}_{uu}[m]}$$  \hspace{1cm} (3.7)$$

where $\hat{S}_{yu}$ and $\hat{S}_{uu}$ are the periodograms at $m = \frac{j\omega m}{2\pi}$ frequency index (i.e. the estimates of the corresponding power spectra) and the discrete Fourier transform $\mathcal{F}_D$ is defined as follows:
DEFINITION 3.4 $\mathcal{F}_D\{\cdot\}$ is the discrete Fourier transform (DFT) of the time domain signal $u[t]$ defined as follows:

$$\mathcal{F}_D\{u[t]\} = U[m] = \frac{1}{\sqrt{N}} \sum_{t=0}^{N-1} u[t] e^{-2j\pi \frac{m}{N} t}$$  \hspace{1cm} (3.8)

where $m = 0,1,2 \ldots N - 1$ [31].

DEFINITION 3.5 $\mathcal{F}_D^{-1}\{\cdot\}$ is the inverse discrete Fourier transform (IDFT) of the frequency domain signal $U[m]$ defined as follows:

$$\mathcal{F}_D^{-1}\{U[m]\} = u[t] = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} U[m] e^{2j\pi \frac{m}{N} t}$$  \hspace{1cm} (3.9)

where $t = 0,1,2 \ldots N - 1$ [31].

Note, that in the definitions above instead of the scaling factor $1/\sqrt{N}$, an arbitrary constant can be used – depending on the needs of application – as long as the Parseval’s theorem is satisfied.

The drawback is that for this method advanced digital signal processing (DSP) knowledge is required. For example, we should count on spectral leakage when the observation window is not well chosen with respect to the excitation signal. There are different methods to increase the quality of the estimation such as Welch's method [82]. More details on the estimation of cross-correlations, periodograms can be found in [26].

3.2.3 Parametric estimation of the coefficients of the impulse response function

DEFINITION 3.6 Let us consider the following – causal, stable and finite – base line model, which one is called finite impulse response (FIR) as follows:

$$y_m[t] = y[t] + e[t] = \sum_{\tau=0}^{L-1} h[\tau] u[t - \tau] + e[t]$$  \hspace{1cm} (3.10)

For practical reasons it can be rewritten in vector form as:

$$y_m = y + e = Kh + e$$  \hspace{1cm} (3.11)

where the regressand variable is $y_m=[y_m[0] \ldots y_m[N - 1]]^T$, the observation noise is $e=[e[0] \ldots e[N - 1]]^T$, the ideal noiseless output is $y=[y[0] \ldots y[N - 1]]^T$, the IRF’s model is $h=[h[0] \ldots h[L - 1]]^T$. $K$ is the regressor (the observation) matrix defined as the lower triangular $N \times L$ Toeplitz matrix of the input vector $u$ ($u=[u[0] \ldots u[N - 1]]^T$) i.e.:
Chapter 3 - Scope of the thesis

The FIR model order is \((L + 1)\), and the length of the IRF estimate is \(L\) \((h[\tau] = 0, \text{ when } \tau > L)\). This model structure will be extended in the following chapters.

The most often used estimation methods are based on the minimization of the squared errors between measured and estimated outputs. As explained in the previous section, it is a common practice to use a ML estimation to obtain the parameters of the FIR. Because the input is assumed to be known and on the output there is an additive zero mean Gaussian noise, it leads to a LS problem with the following well-known cost function [57]:

\[
V_{LS} = \|y_m - Kh\|_2^2
\]  

(3.13)

and from here the simple ordinary LS solution is given by the generalized Moore-Penrose pseudoinverse (denoted as \(\dagger\)) as follows [83]:

\[
h_{LS} = \arg\min_h V_{LS} = K^\dagger y_m = (K^T K)^{-1} K^T y_m
\]  

(3.14)

Note, that this equation holds only, if some additional assumptions are satisfied. These will be introduced and discussed later on – for the special LTV cases.

3.3 Scope of the thesis

This thesis is limited to nonparametric models only. As it is explained previously, the contributions are given for systems and models that are linear and smooth.

Principally, a black-box time domain approach is used to describe the impulse response functions. Apart from different IRF techniques there is a small contribution to the frequency response function estimation as well.

In this thesis a special kind of maximum likelihood estimators will be introduced (based on B-splines and regularization technique) which will boil down either to least squares or to a maximum a posteriori (MAP) problem.

To support the right model selection, a special cross-validation technique will be presented later.
Chapter 4
Experiment design

In the case of a well-designed experiment – by measuring the output of a system – it is possible to maximize the amount of the extracted information from a dynamic system. This extraction is performed by system identification methods. In this particular situation – as in many cases in real-life –, we have the possibility to choose the excitation signals – with the limitation that the excitation signal should be in the permitted amplitude range. In this thesis some simulation and measurement examples are shown to demonstrate the efficiency of the proposed methods using noise and random phase multisine excitations. In this chapter some of the most common excitation signals – including the random phase multisine – are briefly discussed.

A detailed description about excitation signals and experiment design can be found – for instance – in [72], [84]. Excitation signals can be rapidly and easily generated by the recommended user friendly toolbox [73]. The questions regarding the instrumentation are beyond the scope of this thesis.

4.1 Introduction

Before introducing some typical excitation signals, in this section some important quality measures are defined which are important to classify the excitation signals.

The first important notion is the persistency of excitation (PE) [85]. This gives an indication of the “richness” of the signal. Our expectation from the excitation signal is that it should allow us to give access to as many parameters of the observed system as possible. The goodness of the estimation depends strongly (among others) on the order of the system and on the persistency order of the excitation signal.
**Definition 4.1** If the unbiased auto-correlation matrix of \( u \) is \( R_{uu} \) and the largest \( n_{PEd} \times n_{PEd} \) submatrix of \( R_{uu} \) is a positive semidefinite matrix, then the input signal is said to be persistently exciting of order \( (PEd) \) \( n_{PEd} \). If \( R_{uu} \) is non-singular, then \( u[t] \) is said to be a persistent excitation signal. If \( n_{PEd} \) is zero, then the \( u[t] \) is non-persistent excitation signal [37].

When PE order of \( u \) is \( n_{PEd} \), then it can be assumed that at least \( n_{PEd} \) parameters of a model can be estimated. It is important to remark that many textbooks such as [37], [57], [73] and [86] define the PE differently.

The next important notion is the Signal-to-noise ratio (SNR). This measure compares the power of the (useful) signal to the power of observation noise.

**Definition 4.2** The signal-to-noise ratio with a given noiseless observation signal \( y[t] \) and with a given observation noise is the following:

\[
\text{SNR} = \frac{E[y^2]}{\sigma^2}
\]  (4.1)

The larger the SNR is, the better the measurement quality (with respect to the noise power). An ideal measurement has infinitely large SNR. By increasing the power of the excitation signal, the SNR will be increased as well. It is very important to remark, that the power (and/or the amplitude) of the excitation signal is usually limited to a certain interval, and the observation noise cannot be decreased arbitrarily small. Therefore the SNR cannot be chosen arbitrarily large. An ideal measurement of a linear system has as large SNR as possible – with respect to the system constraints.

The last measure is the crest factor (Cr) which gives an idea about the compactness of a signal.

**Definition 4.3** The crest factor \( Cr(u) \) of the signal \( u[t] \) is given by the ratio of the largest peak value of \( u \) and \( rms \) (root mean square) value of \( u \):

\[
Cr(u) = \frac{u_{peak}}{rms(u)} = \frac{\max(|u|)}{rms(u)}
\]  (4.2)

where \( rms(u) = \sqrt{E[u^2]} \).

When the value of Cr is large, it means that the observed signal has an impulsive behavior, meaning that less energy is injected into the system. The (square root of the) total energy is represented by \( rms \).

Note, that there are many more quality measures available [57], [72], [78] which are beyond the scope of this thesis.
4.2 Classical excitation signals

In this section some of the classical excitation signals are briefly discussed. A graphical illustration is shown for every mentioned signal in Section 4.4.

4.2.1 Unit impulse

An impulse function is a special function that is often used by engineers to model certain events. Because an ideal impulse function is not directly realizable, the unit-impulse function can be used in practice. This is known as Dirac delta function. An example of this signal is shown in Figure 4.1.

A unit-impulse signal can be defined at any time shift $T$ as follows:

$$\delta[t - T] = \begin{cases} 1, & t - T = 0 \\ 0, & t - T \neq 0 \end{cases}$$  \hspace{1cm} (4.3)

The issue with this signal is that it has a low SNR, a low persistency degree and a very high crest factor. This means that the injected energy is very low and the signal is very poor (in terms of persistency), therefore the usage of this signal is quite limited.

4.2.2 Unit step signal

The unit step signal is one of the most popular signals in industry to estimate the parameters of a process model (see Section 3.2.1). An example of this signal is shown in Figure 4.1. The unit step signal can be defined as follows:

$$u_\circ[t - T] = \begin{cases} 1, & t - T \geq 0 \\ 0, & t - T < 0 \end{cases}$$  \hspace{1cm} (4.4)

The main drawback is that only limited information is available about high frequencies (since the frequency components are decreasing as $1/f$). The crest factor depends on $T$ (time shift). The advantage that is has a good SNR. The drawback is that it has low order of persistency.

4.2.3 Pseudorandom binary sequence

The pseudorandom binary sequence (PRBS) is a discrete persistent periodic signal switching its values between two levels (pseudo-)randomly. By choosing the period length to $N$ the spectral leakage can be avoided. The crest factor is 1, and the SNR is (typically) large. The most power can be found below $0.4f_c$ [84] ($f_c$ is the clock frequency [87]). Some implementation and optimization algorithms can be found in [78], [88]. An example of this signal is shown in Figure 4.1.
4.2.4 Stepped sine

This signal consists of increasing (or decreasing) frequency sine wave. The main advantages: there is no problem with the spectral leakage – due to its periodicity –; it is persistent; and it has a low Cr (typically around 1.45) and a high SNR. The drawback is that for a full-band excitation a very long signal is required – and therefore the measurement is longer. An example of this signal is shown in Figure 4.1.

4.2.5 Gaussian white noise

One of the widely used quasi-stationary persistent excitation signals is the white noise. When the samples are independent and identically distributed (i.i.d.) and each sample has a normal distribution with zero mean, then the signal is said to be Gaussian white noise. Its power spectral density is constant (in theory). An example of this signal is shown in Figure 4.1. This signal can be formulated as follows:

**Definition 4.4** The \( u_{\text{gwn}}[t] \) signal is said to be Gaussian white noise when

\[
\text{DEFINITION 4.4}
\]

\[ u_{\text{gwn}}[t] \sim \mathcal{N}(0, \sigma_u^2) \]  

and every pair of samples is uncorrelated.

Since the maximum absolute value of a finite length observation \( N \) of Gaussian white noise is a random variable and therefore the exact value of the Cr cannot be given. The observed Cr will slowly (quasi logarithmically) increase with length \( N \) \[89\]. The disadvantage is that we should expect leakage issues resulting in a non-perfect flat amplitude spectrum for a given realization. More details with examples can be found in \[84\].

4.3 Random phase multisine

Since the development of advanced digital signal processing algorithms and increased computational capability, it became possible to use complex input signals. To avoid unexpected issues – leakage, inconsistency, non-persistency – the random phase multisine (rpms) excitation can provide a handy, robust solution. It can be defined as follows:

**Definition 4.5** The random (uniformly distributed) phase multisine is a sum of harmonically related sinusoids as shown in the following equation:

\[
\text{DEFINITION 4.5}
\]

\[
u[t]_{\text{rms}} = \sum_{m=1}^{F} A_m \cos(\omega_0 mt + \phi_m), \quad \phi_m \sim \mathcal{U}(0,2\pi)
\]  

where \( \omega_0 \) is the fundamental angular frequency, \( A_m \) is the amplitude of \( m^{th} \) harmonic and \( F \) is the highest harmonic component.
These periodic persistent multisines are general purpose signals that can be applied without any optimization. The only parameters to be selected – by the user – are the magnitude, the bandwidth of the excitation signal and the frequency resolution of the measurement (by choosing \( N \)).

If the multisine contains all/only odd/even harmonics then it is called full/odd/even multisine, respectively.

The amplitude distribution of a random phase multisine is approximately normal or more precisely, it approaches a Gaussian distribution as the number of harmonics tends to infinity.

An example of this signal is shown in Figure 4.1. A typical crest factor is around 2–3 but to decrease it some Cr optimization techniques can be found in [72]. These signals are also very popular in nonlinear identification techniques and it can be used for living biological systems as well [90].

4.4 Example

In this section a simulation example is given with the excitation signals explained in this chapter. They are shown in the time- and frequency domain, respectively.

Simulation parameters: \( |u| \leq 1 \text{[V]} \), \( u_{\text{peak-to-peak}} = 1 \text{[V]} \), \( N = 128 \), \( f_s = 128 \text{[Hz]} \). The objective was to cover with these signals as wider frequency band as possible. In the figures the rms, Cr and PEd estimated values are shown.
Figure 4.1: Different excitation signals are compared in the time domain (on the left) and in the frequency domain (on the right).

Note the following important remarks:

- for the PRBS signal the length is $N = 127$,
- PE$_d$ are estimated with \textit{ident} toolbox [74],
- in Figure 4.1 the frequency domain signals are estimated by CPSD estimation.
4.5 Excitations used in this thesis

There are many available excitation signals. The optimum spectrum of the excitation strongly depends upon the goals and on the system under test.

The ideal excitation signal for nonparametric model estimation of LTI and LTV systems has the following characteristic:

- it must inject high power (the rms is high with a low Cr) resulting in a higher SNR,
- it must be persistent and the frequency band of interest must be excited equally well,
- the signal shall be periodic to avoid additional windowing used for spectral leakage.

Due to its beneficial aspect, in this thesis the simulation and measurements examples are shown by using either Gaussian white noise or a random phase multisine excitation.

Furthermore, it is important to remark that not every excitation signal is suitable for the estimation of LTV systems. This is discussed in Section 7.2.

Of course, the essence of this thesis is independent of the type of excitation signal.
NONPARAMETRIC IDENTIFICATION USING REGULARIZATION TECHNIQUES
Chapter 5
An introduction to regularization technique

When a least squares problem is formulated to determine the parameters of an impulse response of an LTI system, the problem can become easily ill-posed. The problem arises when the inverse of the observation matrix multiplied by its transpose \((K^TK)\) needs to be computed. When the matrix to be inverted is badly conditioned or the disturbing noise is high, then the LS estimate will result in a very large variance of the estimate. With the use of regularization this effect can be diminished.

The term regularization is not new– for instance, it is well-known in statistics as ridge regression or Tikhonov regularization [91], [92], [93]. However, in engineering it became popular in the last decade only, when some novelties from the machine learning framework were transferred to the field of system identification. Therefore regularization can be seen from two different viewpoints, when we want to identify the impulse response function of a linear, stable and causal system. The first viewpoint is the Bayesian framework, where Gaussian processes are used [94], [95]. The second viewpoint is when the regularization is seen as an extension of an ordinary least squares problem – this is the identification viewpoint. Both viewpoints tackle the same problem but from different aspects.

In this thesis, regularization is considered as an extended LS cost function trading-off between the well-known LS term and a regularization term based on prior information assumed about the underlying system.
### 5.1 Motivation

Before showing the novelties of my work, the basic concepts of the LTI regularization technique will be explained in this chapter. When the observation noise is white Gaussian distributed with zero mean, then the FIR model (see Section 3.2.3) can be estimated by the LS equation (3.14).

This estimation has very nice asymptotical properties such as consistency, unbiasedness, normality (the estimate is normally distributed) and efficiency (it achieves the Cramér–Rao lower bound [30]).

The main issue regarding LS estimation is that it is very sensitive to the conditioning of $K^TK$ – when $K^TK$ is singular, then the system is not identifiable – and to the noise.

A simple example is given here to demonstrate the noise sensitivity of the LS estimation. Let us estimate a – stable and causal – second order LTI system using an appropriate FIR model ($L = 30$). The input excitation is white noise ($N = 100$), and the SNR is 20 dB. The result of the simulation is shown in Figure 5.1. Observe that the estimation around the peak value is good, but around the tail part it is very bad. From the statistical viewpoint, the estimation is unbiased with a large variance. Observing this figure a question arises: “Is it possible to estimate it better?” The answer is yes. This will be explained further in this section.

![Figure 5.1: An example of a noisy LS LTI estimation.](image)

### 5.2 The regularized least squares cost function

It might sound contradictory but the key idea of the regularization technique lies in the fact that we need to inject some bias into the – asymptotically – unbiased LS estimate by modifying the cost function.

When the injected bias term is constructed based on the preliminary knowledge – prior information – then it is possible, by losing the unbiasedness, to gain a
significantly lower variance error, which results in a smaller error in total (see Figure 5.2). This total error can be expressed by the following notion.

**Definition 5.1** The mean square error (MSE) of the estimation \( \hat{h} \) of \( h \) is the second moment (details of this computation can be found in Appendix A.1):

\[
MSE(\hat{h}) = E \{(h - \hat{h})^2\} = bias(\hat{h})^2 + variance(\hat{h})
\]  

(5.1)

In the case of stable, causal impulse response function estimation, it is possible to impose two assumptions on the estimate based on our prior information about the system to be estimated. The first assumption is related to the fact that the considered systems are smooth (see **Assumption 2.1**). The second one is that the impulse response function of a stable system (see **Definition 2.13**) is – in general – exponentially decaying and it is possible to use this prior information as well.

In the proposed methodology this bias injection is implemented by introducing an additional term – containing the a priori information – in the least squares minimization criterion (Eq. (3.13)) that penalizes the model flexibility. This leads us to an extended cost function (\( V \)).

**Definition 5.2** The regularized LS cost function (\( V \)) consists of the ordinary LS cost function (\( V_{LS} \), see Eq. (3.13)) and the regularization cost function (\( V_r \)), i.e.:

\[
V = V_{LS} + V_r = V_{LS} + \sigma^2 h^T P^{-1} h
\]  

(5.2)

where \( P \) is a – special covariance – matrix containing the a priori information.

Minimizing Eq. (5.2) (as it is shown for the LTV case in Appendix A.2) with respect to \( h \) the estimated IRF is given by:

\[
\hat{h}_{reg} = \arg\min_h V = (K^T K + \sigma^2 P^{-1})^{-1} K^T y_m
\]  

(5.3)

where \( \sigma^2 \) is the noise variance of the observation.

When \( P \) is close to singular (badly conditioned) then the solution given by Eq. (5.3) is not well-defined. In this case \( P \) must be rewritten with the help of the singular value decomposition (SVD) as follows:

\[
P = SVD(P) = [U_{ns} \ U_s][P_{ns} \ 0] [U_{ns} \ U_s]^T
\]  

(5.4)

where \( P_{ns} \) is positive definite diagonal matrix and \([U_{ns} \ U_s]\) is a unitary matrix [96].

By taking the non-zero singular values \( P_{ns} \) and the corresponding unitary matrix \( U_{ns} \), then \( P \) has to be replaced in Eq. (5.2)–(5.3) by \( U_{ns} P_{ns} U_{ns} \) such that \( U_s \hat{h} = 0 \).
In the following sections, the theoretical and implementational aspects of the covariance matrix will be discussed.

It is important to remark that when the noise variance ($\sigma^2$) is zero, the regularization cost function boils down to the least squares cost function.

![Figure 5.2: Illustration of the concept of the regularization technique. The goal is to find the minimum MSE. The bias error represents the quantity of unmodeled data. The variance error represents the quantity of the modeled noise.](image)

### 5.3 The covariance matrix

The construction of the matrix $P$ plays a key role in the cost function. In order to be able to determine its values the Bayesian viewpoint has to be used, where the following assumption is imposed:

**Assumption 5.1** It is assumed that the impulse response function is a Gaussian process with zero mean and covariance $P$, i.e.

$$h \sim GP(0, P)$$  \hspace{1cm} (5.5)

This allows us to define the covariance matrix in the following way:

**Definition 5.3** The $L \times L$ covariance matrix $P$ is defined as:

$$P = \mathbb{E}\{h_L h_L^T\} = \begin{bmatrix}
P_{0,0} & \cdots & P_{0,L-1} \\
\vdots & \ddots & \vdots \\
\cdots & \cdots & P_{L-1,L-1}
\end{bmatrix}$$  \hspace{1cm} (5.6)

**Definition 5.4** Every element at $\{t_1, t_2\}$ indices in $P$ is determined as follows:

$$P_{(t_1,t_2)} = \mathbb{E}\{h[t_1]h[t_2]\}$$  \hspace{1cm} (5.7)
This is the covariance [30], which should be computed for every possible pair of time instants. The term “covariance” used here might be a little confusing. The notions above can be seen as the correlation (matrix). In the case of regularization technique the covariance and the correlation are equivalent, because in this work it is assumed that the impulse response function has a zero mean.

The Bayesian framework is discussed in details in Appendix A.3.

Assumption 5.2 A suitable choice of the covariance matrix $P$ should reflect what is reasonable to assume on the system. If the system is exponentially stable, the IRF should decay exponentially, and if the IRF is smooth, adjacent elements in $P$ should have a positive correlation.

These will be demonstrated with an example. The true system shown in Figure 5.1 is now computed by Eq. (5.3) using different covariance matrices.

The first, second and the third rows in Figure 5.3 show a situation when the diagonals are more “excited”. In other words, the correlation between adjacent elements is higher. This results in a smoother estimation.

The fourth and fifth rows show a situation when the diagonal elements in the identity matrix are exponentially decreasing. This decreasing means a faster decay rate resulting in a shorter impulse response estimate.

The last row shows a situation when the spreading around the diagonal and decaying are optimally combined resulting in the smallest MSE.

In the next section, a number of computational questions of the covariance matrix are addressed.
Figure 5.3: An example of different covariance matrices and regularized estimates. On the left different covariance matrices are shown. On the right, in black the true system and in gray its estimates are shown. The output is observed with an SNR of 20 dB. In figures (a)–(c) the smoothing is increased. In figures (d)–(e) the exponential decaying is increased. In figure (f) the optimal smoothing and decaying is shown.
5.4 Kernels

The numerical values of elements in $P$ (see Eq. (5.6)) are defined by kernel functions which describe the relation between the structure of $P$ and the type of prior information.

**Definition 5.5** A kernel function $P_K(t_1, t_2)$ is a symmetric, positive-definite function.

It can have a major effect on the quality of the estimate and it includes the information available on the unknown impulse responses.

Here, the most relevant kernels are shortly discussed. It is important to remark that – in principle – any kind of kernel can be used as long as the kernel function is built based on the prior information.

The essence of my thesis is independent of the chosen kernel(s).

5.4.1 Radial Basis Functions

One of the simplest and widely used kernel is the radial basis function (RBF) [97]. It is a function whose values depend only on the distance between different points. This can be defined as follows:

**Definition 5.6** The radial basis function $P_{RBF}$ is a function, which can be defined as:

$$P_{RBF}(t_1, t_2) = c_{RBF} e^{-\frac{(t_1 - t_2)^2}{\gamma}}$$

(5.8)

where $\gamma > 0$ is a parameter representing the length scale, $t_1, t_2$ are the two (time) points, $c_{RBF}$ is a constant such that $c_{RBF} \geq 0$.

This type of RBF kernel is called Gaussian RBF kernel [97], also known as squared-exponential kernel. By using this kernel, the smoothness assumption can be imposed on the system to be estimated. The larger the length scale $\gamma$, the smoother the resulting estimated function is.

This kernel is illustrated in Figure 5.3 (a)–(c).

5.4.2 Diagonal kernel function

In the case of stable FIR estimation, it can be assumed that the IRF is exponential decaying. To impose this – for instance – diagonal (DI) kernel functions can be used.
**Definition 5.7** The diagonal kernel function $P_{DI}$ is a function, which can be defined as:

$$P_{DI}(t_1, t_2) = \begin{cases} c_{DI}e^{-\beta t_1}, & t_1 = t_2 \\ 0, & t_1 \neq t_2 \end{cases}$$

(5.9)

where $\beta$ scales the exponential decaying, $c_{DI}$ is a constant and $c_{DI}, \beta \geq 0$.

In other words, $\beta$ controls the length of the IRF. The larger the value of $\beta$, the shorter the impulse response estimate.

This kernel is illustrated in Figure 5.3 (d)–(e).

### 5.4.3 Diagonal Correlated kernel function

In the case of stable FIR estimation, apart from the exponential decaying of the impulse responses, smoothness can be imposed.

To satisfy these assumptions, the diagonal correlated (DC) kernel function gives a perfect solution [98] as a special combination of RBF and DI kernels. It has the flexibility to tune independently the smoothness and the exponential decay.

**Definition 5.8** The diagonal correlated kernel function $P_{DC}$ is a function which can be defined as:

$$P_{DC}(t_1, t_2) = c_{DC}e^{-\alpha|t_1-t_2|}e^{-\frac{\beta(t_1+t_2)}{2}}$$

(5.10)

where $\alpha$ gives a scaling to the correlation between neighboring impulse response coefficients, $\beta$ scales the exponential decaying, $c_{DC}$ is a constant and $c_{DC}, \alpha, \beta \geq 0$.

In this case, $\alpha$ controls the smoothness: the smaller its value, the smoother the estimation is. And as in the previous case (DI), $\beta$ controls the length of the IRF.

This kernel function is illustrated in the Figure 5.3 (f).

Note that in many cases the formulation of DC differs from Eq. (5.10), like in [98] [96], where instead of the exponential function, powering is used – like in Section 5.4.5. It can be shown that they are equivalent – from the behavior point of view.

### 5.4.4 Tuned-Correlated kernel function

While DC provides a very nice flexibility to tune the model behavior, unfortunately it can lead to a very high computational complexity.

In some cases Tuned-Correlated (TC) kernels – as a special DC kernel function known as first order stable spline kernel [98] as well – can provide a good balance between the flexibility and the computational load.
When in the DC structure \( \alpha \) is set to \( \beta/2 \) then it leads to the TC form [99]:

**Definition 5.9** The Tuned-Correlated kernel function \( P_{TC} \) is a function, which can be defined as:

\[
P_{TC}(t_1, t_2) = c_{TC} \min(e^{-\beta t_1}, e^{-\beta t_2})
\]

where \( \beta \) is a scaling parameter and \( c_{TC}, \beta, t_1, t_2 \geq 0 \).

Note, that the TC kernels are widely used in different regularization toolboxes (e.g. *ident* in Matlab [74]).

### 5.4.5 Stable Spline kernel function

In some cases, the stable spline (SS) kernel can lead to a slightly better estimation fit than the TC kernels (see in [99]).

**Definition 5.10** The stable spline kernel function \( P_{SS} \) is a function, which can be defined as:

\[
P_{SS}(t_1, t_2) = \begin{cases} 
    c_{SS} \lambda t_1 - \lambda t_2 \left( \frac{t_1}{3} - \frac{t_2}{3} \right), & t_1 \geq t_2 \\
    c_{SS} \lambda t_2 - \lambda t_1 \left( \frac{t_1}{3} - \frac{t_2}{3} \right), & t_1 < t_2 
\end{cases}
\]

where \( \lambda \) is a scaling parameter and \( c_{SS} \geq 0, 0 < \lambda \leq 1 \).

### 5.5 Hyperparameters

The \( \gamma, \alpha, \beta, \lambda, c_{RBF}, c_{DI}, c_{DC}, c_{TC}, c_{SS} \) parameters (further jointly denoted as \( p_h \)) in Eq. (5.8)–(5.12) are the so called hyperparameters.

The values of the hyperparameters can be tuned in many different ways but there are basically two perspectives. One is the Bayesian framework, where the hyperparameters are estimated by the marginal likelihood function [100]. This is shown (for LTV case) in Appendix A.3.3.

The second perspective is the cross-validation (CV) technique (see Section 3.1.2.5), which is certainly the most widely used method for model selection. In this particular case there are four – interrelated – steps:

- split the data record into two parts: an estimation and a validation dataset
- estimate models using the estimation dataset for different values of \( p_h \)
- calculate the error between the measured and the model outputs using the validation dataset
- choose the values in $p_n$ which minimize the error

The theoretical aspects of further possible optimization techniques are beyond the scope of this thesis. Some other techniques can be found in [101].

Note, that it is also possible to treat the noise variance $\sigma^2$ as a hyperparameter and estimate its value together with the other hyperparameters. However, it will increase the number of hyper-parameters which is not always desirable in a nonlinear non-convex optimization problem.

## 5.6 Summary

Many experimental studies show that in the case of IRF estimation the TC, SS kernels give quite close results, while in some cases, the DC performs a little better [102].

The reason lies in the fact that the DC kernel uses two parameters to describe the decaying and the smoothness. This additional flexibility enables the DC kernel to capture more complicate impulse responses, but at the price of higher computational complexity.

It is important to remark that, depending upon the prior information, other kernels can be used as well. For instance, the cubic spline kernel, which is suitable for fast systems, but it has difficulties with handling slow systems. Another example is, when there is a periodicity in the system to be estimated. In this case it is possible to use periodic kernels.

A more detailed comparison can be found in [96], [99], [103].
Chapter 6
Time-varying system identification using two dimensional regularization

In this chapter a nonparametric time domain estimation method for linear time-varying systems is presented. The challenge with time-varying systems is that the time-varying two dimensional impulse response functions are not uniquely determined from a single set of input and output signals – like in the case of linear time invariant systems. Due to this non-uniqueness, the number of possible solutions grows quadratically with the number of samples.

To decrease the degrees of freedom, user-defined adjustable constraints will be imposed. This is implemented by a special two dimensional regularization. This regularization is applied once over the system time (the direction of the impulse responses) and once over the global time (representing the system memory).

6.1 Problem formulation

Unlike the linear time invariant systems, where the impulse response function is unique, the time-varying impulse response is not restricted to only one possibility for given input and output signals.

Assume that the length of the input $u[t]$ and the output $y[t]$ is $N$, then the possible variations for the time-varying IRF are $N^2$. Hence the impulse response function that relates the input $u[t]$ to the output $y[t]$ is not unique, because there are only $N$ linear relations (measurement samples) for $N^2$ unknowns. A detailed analysis on the degrees of freedom can be found in Appendix A.5.
This issue is illustrated in Figure 6.1.

These $N^2$ unknowns can be seen from a probability viewpoint as well: during $N$ samples the system can change $N$ times and each IRF can have a length of $N$.

The aim of this work is to decrease these degrees of freedom by using regularization technique, in order to obtain a smooth two dimensional impulse response function $h_{LTV}[t, \tau]$.

6.2 The regularized linear time-varying model

6.2.1 The baseline model

The key idea is that the classical linear time-invariant impulse response baseline model (see Section 3.2.3) can be extended. Next to the “ordinary” time variable $t$, an extra time variable is added into the argument list of IRF’s model leading to a two dimensional form.

In this thesis the following assumptions are imposed:

**Assumption 6.1** Consider the model defined in Eq. (2.4). For practical reasons (see Section 7.3) assume that the longest considered IRF has a finite length $L$ (this is the model order in this sense).

**Assumption 6.2** Assume that the finite length of the observation window is $N$ and the IRF is shorter than the observed data length, i.e. $L < N$.

**Assumption 6.3** It is assumed that the samples of the additive observation (measurement) noise $e[t]$ follow Gaussian, independent and identically distributed distribution with a zero mean and with a variance $\sigma^2$.

**Assumption 6.4** The considered two dimensional impulse responses are causal (see Definition 2.17) and stable (see Definition 2.18).

**Assumption 6.5** In this work it is assumed that the excitation signal $u[t]$ is known exactly.
**ASSUMPTION 6.6** The excitation signal \( u[t] \) and the noise signal \( e[t] \) are independent.

**ASSUMPTION 6.7** In this chapter it is assumed that the past values of the excitation are zero i.e. \( u[t] = 0 \) for \( t < 0 \).

The smoothness property is already defined in **ASSUMPTION 2.2**.

In this case all above-mentioned assumptions lead to the following simplified baseline model:

\[
y_m[t] = y[t] + e[t] = \sum_{\tau=0}^{\infty} h[t, \tau] u[t - \tau] + e[t] = \sum_{\tau=0}^{L-1} h[t, \tau] u[t - \tau] + e[t] \tag{6.1}
\]

This can be written in vector form as:

\[
y_m = K_{LTV} h_{NL \times 1} + e \tag{6.2}
\]

This measurement setup is illustrated in Figure 6.2, too.

![Figure 6.2: An illustration of the measurement setup.](image)

At this point it is very important to understand the logical structure of this model, therefore the structure of \( h_{NL \times 1} \) (see below) and \( K_{LTV} \) (see Eq. (6.5)) will be explained in details. These will play a key role in understanding the proposed method.

It is helpful to imagine \( h_{NL \times 1} \) as vectorization of a two dimensional matrix \( H_{L \times N} \):

\[
H_{L \times N} = \begin{bmatrix}
        h[0,0] & h[1,0] & \cdots & h[N-1,0] \\
        - & h[1,1] & \ddots & \vdots \\
        \vdots & \ddots & \ddots & \vdots \\
        - & \cdots & \cdots & h[N-1, L-1]
\end{bmatrix}
\tag{6.3}
\]

Since we consider the measurement window \( = 0, \ldots, N - 1 \), it can easily be verified that \( h[t, \tau] \) for \( t - \tau < 0 \) does not appear in Eq. (6.1) (because \( u[t - \tau] = 0 \), if \( t - \tau < 0 \)).
This is indicated in $H_{L \times N}$ as well (see Eq. (6.3)). In the matrix the hyphens stands for these – unknown – cases. This can be seen in the corresponding figures (Figure 6.3–Figure 6.5) as well, where no values have been plotted for these cases before the main diagonals.

Theoretically, these values exist but we are unable to determine them with the method presented in this chapter – due to the insufficient information about the past values of the input. When $u[t] \neq 0$ for $t < 0$, then we have additional transients. A proposed transient elimination method is discussed in Section 7.2.

The first column of the matrix $H_{L \times N}$ contains the different IRF values at time instant $t = 0$. As can be seen, in this case the length is only one sample. The first identifiable impulse response, i.e. the response to an impulse applied at $t = 0$ can be found at the diagonal $H_{L \times N}$ ($h[0,0], h[1,1], h[2,2],...,h[L−1,L−1]$).

The second column contains the responses at the time instant $t = 1$. There are two elements here: $h[1,1]$ is originated from the previous excitation time (first IRF) and the other one $h[1,0]$ is the first element of the new – second – IRF. The last column contains the values of IRFs at the excitation time $t = N−1$.

An illustration is shown in Figure 6.3. The IRFs can be found in the (co-)diagonals of $H_{L \times N}$. The values on the $t$ time axis give information about the impulse responses starting at these ($\tau_0 = \tau$) points.

![Figure 6.3: An example of a two dimensional impulse response function.](image)
From $H_{L \times N}$, the parameter vector of the LTV impulse response function is:

$$h_{NL \times 1} = \text{vec}(H_{L \times N}) = [h[0,0] \ldots h[0,L-1] \ldots h[N-1,L-1]]^T$$

(6.4)

$K_{LTV}$ in Eq. (6.2) is the $N \times NL$ observation matrix defined as follows:

$$K_{LTV} = 
\begin{bmatrix}
L & L & \cdots & L \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 \\
\end{bmatrix}$$

(6.5)

6.2.1 The least squares linear time-varying cost function

Assuming that the observation noise is not too large and the system is persistently excited, it seems to be handy to use a simple Maximum Likelihood (ML) estimation to obtain the parameters $h_{NL \times 1}$ in Eq. (6.2). Because the additive noise is zero mean Gaussian, it leads to a LTV LS problem with the following cost function:

$$V_{LTV,LS} = ||y_m - K_{LTV}h_{NL \times 1}\|^2_2$$

(6.6)

with the simple –well-known – ordinary LS solution for LTV model given by ($\dagger$ denotes the pseudo-inverse):

$$\hat{h}_{NL \times 1,LS} = K_{LTV}^\dagger y_m = \left(K_{LTV}^\dagger K_{LTV}\right)^{-1} K_{LTV}^\dagger y_m$$

(6.7)

By checking the dimensionalities of the involved vectors and matrices, it seems that Eq. (6.7) gives a compatible solution for $\hat{h}_{NL \times 1,LS}$. However, due to the rank issues, i.e. the inverse of $(K_{LTV}^\dagger K_{LTV})$ does not exist (the rank is $N$ but the dimensionality is $NL \times NL$) which implies that this equation is not correct.

In order to demonstrate the difficulties of the LS estimation (together with the rank issues), a simple simulation example is shown in Figure 6.4. The simulation is based on the time varying system shown in Figure 6.3. This is excited by white noise. To simplify the simulation, no observation noise has been added.
Figure 6.4 shows the LS solution of $h_{NL\times1}$ (converted to the form in Eq. (6.3)). As explained earlier the IRF solutions are not unique because there are more parameters than independent variables in the observation matrix. This can be found in Eq. (6.7) as well: the inverse of $(K_{LTV}^T K_{LTV})$ does not exist. Therefore, the solution given in Figure 6.4 is just one of the many solutions that minimize Eq. (6.6). With respect to the true system, the solution shown in Figure 6.4 does not correspond to our expectation although it is compatible with the dataset.

A unique solution can be achieved, if some prior information is taken into account as well. This will be explained in the following sections.

6.2.2 The regularized least squares time-varying cost function

Due to the large number of possible solutions given by Eq. (6.7), additional constraints are needed to decrease this freedom. In this particular case it is achieved by a two dimensional extended regularization.

In the case of a two dimensional IRF, there are two types of prior information available. The first one is that the considered systems are smooth. This smoothing is applied once over the system time $\tau$ which refers to the “classical” impulse response time and once over the global time $t$ which gives a support to handle the time varying behavior.

In addition to smoothing properties, another property of the stable impulse response function can be used: the IRFs are exponentially decaying.
In order to include the above-mentioned prior information in our nonparametric representation, an extended LTV cost function ($V_{LTV}$) must be defined.

**Definition 6.1** The extended regularized LS LTV cost function ($V_{LTV}$) consists of the ordinary least squares cost function ($V_{LTV,LS}$, see Eq. (6.6)) and the regularization cost function ($V_{LTV,R}$), i.e.:

$$V_{LTV} = V_{LTV,LS} + V_{LTV,R} = V_{LTV,LS} + \sigma^2 h_{NLx1}^T P_{LTV}^{-1} h_{NLx1}$$  \hspace{1cm} (6.8)

where $P_{LTV}^{-1}$ is a (two dimensional) covariance hypermatrix (or block matrix) containing the prior information. The minimization of Eq. (6.8) with respect to $h_{NLx1}$ provides the solution which estimates the two dimensional IRF of an LTV system (as it is shown in Appendix A.2):

$$h_{NLx1,RLS} = \left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} K_{LTV}^T y_m$$  \hspace{1cm} (6.9)

The detailed statistical properties of this estimator can be found in Appendix A.3.1 –A.3.2.

To show the power of the regularized estimation, the simulation made for the simple ML estimation shown in Figure 6.3 is repeated now with the regularization technique, but at this time 10% noise (compared to the output rms) is added to the true output. In this case prior information is used –smoothness and decaying. The result is shown in Figure 6.5. The method (that will be called flexible approach) used for this simulation is explained in Section 6.4.2.
In the LTI case, when the measurement has a very good quality – or the measurement is sufficiently long – the variance will be smaller and smaller such that the regularization term can be neglected.

However, in the case of LTV systems, the regularization term must always be active. It is because the constraints for the system of linear equations – describing the LTV model – are defined here and they are fundamental in the estimation procedure.

In this particular case – as it is shown in the previously defined kernel functions – $P_{\text{LTV}}$ consists of a matrix (denoted here as $\hat{P}_{\text{LTV}}$) and a scalar multiplier (denoted here as $\hat{c}$). Using this information, the cost function given by Eq. (6.8) can be rewritten as:

$$V_{\text{LTV}} = V_{\text{LTV,LS}} + V_{\text{LTV,R}} = V_{\text{LTV,LS}} + \sigma^2 h_{NL \times 1}^T \hat{P}_{\text{LTV}}^{-1} h_{NL \times 1}$$  \hspace{1cm} (6.10)$$

where $\sigma^2$ cannot be estimated separately (they are stuck together).

### 6.3 The covariance hypermatrix

The covariance hypermatrix $P_{\text{LTV}}$ contains the prior knowledge on the properties of the observed system and its structure determines the expected behavior of the LTV system dynamics.

Note, that the novelty of this chapter is to formulate the regularization technique for the special case of the estimation of a two-dimensional time-varying impulse response function. In the next sections, the practical implementation of the inclusion of the prior knowledge into the covariance matrix is addressed.

It is important to remark that there is a similar formulation of the two dimensional impulse response function of nonlinear systems in [104], where Volterra kernels are being identified by using regularization.

The structure and the indexation system in the covariance matrix are very important. Unlike in the ordinary one dimensional case, where the construction of $P$ is straightforward, here $P_{\text{LTV}}$ becomes a very large matrix consisting of different sub-matrices and therefore it can be seen as a (two dimensional) hypermatrix (2D hypermatrix is known as block matrix as well). This consists of $N \times N$ sub-matrices, and every sub-matrix contains $L \times L$ elements.

The reason is that the covariance should be computed for every $t$ time instant and within two $t$ instants it should be evaluated for every pair of $\tau$ time instants.

Considering the structure of $h_{NL \times 1}$ (see Eq. (6.3)) and $K_{\text{LTV}}$ (see Eq. (6.5)) it can be observed that the first $L$ block in $h_{NL \times 1}$ stands for the first instant of $t$ and within
Chapter 6- Building the covariance hypermatrix

this block there are $L$ instants of $\tau$. The second $L$ block of $h_{NL\times1}$ belongs to the second time instant of $t$ which has also $L$ instants of $\tau$, and so on. This is illustrated in Eq. (6.3) and in Figure 6.3.

Using this information the structure of $P_{LTV}$ is:

$$P_{LTV} = \mathbb{E}\{h_{NL\times1}h_{NL\times1}^T\} = \begin{bmatrix}
P_{LTV(0,0)} & \cdots & P_{LTV(0,N-1)} \\
\vdots & \ddots & \vdots \\
\cdots & \cdots & P_{LTV(N-1,N-1)}
\end{bmatrix} \quad (6.11)$$

Every $P_{LTV(i,j)}$ represents a sub-matrix as follows:

$$P_{LTV(t_i,t_j)} = \begin{bmatrix}
\mathbb{E}\{h[t_i, \tau_1]h[t_j, \tau_1]\} & \cdots & \mathbb{E}\{h[t_i, \tau_1]h[t_j, \tau_L]\} \\
\vdots & \ddots & \vdots \\
\cdots & \cdots & \mathbb{E}\{h[t_i, \tau_L]h[t_j, \tau_L]\}
\end{bmatrix} \quad (6.12)$$

Every element at $\{\tau_k, \tau_j\}$ indices in $P_{LTV(t_i,t_j)}$ can be computed by the expected value as follows:

$$P_{LTV(t_i,t_j),\{\tau_k, \tau_l\}} = \mathbb{E}\{h[t_i, \tau_k]h[t_j, \tau_l]\} \quad (6.13)$$

The numerical values of these elements are defined by kernel functions linking the different values of $P_{LTV(t_i,t_j)}$ and the desired behavior. This is explained in the next sections.

The numerical values in $P_{LTV}$ are derived by using kernel functions. The kernels that are used in this thesis are explained in Section 5.4.

6.4 Building the covariance hypermatrix

6.4.1 General observations

By observing the structure of Figure 6.3 and Eq. (6.3), it can be clearly seen that the main- and co-diagonals in $H_{L\times N}$ are the impulse responses.

The first identifiable IRF can be found in the main diagonal. Every element before the main diagonal refers to the (transient) impulse responses originated before the observation window. Due to the insufficient knowledge of the past values of the input they will be not estimated (see ASSUMPTION 6.7), and the corresponding elements in the covariance hypermatrix $P_{LTV}$ are not taken into account, respectively. A method to eliminate transient is shown in Section 7.2.

Next, two different ways of constructing the covariance matrix will be shown. The key idea and the goal are the same: pushing down the degrees of freedom by introducing constraints (see Appendix A.5). The first approach gives a little better
result when the time variations are less smooth. It is because fewer constraints are implemented in the covariance matrix.

The second approach is a robust method which performs a little better when the time variations are systematic — and not random. In this particular case more constraints are used. This is done at the price of a higher computational load as it is detailed in Appendix A.4.

### 6.4.2 The flexible approach

This technique is illustrated on a simplified two dimensional IRF in Figure 6.6. The diagonal elements in $H_{L\times N}$ are the impulse responses. For these elements TC or – when a higher accuracy is required – DC kernels can be used. Using these kernels the assumptions are imposed on the smoothness and decaying.

$$P_{\Gamma_\delta\{t_1,t_2\},\{\gamma_1,\gamma_2\}} = P_{\Gamma_\delta\{t_1,t_2\},\{\gamma_1,\gamma_2\}} (6.14)$$

$$P_{\Gamma_\delta\{t_1,t_2\},\{\gamma_1,\gamma_2\}} = P_{\Gamma_\delta\{t_1,t_2\},\{\gamma_1,\gamma_2\}} (6.15)$$

when $t_1 - \tau_1 = t_2 - \tau_2$ and $\tau_1 \neq \tau_2$.

Note, that on $(t_1 - \tau_1)^{th}$ diagonal the indices are two points $\{\{t_1, \tau_1\}, \{t_1, \tau_2\}\}$.

In order to use TC or DC kernels the Euclidian distances of these points are needed. The distances are the vector lengths of these points: $\sqrt{(t_1-(t_1-\tau_1))^2 + \tau_1^2} = \sqrt{2}\tau_1$ and $\sqrt{(t_2-(t_2-\tau_2))^2 + \tau_2^2} = \sqrt{2}\tau_2$, respectively. It is illustrated in Figure 6.6. (on the left figure).

![Figure 6.6: An illustration for the vector lengths used for regularization kernels. The plot on the left shows the vector lengths of two points on the same impulse response which can be used for TC or DC kernels. The plot on the right shows the horizontal distance between two impulse responses, this distance can be used for RBF kernels.](image)

The horizontal connections of the different diagonals link the time variations together and for this direction smoothing kernels can be used since no exponentially
decaying should be imposed. In this particular case RBFs are recommended to use. This can be formulated as follows:

\[ P_{\text{LTV}}(t_1,t_2),(\tau_1,\tau_2) = P_{\text{RBF}}(t_1,t_2) \]  \hspace{1cm} (6.16)

when \( \tau_1 = \tau_2 \) and \( t_1 - \tau_1 \neq t_2 - \tau_2 \) (for an illustration see Figure 6.7). At the intersections the points are correlated at maximum therefore:

\[ P_{\text{LTV}}(t_1,t_1),(\tau_1,\tau_1) = P_{\text{TC}}(\sqrt{2}\tau_1,\sqrt{2}\tau_1) \]  \hspace{1cm} (6.17)

\[ P_{\text{LTV}}(t_1,t_1),(\tau_1,\tau_1) = P_{\text{DC}}(\sqrt{2}\tau_1,\sqrt{2}\tau_1) \]  \hspace{1cm} (6.18)

when \( t_1 - \tau_1 = t_2 - \tau_2 \) and \( \tau_1 = \tau_2 \).

In every other case, the elements of the covariance hypermatrix are zero:

\[ P_{\text{LTV}}(t_1,t_2),(\tau_1,\tau_2) = 0 \]  \hspace{1cm} (6.19)

When the order of the longest considered impulse \( (L) \) and the hyperparameters are correctly chosen then the covariance hypermatrix \( P_{\text{LTV}} \) will be full rank. In theory, if the underlying system is LTI and ideally observed, by using the flexible approach, the correlation in the global time direction might be set to one. This could have the consequence that \( P_{\text{LTV}} \) becomes rank deficient. Several simulations and experiments were made on LTI systems and they all shown that by using the flexible approach and the cross validation technique, the covariance hypermatrix stays full rank.

Figure 6.7: A visualization of the regularization directions of a two dimensional IRF. The cross refers to the element before the first IRF, therefore the corresponding elements in the covariance hypermatrix are not taken into account. The diagonal lines refer to the impulse response direction and the corresponding elements shall be linked by DC or TC kernels. The vertical lines refer to the direction of time variations and the corresponding elements in the covariance matrix shall be linked by RBF kernels.
6.4.3 The robust approach

In this approach every element on every diagonal is connected to each other. It means that all the elements in the covariance hypermatrix are non-zero, therefore the number of constraints is large and hence the degrees of freedom are significantly decreased. These connections are defined for every possible pair of points. These points are connected by the relative vectorial lengths (see Figure 6.8).

As in the previous approach, the elements in the direction of the impulse responses are linked by TC or DC kernels. In the horizontal direction they are linked by RBF kernels (see Figure 6.8). This can be formulated as follows:

\[
P_{LTV(t_1,t_2),(t_1,t_2)} = P_{RB}(t_1 - \tau_1, t_2 - \tau_2) \cdot P_{TC}(\sqrt{2}\tau_1, \sqrt{2}\tau_2)
\]

(6.20)

\[
P_{LTV(t_1,t_2),(t_1,t_2)} = P_{RB}(t_1 - \tau_1, t_2 - \tau_2) \cdot P_{DC}(\sqrt{2}\tau_1, \sqrt{2}\tau_2)
\]

(6.21)

for every \(t_1 - \tau_1 \geq 0\) and \(t_2 - \tau_2 \geq 0\)

![Figure 6.8: A visualization of the vector lengths used for regularization kernels. The figure shows the vector lengths of two points on two different impulse responses which can are used as arguments in TC or DC kernel functions. The horizontal distance between two impulse responses is used as argument in the RBF kernel function.](image)

6.5 Tuning the model complexity

The choice of the model order \((L - 1)\) and the values of the hyperparameters play a key role in the estimation procedure. When the model order is increased, the error is – in general – decreased for a given dataset.

In general, when the model order is large \((L \text{ is large})\), then the model might include the noise as well, which is undesired. In the case of regularization technique it can be avoided but even then, it is important that a model is parsimonious (due to computational loads, see Section 7.3). To determine the right model order, several criteria can be used [72]. A special cross-validation technique is used here.
This procedure requires two different datasets \((D_{\text{est}}, D_{\text{val}})\). The input signals \((u_{\text{est}}[t], u_{\text{val}}[t])\) should be different, persistent excitations. Using the first input an estimation dataset \((D_{\text{est}})\) is generated, which is used to estimate the IRF via Eq. (6.9). The estimated IRF is denoted as \(\hat{h}_{NL\times 1,\text{RLS,est}|p_{h}}\) and the estimation output is denoted as \(\hat{y}_{\text{est}|u_{\text{est}},p_{h},L}\).

The second phase is the validation. Using a new input \(u_{\text{val}}[t]\), the previously estimated model is used to compute the validation output, and it is compared to the measured response of the validation input.

The third phase is the evaluation using two normalized mean square cost functions for the estimation and for the validation. The model \((\hat{h}_{NL\times 1,\text{RLS,est}|p_{h}})\) that gives the smallest validation cost is then selected.

The estimation cost function is defined as follows:

\[
V_{\text{est}}(u_{\text{est}}, p_{h}, L) = \frac{(y_{m|u_{\text{est}}} - \hat{y}_{\text{est}|u_{\text{est}},p_{h},L})^T(y_{m|u_{\text{est}}} - \hat{y}_{\text{est}|u_{\text{est}},p_{h},L})}{N} \tag{6.22}
\]

where \(p_{h}\) is the set of hyperparameters and the estimated output for \(u_{\text{est}}\) (using Eq. (6.9)):

\[
\hat{y}_{\text{est}|u_{\text{est}},p_{h},L} = K_{\text{LTV}|u_{\text{est}}} \hat{h}_{NL\times 1,\text{RLS,est}|p_{h}} \tag{6.23}
\]

The validation cost function can be computed in the same way using the validation dataset:

\[
V_{\text{val}}(u_{\text{val}}, p_{h}, L) = \frac{(y_{m|u_{\text{val}}} - \hat{y}_{\text{val}|u_{\text{val}},p_{h},L})^T(y_{m|u_{\text{val}}} - \hat{y}_{\text{val}|u_{\text{val}},p_{h},L})}{N} \tag{6.24}
\]

and from here \(\hat{y}_{\text{val}|u_{\text{val}},p_{h},L}\) is computed with the previously estimated \(\hat{h}_{NL\times 1,\text{RLS,est}|p_{h}}\).

\[
\hat{y}_{\text{val}|u_{\text{val}},p_{h},L} = K_{\text{LTV}|u_{\text{val}}} \hat{h}_{NL\times 1,\text{RLS,est}|p_{h}} \tag{6.25}
\]

In the implementation, the minimization of \(V_{\text{val}}(u_{\text{val}}, p_{h}, L)\) is done by gradient descent method. All corresponding hyperparameters \((\sigma^2/c_{\text{RBF}}, \sigma^2/c_{\text{DC}}, \alpha, \beta; \sigma^2/c_{\text{TC}}, \beta)\) have an initial value of one. When the found local minimum provides a large error, then the optimization method has to be redone with new initial values. The theoretical aspects of further possible optimization techniques are beyond the scope of this thesis. Some relevant techniques can be found in [101].

When the dominating error is due to the noise, then the following equation gives a value around the level of the noise variance.
Chapter 6- A case study

\[
\mathbb{E}\{V_{est}(u_{est}, P_{LTV})\} = \mathbb{E}\left\{\frac{\sum_{t=0}^{N-1} (y_m(u_{est}(t)) - \hat{y}(u_{est}(t), P))^2}{N}\right\} = (6.26)
\]

\[
\mathbb{E}\left\{\frac{1}{N}\cdot \mathbb{E}\{\sum_{t=0}^{N-1} (y(u_{est}[t]) - \hat{y}(u_{est}[t], P))^2\}\right\} \approx \frac{\sum_{t=0}^{N-1} \sigma^2}{N} \approx \sigma^2
\]

A good model order is found when both of these cost functions give a value around the noise variance. If, for different combinations of model orders, a validation criterion is close to the variance level, the simplest model structure (the smallest order) is recommended to be used.

The initial order of the longest considered impulse response (L) can be determined with a simple trick: the estimation and validation cost functions can be evaluated assuming that the system is LTI. Therefore a simple LS estimation method can be used which is significantly faster than the regularization.

### 6.6 A case study

In the previous sections some methodologies (Maximum Likelihood, flexible-and robust methods) have been presented to estimate linear time-varying systems but without numerical illustration. In order to have a complete picture about the performance of the estimators, a comparison based on simulation data from a system is shown here.

The true underlying system used for the performance test is given by its instantaneous transfer function (see DEFINITION 2.19) as follows:

\[
H(t, s) = \frac{(t + 1)0.01s + 1}{(5 + t)0.01s^2 + 1s + (1 + t)0.01} \quad (6.27)
\]

The simulation is repeated twice. In the first one, using the first excitation signal, an estimation dataset is created. With the same conditions (zero initials, time variations in the same state) the simulation is repeated with the second excitation signal and the validation dataset is created.

Both excitation signals are random phase multisines with a length (N) of 150 samples. The disturbing noise is white Gaussian with an SNR of 35 dB. In order to obtain the hyperparameters the cross-validation method is applied (see Section 6.5). To simplify the task, the order \((L - 1)\) is fixed to 30.

The observation window is \(t = [0..N - 1]\). In the simulation at each time instant \(t\) a steady state transfer function is obtained and inverse Fourier transformed, convolved with the corresponding time domain input signal(s). The time interval between adjacent samples is assumed to be 1.34 sec.
The following methods are compared:

- **LTI LS estimation**: ordinary least squares estimation [57], [72], [78] considering that the underlying system is LTI. This estimation is beneficial because it can give an idea about what happens if the time variations are not taken into account. The experiment is shown in Figure 6.9.

- **LTV ML estimation**: This is important in order to show the non-uniqueness issues of the ML estimation. The simulation is shown in Figure 6.9.

- **TC/DC flexible estimation**: this is the “rapid” estimation method. The simulation shows the result with the best obtained set of hyperparameters. The experiment is shown in Figure 6.10.

- **TC/DC robust estimation**: according to our expectation, this estimation method should deliver the best results. The simulation is shown in Figure 6.10.

The personal computer configuration used for the simulation can be found in Appendix A.6.

The mean square error for the model fit gives an idea about the goodness of the model fit, but since the time-varying IRF is not unique – for a given dataset – it is also beneficial to compare the model (validation) output to the measured (validation) output. In this work, to measure this kind of fit, a relative error in rms sense is used. This is defined as follows:

**Definition 6.2** The relative rms error (rrmse) refers to the total rms of the error (rms difference between the measured and estimated output) compared to the total rms of the measured system output and expressed in percentage, i.e.:

\[
rrmse(y_m, \hat{y}) = 100 \cdot \frac{rms(y_m - \hat{y})}{rms(y_m)} \quad [\%] 
\]

(6.28)

The relative rms error for this simulation is shown in Table 6.1 and in Figure 6.11. The computational times are shown in Figure 6.12 and in Table 6.1. The number of iterations to find the (sub)optimal hyperparameters is shown in Table 6.1.
Figure 6.9: A simulation is shown to compare different estimation methods. The figure on the top shows the true system. The second and third rows show the estimated two-dimensional impulse responses (on the left), and their fitting errors compared to the true system (on the right).
Figure 6.10: A simulation is shown to compare different estimation methods. On the left, estimated two dimensional impulse responses are shown. On the right, the corresponding fitting errors are shown compared to the true system. The true system is shown in Figure 6.9.
Chapter 6- A case study

Figure 6.11: A visualization of mean square errors of the different estimation methods.

Figure 6.12: A visualization of relative computational time of the different estimation methods. The smallest value (0 dB) belongs to the LTI LS estimation, other estimation methods are compared to this computational time.

Table 6.1: Comparison of the different estimation methods used for a simulation.

<table>
<thead>
<tr>
<th>Approach</th>
<th>relative time</th>
<th>absolute time [sec]</th>
<th>number of iterations</th>
<th>relative rms error [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTI LS</td>
<td>1</td>
<td>0.02</td>
<td>(1)</td>
<td>33.78</td>
</tr>
<tr>
<td>ML</td>
<td>1.68</td>
<td>0.03</td>
<td>(1)</td>
<td>80.79</td>
</tr>
<tr>
<td>TC flexible</td>
<td>$8 \cdot 10^4$</td>
<td>1836</td>
<td>260</td>
<td>4.55</td>
</tr>
<tr>
<td>DC flexible</td>
<td>$1 \cdot 10^5$</td>
<td>2313</td>
<td>290</td>
<td>4.23</td>
</tr>
<tr>
<td>TC robust</td>
<td>$2.1 \cdot 10^6$</td>
<td>49200</td>
<td>265</td>
<td>3.49</td>
</tr>
<tr>
<td>DC robust</td>
<td>$3 \cdot 10^6$</td>
<td>69170</td>
<td>364</td>
<td>3.34</td>
</tr>
</tbody>
</table>
As it can be observed, the fastest estimation method is the linear time-invariant LS estimation. It has a good fit only around the part, where the time-varying impulse responses are similar to the LTI IRF.

It comes as little surprise, but the LTI estimation beats the time-varying ML estimation. Unlike in the LTI case, where the possible solution is unique – and the time-variations are seen as a kind of noise/nonlinearity – the time-varying ML equations have very high degrees of freedom (see Appendix A.5) and therefore the solution given in Figure 6.9 is only one among the infinitely many possible solutions.

As it is expected, the robust methods based on TC/DC kernel functions provide the best results but at the price of extremely high computational load. Observe that the DC robust approach almost reaches the noise level.

While the latter gives the best result with a computational time of 19 hours (on an average PC), the flexible TC method gives also a nice result but within half an hour. So the flexible methods can be the most attractive for many applications.

Note that, once a two dimensional regularized IRF estimation is obtained, a new simulation carried out relatively fast. The time consuming part is the cross-validation.

### 6.7 Conclusions

In this chapter a powerful time domain method is presented to estimate a two dimensional impulse response function.

The proposed procedure is easy to use and it consists of four simple steps:

- prepare the data sets (Section 6.5),
- construct the observation and covariance matrices (Section 6.2, 6.4),
- estimate the model order and hyperparameters given by the cross-validation method (Section 6.5),
- calculate the coefficients of the two dimensional IRF (Section 6.2.2).

The proposed method is beneficial because:

- it is a simple extension of the LTI theory,
- non-frozen impulse responses are obtained unlike the “frozen” impulse response.

In spite of the beneficial aspects of the proposed methods, a disadvantage must be taken into account. The computational load is very high. To avoid this problem, a partial solution is given in the following Chapter.
Chapter 7
Extension of the proposed two dimensional regularization technique

In the previous chapter, a general description and implementational algorithms are given for the identification of linear time-varying systems without taking care of the limits on the computational resources such as operational memory and computational time.

Despite the numerous beneficial aspects of the proposed methods, there is a main drawback, namely the computational load. Unlike in the LTI regularization case, where the matrix sizes and the computational loads are relatively low, here, considering the matrix sizes it can be clearly seen that by increasing the length of the measurement the required memory is quadratically growing. It means that $N$ cannot be chosen more than a few hundreds. Next to the exaggerated memory usage, the computational load explodes due to the inversion problem.

A further issue is to create an estimation and a validation dataset for an LTV system. This is not a trivial task. In the case of LTI systems, the measurement can be repeated at any time but for an LTV system it must be guaranteed that all the time variations are exactly in the same state during the estimation and validation measurements.

Last but not least, it would be very beneficial to have a procedure to eliminate the transient signals from the real response of the observed system.

In this chapter some possible solutions are provided for the above-mentioned problems.
7.1 Working with real measurements

When working with time-varying systems, it is often very difficult – or impossible – to repeat the experiments in such a way that the time variations are the same.

To avoid that problem, we construct the estimation and validation data sets \((D_{\text{est}}, D_{\text{val}})\) from the data of a single experiment. To achieve that, the measurement data are split into two parts. Using interleaved points, consequently, every even/odd sample is used for estimation/validation. This technique is illustrated in Figure 7.1.

If the sampling frequency \((f_s)\) is high enough compared to the highest considered frequency component \(f_{\text{max}}\) (i.e. \(f_s > 4f_{\text{max}}\)) then we are allowed to do this (due to the highly correlated samples).

In this case it is important to make a clear distinction between the sampling frequency (sampling rate) of the signal generator and the data acquisition system. A full band white excitation can only be used, if the data acquisition collects samples at least four times faster than the signal generator generates the samples. If it is not possible (because, for instance, the same sampling frequencies are used), then only band limited excitations can be used to satisfy the \(f_s > 4f_{\text{max}}\) constraints.

There is an alternative sample selection strategy for the cross-validation technique to make the hyperparameter fitting procedure. The model input is given by the full samples and the output is computed on the full dataset. The odd samples of the output are used to tune the parameters, and the even samples of the output are used to evaluate the validation cost function. In this particular case full band excitation can be used.

It is important to remark that the reliability of this method is limited compared to the ordinary cross-validation method (because the datasets are very similar).

Of course, if the conditions allow us to repeat the measurement many times – with respect to the same time variations – it is easier and more reliable to capture the datasets from different measurements.

![Creation of the estimation and validation datasets from a single dataset.](image)
7.2 Transient elimination

In the LTV baseline model (see Eq. (6.1)–(6.2)) it is assumed that the system is initially at rest, or in other words the initial conditions are zero.

If this is not the case, transient errors will appear but only for a limited time and then they disappear [84]. An illustration of the transient is shown in Figure 7.2. In this section a possible elimination of transient is discussed.

The observable transient term can be imagined as an impulse response to a Kronecker delta function at the excitation time $t = 0$ (see Section 4.2.1).

7.2.1 Dependence on the input excitation

Principally, the effect of the transients depends on the type of the excitation signal. For instance, when a stepped sine excitation is used, despite the beneficial SNR advantage, a big drawback is needed to be taken into account: the longer measurement time. First, it is required to measure at least one period at each frequency harmonics. Secondly, it is also required to wait until the transients have disappeared at each harmonic.

When an LTI system is being measured, in some cases this kind of delay in the measurement is acceptable. Unfortunately, this does not hold for LTV systems. During this delay – while waiting for transients to have disappeared – a significant time-variation can occur. In this case we have information only at one frequency line at a given time, but due to the time-variations, the characteristic at other frequencies is already varied.

Therefore, other signals that excite all frequencies of interest simultaneously have to be used, such as PRBS, Gaussian white noise or random phase multisine (see Chapter 4.). The problem with the noise excitation is that a spectrum leakage must be taken into consideration, since the noise excitation is not periodic. Although it is possible to use the Gaussian white noise as a periodic excitation – with repeated sequences – but this will result in random holes in the spectrum which is undesired. The PRBS excitation is periodic, but its length cannot be chosen arbitrarily. The advised solution is the random phase multisine excitation – due to its flexibility, spectral properties, and user-friendly design. Note, that there are many alternative advanced signals such as [105], [106]. A survey about the excitation signals can be found in [107].
7.2.2 LTI transient elimination

In the case of LTI systems, there are some common practices which can be used to eliminate the effect of the transient, for instance:

- If the excitation is periodic, then it is possible to discard some periods such that the measurement becomes transient free [72], [84]. This technique could be used for LTV systems only, if the time-variations are periodically repeating and we are able to measure these repetitions.
- Different windowing methods can also help to eliminate the effects of transient [108]. In the case of time-varying systems, it has a limited meaning since the spectral properties are continuously changing.
- Using the local polynomial method (LPM) in the frequency domain is also a good solution for LTI systems [72], but due to the varying spectral properties it is not directly possible for LTV systems.

As can be seen, none of these techniques can be (directly) applied to the transient elimination problems of LTV systems. Therefore a more advanced technique is needed. A proposal will be explained in the following section.

7.2.3 Problem formulation

In this section a time domain transient elimination technique is presented. Let us start with three important assumptions which will be needed for the proposed elimination technique.

**Assumption 7.1** It is assumed that the excitation signal is proper (persistent) and the past values of the excitation are independent of the future values. This assumption will be needed for Eq. (7.3). Note, that the Gaussian white noise and the full random phase multisine satisfy this assumption, because its samples are totally uncorrelated.

**Assumption 7.2** It is assumed that the observable transient IRF $h_{\text{transient}}$ in the finite observation window has a finite order of $N_{\text{transient}}$ which is typically smaller than or equal to the order of longest considered impulse response, i.e. $N_{\text{transient}} \leq L$.

This assumption is illustrated in Figure 7.2. Observe that the transient impulse response function originates before the observation window and its observable (measurable) length is smaller than the LTV impulse responses.
It is important to remark that in the classical LTI theory the transient term is handled as a deterministic process. In the regularization technique – like in the case of the steady-state impulse response function – it is seen as a zero mean Gaussian random process. This will allow us to use some statistical methods.

**Assumption 7.3** It is assumed that the transient term is independent of the observation noise.

Using these assumptions (and the fact that the transient term can be imagined as an impulse response to a Kronecker delta function), the Eq. (6.1)–(6.2) can be extended as follows:

\[
y_m[t] = y[t] + y_{\text{transient}}[t] + e[t] = \\
\sum_{\tau=0}^{t-1} h[t, \tau] u[t - \tau] + \sum_{\tau=0}^{N_{\text{transient}}-1} h_{\text{transient}}[\tau] \delta[t - \tau] + e[t] = \\
\sum_{\tau=0}^{t-1} h[t, \tau] u[t - \tau] + h_{\text{transient}}[t] + e[t].
\]

(7.1)

where \(h_{\text{transient}}[t] = 0\) and \(y_{\text{transient}}[t] = 0\), when \(t \geq N_{\text{transient}}\).

This can be rewritten in vector form as:

\[
y_m = y + y_{\text{transient}} + e = K_{\text{LTV}} h_{NL\times1} + h_{\text{transient}} + e
\]

(7.2)

This new setup is illustrated in Figure 7.3.
It is very important to remark that the transient IRF has similar properties as the baseline model and the observed system: it is smooth and exponentially decaying \cite{72}. The aim of this section is to provide a methodology to identify $h_{\text{transient}}$ and to decrease its effects.

7.2.4 The proposed transient elimination technique

The key idea in the proposed technique is that we make use of the statistical framework what we have used in the regularization technique, therefore a theoretical analysis is needed. We need to check the dependency – covariance – between the measured output and the transient term. Using this information, a modified model can be built. This dependency can be computed as follows:

$$\text{COV}\{y_m, y_{\text{transient}}\} =$$

$$\mathbb{E}\{(y_m - \mathbb{E}\{y_m\})(y_{\text{transient}} - \mathbb{E}\{y_{\text{transient}}\})\} =$$

$$\mathbb{E}\{(y_m)(y_{\text{transient}}^T)\} = \mathbb{E}\{(y + y_{\text{transient}} + e)(y_{\text{transient}}^T)\} =$$

$$\mathbb{E}\{y_{\text{transient}}y_{\text{transient}}^T\}$$

Equation (7.3)

It can be observed that the transient term is independent of the true output, and this will be used to extend the regularized two dimensional method. This allows us to eliminate the undesired transient term. In this particular case the – observation and correlation – hypermatrices should be modified. Next, the proposed algorithm follows.

First, an extended IRF vector $h'$ has to be defined. This is a $NL + N_{\text{transient}} \times 1$ vector which is the cumulative – joint – impulse response function consisting of the time-varying and transient IRFs.

The IRF vector $h'$ defined as follows:

$$h' = \begin{bmatrix} h_{NL \times 1}^T & h_{\text{transient}}^T \end{bmatrix}^T$$

Equation (7.4)
The **Assumption 7.1** allows us to define the observation matrix of transient IRF. The $N \times N_{\text{transient}}$ $K_\delta$ observation matrix is defined as follows:

$$
K_\delta = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & \ddots & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0
\end{bmatrix}
$$

(7.5)

From here the cumulative – joint – observation hypermatrix is:

$$
K' = [K_{LTV} \ K_\delta]
$$

(7.6)

where $K'$ is a $N \times (NL + N_{\text{transient}})$ matrix.

Next, the $N_{\text{transient}} \times N_{\text{transient}}$ covariance matrix of the transient impulse response function is defined as follows:

$$
P_{\text{transient}} = \mathbb{E}\{h_{\text{transient}}h_{\text{transient}}^T\} = \\
\begin{bmatrix}
P_{\text{transient}(0,0)} & \cdots & P_{\text{transient}(0,N_{\text{transient}} - 1)} \\
\vdots & \ddots & \vdots \\
\vdots & \ddots & \vdots \\
P_{\text{transient}(N_{\text{transient}} - 1,0)} & \cdots & P_{\text{transient}(N_{\text{transient}} - 1,N_{\text{transient}} - 1)}
\end{bmatrix}
$$

(7.7)

where any element at $\{t_i, t_j\}$ index in $P_{\text{transient}}$ can be computed as follows:

$$
P_{\text{transient}(t_i,t_j)} = \mathbb{E}\{h_{\text{transient}}(t_i)h_{\text{transient}}(t_j)\}
$$

(7.8)

This leads to the cumulative – joint – covariance hypermatrix:

$$
P' = \mathbb{E}\{h'h'^T\} = \begin{bmatrix}
P_{\text{LTV}} & 0 \\
0^T & P_{\text{transient}}
\end{bmatrix}
$$

(7.9)

where $P'$ is a $(NL + N_{\text{transient}}) \times (NL + N_{\text{transient}})$ matrix and $0$ is a $NL \times N_{\text{transient}}$ null matrix.

The corresponding hyperparameters are included in the cumulative vector $p'_h$ such that $p'_h = [p_h^T \ p_{h,\text{transient}}^T]^T$, where $p_{h,\text{transient}}$ represent the hyperparameters of the transient term. This means that there will be two different sets of hyperparameters to be estimated.

Due to the assumptions of transient IRF (smoothness and decaying), the numerical values in $P_{\text{transient}}$ can be determined by TC or DC kernels using the $p_{h,\text{transient}}$ set of hyperparameters as follows.
\[ P_{\text{transient}}(t_i, t_j) = P_{TC}(t_i, t_j) \]  \hspace{1cm} (7.10)

\[ P_{\text{transient}}(t_i, t_j) = P_{DC}(t_i, t_j) \]  \hspace{1cm} (7.11)

Note, that just like in the previous cases, the usable kernel functions are not limited to TC/DC kernels only and other kernel functions can be used as well as long as they satisfy the assumption made on the behavior of the transient term (decaying and smoothness).

The above mentioned terms allow us to redefine the cost function of the problem:

\[ V' = ||y_m - K'h'||_2^2 + \sigma^2 h'^T P'^{-1} h' \]  \hspace{1cm} (7.12)

and from here the regularized solution of \( h' \) is:

\[ \hat{h}' = (K'^T K' + \sigma^2 P'^{-1})^{-1} K'^T y_m \]  \hspace{1cm} (7.13)

At this stage it is important to recall Eq. (6.10). Although the same \( \sigma^2 \) is used for the LTV and for the transient covariance matrices, belonging to the covariance matrices there are two independent scalar hyperparameters. As it explained in Section 6.2.2, these scalar hyperparameters must be estimated together with \( \sigma^2 \) (unless we know the exact value of \( \sigma^2 \)).

The hyperparameters in \( P' \) need to be computed for \( P_{\text{LTV}} \) and for \( P_{\text{transient}} \) in the following way:

i. estimate \( h_{NL \times 1} \) by \( p_h \) set of hyperparameters of \( P_{\text{LTV}} \) assuming that no transient is present (this way the computational load is significantly smaller, see the problem formulation in Section 7.1). Further, \( p_h \) can be used as initial values of \( p_h' \)

ii. estimate \( \hat{h}' \) by \( p_h' = [p_h \ p_{h,\text{transient}}] \) in \( P' \)

When a higher accuracy is required, the following steps can be applied:

iii. remove the transient from the (validation) output signal,

iv. re-estimate the hyperparameters \( p_h \) for \( P_{\text{LTV}} \) using \( p_h' \) as initial values,

v. build a new model of \( h_{NL \times 1} \)
7.2.5 A case study

In this section a simulation example is shown. The underlying system is defined in Section 6.6. For illustration purposes, in order to disturb the system, a transient signal is manually injected. The length of the transient \(N_{\text{transient}}\) is 20. The transient is shown in Figure 7.4. A more realistic case is shown in Section 7.3.1.

The rest of the simulation parameters are kept to their normal values (multisine excitation, \(L = 30, N = 150, \text{SNR} = 35\, \text{dB}\)).

To illustrate the method, the robust DC approach is used here.

Figure 7.5 shows the performed test for two cases. The figure on the top shows the true underlying system. The second row shows the estimation when the model defined in Eq. (7.1) and the transient elimination algorithm are used. The third row shows the simple robust DC estimation, when no transient is taken into account and the model given in Eq. (6.1) is used.

The fitting errors are shown in Table 7.1. Observe that even in presence of transient, the proposed method delivers a good result comparable to the transient free case. If the transient model is not used the error explodes.

Table 7.1: Comparison of the DC robust method used for a simulation with the transient response.

<table>
<thead>
<tr>
<th>Approach</th>
<th>relative rms error [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC robust, transient free data</td>
<td>3.34</td>
</tr>
<tr>
<td>DC robust, modelling the transient</td>
<td>11.35</td>
</tr>
<tr>
<td>DC robust, without modelling the transient</td>
<td>58.43</td>
</tr>
</tbody>
</table>

It is important to remark that the additional computing time required for the transient elimination is very low compared to the complexity of LTV estimation.

Figure 7.4: The injected transient is shown in black. The estimated transient is shown in grey.
Figure 7.5: The regularized estimates of two dimensional IRF shown in presence of transient. The top figure shows the true underlying system. The second and third rows show the estimates of the underlying system.
Chapter 7 - Estimation from a large dataset

7.3 Estimation from a large dataset

Considering the matrix sizes in the previous chapter (Eq. (6.3)–(6.5) and (6.11)–(6.12)) it can be clearly seen that by increasing the length of the measurement, the requested operational memory grows quadratically.

In Eq. (6.1) it seems to be handy to choose the longest considered impulse response function to be the length of the observation window i.e. \( L = N \) because by choosing \( L \ll N \) the memory needs are significantly decreased. Apart from the exaggerated memory usage, the computational load explodes due to the inversion problem in Eq. (6.9).

A partial solution can be a re-ordered equation providing the same result [96] but with smaller computational load. To do this the following matrix equality can be used [102]:

\[
(I_{k \times k} + A_{k \times j}B_{j \times k})^{-1} A_{k \times j} = A_{k \times j}(B_{j \times k}A_{k \times j} + I_{j \times j})^{-1}
\]  

(7.14)

Let us re-order Eq. (6.9) as follows:

\[
\hat{f}_{NL \times 1,RLS} = (K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1} K_{LTV}^T y_m = \\
(P_{LTV} K_{LTV} + \sigma^2 I_{NL \times NL})^{-1} P_{LTV} K_{LTV}^T y_m = \\
\left(\sigma^2 I_{NL \times NL} + \frac{A}{P_{LTV} K_{LTV}^T} \frac{B}{K_{LTV}}\right)^{-1} \frac{A}{P_{LTV} K_{LTV}^T} y_m = \\
P_{LTV} K_{LTV}^T (K_{LTV} P_{LTV} K_{LTV}^T + \sigma^2 I_{N \times N})^{-1} y_m
\]

(7.15)

Here the inversion problem stands for \( N \times N \) size instead of \( NL \times NL \). This formula was already applied for the problem in Section 6.6.

In a typical LTI identification scenario, when the number of parameters to be estimated is significantly smaller than the available number of samples, many available algorithms can be used [101]. Unfortunately, in our case there are more parameters than data samples and therefore the provided algorithms cannot be used.

Despite this fact, a resource saving possibility is developed to speed up the calculation and to be able to handle a very large dataset. This technique is explained in the next section.
7.3.1 Sliding window technique

The key idea is that instead of computing the LTV model in one step for a large dataset, a special sliding window can be used resulting in a smaller computational load.

The width of the sliding window ($N_w$) cannot be smaller than $L+1$. This is because in this window there must be at least one full length impulse. The maximum width is limited to the available operating memory (see Appendix A.4.2) and the width is a trade-off between the width and the computational time.

In general, the proposed minimal width of the sliding window ($N_w$) is $3L$. In the general case this is large enough to have a high quality estimate inside of the window, but it is small enough to have a fast computation.

In this section a simulation example is shown to demonstrate the efficiency of the proposed algorithm. The underlying system is shown in Figure 7.6. The system is excited by a full-band random phase multisine of length $N = 1050$. The longest considered impulse response $L$ is 50, the signal to noise ratio is 35 dB.

To illustrate the difficulties, let us assume that we estimate an LTV model in one step using this observation. In this particular case, the memory needs are the following – provided that the elements are stored in IEEE 754 double format:

- The covariance hypermatrix: $(NL)^2 \cdot 64$ bit i.e. $(1050 \cdot 50)^2 \cdot 8$ bytes. This boils down to more than 22 GB operating memory need. Note that this one has never to be stored.
- The observation hypermatrix: $N \cdot NL \cdot 64$ bit i.e. 441 MB is needed.
- The two dimensional IRF: $N \cdot L \cdot 64$ bit i.e. 420 kB is needed.

If we use the proposed sliding window technique with a width $N_w$ of 150, then the covariance hypermatrix has the following memory needs: $(N_wL)^2 \cdot 64$ bit i.e. 450 MB. So if we use a sliding window technique it is possible to gain around 21 GB memory usage in this example.

The first idea comes from the signal processing, where a simple moving window can be used [55]. Here, the moving window has a step size of one, it means that the measurement window is shifted one by one towards the last data sample ($N - 1$).

The result of this classical sliding window technique is shown in Figure 7.7. As can be seen, the estimated two dimensional IRF has nothing to do with the true underlying system, therefore this – simple – solution is not recommended to be used.
In order to have a better understanding on the problem of the sliding window technique, the construction of $H_{k \times N}$ needs to be revised with respect to the new situation (measurement setup).

In the previous chapter, the considered measurement window is fixed to the timespan $t = 0, \ldots, N - 1$. So no elements in $h[t, \tau]$ for $t - \tau < 0$ are considered in
the estimation (see Eq. (6.3)), although as mentioned, these elements exist but we are unable to identify them.

Let us assume now that we shift the measurement window to the starting point \( t_s \) \( (t_s > 0) \). In this case the measurement window will cover the following time instants: \( t = t_s, \ldots, t_s + N_w - 1 \).

Considering the new measurement window, the values in \( H_{L \times N} \) will be:

\[
H_{L \times N} =
\begin{bmatrix}
  h[t_s,0] & h[t_s + 1,0] & \cdots & h[t_s + N_w - 1,0] \\
  h[t_s - 1,1] & h[t_s + 1,1] & \ddots & \vdots \\
  \vdots & h[t_s + 1,2] & \ddots & \vdots \\
  h[t_s - N_w + 1, L - 1] & \cdots & \cdots & h[t_s + N_w - 1, L - 1]
\end{bmatrix}
\]

This means that there are valid – and at this time identifiable – elements in \( h[t, \tau] \) before \( t_s < 0 \).

In order to have a proper estimation, we need to take these elements into account when identifying \( h[t_s, \ldots, t_s + N_w - 1, \tau] \).

We need to slide the window from \( t_s = 0 \) towards \( N - 1 \) with a certain step size. Provided that there is an overlap between different windows, it is possible to have access to these new elements appearing in (7.16) and correct the measurement with these terms. Next, a proposed technique is explained and illustrated in Figure 7.8. The steps of the algorithm are marked in Figure 7.8.

**Initial steps:**

i. **Initialization of the window parameters:**

Determine the length of the longest considered impulse response function \((L, \text{see Section 6.5})\). The window width \( N_w \) should be at least \( 3L \) but not too large (due to the limited amount of operating memory and computational load). Set the step size of the sliding window to \( N_w - L \). The time window will be now: \( t_{\text{span}} = t_s, \ldots, t_s + N_w - 1 \) (with \( t_s = 0 \)).

ii. **Obtain the hyperparameters:**

Take the first \( N_w \) samples (at \( t_{\text{span}} \)) and run the cross-validation method: obtain a set of hyperparameters \( p_h \) and the covariance hypermatrix \( P_{LTV} \).
Iterative estimation and compensation:

iii. Estimation in the window

Once the estimation at time instants $t_{\text{span}}$ is ready, take the diagonal elements at time instants $t'_{\text{span}} = t_s, \ldots, t_s + N_w - L - 1$ from the reshaped estimated two dimensional impulse response function $\hat{H}_{L\times N}$. These are the first $N_w - L - 1$ diagonals and they are a partial estimation of $\hat{h}[t'_{\text{span}}, \tau]$ and they needed to be stored in a $N \times L$ size matrix at $t'_{\text{span}}$ indices (in the same diagonal orientation, see Figure 7.8, on the bottom).

iv. Correction of the datasets (transient removal):

Using $\hat{h}[t'_{\text{span}}, \tau]$ the measurements need to be corrected. Simulate the (estimation and validation) output at $t_{\text{span}}$ time instants by $\hat{h}[t'_{\text{span}}, \tau]$. Subtract it from the (estimation and validation) output $y_m$ such that $y_m'[t_{\text{span}}] = y_m[t_{\text{span}}] - \bar{y}$. By doing this – theoretically – $y_m'$ will not contain the responses of $h[t'_{\text{span}}, \tau]$. $y_m'$ will contain the effects of the impulse responses after the time $t_s + N_w - L$ only, which will be identified in the next step. Because the estimation is not ideal, the corrected output measurement can contain some systematic errors (see Figure 5.2). Therefore it is recommended to use the transient elimination procedure as well detailed in Section 7.2.

v. Stepping the window:

Move the sliding window with one step such that $t_s = t_s + N_w - L$. Using the previously computed covariance hypermatrix, obtain a new two dimensional IRF.

vi. Go back to (iii.)

Continue with step (iii.) until the sliding window arrives at the last sample $(N - 1)$.

vii. Leave the iteration:

When the sliding window arrives at the last data sample, take all the impulse responses.
viii. **Post processing:**

If at the junctions of the two partial IRF estimations the transitions are not smooth enough (see Figure 7.9, the ripples), an additional RBF smoothing can be applied around the intersections of different IRFs (see at the bottom of Figure 7.8). In this particular case the objective is to have a smooth transition between two estimated surfaces.

Figure 7.8: An illustration of the proposed sliding window technique.

To illustrate the efficiency of the proposed method a test is performed with the system shown in Figure 7.6 for a shorter measurement ($N = 500$) and for the longer measurement ($N = 1050$). For the test the flexible DC approach is used to estimate the two dimensional IRF.

Figure 7.9 and Figure 7.10 show the performed test for the case when the longer observation is used and no transient compensation is made between different partial estimates.
Figure 7.9 and Figure 7.11 show the performed test for the case when the longer observation is used and the transient compensation together with the additional RBF smoothing is made between different partial estimates.

Table 7.2 shows the estimation errors for different scenarios. In this particular case the transient elimination algorithm is used to eliminate the systematic errors as well.

Figure 7.9: An example of the sliding window method. On the top the true system is shown. Second and third rows show different estimates.
Figure 7.10: An estimation of a system with the proposed sliding window technique without transient compensation. The line shows the true time domain system output. The crosses (×) show the fitting error.

Figure 7.11: An estimation of a system with the proposed sliding window technique using transient compensation. The line shows the true time domain system output. The crosses (×) show the fitting error.

Table 7.2: Comparison of the DC robust method used for a simulation with the transient response.

<table>
<thead>
<tr>
<th>Approach</th>
<th>relative rms error [%]</th>
<th>computational time [sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC flexible, short data record, no sliding window is used, see Section 6.6</td>
<td>4.31</td>
<td>36117</td>
</tr>
<tr>
<td>DC flexible, short data record, sliding windowing with transient elimination and RBF smoothing</td>
<td>4.48</td>
<td>2325</td>
</tr>
<tr>
<td>DC robust approach, long data record</td>
<td>not computable</td>
<td></td>
</tr>
<tr>
<td>DC robust approach, long data record, sliding windowing without transient elimination</td>
<td>26.77</td>
<td>2335 (+10.4 additional)</td>
</tr>
<tr>
<td>DC robust approach, long data record, sliding windowing with transient elimination</td>
<td>8.40</td>
<td>2423 (+87.9 additional)</td>
</tr>
<tr>
<td>DC robust approach, long data record, sliding windowing with transient elimination and RBF smoothing</td>
<td>4.90</td>
<td>2428 (+4.2 additional)</td>
</tr>
</tbody>
</table>
It can be clearly concluded that the method performs very well. Of course, the fitting error is above the “ordinary” one window method. The model quality given the SNR level is still good. It is important to highlight the importance of the transient eliminating procedure in the sliding window method.

Apart from the extremely decreased memory usage, the computational speed is significantly faster. The personal computer configuration used for the simulation can be found in Appendix A.6.

7.3.2 Time-varying hyperparameters

When the system dynamics over the time are rapidly changing, these time variations have to be tracked as well. Theoretically it is possible to re-tune the hyperparameters step by step (for each impulse response). Due to the Assumption 2.2, this is not necessary because the time-variations are limited.

In the proposed method, it is possible to combine the hyperparameter re-tuning with the proposed sliding window technique.

This can be implemented by the following simple algorithm:

- Tune the hyperparameters for the first time window \((t = 0, \ldots, N_w - 1)\) as it is stated in Section 6.5.
- Define a threshold (error) level which indicates that the hyperparameters need to be re-tuned.
- Execute the sliding window algorithm till (v.) and check if the estimation error level is still under the threshold. If the error is above this threshold level, then the hyperparameters must be re-tuned.

7.3.3 Storage capacity

In the previous section a computational resource saving methodology is presented.

However, even if the proposed sliding window is used, the operational memory limit can be easily reached with a relatively long measurement. Some authors propose to save the coefficients of the impulse response function to a file.

Another solution is that instead of saving the coefficients of \(\hat{h}[t, \tau]\), it is possible to treat the two dimensional impulse response function as a surface. Then this surface can be described by polynomial interpolation resulting in lower memory needs.

In this thesis it is done by B-splines as discussed in the next part of the thesis.
NONPARAMETRIC IDENTIFICATION USING B-SPLINES
Chapter 8
An introduction to B-splines

Splines were first used in engineering design as drafting tools to create special shapes. It allowed the draftsmen to draw smooth curves with the constraint of the control points. This is illustrated in Figure 8.1, where the pins are the control points.

A special class of splines is the B-splines, where B stands for Basic [109], [110]. Nowadays these B-splines are very popular in computer graphics [111], [112], [113] and in statistics for noisy data fitting [114], [115], [116]. With some modifications, they can also be used for system identification purposes [22], [13], [117], [118] [119].

The main advantage of using B-splines over other spline based smoothing techniques is that the system equations are less complex (it is not necessary to compute and handle the derivatives), fewer assumptions should be made and for any given set of knots, the B-splines are unique. The advantage of the B-spline based identification is significant in real measurement applications, where the observations are disturbed with noise.

Figure 8.1: An example is shown for a Spline used to draw a smooth curve. Source: [120].
8.1 One dimensional B-Splines

In this section the classical one dimensional B-spline technique is discussed. The B-splines are piecewise polynomial functions and their usefulness lies in the fact that any B-spline function above a given set of knots can be expressed as a linear combination of B-splines. Moreover, they are uniquely determined for a given degree. Further advantage of B-splines is that they can have a very high continuity and all pieces have the same continuity within their range of support. Let us continue with the definition of a B-spline function.

**Definition 8.1** A B-spline approximation/smoothing function \( s(x) \) of degree \( d_x \) is a linear combination of basis functions \( B_i^{d_x}(x) \) and ordinates of the control points \( c_i \) evaluated on the axis \( x \) at point \( i \):

\[
s(x) = \sum_{i=0}^{n_x-1} c_i B_i^{d_x}(x)
\]  

(8.1)

where \( n_x \) is the number of control points.

For better understanding, here an intuitive introduction will be given.

![Figure 8.2: An example is shown for the one dimensional second degree B-spline smoothing. The circles (○) are the control points, the thin lines are the basis functions multiplied with the corresponding (coefficients of) control points. The coefficients equal the ordinates of the control points. The thick line is the smoothed approximation: the linear combination of basis functions and control points. The crosses (×) are the knots belonging to Greville abscissas and hence to the control points. The dots (●) are the additional knot points. Observe that in this case (2nd or higher order B-splines) the approximation does not go through the control points. It would be possible to achieve this, but for this, for instance, solution of a LS fitting algorithm is necessary.](image-url)
B-splines are determined by:

- \( d_x \) degree: this controls the smoothing, the higher the degree is, the smoother the function \( s(x) \). This is because the B-spline curves have \( C^{d_x-1} \) continuity. \( C^{d_x-1} \) indicates the order of continuity. For instance, \( C^{-1} \) means that the curve includes discontinuities and \( C^1 \) means that the first derivatives are continuous. Further details can be found in [109].

- \( n_x \) number of coefficients \( c_i \) of the control points \( P_i(x_{gi}, c_i) \): they weigh the basis functions \( B^d_i(x) \). In this example, these coefficients are determined by the control points \( P_i(x_{gi}, c_i) \): the ordinates of \( P_i \) are the coefficients (in identification, the coefficients can be determined by minimizing a cost function of the fit at more points). The basis functions are (horizontally) centered around their control points.

- \( x_{gi} \) abscissas of the control points \( P_i(x_{gi}, c_i) \): they are the so-called Greville abscissas which are the positions of the control points on the x-axis. These are important for implementational purposes. From identification viewpoint, the control points mean the coefficients of the B-spline polynomials. To follow the traditions, in this thesis the coefficients will be called control points.

- \( k \) number of knots \( x_i \): a non-decreasing sequence of points on x-axis which specifies the domain for calculating and determining the basis functions (see Definition 8.2). For implementation purposes it is important that the first \( n_x \) knots are linked to the (left shifted) values of the (Greville) abscissas of control points. The value of this shift is \( (d_x + 1)/2 \) (in grid steps). Hence the sequence of \( x_i \) is defined as follows:

\[
x = [x_0 \leq x_1 \leq \cdots \leq x_{n_x-1} \leq \cdots \leq x_{k-1}]
\]  

(8.2)

The first \( n_x \) knots can be determined from the Greville abscissas as follows:

\[
x_0 = x_{g0} - \frac{(d_x + 1)}{2}, \quad \cdots, x_{n_x-1} = x_{g_{n_x-1}} - \frac{(d_x + 1)}{2}
\]  

(8.3)

Due to the B-spline constraints, there is a connection for the above-mentioned terms to define the minimal number of knots that is required to determine the B-splines:
Chapter 8- One dimensional B-Splines

\[ k_{\text{min}} = d_x + n_x + 1 \]  

(8.4)

where \( k \geq k_{\text{min}} \) and \( k_{\text{min}}, k, n_x \in \mathbb{N}, d_x \in \mathbb{N}_0 \).

Figure 8.2 shows a concrete example. In this case there are three control points, namely \( P = \{(1.5, 1); (2.5, 0.5); (3.5, 1)\} \). From these \( P_i \), the coefficients can be obtained \( c = \{1; 0.5; 1\} \) as well as the Greville abscissas \( x_g = \{1.5; 2.5; 3.5\} \).

To illustrate the most used B-splines form, the degree \( d_x \) is chosen equal to two. By using Eq. (8.4), the (minimal) necessary number of knots can be determined as follows: \( k = d_x + n_x + 1 = 2 + 3 + 1 = 6 \).

There are three knots strictly belonging to the control points \( P_i \), and three additional knots to satisfy the constraint in Eq. (8.4). This can be captured in the example. The first control point \( P_0 \) establishes the first basis function \( (B_0^2) \). This basis function is starting at the first knot \( (x_0) \) which is the shifted value of the first Greville abscissa: \( x_0 = x_{g0} - (d_x + 1)/2 = 1.5 - (2 + 1)/2 = 0 \). The last control point \( (P_2) \) establishes the last basis function \( (B_2^2) \). This basis function is started at the third knot \( (x_2) \) which is linked to the shifted value of the last abscissa: \( x_2 = x_{g2} - (d_x + 1)/2 = 3.5 - (2 + 1)/2 = 2 \). As it is explained later, each basis function has a width of degree \( d_x \) plus one. In this case it means that this length is three and therefore the last basis function has the end at the sixth knot \( (x_5) \).

When the knots (and the corresponding control points) are equidistantly spaced, the B-spline is said to be uniform, otherwise it is called non-uniform. In this work only uniform splines are considered.

Next, the definition of the basis function is given by a recursive formula as follows:

**DEFINITION 8.2** \( B_i^{d_x}(x) \) is the B-spline basis function of degree \( d_x \) on the axis \( x \) at the knot point \( x_i \), defined by the de Boor, Mansfield, Cox recursion [121]:

\[
B_0^p(x) = \begin{cases} 
1 & \text{when } x \in [x_p, x_{p+1}] \\
0 & \text{otherwise}
\end{cases}
\]  

(8.5)

\[
B_i^{d_x}(x) = \frac{x - x_p}{x_{p+d_x} - x_p} B_i^{d_x-1}(x) + \frac{x_{p+d_x} - x}{x_{p+d_x+1} - x_{p+1}} B_{i+1}^{d_x-1}(x)
\]  

(8.6)

It is important to remark that some authors define the order (instead of the degree) in different ways [109], [121].

An illustration of basis functions with different degrees is shown in Figure 8.3. As can be seen from the figures, the width of a basis function is exactly the degree
number plus one (e.g. if the degree is zero, then the length is one or if the degree is two, the length is three). The area of all basis functions is one.

As it was emphasized previously, B-spline techniques can be used for interpolation or approximation, depending upon the degree and the set of control points. In general, if the degree is zero then it is a zero order interpolation (see Figure 8.3 for $d = 0$); if the degree is set to be one then it is a linear interpolation (see Figure 8.3 for $d = 1$); otherwise ($d > 1$), it is a smoothing approximation (see Figure 8.3 for $d = 2,3$). This means that the higher the degree is, the smoother the function, since by increasing the order, the number of continuous derivatives increases. In other words, the degree controls the smoothness. As it can be seen (Figure 8.3, right column), the bias (deviation from the control points) increases with the degree $d$. Thus, a small bias will be achieved if degree $d$ is small.

However, it can be shown that the variance increases as the degree decreases [115]. In system identification – where both noise and/or transients are present – the degree should not be chosen too small. The best value for degree $d$ will be the one that trades off bias against variance. All derivatives up to degree $d$ are imposed to be continuous at all the knot points, which tacitly means more constraints. This allows the user to control the bias/variance trade off: increased smoothing reduces the noise at the cost of a larger bias [122]. This can be captured in the Figure 8.3 (bottom rows, $d = 2,3$) where the shape of the curve does not touch the control points anymore, but is moved towards to other control points a bit. Nevertheless this property can be useful for the cases where the power of the disturbing noise is high.

Of course, it is possible to use higher-order B-splines for interpolation. These are called (rational) regression B-splines or Penalized B-splines (P-spline) [109], [116], where – typically – the positions of the control points are modified. This means that the control points can be defined such that the spline crosses (a subset of) the function’s points to be smoothed. Figure 8.4 shows such an example: an impulse response function is interpolated. The control points are a subset of impulse response’s points.

Depending on the sampling frequency (increasing sampling frequency results in decreasing relative number of control points required) and the system dynamics, not all samples are needed to describe a function. In this example each tenth function point is used as a control point (and consequently the corresponding knots). This number (ten) is an important parameter: the step size. This will be referred to later on in Chapter 10.

A brief overview about the P-splines and the LS fitting can be found in Appendix B.6.

In the next section the two dimensional B-spline form is defined.
Figure 8.3: An example for the one dimensional B-spline used for approximate interpolation. The left figures show the basis functions of different orders. The middle figures show the basis functions together with the control points. On the right the resulting interpolations are shown. The top row belongs to zero degree, the second row belongs to the first degree, the third row shows the second degree case and the bottom row is the third degree. The crosses (×) and asterisks (*) are the knots, the circles (○) are the control points. Observe that in this case (2nd or higher order B-splines) the approximation does not go through the control points. It would be possible to achieve this, but for this, for instance, solution of a LS fitting algorithm is necessary.
Chapter 8 - Two dimensional B-splines

8.2 Two dimensional B-splines

In this thesis one and two dimensional B-splines are studied. Note that B-splines can be extended to any dimension. In this section the two dimensional B-splines are discussed. They will play an important role in the proposed B-spline based time-varying method.

**Definition 8.3** A two dimensional B-spline smoothing function \( s(x, y) \) is a linear combination of basis functions \( B_{i}^{d_{x}}(x) \), \( B_{j}^{d_{y}}(y) \) weighted by the coefficients \( c_{i,j} \) (the ordinates of the control points \( P_{i,j}(x_{i}, y_{j}) \) evaluated on the axis \( x \) with degree of \( d_{x} \) and on the axis \( y \) with the degree \( d_{y} \) at point \( i, j \) (corresponding to the coordinates on the \( x \) and \( y \) axis):

\[
s(x, y) = \sum_{i=0}^{n_{x}-1} \sum_{j=0}^{n_{y}-1} c_{i,j} B_{i}^{d_{x}}(x)B_{j}^{d_{y}}(y)
\]

where \( n_{x} \) and \( n_{y} \) represent the number of control points used for the interpolation in \( x \) and \( y \) axis, respectively.

In the identification case, the control points can be for instance the subset of the points of the impulse response function (in the time domain) to be approximated. Next, the knots are defined.

\( x \) represents a monotonically increasing sequence of knots on the axis \( x \), namely:

\[
x = [x_{0} \leq x_{1} \leq \cdots \leq x_{n_{x}-1} \leq \cdots \leq x_{k-1}]
\]
where $k$ means the total number of knots on the axis $x$.

$y$ represents a monotonically increasing sequence of knots on the axis $y$, namely:

$$y = [y_0 \leq y_1 \leq \cdots \leq y_{ny-1} \leq \cdots \leq y_{l-1}]$$ (8.9)

where $l$ means the total number of knots on the axis $y$.

In order to define a B-spline polynomial it is required to have more knots (on the abscissa) than control points. If the degrees are $d_x, d_y$ and the number of coefficients $c_{i,j}$ is $n_x, n_y$ then the minimal number of knots is set by:

$$k_{\text{min}} = d_x + n_x + 1$$ (8.10)$$l_{\text{min}} = d_y + n_y + 1$$ (8.11)

where $k \geq k_{\text{min}}, l \geq l_{\text{min}}; k_{\text{min}}, k, n_x, l_{\text{min}}, l, n_y \in \mathbb{N}^+$ and $d_x, d_y \in \mathbb{N}^0$.

In the two-dimensional case $C^{d_x−1,d_y−1}$ continuity means that all the partial derivatives $\frac{\partial^{i+j} s(x,y)}{\partial x^i \partial y^j}$ exist for all $0 \leq i < d_x - 1, 0 \leq j < d_y - 1$ and they are continuous in the domain of interest.
Chapter 9
Time domain nonparametric estimate of linear time-varying systems using B-splines

In the previous chapter a two dimensional regularization technique was presented to estimate a linear time-varying system. In that technique the two dimensional LTV impulse response function is directly estimated with a high number of parameters.

In this proposed method two dimensional B-splines are used to describe the underlying LTV system by using a B-spline kernel interpretation. This means that the two dimensional impulse response function is replaced by a spline based kernel and hence the LTV IRF is indirectly estimated. The main advantage of this methodology lies in the fact that the estimation requires a fewer number of parameters, and it has smaller computational load compared to the regularization technique. Practically it means that a generalized B-spline technique is applied as a double smoothing algorithm: once over the system time $\tau$ which is the direction of the impulse responses – referring to the system behavior – and once over the global time $t$ – referring to the system memory.

If the observed LTV system is smooth enough, with respect to the system dynamics, it will be possible to reduce the disturbing noise by additional smoothing. This reduces also the number of model parameters that need to be stored, and it decreases the effect of the undesired transient term.
9.1 The model

The key idea for constructing the model remains the same: the classical linear time-invariant impulse response baseline model can be extended leading to a two dimensional form. This two dimensional form will be rewritten with B-splines. The assumptions given in the regularization part are imposed here as well. For the convenience of the reader, they are repeated shortly:

- **Assumption 2.2**: the smoothness assumption imposed on an LTV system,
- **Assumption 6.1**: the length of the longest considered IRF is $L$,
- **Assumption 6.2**: finite length of the observation window is $N$,
- **Assumption 6.3**: the measurement noise is $e[t]$, $e \sim \mathcal{N}(0, \sigma^2)$,
- **Assumption 6.4**: the underlying system is causal and stable,
- **Assumption 6.5**: the excitation signal $u[t]$ is known exactly,
- **Assumption 6.6**: independent excitation signal $u[t]$ and observation noise $e[t]$.

Due to these assumptions, the baseline model given by Eq. (6.1)–(6.2) can be recalled:

$$y_m[t] = y[t] + e[t] = \sum_{\tau=0}^{\tau=\infty} h[t, \tau]u[t - \tau] + e[t]$$

where $t = 0,1,2 \ldots N - L$.

The measurement setup is shown in Figure 6.2.

The B-spline based smooth LTV model of the two dimensional impulse response function of an LTV using **Assumption 2.2**, Eq. (9.1) is:

$$h[t, \tau] \approx \tilde{h}_{LTV}[t, \tau] = \sum_{i=0}^{n_t-1} \sum_{j=0}^{n_\tau-1} c_{i,j} B_i^{d_t}(t) B_j^{d_\tau}(\tau)$$

where $n_t$ and $n_\tau$ are the number of knots, and $d_t$, $d_\tau$ are the degrees in global and system time directions, respectively.

To distinguish B-splines from the regularized estimate, the estimated variables wear a tilde (~) instead of a hat (^) symbol. $\tilde{h}_{LTV}[t, \tau]$ is a smooth representation of a LTV system. The B-spline LTV kernel is represented by the control points in Eq. (9.2). In this present approach, the main information is hidden in the ordinates ($c_{i,j}$) of the B-spline control points: the information related to the amplitude characteristics of the IRFs. The abscissas of the control points play here a secondary role: the only restriction that they should be (equidistantly) spread over the whole (double) time scale.
Next, put the above defined response function in the convolution product of the LTV system’s model defined in Eq. (9.1):

\[ y(t) = \sum_{\tau=0}^{\tau_{t}} \sum_{\tau=0}^{\tau_{t}} c_{ij} B_{i}^{d_{t}}(t) B_{j}^{d_{r}}(\tau) u(t - \tau) \]  

(9.3)

Lastly, make equivalent reordering in order to separate the terms depending on \( t \) and terms depending on \( \tau \):

\[ y(t) = \sum_{i=0}^{n_{t}-1} \sum_{j=0}^{n_{r}-1} c_{ij} B_{i}^{d_{t}}(t) B_{j}^{d_{r}}(\tau) u(t - \tau) = \]

(9.4)

This leads to a form which can be used for identification purposes. In order to have a better overview about the formulas, Eq. (9.4) can be rewritten in a matrix form as follows:

\[ y = K_{LTV,BS} C_{LTV,BS} \]  

(9.5)

where the \( N \times 1 \) output vector (the modeled output of the LTV system in the time domain) is:

\[ y = [y[0] \ldots y[N-1]]^{T} \]  

(9.6)

and the unknown coefficients of the impulse response functions are in a \( (n_{t} \cdot n_{r}) \times 1 \) vector:

\[ C_{LTV,BS} = [c_{1,1} \ldots c_{1,n_{r}} c_{2,1} \ldots c_{n_{t},n_{r}}]^{T} \]  

(9.7)

and the \( N \times (n_{t} \cdot n_{r}) \) observation matrix is:

\[ K_{LTV,BS}(t, p) = B_{i}^{d_{t}}(t) (\sum_{\tau=0}^{\tau_{t}} B_{j}^{d_{r}}(\tau) u(t - \tau) ) \]  

(9.8)

The index \( p \) in the matrix \( K_{LTV,BS}(t, p) \) consists of the multiplication of increased variables of \( i \) and \( j \) and corrected with \(-1\) i.e.:

\[ p = (i + 1)(j + 1) - 1 \]  

(9.9)
These increased variables are to avoid a lot of zero terms in the index \( p \), e.g. if \( i \) zero and \( j \) is \( 0 \ldots n_L - 1 \) then all the indices would be zero for this given \( i \). The correction with \(-1\) is to have the starting value of indexing at zero.

### 9.2 The choice of the cost function

It is very important to remark that in this particular identification case the B-spline fitting (in the LTV kernel representation) differs from the classical B-spline control point fitting used in computer graphics. In computer graphics (typically) we have certain points (pixels) that are needed to be interpolated somehow. In this work, we have to create the control points ourselves (without knowing in advance the exact number and the positions of the control points) such that the system dynamics are well described. Therefore it is important to define the cost function of the proposed method.

#### 9.2.1 The cost function

The choice of the cost function is crucial. In this particular case, it seems to be handy to use a Maximum Likelihood (ML) [37] cost function.

Because the additive noise is zero mean i.i.d. Gaussian, it leads to the well-known LS problem with the cost function shown already in Eq. (6.6):

\[
V_{\text{LTV,LS}} = \| y_m - K_{\text{LTV}} h_{\text{NLX}1} \|_2^2 = V_{\text{LTV,LS}} = \| y_m - \tilde{y} \|_2^2 
\]  

(9.10)

This can be rewritten with the smooth estimate (\( \tilde{y} \)) of the output observation and with the B-spline kernel \( C_{\text{LTV,BS}} \) together with the B-spline observation matrix, so the cost function is given by:

\[
V_{\text{LTV,BS}} = \| y_m - \tilde{y} \|_2^2 = \| y_m - K_{\text{LTV,BS}} C_{\text{LTV,BS}} \|_2^2 
\]  

(9.11)

This is a linear least squares problem that can be solved analytically and this way the B-spline coefficients can be computed as follows (computation must be done in a numerically stable way, for instance via singular value decomposition (SVD)) (\(\dagger\) denotes the pseudo-inverse):

\[
\tilde{C}_{\text{LTV,BS}} = K_{\text{LTV,BS}}^T y_m = \left( K_{\text{LTV,BS}}^T K_{\text{LTV,BS}} \right)^{-1} K_{\text{LTV,BS}}^T y_m 
\]  

(9.12)

provided that the inverse of \( K_{\text{LTV,BS}}^T K_{\text{LTV,BS}} \) exists (a proper excitation signal is used).
9.2.2 Observations

Considering Eq. (9.5) the following statements can be formed:

The modeled output ($\tilde{y}$) consists of a product of two terms, namely, $K_{LTV, BS}$ and $\tilde{C}_{LTV, BS}$. The parameters which describe the system can be found in $\tilde{C}_{LTV, BS}$. This is the LTV B-spline kernel. After computing this term with the help of Eq. (9.12), the system output can be determined by the input signal (represented in $K_{LTV, BS}$) and the coefficient matrix $\tilde{C}_{LTV, BS}$ (representing the smoothed impulse responses).

In this particular case, the two dimensional impulse response has been replaced to another presentation, where B-spline kernels ($\tilde{C}_{LTV, BS}$) are considered, but all the LTV IRF properties are inherited.

It means that the B-spline based two dimensional impulse response function model has the same properties as those defined and assumed previously for the considered LTV system: smoothness, linearity, causality and stability properties.

9.3 Tuning of the model complexity

The model order selection is a very important question. It is difficult to have a good balance between the model variability and its systematic error. When the model order is increased, in general, the error is decreased for a given (input/output) dataset.

If the model order is too large ($L, n_t, n_\tau$ are large), then the model will include the noise as well, which is undesired. This is not desired because it can lead to a wrong fit when a different excitation signal (for instance validation signal) is used for computing the model’s output.

In this context the model order consists of two different components. The first one is the component of the impulse response direction (the length of the longest considered IRF $L$, the number of control points $n_\tau$ and the degree of B-splines $d_\tau$ used for the system time direction). The second one is the component of the global time which represents the time variations (the number of control points $n_t$ and the degree of B-splines $d_t$ used for the global time direction).

To determine the right model order, several criteria can be used. One possibility is described here. The proposal is to use a special cross-validation technique.

In order to obtain a good model order, two steps are needed with different (input/output) datasets. The input signals should be different, persistent excitations. In this proposed situation, the input signals are random phase multisines (see Chapter 4.3).
The first step is to obtain an estimation set using the estimation excitation signal \( u_{est}[t] \) and the observation output \( y_{est}[t] \), where the B-spline based two dimensional impulse response vector \( \tilde{c}_{LTV,BS|u_{est,BS}} \) (see Eq. (9.12) is determined

\[
\tilde{c}_{LTV,BS|u_{est,BS}} = \left( K_{LTV,BS|u_{est,BS}}^T K_{LTV,BS|u_{est,BS}} \right)^{-1} K_{LTV,BS|u_{est,BS}}^T y_{est}
\]

(9.13)

where \( p_{BS} \) represents the hyperparameters \( L, n_t, n_r, d_t, d_r \). \( K_{LTV,BS|u_{est,BS}} \) is an observation matrix shown in Eq. (9.8), created by using \( u_{est}[t] \) and \( p_{BS} \).

In the second step, the validation dataset \( (u_{val}[t], y_{val}[t]) \) is used to tune the model complexity. Using the previously computed \( \tilde{c}_{LTV,BS|u_{est,BS}} \) and an independent validation input \( u_{val}[t] \), the model’s output \( (\tilde{y}_{val|u_{est,BS}}) \) can be simulated via Eq. (20) as follows:

\[
\tilde{y}_{val|u_{est,BS}} = K_{LTV,BS|u_{val,BS}} \tilde{c}_{LTV,BS|u_{est,BS}} \]

(9.14)

where \( K_{LTV,BS|u_{val,BS}} \) is an observation matrix created by using \( u_{val}[t] \) and \( p_{BS} \).

To do these steps, a simple (normalized mean square) cost function is defined as follows, using the terms from Section 3:

\[
V_{validation}(u_{val}, p_{BS}) = \frac{\sum_{t=0}^{N-1} (y_m[t] - \tilde{y}_{val|u_{est,BS}}[t])^2}{N}
\]

(9.15)

When the dominating error is due to the noise, the Eq. (9.15) should provide more or less the noise variance. It means that a good model order is found when these cost functions both give a value around the noise variance \( (\sigma^2) \).

This can be achieved when the expected value is computed as follows –by a given \( \sigma^2 \):

\[
E\{V_{estimation}(u_{val}, p_{BS})\} = \frac{\sum_{t=0}^{N-1} (y_m[t] - \tilde{y}_{val|u_{est,BS}}[t])^2}{N}
\]

\[
= E\left\{\left(\frac{1}{N}\right) \cdot \frac{\sum_{t=0}^{N-1} (y_m[t] - \tilde{y}_{val|u_{est,BS}}[t])^2}{N} \right\} = \left(\frac{1}{N}\right) \cdot E\left\{\sum_{t=0}^{N-1} \sigma^2_t \right\} \approx \sigma^2
\]

(9.16)
If, for different combinations of model orders, a validation criterion is close to the variance level, the simplest model structure (the smallest orders) is recommended to be used.

When working with time-varying systems, it is often very difficult – or impossible – to repeat the experiments in such a way that the time variations are equal.

To avoid that problem, we construct the estimation and validation data sets from the data of a single experiment. This technique is explained in Chapter 7.1 and illustrated in Figure 7.1.

9.4 An example

9.4.1 Simulation Setup

The aim of this section is to present a simulation example for the selection of model orders.

A discrete, slowly changing first-order (exponential) system is observed (see Figure 9.1). The observation comprises N=2000 samples. The excitation signal is a full random-phase multisine (see Section 5.2) with an rms value of 1. The assumed length of the longest impulse response is 150 samples.

The system is given by the following equation:

$$h_{LTV}[t, \tau] = e^{\frac{-t}{a[t, \tau]}}$$ (9.17)

The observation can be modeled as follows:

$$y[t] = \sum_{\tau=0}^{L-1} h_{LTV}[t, \tau] u_{ms}[t - \tau] + y_{noise}[t] = \sum_{\tau=0}^{L-1} e^{\frac{-t}{a[t, \tau]}} u_{ms}[t - \tau] + y_{noise}[t]$$ (9.18)

where the noise $y_{noise} \sim N(0, 0.2^2)$ is Gaussian distributed further rms($u_{ms}$)=1 and $a[t, \tau] = 0.001 \cdot \tau(t + 1)$. The SNR is 20 dB.

As a reminder, in this work the time variable $t$ represents the global time and the time variable $\tau$ represents the system time (the direction of the impulse responses).
9.4.2 Results of the estimation

To simplify things, second-order B-spline basis functions are considered. As a first step, the model structure (in this case the number of knots for a given degree) will be chosen using the previously defined validation cost function normalized by the level of the noise variance.

Figure 9.2 shows some realizations of the validation and estimation cost functions at given parameters $n_t$ as a function of $n_r$.

Figure 9.2: The second order B-spline based cost functions are shown as a function of system time parameter $n_t$ at the three different $n_r$ global time parameters ($n_r = 3, 5, 7$). The estimation cost function is in thick solid line, the validation cost function is in thin solid line. The straight dashed line is the constant one (the expected value of the cost functions). The decreasing dashed line shows the theoretical estimation cost function decreasing with $\frac{n_t n_r}{N}$.

In Figure 9.3 the contour plot of the validation error cost function can be seen. Observe that the validation cost function is fairly insensitive to $n_t$. This lies in the fact that $n_t$ represents the time variations (global time direction). The number of necessary parameters in $n_t$, depends on the smoothness of the time variations: the smoother the system is, the smaller the $n_t$. 

Figure 9.1: The observed exponential noiseless slowly varying system to be estimated.
From the many possible solutions, the smallest local minimum has been chosen resulting in the model orders: $n_r = 33$ and $n_t = 5$. Using the coefficients of the smoothed impulse response function a new simulation is made but in this case without noise. To illustrate the quality of the model, the output – generated by an independent excitation signal – together with the simulation error is shown in Figure 9.4. As it can be seen the fitting error is relatively low, although the impulse response function is obtained by using extremely noisy data.
9.5 Convert the B-spline LTV kernel to the two dimensional impulse response form

9.5.1 The conversion

In some cases it might be useful to use the two dimensional impulse response form or to have direct access to the impulse responses instead of using the B-spline LTV kernel form. In order to have a better understanding about the B-spline LTV kernel, the structure of $C_{LTV,BS}$ has to be analyzed. First, let us reshape $C_{LTV,BS}$ into the matrix $C_{n_t \times n_t}$ such that:

$$
C_{n_t \times n_t} = \begin{bmatrix}
       c_{0,0} & \cdots & c_{n_t-1,1} \\
       \vdots & \ddots & \vdots \\
       c_{0,n_{tau}-1} & \cdots & c_{n_t-1,n_{tau}-1}
\end{bmatrix}
$$

(9.19)

Observe that every column in $C_{n_t \times n_t}$ represents the B-spline impulse response coefficients at a certain time. The different columns are connected to each other with B-spline smoothing functions (so the B-spline IRFs are linked to each other). An illustration is shown in Figure 9.5.

![Figure 9.5: An example of the LTV B-spline kernel.](image)

As can be seen, this form is not compatible with the form of $H_{L \times N}$ or $h_{NL \times 1}$ which is used in the regularization part. It means:

$$
C_{n_t \times n_t} \not\in H_{L \times N}
$$

$$
C_{LTV,BS} \not\in h_{NL \times 1}
$$

(9.20)
When necessary, \( C_{n_t \times n_t} \) can be converted to the form \( H_{L \times N} \) (or to \( h_{N \times L} \)) by using the Eq. (9.2). An alternative but more spectacular method is given as follows:

- Create an excitation signal from a vectorized identity matrix of size \( N \times N \). This will result in several excitation signals consisting of time shifted Kronecker-delta inputs, therefore the output to these signals will provide the impulse responses.
- Calculate the output. They will provide directly the elements of \( \tilde{h}_{N \times L} \).
- By reshaping the outputs, the \( \tilde{H}_{N \times L} \) matrix form can be obtained.

The method is illustrated in Figure 9.6.

![Figure 9.6: A methodology to convert the LTV B-spline kernel to the two dimensional LTV IRF form.](image)

### 9.5.2 A simulation example

In this section a simulation example is shown. The underlying system is already defined in Chapter 6.6. The simulation parameters are: \( L=30 \), \( N = 150 \) and \( SNR = 35 \text{ dB} \).

The parameters of the B-splines are obtained by using the tuning method given in Section 9.3. The hyperparameters \( (p_{BS}) \) are \( n_t = 28 \), \( n_r = 4 \), \( d_t = 3 \), \( d_r = 3 \).

The LTV B-spline kernel belonging to this experiment is shown in Figure 9.5. In Figure 9.7, on the top the true underlying system is shown. The second row shows the B-spline estimation, when the conversion algorithm is used. The fitting errors are shown in Table 9.1.

Observe that the estimation quality is close to the flexible regularization method. The difference lies in the fact that the regularization technique uses more prior information about the system, namely the exponential decaying.

The advantage of B-spline approach is that it is very parsimonious and computational needs are lower.

The personal computer configuration used for the simulation can be found in Appendix A.6.
Table 9.1: Comparison of the different estimation methods used for an experiment.

<table>
<thead>
<tr>
<th>Approach</th>
<th>relative rms error [%]</th>
<th>memory need [kB]</th>
<th>absolute time [sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTI LS</td>
<td>33.78</td>
<td>0.12</td>
<td>0.02</td>
</tr>
<tr>
<td>TC flexible</td>
<td>4.55</td>
<td>83718</td>
<td>1836</td>
</tr>
<tr>
<td>DC flexible</td>
<td>4.23</td>
<td>83718</td>
<td>2313</td>
</tr>
<tr>
<td>B-spline</td>
<td>5.16</td>
<td>5</td>
<td>875</td>
</tr>
</tbody>
</table>

Figure 9.7: The B-spline estimate of a linear time-varying system is shown in presence of noise. The top figure shows the true underlying system. The second row shows the estimate of the underlying system.
9.6 Transient elimination

9.6.1 The proposed transient elimination technique

The basic idea comes from the regularized transient elimination technique (see Section 0). In order to be able to eliminate the transient effect, the assumptions made for the previous regularization based elimination technique are needed to be imposed here as well. For the convenience of the reader, they are repeated shortly:

- **ASSUMPTION 7.1**: the excitation signal is proper (persistent) and the past values of the excitation are independent of the future values (i.e. the excitation is white).
- **ASSUMPTION 7.2**: the observable transient response function $h_{\text{transient}}$ has a finite order of $N_{\text{transient}}$.
- **ASSUMPTION 7.3**: the transient term is independent of the observation noise.

The most important consequence of these assumptions is that dependency between the measured output and the transient term (see for derivation Eq. (7.3)) is given by:

$$\text{COV}\{y_m, y_{\text{transient}}\} = \mathbb{E}\{y_{\text{transient}}y_{\text{transient}}^T\} \quad (9.21)$$

As can be seen, the transient term is independent of the noise and the true output. This will allow us to handle the LTV B-spline kernel and the transient term independently. If **ASSUMPTION 7.1** and **ASSUMPTION 7.2** are satisfied, then we can write the new baseline model in B-spline form as follows:

$$y_m[t] = y[t] + y_{\text{transient}}[t] + e[t] =$$
$$\sum_{\tau=0}^{L-1} h[t, \tau]u[t - \tau] + \sum_{\tau=0}^{N_{\text{transient}}-1} h_{\text{transient}}[t] + e[t] =$$
$$\sum_{i=0}^{n_{\text{transient}}-1} \sum_{j=0}^{n_{t}-1} c_{ij}B_{i}^{d_{t}}(t) \{\sum_{\tau=0}^{L-1}[B_{j}^{d_{t}}(\tau)u[t - \tau]\} +$$
$$+ \sum_{j=0}^{n_{\text{transient}}-1} B_{j}^{d_{t}}(t) c_{\text{transient}j} + e[t] \quad (9.22)$$

where $n_{\text{transient}} \leq N_{\text{transient}}$. This can be rewritten in vectorial form:

$$y_m = y + y_{\text{transient}} + e[t] =$$
$$K_{\text{LTV,BS}}C_{\text{LTV,BS}} + h_{\text{transient}}[t] + e[t] =$$
$$K_{\text{LTV,BS}}C_{\text{LTV,BS}} + K_{\delta,BS}C_{\text{transient,BS}} + e[t] \quad (9.23)$$

where $K_{\delta,BS}$ is an $N_{\text{transient}} \times n_{\text{transient}}$ matrix, $K_{\delta,BS} = \sum_{j=0}^{n_{\text{transient}}-1} B_{j}^{d_{t}}(t)$, and $n_{\text{transient}} \times 1$ vector $C_{\text{transient,BS}}$ is $C_{\text{transient,BS}} = [c_{\text{transient}0}, \ldots, c_{\text{transient}n_{t}-1}]$. 
Using this information, the proposed algorithm follows.

First, the extended B-spline kernel $C_{BS}'$ has to be defined. This is $(n_t + n_{\text{transient}}) \times 1$ vector which is the joint B-spline kernel consisting of the time-varying and transient kernels. This vector is defined as follows:

$$C_{BS}' = [C_{LTV,BS}' \ C_{\text{transient,BS}'}]^T$$  \hspace{1cm} (9.24)

The joint observation hypermatrix is given by:

$$K_{BS}' = [K_{LTV,BS} \ K_\delta,BS]$$  \hspace{1cm} (9.25)

The above mentioned terms allow us to redefine the cost function of the problem:

$$V_{BS}' = \|y_m - \tilde{y}\|_2^2 = \|y_m - K_{BS}' C_{BS}'\|_2^2$$  \hspace{1cm} (9.26)

and from here, the solution of $C_{BS}'$ is given by:

$$C_{BS}' = (K_{BS}'^T K_{BS}' - 1) K_{BS}'^T y_m$$  \hspace{1cm} (9.27)

The B-spline transient elimination algorithm is the following:

i. estimate $p_{BS}$ set of parameters of $C_{LTV,BS}$.

ii. estimate $C_{BS}'$, obtain the parameters of the transient kernel,

iii. if necessary (i.e. the accuracy is low) re-estimate $C_{BS}'$ using the previous $p_{BS}$ as initial values and by removing the previously estimated transient from the estimation dataset.

### 9.6.2 An example

In this section a simulation example is shown. The underlying system is defined in Chapter 6.6. For illustration purposes, in order to disturb the system, a transient signal is manually injected. The length of the transient ($N_{\text{transient}}$) is 20. The transient and its estimate are shown in Figure 9.8. The rest of the simulation parameters are untouched (multisine excitation, $L = 30, N = 150, SNR = 35 \text{ dB}$).

Figure 9.9 shows the performed test for two cases: first, when no transient elimination technique is used and second, when the transient elimination algorithm is used.
Chapter 9- Transient elimination

The fitting errors are shown in Table 7.1. In order to have a complete picture, the results are compared with the robust DC approach. Observe that even in presence of transient, the proposed B-spline method without the transient eliminating algorithm delivers a good result comparable to the DC robust method, where transient is eliminated. This beneficial property lies in the fact that B-splines are very resistive against the disturbing noise (see Appendix B.6) and due to the short transient and the double smoothing in the triple summation in Eq. (9.3), the effect of the disturbing transient is “averaged” out.

![Figure 9.8: The injected transient is shown in black. The estimated transient is shown in grey.](image)

Table 9.2: Comparison of the DC robust method used for an experiment with a transient.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Relative rms error [%]</th>
<th>Absolute time [sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC robust, with the transient model</td>
<td>11.35</td>
<td>69203</td>
</tr>
<tr>
<td>DC robust, without the transient model</td>
<td>38.43</td>
<td>69170</td>
</tr>
<tr>
<td>B-spline, with the transient model</td>
<td>8.38</td>
<td>879</td>
</tr>
<tr>
<td>B-spline, without the transient model</td>
<td>18.46</td>
<td>875</td>
</tr>
</tbody>
</table>
Figure 9.9: The B-spline estimates of two dimensional IRF shown in presence of a transient. The top figure shows the true underlying system. The second and third rows show the estimates of the underlying system. (Keep in mind that there is no unique model.)
9.7 Conclusions

In this chapter a method is presented to estimate a nonparametric B-spline based two dimensional impulse response function. The coefficients of the two dimensional B-splines are the LTV B-spline kernels.

The identification procedure is straightforward and consists of three simple steps:

- prepare the data sets,
- estimate the model order, and
- compute the coefficients of the two dimensional impulse response function.

The proposed technique is very useful because:

- the baseline model used in this section is a natural extension of the (one dimensional) linear time invariant system’s theory,
- smoothing constraints are handled with the flexible B-splines,
- unlike the frozen impulse response approaches, here smoothed non-frozen impulse responses are considered (no overlapping effect, see next chapter),
- insensitive to disturbing noise and short transients,
- memory requirements and computational loads are smaller than in the case of the regularization technique.

Using the proposed method a good model quality can be achieved, but the fitting quality is under the proposed regularization methods.

A comparison between the non-frozen B-splines and the two dimensional regularization is given in Section 11.3 (experimental comparison based on the measurement of an LTV system) and in Section 12.2.2.4 (overall conclusions of this thesis).
Chapter 10

Frozen nonparametric estimate of linear time-varying systems using B-splines

When the observed system is slowly changing or in other words, the time variations are very smooth, then it is possible to describe the system as a series of linear time-invariant systems.

In these techniques, at each measurement time an LTI model is built. Because the system dynamics are considered to be frozen in the observation window, these LTI models are called frozen. In this chapter a time and a frequency domain nonparametric technique will be presented. In the proposed method it requires that during a short observation time, when a model is built, the system needs to be well approximated by a time independent model.

Most work found in the literature (e.g. [56], [123], [124], [125]) chooses the time-variation to be modeled non-parametrically in the frequency domain, therefore this chapter starts with a brief overview about the state-of-the-art technique. The present chapter does not aim to study the generalized concept of a transfer function (known as system function, instantaneous transfer function as well). It rather focuses on constructing simple estimation methodologies for slowly time-varying systems to approximate the system function (instantaneous transfer function).

If the parameter changing of the observed system is sufficiently slow, with respect to the system dynamics, we will be able to reduce the disturbing noise by additional smoothing and to reduce the number of model parameters that need to be stored.
Chapter 10- A spectral description of linear time-varying systems

10.1 A spectral description of linear time-varying systems

In the case of linear time-varying systems, the classical frequency transfer function does not exist because the response of an LTV system to a sinusoidal excitation is not sinusoidal anymore. The frequency response of an LTV system to a sine excitation can be written as:

\[
G_{LTV}(u[t]) = G_{LTV}(\sin[\omega_m t]) = |H_{LTV}[t, m]| \cdot \sin[\omega_m t + \angle (H_{LTV}[t, m])]
\] (10.1)

where \( m \) is the frequency index (see DEFINITION 3.3), \( \angle \) refers to the phase and \( H_{LTV}[t, m] \) is defined by DEFINITION 2.19. This means that the amplitude and the phase are modulated.

It is shown in the literature that in the case of slowly varying systems, the time-varying transfer function can be generalized and approximated by a special (frozen) transfer function (for instance in [40], [41], [44], [48], [49]). In this chapter a special (frozen) frequency and time domain technique will be provided.

Because in the present chapter only slowly time-varying systems are considered, an exact definition to the notion “slowly varying” has to be given.

**DEFINITION 10.1** A linear time-varying system is slowly varying, if the system parameters remain almost constant over the duration of the impulse response function i.e. the relative root mean square difference between the impulse response at the beginning and at the end of the observation window is less than approximately 5 (%). This means that at a certain time instant the LTV system can be well approximated by an LTI system by freezing the parameters in the observation window.

Next, a typical model structure of a frozen transfer function of a slowly time-varying system will be given as parallel connection of LTI systems.

**DEFINITION 10.2** When a discrete linear slowly time-varying system (LsTV) is considered in the frequency domain (see DEFINITION 2.19–DEFINITION 2.20) then it can be approximated by a frozen transfer function \( G_{FTF}[t, m] \) as a parallel connection of several time-weighted frozen LTI transfer function \( G_i[m] \) (see DEFINITION 2.21), such that:

\[
G_{FTF}[t, m] = b_0[t]G_0[m] + \ldots + b_{N_{LTI}-1}[t]G_{N_{LTI}-1}[m] = \sum_{i=0}^{N_{LTI}-1} b_i[t]G_i[m]
\] (10.2)

where \( N_{LTI} \leq N, N_{LTI} \in \mathbb{N}^+ \). \( G_i[m] \) is an LTI transfer function. \( N_{LTI} \) refers to the number of LTI models used to describe the LTV system, and \( b_i \in \mathbb{R} \).
In order to simplify, the time index $t$ and frequency index $m$ in $G_{FTF}[t,m]$ and $G_i[m]$ will be eliminated from now on.

There are two very important assumptions that we have to impose on these LTI models:

**Assumption 10.1** Every $G_i$ block in $G_{FTF}$ is a stable LTI model.

**Assumption 10.2** Every $G_i$ block in $G_{FTF}$ is a causal LTI model.

When $b_0$ is a constant with the value one (i.e. $b_0[t] = 1$), and the other parameters are zero mean, then in this branch the LTI model $G_0$ boils down to the BLTI (best linear time-invariant) model. The concept of the BLTI approximation is very important because it tells us what we would obtain, if we would simply use the LTI framework and neglect the time-variations.

At this point of the thesis, it is very important to refer to [40] and [41]. In these works the remaining $b_1 ... b_{N_{LTI}-1}$ parameters are zero mean basis functions (for instance Legendre polynomials [126]) and the corresponding branches can describe the time-varying behavior. This model structure is shown in Figure 10.1.

![Figure 10.1: Equivalent block schematic of the frozen transfer function model based on the works [40], [41].](image)

In these works, it is shown that the spectral output of an LsTV system can be approximated by this model under very similar assumptions as those are imposed in this thesis [40]. The proposed method together with the basis function selection is detailed in Section 10.2.

To show the importance of this concept, let us observe the system given by its instantaneous transfer function (see **Definition 2.19–Definition 2.20**) as follows:

$$H(t, s) = \frac{(t + 1)(0.01)s + 1}{(5 + t)0.01s^2 + 1s + (t + 1)0.01}$$  \hspace{1cm} (10.3)

where $t = 0 ... N - 1$, and $N = 150$. 

117
The excitation signal is a full band random phase multisine (see Definition 4.5). The input-output spectra are shown in Figure 10.2. The observations are shown in the frequency domain without disturbing noise and transient terms.

Figure 10.2: The figure on the left shows the magnitude spectrum of the excitation signal. The figure on the right shows the output magnitude spectrum of an LTV system.

Surprisingly, the output measurement looks very noisy with respect to the fact that the underlying system is smooth and we know for sure that there is no measurement noise or transient present (since it is a simulation). Let us make a second simulation. The excitation signal is now an odd random phase multisine, where only every tenth harmonic is excited. The simulation is shown in Figure 10.3.

Figure 10.3: The figure on the left shows the magnitude spectrum of the excitation signal. The figure on the right shows the observed output magnitude spectrum of an LTV system. The figure on the right is shown separately in Figure 10.4.

Observe that around the excited frequencies skirt-like shapes appear, which could be interpreted as a spectral leakage. Since the simulation is ideal – a full period of the excitation signal is taken and the input/output samples are perfectly synchronized – we know that this is not due to usual spectral leakage. The above-referred works can describe this skirt-like behavior very well.

Because the input excitation is a multisine, it is possible to compute the frequency response function (and the output spectrum) via a simple discrete Fourier transformation (see Definition 3.4) on the excited frequency bins. Figure 10.4 shows the output magnitude spectrum of the simulation. Observe that by interpolating the
magnitudes at excited frequency harmonics, we obtain the BLTI estimate as can be seen in Figure 10.4 in the thick solid line (more precisely, this figure shows the output of the BLTI estimate for the given excitation signal). This is the first branch of the FTF model shown in Figure 10.1. The remaining parallel branches cause the time-variations shown with the crosses in Figure 10.4.

Figure 10.4: The output magnitude spectrum of an LTV system is shown. The thick solid line shows the output of BLTI model. The crosses (×) show the time-variations. The thick plus (+) signs show the output magnitude at the excited frequency lines.

Figure 10.4 leads to the idea of the detection of time variations. In the case of an LTV system, the skirt appearance indicates the time varying behavior under the assumptions that

i. there is no spectral leakage,

ii. a multisine excitation is used with skipped harmonics,

iii. the LTV system can be described by Eq. (10.2),

A detailed description of possible detection methods of time variations are provided in [127], [128], [129].

10.2 The proposed frozen methods

In the proposed frozen approach the viewpoint differs from the above-mentioned concept: the two dimensional impulse response function or the instantaneous transfer function is approximated by a special frozen impulse response function or frozen transfer function. They consist of LTI frozen IRFs or FRFs obtained in a sliding window at certain time instants.

It is important to emphasize that the proposed frozen methods in this chapter do not aim to beat the performance of the regularization or the non-frozen B-spline approaches presented in this thesis. The goal is to provide a simple and reasonably accurate method to estimate systems with slow time-variations.
The assumptions given in the previous chapters (ASSUMPTION 6.1 – ASSUMPTION 6.6) are imposed here as well. In addition to that, the ASSUMPTION 2.1 has to be used as well. For the convenience of the reader, this assumption can be reformulated by B-splines in the frequency domain as follows:

ASSUMPTION 10.3 A discrete LTI system is smooth, if the frequency response function can be described with a limited number of B-spline polynomial terms and the spectral content of the underlying LTI system is highly concentrated at the low frequencies.

In the proposed methods, the number of LTI branches in Eq. (10.2) equals the number of samples, i.e. \( N_{LTI} = N \). The terms \( b_i[t] \) in Eq. (10.2) are defined as follows:

\[
 b_i[t] = \begin{cases} 
 1, & i = t \\ 
 0, & i \neq t 
\end{cases} 
\]  

(10.4)

It means that we have \( N \) frozen impulse or frequency responses, each of them belong to a certain time. The attribute “frozen” refers to the fact that each \( G_i \) describes an impulse or a frequency response function, where the system dynamic is to be frozen at a certain time instant.

This will result in a (very) large number of IRFs or FRFs, which are closely related to each other. As a post processing step, a special two dimensional B-spline algorithm will be used to decrease the number of parameters and to decrease the effect of disturbing noise.

This technique has many advantages, for instance, it is very easy to implement, its computational needs are very low – compared to the previous methods – and for many users it is more understandable since it stays closer to the LTI framework. The drawback is that the accuracy strongly depends on the speed of time-variations.

First, the frequency domain method will be presented.

10.3 The frequency domain approach

10.3.1 The model

In general, in the branches of the LTI systems (see Eq. (10.2)), the \( G_i \) FRF at time instant \( i \) can be estimated. The user is free to choose any kind of (nonparametric) FRF estimation method such as the advanced technique like the smoothed FRF estimation based on IRF truncation [130] or the regularization based frequency domain technique [131].
Due to its simplicity, the proposal in this thesis is to use the simple CPSD (cross-power spectral density) method (see Definition 3.3) as follows:

\[
\hat{G}_i[m] = \frac{\mathcal{F}_D[R_y[i\ldots i+\Delta]u[i\ldots i+\Delta]]}{\mathcal{F}_D[R_u[i\ldots i+\Delta]u[i\ldots i+\Delta]]} = \frac{\hat{S}_y[i\ldots i+\Delta]u[i\ldots i+\Delta][m]}{\hat{S}_u[i\ldots i+\Delta]u[i\ldots i+\Delta][m]} = \frac{\hat{y}[m]}{\hat{u}[m]} 
\]

(10.5)

where \(m\) is the frequency index (harmonic), \(\mathcal{F}_D\) is the discrete Fourier transformation (see Definition 3.4), \(\hat{R}_y[u\ldots u]/\hat{R}_u[u\ldots u]\) are the estimates of the cross-/auto-correlation function (see Definition 3.1), \(\Delta \in \mathbb{N}^+\) is the width of the observation window and \(\hat{G}_i\) is \(1\times\Delta\) complex vector representing an LTI FRF.

Every \(\hat{G}_i[m]\) is a partial estimate of the frozen transfer function at time instant \(i\). Note, that in the case of a multisine excitation, it is possible to estimate the FRF by a simple discrete Fourier transformation applied as follows:

\[
\hat{G}_i[m] = \frac{\mathcal{F}_D[y[i\ldots i+\Delta]]}{\mathcal{F}_D[u[i\ldots i+\Delta]]} = \frac{\hat{y}[m]}{\hat{u}[m]} 
\]

(10.6)

The problem with this method lies in the fact that \(\hat{G}_i[m]\) will contain some undesired spectral leakage. The effect of the spectral leakage can be decreased by choosing \(\Delta\) to be large, and by using a windowing method such as diff, sine, triangular, Hann, Hamming, or Blackman-Harris windowing. A general overview about the spectral leakage and about some typical windowing functions can be found in [31], a study about the bias and variance of the FRF measurement can be found in [132].

The other possibility is to use a periodic excitation signal such as a random phase multisine. In this case the frozen FRFs are evaluated on each of the periods and they are almost free from the spectral leakage error, therefore the estimates are more accurate. This technique implies that \(\Delta\) is set to the length of period, the number of frozen FRFs \((N_{LT1})\) is the number of repetitions and \(i = 0, \Delta - 1, 2\Delta - 1 \ldots (N_{LT1} - 1)\Delta - 1\). It is important to remark that this method is usable, when the time variations are sufficiently slow, i.e. the system parameters do not vary significantly in a period. According to Definition 10.1, this is possible because the LsTV system is sufficiently slow. This technique is known as short-time Fourier transform technique [124].

Further on, it is assumed that the user has chosen an appropriate method to eliminate the spectral leakage.
To reduce the complexity of the notational system, let us denote the N×Δ (complex) matrix $\hat{G}_{\text{FTF}}$ the frozen transfer function. Every row vector in $\hat{G}_{\text{FTF}}$ (denoted as $\hat{G}_{\text{FTF}}[i]$) is a partial estimate of the FTF at time instant $i$, i.e.:

$$\hat{G}_{\text{FTF}}[i] = \hat{G}_i[m] \quad (10.7)$$

The frozen transfer function of the linear time-varying system is now given by:

$$\hat{G}_{\text{FTF}} = \begin{bmatrix} \hat{G}_0[m] \\ \vdots \\ \hat{G}_{N-1}[m] \end{bmatrix} = \begin{bmatrix} \hat{G}_{\text{FTF}}[0] \\ \vdots \\ \hat{G}_{\text{FTF}}[N-1] \end{bmatrix} \quad (10.8)$$

Observe that this formulation of the FTF structure is very close to the structure of the instantaneous transfer function, because in both cases for every time instant a certain transfer function is associated. Of course, the interpretations are different. The ITF is the true description of an LTV system at any given time, and the FTF gives information about the system considered to be frozen at certain time instances. When time variations are sufficiently smooth, the FTF will approximate the ITF quite well [133].

In the proposed estimation technique, the FTF is obtained by a special sliding windowing method, which will be discussed in the following section. Similar approaches can be found for instance in [124], [134], [135], [125].

### 10.3.2 Obtain the frozen transfer function

In order to obtain a frozen transfer function of an LsTV system, a sliding (moving) window technique is used here. In the frame of the sliding window, the samples describe an LTI system at a given time. The window is moved step by step, sample by sample towards the last data sample of the observation. Each shift of the observation window provides a frozen LTI model at a given time. Alternatively, this window can be shifted with a larger step size, when periodic excitation is used and/or when the time-variations are very slow.

Next, the proposed algorithm is discussed. An illustration of the method is shown in Figure 10.5. The steps described here can be followed in the figure as well.

**Initial steps:**

i. *Determine the window width:*

The first step is to determine the window width $N_w$ with respect to the desired frequency resolution and the time variations. If the window width is too narrow, then the frequency resolution will be low and some significant
dynamics cannot be discovered and in addition to that, the effect of spectral leakage is larger. But on the other hand, unfortunately, the window width cannot be chosen arbitrarily wide. This lies in the fact that during the observation in the window, some significant time variations can occur and they can overlap to the other samples, therefore it may result in a large estimation error. The advice is to choose the window width to the minimal length, which can describe the system dynamics well.

ii. Define the (initial) timespan:
The observation time window will be now: \( t_{\text{span}} = t_s, \ldots, t_s + N_w - 1 \) (with \( t_s = 0 \)).

Iterative estimation and compensation:

iii. Estimation:
Take the input/output samples at time instants \( t_{\text{span}} \) and compute the estimate of the frequency response function such that:

\[
\hat{G}_{\text{LTI}} = \frac{\mathcal{F}_D\{R(w[1\ldots N_w-1]|[t_{\text{span}}]|)(w[1\ldots N_w-1]|u[t_{\text{span}}])\}}{\mathcal{F}_D\{\hat{R}(w[1\ldots N_w-1]|u[t_{\text{span}}]|)(w[1\ldots N_w-1]|u[t_{\text{span}}])\}}
\] (10.9)

where \( w[t] \) is a suitable window function to decrease the effect of the spectral leakage.

Note that, the FRF can be computed in other ways, this is just an example. This LTI estimation will be a partial estimate of the frozen transfer function at time instant \( t_s \):

\[
\hat{G}_{\text{FTF}}(t_s) = \hat{G}_{\text{LTI}}
\] (10.10)

Note that, in the case of a periodic multisine excitation, the FRF can be estimated by the simple Fourier transformation as shown in Eq. (10.6).

Of course, this estimation step can be replaced to any – efficient – FRF estimation technique according to the user’s choice.
iv. *Sliding the window:*

Move the sliding window with one step ahead such that \( t_s = t_s + 1 \). In this case the time span of the observation window is \( t_{\text{span}} = t_s, \ldots, t_s + N_w - 1 \). The step size can be larger, if the time variations are slow. In this case the interleaved points can be approximated by using a B-spline interpolation shown in the next post-processing section.

v. *Go back to step (iii.)*:

Continue with step (iii.) until the sliding window arrives at the last sample \((N - 1)\).

vi. *Leave the iteration:*

When the sliding window arrives at the last data sample, the last estimated FRF will be used to estimate the FRFs at the remaining time instants, i.e.:

\[
\hat{G}_{\text{FTF}}\{N - N_w + 1 \ldots N\} = \hat{G}_{\text{FTP}}\{N - N_w\} = \hat{G}_{\text{LTI}} \quad (10.11)
\]

In this case we cannot move forwards the window (because the last sample is reached). Instead of moving towards the sliding window, we can decrease the width of the window. By doing that the frequency resolution might be insufficient to describe the dynamic of the system. In this particular situation what we gain with the frequency resolution is basically more than what we lose with the unmodeled time-variations.
10.3.3 Post-processing

10.3.3.1 Introduction

Because the obtained frozen transfer function consists of many noisy FRFs and the shapes of the adjacent FRFs are very similar – or in other words, the time variations are smooth – it is possible to use a B-spline model order reduction technique. Apart from the gain in memory needs, there are two more advantages of the use of B-spline smoothing to be remarked. First, the effect of the disturbing noise and overlapped time-variations will be lower. Second, the frozen transfer function can be estimated at any arbitrary time within the observation interval, and not only at time of the measurement. Next, the discussion of the B-spline proposal follows.

In the previous chapter a special double smoothing technique is applied once over the system time $\tau$ (direction of the impulse responses, referring to the system behavior) and once over the global time $t$ (referring to the system memory). This basic idea remains the same and applied – in a special form – for the current problem as well.
It is important to remark that apart from Spline techniques, there is a distinguished method called LOWESS (locally weighted scatterplot smoothing) technique [136]. It is already shown in [21] that the computing time of LOWESS explodes as the number of samples is increasing, while the B-spline fitting technique is linear in number of knots and the accuracy of the LOWESS method is somewhat limited compared to B-splines.

10.3.3.2 Complex surface fitting using B-splines

The only difficulty related to B-splines can be found in the nature of the frequency response functions: they are complex functions and therefore we need to tackle complex numbers with the smoothing B-splines. This is not a trivial task and until now it was implicitly assumed that the control points are real numbers – in the time domain we work with real numbers only.

The problem in this particular case lies in the fact that there is no guarantee that the real and the complex parts of the FRF are varying together – an illustration for this problem is shown in Figure 10.6. This can lead to a difficult situation of choosing the positions of the control points – as a reminder, in this thesis only equidistantly spaced B-splines are considered.

\[ s_c(x) = \sum_{i=0}^{n_{\text{real}}-1} c_{\text{real}_i} B_i^{d_{\text{real}}(x)} + i \sum_{i=0}^{n_{\text{imag}}-1} c_{\text{imag}_i} B_i^{d_{\text{imag}}(x)} \]  

(10.12)

where \( s_c \in \mathbb{C} \), \( c_{\text{real}_i}, c_{\text{imag}_i} \in \mathbb{R} \), \( n_{\text{real}} \) and \( n_{\text{imag}} \) are the cardinality of the real and imaginary control points, and \( i = \sqrt{-1} \).

Figure 10.6: An example of a frequency transfer function of an LTI system.

To describe complex functions using B-splines there are two possibilities. The first one is that we define an independent B-spline function for the real part, and another independent B-spline function for the imaginary part. In this case the approximation function given in Eq. (8.1) can be rewritten as:

\[ s_c(x) = \sum_{i=0}^{n_{\text{real}}-1} c_{\text{real}_i} B_i^{d_{\text{real}}(x)} + i \sum_{i=0}^{n_{\text{imag}}-1} c_{\text{imag}_i} B_i^{d_{\text{imag}}(x)} \]  

(10.12)

where \( s_c \in \mathbb{C} \), \( c_{\text{real}_i}, c_{\text{imag}_i} \in \mathbb{R} \), \( n_{\text{real}} \) and \( n_{\text{imag}} \) are the cardinality of the real and imaginary control points, and \( i = \sqrt{-1} \).
Chapter 10 - The frequency domain approach

The second possibility is that we define several complex coefficients such that the approximation function given in Eq. (8.1) can be rewritten as:

\[ s_c(x) = \sum_{i=0}^{n_x-1} c_i B_i^{d_x}(x) = \sum_{i=0}^{n_x-1} (\Re\{c_i\} + i\Im\{c_i\})B_i^{d_x}(x) \]  

(10.13)

where \( s_c, c_i \in \mathbb{C} \), the operators \( \Re \) and \( \Im \) denote the real and imaginary part of a complex number.

The second possibility seems to be more useful, therefore the proposed post-processing method is based on the latter possibility. The drawback that – in theory – the amount of memory needs to be used is a little higher (because we need to have more knots/ control points), but the difference is not significant (assuming that the number of control point for the real and imaginary parts are almost equal).

10.3.3.3 The proposed post processing technique

The equations above can describe a complex one dimensional function such as the FRF of an LTI system. The frozen transfer function is a two dimensional function consisting of many FRFs and due to its nature it can be imagined as a (complex) surface.

This knowledge allows us to use a two dimensional B-spline form as it is shown in Eq. (8.7):

\[ s_c(x,y) = \sum_{i=0}^{n_x-1} \sum_{j=0}^{n_y-1} c_{i,j} B_i^{d_x}(x)B_j^{d_y}(y) \]  

(10.14)

To simplify the equations and the notations, this can be rewritten in a matrix form as:

\[ s_c = B^{d_x}c_{ij}B^{d_y} \]  

(10.15)

where

\[ c_{ij} = \begin{bmatrix} 
  c_{0,0} & \cdots & c_{0,n_y-1} \\
  \cdots & \cdots & \cdots \\
  c_{n_x,0} & \cdots & c_{n_x-1,n_y-1} 
\end{bmatrix} \]  

(10.16)
Chapter 10- The frequency domain approach

\[
B^d_x = \begin{bmatrix}
B^d_x(x_0) & \cdots & B^d_x(x_{N_x-1}) \\
\vdots & \ddots & \vdots \\
B^d_x(x_{N_x-1}) & \cdots & B^d_x(x_{N_x-1})
\end{bmatrix}
\]

(10.17)

\[
B^d_y = \begin{bmatrix}
B^d_y(y_0) & \cdots & B^d_y(y_{N_y-1}) \\
\vdots & \ddots & \vdots \\
B^d_y(y_{N_y-1}) & \cdots & B^d_y(y_{N_y-1})
\end{bmatrix}^T
\]

(10.18)

\[
s_c(x_0,y_0) & \cdots & s_c(x_{N_x-1},y_0) \\
\vdots & \ddots & \vdots \\
s_c(x_0,y_{N_y-1}) & \cdots & s_c(x_{N_x-1},y_{N_y-1})
\]

(10.19)

Using this matrix structure, we can redefine the matrices above (Eq. (10.15)–(10.19)) into the case, where the surface \( s_c \) to be estimated is the complex FTF – using complex control points. Simply the only thing that we need to do is just to replace the \( x \) variable by the frequency index \( m \), \( N_x \) by \( N_w \), the \( y \) variable to the global time variable \( t \), \( N_y \) by \( N \) and the \( s_c \) surface to the estimate of the frozen transfer function \( \tilde{G}_{\text{FTF}} \) such that:

\[
\tilde{G}_{\text{FTF}} \sim \tilde{G}_{\text{FTF}}
\]

\[
\tilde{G}_{\text{FTF}} = B^d_t c_{ij} B^{d_m} = B^d_t \left( \Re \{c_{ij}\} + i \Im \{c_{ij}\} \right) B^{d_m}
\]

(10.20)

A similar application to this data reduction technique can be found for instance in the works [13], [15], [117], [118]. Using this information, the proposed algorithm follows.

10.3.3.4 The cost function

The above mentioned terms allow us to define a cost function for the estimation problem. We need to fit the real and the imaginary parts separately – as it is recommended for the LTI case in [72], [84], [13].

\[
V_{BS}^{\text{real}} = \|\Re \{\tilde{G}_{\text{FTF}}\} - \Re \{\tilde{G}_{\text{FTF}}\}\|^2
\]

(10.21)

and
Chapter 10- The frequency domain approach

\[ V_\text{BS}^{\text{imaginary}} = \| \Im\{ \tilde{G}_{\text{FTF}} \} - \Im\{ \tilde{G}_{\text{FTF}} \} \|_2^2 = \] (10.22)

\[ \| \Im\{ \tilde{G}_{\text{FTF}} \} - B^d t (\Im\{ c_{ij} \}) B^d m \|_2 \]

The solution of \( c_{ij} \) is given by (the derivation is shown in Appendix B.2):

\[ c_{ij} = \arg\min_{c_{ij}} V_{\text{BS}}^{\text{real}} + i \arg\min_{c_{ij}} V_{\text{BS}}^{\text{imaginary}} = \]

\[ \left( B^d t^T B^d t \right)^{-1} B^d t^T \Re\{ \tilde{G}_{\text{FTF}} \} \left( B^d m^T B^d m \right)^{-1} B^d m^T + \]

\[ \left( B^d t^T B^d t \right)^{-1} B^d t^T \Im\{ \tilde{G}_{\text{FTF}} \} \left( B^d m^T B^d m \right)^{-1} B^d m^T \] (10.23)

In order to determine the number of necessary control points, there are two possibilities.

First, [13], [15], [117], [118], [14] recommend to increase the number of knots and to perform a whiteness test for the residuals. When the residuals are Gaussian white noise, the optimal number of control points is found. The advantage of this technique is that it has lower computational load.

Secondly, it is recommended to perform a cross-validation. In this case we have more influence on the parameters. While the previous proposal does not say anything about the degrees of the B-splines, in this case by varying the degrees (typically between 2 and 9, with a median 2..3) and number of used knots (typically 5..25% of the number of samples), the optimal hyperparameters can be found.

It is important to remind that increasing the number of knots is equivalent to decreasing the step size. This is discussed in details in Appendix B.6.

As a result of B-spline smoothing, the variance of the estimate is decreasing compared to the original estimated FRFs. The variance of the B-spline based double smoothing is given by:

\[ \text{variance}\{ c_{ij} \} = \]

\[ \left\{ \left( B^d t^T B^d t \right)^{-1} B^d t^T \right\}^2 \text{variance}\{ \tilde{G}_{\text{FTF}} \} \left\{ \left( B^d m^T B^d m \right)^{-1} B^d m^T \right\}^2 \] (10.24)

where the squaring operator \{\cdot\}^2 and the variances are applied componentwise.

Further computational details can be found in [13].
10.4 Time domain approach

10.4.1 The model

The previously discussed technique can be easily implemented in the time domain as well. A branch $G_i$ in the series of the LTI systems (see Eq. (10.2)) now represents an impulse response function. To clearly distinguish the time domain notation from the frequency domain notation, $\hat{h}_i[\tau]$ denotes a partial estimate of the frozen impulse response function (FIRF) at time instant $i$. Each $\hat{h}_i[\tau]$ is obtained from the corresponding input ($u[i \ldots i + \Delta]$) and output data ($y[i \ldots i + \Delta]$). In this particular case, $\Delta$ will be the length of the longest considered impulse response function ($L$).

To obtain a partial frozen IRF estimate, there is no restriction on the estimation method, so the user is free to choose any preferred technique, provided that the IRF estimate will be sufficiently accurate with respect to the target application. Despite the freedom of choice, due to their beneficial properties, the proposal is to use either the regularization or the B-spline technique.

The $N \times L$ (real) matrix $\hat{h}_{\text{FIRF}}$ represents the frozen impulse response function. Every row vector in $\hat{h}_{\text{FIRF}}$ (denoted as $\hat{h}_{\text{FIRF}}(i)$) is a partial estimate of the frozen impulse response function at time instant $i$, i.e.:

$$\hat{h}_{\text{FIRF}}(i) = \hat{h}_i[\tau]$$  \hspace{1cm} (10.25)

It is important to remark that this interpretation is not equivalent to the two dimensional interpretation shown in the regularization chapters. They differ in the notion (true vs frozen estimates) and in the structure of the two dimensional formulation (diagonal (see Eq. (6.3) vs row-wise structure)).

10.4.2 Obtain the frozen impulse response function

In the proposed estimation technique – similarly to the previous case – the FIRF is obtained by a special sliding windowing method. In the frame of the sliding window, the samples describe an LTI system. The window is moved step by step, sample by sample towards the last data sample of the observation. Each shift of the observation window provides a frozen LTI model at a given time. Alternatively, this window can be shifted with a larger step size, when the time-variations are very slow.

Next, the proposed algorithm based on the algorithm of FTF is discussed.
Initial steps:

i. *Initialization of the window parameters:*

Determine the length of the longest considered impulse response function. The window width $N_w$ should be at least $L$ but not too large because some significant time variations can occur and they can overlap to the other samples.

Own experiences show that the regularization technique will provide a good estimate when $N_w \geq 1.5L$ and for the B-spline technique $N_w \geq 1.3L$. The typical LS estimation technique requires $N_w \geq 3L$ [78].

ii. *Define the initial timespan:*

The observation time window will be now: $t_{span} = t_s,\ldots,t_s + N_w - 1$ (with $t_s = 0$).

Iterative estimation and compensation:

iii. *Estimation:*

Take the input/output samples at time instants $t_{span}$ and compute the estimate of the IRF function such that:

$$\hat{h}_{\text{FIRF}}(t_s) = \hat{h}_{t_s}[\tau] \tag{10.26}$$

iv. *Sliding the window:*

Move the sliding window with one step ahead such that $t_s = t_s + 1$. In this case the time span of the observation window is $t_{span} = t_s,\ldots,t_s + N_w - 1$.

The step size can be larger, if the time-variations are very slow.

v. *Go back to step (iii.):*

Continue with step (iii.) until the sliding window arrives at the last sample ($N - 1$).
vi. Leave the iteration:

When the sliding window arrives at the last data sample, the last estimated IRF will be used to estimate the IRFs at the remaining time instants, i.e.:

\[
\hat{h}_{\text{FIRF}}[N - N_w + 1 \ldots N] = \hat{h}_{\text{FIRF}}[N - N_w] = \hat{h}_t[\tau]
\]  

(10.27)

10.4.3 Post-processing

Because all impulse responses have real values, the B-spline post-processing algorithm becomes simpler than in the FTF case. Based on Eq. (10.14)–(10.19), by replacing some variables, the two dimensional B-spline form is given by:

\[
\hat{h}_{\text{FIRF}} \rightarrow \tilde{h}_{\text{FIRF}}
\]

\[
\tilde{h}_{\text{FIRS}} = B^{d_t} c_{ij} B^{d_r}
\]

(10.28)

where in Eq. (10.15)–(10.19) the variable \(x\) is replaced to the system time variable \(\tau\), \(N_x\) by \(N_w\) (or by \(L\)), the variable \(y\) is replaced by the global time variable \(t\), \(N_y\) by \(N\) and the \(s_c\) surface is replaced to the estimate of the FIRF \(\hat{h}_{\text{FIRF}}\).

The terms above allow us to define a cost function for the particular time domain problem:

\[
V_{BS} = \| \hat{h}_{\text{FIRF}} - \tilde{h}_{\text{FIRF}} \|_2^2 = \| \hat{h}_{\text{FIRF}} - B^{d_t} c_{ij} B^{d_r} \|_2^2
\]

(10.29)

The solution of \(c_{ij}\) is given by (a similar derivation is shown in Appendix B.2):

\[
c_{ij} = \arg\min_{c_{ij}} V_{BS} = \\
\left( B^{d_t} B^{d_t} \right)^{-1} B^{d_t} \hat{h}_{\text{FIRF}} \left( B^{d_t} B^{d_r} \right)^{-1} B^{d_r T}
\]

(10.30)

In order to determine the number of necessary control points, the two techniques presented in Section 10.3.3 can be used here as well: whiteness test of the residuals or a cross validation technique.
10.5 Example

In this section several examples are shown to illustrate the frozen nonparametric approach. First, let us observe the system given by Eq. (10.3). According to the DEFINITION 10.1, during the length of the impulse responses, the system parameters should remain almost the same. Figure 10.7 shows two impulse responses of the observed system: the first one starts at time instant $t_0 = 0$, the second one starts at the last sample of the first impulse response function. Observe that the two impulse responses differ a lot. It means that the underlying system is not slowly time-varying, consequently it means that the proposed frozen methods cannot be used.

![Figure 10.7: An example of a frequency transfer function of an LTI system.](image1)

By slowing down the changes in the system given by Eq. (10.3), the system will become LsTV. The new underlying system used for the performance test is given by:

$$H(t, s) = \frac{(0.1)(t + 1)(0.01)s + 1}{(5 + 0.1t)0.01s^2 + 1s + (1 + 0.1t)0.01}$$

(10.31)

where $t = 0 \ldots N - 1$.

Figure 10.8 shows two impulse responses of the new observed system at time instants $t_0 = 0$ and $t_0 = 29$. In this case it can be observed that the difference between the two impulse responses is not significant.

![Figure 10.8: An example of a frequency transfer function of an LTI system.](image2)
In order to obtain an estimation and validation set, there are two simulations made with the same conditions (zero initials, time variations in the same state). Both excitation signals are 25 repeated random phase multisines with a length of 45, giving a total length \( N \) of 1125. The disturbing noise is white Gaussian with an SNR of 35 dB. The observation window is evaluated at five different window widths: \( N_w = 30; 45; 60; 75; 90 \).

For the time domain estimations both regularization (using TC kernels) and the B-spline LTI estimation methods are used. For the frequency domain approach, when spectral leakage is present, a half-sine window is used – as it is recommended by [132].

Table 10.1 shows the results of all the estimation outputs and Figure 10.9 shows the best achieved time and frequency domain estimations.

At first glance it seems that the frequency domain method does not perform as well as the time domain method. The reason behind that lies in the fact that for the time domain estimates the most advanced nonparametric methods were used and for the frequency domain estimation just a basic procedure.

As expected, the accuracy of the estimates is increasing with the window width up to the moment, where the time-variations are (significantly) showing up. According to the results of this simulation, the inflexion point is around \( N_w = 75 \).

When the sliding window step size equals to the window width and to the integer number of the period length \( N_w = 45; 90 \), the frequency domain estimates perform better compared to the case, where the step size is one. This is due to the fact that there is no spectral leakage present in these special cases.

When the appropriate window width and B-spline post processing algorithm are used, the error drops to 10% of the case, where the system is considered to be an LTI system.

Also observe that, when a small observation window is used \( (N_w = 30) \), the B-spline IRF estimation method performs better. This performance originates from the nature of the model reduction behavior of B-splines: equations are better conditioned.

The personal computer configuration used for the simulation can be found in Appendix A.6.
Table 10.1: Comparison of different methods used for a simulation.

<table>
<thead>
<tr>
<th>Approach</th>
<th>( N_w ) (step size)</th>
<th>\text{relative rms error [%]}</th>
<th>\text{computational time [sec]}</th>
<th>\begin{tabular}{l} before post processing \ after post processing \end{tabular}</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTI estimate using regularization</td>
<td>1125 (-)</td>
<td>44.12 (3.41)</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>frequency domain</td>
<td>30 (1)</td>
<td>64.30 (4.45)</td>
<td>46.20 (17.39)</td>
<td></td>
</tr>
<tr>
<td>time domain, regularization</td>
<td>30 (1)</td>
<td>45.09 (9.23)</td>
<td>44.00 (16.48)</td>
<td></td>
</tr>
<tr>
<td>time domain, B-splines</td>
<td>30 (1)</td>
<td>38.01 (7.54)</td>
<td>34.51 (15.51)</td>
<td></td>
</tr>
<tr>
<td>frequency domain</td>
<td>45 (1)</td>
<td>39.76 (4.01)</td>
<td>28.65 (17.12)</td>
<td></td>
</tr>
<tr>
<td>time domain, regularization</td>
<td>45 (1)</td>
<td>24.43 (8.17)</td>
<td>18.48 (16.37)</td>
<td></td>
</tr>
<tr>
<td>time domain, B-splines</td>
<td>45 (1)</td>
<td>25.40 (7.04)</td>
<td>19.36 (16.98)</td>
<td></td>
</tr>
<tr>
<td>frequency domain</td>
<td>45 (45)</td>
<td>27.78 (2.66)</td>
<td>21.54 (4.63)</td>
<td></td>
</tr>
<tr>
<td>frequency domain</td>
<td>60 (1)</td>
<td>31.49 (3.08)</td>
<td>24.20 (18.74)</td>
<td></td>
</tr>
<tr>
<td>time domain, regularization</td>
<td>60 (1)</td>
<td>14.87 (7.29)</td>
<td>8.92 (17.47)</td>
<td></td>
</tr>
<tr>
<td>time domain, B-splines</td>
<td>60 (1)</td>
<td>14.92 (7.87)</td>
<td>10.30 (16.86)</td>
<td></td>
</tr>
<tr>
<td>frequency domain</td>
<td>75 (1)</td>
<td>26.15 (3.05)</td>
<td>15.6 (19.97)</td>
<td></td>
</tr>
<tr>
<td>time domain, regularization</td>
<td>75 (1)</td>
<td>7.37 (8.79)</td>
<td>4.97 (17.54)</td>
<td></td>
</tr>
<tr>
<td>time domain, B-splines</td>
<td>75 (1)</td>
<td>8.36 (8.10)</td>
<td>6.60 (17.01)</td>
<td></td>
</tr>
<tr>
<td>frequency domain</td>
<td>90 (1)</td>
<td>41.34 (2.34)</td>
<td>36.87 (18.39)</td>
<td></td>
</tr>
<tr>
<td>time domain, regularization</td>
<td>90 (1)</td>
<td>17.87 (7.89)</td>
<td>11.09 (16.95)</td>
<td></td>
</tr>
<tr>
<td>time domain, B-splines</td>
<td>90 (1)</td>
<td>19.65 (8.72)</td>
<td>12.01 (17.12)</td>
<td></td>
</tr>
<tr>
<td>frequency domain</td>
<td>90 (45)</td>
<td>34.59 (0.73)</td>
<td>27.60 (3.41)</td>
<td></td>
</tr>
</tbody>
</table>
Figure 10.9: A simulation is shown to compare different estimation methods. The left column shows the time domain and the right column shows the frequency domain data. The figure on the top shows the true system. The second row shows the frozen B-spline estimates. The third row shows the true output (gray line) and the absolute error between the modeled output and the true validation output (black asterisk).
10.6 Conclusions

In this chapter a simple (parameter reducing) method is presented to estimate the slowly time-varying systems non-parametrically with modified and generalized B-spline technique. The method is intended to provide a good estimation, when the behavior of the system is slowly varying.

The main advantage of this technique that it is very easy to understand and to implement, and last but not least, its computational resources are very low – compared to the previous methods.

The discussed parameter reducing technique is very useful because:

- it reduces the effect of the disturbing noise and the undesired effects of time-variations,
- it is more flexible and faster than the single polynomial fitting,
- the computing time is linear in the number of knots and much faster than the most similar smoothing and parameter reducing LOWESS (locally weighted scatterplot smoothing) method [21].

The drawback is that the system is estimated not at once – like in the case of the regularization and the B-spline non-frozen technique – and it is possible that a significant time variation can happen (can overlap) resulting in a larger estimation error around the time of the significant variation.

It is important to remark that the B-spline smoothing technique – with some limitations – is able to smooth out the errors caused by some overlapping.
Chapter 11
An experimental comparison of the proposed methods based on the measurement of a time-varying system

Real measurement experiments play a key role in system identification. They allow us to verify the built models – using the proposed methods – on real experimental conditions. Sometimes simulations are not realistic enough, possible error sources are among others: no disturbing noise is added to the observations, some possible error sources are not taken into account or the dataset generated by simulation is based on a wrong description of the system.

In this thesis the presented simulation examples are carefully chosen. They show realistic scenarios (when looking at the SNR level, we can say that they are a little pessimistic).

Despite all the efforts to make the simulations realistic, nothing can be as realistic as the real life. Therefore it is needed to validate the proposed techniques on real experimental data. This was done during the research progress several times: plenty of real measurements were taken and all confirmed the goodness of the proposed methods.

To support the reader, this chapter shows a simple measurement experiment as a toy example. Here, the non-frozen B-spline and the regularization approaches are compared to each other together with the state-of-the-art frequency domain method.
11.1 The measurement setup

11.1.1 The underlying system

In order to verify and compare the proposed methods, a measurement example is shown. The underlying system is a second order time varying active (high-Q/low bandwidth) band-pass filter [137] with controllable resonance frequency. Figure 11.1 shows a schematic of the observed system. The Q factor is the inverse of the fractional bandwidth, the higher the Q factor, the narrower the pass-band.

![Schematic of the underlying system](image)

The operational amplifier is a CA741CE and the J-FET transistor is BF245B with gate voltage $p_0[t]$, the resistors are $R_1 = 10 \, k\Omega$, $R_2 = 470 \, k\Omega$ and two capacitors $C = 10 \, nF$. The transfer function of the underlying system – without the transistor – is given by:

$$H(s) = \frac{V_{out}}{V_{in}} = \frac{H_0 \beta s}{s^2 + \beta s + \omega_0^2} \quad (11.1)$$

where $H_0 = -\frac{R_1 C}{R_1 + 2C}$ is the gain at the center frequency, $\beta = \frac{2C}{R_1 C^2} = \omega_0 / Q$ is the bandwidth, $Q$ is the quality factor, $\omega_0 = \frac{1}{\sqrt{R_1 C^2 \cdot R_1 R_2}}$ is the center frequency.

The time-varying behavior originates from the time-varying gate voltage $p_s[t]$ of the J-FET transistor resulting in time varying resistor $R_2(t)$. This leads to the ITF:

$$H(t, s) = \frac{H_0 \beta s}{s^2 + \beta s + \omega_0^2} =$$

$$= \frac{1}{R_1 C^2} \cdot \frac{R_1 R_2(t)}{R_1 + R_2(t)} \quad \text{varying center frequency} \quad (11.2)$$
11.1.2 Instrumentation

The measurement was executed by using a HP VXI bus mainframe. The excitation signals were generated by the arbitrary function generator Hewlett-Packard (HP) E1445A (Direct Digital Synthesis technique, 40 MSa/s DAC, 13 bit resolution).

Signals are measured by HPE1430A digitizer module (10 MSa/s, 18 bit resolution, antialiasing filter). The mainframe was connected to the PC with the VXI bus interface module National Instrument (NI) VXI-MXI-2 (38 MB/s DMA transfer rate). The PC data acquisition software was Measurement & Automation (v 2.2) from NI. The PC configuration used to build models can be found in Appendix A.6.

11.1.3 Excitation and scheduling signals

The total length of the measurement consists of 500 samples. The filter is excited by a frequency bounded random phase multisine (see Figure 11.2). The input and output signals are band limited. The output spectrum is shown in Figure 11.3. The scheduling variable – causing the time-varying behavior – is shown in Figure 11.4.

Figure 11.2: The measured excitation signal in the frequency domain.

Figure 11.3: The measured output signal in the frequency domain.
In order to use the cross-validation methods presented in Sections 6.5 and 9.3, two measurements are executed providing an estimation and a validation dataset. To avoid any unnecessary high computational load, the noise variance $\sigma^2$ is estimated from the sample variance. This gives a value around 60 dB (precisely 58.43 dB) below the excitation signal level. This noise floor can be seen in Figure 11.3., in the non-excited frequency band.

Before obtaining the two dimensional impulse response function, some preparations are needed in order to use the B-spline technique. The main issue in this case is that the B-spline estimation is used for all the data in the frequency domain but not all the frequencies are excited. This problem stands for the extrapolation which is not possible with the nonparametric smoothing interpolation.

For the B-spline estimation and validation procedures, the input signal should be modified in the frequency domain. The non-excited frequency harmonics should be set to zero. Corresponding to this step, the output spectrum should be modified, too. It is because the B-spline estimation should be computed only on the excited frequency band.

Because of the computational load, to determine the right model order, a simple LS validation cost function is evaluated assuming that the system is LTI (see Figure 11.5). The selected model order is 80. The initial TC or DC hyper-parameter optimization is done for this LTI model. Here the robust approach is used (see Section 6.4.3) with DC kernels.
Figure 11.5: The validation cost function is shown as a function of the order (the length of the impulse responses).

In order to classify the proposed methodologies the results are also compared with the output of the FDTVident (Frequency Domain Time Varying identification) toolbox written by John Lataire [40], [46], which implements the (state-of-the-art) method discussed in Section 10.1. This toolbox uses a parametric representation. The order of the numerator of the FDTVident estimation is 3, the order of the denominator is 2.

This comparison is also made for the ordinary least squares estimation considering that the underlying system is LTI. This can give an idea about what happens if the time variations are not taken into account.

11.3 Results and conclusions

The representations are different in the time and frequency domain which is not a problem because there is no unique solution. The filter has a varying resonance frequency and damping ratio and it would be beneficial to analyze how the estimators can approximate the system around the moving resonance frequency. For that reason the results are compared with the output measurement in the frequency domain. The discrete output frequency spectrum is computed via the discrete Fourier transform.

Table 11.1 and Figure 11.6 show the results of the best achieved estimations. The black solid line shows the measurement output in the frequency domain at the excited frequency lines around the frequency band of interest. The discrete signs (+,×,*,●) show the relative errors of the estimation.

Observe that the LTI model is not able to follow the time-varying behavior and the error level is very high.
By using the proposed non-frozen methods a good estimation can be – not surprisingly – achieved using the right model orders. The results of these methods are comparable with the frequency domain approach using the FDTVident toolbox.

The best result is achieved by using the 2D regularization technique, but at a price of a high computational load.

As a final conclusion, when the model quality is crucial and the target application is not time sensitive, then it is recommended to use the robust or the flexible regularization approach.

When a good model quality is needed but the available memory and time are limited, then it is recommended to use the B-spline approach instead.

![Figure 11.6: The solid line shows the measurement output in the frequency domain. The asterisk signs (*) are the modeling errors between the B-spline estimation and the measured outputs, the plus signs (+) are the modeling errors between the regularized estimation and the measured outputs, the cross signs (×) are the modeling errors between the FDTVident and the measured outputs, the rhombuses (♦) are the modeling errors between the LTI estimation and the measured outputs.](image)

Table 11.1: Comparison of different methods used for a measurement.

<table>
<thead>
<tr>
<th>Approach</th>
<th>relative rms error [%]</th>
<th>computational time [sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS LTI estimate</td>
<td>81</td>
<td>1.4</td>
</tr>
<tr>
<td>Robust regularization using DC kernels</td>
<td>$4.8 \cdot 10^{-3}$</td>
<td>136117</td>
</tr>
<tr>
<td>Non-frozen B-splines</td>
<td>$8.9 \cdot 10^{-3}$</td>
<td>2381</td>
</tr>
<tr>
<td>FDTVident</td>
<td>$5.7 \cdot 10^{-3}$</td>
<td>154</td>
</tr>
</tbody>
</table>
It was already mentioned that one of the challenges with the time-varying systems is that it is not every time possible to repeat the experiments with the same conditions, therefore in practice we should be able to build an accurate model by using only one single measurement. Table 11.2 shows this case: a single experiment is used to obtain the estimates. In the table the ordinary cross-validation technique (where two independent experiments are used) is compared with the interleaved cross-validation technique (detailed in Section 7.1), the residual analysis (using F-test with 95% confidence interval, discussed in Section 10.3.2) and the empirical Bayes method (detailed in Appendix A.3.3). The relative error is computed on an independent validation dataset. One can observe that the interleaved cross-validation performs slightly better than the residual analysis but at the price of a higher computational time. The empirical Bayes method is the slowest and the less accurate method (it is due to the highly non-convex optimization problem). It is important to remark that the residual analysis cannot be used in every case: the whiteness property of the residuals cannot be guaranteed. For instance, when a process noise that passes through the varying system, the stationarity does not hold anymore.

Table 11.2: Comparison of different hyperparameter fitting methods used for a measurement.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Cross-validation, using different datasets</th>
<th>Cross-validation, using interleaved samples</th>
<th>Residual analysis</th>
<th>Empirical Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robust regularization (DC)</td>
<td>4.8 $\cdot$ 10$^{-3}$ (136117)</td>
<td>6.3 $\cdot$ 10$^{-3}$ (110108)</td>
<td>6.4 $\cdot$ 10$^{-3}$ (98211)</td>
<td>7.9 $\cdot$ 10$^{-3}$ (234309*)</td>
</tr>
<tr>
<td>Non-frozen B-splines</td>
<td>8.9 $\cdot$ 10$^{-3}$ (2381)</td>
<td>1.4 $\cdot$ 10$^{-2}$ (2011)</td>
<td>1.7 $\cdot$ 10$^{-2}$ (1762)</td>
<td>-</td>
</tr>
</tbody>
</table>

* optimization method restarted three times using new initial values of hyperparameters
CONCLUSIONS
Chapter 12
Conclusions

In this PhD thesis three different but interrelated nonparametric identification methodologies were developed, implemented and statistically analyzed. Using the proposed methods it is possible to obtain a good quality model of a linear time-varying system based on noisy observations.

The first method applies kernel based regularization (see Section 12.1.1).

The second and the third methods (see Section 12.1.2) are based on B-splines techniques, where a frozen (Section 12.1.2.1) and a non-frozen (Section 12.1.2.2) models can be obtained.

All methodologies have been validated for correctness on simulation examples and also on measurements of a time varying device. The problems were formulated mainly in the time domain.

The main contributions of my research are summarized in the present chapter. Because of lack of space, some of the results were not discussed in this thesis but they are mentioned in this chapter in a distinguished section. The references – as the outputs of the research – placed in this chapter refer to own works.
12.1 Main contributions of this thesis and scientific statements

12.1.1 Regularization technique for LTV Systems

Regularization is of particular interest in the bias-variance trade-off that characterizes model estimation processing noisy observations. The main idea in regularization is the introduction of a penalty term to the least squares cost function. By doing this we are able to put a limit to the model complexity.

The main feature of the regularization method is to punish the model complexity in an advanced way. For noisy and short data records, the regularized least squares approach turns out to be more robust and even slightly more accurate than the standard prediction error method/maximum likelihood approach. The underlying reason for such behavior is that it may be beneficial to allow some bias to be able to reduce the variance. Based on a Bayesian interpretation of the problem, one gets insight in the choice of the regularization matrix.

My contribution is to combine the two dimensional impulse response function of the LTV systems with the regularization technique in a special way.

Statement 1. I developed a new method to estimate the two dimensional nonparametric impulse response function of linear time-varying systems using a time domain two-dimensional regularization technique. The technique is discussed in details in Chapter 6 – 7.

As a consequence, the solution for the estimation problem is a unique solution, contrary to the classical maximum likelihood estimator, where infinitely many solutions are equally possible.

Following the general analysis results of regularization, I determined the main statistical properties of the estimator: expected value, bias, variance, and the mean square error. These are computed in Appendix A.3.

I verified that – by satisfying ASSUMPTION 5.1 and ASSUMPTION 5.2 – the proposed method boils down to the maximum a posteriori estimation. This is shown in Appendix A.3.1.

In the target application, where LTV systems are considered, the regularization technique plays an essential role. The aim in this particular case is to reduce the degrees of freedom such that a unique, smooth and stable estimate can be obtained.

In the LTI case, when the measurement has a very good quality – or the measurement is sufficiently long – resulting in a lower variance, the regularization term can be neglected. However in the case of LTV systems, the regularization term
is always active. It is needed because all the constraints (smoothness of the impulse responses and time variations, decaying of the impulse responses) are defined here and they are fundamental in the estimation procedure. These constraints are needed to decrease the degrees of freedom.

**Sub statement 1.1** I proved that under certain conditions – by satisfying **Assumption 2.2, Assumption 6.1—Assumption 6.6** – the proposed method reduces the degrees of freedom of the system of linear equations which describe a linear time-varying system by its two dimensional impulse response. The problem formulation is detailed in Section 6.1. The essence of the proposed method is explained in Section 6.4. The derivation of the degrees of freedom is shown in Appendix A.5.

A further issue with time varying systems is that in general the measurement cannot be repeated under the same conditions – due to the time-varying behavior – and therefore all the measured samples must be used for the estimation procedure. It has the consequence that the first impulse responses can be corrupted by the transients. To handle this situation I developed a special method.

**Sub statement 1.2** I proved that using an extension of the proposed technique it is possible to eliminate the effect of the transient – if **Assumption 7.1—Assumption 7.2** are satisfied. The method is explained in details in Section 7.2.4.

A further problem with the proposed two-dimensional regularization is that the memory need grows quadratically with the length of the observation. This means that after some hundreds of samples, the computational limits are reached. I designed an extended estimation method which can tackle this problem.

**Sub statement 1.3** I developed a rapid algorithm to handle large datasets. The proposed method uses a special sliding window technique and the advanced transient elimination method (Sub statement 1.2). This is discussed in details in Section 7.3.

This methodologies presented can be found in Chapter 6–7. The research outputs are [19], [20], [138].

12.1.2 B-spline technique used for Linear Time-variant Systems

The key idea of this method is that generalized B-splines can be used for double smoothing: over the global and system time. If the parameter variation of the observed system is sufficiently slow, with respect to the system dynamics, we will be able to

- reduce the disturbing noise by additional smoothing,
- reduce the number of model parameters that need to be stored,
- decrease the effect of the undesired transient.
To do this, two different approaches can be used. The first approach assumes that the LTV system can be described with frozen FRFs. The second approach replaces the (non-frozen) impulse response functions by a smooth B-spline representation.

12.1.2.1 The frozen FRF and IRF approach

In the case of linear slowly time-varying systems it is a common practice to describe the system as a series of time invariant systems, one at each measurement time. B-splines are used to smooth either over the “frozen” frequency response functions or over the impulse response functions. Assuming that the parameters are slowly changing it is possible to use a sliding window technique to estimate an LTI model using the data in the window. The output of this sliding window is either a series of FRFs or IRFs. By placing them next to each other, we get a surface that can be modeled with several methods, like for instance with B-spline based spline fitting.

Statement 2. I developed a method to estimate the two dimensional nonparametric frozen transfer function/ frozen impulse response of smooth, linear slowly time-varying systems. The method uses a special sliding window technique combining the ideas of the short Fourier transform and a surface description method based on multidimensional complex B-spline basis functions. The problem formulation is discussed in Chapter 10.

Sub statement 2.1 I shown that under certain conditions – by satisfying ASSUMPTION 2.1—ASSUMPTION 2.2, ASSUMPTION 6.1—ASSUMPTION 6.6, ASSUMPTION 10.1—ASSUMPTION 10.3 – the proposed method can estimate the frozen transfer function / frozen impulse response function well. The time domain method to estimate the frozen impulse response function is explained in Section 10.4. The frequency domain method to estimate the frozen transfer function is explained in Section10.3.

Sub statement 2.2 I developed a two dimensional coefficient reduction method which can be used to improve the signal-to-noise ratio of the obtained frozen estimates. This is discussed in Section 10.3.3 and in Section 10.4.3.

This technique is discussed in Chapter 10. The research outputs are [20], [21], [23], [139], [140], [141].

12.1.2.2 The non-frozen impulse response approach

This part presents a new methodology which estimates non-parametrically a two dimensional impulse response function of time-varying systems fully described in the time domain. In this approach we make use of the prior information about the smoothness of the system: the impulse responses and the time variations are smooth. This allows us to use the flexible B-splines such that the classical one dimensional
impulse response function is redefined and replaced by a twodimensional B-spline kernel interpretation.

Unlike the frozen impulse response approaches, here smoothed non-frozen impulse responses are considered such that we do not have overlapping effects. Apart from decreasing the effects of the disturbing noise, it is possible to eliminate the effect of the undesired transient terms. Using the proposed method a good model quality can be achieved with a moderated computational time.

**Statement 3.** I developed a new method to describe smooth linear time-varying systems using the over-defined two dimensional nonparametric B-spline impulse responses. The method is discussed in Chapter 9.

In this technique, instead of directly estimating the coefficients of the two dimensional linear time-varying impulse response function, the coefficients of the linear time-varying B-spline kernels are estimated.

**Sub statement 3.1** I proved that under certain conditions – by satisfying **Assumption 2.2**, **Assumption 6.1** – **Assumption 6.6** – the proposed method reduces the degrees of freedom of the systems of linear equations describing a linear time-varying system by a two dimensional kernel representation. This is shown in Appendix B.5.

As a consequence, the solution for the estimation problem is a unique solution, contrary to the classical maximum likelihood estimator where infinitely many solutions are equally possible.

**Sub statement 3.2** I proved that using an extension of the proposed technique it is possible to eliminate – if the **Assumption 7.1** – **Assumption 7.2** are satisfied – the effect of the transient. This is shown in Section 9.6.

**Sub statement 3.3** I determined the main statistical properties of the estimator: expected value, bias, variance, and the mean square error. This is computed in Appendix B.3.

The related publications are [19], [22], [23], [140], [141]. This technique is discussed in Chapter 9.

**12.1.3 A comparison between different LTV approaches**

Table 12.1 shows the overall conclusions of the estimated methods compared to the FDTVident toolbox. This can give an indication to the potential users how to choose the appropriate modelling method. In the referred table +++ means very good, --- means very bad. It is important to remark that this table is subjective and it reflects the opinion of the author only.
Table 12.1: A general overview of different methods.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Interpretability</th>
<th>Comple-xity</th>
<th>Fit quality</th>
<th>Insensitive to</th>
<th>Resource needs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LsTV</td>
<td>LTV</td>
<td>noise</td>
<td>transient</td>
<td>RAM</td>
</tr>
<tr>
<td>LS LTI estimate</td>
<td>+++</td>
<td>+++</td>
<td>-</td>
<td>---</td>
<td>+++</td>
</tr>
<tr>
<td>Regularization</td>
<td>++</td>
<td>-</td>
<td>+++</td>
<td>+++</td>
<td>--</td>
</tr>
<tr>
<td>Windowed regularization</td>
<td>+</td>
<td>-</td>
<td>+++</td>
<td>++</td>
<td>+++</td>
</tr>
<tr>
<td>Frozen B-splines</td>
<td>+++</td>
<td>++</td>
<td>-</td>
<td>+++</td>
<td>-</td>
</tr>
<tr>
<td>B-spline kernels</td>
<td>-</td>
<td>+</td>
<td>+++</td>
<td>++</td>
<td>+</td>
</tr>
<tr>
<td>B-spline kernels with transient elimination</td>
<td>-</td>
<td>+</td>
<td>+++</td>
<td>++</td>
<td>+++</td>
</tr>
<tr>
<td>FDTVIdent</td>
<td>+</td>
<td>-</td>
<td>+++</td>
<td>++</td>
<td>++</td>
</tr>
</tbody>
</table>

12.2 Contributions presented elsewhere but not included in this thesis

12.2.1 Nonlinearities

In the beginning of my studies and my research I concentrated in my first three semesters on the analysis of (non)linear time invariant systems. I learned the important aspects of the time and frequency domain system identification methods. In my early works I studied how to present and how to handle static, weakly nonlinear time invariant systems using the Best Linear Approximation as an extension of the linear LS technique. This was partly done in Budapest and in Brussels.

Research outputs are [142], [143], [144], [145].

12.2.2 B-spline based LTI identification

At the beginning of my research at the VUB I focused on the smoothing (interpolation) techniques: Bézier polynomials, Spline- and Hermite interpolations and last but not least the B-spline technique. The idea behind this was that it might be possible to use them for linear time-invariant system identification purposes. As a result of this research I developed a modified B-spline technique which can be used for nonparametric estimation.

In this particular case the B-spline methodology serves as a tool to model LTI systems, whose impulse response is sufficiently smooth in the presence of disturbing noise and an initial transient. In this project, discrete time domain and frequency domain descriptions are used. The main advantage of using B-splines over other spline based smoothing techniques is that the system equations are less complex (it is not necessary to compute derivatives). Also, fewer assumptions are made. The power
of B-spline based identification appears in real measurement applications: systems disturbed by noise and transient.

12.2.2.1 Smooth impulse response function estimation

Assuming that the observed LTI system is smooth and its impulse response function has a finite length, it can be described by a special B-spline vector. Depending on the sampling frequency and the system dynamics, adjacent elements of the impulse response function can be the same or very close to each other – or in other words, they can be highly correlated. For the smoothing approximation it is not necessary to use all the elements (in practice around 10…20% of the elements are used). For that reason a truncated impulse response function can be used to describe the plant model. This technique was partly presented in Section 9.6.

12.2.2.2 Smooth frequency response estimation

For LTI systems, the modified B-spline method can be used as a tool for estimating nonparametric smooth FRFs in the frequency domain. B-spline estimations are based on other nonparametric estimates in the frequency domain. Once they are computed, one can use the smoothing technique over the previously estimated system.

12.2.2.3 Virtual knotting technique

The repetition of knots is important, when trying to satisfy the Eq. (8.1)–(8.4). Most textbooks recommend to repeat the first and/or the last knots to satisfy the referred equations. It is already mentioned in this thesis that it can cause a kind of bias error (shrinkage bias, see [122], [140]), especially when the first and/or the last control point is not zero.

Figure 12.1 shows a simple example on the axis t with the proposed algorithm. This kind of “virtual knots” creating yields a better result than “traditional” one, where the function value outside of the domain of interest tends to zero and the first and last knots are repeated.

In this solution the sequences of knots are strongly monotonically increasing (no repetition of knots is allowed on the whole domain). The spacing between the virtual knots is (typically) 1/10 of the spacing between the ordinary knots.
Chapter 12- Contributions presented elsewhere but not included in this thesis

12.2.2.4 Summary

In case of the time domain estimation, the B-spline based impulse response estimation leads to a good fit: the performance of B-spline estimations is better – from the statistical point of view – than the LS solution in the time domain or CPSD or the Hann windowed estimates in the frequency domain. For frequency domain estimation it can be shown that B-splines can improve the statistical properties of these estimators, for instance the expected value of the error and the variance level can be smaller, depending upon the SNR of the data. The distribution of the model fitting errors may change also: in many cases it converges faster to the normal distribution.

In this project a powerful time and frequency domain estimation method was developed for smoothing time-invariant systems with the modified and generalized B-spline technique. The research outputs are [23], [140], [141].

12.2.3 Research activities not related to system identification

This part of my research is still an ongoing progress. The aim of this activity is to verify and validate bibliographical records in publication databases (mainly in the publication database systems called ‘MyCite’) in a reliable, fault-tolerant, interactive way. To solve the task, imperative procedural programming paradigms are used, as well as object-oriented paradigms and behavior-driven models. In the communication
channel the data flow is in a self-created XML format – which is well used at the Hungarian Scientific database MTMT. These data contain identifiers that allow back-tracking of records to web pages in international data bases or catalogs, for instance identifier or DOI in case of IEEE or article identifier in case of PubMed (these data bases usually do not provide API usable for this purpose), independently of its source – using data mining technologies.

Information contained in XML is identified and compared, the records are validated. To the fields contained in the XML, different weights (points) can be assigned, based on publication and website type, since data are on both sides can be misspelt or not correct at all. For a record to be accepted as valid (existing and verified), it has to reach a certain amount of points i.e. the chance for the match of records in the publication database and on the (publisher) website/database is high. In other cases, the record can be validated only manually. Various kinds of further webpages can be identified and integrated within short time and energy, by only filling out a format descriptor.

The research output is [146]. In addition to that, there are many software applications based on this concept.

12.3 Future research

Of course, there is always something more to do. Related to this PhD research several possibilities are available to extend the proposed techniques with new aspects. Some examples are given here.

12.3.1 Extend the methods to linear parameter-varying systems

At the beginning of this thesis, the linear time-varying systems were defined as a special class of linear parameter-varying systems such that the scheduling variable is continuously changing by the time resulting in time-varying behavior.

In the general case, the scheduling variable of the LPV systems is limited to a certain interval and to certain values. Think of the tower crane, where the cable length is limited. The moving of the scheduling variable is also limited.

This means that – as it is expected – using the proposed two dimensional B-spline and regularization technique with some application dependent modification, it might lead to a better fit than the existing nonparametric methods.

This lies in the fact that typically the system of linear equations belonging to an LPV system is better conditioned and the degrees of freedom are lower.
12.3.2 Formulate the problem in the frequency domain

It could be interesting to see how the performance would be of the proposed methods, if reformulated in frequency domain.

There are some initiative works to define frequency response functions of the LTI systems by using the regularization technique and by reformulating the kernels for frequency domain usage.

With B-splines it is already shown by own works that they can be used for frequency domain nonparametric identification.

As an extension of the B-spline approach in the frequency domain, it might be beneficial to use a special redefined and extended local polynomial method. Related to this, it is already shown in own earlier work [23], that LPM based time-invariant frequency domain estimation – in the case of smooth LTI systems – can beat the simple LPM method.

12.3.3 Development of a non-equidistant B-spline algorithm

As it was emphasized at the beginning of the B-spline part of the thesis, in the proposed B-spline methods equidistantly spaced knots were used only. This is because the fitting of the uniform B-splines is a linear-in-parameters problem therefore the simple LS fitting can be used.

When the non-uniform rational B-splines are used, this problem becomes a non-linear optimization problem since the basis functions can have extremely overlapped or extremely small – instead of equal – contributions.

There are existing algorithms mainly used in computer graphics, but there the number of points (pixels) are typically over million and they are highly correlated.

The possible contribution could be to extend the B-spline kernel based LTV estimation method for the non-uniform case.

12.3.4 New kernels for regularization

Nowadays the kernel methods (typically used for LTI systems) are more and more fashionable and therefore subject of new ongoing researches. For instance, the diagonal correlated kernel is also relatively new.

The idea is either to develop new kernel functions – based on advanced prior knowledge – or to try out newly published kernels.
12.3.5 Speeding up the regularization

It is an obvious step to reconsider the observation and covariance hypermatrix structure in order to give a faster computational algorithm. By having an increased computational performance using the sliding window technique, it would be possible to use the regularization technique for online identification purposes.

One possible way of doing this is it to use a decomposition technique. By using advanced vectorial methods, it might be possible to gain some computational load. The other possibility could be to use sparse matrix structures (think of the sparse observation hypermatrix).

12.3.6 Closed-loop identification method

In this thesis, the underlying system is operated in open-loop and there was no feedback observed.

However, we know that in many cases (especially in industrial environment), the underlying system cannot be observed in open-loop because we do not have direct access to the input and output signals.

This will require to reconsider the base-line models and the statistical methods used to describe the system.

12.3.7 Extension to periodicity

In this thesis, arbitrary time varying systems were considered only. It might be worthy to extend the method to periodically time-varying systems and use advanced statistics in order to have higher quality models.

12.3.8 Regularization of B-spline kernels

It would be beneficial to do some research on the control point regularization using the non-frozen B-spline approach. Think of decaying control points in the direction of the impulse responses. It might increase the performance of the estimator.
12.4 List of publications

**Journal papers:**


**Accepted journal papers:**


**International conference papers:**


**PhD symposium papers:**


**Short communications**

These publications have been used for professional research workshops – as supplements for the oral presentations/posters. These European (European Research Network on System Identification) and Belgian-Dutch (BENELUX) workshops are important for the system identification society.


**Miscellaneous**


APPENDICES
Appendix A.

A.1 Derivation of mean square error of model fitting

The derivation of the mean square error (MSE) shown in Definition 5.1 is discussed here.

This can be computed as follows:

$$MSE(h - \hat{h}) = \mathbb{E}\{(h - \hat{h})^2\} =$$

$$\mathbb{E}\{ (h + (\mathbb{E}\{\hat{h}\} - \mathbb{E}\{\hat{h}\})) + \hat{h} \}^2 \} =$$

$$\mathbb{E}\{(\hat{h} - \mathbb{E}\{\hat{h}\})^2 + 2(\hat{h} - \mathbb{E}\{\hat{h}\})(\mathbb{E}\{\hat{h}\} - h) + (\mathbb{E}\{\hat{h}\} - h)^2 \} =$$

$$\mathbb{E}\{(\hat{h} - \mathbb{E}\{\hat{h}\})^2 \} + \mathbb{E}\{ 2(\hat{h} - \mathbb{E}\{\hat{h}\})(\mathbb{E}\{\hat{h}\} - h) \} + \mathbb{E}\{ (\mathbb{E}\{\hat{h}\} - h)^2 \} =$$

$$\mathbb{E}\{(\hat{h} - \mathbb{E}\{\hat{h}\})^2 \} + 2(\mathbb{E}\{\hat{h}\} - h)\mathbb{E}\{\hat{h} - \mathbb{E}\{\hat{h}\}\} + (\mathbb{E}\{\hat{h}\} - h)^2 =$$

$$\mathbb{E}\{(\hat{h} - \mathbb{E}\{\hat{h}\})^2 \} + (\mathbb{E}\{\hat{h}\} - h)^2 =$$

$$\text{var}(\hat{h}) + \text{bias}(\hat{h})^2$$

When an estimator is unbiased ($\mathbb{E}\{\hat{h}\} = h$, i.e. the bias is zero) then the mean square error is given by the variance.

There is often a trade-off between bias and variance: low bias can imply high variance and vice versa.

In the case of the regularization techniques, the objective is to find the smallest MSE.
A.2 Derivation of the regularized time-varying least squares estimation from the cost function

In this section, the derivation of the solution given by Eq. (6.9) is shown.

The regularized cost function is shown in Eq. (6.8), the least squares LTV cost function is shown in Eq. (6.6):

\[ V_{\text{LTV,LS}} = \| y_m - K_{\text{LTV}} h_{\text{NL}\times 1} \|_2^2 \]  

(A.2)

First, expand the Eq. (6.8):

\[
\text{argmin}_{h_{\text{NL}\times 1}} V_{\text{LTV}} = \text{argmin}_{h_{\text{NL}\times 1}} \left\{ V_{\text{LTV,LS}} + V_{\text{LTV,R}} \right\} = \text{argmin}_{h_{\text{NL}\times 1}} \left\{ \| y_m - K_{\text{LTV}} h_{\text{NL}\times 1} \|_2^2 + \sigma^2 h_{\text{NL}\times 1} P^{-1}_{\text{LTV}} h_{\text{NL}\times 1} \right\} \]

(A.3)

Secondly, \( \text{argmin} \) is given by the solution:

\[
\frac{\partial}{\partial h_{\text{NL}\times 1}} \left\{ \| y_m - K_{\text{LTV}} h_{\text{NL}\times 1} \|_2^2 + \sigma^2 h_{\text{NL}\times 1} P^{-1}_{\text{LTV}} h_{\text{NL}\times 1} \right\} = 0
\]

\[-2K_{\text{LTV}}^T y_m + 2K_{\text{LTV}}^T K_{\text{LTV}} h_{\text{NL}\times 1} + 2\sigma^2 P^{-1}_{\text{LTV}} h_{\text{NL}\times 1} = 0 \]

(A.4)

And from here, the estimation is given by:

\[ h_{\text{NL}\times 1,RLS} = (K_{\text{LTV}}^T K_{\text{LTV}} + \sigma^2 P_{\text{LTV}}^{-1})^{-1} K_{\text{LTV}}^T y_m \]

(A.5)
A.3 The regularized estimate and the maximum a posteriori estimation

This section is based on the ideas of [98], [96], [102].

A.3.1 Equivalence

The key idea of the Bayesian framework is that the prior (two dimensional) impulse response function $h_{\text{LTV}_{a \text{priori}}}$ exists and it is seen as a stochastic variable. The goal is to find the posterior distribution given the observation $y_m$.

Before carrying out any statements, let us repeat the joint distribution of two Gaussian variables $x_1$ and $x_2$ such that:

$$
\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} m_1 \\ m_2 \end{bmatrix}, \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \right)
$$

(A.6)

In this case the conditional distribution of $x_1$ given $x_2$ is [30]:

$$
x_1 | x_2 \sim \mathcal{N} (m, C)
$$

$$
m = m_1 + C_{12} C_{22}^{-1} (x_2 - m_2)
$$

$$
C = C_{11} - C_{12} C_{22}^{-1} C_{21}
$$

(A.7)

Let us assume that the prior two dimensional IRF is a Gaussian process with zero mean and $\Pi$ covariance, i.e:

$$
h_{\text{LTV}_{a \text{priori}}} \sim \mathcal{N}(0, \Pi)
$$

(A.8)

Using the baseline model defined in Eq. (6.1) and assumption 6.3-assumption 6.5 the output is given by:

$$
y_m = y + e, \ e \sim \mathcal{N}(0, \sigma^2)
$$

(A.9)

In this case $y_m$ is a stochastic variable with the following distribution:

$$
y_m \sim \mathcal{N}(0, K_{\text{LTV}} \Pi K_{\text{LTV}}^T + \sigma^2 I)
$$

(A.10)

The joint distribution is now:

$$
\begin{bmatrix} h_{\text{LTV}_{a \text{priori}}} \\ y_m \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} 0 \\ K_{\text{LTV}} \Pi \end{bmatrix}, \begin{bmatrix} \Pi & \Pi K_{\text{LTV}}^T \\ K_{\text{LTV}} \Pi & K_{\text{LTV}} \Pi K_{\text{LTV}}^T + \sigma^2 I \end{bmatrix} \right)
$$

(A.11)
From here, the a posteriori estimate of the LTV impulse response is given by:

$$h_{\text{LTV}|y_m} \sim \mathcal{N}\left(h_{\text{a posteriori}}, \Pi_{\text{a posteriori}}\right)$$

$$h_{\text{LTV}_{\text{a posteriori}}} = h_{\text{a posteriori}} = \Pi K_{\text{LTV}}^T (K_{\text{LTV}} \Pi K_{\text{LTV}}^T + \sigma^2 I_{N \times N})^{-1} y_m \quad (A.12)$$

$$\Pi_{\text{a posteriori}} = \Pi - \Pi K_{\text{LTV}}^T (K_{\text{LTV}} \Pi K_{\text{LTV}}^T + \sigma^2 I_{N \times N})^{-1} K_{\text{LTV}} \Pi$$

If we choose $\Pi$ to be equal to $P_{\text{LTV}}$ and by using the matrix equality shown in Section 7.3:

$$(I_{k \times k} + A_{k \times j} B_{j \times k})^{-1} A_{k \times j} = A_{k \times j} (B_{j \times k} A_{k \times j} + I_{j \times j})^{-1} \quad (A.13)$$

then the posterior estimate is given by:

$$h_{\text{LTV}|y_m} \sim \mathcal{N}\left(h_{\text{a posteriori}}, P_{\text{a posteriori}}\right)$$

$$h_{\text{a posteriori}} = \Pi K_{\text{LTV}}^T (K_{\text{LTV}} P_{\text{LTV}} K_{\text{LTV}}^T + \sigma^2 I_{N \times N})^{-1} y_m =$$

$$P_{\text{LTV}} K_{\text{LTV}}^T (K_{\text{LTV}} P_{\text{LTV}} K_{\text{LTV}}^T + \sigma^2 I_{N \times N})^{-1} y_m =$$

$$(P_{\text{LTV}} K_{\text{LTV}}^T K_{\text{LTV}} + \sigma^2 I_{N \times N})^{-1} P_{\text{LTV}} K_{\text{LTV}}^T y_m =$$

$$(K_{\text{LTV}}^T K_{\text{LTV}} + \sigma^2 P_{\text{LTV}}^{-1})^{-1} K_{\text{LTV}}^T y_m =$$

$$\hat{h}_{\text{NLX1,RLS}}$$

In addition, the Bayesian interpretation gives an uncertainty bound as well:

$$P_{\text{a posteriori}} = P_{\text{LTV}} - P_{\text{LTV}} K_{\text{LTV}}^T (K_{\text{LTV}} P_{\text{LTV}} K_{\text{LTV}}^T + \sigma^2 I_{N \times N})^{-1} K_{\text{LTV}} P_{\text{LTV}} \quad (A.15)$$

### A.3.2 The optimal choice of the covariance hypermatrix

In this section the description of the statistical properties of the regularized time-varying estimate is given.

These properties allow us to determine the optimal choice of $P_{\text{LTV}}$. In the ideal case, the optimal choice minimizes the mean square error. In order to be able to obtain the MSE of the estimation the following computations are needed. The final result of the MSE computation is identical to the result of [102].

The regularized two dimensional IRF estimate is given by:

$$\hat{h}_{\text{NLX1,RLS}} = (K_{\text{LTV}}^T K_{\text{LTV}} + \sigma^2 P_{\text{LTV}}^{-1})^{-1} K_{\text{LTV}}^T y_m \quad (A.16)$$
The expected value of the estimation is given by:

$$
\mathbb{E}\{h_{NLX1,RLS}\} = \\
\mathbb{E}\left\{\left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} K_{LTV}^T Y_m\right\} = \\
\left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} K_{LTV}^T \mathbb{E}\{Y_m\} = \\
\left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} K_{LTV}^T K_{LTV} h_{LTV}
$$

(A.17)

The bias of the estimation is given by:

$$
bias\{h_{NLX1,RLS}\} = \mathbb{E}\{h_{NLX1,RLS}\} - h_{LTV} = \\
\left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} K_{LTV}^T K_{LTV} h_{LTV} - h_{LTV} = \\
\left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} \times \\
\left(K_{LTV}^T K_{LTV} h_{LTV} - \left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right) h_{LTV}\right) = \\
\left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} \times \\
\left(K_{LTV}^T K_{LTV} h_{LTV} - K_{LTV}^T K_{LTV} h_{LTV} - \sigma^2 P_{LTV}^{-1} h_{LTV}\right) = \\
-\sigma^2 \left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} P_{LTV}^{-1} h_{LTV}
$$

(A.18)

The squared bias is given by:

$$
bias\{h_{NLX1,RLS}\}^T bias\{h_{NLX1,RLS}\} = \\
-\sigma^2 \left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} P_{LTV}^{-1} h_{LTV} \times \\
\left(-\sigma^2 \left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} P_{LTV}^{-1} h_{LTV}\right)^T = \\
\sigma^4 \left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1} P_{LTV}^{-1} h_{LTV} \times \\
P_{LTV}^{-1} \left(K_{LTV}^T K_{LTV} + \sigma^2 P_{LTV}^{-1}\right)^{-1}
$$

(A.19)
In order to compute the variance of the estimation, the following term is needed:

\[
\begin{align*}
\hat{h}_{NL\times1,RLS} - \mathbb{E}\{\hat{h}_{NL\times1,RLS}\} &= \\
(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1}K_{LTV}^T \gamma_m - \\
(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1}K_{LTV}^T h_{LTV} &= \\
(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1}K_{LTV}^T(\gamma_m - K_{LTV} h_{LTV})
\end{align*}
\]

(A.20)

From here the second central moment (variance) of the estimate is:

\[
\begin{align*}
\text{variance}\{\hat{h}_{NL\times1,RLS}\} = \\
\mathbb{E}\{(\hat{h}_{NL\times1,RLS} - \mathbb{E}\{\hat{h}_{NL\times1,RLS}\}) (\hat{h}_{NL\times1,RLS} - \mathbb{E}\{\hat{h}_{NL\times1,RLS}\})^T\} = \\
\begin{cases}
\begin{align*}
(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1}K_{LTV}^T \times \\
\mathbb{E}_{(y_m - \gamma)(y_m - \gamma)^T = \sigma^2}
\end{align*}
\end{cases} = \\
(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1} \sigma^2 K_{LTV}^TK_{LTV}(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1}
\end{align*}
\]

(A.21)

Using Appendix A.1 and the terms above, the mean square error is given by:

\[
\begin{align*}
\text{MSE}\{\hat{h}_{NL\times1,RLS}\} = \text{variance}\{\hat{h}_{NL\times1,RLS}\} \\
+ \text{bias}\{\hat{h}_{NL\times1,RLS}\} &\text{bias}\{\hat{h}_{NL\times1,RLS}\}^T = \\
(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1} \sigma^2 K_{LTV}^TK_{LTV}(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1} + \\
\sigma^4(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1} \times \\
P_{LTV}^{-1} h_{LTV} h_{LTV}^T P_{LTV}^{-1} T \times \\
(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1} T = \\
(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1} \times \\
(\sigma^2 K_{LTV}^TK_{LTV} + \sigma^4 P_{LTV}^{-1} h_{LTV} h_{LTV}^T P_{LTV}^{-1} T) \\
(K_{LTV}^TK_{LTV} + \sigma^2 P_{LTV}^{-1})^{-1} = \\
(P_{LTV}^TK_{LTV}^T K_{LTV} + \sigma^2 I_{N\times N})^{-1} \times \\
(\sigma^2 P_{LTV}^TK_{LTV}^T K_{LTV} P_{LTV} + \sigma^4 h_{LTV} h_{LTV}^T) \\
(K_{LTV}^TK_{LTV} P_{LTV} + \sigma^2 I_{N\times N})^{-1} = \\
\sigma^4 \left(\frac{P_{LTV}^TK_{LTV}^T K_{LTV} P_{LTV}}{\sigma^2} + I_{N\times N}\right)^{-1} \times \\
\left(\frac{P_{LTV}^TK_{LTV} P_{LTV} + h_{LTV} h_{LTV}^T}{\sigma^2} + \left(\frac{K_{LTV}^TK_{LTV} P_{LTV} + I_{N\times N}}{\sigma^2}\right)^{-1}
\right)
\end{align*}
\]
The last line of eq. (A.22) allows us to use a theorem shown and proofed in [102]. Consider the following matrix equation:

\[
M(Q) = (QR + I)^{-1}(QRQ + Z)(RQ + I)^{-1}
\]  

(A.23)

where the \(Q, R, Z\) matrices are positive semi-definite, and \(I\) is an identity matrix. For all \(Q\) it can be shown that:

\[
M(Q) \preceq M(Z)
\]

In the sense that \(M(Q) - M(Z)\) is positive semidefinite.

By choosing \(R = \frac{K_{LTV}^T K_{LTV}}{\sigma^2}, Z = E\{h_{LTV} h_{LTV}^T\}\) and \(I = I_{N \times N}\):

\[
MSE\{\hat{f}_{NL \times 1, RLS}\}(Z) \leq MSE\{\hat{f}_{NL \times 1, RLS}\}(P)
\]

This means that the best choice of \(P\) for given \(h_{LTV}\) is \(Z\). That means that the optimal choice for \(P_{LTV}\) is \(E\{h_{LTV} h_{LTV}^T\}\).

A.3.3 Tuning the hyperparameters via a Bayesian view

The hyperparameters can be tuned by using the Bayesian view as well.

Let us recall the observation distribution:

\[
y_m \sim N(0, K_{LTV}^T P_{LTV} K_{LTV} + \sigma^2 I)
\]  

(A.24)

Let \(P_{LTV} = K_{LTV}^T P_{LTV} K_{LTV} + \sigma^2 I\), in this case the probability density function of the observation is given by:

\[
f(y_m) = \frac{1}{\sqrt{(2\pi)^N \det(P_{LTV})}} e^{-\frac{y_m^T P_{LTV}^{-1} y_m}{2}}
\]  

(A.25)

From here the log-likelihood function is:

\[
\ln(L) = -\frac{N}{2} \ln(2\pi) - \ln(\det(P_{LTV})) - \frac{1}{2} y_m^T P_{LTV}^{-1} y_m
\]  

(A.26)

The hyperparameters \(p_h\) influence \(P_{LTV}\) only. From here \(p_h\) can be obtained by minimizing the following expression:

\[
p_h = \arg \min_{p_h} \{y_m^T P_{LTV}^{-1} y_m + \ln(\det(P_{LTV}))\}
\]

(A.27)

This approach is also known as empirical Bayes method [100]. Special numerically stable and fast implementations for LTI cases can be found for instance in [101].
Appendix A.

A.4 Computational complexity and memory needs

A.4.1 Computational complexities

For practical purposes, it is useful to analyze the proposed methods for efficiency. This time analysis is based on [147]. In this section the symbol $O$ is used to express the time complexity of an algorithm. This gives an idea about the asymptotic running time ($T$).

In order to have an idea about the computational times, some basic complexities follow:

- The addition of two $N \times L$ matrices: $T_{\text{addition}} \sim O(NL)$.
- The inversion of an $N \times N$ squared matrix: $T_{\text{inversion}} \sim O(N^3)$.
- The multiplication of a $N \times M$ and an $M \times P$ matrix:
  $T_{\text{multiplication}} \sim (NMP)$.
- Typical optimization problem (weak estimation): $T_{\text{optimazitation}} \sim O(N^3)$.

Note, that some faster algorithm can be used as well, so the asymptotic running times above are indications only. It is also very important to remark, that on some hardware, where vector processors (for instance GPU) can be used, the algorithms might run faster than the asymptotic running times are given here (see [147]).

The most important computational complexities are:

i. Output simulation of an LTV model with a given $P_{\text{LTV}}$ using Eq. (6.9):

   $T \sim 3 \cdot O(((NL)^3) + O((NL)^2N) + O(NLN) + O(NL))$  \hspace{1cm} (A.28)

ii. Output simulation of an LTV model with a given $P_{\text{LTV}}$ using Eq. (7.15):

   $T \sim 3 \cdot O((NL)^2N) + O(N^2) + O(NLN)$  \hspace{1cm} (A.29)

iii. Create an observation hypermatrix by given set of hyperparameters using the flexible approach:

   $T_{\text{PLTV}} \sim O(2NL)$  \hspace{1cm} (A.30)

iv. Create an observation hypermatrix by given set of hyperparameters using the robust approach:

   $T_{\text{PLTV}} \sim O((NL)^2)$  \hspace{1cm} (A.31)
v. Running the optimization algorithm and the cross-validation method:

\[ T_{p_{\theta h}} \sim O((NL)^3) \cdot (T_{PLTV} + T_{\hat{\theta}}) + T_{PLTV} + T_{\hat{\theta}} \]  

(A.32)

A.4.2 Memory needs

The operational memory needs are important because the amount of the available memory puts a strict limit on the maximal length of the measurement that can be processed by the regularization technique.

To compute the regularized LTV estimate, the following operational memory needs are required to store the corresponding matrixes, in the order of decreasing memory needs:

- The prior covariance hypermatrix

\[ size(P_{LTV}) = NL \cdot NL \cdot \frac{r_{bit}}{8} \text{ [byte]} \]  

(A.33)

- The observation hypermatrix

\[ size(K_{LTV}) = N \cdot NL \cdot \frac{r_{bit}}{8} \text{ [byte]} \]  

(A.34)

- The impulse response coefficients

\[ size(h_{LTV}) = N \cdot L \cdot \frac{r_{bit}}{8} \text{ [byte]} \]  

(A.35)

Apart from these memory needs, the sizes of the datasets must be taken into account as well. In addition to that, for calculation purposes, a temporary operational memory is required as well (typically, the size of \( P_{LTV} \)).

Note, that in many mathematical programs the IEEE 754 format is used. That means that \( r_{bit} \) is either 32 or 64 bit.
A.5 Degrees of freedom

In this section the degrees of freedom (DoF) are studied. The DoF of a system of linear equations imply the number of possible solutions and therefore they are very important in the IRF estimation.

In this work it is assumed that the estimate of the LTV system is obtained from a single measurement.

A.5.1 General case

Let us observe Eq. (2.4):

\[ y[t] = \sum_{\tau=-\infty}^{+\infty} h_{LTV}[t, \tau]u[t - \tau] \]  

Assume that the finite length of the observation window is \( N \) and using Definition 2.12, the output is given by:

\[ y[t] = \sum_{\tau=0}^{N-1} h_{LTV}[t, \tau]u[t - \tau] \]  

where \( t = 0 \ldots N - 1 \).

This can be rewritten in vector form:

\[ y_m = K_{LTV_{N \times N^2}} h_{LTV_{N^2 \times 1}} \]

\( K_{LTV_{N \times N^2}} \) and \( h_{LTV_{N^2 \times 1}} \) are created in the same way as \( K_{LTV} \), \( h_{LTV_{NL \times 1}} \), see Section 6.2.1.

This is a linear inhomogeneous system. The degrees of freedom are in this particular case:

\[ \text{DoF} = \dim(h_{LTV_{N^2 \times 1}}) - \text{rank}(K_{LTV_{N \times N^2}}) \]

As it is shown for instance in the Rouché–Capelli theorem [148], the inhomogeneous linear system has a unique solution only, if

\[ \text{DoF} \leq 0 \]

Otherwise it has infinitely many solutions and DoF express the number of free variables.
The rank of $K_{LTV_{N \times N^2}}$:

$$0 \leq \text{rank} \left( K_{LTV_{N \times N^2}} \right) \leq N \quad (A.41)$$

Using Assumption 6.5 and provided that $u[0] \neq 0$, then the rank is:

$$\text{rank} \left( K_{LTV_{N \times N^2}} \right) = N \quad (A.42)$$

In this case, the number of unknown parameters are the dimensionality of $h_{LTV}$:

$$\text{dim}(h_{LTV}) = N^2 \quad (A.43)$$

So the DoF are:

$$\text{DoF} = \text{dim}(h_{LTV}) - \text{rank} \left( K_{LTV_{N \times N^2}} \right) = N^2 - N \quad (A.44)$$

That means there are infinitely many solutions, and $N^2 - N$ variables are free to choose.

### A.5.2 Realistic case

Using the previous assumptions, and Assumption 6.1, Assumption 6.2, the output is given by:

$$y[t] = \sum_{\tau=0}^{L-1} h_{LTV}[t, \tau] u[t - \tau] \quad (A.45)$$

Or in vectorial form:

$$y_m = K_{LTV} h_{LTV_{NL \times L}} \quad (A.46)$$

The rank of $K_{LTV}$ remains $N$ but the dimensionality of $h_{LTV}$ is now:

$$\text{dim}(h_{LTV}) = NL \quad (A.47)$$

Hence, the DoF are:

$$\text{DoF} = \text{dim}(h_{LTV}) - \text{rank} \left( K_{LTV_{N \times N^2}} \right) = NL - N \quad (A.48)$$

Although the DoF are smaller, there are still infinitely many solutions.
A.5.3 Special case

Let us revise the Eq. (6.9), and assume that elements before $t - \tau < 0$ are zeros (instead of unidentifiable), then $H_{L\times N}$ is:

$$H_{L\times N} = \begin{bmatrix} h[0,0] & h[1,0] & \cdots & h[N-1,0] \\ 0 & h[1,1] & \vdots & \vdots \\ \vdots & 0 & \vdots & \vdots \\ 0 & \cdots & \cdots & h[N-1,L-1] \end{bmatrix}$$

This indirectly reduces the number of unknowns in $h_{LT\nu}$ such that

$$\dim(h_{LT\nu}) - \frac{(N-1)L}{2} = NL - \frac{(N-1)L}{2}$$

Hence, the DoF are:

$$\text{DoF} = \dim(h_{LT\nu}) - \frac{(N-1)L}{2} - \text{rank} \left( K_{LT\nu_{N\times N^2}} \right) = NL - \frac{(N-1)L}{2} - N > 0$$

The DoF are still larger than zero, and there are still infinitely many solutions.

It leads to the following conclusion: none of the possible combinations of the regressor, regressand and the parameter vector in the least squares cost function can provide a unique solution.

A.5.4 Regularized case

Let us observe the estimation case, when the regularization is used. Consider the cost function from Eq. (6.8):

$$V_{LT\nu} = V_{LT\nu, LS} + V_{LT\nu, R} = \|y_m - K_{LT\nu}h_{NL\times 1}\|_2^2 + \sigma^2 h_{LT\nu_{NL\times 1}}^T P_{LT\nu}^{-1} h_{LT\nu_{NL\times 1}}$$

Next to the ordinary LS cost function, there is an additional quadratic term which provides constraints to the inhomogeneous system.

Assume that the inverse of $P_{LT\nu}$ in Eq. (6.8) exists, then:

$$\text{rank}(P_{LT\nu}^{-1}) = \text{rank}(P_{LT\nu}) = NL$$
In this particular case, the Hessian of the cost function given by Eq. (6.8) needs to be used:

\[
\frac{\partial^2 V_{LTV}}{\partial h_{LTV}^{T} \partial h_{LTV}^{T}} = 2K_{LTV}^{T}K_{LTV} + 2\sigma^2 P_{LTV}^{-1}
\] (A.54)

The rank of the $K_{LTV}^{T}K_{LTV}$ term is:

\[
\text{rank}(K_{LTV}^{T}K_{LTV}) = \text{rank}(K_{LTV}^{T}K_{LTV}) = \text{rank}(K_{LTV}) = N
\] (A.55)

Let us observe the joint case – the rank is invariant to a non-zero scalar multiplication $(2, 2\sigma^2)$ –:

\[
1 \leq \text{rank}(K_{LTV}^{T}K_{LTV} + P_{LTV}^{-1}) \leq \text{rank}(K_{LTV}^{T}K_{LTV}) + \text{rank}(P_{LTV}^{-1}) = NL
\] (A.56)

The lower bound is one, because using the fact that they are independent, the least rank is one (only zero matrix has zero rank). Further, the rank is limited to the maximum dimensionality, in this case to NL.

For the continuation, it is important to recall a notion. A $n \times n$ real matrix $M$ is said to be positive definite, for every non-zero column vector $z$ of $n$ real numbers, if and only if:

\[
z^{T}Mz > 0
\] (A.57)

Matrix $M$ is said to be positive semidefinite, if and only if:

\[
z^{T}Mz \geq 0
\] (A.58)

Let us analyze the matrices.

The $K_{LTV}^{T}K_{LTV}$ is positive semidefinite, because for every $n \times m$ matrix $A$ it is true that $A^{T}A$ and $AA^{T}$ are positive semidefinite [149].

Next, $P_{LTV}^{-1}$ is positive definite matrix, because by definition a covariance matrix is positive semidefinite and since the inverse exists, it must be a positive definite matrix.
Let us calculate the definiteness of the sum:

\[
z^T \left( 2K_{LT}^T K_{LT} + 2\sigma^2 P_{LT}^{-1} \right) z = \\
\begin{array}{c}
\geq 0 \\
\end{array}
\begin{array}{c}
> 0 \\
\end{array}
\]

This allows us to conclude that sum of these two matrices is positive definite. A positive definite matrix has the property to have only positive eigenvalues. Therefore this is a full rank matrix.

So the degrees of freedom are:

\[
\text{DoF} = \dim(h_{LT}) - \text{rank}(K_{LT}^T K_{LT} + P_{LT}^{-1}) = \\
NL - NL = 0
\]

Because the degrees of freedom are zero and the Hessian is positive, a unique solution exists. The shape and the properties of the solution depend upon the covariance hypermatrix \( P_{LT} \) as it is shown in Section 5.3.

It is important to remark that these degrees of freedom differ from the effective (statistical) degrees of freedom of a smoother discussed in [150].
A.6 Computer configuration used for calculating the estimations

In the table below one can find the computer configuration used for calculating the simulations and for processing the data of the measurement given in Chapter 11.

Table A.1: The computer configuration used for the intensive calculation

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>Windows 7 Professional 64-bit (6.1, Build 7601) Service Pack 1</td>
</tr>
<tr>
<td>Matlab version</td>
<td>R2014a</td>
</tr>
<tr>
<td>Computer model</td>
<td>Dell OptiPlex 990, bios version: A11</td>
</tr>
<tr>
<td>Processor</td>
<td>Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz (8 threads, 4 cores)</td>
</tr>
<tr>
<td>Operating memory</td>
<td>16 GB DDR3 @ 1600 MHz</td>
</tr>
<tr>
<td>GPU</td>
<td>AMD Radeon HD 6450</td>
</tr>
<tr>
<td>HDD</td>
<td>2 × Seagate ST95005620AS 500 GB @ 7200 RPM, 32 MB cache in raid 1 mode</td>
</tr>
</tbody>
</table>
Appendix B.

B.1 Derivation of the B-splines time-varying least squares estimation from the cost function

In this section, the derivation of the solution given by Eq. (9.12) as follows:

\[
\tilde{C}_{\text{LTV, BS}} = \left( K_{\text{LTV, BS}}^T K_{\text{LTV, BS}} \right)^{-1} K_{\text{LTV, BS}}^T y_m
\]  

(B.1)

where \( K_{\text{LTV, BS}} \) is the B-spline observation matrix discussed in Section 9.1.

The B-spline LTV cost function is shown in Eq. (9.11) as follows:

\[
V_{\text{LTV, BS}} = \| y_m - \bar{y} \|^2_2 = \| y_m - K_{\text{LTV, BS}} C_{\text{LTV, BS}} \|^2_2
\]  

(B.2)

This cost function has to be minimized such that:

\[
\arg\min_{C_{\text{LTV, BS}}} V_{\text{LTV, BS}} = \arg\min_{C_{\text{LTV, BS}}} \left\{ \| y_m - K_{\text{LTV, BS}} C_{\text{LTV, BS}} \|^2_2 \right\}
\]  

(B.3)

The minimum of argmin is given by:

\[
\frac{\partial}{\partial C_{\text{LTV, BS}}} \left\{ \| y_m - K_{\text{LTV, BS}} C_{\text{LTV, BS}} \|^2_2 \right\} = 0
\]  

(B.4)

And from here, the estimation is given by:

\[
\tilde{C}_{\text{LTV, BS}} = \left( K_{\text{LTV, BS}}^T K_{\text{LTV, BS}} \right)^{-1} K_{\text{LTV, BS}}^T y_m
\]  

(B.5)
Appendix B.

B.2 Derivation of the B-spline surface fitting cost function (complex case)

In this section the derivation of the solution given by Eq. (10.23) is shown. Eq. (10.23) is given by:

\[
C_{ij} = \arg\min_{c_{ij}} V_{BS}^{real} + i \arg\min_{c_{ij}} V_{BS}^{imaginary} = \left( B_{d_{t}}^{T}B_{d_{t}} \right)^{-1} B_{d_{t}}^{T} \Re \{ \hat{g}_{FTF} \} \left( B_{d_{m}}^{T}B_{d_{m}} \right)^{-1} B_{d_{m}}^{T} + \left( B_{d_{t}}^{T}B_{d_{t}} \right)^{-1} B_{d_{t}}^{T} \Im \{ \hat{g}_{FTF} \} \left( B_{d_{m}}^{T}B_{d_{m}} \right)^{-1} B_{d_{m}}^{T} \tag{B.6}
\]

where \( V_{BS}^{real} \) is given by Eq. (10.21):

\[
V_{BS}^{real} = \left\| \Re \{ \hat{g}_{FTF} \} - \Re \{ \hat{g}_{FTF} \} \right\|_{2}^{2} = \left\| \Re \{ \hat{g}_{FTF} \} - B_{d_{t}} \left( \Re \{ c_{ij} \} \right) B_{d_{m}} \right\|_{2}^{2} \tag{B.7}
\]

and \( V_{BS}^{imaginary} \) is given by Eq. (10.22).

To simplify, the derivation of the real part of the cumulative cost function is shown only. The steps are the same for the imaginary part.

The cost function given by Eq. (10.21) has to be minimized such that:

\[
\arg\min_{\Re \{ c_{ij} \}} V_{BS}^{real} = \arg\min_{c_{LTVBS}} \left\{ \left\| \Re \{ \hat{g}_{FTF} \} - B_{d_{t}} \left( \Re \{ c_{ij} \} \right) B_{d_{m}} \right\|_{2}^{2} \right\} \tag{B.8}
\]

In the further steps the dimensionalities will be shown in the subscripts of the matrices. The minimum of Eq. (10.21) is given by:

\[
\frac{\partial}{\partial \Re \{ c_{ij} \}} \left\{ \left\| \Re \{ \hat{g}_{FTF} \}_{N \times N_{w}} - B_{d_{t}}^{T} \left( \Re \{ c_{ij} \}_{N_{t} \times N_{t}} \right) B_{d_{m}}^{N_{t} \times N_{w}} \right\|_{2}^{2} \right\} = 0
\]

\[
\frac{\partial}{\partial \Re \{ c_{ij} \}} \left\{ \left( \Re \{ \hat{g}_{FTF} \}_{N \times N_{w}} - B_{N \times N_{t}} \left( \Re \{ c_{ij} \}_{N_{t} \times N_{t}} \right) B_{d_{m}}^{N_{t} \times N_{w}} \right)^{T} \right. \times \left( \Re \{ \hat{g}_{FTF} \}_{N \times N_{w}} - B_{d_{t}}^{N \times N_{w}} \left( \Re \{ c_{ij} \}_{N_{t} \times N_{t}} \right) B_{d_{m}}^{N_{t} \times N_{w}} \right) \right\} = 0 \tag{B.9}
\]

\[
\frac{\partial}{\partial \Re \{ c_{ij} \}} \left\{ \Re \{ \hat{g}_{FTF} \}_{N \times N_{w}}^{T} \Re \{ \hat{g}_{FTF} \}_{N \times N_{w}} - \Re \{ \hat{g}_{FTF} \}_{N \times N_{w}}^{T} B_{N \times N_{t}} \left( \Re \{ c_{ij} \}_{N_{t} \times N_{t}} \right) B_{d_{m}}^{N_{t} \times N_{w}} - \left( B_{d_{t}}^{N \times N_{t}} \left( \Re \{ c_{ij} \}_{N_{t} \times N_{t}} \right) B_{d_{m}}^{N_{t} \times N_{w}} \right)^{T} \Re \{ \hat{g}_{FTF} \}_{N \times N_{w}} + \left( B_{d_{t}}^{N \times N_{t}} \left( \Re \{ c_{ij} \}_{N_{t} \times N_{t}} \right) B_{d_{m}}^{N_{t} \times N_{w}} \right)^{T} \left( B_{d_{t}}^{N \times N_{t}} \left( \Re \{ c_{ij} \}_{N_{t} \times N_{t}} \right) B_{d_{m}}^{N_{t} \times N_{w}} \right) \right\} = 0
\]

184
The computations with the derivatives are as follows:

\[
-\Re\{\mathcal{G}_{\text{FTF}}\}_{N \times N_w}^T B_{N \times n_t}^{d_t} B_{n_t \times N_w}^{d_m} \nonumber
- \left( (B_{N \times n_t}^{d_t} B_{n_t \times N_w}^{d_m})^T \Re\{\mathcal{G}_{\text{FTF}}\}_{N \times N_w} \right)^T \nonumber

+ 2 \left( B_{N \times n_t}^{d_t} \Re\{c_{ij}\}_{n_t \times n_t}^T B_{n_t \times N_w}^{d_m} \right)^T (B_{N \times n_t}^{d_t} B_{n_t \times N_w}^{d_m}) = 0 \nonumber

-2 \Re\{\mathcal{G}_{\text{FTF}}\}_{N \times N_w}^T B_{N \times n_t}^{d_t} B_{n_t \times N_w}^{d_m} \nonumber
+ 2 B_{n_t \times N_w}^{d_m} \Re\{c_{ij}\}_{n_t \times n_t}^T B_{N \times n_t}^{d_t} B_{n_t \times N_w}^{d_m} = 0
\] 

\[
B_{n_t \times N_w}^{d_m} \Re\{c_{ij}\}_{n_t \times n_t}^T B_{N \times n_t}^{d_t} = \Re\{\mathcal{G}_{\text{FTF}}\}_{N \times N_w}^T
\nonumber
\]

\[
B_{N \times n_t}^{d_t} \Re\{c_{ij}\}_{n_t \times n_t}^T B_{n_t \times N_w}^{d_m} = \Re\{\mathcal{G}_{\text{FTF}}\}_{N \times N_w}
\] 

And from here, using the pseudo inverse, the estimation is given by:

\[
\Re\{c_{ij}\}_{n_t \times n_t} = \nonumber
(B_{N \times n_t}^{d_t} B_{N \times n_t}^{d_t})^{-1} B_{N \times n_t}^{d_t} \Re\{\mathcal{G}_{\text{FTF}}\}_{N \times N_w} (B_{n_t \times N_w}^{d_m} B_{n_t \times N_w}^{d_m})^{-1} B_{n_t \times N_w}^{d_m}
\]
B.3 Statistical properties of the proposed two dimensional non-frozen B-spline LTV estimator

The base-line model is given by Eq. (9.1) such that:

\[ y_m = y[t] + e[t] = \sum_{\tau=0}^{1} h_{LTV}[t, \tau] u[t - \tau] + e[t] \]  

(B.12)

The two dimensional LTV IRF is given by:

\[ h_{LTV}[t, \tau] = \sum_{i=0}^{n_t-1} \sum_{j=0}^{n_r-1} c_{ij} B_i^d(t) B_j^d(\tau) \]  

(B.13)

This can be rewritten in a vectorial form as follows:

\[ \hat{h}_{LTV} = (B^{dt}_{NL \times n_t \times n_t} B^{dt}_{n_r \times n_r \times n_t}) (B^{dt}_{NL \times n_t \times n_t} B^{dt}_{n_r \times n_r \times n_t}) C_{LTV,BS} \]  

(B.14)

The estimated output is now given by:

\[ \hat{y} = K_{LTV} (B^{dt}_{NL \times n_t \times n_t} B^{dt}_{n_r \times n_r \times n_t}) C_{LTV,BS} \]  

(B.15)

This can be rewritten as:

\[ \hat{y} = K_{LTV,BS} C_{LTV,BS} \]  

(B.16)

where \( K_{LTV,BS} \) contains all the deterministic variables (the two B-spline basis functions and the excitation signal, as given in Section 9.1).

This means that the two dimensional IRF estimate is given by the B-spline kernel such that:

\[ \hat{C}_{LTV,BS} = (K_{LTV,BS}^T K_{LTV,BS})^{-1} K_{LTV,BS}^T y_m \]  

(B.17)

The expected value of the estimation is given by:

\[ \mathbb{E}\{\hat{C}_{LTV,BS}\} = \mathbb{E}\{(K_{LTV,BS}^T K_{LTV,BS})^{-1} K_{LTV,BS}^T y_m\} = \]  

\[ (K_{LTV,BS}^T K_{LTV,BS})^{-1} K_{LTV,BS} \mathbb{E}\{y_m\} = \]  

\[ \left( K_{LTV,BS}^T K_{LTV,BS} \right)^{-1} K_{LTV,BS}^T y = \]  

\[ \left( K_{LTV,BS}^T K_{LTV,BS} \right)^{-1} K_{LTV,BS}^T K_{LTV,BS} C_{LTV,BS} = C_{LTV,BS} \]  

(B.18)
The bias of the estimation is given by:

\[
\text{bias}\{\mathbf{C}_{\text{LTV,BS}}\} = \mathbb{E}\{\mathbf{C}_{\text{LTV,BS}}\} - \mathbf{C}_{\text{LTV,BS}} = \frac{1}{I_{r,t} \times r \times I_{r,t}} (\mathbf{K}_{\text{LTV,BS}}^T \mathbf{K}_{\text{LTV,BS}})^{-1} \mathbf{K}_{\text{LTV,BS}}^T \mathbf{C}_{\text{LTV,BS}} - \mathbf{C}_{\text{LTV,BS}} = 0
\]

(B.19)

The squared bias is given by:

\[
\text{bias}\{\mathbf{h}_{\text{NLx1,RLS}}\} \text{bias}\{\mathbf{h}_{\text{NLx1,RLS}}\}^T = 0
\]

(B.20)

In order to compute the variance of the estimation, the following term is needed:

\[
\tilde{\mathbf{C}}_{\text{LTV,BS}} - \mathbb{E}\{\tilde{\mathbf{C}}_{\text{LTV,BS}}\} =
\begin{align*}
(\mathbf{K}_{\text{LTV,BS}}^T \mathbf{K}_{\text{LTV,BS}})^{-1} \mathbf{K}_{\text{LTV,BS}}^T \mathbf{y}_m - \\
(\mathbf{K}_{\text{LTV,BS}}^T \mathbf{K}_{\text{LTV,BS}})^{-1} \mathbf{K}_{\text{LTV,BS}}^T \mathbf{y} = \\
(\mathbf{K}_{\text{LTV,BS}}^T \mathbf{K}_{\text{LTV,BS}})^{-1} \mathbf{K}_{\text{LTV,BS}}^T (\mathbf{y}_m - \mathbf{y})
\end{align*}
\]

(B.21)

The variance of the estimate is given by:

\[
\text{variance}\{\tilde{\mathbf{C}}_{\text{LTV,BS}}\} = 
\begin{align*}
\mathbb{E}\left\{\left(\mathbf{C}_{\text{LTV,BS}} - \mathbb{E}\{\mathbf{C}_{\text{LTV,BS}}\}\right)\left(\mathbf{C}_{\text{LTV,BS}} - \mathbb{E}\{\mathbf{C}_{\text{LTV,BS}}\}\right)^T\right\} = \\
\mathbb{E}\left\{ \left(\mathbf{K}_{\text{LTV,BS}}^T \mathbf{K}_{\text{LTV,BS}}\right)^{-1} \mathbf{K}_{\text{LTV,BS}}^T (\mathbf{y}_m - \mathbf{y}) \times \\
\frac{\sigma^2}{\mathbf{K}_{\text{LTV,BS}} (\mathbf{K}_{\text{LTV,BS}}^T \mathbf{K}_{\text{LTV,BS}})^{-1}} \times \\
\mathbf{K}_{\text{LTV,BS}} (\mathbf{K}_{\text{LTV,BS}}^T \mathbf{K}_{\text{LTV,BS}})^{-1} \right\} = \\
\sigma^2 (\mathbf{K}_{\text{LTV,BS}}^T \mathbf{K}_{\text{LTV,BS}})^{-1}
\end{align*}
\]

(B.22)

Using Chapter A.1 and the terms above, the mean square error is given by:

\[
\text{MSE}\{\tilde{\mathbf{C}}_{\text{LTV,BS}}\} = \text{variance}\{\tilde{\mathbf{C}}_{\text{LTV,BS}}\} + \text{bias}\{\mathbf{h}_{\text{NLx1,RLS}}\} \text{bias}\{\mathbf{h}_{\text{NLx1,RLS}}\}^T = \\
\sigma^2 (\mathbf{K}_{\text{LTV,BS}}^T \mathbf{K}_{\text{LTV,BS}})^{-1}
\]

(B.23)
Appendix B.

B.4 Computational complexity and memory needs

B.4.1 Computational complexities

In this Section the computational needs of the non-frozen B-spline linear time-varying identification method can be found. This appendix is based on notations of the Appendix A.4.

i. Output simulation of an LTV model with a given \( \bar{C}_{LTV,BS} \) using Eq. (9.5):

\[
T_y \sim O(N \cdot n_t n_r)
\]

(B.24)

ii. Create an observation matrix by given set of hyperparameters:

\[
T_{K_{LTV,BS}} \sim O(N \cdot (n_t)^{d_t} (n_r)^{d_r})
\]

(B.25)

iii. Running the optimization algorithm and the cross-validation method:

\[
T_{\bar{c}_{LTV,BS}} \sim O((NL)^3) \cdot (T_{PLTV} + T_y) + T_{PLTV} + T_y
\]

(B.26)

B.4.2 Memory needs

- The B-spline kernel vector used to describe a LTV system has the following size:

\[
\text{size}(\bar{C}_{LTV,BS}) = n_t n_r \cdot \frac{r_{bit}}{8} \ [byte]
\]

(B.27)

- The observation matrix belonging to the non-frozen B-spline kernel approach:

\[
\text{size}(K_{LTV,BS}) = N \cdot n_t n_r \cdot \frac{r_{bit}}{8} \ [byte]
\]

(B.28)
B.5 Degrees of freedom

Let us observe the estimation case, when the two dimensional B-spline kernel approach is used to estimate LTV systems. Consider the model equation given by Eq. (9.5):

\[
\tilde{y} = K_{LTV,BS}C_{LTV,BS}
\]  \hspace{1cm} (B.29)

In this particular case instead of a large \( h_{LTV} \) parameter vector, we estimate a model which has significantly smaller number of parameters.

Then the number of parameters to be estimated can be found in the dimensionality of the model:

\[
dim(C_{LTV,BS}) = n_t \cdot n_r < N
\]  \hspace{1cm} (B.30)

The rank of the observation matrix is the minimum of its dimensionality, i.e.:

\[
\text{rank}(K_{LTV,BS}) = \min(n_t \cdot n_r, N) = n_t \cdot n_r
\]  \hspace{1cm} (B.31)

So the degrees of freedom are:

\[
\text{DoF} = \dim(C_{LTV,BS}) - \text{rank}(K_{LTV,BS}) = 0
\]  \hspace{1cm} (B.32)

Because the DoF are zero, a unique solution exists. The shape and the properties of the solution depend upon the degrees of splines, the knot sets as it is shown in Chapter 9.
Appendix B.

B.6 Penalized B-splines

B.6.1 Introduction

In this section some practical properties of B-splines are discussed. To simplify the explanation, the one dimensional case is considered here. These properties and statements are also valid for two dimensional cases [151].

In typical identification scenarios we are more interested in specially determined B-splines, in the so-called P-splines (penalized B-splines) [116]. Using P-splines, for instance we are able to prescribe a certain level of smoothness and/or interpolation.

B.6.2 The cost function from the viewpoint of identification

For P-splines, the coefficients are determined partly by the data to be fitted, and partly by an additional penalty term on smoothness. Therefore, the task and the cost function will be very similar to the LTI regularization cost function. The cost function $V_{PS}$ (using the notations from Section 8.1) is given by:

$$V_{PS} = \sum_{t=0}^{N-1} \left( s(x_t) - \hat{s}(x_t) \right)^2 + V_{\text{smoothing}}$$

$$= \sum_{t=0}^{N-1} \left( s(x_t) - \sum_{i=0}^{n-1} c_i B_i^d(x_t) \right)^2 + V_{\text{smoothing}}$$

where the smooth estimate of $s(x)$ is $\hat{s}(x)$ and $V_{\text{smoothing}}$ is an additional smoothing penalty.

There are several proposals in the literature for choosing the smoothing penalty term $V_{\text{smoothing}}$ such as the most popular Eilers and Marx penalty proposal [152] based on finite differences. The idea of these penalties is to avoid overfitting [116], [152], [122]. In this work this penalty term is not necessary because a cross-validation method is used to fit the splines on the data. Therefore the cost function boils down to a simple LS cost function as follows:

$$V'_{PS} = \sum_{t=0}^{N-1} \left( s(x_t) - \sum_{i=0}^{n-1} c_i B_i^d(x_t) \right)^2$$

This form of P-splines is called regression (rational) B-splines. There are several advantages to take into account: the boundary effects (errors at the borders of the observation window) are negligible [122], it keeps the moments (for instance, the mean and variances) and a cross validation technique can be used. Further properties are valid for these regression B-splines only. The general P-spline cases are well studied in [152], [122].
B.6.3 Bias and variance trade-off

By minimizing fitting error we can mean two different objectives. First one is the situation, when we want to interpolate some given points (this viewpoint is mainly used in statistics and computer graphics). This viewpoint is used in Chapter 11 (the frozen LTV B-spline approach). The second objective, in the case of classical identification, we can understand as the output fitting error minimization, i.e. the model should be able to predict the system output well. This approach is used in Chapter 10.

When (the regression) B-splines are used to interpolate certain data points, it can be seen as an LS estimator with the following well-known bias and variance properties:

\[
\text{bias}\{\hat{s}\} = \mathbb{E}\{\hat{s}(x_t)\} - s(x_t) = 0 \tag{B.35}
\]

\[
\text{variance}\{\hat{s}\} = \sigma^2 \left( B_x^{d_x+1} B_x^d \right)^{-1} \tag{B.36}
\]

It is important to remark that this unbiasedness (and the level of variance) holds only, when enough (if not all) control points are used. Therefore, in practice, where only some points are used, it is difficult to predict the exact levels of the bias and variance because they depend not only on the degree but also on the density and positions of the control points. For example, there is an influence of both the degree of the polynomial and the number of knots on the quality of smoothing and on the fitting error.

For the case, when (significantly) fewer control points are used than data points available, a good approximation – with some modification – can be found in [122], [153]. The detailed description is out of the scope of this work. Based on these works, the bias and the variance are given by:

\[
\text{bias}\{\hat{s}\} \approx b_a(x) + O\left( \Delta_x^{d_x+1} \right) \tag{B.37}
\]

\[
\text{var}(\text{bias}\{\hat{s}\}) = \approx O\left( \frac{\sigma^2}{N} \Delta_y^{d_y+1} + \frac{1}{N \Delta_x^{d_x+1}} \right) \tag{B.38}
\]

where \( b_a(x) \) is the approximation bias defined as follows:

\[
b_a(x) = \frac{s^{(d_x+1)}(x)}{(d_x+1)!} \Delta_x^{d_x+1} \Delta_{x_t}^{d_x+1} \text{Bernoulli}_{d_x}(\frac{x-x_t}{\Delta_x}) \tag{B.39}
\]
and $B_{\text{ernoulli}}_d(i)$ is a Bernoulli polynomial of degree $d$ at $i$ [32]. $\Delta_x$ is the distance between two used knots and $\Delta_y$ is the (vertical) distance between two control points.

The density of the knots can be expressed as the distance between two points compared to original the measurement grid. In this work $\Delta_x$ is called step size. It defines the frequency of the used data points. It is important because not all the grid points are needed and for that reason a truncated dataset can be used. If the step size is chosen as unity then it means that all the control and knot points are used to describe the curve. For instance step size ten means that only each tenth data point (control and knot point) is used.

### B.6.4 Example

To illustrate the above-mentioned statements in this section, a simulation example is shown. The observed impulse response function is shown in Figure B.2. This is a second order low-pass filter which contains 256 samples. The B-spline fitting is done in the time domain.

When fitting, we do not have direct influence to $\Delta_y$ (vertical density of the control points). The tunable parameters are the step size and the degree of splines. Figure B.3 shows the fitting errors as a function of degree/step size for different noise levels. It can be observed that for noiseless observations the error strongly depends on the step size and on the degree of splines.

For the noisy cases, it can be seen that the higher the order of the polynomial is, the lower the error. An intuitive graphical explanation is that sometimes the points to be estimated are higher or lower than the true signal value and due to the smoothing-averaging properties, spline fitting leads to a lower error. This shows that the optimal tuning of the spline parameters will depend upon the SNR of the measurement data.
Figure B.3: Fitting error at different degrees, step sizes and noise levels.
While Figure B.3 shows an overview about the error level for varying SNR/step size/degrees, Figure B.4 shows a selected case with three different scenarios at an SNR level of 40 dB. Figure B.4 a) shows the case, when all the data points are used to linearly interpolate. Figure B.4 b) shows the case, when every fifth point is used with second order splines. Figure B.4 c) shows the case, when every fifth point is used with fourth order splines. Observe that using a higher degree of splines with a larger step size leads to a better fit, than using all the data points (and linear interpolation).

Figure B.4: Fitting error at different degrees and step sizes at 40 dB SNR. The black line shows the impulse response function to be estimated (interpolated). The gray line shows the estimated IRF. The crosses (×) are the differences between the true and the estimated values.


Bibliography


Bibliography


