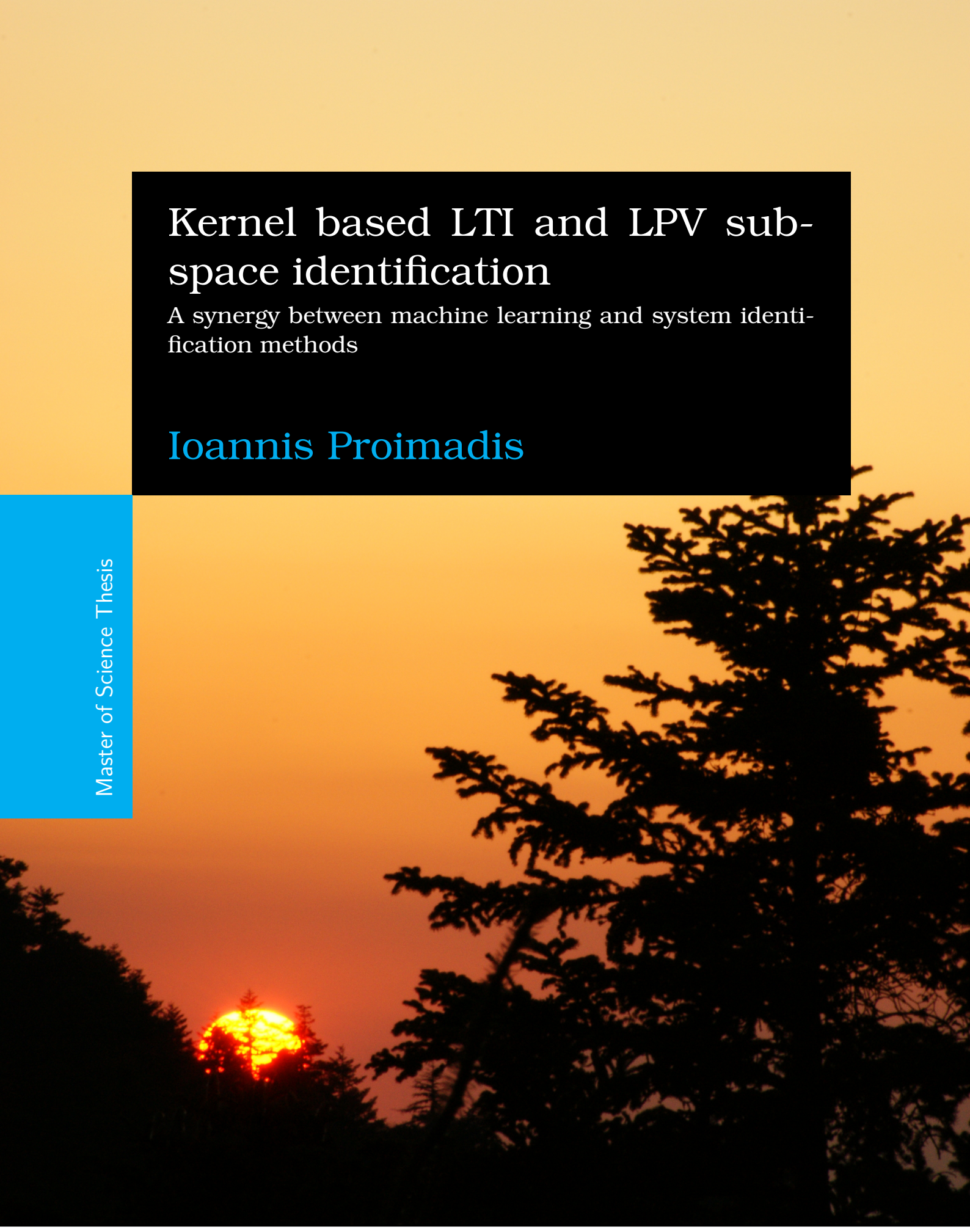
Kernel based LTI and LPV subspace identification

A synergy between machine learning and system identification methods

Ioannis Proimadis

Master of Science Thesis



TU Delft Delft University of Technology

Delft Center for Systems and Control

Kernel based LTI and LPV subspace identification

A synergy between machine learning and system identification methods

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Ioannis Proimadis

May 27, 2015

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Abstract

System identification is the art of constructing mathematical models from observed data. It is a well established field with a history of over 40 years, characterized by a rich theoretical background, while it has proven its worth in many real life applications. On the other hand, Gaussian processes form a specific category of kernel based machine learning algorithms. Both methods aim at making predictions based on the past data. However, in contrast with the system identification algorithms, Gaussian processes do not deliver a parametric model but a mere (non-parametric) relation between the available inputs and outputs.

Over the past few years the possible synergy between the two fields is extensively investigated. More specifically, the incorporation of the Gaussian process framework in the Prediction Error Identification (PEI) methods for Linear Time Invariant (LTI) systems was recently achieved. In this way, desirable properties of the Gaussian processes such as the increased flexibility and the minimum variance property of the estimator were included in the PEI framework. Moreover, these methods manage to incorporate simple prior knowledge to the algorithm through the sophisticated determination of the covariance (kernel) properties of the related coefficients.

This synergy was also recently extended to the Subspace Identification (SID) framework for LTI systems. This is exactly the starting point of this thesis. After this point, we analyse the effect of various aspects on this new algorithm, such as the kernel structure, the effect of Signal-to-Noise Ratio (SNR) ratio and the effect of the available data points. More importantly, the effect of the past window value is extensively investigated, since its value is critical towards the accurate identification in the classical SID methods. In this thesis it is shown that the kernel based SID methods exhibit a superior accuracy compared to the up-to-date SID algorithms. Moreover, it is shown that they are the least affected by the choice of the past window value, thus opening the way for more automatic methods, less affected by the specific choices of the users.

Following the examination of the LTI case, the possible synergy between Gaussian processes and SID methods for Linear Parameter Varying (LPV) systems is under consideration. To this end, we start with an analytic investigation of the LPV SID methods in order to highlight their characteristics. At this point it becomes obvious that a direct use of the kernel methods

for LTI systems is impossible due to the differences between the two classes of systems. Therefore, new methods were sought, opening the way to two novel approaches for the kernel based SID of LPV systems. The first one is based on the introduction of a prior on the LPV equivalent Markov parameters, while the second one introduces a prior on the time varying impulse response coefficients. The theoretical aspects of the proposed algorithms are then highlighted to reveal their merits and deficiencies.

Moreover, the structure of the kernels is a crucial aspect of the kernel based SID methods for LPV systems. Especially for the second proposed approach, the proposed kernels balance between two desirable but contradictory characteristics. On the one hand, simple structures alleviate the computational burden of the involved (non-convex) optimization algorithms but they can be too restrictive and so they may fail to capture the underlying dynamics. On the other hand, rich kernel structures are expected to offer better results but only if they manage to avoid local-minima, while the computational time is expected to be a serious limitation. Our solution follows after an assiduous investigation of the coefficients to be estimated and the subsequent establishment of a correlation between the kernel structure and the impulse response coefficients. This could be seen as the LPV equivalent of introducing simple prior knowledge in the proposed methods.

Finally, the validity of the proposed algorithms is verified through a series of identification examples. The main result of this thesis project is that the new, kernel based algorithms show a superior accuracy compared to the standard SID methods for LPV systems. From these algorithms, the so called "LPV-RKHS-PBSID$_{opt}$" algorithm exhibits the most accurate results, as it is both theoretically justified and also observed in the simulation examples. All in all, the performance of the kernel based SID methods for LPV systems shows a high potential, which can lead to a change of paradigm of how mathematical models can be constructed from the observed data.

Contents

Acknowledgements

This report concludes my work as an MSc student in the Technical University of Delft. When I came here, it was pretty unclear to me what possible horizons would open to my knowledge. After some years, this journey has come to an end. Hopefully, it was a interesting one. And for this I have to express my gratitude to my professors in DCSC, since their contribution in defining my research interests and keeping my motivation high was crucial.

I would like to express my gratitude especially to my supervisor, dr. ir. J.W. van Wingerden, for his assistance during this MSc thesis project. His idea of addressing a state-of-the-art approach in system identification defined a very challenging and intriguing topic that completely matched my expectations. Moreover, I would like to thank him for giving me the opportunity to work as a student assistant in introduction project as well as Prof. Babuska for also giving me the opportunity to work as a student assistant in integration project. These two experiences were really valuable since they gave me the joy of interacting with my fellow students and future colleagues, to assist and also learn from them.

Concerning the work in my final MSc project, I would also like to thank dr.ir. Roland Toth for his helpful remarks and the interesting discussions that we had. His deep insight into the LPV systems was really valuable towards the progress of my work. For the same reason I would also like to thank Prof. Verhaegen for the meeting that we had. The discussions that I had with Hildo Bijl were also very helpful and so I would also like to thank him.

I would also like to express my thankfulness to all my friends and fellow students. Especially my fellow students and friends Arman, Harsh, Nikos, Paolo and Valentin. Last but not least, special thanks go to all my friends here in Delft, for all these joyful moments that we shared. And also to my friends Anna, Ilias and Vangelis, with whom I spent plenty of my time in the Netherlands and in Belgium, as well as my friends in Greece, London, Berlin and Zurich. For my friends who are dreaming and building a better tomorrow, you have to know that I admire and respect you.

Finally, this work is dedicated to my family for their support during all these years. Without them, I would not fulfil any of my aspirations.

Delft, University of Technology — Ioannis Proimadis
May 27, 2015

“ ἓν οἶδα ὅτι οὐδὲν οἶδα”

— *Σωκράτης*

Chapter 1

Introduction

1-1 Introduction to the identification of dynamical systems

System identification is the art of constructing **mathematical models** from observed data. It has a long history that spans more than 40 years in its contemporary form. Its origins, though, lay back on the work of Legendre and Gauss in the 19th century, when both used the least squares method during their independent efforts to understand and predict the motion of the planets and comets [1]. The theory of system identification defines an alternative approach to the analytic modelling, in which the underlying mathematical model is based on physical intuition. Nowadays, it is a well defined scientific field that has proven its merits in a large spectrum of applications.

The two most prominent identification schemes are the so called Prediction Error Identification (PEI) and the Subspace Identification (SID) methods. While the first ones include an explicit (possibly nonlinear) optimization criterion whose minimization delivers the most suited system variables, the SID methods follow a different scheme. More specifically, by incorporating tools from linear algebra, they are also capable of estimating models with high accuracy without requiring an **explicit** optimization criterion, thus avoiding the pitfalls of a possibly nonlinear optimization problem.

Both identification techniques have been successfully applied in the Linear Time Invariant (LTI) class of systems. However, this class is not capable of capturing the dynamics of more complex systems, that may exhibit a nonlinear behaviour. On the other hand, nonlinear models are able to interpret a larger variety of systems but the high complexity of these models creates many problems when it comes to their identification and control and so it has restricted their applicability. For this reason, the class of Linear Parameter Varying (LPV) systems was proposed at the beginning of the '90s as the middle ground between LTI and nonlinear systems. Following the development of the related theory, the identification of the LPV systems based on both identification techniques was addressed in a series of publications [2, 3].

Another popular scientific field is machine learning. In simple terms, machine learning is trying to answer the question of how a "machine" can be programmed to automatically learn

and improve its performance based on the experience that it gains over time. This learning procedure also includes the task of predicting the future behaviour of the machine, based on the past paradigms. In contrast with the system identification techniques, the machine learning techniques do not deliver an explicit mathematical model to describe the underlying system, but instead they establish relations between the given data points. Its origins are traced far beyond the advent of digital computers. The first known analogue computer was the Antikythera mechanism, used to predict celestial information such as the phases of the moon [4]. Gaussian processes are a specific way to perform learning tasks and it is based on the Gaussian process statistical framework [5]. The Gaussian processes encode the information about the statistical properties (such as covariance) of the variables in **kernels**. By suitably shaping these kernel matrices it is possible to derive meaningful solutions, able to describe in an accurate way the behaviour of the underlying system.

In total, it becomes evident that the two scientific fields, namely system identification and machine learning, share a common target; the accurate prediction of the trajectory of the system, taking into account a trade-off between too complex models that may suffer from limited applicability and too simple models that may suffer from limited interpretability.

1-2 Problem statement

In the past few years, the common ground between the fields of system identification and machine learning attracted the attention of the scientific community [6]. More specifically, the incorporation of the merits of the Gaussian process framework (such as the view of the coefficients to be estimated as the maximum a posteriori estimates) in the (parametric) identification of dynamical systems was pursued.

This goal was eventually accomplished in 2008 with the successful incorporation of a step based on Gaussian processes in the PEI method for LTI systems [7]. From that time onwards the research interest on this topic has attracted the interest of many more scientists, as the exponential growth in the number of the related publications witnesses. In the last five years new kernels that incorporate simple prior knowledge about the underlying systems (such as stability) have been proposed and many theoretical aspects of the kernel based PEI methods have been investigated. The derived results were rather astonishing; the kernel based PEI methods almost always showed a superior performance compared to the standard PEI methods, which were considered to deliver the optimal estimates (due to Cramér-Rao bound), up to that time.

The extension of the kernel methods in the SID algorithms was the next step to be taken. The solution was rather straightforward. For the SID methods that use the so called Vector ARX (VARX) step this extension was accomplished some years later [8]. However, this synergy between Gaussian processes and SID methods is still in a preliminary stage, since various aspects of the algorithm are not examined. This includes general questions (what is the effect of noise, excitation signal etc.) as well as questions about how specific selections in the SID algorithms affect the kernel based methods (such as the chosen past window value). It is therefore our first aim to examine this field, explicitly from a SID perspective. To this end, the analytic comparison of the differences between kernel based PEI and SID methods offers new insights. After this point, we aim at investigating the effect of various parameters in the

accuracy of the kernel based SID. Among these, the most important will be the investigation of the effect of the *past window*, which is a critical parameter in the SID algorithms. An investigation of the kernel based SID method in different simulation examples will be used to reveal in a clear way its advantages and possible pitfalls.

Following the results for the kernel based SID of LTI systems, an important question arises naturally. **Is it possible to extend the kernel based methods in the LPV SID framework**? This question is not trivial at all. Up to our knowledge, there is only one preliminary result on the synergy between the PEI and the kernel methods for LPV systems [9]. It is therefore clear that the incorporation of kernel based methods in SID of LPV systems (and specifically of discrete-time LPV state-space systems) is an open question. And this will be exactly the **main question** of this thesis project.

In order to achieve this goal, we have to follow some necessary steps. First, by using the kernel based SID methods as the basis, the investigation of the similarities and differences between the SID algorithms for LTI and LPV systems has to be investigated. Additionally, the changes that a kernel based approach induces in the LPV SID algorithm have to be identified and taken into account. Questions such as what is the optimal kernel, what kernels can be used in practice and how the algorithm can be implemented in an efficient way have to be answered. Finally, the possible approaches in this problem have to be validated in simulation examples and further re-evaluated.

All in all, the synergy of the SID methods and the kernel based methods has been partially explored in the LTI case but not at all in the LPV case. Therefore, this thesis aims at providing a complete framework, starting from the identification of LTI systems and using this knowledge as a basis for the identification of LPV systems.

1-3 Contributions

This thesis project investigates the synergy between SID methods and kernel based approaches for two different classes of systems, namely the LTI and the LPV systems. It offers some new insights in the LTI case, while in the LPV case novel developments are presented.

More specifically, in the LTI case the following contributions are made:

- A comparison of the proposed kernel structures is performed, using different models as well as different configurations (SNR, available data points). This comparison is not thoroughly performed in the related literature, but it is necessary towards the implementation of the proposed algorithms in an experimental setup. The results showed that the kernel based methods are able of delivering highly accurate models under very challenging simulations setups, such as low SNR, bad excitation and small number of available data points, while the performance of the other SID methods that were used as a comparison was much lower.
- The choice of the past window value is crucial for the accurate estimation in the SID algorithms. In this thesis we investigated the effect of the past window in the kernel based SID. This gave a new insight into the SID methods, currently missing from the literature. More specifically, it is shown that the choice of the past window is not a

crucial choice in the kernel based SID, since these algorithms are able to deliver highly accurate models for almost any past window value.

In the LPV case, we investigated the possible synergy of the kernel methods and the SID algorithms. This led to a number of new insights and developments.

- The common and different aspects between the kernel based SID approaches for LTI systems and the possible ones for the LPV systems is investigated. Following this comparison, we propose and investigate two possible approaches towards a kernel based SID method for LPV systems.
- The notion of optimal regularization for LPV systems is introduced. This will be useful since it sets an upper bound for the accuracy of the estimated model in the asymptotic case (when the past window value is infinite).
- The first proposed approach is investigated and its merits and deficiencies are highlighted. This method is based on assigning a prior on the LPV equivalent Markov parameters, while the proposed kernels show some similarities with the ones in the LTI case.
- The second proposed approach is treating the impulse response coefficients as Gaussian processes. It gives itself rise to two novel methods: the LPV-K&PBSID$_{opt}$ and the LPV-RKHS-PBSID$_{opt}$ algorithm. These algorithms are investigated in depth and we also justify when and why each of these two algorithms is expected to yield better results. These theoretical aspects are further investigated in simulation examples.
- New kernel structures are proposed for the second approach. These kernels aim at accurately capturing the dynamics of the underlying systems, while keeping the number of the parameters to be estimated low. This is necessary to avoid the local-minima in the involved non-convex optimization. The desired result is achieved by exploiting the structure of the impulse response coefficients and using as a basis the Radial Basis Function (RBF) kernel structure.
- All the proposed approaches lead in general to a more accurate estimation of the underlying systems, compared with the up-to-date LPV-PBSID$_{opt}$ based methods. The methods that follow from the second approach are better than the ones from the first approach especially when the number of local systems is relatively low, as it is theoretically justified. Among the investigated methods, the LPV-RKHS-PBSID$_{opt}$ algorithm is in overall the most accurate one as it was indeed expected, following the theoretical analysis that was performed.

1-4 Outline of the thesis

This thesis project is divided in three parts. In the first part the necessary theoretical aspects of the subspace methods for LTI and LPV systems are given, followed by an introduction in Gaussian processes. In the second part we introduce the kernel methods for LTI systems based on the PEI framework and see how they can be extended in the subspace methods,

followed by simulations results and the related conclusions. Finally, in the third part we move towards the kernel based identification of LPV systems. In this part we will propose some novel approaches and their performance will be evaluated based on simulation examples, while in the end we will draw the necessary conclusions.

More specifically, the thesis project is divided in chapters as follows.

In **Chapter 2** we make a brief introduction on the system identification methods for LTI systems. The main focus will be on the state-of-the-art SID algorithms for LTI systems and specifically on the PBSID$_{opt}$ algorithm. In this Chapter we will also highlight the role of the VARX formulation, which will prove to be crucial for the incorporation of the kernel based methods in the SID framework.

In **Chapter 3** we will show how the SID for LTI systems can be extended to cope with discrete-time LPV systems in state-space form. We will mainly focus on the first steps of the LPV-PBSID$_{opt}$ algorithm, which will be proven to be the most important towards the incorporation of the kernel methods in the LPV SID methods.

In **Chapter 4** we outline the main characteristics of the Gaussian process framework. We will define notions such as kernels and we will see how quantities such as marginal likelihood and maximum a posteriori estimates will become useful for the derivation of the estimates.

In **Chapter 5** we will make the transition towards the kernel based approaches. More specifically, by first pointing out the pitfalls of the SID algorithms, we will proceed to the description of the recent developments in the identification of LTI systems, namely the incorporation of Gaussian processes in the PEI methods. Next, we will show how these methods can also be incorporated in the SID framework, pointing out their main characteristics, as well as the differences between the kernel based SID and PEI methods.

In **Chapter 6** we will compare the discussed kernel based SID methods in different simulation setups. Various aspects of the algorithms will be investigated, such as the effect of the past window value, the SNR value and the data length.

At the end of Part I, namely in **Chapter 7**, we will conclude on the kernel based methods for LTI systems and moreover we will discuss about the possible extensions and improvements of these methods.

Part II begins with **Chapter 8**, in which the possible synergy between kernel based methods and SID methods for LPV systems will be examined. To achieve this target, the following steps are followed. First, an investigation on the already existing regularization methods is performed. Then, we delineate two possible approaches for the kernel based identification of LPV systems, each one of which gives rise to different novel methods.

In **Chapter 9** we will resort to simulation examples to evaluate the validity of the proposed methods and highlight their advantages and disadvantages.

Finally, in **Chapter 10** we will conclude on the kernel based SID methods for LPV systems. Numerous questions arise from the novel developments. For this reason, we will analytically elaborate on these aspects of the algorithms that can be improved or further investigated in future work.

Part I

Theoretical Background

Chapter 2

Introduction to Subspace Identification methods for LTI systems in state-space form

In this chapter we will introduce the state-of-the-art framework for the Subspace Identification (SID) of Linear Time Invariant (LTI) systems in state-space form. We will start with a brief overview of the available model descriptions of a state-space model and we will describe in a nutshell the two main identification methods, namely the Prediction Error Identification (PEI) and the SID methods. Finally, we will introduce the PBSID$_{opt}$ method, which will be used as a basis for the kernel based methods that will be developed in the next chapters.

2-1 Introduction to model description forms

The mathematical description of a dynamic system is at the core of control theory. The various model descriptions do not simply describe the same system by different means, but they rather highlight (or hide) different properties of this system. In other words, the model descriptions offer a different point of view. Discrete-time LTI systems are usually described by two different model descriptions: the input-output and the state-space description [10, Ch. 4].

In the general setting, a system in input-output form is described by

$$y_k = g(u_{k-1}, u_{k-2}, \cdots, u_{k-n_u}, y_{k-1}, y_{k-2}, \cdots, y_{k-n_y}), \tag{2-1}$$

where g is a linear function of its inputs. On the other hand, a state-space model is described by

$$\begin{aligned} x_{k+1} &= f(x_k, u_k) \\ y_k &= g(x_k, u_k) \end{aligned} \tag{2-2}$$

where u_k denotes the input at time instant k, y denotes the output and x denotes the states of the system, while f and g denote linear functions, with respect to x_k and u_k.

2-2 Introduction to identification methods for LTI systems

In this thesis we will focus on the identification of discrete-time state-space systems. It is well known that the state-space description offers a very interesting point of viewing the dynamical systems, while many modern control methods are specifically tailored for state-space models. The two most prominent identification schemes for these models are the PEI and the SID methods.

As far as the PEI methods are concerned, the basis for these methods were laid in 60's, while the most significant steps towards the contemporary form of PEI were taken around 1980, most notably by researchers in Swedish universities such as Ljung, Stoica, Wahlberg and Söderström. The PEI methods are mainly used for the identification of systems described by an input-output model, however they can also be employed for the identification of state-space models. Over the past 40 years these methods have shown a high applicability, while the associated theoretical framework offers many tools for the investigation of the asymptotic properties of the estimated coefficients [10]. PEI are based on the explicit minimization of a mathematical criterion, such as $J = \arg\min_{\theta} \sum_{k=1}^{N} ||y_k - \hat{y}_k(\theta)||_2^2$, that is to say, the estimated output value $\hat{y}_k$ is parametrized with the use of the coefficients θ, the value of which is estimated by solving the related optimization problem.

Nonetheless, the PEI methods are also characterized by a number of limitations. First of all, it is often the case that a non-convex optimization algorithm is required for the estimation of the unknown system parameters, which of course can lead to suboptimal solutions. Moreover, if a full block parametrization is performed (e.g. in the case where there is no physical insight about the relation of the states), then the accompanied optimization problem will show high complexity. On the other hand, known state-space parametrizations that require a smaller number of coefficients (such as companion or observer canonical form) usually lead to numerically ill-conditioned problems. The efforts of the system identification community to circumvent these problems finally led to a different approach, which was presented around 1990 and it is now known as Subspace Identification (SID).

SID methods are non-parametric, in the sense that they do not require any a priori parametrization of the unknown system. This leads to the absence of an explicit optimization criterion, thus making cumbersome the investigation of the asymptotic properties of the identified model. However, SID have gained an increased interest over the past 20 years, mainly because the estimated quantities are computed with the use of the Least Squares (LS) method and some suitable projection operations in the vector space defined by the collected input-output data and so any non-convex optimization routines are avoided.

All in all, it would be rather unfair to completely disregard one of the two methods in favour of the other. On the contrary, we could say that the identification method should be chosen based on the specific characteristics of the setup, the desired identification and control goals etc., while a combination of the two methods (by using the identified model based on a SID method as an initialization point for the PEI method) is also an attractive way to tackle the

identification problem. Nonetheless, some specific characteristics of the SID methods, such as the convenience in identifying Multiple Input Multiple Output (MIMO) systems and more importantly, the fact that they do not involve any non-convex optimization problem (which is not always the case in PEI methods) lead to a preference of these methods and so they will also be in the center of our attention in this thesis.

2-3 Subspace Identification methods for LTI systems

Brief history of Subspace Identification methds for LTI systems

Subspace identification methods are a powerful tool for the estimation of state-space models from a given input-output data. Their roots lie in concepts from statistics and linear algebra, as well as system theory. The first related work was presented by Ho and Kalman in 1966 [11]. In this work, the authors showed how the system matrices of a deterministic system can be derived up to a similarity transformation, based on realization theory (which was established in this publication). Nonetheless, this publication didn't directly result in the birth of what is known today as subspace identification. In the next 20 years only one major contribution was made towards the birth of subspace identification. This was the publication by Akaike in 1974, in which he presented an approach for the realization of purely stochastic systems [12].

It was not until 1990 when the first algorithms that belong to subspace identification methods made their appearance. These early developments were focusing on the identification of discrete-time LTI state-space models, which operate under open-loop conditions. The most characteristic works of that period are the ones from Van Overschee and De Moor [13] (N4SID method), Verhaegen [14] (Multivariable Output- Error State-sPace (MOESP) method) and Larimore [15] (CVA method), which were finally incorporated in the unifying theorem, proposed by Van Overschee and De Moor in [16]. From that time on, the subspace identification field experienced a rapid development.

Nowadays, the subspace identification of open-loop LTI systems is a well developed area, while this work is summarized in the books of Van Overschee and De Moor [17] (1996), Katayama [18] (2005) and Verhaegen [11] (2007). Moreover, many prominent researchers in the field of PEI have shown an increasing interest in subspace methods, such as Ljung [19]. The combination of these two methods is also used in the System Identification toolbox in matlab, where subspace methods are used to derive a first estimation when the predictor is non-linear in the parameters to be estimated (e.g. Box-Jenkins parametrization) [20]. Subspace identification for open-loop systems has also been implemented successfully for input-output data given in frequency domain (e.g. [21]).

In contrast to the identification of open-loop systems, the direct use of the aforementioned subspace methods for the identification of closed-loop systems is not possible, because the inherent assumption that the process and measurement noise sources are uncorrelated with the input signal is no longer true. In order to circumvent this limitation, many different approaches are followed, as it will be shown in the next section. These approaches led to state-of-the-art methods, which can also be used for the identification of open-loop systems.

State-of-the-art Subspace Identification methods for LTI systems

A breakthrough in the SID field was accomplished by Chiuso, most notably in [22]. The new approach that was proposed, known as PBSID$_{opt}$, is characterized by a synergy between SID and PEI methods. More specifically, a Vector ARX (VARX) model is used to capture the impulse response of the unknown LTI system, while the estimated quantities are used in a novel SID algorithm in which the state sequence is estimated and subsequently the unknown system matrices are derived. It is well known that the coefficients of an Auto-Regressive with Exogenous Input (ARX) model can be estimated with the use of convex optimization methods, such that all the related steps in this algorithm don't involve any non-convex optimization routine. Moreover, PBSID$_{opt}$ does not require any assumption on the correlation between the noise and the input of the unknown system (except for the requirement that there is at least one delay instant in the open- or closed-loop [23]), thus rendering it proper for the identification of closed-loop systems, too.

The VARX parametrization step, first appeared in [24], is essential to the derivation of accurate models with the least required assumptions, while it also enabled the adaptation of old or the creation of new algorithms that involve this step [23], such as closed-loop MOESP. They key theoretical result for the justification of the VARX based methods lies in the ability of high order ARX model structures to describe any LTI system, as the order goes to infinity [25].

The comparison of these new methods, although it is not exhaustively investigated in the literature (the only related publication is [23]), has shown that in general the PBSID$_{opt}$ algorithm leads to a more accurate identified model than the other state-of-the-art methods. For this reason we will use this algorithm as the basis for the ideas that will be developed in the next part of this thesis project. To do so, the required mathematical framework will be developed in the next section.

Mathematical framework for the PBSID$_{opt}$ algorithm

In this section we will give all the required definitions that are crucial for the ideas that will be developed in Chapter 5. For the same reason, the first steps of the PBSID$_{opt}$ algorithm will be given in this section, while the rest of the steps are given in Appendix A.

First of all, the state-space model description is given in (2-3) and it is depicted in Figure 2-1.

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + w_k \\ y_k &= Cx_k + Du_k + v_k \end{aligned} \tag{2-3}$$

where $dim(A) = n \times n$, $dim(B) = n \times n_u$, $dim(C) = n_y \times n$ and $dim(D) = n_y \times n_u$. The vectors w_k and v_k are called the process noise and measurement noise, respectively. They are assumed to be **zero-mean white noise** sequences and their joint covariance matrix is given by

$$E\left[\begin{bmatrix} v_k \\ w_k \end{bmatrix} \begin{bmatrix} v_j^T & w_j^T \end{bmatrix}\right] = \begin{bmatrix} R & S^T \\ S & Q \end{bmatrix} \delta_{k-j}. \tag{2-4}$$

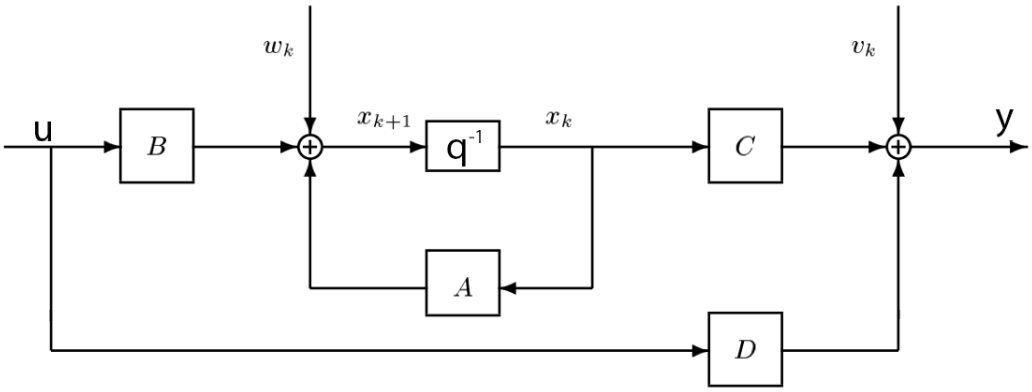


Figure 2-1: Discrete Time System representation

This model will be called from now on the **process form**. Due to the assumptions about the linearity of the system and the whiteness of the noises, we can design a Kalman filter to estimate the state variables. Usually the estimated states are denoted as $\hat{x}$, but here we will ignore this notation for simplicity. Moreover, from now on we will assume for simplicity that there is no feed-through term D. The extension to this case is straightforward, however, specific requirements about the delay of the system have to be fulfilled to render the system identifiable [10, 23]. The so called **innovation form** is given by

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + Ke_k, \\ y_k &= Cx_k + e_k, \end{aligned} \tag{2-5}$$

where the innovation $e_k \in \mathbb{R}^{n_y}$ is a white noise sequence with zero mean and variance equal to $\mathbb{E}(e_j e_k^T) = W\delta_{jk}$ with $W \in \mathbb{R}^{n_y \times n_y}$, $W > 0$, while K is a standard notation for the Kalman gain. Finally, another useful formulation of the state-space model is the **one-step ahead predictor form**, given by

$$\begin{aligned} x_{k+1} &= \tilde{A}x_k + Bu_k + Ky_k, \\ y_k &= Cx_k + e_k, \end{aligned} \tag{2-6}$$

where $\tilde{A} = A - KC$.

At this point we will slightly divert from the conventional PBSID$_{opt}$ nomenclature (e.g. as the one used in [23]). This decision is justified by the need in the next part of this thesis to treat input and output signals separately, as it will be shown in the corresponding chapters. The following definition holds for both input and the output signals.

Definition 2.1. We define the input vector $\bar{u}_k^{(p)} \in \mathbb{R}^{n_u p}$ (similarly, the output matrix $\bar{y}_k^{(p)} \in \mathbb{R}^{n_y p}$)

$$\bar{u}_k^{(p)} = \begin{bmatrix} u_{k-1}^T & u_{k-2}^T & \cdots & u_{k-p}^T \end{bmatrix}^T \tag{2-7}$$

where p is used to denote the size of the past window and the parenthesis is used to distinguish it from u_k to the power of p. △

Moreover, the formulation of Toeplitz or Hankel matrices [26] is a necessary step in SID algorithms. For example, the input and the output measurements are used for the construction of the corresponding Toeplitz/Hankel matrices. A Toeplitz matrix constructed by input measurements is given by

$$U_{i,s,N} = \begin{bmatrix} u_{i+s-1} & u_{i+s} & \dots & u_{i+N+s-2} \\ \vdots & \vdots & \ddots & \vdots \\ u_{i+1} & u_{i+2} & \dots & u_{i+N} \\ u_i & u_{i+1} & \dots & u_{i+N-1} \end{bmatrix} \tag{2-8}$$

Remark 2.1. In the related literature (e.g. [11,23]) a Hankel formulation is usually employed. Even though both choices are possible, in the kernel based methods that we will develop later it is more convenient to use the Toeplitz formulation. Nonetheless, in the Appendix A we will describe the steps of the PBSID$_{opt}$ algorithm using the Hankel formulation to stick with the related SID literature. △

Finally, we will define the impulse response of the LTI system based on (2-6). To keep the notation tractable, **without any loss of generality we will assume from now on that the unknown system is Single Input Single Output (SISO)**.

Definition 2.2. The impulse response of a SISO LTI system described by (2-6) is given by

$$y_k = \sum_{t=1}^{\infty} h_t^u u_{k-t} + \sum_{t=1}^{\infty} h_t^y y_{k-t} + e_k. \tag{2-9}$$

△

Lemma 2.1. The impulse response coefficients h_t^u and h_t^y are given by

$$\begin{aligned} h_t^u &= C\tilde{A}^{t-1}B \\ h_t^y &= C\tilde{A}^{t-1}K \end{aligned} \tag{2-10}$$

Proof. The result in (2-10) can be obtained by propagating (2-6) to the past. Together with the assumption that the eigenvalues of $\tilde{A}$ are inside the unity circle (that is to say, the predictor is stable) then it holds that $\lim_{t\to\infty} \tilde{A}^t \to 0$. □ □

For practical reasons we have to limit ourselves to a finite past window p. This in turn means that (2-9) will not hold exactly. In this case, the output is described by [23]

$$y_k = C\tilde{A}^p x_k + \sum_{t=1}^{p} h_t^u u_{k-t} + \sum_{t=1}^{p} h_t^y y_{k-t} \tag{2-11}$$

If the predictor state-space model is exponentially stable, it means that the eigenvalues of the matrix $\tilde{A}$ are strictly inside unity circle. Consequently, for a large enough value of p, $\tilde{A}^p \approx 0$. Taking this into account, the **VARX** form can be constructed. First, we define the quantities $Y = Y_{p+1,1,N-p}$, $Y_p = Y_{1,p,N-p}$ and $U_p = U_{1,p,N-p}$, $E = E_{p+1,1,N-p}$ based on (2-8). Now, based on straightforward computations we derive the following relationship.

$$Y = \begin{bmatrix} h_1^u & h_2^u & \cdots & h_p^u \end{bmatrix} U_p + \begin{bmatrix} h_1^y & h_2^y & \cdots & h_p^y \end{bmatrix} Y_p + E_p. \tag{2-12}$$

It is well known [10] that the unknown coefficients h_t^u, h_t^y, $t \in \{1, \ldots, p\}$ can be estimated with the use of the LS method, which minimizes the objective function

$$\min_{h_t^u, h_t^y, t \in \{1,\ldots,p\}} \left\| Y - \begin{bmatrix} h_1^u & h_2^u & \cdots & h_p^u & h_1^y & h_2^y & \cdots & h_p^y \end{bmatrix} \begin{bmatrix} U_p \\ Y_p \end{bmatrix} \right\|_2^2. \tag{2-13}$$

As it was explained in [25], the ARX model structure can approximate arbitrarily well any LTI system as $p \to \infty$. In the classical VARX based SID framework the selection of p reflects a trade-off between two competitive aspects: a large past window renders the approximation $\tilde{A}^p \approx 0$ accurate enough and leads to an accurate approximation of the impulse response but it may lead to overfitting. The latter arises from the fact that, for a large past window, the number of coefficients to be estimated is too high and so the unknown coefficients may adjust to the noise characteristics, which is undesirable. Since this trade-off **is very important for the success of the PBSID$_{opt}$ algorithm**, it will be further investigated in Chapter 5.

After the estimation of the VARX model coefficients, the algorithm proceeds to the estimation of the state sequence and finally to the estimation of the LTI system matrices. This procedure is analytically described in Appendix A.

2-4 Conclusion on Subspace Identification methods for LTI systems

The state-of-the-art SID algorithms for LTI systems are characterized by a VARX formulation followed by a model reduction step, as it is described in the previous section and in Appendix A. For an accurate estimation of an LTI system, the selection of the past window value p is a crucial aspect of the algorithm, since it affects the solution the LS problem (2-13). The PBSID$_{opt}$ algorithm was shown to lead to more accurate identified models, following the results in [23]. For this reason PBSID$_{opt}$ will serve as a basis for the methods that will be developed in Part 2 of the thesis and it will be explained how kernel based regularization methods can lead to a change of paradigm in the up to now discussion on the selection of p.

Chapter 3

Introduction to SID methods for LPV systems in state-space form

Following the developments of Subspace Identification (SID) and Prediction Error Identification (PEI) methods for Linear Time Invariant (LTI) systems, the investigation of how these two methods can be extended to Linear Parameter Varying (LPV) systems was a major research subject over the recent years. Both research directions were investigated, while a good summary for both can be found in a series of books [2, 3, 27].

LPV systems themselves are also at the epicentre of an intensive scientific research that aims at building a more concrete theoretical basis, developing advanced modelling and control methods [28, 29] and of course, finding ways to identify such systems.

In this chapter, we will make a brief review of the characteristics of the PEI and SID methods for the identification of discrete-time LPV systems in state-space form. The main focus will be on the SID methods and especially on the LPV-PBSID$_{opt}$ method, which is a global SID method [2]. For this purpose we will develop the necessary mathematical framework, which will slightly differ from the one found in [30]. Following the same structure as in the previous chapter, we will first show how an LPV VARX equivalent form can be derived, which will be crucial for the ideas that will be developed in Part III. After this point, the steps that have to be taken are thoroughly explained in [30], while they are also outlined in Appendix B.

3-1 PEI methods for LPV systems

As it is already discussed in Chapter 2, PEI methods deliver a model which is in the I/O form. Within the context of PEI many different LPV identification methods have been developed [31], also by extending the model structures for LTI systems (such as ARX, Box-Jenkins etc.) to the LPV case. A rather complete and mathematically analytical description of the related framework is also given in [3].

The PEI methods offer an elegant statistical framework to characterize the asymptotic properties of the estimated variables. Nonetheless, there is a number of drawbacks associated

with these methods. First of all, many of the LPV equivalent model structures (such as ARMAX or Box-Jenkins) require the solution of a non-convex problem, similar to the LTI case. Most importantly, in the LPV case, the relation between the I/O and the state-space equivalent model is not trivial. As it was shown in [32], the state-space equivalent of an I/O model requires the introduction of dynamic dependency in the scheduling parameters, thus increasing the complexity of the model. This factor could be crucial in terms of control, since the majority of modern control methods for LPV systems require a state-space model. For this reason, efforts to circumvent this problem have been made (based on realization and model reduction techniques), which perform a trade-off between accuracy and complexity of the derived state-space model. However, in order to ensure static dependency in the LPV state-space model, the resulting state-space model is often not minimal [33, 34].

3-2 SID methods for LPV systems

The SID methods offer a viable alternative to the PEI methods also for the LPV case. These methods deliver an LPV state-space model, which is usually described by an affine function of the various local systems. Nonetheless, other model descriptions have also been investigated, such as Linear Fractional Transformation (e.g. [35], where the equivalence between this form and the affine one can also be established [36]) or different basis functions such as polynomial. In general, a parameter dependent matrix can be expressed in the affine form

$$A(\delta) = A^{(0)} + A^{(1)}\delta^{(1)} + \ldots + A^{(m)}\delta^{(d)}, \quad \in \mathbb{N}, \tag{3-1}$$

where $\delta = \left[\delta^{(1)}, \ldots, \delta^{(m)}\right] \in \mathbb{R}^{n_\delta}$ are functions of the scheduling parameters μ. It can be proven that any LPV model can be written in an affine form similar to (3-1) under some finite substitutions of the variables [32].

The origins of SID methods for LPV systems were laid at the end of nineties. Like many pioneering ideas, the first developed algorithms were making some rather restrictive assumptions about the available information and the operating conditions of the underlying model. For example, this was the case in the algorithms developed by Verdult, in which the unknown system was allowed to operate only in open loop [36]. The algorithms developed in his work offer a so-called global approach to identify an LPV system [2]. In other words, these algorithms are based on the assumption that all the matrices of the local systems can be estimated by a single experiment, by using information about the input, the output and the scheduling parameter of the unknown system. It is worth noting that a global approach was followed in one of the very first publications regarding the identification of LPV state-space models [35]. However, in this approach the unknown coefficients were estimated via a gradient-based non-convex optimization algorithm, so they don't belong to the family of SID methods.

Nonetheless, the global approaches for the identification of LPV systems show mainly two limitations: first, the input and the scheduling parameter were assumed to be sufficiently excited and second, these methods were characterised by the so-called curse of dimensionality, that is to say, the number of the parameters that have to be estimated is increasing

exponentially with respect to the past window p [37]. The effort of the scientific community to circumvent this problem led to different approaches.

The first approach was to follow a different identification paradigm, leading to methods called **local**. In these methods the scheduling parameter is assumed to be constant and only the input is excited. This procedure, though, creates another implication; the identified local models are not expressed in the same basis due to the fact that the SID methods identify a system **up to a similarity transformation**. For this reason, an extra step has to be introduced so that all local systems are transformed to a common state basis [38]. Finally, when the models are brought into a common basis, the parameter dependent model is derived by interpolating between the various local models [2].

The complexity of the local methods and the expected loss of accuracy due to the interpolation render these approaches definitely not optimal. On the other hand, many scientists continued to work on the global methods, aiming at improving the accuracy of these methods and circumventing the aforementioned limitations. This effort led to some very interesting results, which are characterized by less computational complexity and less assumptions regarding the operation and the characteristics of the underlying model, compared to the first developed global methods.

3-3 State-of-the-art SID methods for state-space models

The recent developments in the SID methods for LTI systems, as they were described in Chapter 2, had also some important consequences in the LPV SID methods. More specifically, the synergy between LTI and LPV identification techniques (accomplished via the VARX parametrization of the state-space model in the LTI case) and the improved results that were derived with these new methods (such as PBSID$_{opt}$ [22]), let along the fact that they can identify systems operating in open- or closed-loop, led to the question: Is it possible to enjoy these desirable properties in SID methods for LPV systems?

The LPV-PBSID$_{opt}$ algorithm

This question was finally answered with the development of the LPV-PBSID$_{opt}$ [30]. In order to describe this method, we have to develop the related mathematical framework. The LPV-PBSID$_{opt}$ assumes that the system is given in the affine form (called the innovation form)

$$x_{k+1} = \sum_{i=1}^{m} \mu_k^{(i)} \left(A^{(i)} x_k + B^{(i)} u_k + K^{(i)} e_k \right), \tag{3-2}$$

$$y_k = C x_k + D u_k + e_k, \tag{3-3}$$

where $m \in \mathbb{N}^+$, $x_k \in \mathbb{R}^n$, $u_k \in \mathbb{R}^{n_u}$, $y_k \in \mathbb{R}^{n_y}$ and $e_k \in \mathbb{R}^{n_y}$ denote the number of local systems, the state, the input, the output and the zero mean white innovation process, respectively. The innovation process will be assumed to be normally distributed, with covariance described by $\text{cov}(e_k, e_k') = W \delta_{k-k'}$, where the function δ represents the Kronecker delta and $W \in \mathbb{R}^{n_y \times n_y}$, $W > 0$ is a diagonal matrix. Similar to the LTI case, the innovation sequence

is given by $e_k = y_k - \hat{y}_k$, where y_k is the measured and $\hat{y}_k$ is the estimated output. The scheduling parameters $\mu_k^{(i)} \in \mathbb{R}$ could be seen as weighting factors among the various local systems $A^{(i)}$, while we will always assume that $\mu_k^{(1)} = 1$. Without loss of generality (as long as specific conditions concerning the delay of the system are fulfilled, see [10, 23]), we will assume that $D = 0$. Taking this into account we can rewrite (3-2)-(3-3)in the predictor form

$$x_{k+1} = \sum_{i=1}^{m} \mu_k^{(i)} \left(\tilde{A}^{(i)} x_k + B^{(i)} u_k + K^{(i)} y_k \right), \tag{3-4}$$

$$y_k = C x_k + e_k, \tag{3-5}$$

where $\tilde{A}^{(i)} = A^{(i)} - K^{(i)} C$. Now we are in position to formulate the identification problem:

Problem 3.1. Given the input sequence u_k, the output sequence y_k and the scheduling parameter μ_k for $k = \{1, \ldots, N\}$, $k \in \mathbb{N}^+$ estimate, if they exist, the system matrices $A^{(i)}, B^{(i)}, C, D$ and $K^{(i)}$ with $i \in \{1, 2, \cdots, m\}$ **up to a global similarity transformation**.

At this point we have to make an important remark. In order to keeep consistency with the framework used in the next sections, we will deviate a little from the formulation developed in the work of J.W. van Wingerden [30, 39]. Nonetheless, in the end we will end up with the same quantities and variables. Now let us first introduce the following definitions.

Definition 3.1. We define the following matrices:

$$\begin{aligned} \mathcal{L}_1^u &= \left[B^{(1)},\ \cdots,\ B^{(m)} \right], \\ \mathcal{L}_1^y &= \left[K^{(1)},\ \cdots,\ K^{(m)} \right], \end{aligned} \tag{3-6}$$

Based on (3-6), we extend this definition to include $\mathcal{L}_t^u$, $\mathcal{L}_t^y$ for every $t \in \mathbb{N}^+$ [30].

$$\begin{aligned} \mathcal{L}_t^u &= \left[\tilde{A}^{(1)} \mathcal{L}_{t-1}^u,\ \cdots,\ \tilde{A}^{(m)} \mathcal{L}_{t-1}^u \right], \\ \mathcal{L}_t^y &= \left[\tilde{A}^{(1)} \mathcal{L}_{t-1}^y,\ \cdots,\ \tilde{A}^{(m)} \mathcal{L}_{t-1}^y \right], \end{aligned} \tag{3-7}$$

where $\mathcal{L}_t^y,\ \mathcal{L}_t^u\ \in \mathbb{R}^{n \times m^t}$. △

Based on (3-7), we can verify that the number of columns increases exponentially as m^t, leading to the well known curse-of-dimensionality.

Definition 3.2. We define the μ dependent vector $P_{t|k}$ as

$$P_{t|k} = \mu_{k-1} \otimes \ldots \otimes \mu_{k-t}, \quad P_{t|k} \in \mathbb{R}^{m^t}, \tag{3-8}$$

△

where $\mu_k = \begin{bmatrix} 1, & \mu_k^{(2)}, & \cdots, & \mu_k^{(m)} \end{bmatrix}^T$ and $\otimes$ represents the Kronecker product. Moreover, we will make use of the following transition matrix for discrete-time varying systems, defined as [40]

$$\begin{aligned} \phi_{k,j} &= \tilde{A}_{k-1} \cdots \tilde{A}_{k-j+1} \tilde{A}_{k-j} \\ \text{where } \tilde{A}_k &= \sum_{i=1}^{m} \mu_k^{(i)} \tilde{A}^{(i)}. \end{aligned} \tag{3-9}$$

Remark 3.1. From this point onwards to keep the notation tractable **we will present the framework for a SISO system**. The extension to the MIMO case is a straightforward procedure, see for example [30]. However, when we find it necessary, we will give information about the modification that have to be done in the MIMO case.

Now, we are in position to describe the impulse response of an LPV system that corresponds to (3-4),(3-5), which will then lead to the VARX form of the LPV system.

Definition 3.3. The impulse response of an LPV system is described by

$$\begin{aligned} y_k = &\sum_{t=1}^{\infty} h^u(\mu_{k-t}, \ldots, \mu_{k-1}; t) u_{k-t} \\ &+ \sum_{t=1}^{\infty} h^y(\mu_{k-t}, \ldots, \mu_{k-1}; t) y_{k-t} + e_k, \end{aligned} \tag{3-10}$$

△

where the impulse response coefficients can be analytically characterized with the use of Definitions 3.1, 3.2, as the next lemma shows.

Lemma 3.1. The impulse response coefficients
$h^u(\mu_{k-t}, \ldots, \mu_{k-1}; t)$ and correspondingly for $h^y(\mu_{k-t}, \ldots, \mu_{k-1}; t)$ are given by

$$\begin{aligned} h^u(\mu_{k-t}, \ldots, \mu_{k-1}; t) &= C \mathcal{L}_t^u P_{t|k}, \\ h^y(\mu_{k-t}, \ldots, \mu_{k-1}; t) &= C \mathcal{L}_t^y P_{t|k} \ . \end{aligned} \tag{3-11}$$

Proof 3.1. The result in (3-11) can be obtained by propagating (3-4)-(3-5) to the past and taking into account that for a stable predictor matrix $\tilde{A}_k$, $\lim_{j \to \infty} \phi_{k,j} \to 0$. □

Remark 3.2. The previous Lemma **holds only in the case where both B and K matrices depend on the scheduling parameter**. If this is not the case for one/both matrices, then we have to change the corresponding $P_{t|k}$ vector to preserve consistency.

In reality, we will limit ourselves to a finite past window p, similar to the LTI case, assuming that $\phi_{k,j} \approx 0$ for $j > p$ for a stable predictor. Again, the value of the past window, p, has to be kept small enough to avoid the curse-of-dimensionality and to avoid the problem of overfitting but on the other hand it has to be large enough so that the aforementioned approximation holds.

A standard way to treat the output data is by discarding the first p samples [10, p.3]. By doing so, we define the stacked output matrix as

$$Y = [y_{p+1}, y_{p+2}, \dots, y_N], \tag{3-12}$$

while we can define the matrix E in a similar way. The output signals for the finite case are given by

$$\begin{aligned}
y_{p+1} &\approx C\mathcal{L}_1^u P_{1|p+1}u_p + \dots + C\mathcal{L}_p^u P_{p|p+1}u_1 + C\mathcal{L}_1^y P_{1|p+1}y_p + \dots + C\mathcal{L}_p^y P_{p|p+1}y_1 + e_{p+1} \\
y_{p+2} &\approx C\mathcal{L}_1^u P_{1|p+2}u_{p+1} + \dots + C\mathcal{L}_p^u P_{p|p+2}u_2 + C\mathcal{L}_1^y P_{1|p+2}y_{p+1} + \dots + C\mathcal{L}_p^y P_{p|p+2}y_2 + e_{p+2} \\
&\vdots \\
y_N &\approx C\mathcal{L}_1^u P_{1|N}u_{N-1} + \dots + C\mathcal{L}_p^u P_{p|N}u_{N-p} + C\mathcal{L}_1^y P_{1|N}y_{N-1} + \dots + C\mathcal{L}_p^y P_{p|N}y_{N-p} + e_N.
\end{aligned} \tag{3-13}$$

Now we can define the stacked matrix Z as

$$Z = \begin{bmatrix}
P_{1|p+1}u_p & P_{1|p+2}u_{p+1} & \cdots & P_{1|N}u_{N-1} \\
\vdots & \vdots & \vdots & \vdots \\
P_{p-1|p+1}u_2 & P_{p-1|p+2}u_3 & \cdots & P_{p-1|N}u_{N-p-1} \\
P_{p|p+1}u_1 & P_{p|p+2}u_2 & \cdots & P_{p|N}u_{N-p} \\
P_{1|p+1}y_p & P_{1|p+2}y_{p+1} & \cdots & P_{1|N}y_{N-1} \\
\vdots & \vdots & \vdots & \vdots \\
P_{p-1|p+1}y_2 & P_{p-1|p+2}y_3 & \cdots & P_{p-1|N}y_{N-p-1} \\
P_{p|p+1}y_1 & P_{p|p+2}y_2 & \cdots & P_{p|N}y_{N-p}
\end{bmatrix}, \quad Z \in \mathbb{R}^{\tilde{q}\times N}. \tag{3-14}$$

where, in the general MIMO case, $\tilde{q} = (n_u + n_y)\sum_{j=1}^{p} m^j$. Before we formulate the Least Squares (LS) problem that we have to solve, we will define the **LPV extended controllability matrix** as

$$\mathcal{K}^{(p)} = \begin{bmatrix} \mathcal{L}_1^u & \cdots & \mathcal{L}_p^u & \mathcal{L}_1^y & \cdots & \mathcal{L}_p^y \end{bmatrix} \qquad \in \mathbb{R}^{n\times\tilde{q}}. \tag{3-15}$$

Finally, we are in position express (3-14) in vector form and solve the corresponding LS problem to derive the unknown matrix CK^p.

$$\begin{aligned}
&Y = C\mathcal{K}^{(p)}Z + E \Rightarrow \\
&\min_{CK^p} ||Y - C\mathcal{K}^{(p)}Z||_2^2
\end{aligned} \tag{3-16}$$

where, $E = [e_{p+1}, \quad e_{p+2}, \quad \dots, \quad e_N]$, while in the MIMO case the norm-2 LS problem has to be replaced by the Frobenius norm [26]. The solution for the LS problem in (3-16) can be computed analytically and it is given by

$$C\mathcal{K}^{(p)} = YZ^T(Z^TZ)^{-1} \tag{3-17}$$

At this point it becomes obvious that we indeed came up with the same LS problem as in [30] by introducing some additional coefficients in the intermediate steps. More specifically, the main difference between the notation that we introduced and the standard one is that we split the various coefficients such that they correspond to one specific input or output signal. As we will show in Part III, with this notation we can more easily describe the kernel based regularization. The next steps of this algorithm are identical to the ones given in [30] and they are given for convenience in Appendix B.

3-4 Conclusion

The LPV systems form a highly active scientific area over the past few years. As far as the SID methods are concerned, we have seen that the LPV-PBSID$_{opt}$ algorithm is a very attractive way to identify discrete-time state-space models. The basic remark for this algorithm is that it also involves a VARX step, in which the impulse response coefficients of the LPV system are estimated. This specific property will prove to be very useful (as it will be described in Part III) towards the introduction of kernel based methods in order to increase the accuracy of the identified LPV model.

Chapter 4

Introduction to Gaussian processes

In this chapter we will make a short introduction to Gaussian processes. We do not intend to give an exhaustive review of this field but an interested reader can find a well-written treatise in [5]. Machine learning and especially Gaussian processes for machine learning are in the center of an extensive scientific research over the last ten years. Although being developed to tackle problems similar to the ones that data-driven identification and model-based control are trying to tackle, the two scientific fields seemed to move on two parallel lines. It is characteristic that this process is also reflected in the vocabulary that the two fields are using. When someone hears about "training set" in the machine learning community, it refers to the "estimation set" within the system identification community. The same holds for the words "test set" and "validation set", correspondingly. Nonetheless, in the recent years, following the thorough cover of the Gaussian processes topic by Rasmussen [5], not only a common ground between these two methods was established but also a possible synergy started to grow up as an idea. It was after all a natural process, following the employment of Least Squares Support Vector Machines (LS-SVM) in system identification (e.g. [41, 42]), which is another popular machine learning approach.

4-1 Gaussian processes for Machine Learning

Gaussian process regression, in simple terms, focuses on inferring a relation between inputs u_k and outputs y_k, given a dataset $\mathcal{D} = \{(u_1, y_1), (u_2, y_2), \ldots, (u_N, y_N)\}, N \in \mathbb{N}^+$ (also known as "supervised learning", where the teacher is the dataset and the student is the function!). It is usually the case that the output is corrupted by additive noise. In this case, a mathematical formulation can be given by

$$y = f(u) + e, \tag{4-1}$$

where $u \in \mathbb{R}^d$, $f : \mathbb{R}^d \to \mathbb{R}$. In the Gaussian process framework, the additive noise is assumed to follow an independent, identically distributed Gaussian distribution, that is to say,

$$e \sim \mathcal{N}\left(0, \sigma^2\right). \tag{4-2}$$

The function f can be any nonlinear functional that maps the input space to the space of real numbers. In the Gaussian process regression it is customary to introduce a prior over the unknown function f that expresses our beliefs about it. Based on this prior and the collected training set, we can compute the posterior distribution of the function f at the training points and moreover we can use this posterior distribution to make predictions for a given input vector u. In other words, Gaussian process regression is based on a Bayesian framework, following the Bayes rule

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{marginal likelihood}}. \tag{4-3}$$

Gaussian processes are named so due to the fact that the function f is modelled as a Gaussian process, that is to say, its distribution can be fully characterized by its mean and covariance.

$$\begin{aligned} \mathbb{E}\left[f((x)\right] &= m(x), \\ \mathbb{E}\left[f((x)\, f((x')\right] &= k(x, x'), \end{aligned} \tag{4-4}$$

where, usually, $m(x)$ is set to zero. The Gaussian process will also be written as $f(x) \sim \mathcal{GP}$.

The latter assumptions, together with the assumption about the noise characteristics renders the output signal a normally distributed random process. This in turn means that all the required quantities for the estimation of the posterior distribution of f can be computed in an analytical way [1]. If this was not the assumption, then more cumbersome, non-analytic methods (such as Markov Chain Monte Carlo or particle based methods [44]) or analytic approximation methods can be used, for which the computational burden is usually much higher than in the Gaussian case or the estimation is not accurate.

The posterior distribution of f can be used to predict the value of the function at the training points (auto-validation) as well as at the test points. To illustrate this idea, let us assume that we have collected N data points, where the outputs are stacked in the vector $\mathbf{Y} = [y_1 \dots y_N]^T$ and the function evaluations are stacked in the vector $\mathbf{f}(U) = [f(u_1) \dots f(u_N)]^T$. Then, the joint distribution of the outputs and the function values at the test locations is given by

$$\begin{bmatrix} Y \\ f(U_*) \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(U,U) + \sigma^2 I & K(U,U_*) \\ K(U_*,U) & K(U_*,U_*) \end{bmatrix}\right) \tag{4-5}$$

where

$$K(U,U) = \begin{bmatrix} k(u_1,u_1) & k(u_1,u_2) & \cdots & k(u_1,u_N) \\ k(u_2,u_1) & k(u_2,u_2) & \cdots & k(u_2,u_N) \\ \vdots & \ddots & \ddots & \vdots \\ k(u_N,u_1) & k(u_N,u_2) & \cdots & k(u_N,u_N) \end{bmatrix} \tag{4-6}$$

[1] Based on the assumptions about the normal distribution of f and e, the posterior of f is also Gaussian and so its mean is equal to its mode (the value that appears most often in the data). If we set the posterior estimate to be equal to its mean value, then the estimate coincides with the Maximum a Posteriori (MAP) estimate [43]

and the the star is used to denote the predictive test points. The matrix $K(U_*, U)$ (and its transpose, $K(U, U_*)$) is built following the same rationale as $K(U, U)$. Based on these equations and the known assumptions about normality, we can compute the posterior distribution at the training points, $f(U)$.

$$\begin{aligned} \mathbb{E}\left[f(U)|U, Y\right] &= K(U,U)\left(K(U,U)+\sigma^2 I\right)^{-1} Y \\ \mathbb{E}\left[f(U)f(U)^T|U, Y\right] &= K(U,U) - K(U,U)\left(K(U,U)+\sigma^2 I\right)^{-1} K(U,U). \end{aligned} \tag{4-7}$$

Moreover, we can compute the value of the function at the test points $f(U_*)$. In this case the mean and the covariance are given by

$$\begin{aligned} \mathbb{E}\left[f(U_*)|U, U_*, y\right] &= K(U_*,U)\left(K(U,U)+\sigma^2 I\right)^{-1} Y \\ \mathbb{E}\left[f(U_*)f(U_*)^T|U, U_*, y\right] &= K(U_*,U_*) - K(U_*,U)\left(K(U,U)+\sigma^2 I\right)^{-1} K(U,U_*). \end{aligned} \tag{4-8}$$

The main difference between the Gaussian process and the parametric model description is that the former one does not parametrize explicitly the unknown function based on a fixed basis (e.g. linear or polynomial). This in turn means that in the Gaussian process framework we do not acquire an explicit mathematical formula for the unknown function but a relation between the inputs and outputs based on their statistical properties, which is also called a **non-parametric** model. In other words, even if a basis is infinite dimensional (e.g. the Radial Basis Function [5]), the Gaussian process regression framework is able to compute the a posteriori estimate of f. It is therefore obvious that this framework is characterized by increased flexibility compared to the parametric one, thus potentially enabling the approximation of almost every (non)linear function.

Corrsepondence between Reproducing Kernel Hilbert Space (RKHS) and posterior estimates

RKHS defines an hypothesis space which is useful when it comes to the derivation of a regularized solution to an ill-posed problem [45]. An introduction to this theory is given in Appendix D. Here we only want to highlight the relation between the solution derived in (4-7) and the Tikhonov regularization [26, p. 309], which is derived by facilitating the RKHS framework. As it is shown in Appendix D, the mean of the posterior $\mathbb{E}\left[f(U)|U, Y\right]$ corresponds to the following Least Squares (LS) Tikhonov regularization problem.

$$\min_{\alpha \in \mathbb{R}^N} ||Y - \alpha K(U,U)|| + \alpha K(U,U)\alpha^T, \tag{4-9}$$

and the quantity $\mathbb{E}\left[f(U)|U, Y\right]$ is equal to

$$\mathbb{E}\left[f(U)|U, Y\right] = \alpha K(U,U). \tag{4-10}$$

4-2 Estimation of the hyperparameters

Up to this point there was no discussion about the covariance matrix K. It is usually the case that K is expressed as a function of some **hyperparameters**, symbolized here by the variable η, so we will write it as $K(\eta)$. The choice of these variables is crucial for the accurate approximation of the unknown function, since they encode important information about the characteristics of the unknown system.

Example 4.1. To illustrate how the selection of the hyperparameters plays a crucial role in the approximation of the unknown function, let as assume that we have collected an input-output data $(\{y_1, u_1\}, \ldots, \{y_{20}, u_{20}\})$, where the output is corrupted by an additive noise, and we are trying to approximate the underlying function with the use of an RBF kernel [5]. Since we intend to focus only on the importance of the hyperparameters, it suffices to state that the RBF kernel is affected by two parameters, denoted as σ_u and λ_u [46]. In the following two figures we investigate especially the role of the parameter λ_u.

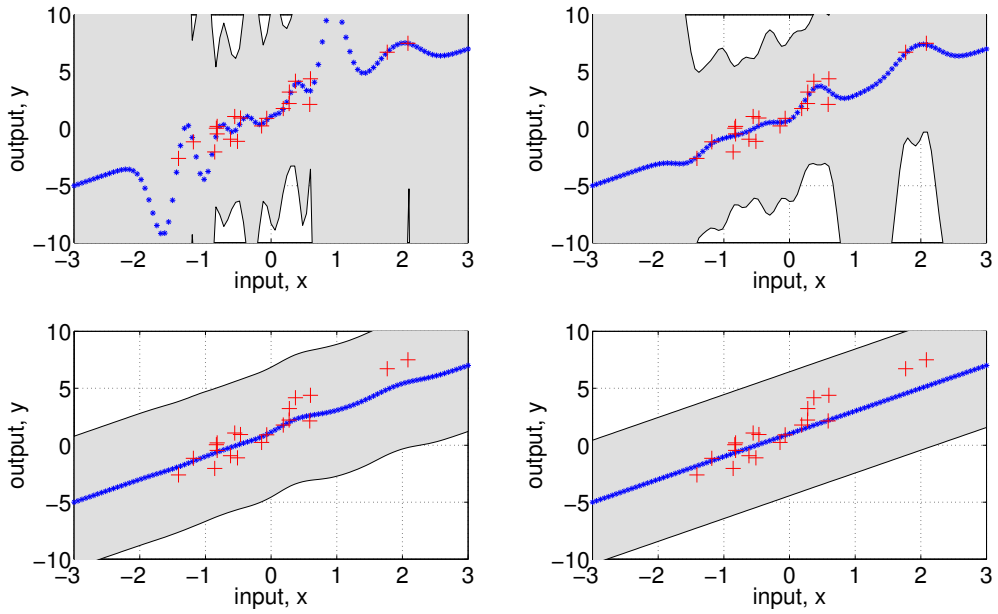


Figure 4-1: Effect of λ_u in the estimation and prediction of the unknown function. The red crosses correspond to the measured outputs and the blue stars corresponds to the predicted values, while the grey area corresponds to the 95% confidence bounds. Top left: $\lambda_u = 100$. Top right: $\lambda_u = 10$. Bottom left: $\lambda_u = 1$. Bottom right: $\lambda_u = 0.1$ [47].

As we can see from Figure 4-1, the correct selection of the hyperparameters plays a crucial role. If we choose a very small value for λ_u the confidence bounds are rather tight but the estimated function is not really flexible, in the sense that the estimated model shows too low complexity. On the other hand, a large λ_u value leads to a very flexible model, which is able to perform estimations close to the measured outputs. However, this flexibility leads to very

large confidence bounds and it also means that the Gaussian process has adopted the noise characteristics, which is of course undesirable since it can lead to bad predictions.

It is now evident that a good estimation with the Gaussian process framework de facto requires a careful selection/estimation of the hyperparameters. For this purpose, the two most often used methods are the **marginal likelihood** and the **cross-validation**.

Estimation of hyperparameters via Marginal Likelihood

The most common method for the estimation of the hyperparameters is through a Bayesian framework lying on the **marginal likelihood** function, which is also known as the **empirical Bayes** method.

This method can also be viewed based on an hierarchical model, as the one depicted in Figure 4-2. The quantities f and e are stochastic according to (4-2),(4-4) and they depend on some fixed (deterministic) hyperparameters (η and σ correspondingly). Finally, the quantity y is also stochastic, following (4-1).

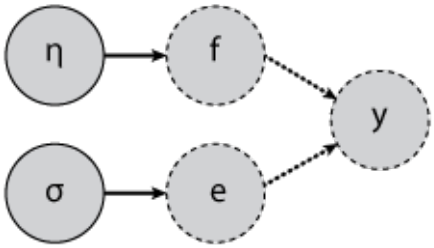


Figure 4-2: Hierarchical model depiction of the model. Dashed lines and circles represents stochastic quantities/relations, while the compact lines and circles represent deterministic ones.

As we mentioned before, at the core of the Bayesian scheme is the **marginal likelihood**. Its name comes from the fact that we marginalize the probability distribution function of y with respect to f, that is to say,

$$p(y|u,\eta) = \int p(y,f|u,\eta)df = \int p(Y|f,u,\eta)p(f|u,\eta)df \tag{4-11}$$

Taking into account the assumptions stated above, we can compute (4-11) analytically, as it is shown in the following equation.

$$p(Y|U,\eta) = \frac{1}{(2\pi)^{\frac{N}{2}}\left(\det\left(K+\sigma^2 I\right)\right)^{\frac{1}{2}}} \exp\left(-\frac{1}{2}Y^T\left(K+\sigma^2 I\right)^{-1}Y\right). \tag{4-12}$$

An intrinsic characteristic of the Bayesian scheme is that it automatically incorporates a trade-off between model complexity and data fit [48]. This effect is also called “Occam’s

Razor", even though this term usually refers to the automatic trade-off between the number of the parameters and data fit, which in our case is not applicable since we focus on non-parametric ways to model the output. This can be clearly seen when the negative logarithm of (4-12) is employed, which is the criterion that is usually minimized in order to estimate the hyperparameters. This quantity is given by

$$-\log p(Y|U,\eta) = \frac{N}{2}\log(2\pi) + \frac{1}{2}\log\left(\det\left(K(\eta)+\sigma^2 I\right)\right) + \left(\frac{1}{2}Y^T K(\eta) + \sigma^2 I\right)^{-1} Y. \quad (4\text{-}13)$$

The last two terms of (4-13) account for different characteristics of the system, while the first one is simply a normalization term. More specifically, the second term is not affected by the output signal and it can be viewed as a penalizing term for the complexity of the system. On the other hand, the last term can be seen as a data-fit term due to output vector Y. All in all, the intrinsic regularization of the marginal likelihood qualifies it as an attractive way to compute the unknown hyperparameters η. This can be done by minimizing (4-13) with respect to η. However, there is also a price to pay: **the optimization algorithm is non-convex**. This is not necessarily devastating, since ending up in a local minimum means that the data is interpreted with a different way; one local minimum may corresponds to a high complexity and low noise system and another one to a low complexity but high noise system. Nonetheless, the possibility of ending up in a "bad" local minimum that leads to a totally wrong interpretation of the data cannot be excluded (more discussion about this subject can be found in Section 5.4 of [5]).

Estimation of hyperparameters via cross-validation

The idea of cross-validation, in simple terms, requires the division of the data in smaller parts, where some of them are used for the estimation (training) of the unknown hyperparameters and the rest are used for the validation (test) of the derived non-parametric model. The selection of a measure of fit gives rise to many possibilities; the predictive log probability and the squared loss functions are among them. The latter one is also used in the classical Prediction Error Identification (PEI) framework when use of a parametric model is made [49], which is however not useful in the non-parametric case since it ignores the variance of the validation set.

Let as assume that we leave one element (y_i) for validation. In this case, the predictive minus log probability is given by [5]

$$-\log p(y_i|U,Y_{-i},\eta) = \frac{1}{2}\log(2\pi) + \frac{1}{2}\log\det\left(\Sigma_i(\eta)\right) + \frac{1}{2}\left(y_i-\mu_i\right)^T \Sigma_i^{-1}\left(y_i-\mu_i\right), \quad (4\text{-}14)$$

where Y_{-i} denotes the outputs except for y_i, while the predictive mean μ_i and covariance Σ_i are computed by (4-8). This method is also known as **leave-one-out** (LOO) method due to the fact that only one output point is used for validation, but this is repeated for **all** the outputs, so the total LOO minus log predictive probability is given by

$$J(\eta) = -\sum_{i=1}^{n} \log p\left(y_i | U, Y_{-i}, \eta\right). \tag{4-15}$$

Similar to the marginal likelihood method, the estimation of the hyperparameters by (4-15) is a non-convex problem. In the literature there are many different versions of the cross-validation method, one of which is the LOO method. It is beyond the scope of this thesis to fully describe the available methods, but a concise introduction for the parametric case can be found in [46]. It is also worth mentioning that in the same family belongs the Generalized Cross-Validation (GCV) method, which also appeared in a series of publications for the regularization of the LPV SID methods [30, 50]. However, in the nonparametric case that we discuss here, it is known that GCV is not expected to yield good results due to the fact that it neglects the information regarding the covariance of the estimates, since it minimizes a weighted squared error between the measured y and the estimated $\mathbb{E}\left[f(U_*)|U, U_*, Y\right]$ (for an analytic treatment of the GCV method in the parametric case see [51] [26, p. 307]).

4-3 Conclusion

Gaussian processes offer an attractive framework for the nonparametric estimation of linear or nonlinear functions. In this brief overview we have highlighted the role of hyperparameter selection in order to obtain a function that can approximate well the underlying system. The two most prominent methods for the estimation of the hyperparameters, namely the marginal likelihood and the cross validation (such as the LOO method), were also discussed, while it was pointed out that the related optimization routines are non-convex. In the next parts of this dissertation we will show how this framework can be incorporated in the Subspace Identification (SID) algorithms. More specifically, it will be shown how this synergy between Gaussian processes and system identification can improve the accuracy of the estimated models, while the cost to be paid for this is an increase in the complexity of the algorithm.

Part II

Kernel methods for the identification of LTI systems

Chapter 5

Kernel based regularization for Linear Time Invariant (LTI) systems

In this chapter we will investigate the role of regularization in the accuracy of the identified LTI model with the use of PBSID$_{opt}$ algorithm. The chapter is organized as follows. In Section 5-1 we will illustrate how the solution of the VARX based Least Squares (LS) problem is affected by the selection of the past window p, keeping in mind the way of reasoning at the end of Section 2-3. In order to alleviate this problem, in the next section we will introduce the notion of regularization and we will establish the link between regularization methods and Prediction Error Identification (PEI) following the frequentist perspective. It will also be shown how the incorporation of kernels can lead to the optimal identification (in Mean Squared Error (MSE) sense) of the coefficients in a linear regression problem. In Section 5-3 we will follow a different perspective, namely a Gaussian process perspective to investigate the derivation of the estimator in terms of a posteriori distributions. The practical construction and selection of the proper kernels is another important topic that is covered in the same section. After this point, in Section 5-4 we will move from the kernel based PEI methods towards the kernel based SID methods, while in the last section we will discuss about the similarities and differences between the two kernel based identification methods.

5-1 Limitations of the classical PBSID$_{opt}$ method

As it was discussed in Section 2-3, the value of the past window p plays a significant role regarding the accuracy of the identified model. One of the main problems related to this quantity is that the engineer cannot deduce what should be the value of p, such that the trade-off between the error in the approximation $\tilde{A}^p \approx 0$ and the effort to avoid overfitting is well balanced. To illustrate this idea, consider the following example.

Example 5.1. In this example we will use the (without any regularization) PBSID$_{opt}$ algorithm, implemented in the PBSID toolbox [52], to identify a 4th order, 2-inputs 1-output

system. The system, given the innovation form, is simulated with Signal-to-Noise Ratio (SNR)= 5.3 [1], and the length of the collected data is 400. The system matrices are given in Example 1 of [52] with the difference that we removed the second output. To show how the aforementioned trade-off shapes the final estimate, we present in Table 5-1 the Variance Accounted For (VAF) results when the true Markov parameters are used (thus skipping the VARX estimation step), as well as the quantity $\left|\left|C\tilde{A}^p \begin{bmatrix} x_{p+1} & \dots & x_{400} \end{bmatrix}\right|\right|_2$. The quantity $C\tilde{A}^p x_k$ is the approximation error when the true coefficients are used, see also (2-11). Moreover, we give in the same table the VAF results of the standard PBSID$_{opt}$ algorithm. The simulations were performed for different past window values p, while the future window was kept constant, equal to $f = 5$. The VAF is given by the equation [11]

$$\text{VAF}(y_k, \hat{y}_k) = \max\left\{1 - \frac{\sum_{k=1}^{N} ||y_k - \hat{y}_k||_2^2}{\sum_{k=1}^{N} ||y_k||_2^2}, 0\right\} 100\%. \tag{5-1}$$

Table 5-1: VAF results and norm of the approximation error for different past window values p

p	5	10	25	50	100
VAF, true coef.	92.32 %	95.18 %	98.97 %	99.32 %	99.73 %
Approx. error norm	1030.62	63.70	6.49	0.29	0.00049
VAF, standard PBSID$_{opt}$	95.91 %	97.78 %	97.60 %	97.35 %	0 %

From this table it becomes clear that the performance of the PBSID$_{opt}$ algorithm, when the true Markov parameters are used, is improving as the past window value p gets larger. This is of course reasonable, since it means that more information is taken into account. On the other hand, the standard PBSID$_{opt}$ shows a peak value for $p = 10$, while for larger p values its performance acutely drops. This demonstrates in a clear way the importance of choosing a past window value that avoids both a large bias error as well as the overfitting problem.

As it evident from the previous example, the aforementioned trade-off is crucial for the successful identification of the unknown LTI system. A large past window means that the approximation error is small, so the **algorithm is capable** of estimating correctly the unknown coefficients. However, this capability is hindered by the problem of overfitting. The question arises naturally: is there a way to combine the benefits of a large past window without the problem of overfitting? As we will explain in the following sections, this answer lies in the regularization methods and especially in the kernel based regularization.

[1] The SNR is given by SNR $= \frac{\text{var}(y)}{\text{var}(e)}$, where var$(y)$ denotes the variance of the output, corrupted by noise and var(e) is the noise variance. Usually it is expressed in dB. In this case $\text{SNR}_{dB} = 10 \log_{10} \frac{\text{var}(y)}{\text{var}(e)}$

5-2 Regularization in LS problems

The bias-variance trade-off

Regularization is a standard way to influence the solution of a LS problem, for example by putting a larger significance is some quantities than others [26]. Moreover, regularization can be used to derive numerically robust solutions in ill-posed problems, such as the estimation of the (infinite) impulse response of an LTI system with a finite number of observations or the inversion of a matrix with large condition number [2].

A very informative perspective on the regularization problem is in terms of the MSE. MSE is broadly used in the area of statistics, since it gives information about the accuracy of the estimator[3] both in terms of variance and bias. Mathematically, the MSE of an estimator θ of a deterministic parameter θ_0 is given by

$$\begin{aligned} MSE(\theta) &= \mathbb{E}\left[(\theta-\theta_0)^2\right] \\ &= \left(\mathbb{E}\left[\theta\right]-\theta_0\right)^2 + \mathbb{E}\left[\left(\theta-\mathbb{E}\left[\theta\right]\right)^2\right] \\ &= \text{Bias}^2 + \text{Variance}, \end{aligned} \tag{5-2}$$

while in the case where θ_0 is a random variable, only the first equation of (5-2) holds. Before we give a mathematical description of the regularization methods, let us present an intuitive example.

Example 5.2. As a simple example, consider the Figure 5-1. In this hypothetical experiment, we are trying to estimate two variables, namely $\theta_1, \theta_2 \in \mathbb{R}$ (the number of the coefficients was chosen to enable a graphical representation of the MSE value). On the left side we present the (hypothetical) estimates of the unknown deterministic variables θ_1, θ_2, assuming that the experiment has been repeated multiple times and each experiment delivered a different estimate of them. In this case, the bias is rather small, since the average of the estimates is close to θ_1, θ_2 . However, it is obvious that the variance of the estimates is high. So, in terms of MSE, it would be rather fair to seek for a better estimator. On the other hand, in the right figure we present the opposite case: an estimate with relatively large bias (due to the distance of the real, red cross and the mean of the estimates, represented by the green cross) and low variance.

In total, if we were only based on the mean value or only on the variance of the estimator, we could end up with false conclusion. On the other hand, the MSE is taking both bias and variance into account, thus enabling a mathematically fair justification of what is a "good" or "bad" estimator.

[2]The condition number is defined as the ratio $\frac{\sigma_{\max}}{\sigma_{\min}}$, where $\sigma_{\max}$ denotes the largest singular value of a matrix and $\sigma_{\min}$ the smallest one [26]. In problems such as LS, one of the steps requires the inversion of a matrix of the form A^TA. If this matrix shows a large condition number then the derived solution may be inaccurate.

[3]The estimator should not be confused with the estimate, which is the result of the estimator. See also [53, Ch.2].

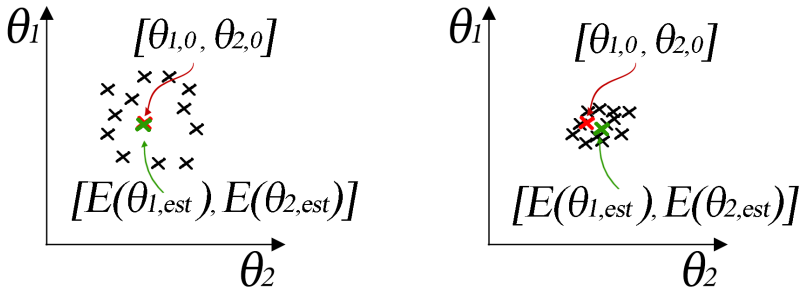


Figure 5-1: Estimates of two variables θ_1, θ_2. Left: a low bias, high variance estimate. Right: a high bias, low variance estimate. The black crosses represent the estimates from different experiments. The green ones represent the mean value of these estimates and the red ones the actual values of the variables.

A mathematical framework for regularization

The discussion about the MSE value reflects the underlying bias-variance trade-off. It is therefore of high importance to find a way to balance between these two antagonistic characteristics of the estimate. In order to do so, we have to develop the corresponding mathematical framework. For simplicity, let us assume that the data generating system is given by

$$Y = \theta_0 Z + E, \tag{5-3}$$

where $Y \in \mathbb{R}^N$ is the measured output of the system, $\theta_0 \in \mathbb{R}^p$ are the unknown coefficients and $Z \in \mathbb{R}^{p \times N}$ is a known matrix (e.g. in the ARX model it contains the past input and output data formed in a block Toeplitz or Hankel matrix form). $E \in \mathbb{R}^N$ are the N samples of a Gaussian distributed zero-mean white-noise sequence with variance σ^2. The same assumptions for the noise will **hold for the rest of this dissertation**, unless otherwise stated.

Let us assume that the model used for estimation is described by

$$\hat{Y} = \theta Z, \tag{5-4}$$

where $\hat{Y} \in \mathbb{R}^N$ are the estimated outputs of the system and $\theta \in \mathbb{R}^p$, that is to say, we assume that **we have chosen the correct number of coefficients**. Then, the standard LS problem and the solution are given by [11]

$$\begin{aligned} &\min_{\theta} ||Y - \theta Z|| \Rightarrow \\ &\theta = Y Z^T \left(Z Z^T \right)^{-1} \end{aligned} \tag{5-5}$$

The estimate in (5-5) is characterized by a very attractive asymptotic property, namely the fact that **there is no better unbiased estimate** due to the Cramèr-Rao limit. This can

be verified by observing that the Maximum Likelihood (ML) estimation of θ coincides with the LS estimation under the aforementioned noise assumptions [11, p. 120]. It is trivial to show that indeed θ is unbiased by computing the quantity $\mathbb{E}[\theta - \theta_0]$.

In the general setting, the regularized equivalent of the LS problem in (5-5) is expressed as

$$\min_{\theta} ||Y - \theta Z|| + \mathcal{J}(\theta). \tag{5-6}$$

The term $\mathcal{J}(\theta)$ can represent any form of regularization, such as the nuclear-norm regularization or Tikhonov-type. In case of a Tikhonov regularization (5-6) can be written as

$$\min_{\theta} ||Y - \theta Z||_2^2 + \gamma \theta D^{-1} \theta^T, \quad D \succcurlyeq 0, \quad \gamma \geq 0, \tag{5-7}$$

where γ and D are the regularization parameter and matrix, respectively, while the symbol $\succcurlyeq$ denotes an inequality in the matrix sense, that is to say, D is assumed to be positive semi-definite.

In this **classical (or frequentist) perspective** we aim at minimizing the MSE value. Compared to the solution in (5-5), it is to our hope that by enabling an amount of bias in the final estimate, we will manage to reduce the associated variance, eventually reducing the MSE [4]. This is exactly the role of the regularization parameter in (5-7). In order to show this, we compute first the solution of (5-7). With the use of the matrix inversion lemma [53], we find that

$$\theta = Y \left(Z^T D Z + \gamma^2 I \right)^{-1} Z D, \tag{5-8}$$

for which it is evident now that $\mathbb{E}[\theta - \theta_0] \neq 0$.

Moreover, it is interesting to note that the same result as the one in (5-8) can also be derived with the use of a **Bayesian framework** by treating θ as a normally distributed random variable, that is to say, $\theta \sim \mathcal{N}(0, P)$. The steps to be taken are **similar** to the ones described for the derivation of the posterior estimates of Gaussian processes in Chapter 4 and for this reason will be omitted from this chapter, however they can be found in Appendix C. Based on this analysis, one can easily deduce that (C-9) and (5-8) are equivalent for $\gamma = \sigma$ and $D = P$.

Estimation of the kernel parameters

In the classical (frequentist) approach, certain selections are being accomplished with the use of the Cross-Validation (CV) criterion. For example, these variables can be the number of the coefficients in θ (in case that we do not know the exact number), or the parameters γ and D. More specifically for D, this is usually expressed in a parametric form so we will write it as $D(\eta)$, where η denotes the collection of all the required hyperparameters to describe D. The CV criterion in the LS setting is being carried out as follows [54] (this approach can be seen as a category of the methods discussed in Section 4-2) :

[4] It is important to notice that the sample MSE value is given by $\hat{\text{MSE}} = \frac{1}{N}||Y - \hat{Y}||_2^2$. By comparing it with the VAF definition in (5-1) it is obvious that the two criteria show a strong resemblance. Consequently the reduction of the MSE value means that the VAF value is increasing.

- Split the data in one estimation and one validation part.
- Estimate θ for different selections of the variables, as it was discussed above.
- For each selection, compute the error between measured and model outputs (e.g. using the quadratic criterion):

$$J = ||Y - \hat{Y}||_2^2 \tag{5-9}$$

It is rather obvious that the steps in the previous algorithm describe a cumbersome procedure for the estimation of the unknown parameters η, γ or the number of θ parameters (the order of the system), since the procedure has to be repeated for each fixed selection, practically forcing us to limit the search in only some fixed values on a selected grid. Moreover, it is also mathematically justified that the CV method may lead to poor estimations [55]. For these reasons the classical approach for selecting these parameters is not preferred. A viable alternative is the maximization of the log marginal likelihood given in (C-5). This opens the road for a fully Bayesian approach, which finally makes the Gaussian process regression framework more attractive, as we will show in Section 5-3.

Regularization meets system identification

The reduction of the MSE value by introducing bias was well known in the statistical community for over 20 years (e.g. [56]). However, in the system identification community it was less used, while in most times its usability was restricted in rendering the inversion of a possibly ill-conditioned matrix (such as $Z^T Z$ in (5-5)) numerically stable. It was only recently when the interest in the regularization methods was revived and, in parallel to a Gaussian regression framework (which will be presented in the next section), it led to new regularization techniques, which were observed to be capable of delivering more accurate estimates than the classical PEI/ML framework [6, 57].

The basic attribute of these techniques, which are mainly investigated in the PEI framework, is that they are trying to incorporate simple prior information about the underlying system in the regularization parameters. In the last two years numerous papers have appeared, which investigate the various aspects of these algorithms [54, 58–67].

Most of the related publications mainly focus either on an Finite Impulse Response (FIR) or an Auto-Regressive with Exogenous Input (ARX) model structure. The first choice is justified by the fact that the newly proposed methods are trying to incorporate information about the impulse response of the unknown system (and so an FIR is a natural choice), while the latter one is based on the fact that an ARX model can approximate arbitrarily well any LTI system when its order tends to infinity [25]. Moreover, since both methods do not require any non-convex optimization step, it renders them even more attractive.

In this dissertation, due to the direct relation with the VARX step in the PBSID$_{opt}$ algorithm, we will consider the ARX model case. In this case it is straightforward to establish the relation between the quantities in (5-3) and the ARX model. More specifically, by borrowing the notation from (2-12), the following relations hold.

$$\theta_0 = \begin{bmatrix} h_1^u & h_2^u & \cdots & h_p^u & h_1^y & h_2^y & \cdots & h_p^y \end{bmatrix}$$
$$Z = \begin{bmatrix} U_p \\ Y_p \end{bmatrix} \tag{5-10}$$

It is therefore obvious that the previous discussion about the regularization of LS problems can be directly implemented to the ARX case. The next challenge that we face now is how we will manage to balance the bias-variance trade-off. To this end, the optimal regularization can be used as a guide.

Remark 5.1. In (5-10) we assumed that the number of the h^u and h^y coefficients is the same. This is by no means a general assumption for the ARX models [10], so the two number may differ from each other, depending on the specific problem. On the other hand, this is always the case in the VARX formulation used in SID methods, which are in the center of our attention on this dissertation and so it is more convenient to use the same number even in the ARX case.

Optimal regularization

To define the optimal regularization parameter, let us first compute the MSE value for the estimate (5-8). For the optimal estimate **it is necessary to assume that it is of the same order as** θ_0 [54]. First let us derive a general expression for the MSE value.

Lemma 5.1. The MSE value for the estimate θ in (5-8) is given by

$$MSE(\theta)(D) = \left(ZZ^T + \gamma D^{-1}\right)^{-1} \left(\sigma^2 ZZ^T + \gamma^2 D^{-1}\theta_0^T\theta_0 D^{-1}\right) \left(ZZ^T + \gamma D^{-1}\right)^{-1}. \tag{5-11}$$

Proof 5.1. The proof can be derived by using the definition of MSE in (5-2) together with (5-5) and (5-8). □

Now we are in position to define the optimal regularization, which is given in the following lemma.

Lemma 5.2. The following inequality holds for the MSE value, given the selection $\gamma = \sigma$:

$$\text{MSE}(\theta)(D) \geq \text{MSE}(\theta)(\theta_0^T\theta_0). \tag{5-12}$$

Proof 5.2. The proof is given in the appendix of [54]. □

The corresponding estimate is expressed in the following way to avoid ill-condition.

$$\theta_{opt} = Y\left(\sigma^2 I + Z^T\theta_0^T\theta_0 Z\right)^{-1} Z^T\theta_0^T\theta_0. \tag{5-13}$$

As it would be expected, the optimal MSE value is related to the "true" system parameters θ_0, which in a real life problem are of course unknown. However, this result is still useful in the sense that it gives a hint on how we should choose the regularization parameters.

Finally, for the better explanation of the results we have to investigate how close the estimated optimal θ_{opt} is to the θ_0 value. In other words, finding an optimal estimate is itself not sufficient unless if it is close to the real value of the variable. By substituting (5-3) in (5-13), we have that

$$\theta_{opt} = (\theta_0 Z + E)\left(\sigma^2 I + Z^T\theta_0^T\theta_0 Z\right)^{-1} Z^T\theta_0^T\theta_0 \Rightarrow \tag{5-14}$$

$$\begin{aligned}\mathbb{E}\left[\theta_{opt}\right] &= (\theta_0 Z)\left(\sigma^2 I + Z^T\theta_0^T\theta_0 Z\right)^{-1} Z^T\theta_0^T\theta_0 \\ &= (\theta_0 Z)\left(\sigma^2 I + Z^T\theta_0^T\theta_0 Z\right)^{-1} Z^T\theta_0^T\theta_0 Z Z^T (ZZ^T)^{-1},\end{aligned} \tag{5-15}$$

where we assumed that the matrix ZZ^T is full rank. From the expression above it becomes obvious that

$$\mathbb{E}\left[\theta_{opt}\right] \approx \theta_0 \text{ as } \sigma^2 \to 0 \text{ or } Z^T\theta_0^T\theta_0 Z >> \sigma^2 I \tag{5-16}$$

5-3 Gaussian processes meet system identification

Another interesting point of view is closely related to the Gaussian process regression framework, which was also presented in Chapter 4. In order to describe this approach, let us start with an ARX model, using the assumptions and the notation from (2-9), which is repeated here for convenience.

$$y_k = \sum_{t=1}^{\infty} h_t^u u_{k-t} + \sum_{t=1}^{\infty} h_t^y y_{k-t} + e_k. \tag{5-17}$$

In the Gaussian process framework we treat the impulse response coefficients h_t^u and h_t^y as zero-mean Gaussian processes, defined over the set of positive natural numbers, $t \in \mathbb{N}^+$. They are assumed to be mutually independent, as well as independent of the noise e_k [7,59]. Their covariance (kernel) is described by the equations

$$\begin{aligned}&\text{cov}(h_{t_1}^u, h_{t_2}^u) = k^u(t_1, t_2), \\ &\text{cov}(h_{t_1}^y, h_{t_2}^y) = k^y(t_1, t_2), \\ &\text{cov}(h_{t_1}^u, h_{t_2}^y) = 0, \text{ for every } t_1, t_2 \in \mathbb{N}^+,\end{aligned} \tag{5-18}$$

where $k(\cdot,\cdot)$ is a function : $\mathbb{N}^+ \times \mathbb{N}^+ \to \mathbb{R}$. Again, we let the function k be parametrized by some hyperparameters η_u and η_y, so a more accurate notation of this function would be as $k(\cdot,\cdot;\eta_u)$ (similarly for the output's kernel). Nonetheless, in order to simplify the notation, this dependency will usually be omitted.

A first observation regarding this framework is that it resembles the Bayesian interpretation of the regularization methods, as it was discussed in Section 5-2 and further described in Appendix C. Consequently, the Bayesian point of view arises naturally in this case. Therefore, the hyperparameters η can be derived by solving the non-convex optimization problem, that

is to say, by minimizing the minus log marginal likelihood $-\log p\left(Y|\eta_u, \eta_y, U_p, Y_p\right)$, given by (C-5). Moreover, it is usually the case that the noise variance, involved in (C-5), is not known and so it can be seen as a hyperparameter and be estimated together with the other hyperparameters via the marginal likelihood algorithm. However, a different paradigm can be used, following the remarks in [68]. By using a low-bias estimate, that is to say a model that contains a relatively high number of coefficients, it was observed that the sample variance of noise given by

$$\hat{\sigma}^2 \approx \frac{1}{N-1}\sum_{k=1}^{N}\left(\hat{y}_k - y_k\right)^2 \tag{5-19}$$

can be used as an estimation of σ^2.

After having estimated the hyperparameters, the mean and variance of the posterior distributions of h^u and h^y can be derived. Following a MAP approach, we finally end up with the estimates of h^u and h^y by setting their values to be equal to their average values $\mathbb{E}\left[h^u|Y, Y_p, U_p, \eta_u, \eta_y\right]$, $\mathbb{E}\left[h^y|Y, Y_p, U_p, \eta_u, \eta_y\right]$, see also (C-9).

Remark 5.2. Under some specific assumptions, a connection can be established between the MSE and the minimizer of the minus log marginal likelihood value. More specifically, under some assumptions to ensure a unique SVD decomposition of Z, as well as the assumption that the matrix Z^TZ converges to a constant value (see also the discussion in Section 5-3) and the selection of a diagonal kernel, that is to say, $\mathbb{E}\left[\theta^T\theta\right] = \lambda I$, it can be shown that the value that minimizes the minus log marginal likelihood converges asymptotically to a scaled version of the MSE. Therefore, Aravkin proposed that the latter one can be used for the estimation of the hyperparameters [60]. To overcome the dependency of the MSE on the real parameters of the system (see also (5-11)), he proposed to set their values to the ones estimated by an unregularized LS problem. Nonetheless, based on simulation results it was shown that this new algorithm is not superior to the Marginal Likelihood. Some other theoretical perspectives of the proposed algorithm are given in [69].

Relation between Gaussian process framework for System Identification (SysID) and Reproducing Kernel Hilbert Space (RKHS) theory

In this section we will reveal the links between the Gaussian process modelling for SysID, described in the previous section, and the RKHS theory. To this end, we have to make some useful remarks. First of all, here we notice that now the various impulse response instants, e.g. h_1^u, h_2^u etc. are not treated as coefficients but instead **they are treated as functions**, evaluated at the points $t = 1, t = 2$ correspondingly. For this reason, it is appealing to explicitly write them as a function of t, that is to say, $h^u(t) := h_t^u$ (similarly for h_t^y). Due to the Bayesian approach that is more natural to use in this case, the discussion in terms of LS problems is replaced by the discussion about the minimum variance estimate (or the Maximum a Posteriori (MAP) estimate or the mean of the posterior estimate in the Gaussian case, since all of them coincide).

The relation with a Tikhonov type LS problem can eventually be established. The tool for this is the RKHS theory, introduced earlier in Section 4-1 and further elaborated in Appendix D for the nonparametric case.

For the present problem formulation, the combination of a nonparametric part (namely h^u and h^y) with a parametric one (the past input and output data) creates no complications. Under the assumption that h^u and h^y are continuous functionals in the corresponding (possibly infinite dimensional) Hilbert spaces $\mathcal{H}_u, \mathcal{H}_y$, then similar to the nonparametric case, we can express them as a function of a finite number of coefficients.

In order to show how this can be accomplished, we need first to truncate the impulse response such that a finite number of impulse response instants are used, making sure that it is large enough to capture well the impulse response of the underlying system. This number will again be denoted by p, the past window. Using the results from the RKHS theory, we can express h^u and h^y with the use of only p basis functions. The corresponding equations are given by

$$\begin{aligned} h^u(t) &= \sum_{i=1}^{p} \alpha_i^u k^u(t,i), \quad \alpha_i^u \in \mathbb{R} \\ h^y(t) &= \sum_{i=1}^{p} \alpha_i^y k^y(t,i), \quad \alpha_i^y \in \mathbb{R} \end{aligned} \tag{5-20}$$

Based on this approach, we can stack the values of the functions, evaluated at $t = \{1, 2, \ldots, p\}$ in a row vector, so

$$\begin{bmatrix} h^u(1) & h^u(2) & \cdots & h^u(p) \end{bmatrix} = \begin{bmatrix} \alpha_1^u & \alpha_2^u & \cdots & \alpha_p^u \end{bmatrix} \begin{bmatrix} k^u(1,1) & k^u(1,2) & \cdots & k^u(1,p) \\ k^u(2,1) & k^u(2,2) & \cdots & k^u(2,p) \\ \vdots & \cdots & \cdots & \vdots \\ k^u(p,1) & k^u(p,2) & \cdots & k^u(p,p) \end{bmatrix}$$
$$H^u = A^u K^u, \tag{5-21}$$

where the matrices K^u and K^y should be by definition positive semi-definite, due to the fact that they are covariances matrices.

Making use of (5-20) and (5-21), we formulate the following Tikhonov LS problem:

$$||Y - H^u U_p - H^y Y_p||_2^2 + \sigma^2 ||H^u||_{\mathcal{H}_u}^2 + \sigma^2 ||H^y||_{\mathcal{H}_y}^2, \tag{5-22}$$

where, in general, $||\cdot||_{\mathcal{H}_z}$ denotes the norm associated with the Hilbert space $\mathcal{H}_z$. This norm can be analytically computed (see Appendix D). For example, for H^u (similarly for H^y) it is given as

$$||H^u||_{\mathcal{H}_u}^2 = A^u K^u A^{uT}. \tag{5-23}$$

We have thus lifted the problem of estimating H^u and H^y to a LS problem of estimating A^u and A^y. Now it is straightforward to compute the solution of the LS problem (5-22) with respect to A^u and A^y. The corresponding equations are given by

$$\begin{aligned} A^u &= Y \left(Y_p^T K^y Y_p + U_p^T K^u U_p + \sigma^2 I \right) U_p^T \\ A^y &= Y \left(Y_p^T K^y Y_p + U_p^T K^u U_p + \sigma^2 I \right) Y_p^T \end{aligned} \tag{5-24}$$

and so we can directly compute H^u and H^y with the use of (5-21). It is therefore trivial to show that the solution derived with the use of the RKHS theory coincides with the one derived with the use of the regularization method discussed in the previous section, see also (C-9). However, as we will see in the next part of this dissertation, the RKHS framework can offer more insight in specific cases.

Remark 5.3. The extension to the MIMO case can be accomplished in a straightforward manner, following the same hypotheses that were stated in this section.

Kernels that incorporate information about the underlying LTI system

Until now we have described the mathematical tools that are used in the kernel methods and we have also seen what is the optimal regularization kernel. However, we still have to answer an important question regarding the kernel themselves. As we already outlined earlier, these kernels are usually described as a function of some specific hyperparameters, denoted by η. We have also reasoned why the Bayesian framework (and consequently, the Gaussian process regression framework) provides an attractive way to compute these hyperparameters through the marginal likelihood.

The analysis so far in this part was rather analytic, since the framework described here is also important for the ideas developed in the next part of this dissertation. As far as the kernels for the LTI case are concerned, there is currently a huge work being done and so it is rather fair to say that the best selection of a kernel is still an open question and the various ideas are not yet brought to a common ground.

Moreover, due to the different origins of the Gaussian process regression [59] and the regularization in LS framework for the identification of LTI systems, the kernels proposed for each method are different but with strong resemblances. For the reasons discussed above, in this framework we will focus on the kernels for the Gaussian regression approach (with the only exception of the diagonal kernel), but an interested user can find information about the kernels for the regularized LS problems in [54, 63].

A first approach to the kernel selection problem

Maybe the simplest kernel that can incorporate prior knowledge about the underlying system is the so-called **diagonal kernel**. Despite the fact that the followed way of reasoning for the derivation of this kernel is based on the frequentist approach, it is still very useful to understand the rationale behind the construction of kernels. First let us assume that the underlying LTI system admits an FIR structure and so does the model used for the identification. Using (5-11) as a starting point, let as assume that the input u is being generated by a zero-mean white-noise random sequence with variance equal to s. Then it follows that $Z = U_p$. It is well known (e.g. [11]) that the following condition holds.

$$\lim_{N\to\infty} \frac{1}{N-p} ZZ^T = sI \tag{5-25}$$

Let as assume that $\gamma D^{-1} = diag(l_1, l_2, \ldots, l_p)$. Then, by employing (5-2), the n^{th} diagonal element of the MSE is expressed as

$$MSE(n,n) = \frac{\sigma^2 s(N-p) + l_n (h_n^u)^2}{(s(N-p) + h_n^u)^2}, \tag{5-26}$$

which is minimized with respect to l_n for $l_n = \sigma^2/(h_n^u)^2$ [54]. Consequently, if the system is exponentially stable, the coefficients should decay exponentially, so we can choose $l_n = \sigma^2/(\lambda\alpha^n)$, where $\lambda > 0$ and $0 < \alpha < 1$. The corresponding γ coefficients can then be chosen to be equal to σ^2 and we also define $D = \text{diag}\left(\lambda\alpha, \lambda\alpha^2, \ldots, \lambda\alpha^p\right)$. The two parameters λ and α are of course unknown and have to be estimated through the log Marginal Likelihood (MargLik) algorithm. The exponential decay of the diagonal terms is also verified in Figure 5-2.

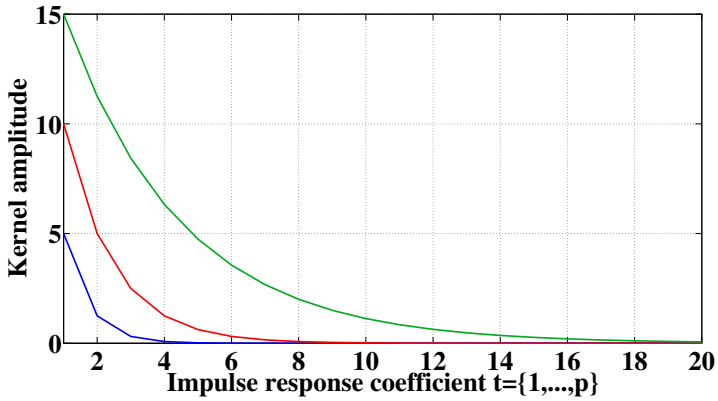


Figure 5-2: Diagonal elements of the diagonal kernel with exponential decay with $\lambda = 20$ and past window $p = 20$. The three plots correspond to $\alpha = 0.25$ (blue), $\alpha = 0.5$ (red) and $\alpha = 075$ (green)

Remark 5.4. The results for the FIR case cannot be straightforwardly extended to the ARX case, due to the fact that the condition that (5-25) does not hold for the output matrix $Y_p Y_p^T$. However, in practice it is observed that following the same approach for an ARX model may also yield satisfactory results. This claim is also investigated in Chapter 6.

Stable spline kernels

The diagonal kernel is a first approach to the optimization problem, which is expected to improve the accuracy of the identified model. Nonetheless, in the very recent years the so-called stable spline kernels gained a high interest following a stochastic approach, while their origins date back in the work of Wahba on cubic smoothing splines [70]. The stable spline kernels are characterized by the fact that they do not only incorporate information about the smoothness of the function, but also information about the stability of the LTI system. The theory of the stable spline kernels is itself pretty rich, while many aspects of this family of kernels is currently in the center of attention of the related scientific community. It is also worth stressing that the related framework was initially developed for continuous time functions, but its implementation in the discrete time case is also possible.

Depending on the number of the absolutely continuous derivatives of a spline function, two specific choices are the most common. If the function has no continuous derivatives or only one continuous derivative, it means that the corresponding spline constitutes of 2nd or 3rd order polynomials, correspondingly. The regularization term, defined over the RKHS with $x \in \mathcal{X} = [0, 1]$ is described by

$$||g||_{\mathcal{H}}^2 = \int_0^1 \left(g^{(\zeta)}(x)\right)^2 dx \tag{5-27}$$

and the corresponding kernel is given by

$$K\left(e^{-\beta s}, e^{-\beta t}\right) = \int_0^1 \frac{\left(e^{-\beta s} - u\right)_+^{\zeta-1}}{(\zeta-1)!} \frac{\left(e^{-\beta t} - u\right)_+^{\zeta-1}}{(\zeta-1)!} du, \quad u_+ = \begin{cases} u, & \text{if } u \geq 0 \\ 0 \text{ otherwise} \end{cases} \tag{5-28}$$

while the notation $g^{(\zeta)}$ denotes the ζ^{th} derivative. Under the choices of $\zeta = 1$ and $\zeta = 2$ we get the **1st order and the 2nd order stable spline kernel**, correspondingly.

$$K\left(e^{-\beta s}, e^{-\beta t}\right) = e^{-\beta \max\{s,t\}}, \quad \text{for } \zeta = 1, \tag{5-29}$$

$$K\left(e^{-\beta s}, e^{-\beta t}\right) = \frac{e^{-\beta(s+t+\max(s,t))}}{2} - \frac{e^{-3\beta \max(s,t)}}{6}, \quad \text{for } \zeta = 2 \tag{5-30}$$

The increased smoothness of the kernel in (5-30) compared with the kernel in (5-29) can also be verified by the surf plots in Figure 5-3. The 1st order stable spline exhibits a square shape of decay, while the 2nd order stable spline resembles a smoother, circle-like decay as we move from the top left to the lower right element.

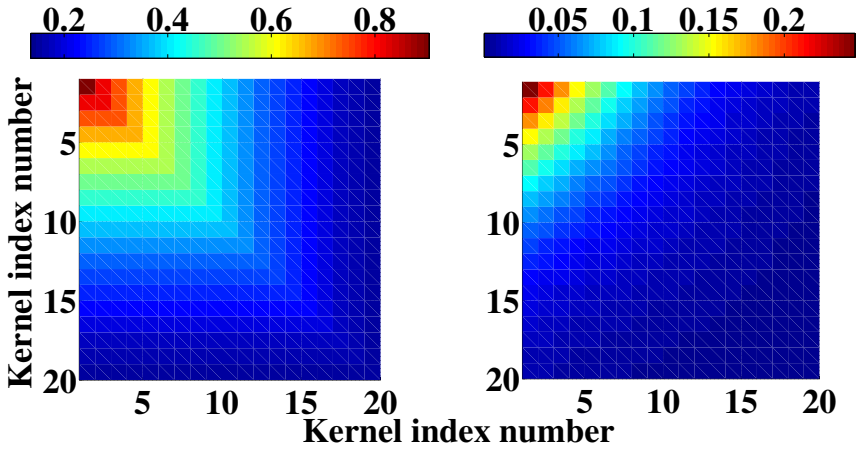


Figure 5-3: Surf plots of the 1st order stable spline kernel (left figure) and the 2nd order stable spline (right figure). The chosen β value is 0.1 and the size of the kernel is 20.

Moreover, another kernel that will be employed in this thesis is a modification of the stable spline kernels, the so-called **high frequency (HF) stable spline** [62]. The main purpose of this kernel is to capture highly oscillating dynamics. In terms of kernel structure this is translated to a negative correlation between adjacent elements of the covariance matrix (kernel), e.g. the kernel K^u. It is built based on the 1st order stable spline, so it is described by

$$K\left(e^{-\beta s}, e^{-\beta t}\right) = \begin{cases} e^{-\beta \max\{s,t\}} & \text{if } s+t \text{ is even,} \\ -e^{-\beta \max\{s,t\}} & \text{if } s+t \text{ is odd} \end{cases} \tag{5-31}$$

A better view of this kernel can also be offered by comparing it with the two other stable spline kernels. By taking the first line of each kernel it is possible to plot the three kernels in a 2-D plot, as it is shown in Figure 5-4. In this plot it becomes evident that the HF stable spline kernel exhibits a highly oscillatory form, compared to the other two kernels.

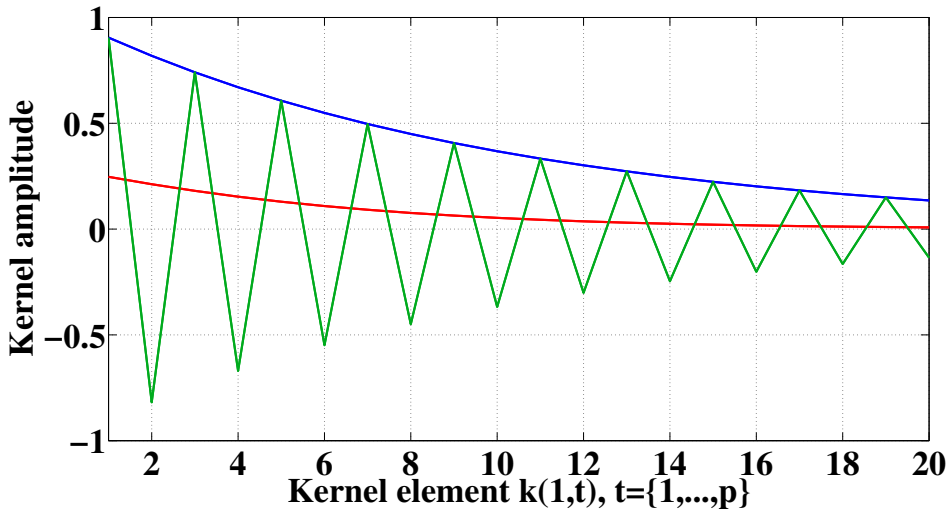


Figure 5-4: 1st row of the kernels that correspond to the 1st order stable spline (blue), 2nd order stable spline (red) and HF stable spline (green). The chosen β value is 0.1 and the size of the kernel is 20.

Remark 5.5. As we already mentioned, the theory of the stable spline kernels is much richer as it was presented here. An interested reader is referred to [58, 59, 71] for a more in-depth discussion on this subject.

Selecting the kernel structure

The different kernels incorporate different information about the underlying system. A diagonal kernel is a simple form kernel that accounts for the exponential decay of the impulse response coefficients but it neglects the possible cross-variance by setting it to be zero. On the other hand, the 1st and the 2nd order stable spline kernels set a prior distribution on the cross-terms too. The 1st order assumes that the impulse response coefficients exhibit a less smooth relation, which is reflected in the shape of the kernel, while the 2nd order is much

smoother, as it was also shown in Figure 5-3. Finally, in cases where the underlying system shows a highly oscillatory behaviour it is recommended to use the HF stable spline, which incorporates an oscillatory shape.

All in all, it becomes obvious that the kernel selection problem is still an open question, with many new approaches appearing up to this day. It would be rather fair to admit that the theoretical aspects of the proposed kernels are not fully grasped by the scientific community up to this point and for this reason a great effort to this direction is currently being made. Taking as a starting point of this new synergy between Gaussian processes and system identification the year 2008 [7], it is characteristic that since then the number of related publication is exponentially increasing. This research includes new kernel structures, such as multiple kernels to capture more complicated dynamics [64], as well as regularization techniques that combine the kernels methods with nuclear norm regularization [72]. Moreover, it can be the case that the introduction of a specific kernel is introducing further restrictions in the class of the systems that can interpret the data. As a solution to this problem, Pillonetto combined the kernel based non-parametric framework with a parametric part, such that the algorithm exhibits increased flexibility. However, the increased complexity of the final algorithm makes it rather uncomprehending and so no further investigation of this approach was reported in the literature, at least up to our knowledge.

In this dissertation it is not our aim to exhaustively investigate the various methods proposed in the scientific community, while even if this was the case, the high publication activity renders prohibitive for now any idea to review this field at the same time where the questions are still open. It is however our purpose to investigate how the general idea can be extended to the subspace identification framework for LTI systems, towards the incorporation of the kernel methods in the subspace identification of LPV state-space models.

5-4 Regularization of the VARX solution in the PBSID$_{opt}$ algorithm

As we have seen in Chapter 2, the state-of-the-art subspace identification algorithms for LTI systems involve a step where the Markov parameters of the unknown system are estimated via a Vector ARX (VARX) parametrization, leading to a LS problem. In the same chapter we explained that the accurate identification of these parameters plays a crucial role in the success of the algorithm. For this reason, the improvement of this estimation is a necessity towards the improvement of the PBSID$_{opt}$ algorithm. Ideally, the estimation of the impulse response coefficients is pursued in the VARX step, but in reality a trade-off has to be achieved such that the selected past window avoids the overfitting problem and simultaneously achieves the smallest bias, due to the approximation in the state term (2-11).

However, this difficult task can be circumvented by employing the kernel methods. More specifically, by taking advantage of the characteristics of the ARX model structure, we can introduce a kernel based regularization in LPV-PBSID$_{opt}$ algorithm in a way which is identical to the one for the kernel based PEI of ARX models, developed in the previous sections. This procedure is also briefly outlined in [8] and the accuracy of the algorithm is verified in four different systems, for which the length of the data is relatively small.

In this section we will only highlight what approaches can be followed to alleviate the computational burden of the kernel based PBSID$_{opt}$ algorithm and we will summarize the steps

that have to be followed.

Computational and practical aspects of the kernel based subspace identification

Until now we have given the general framework for the available kernels. In practice, though, the kernels should have more flexibility. More specifically, for the stable spline kernels family, the hyperparameter β and the structure of the kernel specify the general characteristics of the kernels, such as the smoothness or the exponential decay rate. Nonetheless, depending on the system to be identified, we have to scale these kernels up or down. In practice, this means that the kernels that describe the covariance of the impulse response coefficients are given by

$$\text{cov}\left(h_i^u, h_j^u\right) = \lambda_u^2 k^u(i, j; \beta_u), \tag{5-32}$$

where k_u can be any stable spline kernel element that we described above. On the other hand, the diagonal kernel that we described in the previous section already incorporates this scaling factor.

If we pre-estimate σ then the number of the unknown hyperparameters, in the general MIMO case is given by $2(n_u + n_y)n_y$, because we create one kernel for each signal and each kernel contains two hyperparameters, while this procedure has to be repeated for each output separately. It is obvious that the number of hyperparameters grows linearly with respect to n_u and quadratically with respect to n_y. It is therefore important to alleviate the computational burden, considering that these hyperparameters are estimated via a non-convex optimization problem.

To this end, let us consider the impulse response coefficients, as they were given in (2-10). For convenience we rewrite them here.

$$\begin{aligned} h_t^u &= C\tilde{A}^{t-1}B \\ h_t^y &= C\tilde{A}^{t-1}K. \end{aligned} \tag{5-33}$$

As we can see, the two impulse response coefficients share a common product, namely $C\tilde{A}^{t-1}$, and this is multiplied by the vectors that correspond to each signal. Remember that here we treat each signal separately, so in case of a MIMO system (where $B \in \mathbb{R}^{n \times n_u}$ and $K \in \mathbb{R}^{n \times n_y}$) the impulse response coefficients of each input (similarly for the outputs) will be multiplied by the corresponding column vector of the B matrix (correspondingly, of the K matrix).

For the total impulse response sequence of each signal we deduce that the exponential decay rate is mainly attributed to the $C\tilde{A}^{t-1}$ row vector, which is common among the various signals, while the B and K vectors (or matrices) can be seen as a scaling factor that scale up or down all the impulse response coefficients of a signal. Moreover, due to the fact that the C matrix is different for each output signal, it is preferable to repeat the estimation of the hyperparameters for each output signal.

However, with the previous observation we can partially reduce the computational complexity. For each output signal we can use the same β for all the involved impulse response sequences to capture the exponential rate of $C\tilde{A}^{t-1}$, $t \in [1, \ldots, p]$ and use a different λ_u (correspondingly,

λ_y) for each signal to capture the scaling facto of B (correspondingly, K). So, we will need n_y different β and $(n_u + n_y)n_y$ different λ for the stable spline kernels. A comparison of the number of the hyperparameters involved in this approach and the conventional one is also given in Table 5-2.

Table 5-2: Number of required hyperparameters for the conventional approach and the approach that requires less hyperparameters versus the number of the inputs and outputs. In each cell, the number on the left side corresponds to the former approach, while the number on the right side corresponds to the latter one.

nu \ny	1		2		5	
	Full par.	Reduced par.	Full par.	Reduced par.	Full par.	Reduced par.
1	4	3	12	8	60	35
2	6	4	16	10	70	40
5	12	7	28	16	100	55

Summary of the algorithm

Now we have described all the required steps to be taken for the implementation of the kernel based PBSID$_{opt}$ for LTI systems. These steps are summarized in the following algorithm.

Algorithm 5.1. Kernel based LTI-PBSID$_{opt}$

- Construct the quantities $Y = Y_{p+1,1,N-p}$, $Y_p = Y_{1,p,N-p}$ and $U_p = U_{1,p,N-p}$ based on (2-8)
- Choose a kernel structure such as a diagonal, or the ones described in (5-29), (5-30) or (5-31)
- By defining $Z = \begin{bmatrix} U_p^T \\ Y_p^T \end{bmatrix}^T$ and using the assumptions in (5-18) together with the assumption that the noise is independent of the impulse response coefficients, solve the non-convex problem (the minus log marginal likelihood)

$$\begin{aligned} \arg\min_{\eta} \left(-\log p\left((Y|Z,\eta)\right)\right) = \frac{N-p}{2}\log(2\pi) + \frac{1}{2}\log\left(\det\left(Z^T\mathcal{K}Z + \sigma^2 I\right)\right) \\ + \frac{1}{2}Y\left(Z^T\mathcal{K}Z + \sigma^2 I\right)^{-1}Y^T. \end{aligned} \tag{5-34}$$

where

$$\mathcal{K} = \begin{bmatrix} K^u & 0 \\ 0 & K^y \end{bmatrix}, \quad \mathcal{K} \in \mathbb{R}^{2p\times 2p}, \tag{5-35}$$

p denotes the past window, η contains all the related hyperparameters and the quantity K^u (similarly for K^y) is defined in (5-21). If the system has multiple inputs or/and outputs then the matrix (5-35) is extended in an obvious way, preserving its block diagonal form due to the assumptions in (5-18). This procedure has to be repeated for

each output signal. Moreover, if the noise variance σ is treated as a hyperparameter, then it can be estimated together with η, otherwise it can be pre-estimated, see also (5-19).

- Compute the MAP estimates by making use of (5-24) and (5-21).

5-5 Comparison of the PEI and SID kernel based identification

A necessary step towards the comprehension of the kernel based methods is to see how they are incorporated in the PEI and in SID methods. First of all, it is of high importance to clarify that the kernel based methods are used in cases where the estimator is linear in the parameters. For this reason the ARX and FIR structures are used. For both model structures the predictor can generally be expressed as

$$\hat{y}(k|k-1) = B(q^{-1})u(k) + \left(1 - A\left(q^{-1}\right)\right) y(k), \tag{5-36}$$

where $B(q^{-1}) = \sum_{t=0}^{\infty} b_t q^{-t}$ and $A(q^{-1}) = 1 + \sum_{t=1}^{\infty} a_t q^{-t}$ [10]. In the ARX case we have to substitute the infinite sums with the finite ones that correspond to the "real" system. In the FIR case the sequence A is equal to one and so the output term is removed, while B is also expressed by a finite number of coefficients.

It is nonetheless unclear if and when these model structures correspond to the "real" system. As far as the FIR case is concerned, someone would expect that the kernel based identification, being able to avoid the over-fitting problem, will manage to estimate more accurately the unknown system. Unfortunately, this way of reasoning is not theoretically justified, mainly due to the fact that it does not cover the case of coloured noise. However, the FIR model structure offers an interesting way to view the regularization problem, as we showed in Section 5-3.

On the other hand, the selection of the ARX model structure can be much better justified; in this case the result in [25] asserts that if the number of a and b coefficients tends to infinity, as shown in (5-34), and the number of data points N increases even faster than the ARX model is able to approximate arbitrarily well every LTI system. In practice, selecting a sufficiently large number of coefficients (the number of which in the kernel based methods does not reflect any more a bias-variance trade-off, as long as the number of the coefficients is large enough to capture the dynamics of the system [59]) results in a model capable of capturing in an accurate way the dynamics of the system.

The latter argument directly holds for the VARX step in the PBSID$_{opt}$ algorithm. However, **there is a crucial difference between the two methods**. Following a control purpose perspective, we are interested in estimating low complexity systems, capable of accurately estimating the dynamics of the system. For this reason, **in the PEI setting the introduction of a model reduction step is necessary** to end up with a low order system [59]. On the other hand, **this extra step is already incorporated in the subspace identification algorithms** since most of the algorithms use an SVD decomposition to select the most dominant singular values (see also Appendix A). In this sense, the kernel methods seem to be better suited for the subspace algorithms.

Another difference between kernel based PEI and SID methods is related to the allowed past window value. In the kernel based PEI methods it is usually stated that the past window (or the order of the system) can be very large, possibly larger than the number of data points. This choice in general does not create any overfitting problems due to the intrinsic regularization of the kernel methods. Consequently, the same assumption can also be made for the SID methods that employ kernel methods. Nonetheless, a large past window is not a safe choice for the SID methods (more specifically for the PBSID$_{opt}$ algorithm) due to the complications in the steps that follow the estimation of the Markov parameters.

The Singular Value Decomposition (SVD) performed after the calculation of the Markov parameters is used to deliver an estimation of the state sequence $X_{p+1,1,N-p}$ in the PBSID$_{opt}$ algorithm. As it was discussed in Appendix A, it is assumed that the state sequence is having full row rank n and the same holds for the extended observability (full column rank n), the controllability matrix (full row rank n) and the data matrix (full row rank $pn_u + pn_y$). It is important, though, to clarify that the rank property of the data matrix is not related any more to the accurate estimation of the Markov parameters (as it is the case in the classical PBSID$_{opt}$ algorithm). Based on these assumptions it is possible to consistently estimate the state of the system and the state sequence, up to a similarity transformation. However, when the number of parameters $p(n_u + n_y)$ is larger that the data points N the derived state sequence estimate is not necessarily of rank n. To see this, based on Sylvester's lemma [11], we have that

$$\begin{aligned}
\underbrace{X_{p+1,1,N-p}}_{n\times N} &= \underbrace{\tilde{\mathcal{K}}^{(p)}}_{n\times(n_u+n_y)p}\ \underbrace{Z_{1,p,N-p}}_{(n_u+n_y)p\times N} \Rightarrow \\
\operatorname{rank}(\tilde{\mathcal{K}}^{(p)}) + \operatorname{rank}(Z_{1,p,N-p}) - (n_u+n_y)p &\leq \operatorname{rank}(X_{p+1,1,N-p}) \leq \min \operatorname{rank}(\tilde{\mathcal{K}}^{(p)}, Z_{1,p,N-p}) \\
n + \underbrace{\operatorname{rank}(Z_{1,p,N-p}) - (n_u+n_y)p}_{\leq 0} &\leq \operatorname{rank}(X_{p+1,1,N-p}) \leq n,
\end{aligned} \tag{5-37}$$

so the rank of the state sequence can be less than n, thus leading to an inaccurate approximation. Moreover, the larger the number $p(n_u + n_y)$ is, the lower the rank of matrix can be, as it can be deduced from (5-37).

The celebrated advantages of the kernel based methods are not without a cost. More specifically, it is a common characteristic of both the PEI methods using ARX or FIR model structure and the subspace methods that they can treat MIMO systems in a direct manner, while the involved optimization routines are convex. Unfortunately, the kernel methods lead to the loss of these two properties in both identification approaches. As we already saw, the kernel methods require a non-convex optimization step for the estimation of the hyperparameters. Moreover, this estimation is usually repeated for each output, since the characteristics between the various input-output relations, such as the exponential decay rate, are not shared among the outputs, that is to say, each output is described by a different model and so the kernel specifications are different.

All in all, the kernel based approaches offer a very attractive framework for the identification of LTI systems. Despite their drawbacks, namely the increase in the complexity of the algorithm, the introduction of prior knowledge in the identification process is a very interesting property

which is expected to increase the accuracy of the estimated models. Since the subspace methods (through the use of the PBSID$_{opt}$ algorithm) are in the center of our interest, we will resort to various simulation examples in order to verify the validity of the claims in this chapter using the Algorithm 5.1.

Chapter 6

Simulations for the LTI case

In this chapter we will present some informative simulation examples to investigate the various aspects of the kernel-based SID method. More specifically, it is in our aims to investigate the following characteristics:

- The effect of the past window value
- The effect of the SNR value
- The effect of data length
- The relation between the kernels and the unknown system

In order to do so, we will use both SISO and MIMO systems as examples. The investigated kernels will be the 1st order stable spline, the 2nd order stable spline, the HF stable spline and the diagonal kernel, while for all of these we used the reduced number of hyperparameters, as it was discussed in Section 5-4. We also intend to compare these results with a kernel based method that uses an identity matrix as kernels but makes use of the estimated noise variance, σ^2. This practically means that the estimation is ridge regression [26], since the regularization term is $\sigma^2||\theta||_2^2$. In the related literature the latter estimator is not mentioned. This method contains no hyperparameters if σ is pre-estimated, so there is no requirement for any non-convex optimization. Therefore it is an attractive way to see if we can avoid this optimization without sacrificing the accuracy of the algorithm.

Moreover, we will compare these results with the optimal estimate following the analysis in Section 5-2. Additionally, the direct use of the real coefficients (based on the selected past window) will also be used to see how close the optimal regularization is to the real coefficients. Finally, a regularization method that makes use of the Generalized Cross-Validation (GCV) criterion will also be used [51]. The latter one is extensively used in the LPV-PBSID$_{opt}$ algorithm, but it can of course be employed in the LTI case too.

In these examples we will assume that the order of the system is known. In general, a commonly used way to find the order of the system is by examining the amplitude of the singular

values of the extended observability-times-controllability matrix (see also Appendix A) and detecting a gap [11].

Up to the author's knowledge, the only publication focusing on the synergy of kernel methods with SID methods is [8]. Nonetheless, in this publication some major aspects of the proposed algorithm are not investigated or clarified. More specifically

- It is not clear if a parametric part is finally introduced, following the idea in [59]
- No information about the noise characteristics used in the simulations is given.
- The effect of different past window values, as well as different data length values (especially of large data sets) is not investigated.
- The stable spline of order 2 was only investigated.

For these reasons, it is important to try to shed light in all these aspects of the kernel based PBSID$_{opt}$ algorithm. In all the following simulation examples we performed 50 Monte Carlo simulations for each setting, that is to say, we identified each system 50 times and each time a fresh input and noise sequence were used for the identification procedure.

As far as the noise is concerned, we simulated the system for specific SNR values. First of all, let us express again here the SNR definition for simplicity.

$$\text{SNR} = \frac{\text{var}(y)}{\text{var}(e)}, \tag{6-1}$$

where var(y) denotes the variance of the output, corrupted by noise and var(e) is the noise variance. Usually it is expressed in dB. In this case $\text{SNR}_{dB} = 10 \log_{10} \frac{\text{var}(y)}{\text{var}(e)}$. Since $y = y_{\text{noiseless}} + e$, it is not feasible to produce the desired SNR value with the use of (6-1) in our algorithms. For this reason, we will instead use a modification of SNR value, given by

$$\text{SNR}_{\text{mod}} = \frac{\text{var}(y_{\text{noiseless}})}{\text{var}(e)}. \tag{6-2}$$

Of course, after the execution of the simulation we can compute (6-1) (based on the samples) and so we can get an estimate of its value based on the 50 Monte Carlo simulations that we perform.

In each experiment we estimated a model based on various regularization schemes. More specifically, we examined in total the following cases:

1. Diagonal kernel + PBSID$_{opt}$ (Diag)
2. Stable spline kernel of order 1 + PBSID$_{opt}$ (SS-1)
3. Stable spline kernel of order 2 + PBSID$_{opt}$ (SS-2)
4. HF stable spline kernel + PBSID$_{opt}$ (SS-HF)
5. Ridge regularization with noise estimate + PBSID$_{opt}$ (NP ridge)

6. Classical PBSID$_{opt}$ (no regularization) (PBSID$_{opt}$)
7. GCV ridge regularization + PBSID$_{opt}$ (GCV-PBSID$_{opt}$)
8. Optimal regularization + PBSID$_{opt}$ (Opt)
9. True VARX coefficients + PBSID$_{opt}$ (MSE opt)

Let us summarize the main characteristics of the investigated algorithms 1-9. The first four methods are based on the kernels explained in Section 5-3. The ridge regularization with noise estimate adds a regularization term of the form $\sigma^2||\theta||_2^2$, where the noise variance was pre-estimated. The GCV ridge regularization is a standard method to perform regularization. It is implemented in PBSID Toolbox [73] and is based on [51]. The optimal regularization is based on the theory developed in Section 5-2. Finally, the "True VARX coefficients + PBSID$_{opt}$" method is skipping the VARX step by directly using the actual values of the Markov parameters. Here it is important to make an important remark, that will be further investigated in the examples to follow. The fact that the latter two methods use the real Markov parameters somewhere in the algorithm does not mean at all that they will always exhibit the best results. Even though they are not expected to suffer from any overfitting problems, the approximation error is directly related to the value of past window. Consequently, if the past window is small, the perfect knowledge of the Markov parameters is not enough to lead to a highly accurate model because there can be too much "information" about the system in the term $C\tilde{A}^p X_{p+1,1,N-p}$, which is nonetheless neglected due to the approximation. Finally, we will either call these methods based on the given numbers or based on the abbreviations, given in brackets above.

6-1 Example 1: A 2nd order open-loop LTI SISO system in ARX form

The system under investigation is taken from [8]. In order to keep consistency with the notation, we will use the schematic in Figure 6-1 to characterize the involved transfer functions.

In this example we examine an open-loop system, described by the following transfer functions:

$$\begin{aligned} F(z) &= \frac{0.5778z - 0.242}{z^2 - 0.7z - 0.18} \\ G(z) &= \frac{z^2 + 0.4z - 0.21}{z^2 - 0.7z - 0.18} \end{aligned} \tag{6-3}$$

This system is rather not oscillatory, since its poles are at -0.2 and 0.9 (for relation between pole location and impulse response behaviour see [74]). So, it is expected that smooth kernels will yield the best results.

The noise variance was pre-estimated, based on (5-19), by using a VARX model with a past window 4 times the order of the system. In general, it was observed that a past window of 4-6 times the order of the system yield a σ estimation which is close to the actual noise of the system.

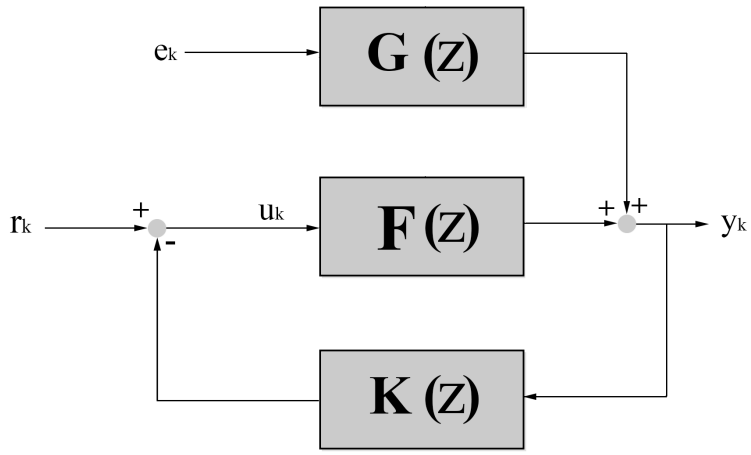


Figure 6-1: Schematic of an LTI system

We simulated the system using $N = 150$ data points and with an $\text{SNR}_{\text{mod}} = 10$, while the average of the SNR value was 5dB. The future window was kept constant in all simulations, $f = 10$. The input signal is Random Binary Sequence (RBS) with amplitude 1, exciting all the frequencies (0 until π rad/s), while the sampling frequency is $1s$. In order to enable the clear exhibition of the results, we present on the left plot of Figure 6-2 a comparison of the non-parametric kernel methods (1-4) and on the right plot of Figure 6-2 a comparison of the best non-parametric kernel method with the rest algorithms (5-9). The validation results are generated using a fresh input sequence with the same characteristics as in the estimation set but without the addition of noise.

Moreover, in order to investigate the susceptibility of the methods to the data length, we repeated the simulations with the same setting except for the data length, which is now chosen to be $N = 500$. The results are given in Figure 6-3.

As we can see, all the kernels methods are capable of performing much better than the other methods for all past window values. Among them, it is evident that the "Diag" algorithm is performing very well but in most cases it is overweighted by the "SS-2" algorithm. The fact that all the stable spline kernels show these deviations is mainly attributed to their sensitivity on the chosen initialisation values for the hyperparameters. Taking into account that a multistart approach in the optimization routine was avoided (mainly for computational reasons), we have a clear explanation of the results. Moreover, we observe that indeed the "HF-SS" is not capable of capturing well the dynamics of this non-oscillatory system, but still it delivers better results than the methods 5-9. As far as the "SS-1" algorithm is concerned, by closely investigating the derived results it was observed that the algorithm fails to deliver reasonable values for the hyperparameters only at a few Monte Carlo simulations. However, this leads to a completely wrong estimation and so a zero VAF is accounted at these simulations. Nonetheless, if we neglect these cases (which were about the 10% of the total Monte Carlo simulations) then the VAF results are close to these of the other kernel methods (1-4). The same conclusions can be verified by both simulation settings, as they are shown in Figure 6-2 and Figure 6-3.

We also see that the "MSE opt" method shows almost the same accuracy as the "Opt"

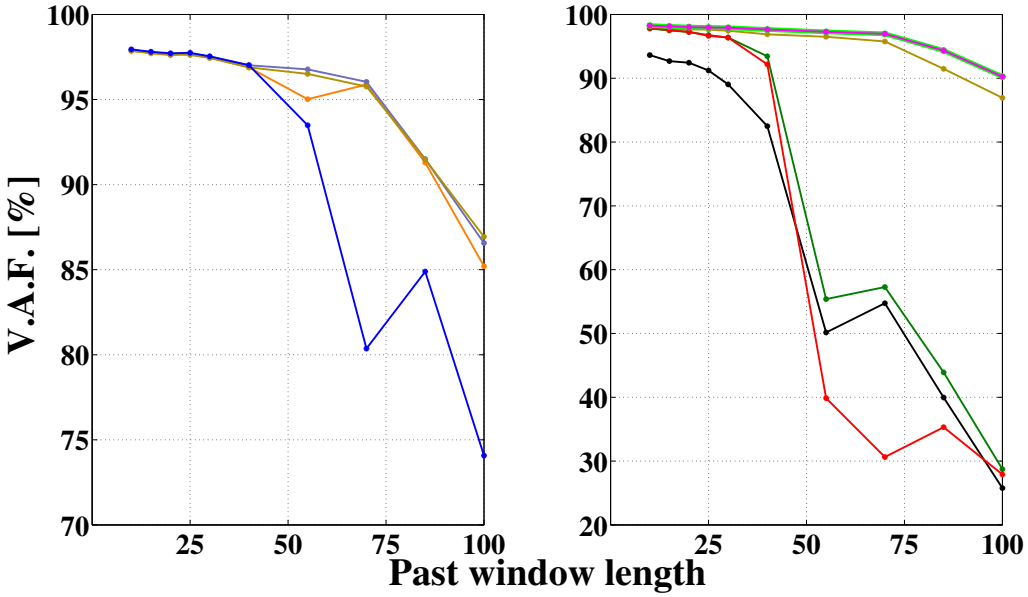


Figure 6-2: Example 1: validation results for different p values and $N = 150$. Left plot: The orange curve corresponds to "HF-SS", the steel blue to "Diag", the blue one to "SS-1" and the gold one to "SS-2". Right plot: The gold curve corresponds to SS-2, the dark green to "NP-ridge", the red one to "PBSID$_{opt}$", the black to the "GCV-PBSID$_{opt}$", the magenta to the "MSE opt" and the light green to "Opt".

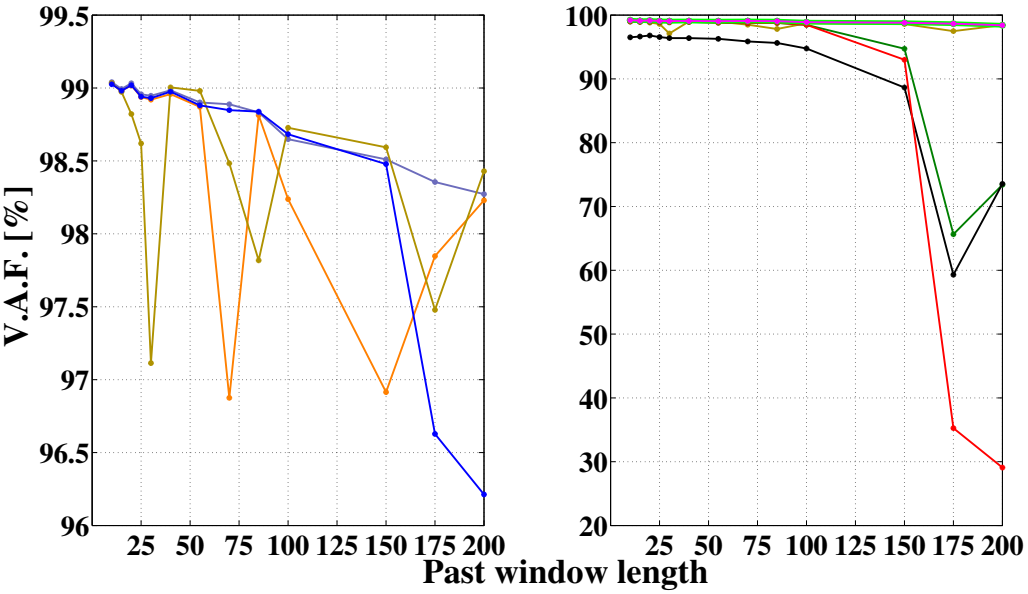


Figure 6-3: Example 1: validation results for different p values and $N = 500$. Left plot: The orange curve corresponds to "HF-SS", the steel blue to "Diag", the blue one to "SS-1" and the gold one to "SS-2". Right plot: The gold curve corresponds to SS-2, the dark green to "NP-ridge", the red one to "PBSID$_{opt}$", the black to the "GCV-PBSID$_{opt}$", the magenta to the "MSE opt" and the light green to "Opt".

algorithm (see also the discussion at the end of Section 5-2). At this point it is important to clarify their drop in terms of VAF results for the case where $N = 150$, which is not related with the accuracy of the Markov parameters. This result is related to the inaccurate estimation of the state sequence, following the lines in Section 5-5. The same justification is enhanced by noticing that the "Opt" method completely skips the VARX steps (since it uses directly the real Markov parameters) and yet it experiences a deterioration in the accuracy of the estimated model, as it can be seen in Figure 6-2.

This example is a rather simple one. In order to better investigate the validity of the new kernel based methods, we will proceed to the identification of more complex systems.

6-2 Example 2: A 2nd order closed-loop LTI SISO system in ARX form

In this example we will investigate the behaviour of the kernel based algorithm in a closed-loop setting. The related transfer functions are given by

$$\begin{aligned} F(z) &= \frac{0.4802z - 1.351}{z^2 - 0.6z + 0.73} \\ G(z) &= \frac{z^2 + 0.6z - 0.27}{z^2 - 0.6z + 0.73} \\ K(z) &= 1 \end{aligned} \tag{6-4}$$

Following the same approach as in the previous example, we performed 50 Monte Carlo simulations for each past window. The chosen future window was kept constant, $f = 10$. In this example the derived VAF results for the free run outputs (SNR $\to \infty$) do not lead to safe conclusions because the VAF values of most of the examined methods are really close. For this reason, we evaluated the accuracy of the algorithms by making use of the one step ahead predictor. In these validation simulations we used an input of the same characteristics as the one used in the estimation dataset, while the noise sequence e_k is a zero mean normally distributed white noise with variance 0.1. In order to view a different aspect of the kernel based methods, we present the one step ahead predictor VAF results for two different cases: the first one is based on a system with high noise (SNR $\approx$ 7dB), while the second one is characterized by a better SNR value equal to 10dB. In both simulations we used 500 data points, while the results for the former case are given in Figure 6-4 and for the latter case in Figure 6-5.

The best kernel based method for this case was the "HF-SS", a result which was rather expected due to the fact that the system poles are located at $0.3 \pm 0.8i$ and so the system exhibits an oscillatory behaviour. These simulations also reveal a different aspect of the examined methods. More specifically, it is observed that the "NP ridge" method is able to identify accurately the underlying model when the past window is relatively small. On the other hand, when the past window is getting large, the accuracy of the "NP ridge" method is falling, as it can be seen in Figure 6-6, which is a zoom-in of Figure 6-4.

In order to understand this result we have to make some remarks. Concerning the drop in performance as the past window value increases, it is obvious that it is related to the fact that

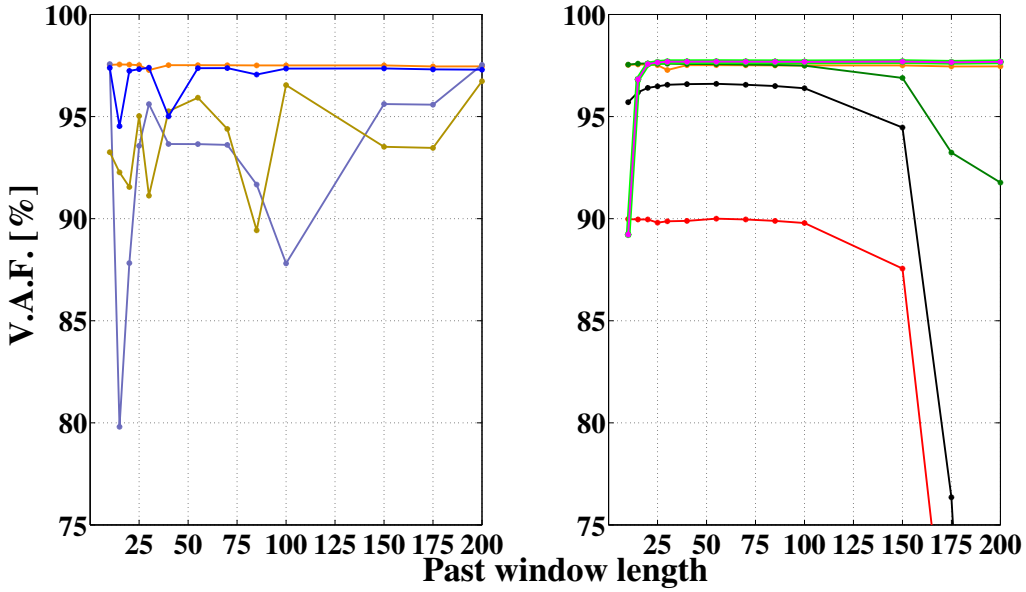


Figure 6-4: Example 2: validation results for different p values and $N = 150$. Left plot: The orange curve corresponds to "HF-SS", the steel blue to "Diag", the blue one to "SS-1" and the gold one to "SS-2". Right plot: The gold curve corresponds to "HF-SS", the dark green to "NP-ridge", the red one to "PBSID$_{opt}$", the black to the "GCV-PBSID$_{opt}$", the magenta to the "MSE opt" and the light green to "Opt".

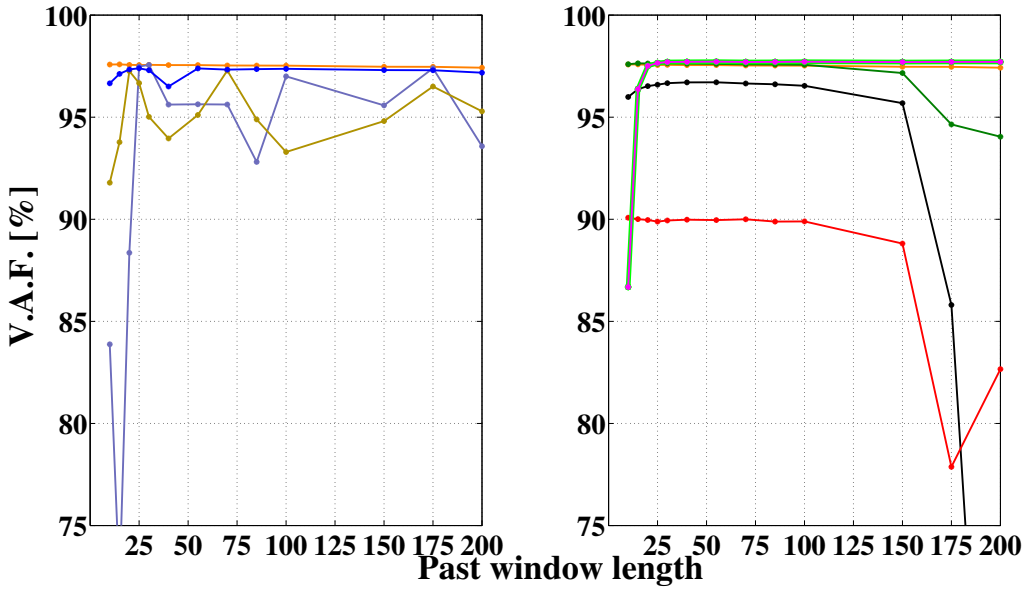


Figure 6-5: Example 2: validation results for different p values and $N = 500$. Left plot: The orange curve corresponds to "HF-SS", the steel blue to "Diag", the blue one to "SS-1" and the gold one to "SS-2". Right plot: The gold curve corresponds to SS-2, the dark green to "NP-ridge", the red one to "PBSID$_{opt}$", the black to the "GCV-PBSID$_{opt}$", the magenta to the "MSE opt" and the light green to "Opt".

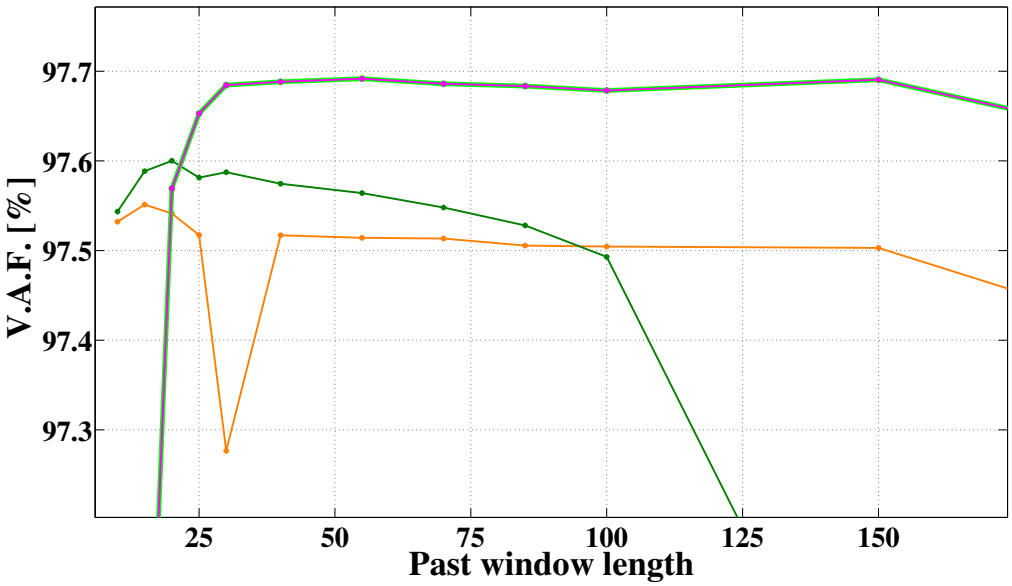


Figure 6-6: Example 2: zoom-in of Figure 6-4.The dark green curve corresponds to "NP-ridge", the orange one to "HF-SS", the magenta to the "MSE opt" and the light green to "Opt".

the regularization term of "NP ridge" assigns the same significance in all the coefficients due to its structure ($\sigma^2||\theta||_2^2$). Therefore, it is not able to "realize" that for large past windows the difference between the values of the first impulse response coefficients (CB, CK) and the last ones ($C\tilde{A}^{p-1}B, C\tilde{A}^{p-1}K$) is getting too large and so the imposition of an identity variance is not valid. However, the ability of this method to estimate accurately the underlying system when p is small is related to the fact that it is not using any hyperparameters (except for σ, which is though not treated as such) and so it is avoiding any pitfalls due to the non-convex optimization routine of methods 1-4.

Finally, another interesting remark is related to the VAF results for "MSE opt" and "Opt". For small past window values, the VAF results for these two methods are lower than the other methods. This result though is not unexpected and it is related to the neglect of the term $C\tilde{A}^p X_{p+1,1,N-p}$, as it was already discussed in the beginning of this chapter. Therefore, for systems that show an oscillatory behaviour (as the one in this example) it is necessary to take more impulse response coefficients into account in order to achieve the optimal approximation of the impulse response. Of course, the value of C and $X_{p+1,1,N-p}$ also play a role to this.

6-3 Example 3: A 4th order open-loop LTI SISO system in ARX form

In this example we estimated a more complex system, namely an open-loop 4th order LTI system described by the following transfer functions.

$$F(z) = \frac{1.067z^3 - 6.824z^2 - 1.39z - 0.8556}{z^4 - 1.1z^3 + 0.95z^2 - 0.523z - 0.153}$$
$$G(z) = \frac{z^4 + 0.8z^3 + 0.8z^2 + 0.256z - 0.1785}{z^4 - 1.1z^3 + 0.95z^2 - 0.523z - 0.153}. \tag{6-5}$$

The future window was set to $f = 10$, while we compared the accuracy of the algorithms for different past windows. Again, due to the highly oscillatory behaviour of the system, the "HF-SS" was observed to offer the most accurate results. In this example, though, we would like to investigate some other aspects of the kernel based SID methods.

Results for a sufficiently excited system

First of all, elaborating on the observations in the previous example concerning the "NP ridge" method, we turn our attention to the comparison of the "NP ridge" with the 'HF-SS" method. On the left plot of Figure 6-7 we compare the accuracy of the one-step ahead predictor of 'HF-SS" with the ones of the methods 5-9, while on the right plot we zoom in to examine better the results of the "NP ridge" and 'HF-SS" methods. The input used in the validation simulations was an RBS signal, which excited uniformly the frequencies 0 until π and the noise sequence e_k is a zero mean normally distributed white noise with variance 0.1.

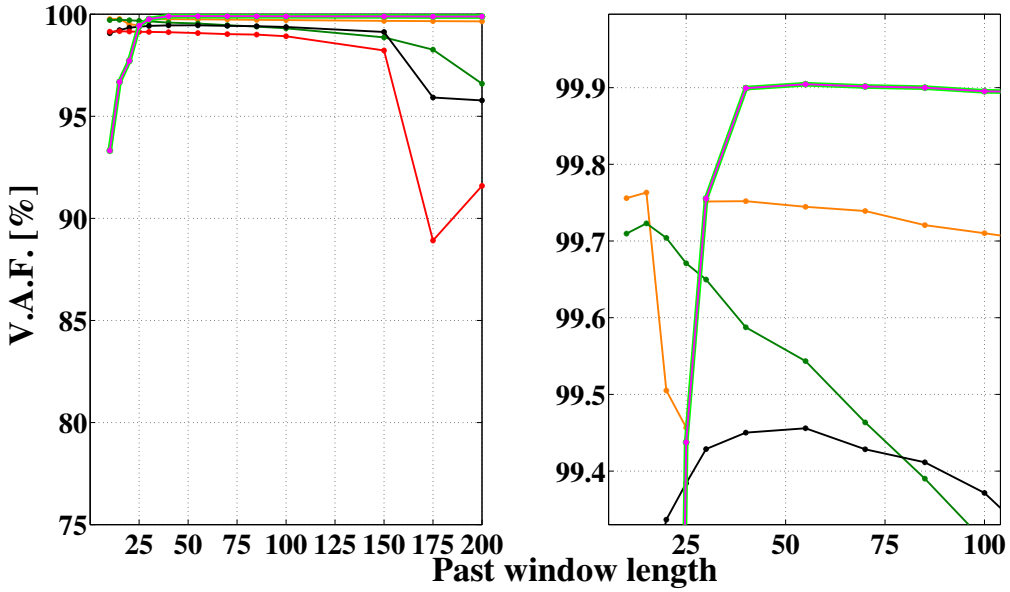


Figure 6-7: Example 3: validation results for $N = 500$. Left plot: the orange curve corresponds to "HF-SS", the dark greento "NP-ridge", the red one to "PBSID$_{opt}$", the black to the "GCV-PBSID$_{opt}$", the magenta to the "MSE opt" and the light green to "Opt". Right plot: zoom-in of the left plot.

In the right plot we observe a slightly different behaviour for "NP-ridge" compared to the one in Example 3. The obtained VAF results for this method are still high, but now the "HF-SS" is having in general a higher VAF value even for small past window values, except for $p = 25$,

which is rather due to a local minimum in the non-convex optimization routine. In order to understand this difference it is required to investigate the properties of the underlying model, which is the only setting that is different from the previous example (the future window, the data length and the past window values were the same). The system in this example is showing a more oscillatory response compared to the one in Example 2. The poles of the system in this example are at $0.9, -0.2$ and $0.2 \pm 0.9i$. Therefore, it is more crucial to capture these dynamics through a well tuned kernel and the "HF-SS" kernel is the most appropriate for it. For this reason the "NP-ridge" method, by completely disregarding the off-diagonal terms, is not able to perform any more well due to the assigned covariance, which is identity.

Poor excitation of the system

The same results can also be verified in the case of poor excitation. By keeping the various parameters the same, we only changed the input signal to an RBS signal, which excites uniformly only the frequencies from 0 until $\pi/2$rad/s. Consequently, the high frequency dynamics of the system are not well excited, which can be crucial for this system that contains high frequency dynamics. In order to make this point clear, we present in Figure 6-8 the spectrum of the input signal (expressed in dB) versus the frequency, as well as the magnitude of the bode diagram for the system (6-5).

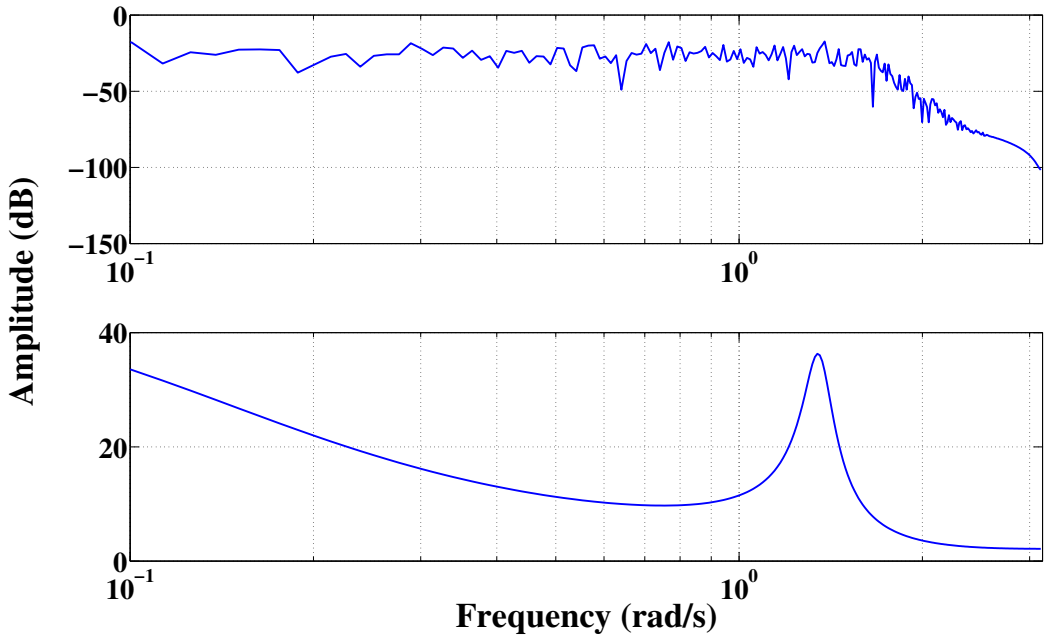


Figure 6-8: Upper figure: spectrum of input signal. Lower figure: magnitude of the bode diagram for the system in (6-5)

The resonant frequency is rather well excited, but after this point the amplitude of the spectrum for the input signal drops rapidly and so the higher frequencies (which contain three zeros and one pole of $F3$ in $[\pi/2, \pi]$) are not well excited.

This study case is presenting in a clear way the merits of the kernel based methods and especially of "HF-SS". More specifically, the incorporated information about the system in

the kernel based methods is the key to preserve the accuracy of the estimation. On the other hand, the "NP-ridge" lacks this information and so it is not able to estimate the unknown system with an accuracy close to the one of the "HF-SS" method. These results are visualized in Figure 6-9, where the one-step ahead predictor estimates are being shown. Now, the VAF results for "HF-SS" are around 96%, while the VAF results for "NP-ridge" are around 93% for past window values until $p = 85$. It is also interesting that the classical PBSID$_{opt}$ algorithm completely fails to estimate correctly the underlying system, making more clear that the regularization is necessary in the general setting.

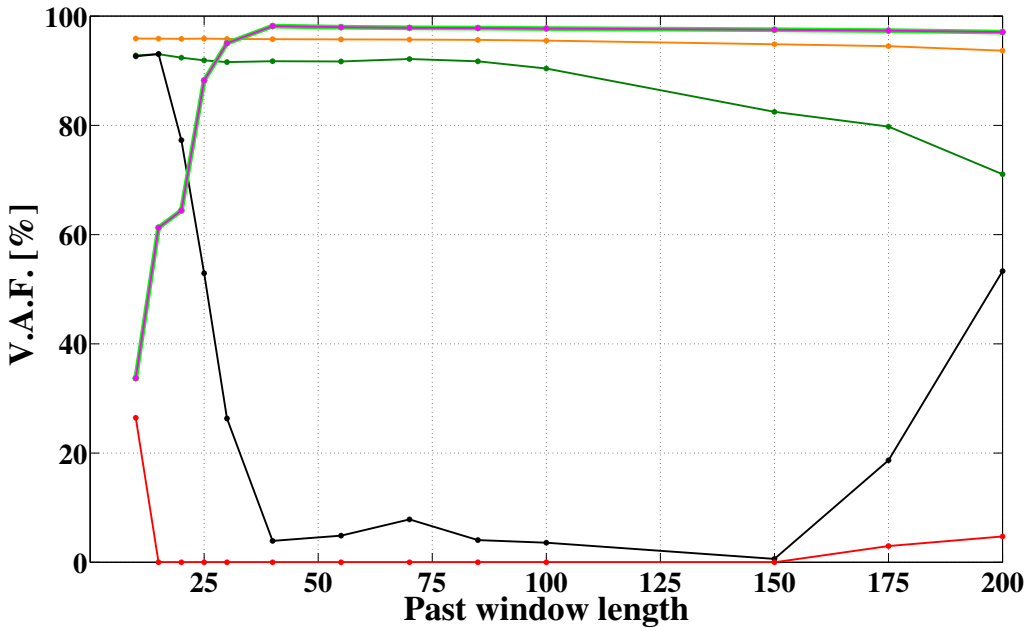


Figure 6-9: Example 3: validation results for $N = 500$ under poor excitation. The orange curve corresponds to "HF-SS", the dark green to "NP-ridge", the red one to "PBSID$_{opt}$", the black to the "GCV-PBSID$_{opt}$", the magenta to the "MSE opt" and the light green to "Opt".

Comparison of the singular values

Before we conclude on this example, it is useful to show another property of the kernel based methods, related to the fact that they estimate accurately the Markov parameters even for very large past window values. More specifically, this accurate estimation is reflected in the estimation of the order based on the detection of the gap in the SVD of (A-8). To see this, let us compare the singular values of "HF-SS" and the classical PBSID$_{opt}$ algorithm, based on the data used in Section 6-3. For small past window values (until $p = 50$), at is was shown in Figure 6-7, the VAF results of the "HF-SS" are slightly better than the ones of the classical PBSID$_{opt}$ (approximately 0.5-1%) and so the gap in the singular values is almost the same for both methods. For this specific example, the difference becomes obvious when $p = 200$, as it is presented in Figure 6-10.

Therefore, the kernel based methods can lead to a better selection of the order of the system, when this is not known (as it is assumed in these examples). In general, it was observed

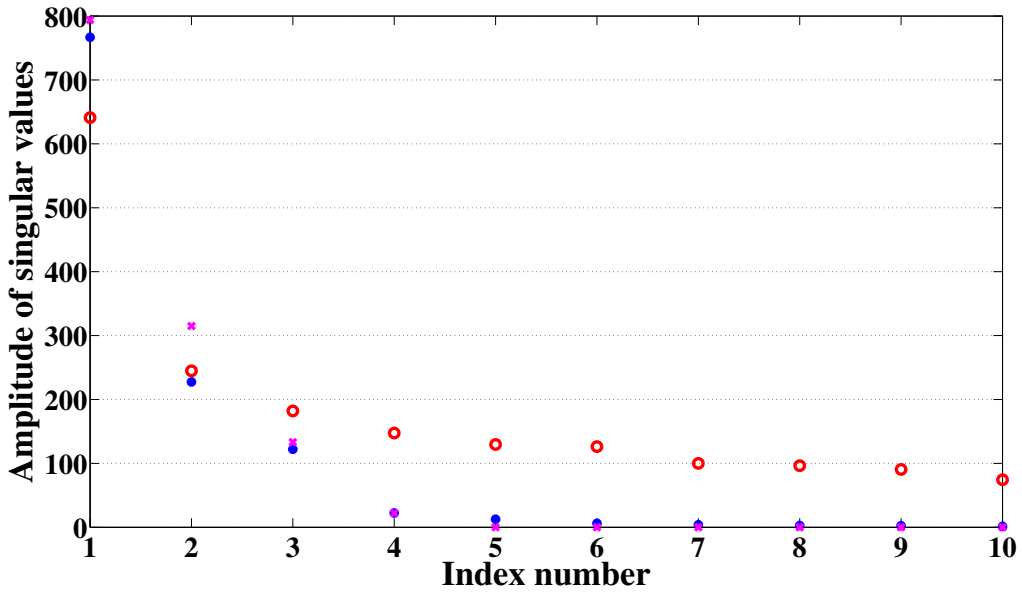


Figure 6-10: Example 3: singular values of "HF-SS" (blue stars), classical PBSID$_{opt}$ (red circles) and "MSE opt" (magenta crosses) for a past window $p = 200$.

that the same results are derived in cases where the input is not sufficiently exciting but the kernel methods can still lead to an accurate estimation of the Markov parameters, due to the incorporated information about the system properties. This observation is useful since this gap is the main information used both in cases where the engineer himself chooses the order of the system or when automated procedures (such as the singular value criterion [75]) are used.

6-4 Example 4: A 4th order open-loop LTI MIMO system in state-space form

As a final example to evaluate the validity of the kernel based methods we would like to compare the accuracy of the kernel methods with the accuracy of the classical PBSID$_{opt}$ method in the system described in [23]. This is a 2-inputs, 2-outputs system, described by

$$\left[\ A\ \right] = \begin{bmatrix} 0.67 & 0.67 & 0 & 0 \\ -0.67 & 0.67 & 0 & 0 \\ 0 & 0 & -0.67 & -0.67 \\ 0 & 0 & 0.67 & -0.67 \end{bmatrix}, \quad \left[\ B\ \right] = \begin{bmatrix} 0.6598 & -0.5256 \\ 1.9698 & 0.4845; \\ 4.3171 & -0.4879 \\ -2.6436 & -0.3416 \end{bmatrix},$$

$$\left[\ C\ \right] = \begin{bmatrix} -0.3749 & 0.0751 & -0.5225 & 0.5830 \\ -0.8977 & 0.7543 & 0.1159 & 0.0982 \end{bmatrix}, \quad \left[\ D\ \right] = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

$$\left[\ K\ \right] = \begin{bmatrix} -0.6968 & -0.1474 \\ 0.1722 & 0.5646 \\ 0.6484 & -0.4660 \\ -0.9400 & 0.1032 \end{bmatrix}.$$

For these simulations we used a past window of $f = 5$, which is just above the order of the system. The average SNR value for the two outputs were 4.5dB and 2.67dB. In this example we would like to highlight the general superiority of the kernel based methods and specifically of the "HF-SS" algorithm but also the possible inaccurate results of this method in cases where the non-convex optimization is ending up in a local minimum. For the simulations we used again 500 data points and we used an input that excites uniformly all the frequencies up to Nyquist frequency. The VAF results of the one step ahead predictors for different past window values are given in Figure 6-11.

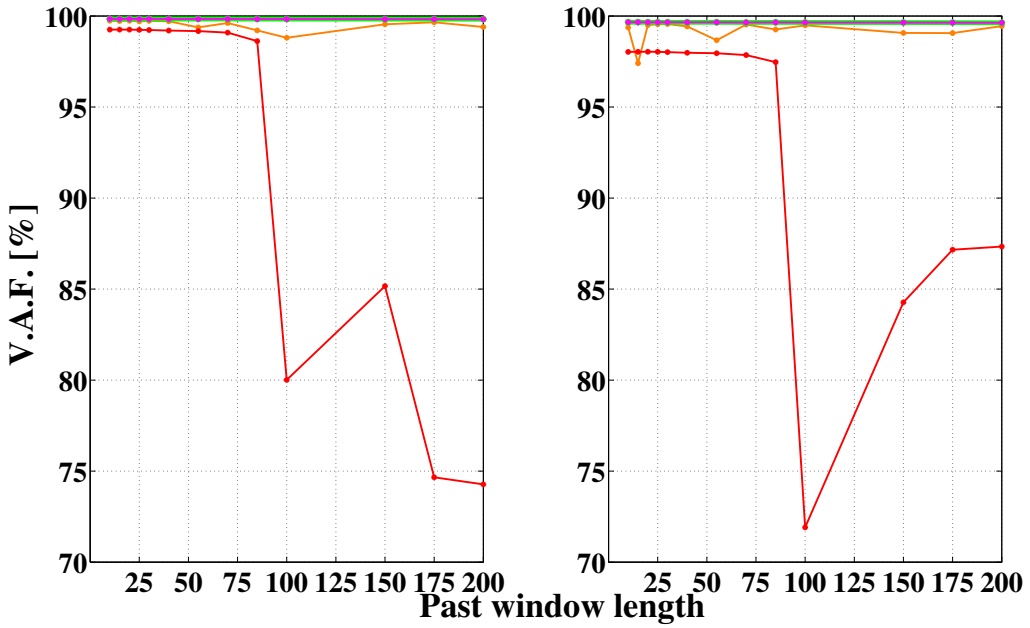


Figure 6-11: Example 4: validation results of the one step ahead predictors for $N = 500$. The left plot corresponds to the first output and the right one to the second output. The orange curve corresponds to "HF-SS", the red one to "PBSID$_{opt}$", the magenta to the "MSE opt" and the light green to "Opt"

As it is expected, the validation results favour the "HF-SS" method. In this example, though, we would also like to highlight the variance properties of the "HF-SS" method, based on the

50 Monte Carlo simulations that we performed for each past window. By plotting the 25th and the 75th percentiles (given by the grey frames) based on the Monte Carlo results, we present in Figure 6-12 only the results for "HF-SS".

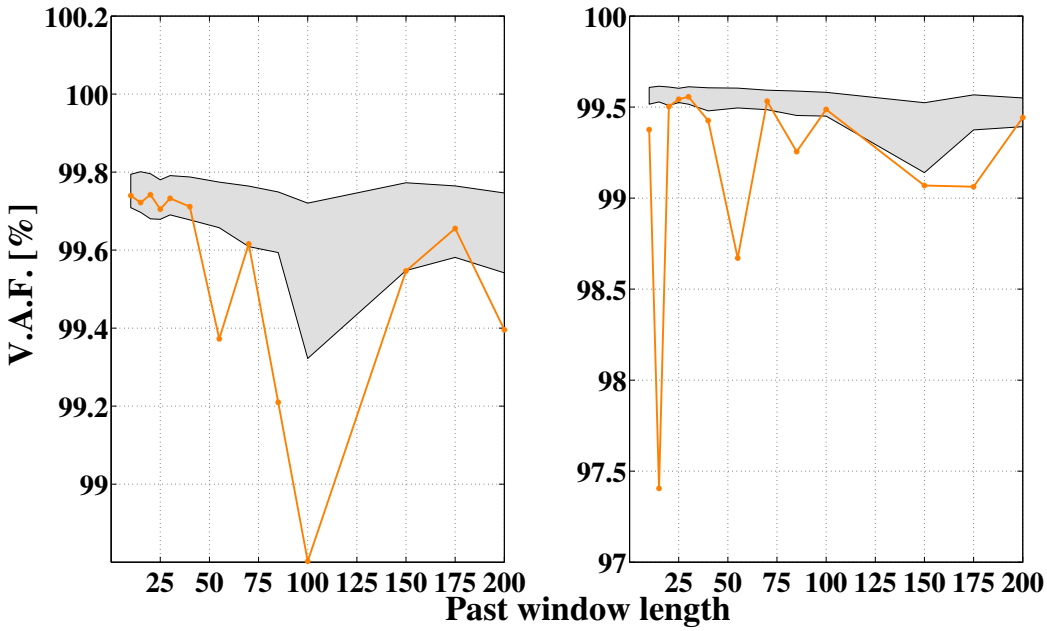


Figure 6-12: Example 4: validation results of the one step ahead predictor of"HF-SS", together with the 25th and 75th percentiles. The left figure corresponds to the first output and the right one to the second output.

The Figure 6-12 shows more clearly another property of the kernel based methods. By observing that the 25th and 75th percentiles are very high while the sample average VAF value for each past window is sometimes outside this region, it becomes evident that the drop in performance is related to the local-minima in the non-convex optimization. Even though an inaccurate estimation of the underlying model is not the rule, it is still enough to bring the sample average down. Actually, this is also verified by investigating one by one all the VAF results. In total there are 50 Monte Carlo simulations for each p times 13 different past window values, so in total 650 identification procedures. Among these it was observed that there is at most one identification procedure for each past window value that delivers lower VAF results. Consequently, this enhances the argument that in experimental setups it is advisable to repeat the non-convex optimization of the marginal likelihood for different initialization points. This is however far from trivial. More specifically, even a multistart procedure does not necessarily lead to a better accuracy. This is related to the fact that the minus log marginal likelihood optimization problem is not trying to optimize with respect to σ^2 (even though this is possible) and more importantly, there is no optimization taking place with respect to the kernel structure itself. Therefore, a smaller value of the objective function does not necessarily lead to a more accurate estimation. This was indeed verified in the case of $p = 15$. More specifically, we performed a multistart approach for β and all the λ. For each output, the value of β was initialized based on a logarithmic grid with $\beta = [0.0003, \quad 0.0013, \quad 0.0056, \quad 0.0237, \quad 0.1]$ and λ based on a linear grid with

$\lambda = [0.1, \quad 5.075, \quad , 10.05, \quad 15.025, \quad 20]$. For all possible combinations we computed the hyperparameters that minimize the minus log marginal likelihood function and we kept the ones that give the lowest value for the objective function. Moreover, we performed the same procedure by skipping the pre-estimation step for σ^2 since we set it directly to its real value. On the other hand, we identified the system by using only one initialization point for the hyperparameters, namely $\beta = 0.005$ and $\lambda = 1$, which correspond to the initialization values that we used in **all** the simulations of this chapter. Of course, for all these identification procedures we used the same data set to enable a fair comparison between the 1-step ahead predictor VAF values, while in these validation results we simulated the system without noise. The results are concentrated in Table 6-1.

Table 6-1: 1-step ahead predictor (without noise) VAF results for Example 4 with $p = 15$. Mult. stands for a multistart approach.

Method	Mult.. HF-SS	HF-SS	Mult.. HF-SS, known σ^2	HF-SS, known σ^2
1st output VAF	98.10 %	99.84 %	99.47 %	99.77 %
2st output VAF	97.12 %	97.69 %	97.08 %	99.86 %

As we can see, the multistart procedures actually deteriorate the VAF results. Moreover, when σ^2 is known, the VAF results are improved but still it is not enough to lead to an accurate estimation of the underlying LTI system in the multistart case, thus enhancing the argument that the kernel structure itself plays an important role in the accuracy of the kernel based PBSID$_{opt}$ method. Even though there is no reason to assume that this will be always the case for other LTI systems, it is nonetheless obvious that by finding the hyperparameters that minimize the objective function there is no guarantee that the corresponding VAF results will be improved.

Chapter 7

Conclusions and recommendations for the kernel based SID of LTI systems

7-1 Conclusions on the kernel based methods for Subspace Identification (SID)

In this chapter we introduced the kernel based identification methods for LTI systems. By introducing simple prior knowledge in the identification algorithm and sophisticatedly tuning the bias variance trade-off, these methods have shown in multiple experiments to deliver superior results with respect to the classical identification methods and therefore they attracted the interest of many scientists in the system identification field. In this thesis we took as a staring point the research on the kernel based Prediction Error Identification (PEI) methods and we showed how this framework can also be applied in SID methods and specifically in the PBSID$_{opt}$ algorithm, following the lines in [8]. Moreover, since an in depth investigation of this synergy between the SID methods and the kernel methods was missing from the related literature, both in theoretical and in practical (based on simulations) terms, we investigated in depth its various aspects.

What did we win and what did we sacrifice in the kernel based PBSID$_{opt}$ algorithm? In a nutshell, in this new approach we had to sacrifice the convexity and the direct treatment of MIMO systems (both are the main characteristics of the classical PBSID$_{opt}$ algorithm), but we won in terms of a much higher accuracy in the identified model, which also had the side-effect of revealing a more clear gap in the singular values, which are used to determine the order of the system. In this sense, it becomes evident that the kernel based methods are highly attractive due to their superior identification properties.

Moreover, it is well known that the value of the past window is a crucial parameter in the classical SID methods. However, we have reasoned that in the kernel based SID methods its value does not reflect any bias-variance trade-off as long as it is large enough to capture the dynamics of the impulse response. On the contrary, the bias-variance trade-off was achieved through the introduction of the Gaussian prior in the impulse response coefficients and the

subsequent estimation of the maximum a posteriori estimates. Of course, when the past window is extremely high, a small drop in the performance of the kernel based algorithms was observed, but one has to take into account that in this case the number of the parameters that have to be estimated is high and so numerical errors are to be expected. For example, for a past window value $p = 200$, which was also used in Example 4, the number of the parameters to be estimated for each output (by treating each output separately) from a Least Squares (LS) problem was $200 \times 2 \times 2 = 800$, while the corresponding number of hyperparameters (using the approach in Section 5-4) that have to be estimated by a non-convex optimization was 5 (so 10 for both outputs).

As far as the examined kernel structures is concerned, it was observed that the HF stable spline kernel is better suited in systems that show highly oscillatory behaviour, compared to the other stable spline kernels. The diagonal kernel seems to be a decent selection, especially in cases where the optimal (off-diagonal) covariance terms are close to zero. This is related to the fact that the diagonal kernel is not even trying to set any value in the off-diagonal elements of the kernels K^u and K^y. On the other hand, the stable spline kernels are always trying to assign values in these off-diagonal elements due to their inherent structure. However, due to the non-convex nature of the optimization routine for the calculation of the hyperparameters it is possible that these off-diagonal terms will drive the hyperparameters to a rather "bad" local minimum, due to the increased complexity of the algorithm. In order to allow this flexibility in the stable spline kernels it is necessary to include another hyperparameter that accounts for the off-diagonal elements in an independent way. For example, this is the case in the recently proposed diagonal/correlated kernel [54]. Of course, the addition of a hyperparameter increases the complexity of the algorithm and so it may also lead to the same result (ending up in a local minimum) due to a different reason (further increase in the computational complexity).

As a general remark about the kernel selection, we clearly observed that the stable spline family of kernels can deliver more accurate results than the diagonal kernel, but they are more susceptible to local-minima issues, which of course could be bypassed with the use of multistart methods in the optimization routine. However, this is not a trivial procedure. As we explained in Example 4, using different initialization points for the hyperparameters and choosing those that minimize the minus log marginal likelihood is not itself enough to guarantee improved results. In fact, the inherent assumptions contained in this non-convex optimization, such as the value of the noise variance σ^2, as well as the structure of the kernel are not optimized in it. Therefore, finding the global minimum without taking care of the latter two parameters does not necessarily mean that the identified model will exhibit the highest accuracy compared to the other possible selections of the hyperparameters. Consequently, if a multistart approach is to be followed, the chosen hyperparameters should be the ones that increase the accuracy of the estimated model and not the ones that correspond to the lowest value of the minus log marginal likelihood function.

In total, the kernel based methods are a highly attractive way to treat the SID problem. By practically inactivating the role of the past window value in the accuracy of the SID algorithms and simultaneously delivering more accurate models with respect to the so far state-of-the-art identification methods, it is rather fair to say that they lead to a change of paradigm in the system identification process. The whole discussion about these methods has just began and there are still many aspects to be investigated, both theoretical and practical. For this reason it is important to show some of these topics that could be treated in future works.

7-2 Recommendations and possible future work

As it is the case with many new ideas, by offering answers to some questions they simultaneously lead to many new questions yet to be answered. For the kernel based identification the aspects that have to be investigated are numerous. It is important though to realize that the investigation of kernel based SID methods is in close terms with the discussion on the kernel based ARX model identification and so many of the scientific questions that have to be addressed can be applied in both system identification methods (PEI and SID).

First of all, for the improvement of the results, the multiple kernel based identification, presented at the end of 2014 [64] seems to be a promising one. As we already discussed in Section 5-3, in this new approach multiple kernels are used to capture more complex impulse responses. However, this approach still has many open questions, such as what is the optimal number of kernels, while the possibility of combining different kernel structures is not yet reported in the related literature.

Of course, the available kernel structures still need investigation and the possibility of creating new ones has also to be examined. This research effort is also reflected on the fact that new kernels have also been proposed very recently, merely because the discussion about what is the **practically optimal kernel structure** is not fully answered yet.

Another open question is how the computational complexity of the algorithm, related to the non-convex optimization of the marginal likelihood function, can be reduced, possibly by being viewed as a trade-off between accuracy of the estimation and the computational complexity. One partial solution to this problem is investigated in [72], where the hyperparameters for all the outputs are estimated by their joint marginal likelihood and so the same hyperparameter values can be used for all the outputs. However, in the same publication it is mentioned that a parametric part similar to the one in [59] has to be introduced to increase the flexibility of the algorithm. In general, the computational aspects of the proposed algorithm are of high importance, mainly due to the introduction of the non-convex optimization routine. Therefore, it is recommendable to explicitly investigate this aspect and propose more efficient algorithms, a task which is yet in a preliminary stage [76, 77].

On the other hand, if the computational aspects and the time limitations are not crucial in a specific application, a different research approach can be investigated. Based on the experience that we gained during this thesis it required a lot of effort, partially by employing a trial and error method, to tune the optimization parameters (initialization values, number of iterations etc.). Especially for the family of stable spline kernels it was rather difficult to find a good initialization for β. Therefore it is expected that an improvement in the results can be achieved through a multi-start approach. To this end, the remarks on the conclusions above should necessarily be taken into account.

Additionally, a very interesting topic to be investigated is related to the comparison of the kernel based SID and PEI methods. As we saw in this part, the kernel based methods aim at identifying the impulse response of the underlying system. This subsequently means that we have to estimate many coefficients, much more than the order of the system. In the PBSID$_{opt}$ algorithm the step of order reduction is already embedded in it. On the other hand, the kernel based PEI methods deliver a model of very high order. If this model is to be used for control purposes it is necessary to employ model reduction techniques in order to end up with a low order model, as it was indeed observed in [59]. However, it is not yet reported in the related

literature what is the cost of adding this extra step in terms of accuracy of the PEI methods. Therefore, a comparison between the kernel based SID methods and the kernel based PEI methods combined with a model reduction step is expected to reveal a new insight into the potentials of the two methods.

The previous remarks are focusing on the kernel based methods for the identification of an ARX model. However, in the SID method there are also another two LS problems that have to be solved, given in (A-9) and (A-10). The idea of an optimal kernel (even though it cannot be practically implemented), as it was shown in Section 5-2 is rather intriguing. So someone could ask if there is an optimal kernel and if a kernel based method can be used in these two problems. The questions that arise are multiple. The state sequence is itself an estimated quantity, so it is rather unclear if the estimation of C based on a kernel based method can indeed lead to better results. Moreover, the system matrices A, B, C are not characterized by a specific structure (as it holds true for the impulse response coefficients) and so a possible use of a kernel will require many hyperparameters to enable a flexible structure. However, for systems that have a low number of states and a few outputs (possibly a MISO system) it may be possible to follow a kernel based approach. At least for the estimation of C matrix, it may be possible to view its coefficients as random variables and set a prior, whose hyperparameters will be estimated via the marginal likelihood optimization.

It is also very important to investigate the validity of the claims in this thesis report on an experimental setup. In general, the kernel based methods have not yet been applied to experimental setups, with the only exception of the identification of a robotic arm using a kernel-based FIR model [58]. It is more than obvious that the usefulness of the kernel based methods and so of the kernel based PBSID$_{opt}$ algorithm has to be tested in experimental identification procedures. A systematic investigation of the effectiveness of these methods in real setups can further justify their selection and possibly lead to new ideas about how these methods could be improved.

Finally, another still unexplored path is the possible adaptation of recursive methods on the kernel based SID setting. It is well known that the SID algorithms can be adapted for recursive estimation of the unknown coefficients, a method which enables the online identification of the unknown model [78, 79]. Moreover, similar ideas for recursive estimation have also been developed for the Gaussian process regression [80]. Therefore, the combination of these two ideas, in the first place, and the adaptation of them so that they cope with the kernel based identification in a more suitable way, in the second place, forms an interesting and demanding research project.

All in all, it becomes clear that the kernel based LTI system identification methods form a promising research area that will be in the center of focus for many researchers in the years to follow. The close cooperation of the machine learning and the classical system identification approaches has still many things to offer in both communities. Moreover, the extension of these methods in other classes of systems has barely been investigated. For this reason, in the next part we will investigate if an extension of the kernel based methods to the identification of LPV systems in state-space form is possible, having as a guide the results and conclusions that were derived in this part of the thesis.

Part III

Kernel methods for the identification of LPV systems

Chapter 8

Kernel methods for SID of LPV systems

In this chapter we will give an overview of the available regularization techniques for the LPV-PBSID$_{opt}$ algorithm and propose two novel approaches, influenced by the developments in the LTI case, as shown in Part 2 of the present dissertation. More specifically, by elaborating on the discussion in Chapter 3, we will first focus on the Generalized Cross-Validation (GCV) based ridge regression, while the nuclear regularization method will also be briefly outlined. Then, in Section 8-2 we will open the way for a kernel based approach in the subspace identification of LPV systems. To this end, it is crucial to investigate the similarities and differences between the LTI and the LPV case. Finally, in Section 8-3 we will develop two frameworks for the subspace identification of LPV systems and we will highlight their advantages and disadvantage. The discussion of this Chapter will provide the theoretical framework to understand the results in Chapter 9. Due to the close relation of this chapter with Chapter 5, some of the necessary assumptions that we have to make here are identical to the ones in the latter one and so we will avoid repeating them here.

8-1 Regularization in SID methods for LPV systems

As we have seen in Chapter 3, the curse of dimensionality was undoubtedly one of the most severe limitations of the first global approaches for the identification of LPV systems in state-space form. It was therefore necessary to find ways to circumvent this problem. For a specific type of LPV systems a rigid way to cope with this problem was accomplished in 2007. More specifically, it was shown that when the scheduling parameter shows a specific structure, such as periodic [81] or piecewise constant [82], this limitation can be circumvented.

However, for the general case it was not possible to avoid the curse of dimensionality problem, since for many systems (such as oscillatory ones) the value of the past window had to be large to efficiently capture the characteristics of the impulse response, and so the number of rows in the Z matrix, $\tilde{q}$, was large. There were, though, improvements in some specific cases.

More specifically, when the unknown parameters are more than the available data points (for example, due to the curse of dimensionality) there should still be a way to derive a satisfying solution. In [50], this problem was tackled by considering the solution in the dual space (this method is called "the kernel method" but it should not be confused with the ideas developed in the next sections). The same result can also be derived within the framework of the Least Squares Support Vector Machines, as it was shown in [83]. This solution **corresponds to the minimum norm solution** and it can be computed by assuming that the solution is of the form $C\mathcal{K}^p = \alpha Z^T$, where Z was defined in (3-14) and $\mathcal{K}^p$ was defined in (3-15).

In order to clarify this approach, let us first remember that in (3-17) we have derived the solution for the case where Z is full row rank. However, if the number of data points is large, the matrix $Z \in \mathbb{R}^{(n_u+n_y)\tilde{q}\times N}$ can become ill-conditioned and may lose its full (row) rank property (e.g. when $N > (n_u + n_y)\tilde{q}$), thus making the inversion of the matrix ZZ^T impossible [53, p.138]. The latter could also be true when there is a poor excitation of the system, which leads to a row rank deficient Z. For this reason, as it was discussed in [50], a minimum norm solution is sought, as it was explained in the previous paragraph. However, it is usually the case that now the matrix Z^TZ is ill-conditioned and so regularization has to be employed once more as a remedy to this problem. Verdult discussed both the cases of Singular Value Decomposition (SVD) truncation (keeping and inverting only the most dominant singular values), as well as the **Ridge regression** LS problem for the identification of the LPV equivalent Markov parameters. Mathematically, the latter method means that instead of solving (3-16), we solve the LS problem

$$\min_{\alpha} ||Y - \alpha Z^T Z||_2^2 + \gamma^2 ||\alpha Z^T||_2^2, \tag{8-1}$$

where γ is the regularization parameter, while we have to clarify that in the dual space the LS problem of (3-17) corresponds to a **Tikhonov regularization** [1].

In the next sections we will establish the relationship between this regularized problem and the kernel regularization (which we will introduce later on). More specifically, we will show that the solution of (8-1) can be seen as a special case of the kernel method, when considered in a Bayesian framework.

The ridge regression is shown to deliver consistent results in many simulation examples. However, different approaches can also be followed. An interesting regularization method can be derived with the use of the nuclear norm. The motivation behind this method stems from the requirement to derive a minimum rank model (from a philosophical aspect every regularization method is based on the principle that “nature is simple” [57]).

In the LPV SID framework, the rank of the model is derived by the SVD decomposition in (B-7). Consequently, a reasonable regularization scheme would be to impose a minimum rank condition in this matrix. Nonetheless, the rank minimization is a non-convex problem. For this reason, an approximation of the rank condition can be introduced with the use of the nuclear norm (the sum of the singular values of a matrix), which eventually renders the minimization problem convex. The SID of LPV systems with the use of nuclear norm

[1] For reasons of clarity and according to [26] we will call a regularized LS problem as a **Ridge regression** when the regularization term is of the form $\gamma^2\theta\theta^T$ (where θ is a row vector containing the unknown coefficients and γ^2 is the regularization parameter), while we will call it a Tikhonov regularization LS problem when this term is of the form $\gamma^2\theta K\theta^T$, where K is a positive semi-definite regularization matrix.

regularization was presented in [84]. Following the notation, the minimization problem is mathematically expressed (in dual space) as

$$\min_{\alpha} ||Y - \alpha Z^T Z||_2^2 + \gamma^2 ||L(\alpha)||_*, \tag{8-2}$$

where $||\cdot||_*$ denotes the nuclear norm and $L(\alpha) = \Gamma^p \mathcal{K}^p$, since the latter quantity can be expressed as a linear function of α.

The results presented in [84] show the potential of this method. Nonetheless, the optimal selection of the regularization parameter γ is a rather difficult task, since there is no physical intuition about what is the ideal value and a wrong selection can even lead to a model that is less accurate than the one derived with the unregularized LPV-PBSID$_{opt}$ algorithm. Consequently, a multi-start approach is necessary to assure that the identified model will be a close approximation of the real one, thus increasing the computational complexity. Of course, once a γ value is selected, the optimization problem is convex. Moreover, another serious limitation of this method is related to the fact that the effort to find a minimum rank model may suppress the values of too many singular values of (B-7), thus making it cumbersome to estimate the rank of the system by the gap in the singular values [11].

So far we have discussed how the regularization methods can potentially increase the accuracy of the estimated model. It is nonetheless an open question what is the optimal regularization for the SID of LPV systems and how it can be achieved, while we also seek to avoid computation in large spaces. To answer these questions, we will turn our look to the recent developments for the LTI systems. By using these as a starting point for our thoughts, we will develop a novel regularization framework for the SID of LPV systems.

8-2 Introduction to kernel methods for LPV systems

The advent of the kernel based identification methods for LTI systems (primarily investigated within the framework of PEI methods), boosted by the development of Gaussian processes led to some very interesting results, as we saw in Part I of this thesis. Nonetheless, it is still an open question if these methods can succesfully be extended to SID methods for LPV systems. It is therefore of high interest to investigate if a synergy between kernel based methods and LPV systems could be established.

Up to our knowledge, there is only one related publication, which focuses on a kernel based PEI method for LPV systems [9]. In this publication a Gaussian process formulation was chosen to model the scheduling parameter dependency of the LPV system (which could might as well be described by a nonlinear function). The results showed the potential of this method in predicting the output of the system through the non-parametric framework. However, this method relied on some very specific assumptions such as the periodicity of the scheduling parameter and the input signal.

It is therefore obvious that there is a gap in the kernel-based SID of LPV systems and so we have to introduce a novel framework. It is also important to make as few as possible assumptions in order to enable the broad applicability of the new kernel based approaches. In order to tackle this problem, we will start with examining the similarities and differences between the LTI and LPV SID methods.

LPV and LTI SID methods: Similarities and Differences

A first necessary clarification is related to the difference between the impulse response coefficients in the LTI and the LPV case, the estimation of which is the main purpose of the kernel based methods. Let as make useof the notation introduced in Chapter 3. More specifically, let us investigate (3-10). It becomes directly obvious that in the LPV case the impulse response coefficients are **not only a function of the impulse response instant t, but also a function of the various scheduling parameters**, the number of which is related to t. In practice, this means that properties such as exponential stability are affected by the scheduling parameter (e.g. see Figure 8-1).

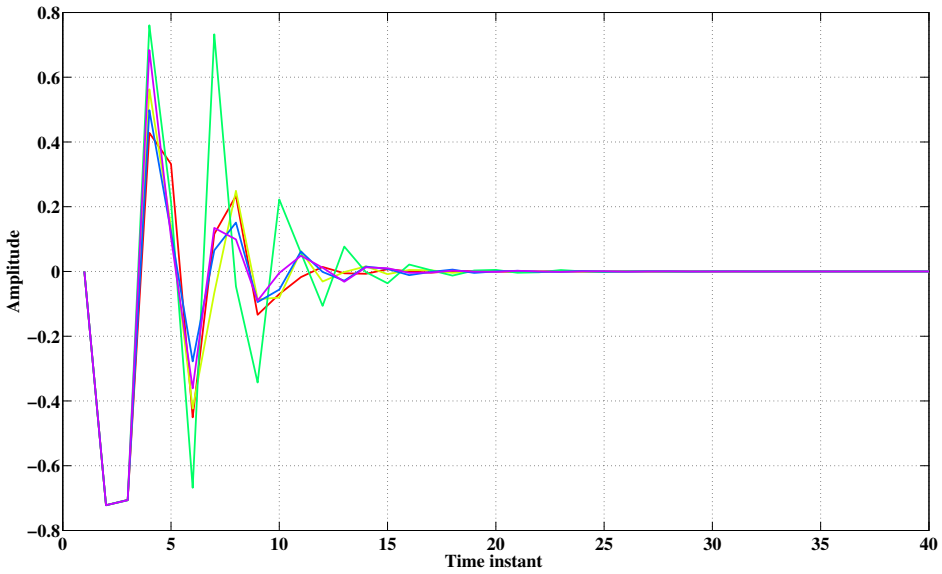


Figure 8-1: Five different implementations of the impulse response for a SISO 2nd order system with $m = 2$ and $\mu_k = [1\ v_k]$, where v_k is a normally distributed random variable. The characteristics of the impulse response are highly affected by the scheduling parameter.

This condition can also be expressed mathematically. For example, in the case where the input is zero, the exponential stability condition is expressed as follows [85].

Lemma 8.1. The LPV state equation where $u_k = 0$ is uniformly exponentially stable if, given a finite constant $\beta > 0$, there exist a finite constant $\gamma > 0$ and a constant $\lambda, 0 \leq \lambda < 1$ such that

$$\left\| \prod_{n=j}^{k-1} \left(A^{(1)} + \sum_{i=1}^{m} \mu_n^{(i)} A^{(i)} \right) \right\|_2 \leq \gamma \lambda^{k-j} \tag{8-3}$$

for all k, j such that $k > j$ and for all μ_k such that $\sup ||\mu_k||_2 \leq \beta$. △

On the other hand, the LTI systems are not affected by any time varying parameter. Actually, this difference is the most important one between the two classes of systems. Consequently,

a naive, straightforward implementation of the LTI related kernels to the LPV systems, identified by a method such as LPV-PBSID$_{opt}$ is bound to fail because it neglects the time-varying nature of the impulse response coefficients. Nonetheless, it is worth mentioning that other SID methods for LPV systems, such as the local methods, face no such obstacles since the LTI kernel based methods can be applied to each local system separately.

8-3 Novel approaches for the kernel based SID of LPV systems

In an effort to derive a kernel based SID algorithm for the LPV systems we followed two different methods. The first method, described in Section 8-3 is an adaptation of the kernel based LTI methods. On the other hand, the method presented in Sections 8-3 and 8-3 define a new approach, based on the modelling of the LPV system's impulse response as a Gaussian Process (GP), following the relatively recent developments in [5].

Assigning prior knowledge to the unknown coefficients

A first approach to overcome the μ dependency of the impulse response in the LPV case would be the following. Since there is an analytic description for each impulse response coefficient, presented in (3-10), a possible approach would be to assign a prior distribution directly to the LPV equivalent Markov coefficients instead of the impulse response coefficients. This is also schematically presented in Figure 8-2.

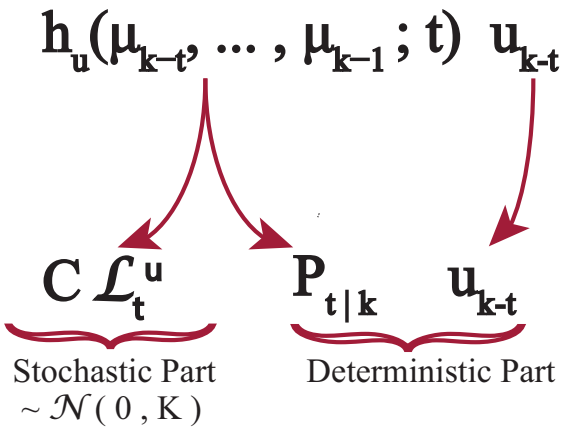


Figure 8-2: Schematic representation of the new stochastic framework for the LPV case. The scheduling parameter is lumped together with the input signal, while the unknown coefficients $C\mathcal{L}_t^u$ are treated as random variables.

By this simple trick, we can introduce a prior to the LPV equivalent Markov parameters, which are not μ-dependent. In this case, the steps to be taken are in close terms with the ones for the LTI case. First, to simplify the notation, let us denote in this section the vector $C\mathcal{K}^p$ as θ (or matrix in the MIMO case, but we still treat each output separately). The output equation is then given by

$$Y = \theta Z + E, \tag{8-4}$$

where the quantities Y, E and Z were defined in (3-12) and (3-14), respectively. The noise sequence e_k is assumed to be a zero-mean white-noise with normal distribution. Moreover, we will assign a prior normal distribution on the unknown coefficients θ. It is important to mention once more that these are necessary conditions so that the related calculations can be done in an analytic way. More general distributions though can be treated with the use of approximation techniques such as Markov Chain Monte Carlo (MCMC) or analytical approximations of the marginal likelihood [44].

Estimating the hyperparameters

We will describe the statistical properties of θ by $\theta \sim \mathcal{N}(0, K(\eta))$, where $K(\eta)$ is the covariance of the related coefficients, written as a function of some unknown hyperparameters η. Following the Bayesian paradigm (see Appendix C for more details), similar to the LTI case, we can set the unknown hyperparameters by minimizing the minus log marginal likelihood. The objective function that has to be minimized is given by

$$\begin{aligned} J(\eta) := -\log\left(p\left(Y|Z,\eta\right)\right) = \frac{N-p}{2}\log(2\pi) \\ + \frac{1}{2}\log\left(\det\left(Z^T K(\eta) Z + \sigma^2 I\right)\right) + \frac{1}{2} Y \left(Z^T K(\eta) Z + \sigma^2 I\right)^{-1} Y^T. \end{aligned} \tag{8-5}$$

An open issue that is not discussed in the section concerns the parametrization of the kernel $K(\eta)$. As we will see in the following sections, this is not a trivial selection, especially in the Subspace Identification (SID) framework for Linear Parameter Varying (LPV) systems. For now, it suffices to assume that the parametrization is known.

After minimizing the function (8-5), we set the unknown coefficients η to the estimated values. The final step is to compute the mean of the posterior estimate of $\theta, \mathbb{E}\left(\theta|Y,\eta,Z\right)$, the value of which is given by (C-9).

Remark 8.1. The GCV based kernel method proposed by Verdult in [50] can also be interpreted by the Bayesian framework. In this publication, the solution was of the form

$$\theta = Y\left(Z^T Z + \gamma I\right)^{-1} Z^T, \tag{8-6}$$

where $\gamma \geq 0$ is estimated via the GCV method [51]. Based on the assumptions about the statistical properties of θ and e_k it is straightforward to establish the relation. Imposing an identity covariance for θ and assuming that $\sigma^2 = \gamma$, it becomes obvious that Verdult's solution presented in (8-6) coincides with the Maximum a Posteriori (MAP) estimate of θ, see also (C-9) in Appendix C. △

Limitations of the kernel based approach for LPV systems

Up to this point, the analysis concerning the kernel-based method for the LPV SID approach is similar to the one presented for the LTI case. However, there is a major difference, related to the selection of the kernel structure. The latter problematic will become clear through the following example.

Example 8.1. Let us assume that we have an affine LPV system described by (3-4)-(3-5). For simplicity let us choose a past window of $p = 2$ and assume that only A matrix depends on the scheduling parameter. Then, by backward substitution, the output can be expressed as

$$y_k = C\left[\tilde{A}^{(1)^2} + \mu^{(2)}_{k-1}\tilde{A}^{(1)}\tilde{A}^{(2)} + \mu^{(2)}_{k-1}\tilde{A}^{(2)}\tilde{A}^{(1)} + \mu^{(2)}_{k-1}\mu^{(2)}_{k-2}\tilde{A}^{(2)}\tilde{A}^{(2)}\right]x_{k-2} \tag{8-7}$$

$$+ C\left[Bu_{k-1} + \left(\tilde{A}^{(1)} + \mu^{(2)}_{k-2}\tilde{A}^{(2)}\right)Bu_{k-2}\right] \tag{8-8}$$

$$+ C\left[Ky_{k-1} + \left(\tilde{A}^{(1)} + \mu^{(2)}_{k-2}\tilde{A}^{(2)}\right)Ky_{k-2}\right] \tag{8-9}$$

Let us now assume, just for the purposes of this example, that the past window is large enough such that terms which are multiplied by x_{k-2} are zero. Now let us investigate only the terms related to the past input signals, while the same analysis holds for the terms related to the past output signals. We can immediately observe that there is only one unknown term that corresponds to the impulse response instant $t = 1$, namely CB. On the other hand, there are two unknown terms related to the impulse response instant $t = 2$. By increasing the past window, the number of coefficients that refer to an "older" input grows exponentially. This is of course not a new result; it is actually the curse of dimensionality, for which we discussed about in Chapter 3. As we will see though, this result is also creating complications when it comes to the assignment of a prior knowledge in the LPV equivalent Markov coefficients.

Based on the observations in the previous example, it is evident that the size of the kernel will also increase exponentially with p, due to the curse of dimensionality. A kernel of this type would have a form similar to the one in Figure 8-3.

Figure 8-3 highlights in a clear way the complications of this approach. In this LPV kernel the cross terms are not any more always referring to the covariance of coefficients that belong to different impulse response instants (depicted in yellow in Figure 8-3), since they could also refer to the covariance of coefficients that belong to the **same** impulse response instant (the off diagonal elements of the grey block matrices in the same figure). Therefore, it is not any more clear how we can create a kernel structure that incorporates all these characteristics. On the other hand, by employing the kernels used in the LTI case, then it is the case that the richer a structure is, the more restrictive becomes a direct implementation in the LPV case. In order to clarify this point, we will give a small example.

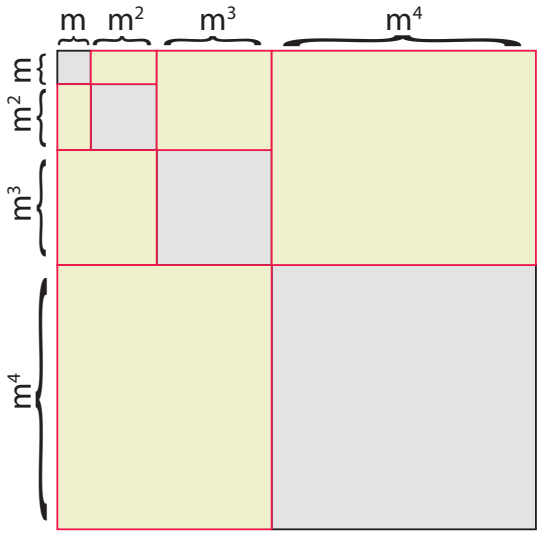


Figure 8-3: General shape of an LPV kernel. The grey boxes correspond to the elements that are referring to the same impulse response instant, based on the assigned number t and they are of dimensions $m \times m$, $m^2 \times m^2$ etc.

Example 8.2. Let us assume that we would like to use a 1st order stable spline kernel, adjusted for the LPV case. We also assume that $m = 2$ and that the B matrix is not μ-dependent. Here we will only focus on the input, but the same analysis holds for the output or a MIMO system.
Let as choose a past window $p = 3$ and a $\beta = 0.1$. Then, the 1st order stable spline kernel for the LTI case would be the following.

$$\mathcal{K}_{LTI} = \begin{bmatrix} 0.9048 & 0.8187 & 0.7408 \\ 0.8187 & 0.8187 & 0.7408 \\ 0.7408 & 0.7408 & 0.7408 \end{bmatrix} \tag{8-10}$$

In the LPV case, we would actually stretch this matrix, so that it assigns a prior distribution in all the related coefficients. The covariance matrix is given by

$$\mathbb{E}\left[\begin{bmatrix} CB \\ C\tilde{A}^{(1)}B \\ C\tilde{A}^{(2)}B \\ C\tilde{A}^{(1)^2}B \\ C\tilde{A}^{(1)}\tilde{A}^{(2)}B \\ C\tilde{A}^{(2)}\tilde{A}^{(1)}B \\ C\tilde{A}^{(2)^2}B \end{bmatrix}[\cdot]^T\right] = \begin{bmatrix} 0.9048 & 0.8187 & 0.8187 & 0.7408 & 0.7408 & 0.7408 & 0.7408 \\ 0.8187 & 0.8187 & 0.8187 & 0.7408 & 0.7408 & 0.7408 & 0.7408 \\ 0.8187 & 0.8187 & 0.8187 & 0.7408 & 0.7408 & 0.7408 & 0.7408 \\ 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 \\ 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 \\ 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 \\ 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 & 0.7408 \end{bmatrix} \tag{8-11}$$

It is clear that the LPV equivalent 1st order stable spline structure is rather restrictive, since it assumes that many different coefficients are described by the same statistical properties. Moreover, it is worth noticing that the latter kernel is not any more full rank, so it is positive semi-definite, while the 1st order stable spline, like the one in (8-10), is positive definite.

Kernel selection

In the previous section we showed that the LPV equivalent of the stable spline kernel of 1st order poses a strict restriction in the structure of the LPV kernel. In general, a straightforward application of the kernel based SID methods for LTI systems is bound to offer limited or no improvements. However, **simpler kernel structures** may still be able to surpass the modern regularization methods for LPV systems. For example, a diagonal type of kernel can be of use in the LPV case. Following the notation in Section 5-3, the LPV equivalent diagonal kernel is given by

$$\mathcal{K} = diag\left(\underbrace{\lambda\alpha^1, \cdots \lambda\alpha^1}_{m\ elements}, \underbrace{\lambda\alpha^2, \cdots \lambda\alpha^2}_{m^2\ elements}, \cdots \underbrace{\lambda\alpha^p, \cdots \lambda\alpha^p}_{m^p\ elements},\right)\ ,\ \lambda > 0\ ,\ \alpha \in (0,1) \tag{8-12}$$

Two main characteristics are attributed to this kernel. Firstly, by assuming a not so restrictive kernel structure, is may be possible to improve the accuracy of the identified model. Secondly, the results are expected to be especially good when the values of the LPV equivalent Markov parameters that refer to the same impulse response instant are close to each other (e.g. the values of $C\tilde{A}^{(1)}B$ and $C\tilde{A}^{(2)}B$ in Example 8.2). At this point it is also necessary to clarify that we always assume that all A, B and K matrices depend on the scheduling parameter, unless otherwise specified.

In practice, we will assign to the kernels the same characteristics as we did in the LTI case. This means that for each signal (inputs or outputs) we will assign a specific kernel. For example, in the SISO case we will create the kernels $\mathcal{K}^u$ and $\mathcal{K}^y$, assuming that there is no correlation between the coefficients that correspond to a different signal, e.g. $\mathbb{E}\left[\left(C\tilde{A}^{(1)}B\right)\left(C\tilde{A}^{(1)}K\right)\right] = 0$. Therefore, the total kernel $\mathcal{K}$ will be of the form

$$\mathcal{K} = \begin{bmatrix} \mathcal{K}^u & 0 \\ 0 & \mathcal{K}^y \end{bmatrix}. \tag{8-13}$$

Finally, we will also take advantage here of the fact that the exponential decay of the LPV equivalent Markov parameters is attributed to the various products between the $A^{(i)}$ matrices. Since these products appear in the same way at the coefficients of all the signals, we will use a common α coefficient for all the input and output signals and a different λ for each one of them. Of course, for each different output we will re-estimate the related hyperparameters following the same rationale.

Optimal kernel

Before we proceed to the second kernel based method, it is useful to answer an important question, namely what is the optimal regularization for the VARX based SID methods for LPV systems. This question is given in the following Lemma.

Lemma 8.2. The output of an LPV system is described by (3-10). Based on the assumption that for a large enough past window value p the value of $\phi_{k,j} \to 0$, the output can be described by (3-13). Then, by forming the output matrix Y as shown in (3-12) and combining the scheduling parameter values together with the past inputs and outputs as shown in (3-14), we end up with the data generating system

$$Y = \theta Z + E \tag{8-14}$$

where $E \sim \mathcal{N}\left(0\ ,\ \sigma^2 I\right)$ and $\theta = C\mathcal{K}^p$. Then, the optimal regularization problem (in terms of Mean Squared Error (MSE)) is formed as follows.

$$\min_{\hat{\theta}} ||Y - \hat{\theta}Z||_2^2 + \sigma^2 \hat{\theta}(\theta^T\theta)^{-1}\hat{\theta}^T \tag{8-15}$$

where $\hat{\theta}$ is the estimated value and θ is the real one. △

Proof 8.1. Following the construction of Z matrix, we end up with a problem similar to the corresponding Linear Time Invariant (LTI) case. Consequently, the proof is identical to the one presented in [54]. □

Similar to the LTI case, it is obvious that the optimal regularization term requires the knowledge of the true system, which is of course unknown in a real-life experiment.

Modelling the LPV impulse response as a Gaussian process

In this previous section we investigated the possibility of assigning a prior distribution directly to the LPV equivalent Markov parameters. However, it is apparent that this approach faces some serious limitations. For this reason, we will now turn our attention to a different approach. More specifically, **we will assign a prior distribution directly on the impulse response coefficients**. However, there are two major considerations that have to be taken into account. Firstly, the impulse response coefficients, as they were defined in (3-11) are **time varying**, due to their dependence on μ. Therefore, if we want to model the impulse response we cannot use the kernels presented so far. Secondly, the LPV-PBSID$_{opt}$ algorithm requires the LPV equivalent Markov parameters. It is therefore necessary to estimate these coefficients. In this case, the estimation of the impulse response coefficients will be an extra intermediate step before the estimation of the Markov parameters.

Impulse response as a μ dependent Gaussian process

As we have seen in (3-11), the impulse response coefficients are affected by two coefficients, namely the various scheduling parameters μ, which are a function of k and t, as well as explicitly by the impulse response instant, t, since for a different t we have a different expression for the corresponding impulse response instant.

Following once more an approach similar to the ones that we followed in the previous sections, we will model the impulse response coefficients as zero mean Gaussian processes. By assuming that the coefficients of one signal are uncorrelated with the ones from another signal, we can define their covariance properties (kernels) as follows.

$$\begin{aligned}\mathbb{E}\left[h^u(\mu_{k-t},\dots,\mu_{k-1};t)h^u(\mu_{k'-t'},\dots,\mu_{k'-1};t')\right] &= k\left(h^u(\mu_{k-t},\dots,\mu_{k-1},\mu_{k'-t'},\dots,\mu_{k'-1};t,t')\right.\\ \mathbb{E}\left[h^u(\mu_{k-t},\dots,\mu_{k-1};t)h^y(\mu_{k'-t'},\dots,\mu_{k'-1};t')\right] &= 0 \;\textit{for every}\; t,t',k,k'\in\mathbb{N},\end{aligned} \tag{8-16}$$

The relation with the LTI case can be established by removing all the μ terms from (8-16). We will now show what kernel structure can be chosen to tackle this problem.

Selection of kernel structure

The time varying nature of the impulse response coefficients is their most important attribute. The notion of the exponential decay of the impulse response coefficients is more complicated in the LPV case due to the μ dependency, as it was shown in Lemma 8.1. Therefore, it is appealing to drop the dependency of the kernels in (8-16) on the impulse response instants t and instead try to incorporate this exponential decay in a different way.

A broadly used kernel is the Radial Basis Function (RBF) [46]. Its most celebrated characteristic is the so called non-degeneracy [5], that is to say, it can be expressed as a function of possibly infinite basis functions. In practice, it will be a function of N basis functions, where N is the number of available data points, but for large enough N it can lead to a good approximation of the underlying, possibly nonlinear function.

In general, the RBF kernel (covariance) between two different evaluations of a function $f : D \to \mathbb{R}$ at the points u_1, u_2 is given by

$$\begin{aligned}&\mathbb{E}[f(u_1)f(u_2)]\\ &= \sigma_f^2 \exp -\frac{||u_1-u_2||_2^2}{\lambda_f^2}\end{aligned} \tag{8-17}$$

The connection with the impulse response of the LPV systems is direct; the inputs in the latter case are the scheduling parameters. Due to the affine property, the scheduling parameter $\mu_k^{(1)}$ is always 1 for every $k \in \mathbb{N}$ and for this reason it will not be used as an input in the RBF kernel. Moreover, due to the dynamic dependency on the μ coefficients in (3-11), we will define the RBF kernel as (here we show the case where $m = 2$)

$$[K_t^u]_{i,j} = \mathbb{E}[h^u(\mu_{p+i-t},\dots,\mu_{p+i-1};t)h^u(\mu_{p+j-t},\dots,\mu_{p+j-1};t)] = \sigma_{t,u}^2 \exp\left(-\frac{\left\|\begin{bmatrix}\mu_{p+i-t}-\mu_{p+j-t}\\ \vdots \\ \mu_{p+i-1}-\mu_{p+j-1}\end{bmatrix}\right\|_2^2}{\lambda_{t,u}^2}\right). \tag{8-18}$$

where $i,j = \{1,\dots,N-p\}$. The brackets with the right bottom index refer to a specific position in a matrix.

With the use of (8-18) we can construct the kernel matrices K_t^u and K_t^y for $t = \{1,\dots,p\}$, while each of these matrices will be of dimensions $(N-p)\times(N-p)$.

By defining the kernels in this way, it becomes clear that the function that describes the impulse response at a specific instant t is different from a function of different t. Moreover, based on the definition of the impulse response coefficients in (3-11), the number of inputs (that is to say, the number of the involved μ parameters) is different for each impulse response instant. Therefore, based on the RBF kernel choice it is not clear how we should define the inputs of the kernel in the cross-variance terms. Moreover, due to the independent modelling of the impulse response instants, for a past window p and for an RBF kernel selection, we have to **additionally** estimate $2\sum_{r=1}^{p-1} r$ hyperparameters for each signal, instead of $2p$ elements, if we assume that the cross-variances between different impulse response instants is not zero. Still, by modelling only the diagonal terms, we have to estimate in total $2p(n_u+n_y)n_y$ hyperparameters (2 hyperparameters for each signal for each impulse response instant t and this procedure is repeated for all the output channels). It is therefore evident that the computational burden is huge and so it necessary to alleviate it, if we want to render this approach practically feasible.

Practical considerations concerning the kernel structure selection

In order to reduce the number of hyperparameters, we have to investigate what is their role in the kernel and make the connection between them and the analytic expression that we have for each impulse response instant.

First let us investigate the role of $\lambda_{u,t}$, $\lambda_{y,t}$. Since they refer to the same instant, let's take as an example $t=2$. Using (3-7), (3-8) and (3-11) the two impulse responses, when both B and K are μ-dependent and $m=2$, are given by

$$\begin{aligned} h^u(\mu_{k-2},\mu_{k-1};2) &= C\tilde{A}^{(1)}B^{(1)} + C\tilde{A}^{(1)}B^{(2)}\mu_{k-2}^{(2)} + C\tilde{A}^{(2)}B^{(1)}\mu_{k-1}^{(2)} + C\tilde{A}^{(2)}B^{(2)}\mu_{k-1}^{(2)}\mu_{k-2}^{(2)},\\ h^y(\mu_{k-2},\mu_{k-1};2) &= C\tilde{A}^{(1)}K^{(1)} + C\tilde{A}^{(1)}K^{(2)}\mu_{k-2}^{(2)} + C\tilde{A}^{(2)}K^{(1)}\mu_{k-1}^{(2)} + C\tilde{A}^{(2)}K^{(2)}\mu_{k-1}^{(2)}\mu_{k-2}^{(2)}. \end{aligned} \tag{8-19}$$

Moreover, let us also write down the expressions for $t=1$.

$$\begin{aligned} h^u(\mu_{k-1};1) &= CB^{(1)} + CB^{(2)}\mu_{k-1}^{(2)}, \\ h^y(\mu_{k-1};1) &= CK^{(1)} + CK^{(2)}\mu_{k-1}^{(2)}. \end{aligned} \tag{8-20}$$

If we investigate (8-18) in a qualitative way, we could say that the term $\lambda_{t,u}$ (similarly for $\lambda_{t,y}$) actually shows how far the inputs signals should be such that they are uncorrelated. If this hyperparameter is very large, it means that even for two inputs with high difference, the correlation will be relatively high and vice versa. Therefore, we could use this parameter to capture the exponential decay rate, which is mainly attributed to the various $\tilde{A}^{(i)}$ coefficients. Moreover, by investigating (8-19) and (8-20) we see that the $\tilde{A}^{(i)}$ coefficients enter in the same way in all impulse response coefficients of the same instant t. Therefore, we can use the same hyperparameter for the impulse response of all signals, as long as they refer to the same instant t. So, from now on we will simply write that $\lambda_t = \lambda_{t,u} = \lambda_{t,y}$. Moreover, in order to take into account the effect of C matrix, we will estimate a new set of λ_t coefficients when there are multiple outputs. In total, we require now pn_y different λ_t instead of $pn_y(n_u + n_y)$ that we would require if we used a full parametrization.

Another link between the kernel structure and the analytic expression for the impulse response coefficients could also be established. By investigating once more (8-19) and (8-20), we see that all of the impulse response coefficients that correspond to the **same signal** (e.g. to the input) are multiplied on the left by some vectors, namely $B^{(1)}$ and $B^{(2)}$ for the inputs and $K^{(1)}$ and $K^{(2)}$ for the outputs. Therefore, we could view these vectors as scaling factors. Of course, this assumption holds totally true when the systems are SISO of 1st order (such that the B and K matrices are scalars) and all the local vectors are the same or when they are not μ-dependent. In order to alleviate the computational burden, though, we could establish this relation and so **use one** σ_u **for each input signal** and similarly for the σ_y. By applying this relation, instead of requiring in total $p(n_u + n_y)n_y$ σ_u and σ_y hyperparameters, we will need $(n_u + n_y)n_y$.

Finally, another approach could be followed that shows strong similarities with the LTI case. More specifically, the λ_t coefficients can be related to each other. This can be achieved by introducing the following transformation.

$$\lambda_t^2 = \frac{\lambda_{\text{common}}^2}{t}. \tag{8-21}$$

With the use of (8-21) the exponential decay rate among the impulse response coefficients is incorporated in the RBF kernel. Although this can be restrictive in some specific cases (due to the additional relation that we introduced), this is an interesting approach, since (8-21) reflects a real characteristic of the impulse response coefficients. Moreover, in cases where the past window is very large, the simplification of the marginal likelihood problem through this assumption can be crucial in order to avoid local minima. By combining this assumptions with the assumptions regarding σ_u, σ_y, the total number of needed coefficients to be estimated is $(n_u + n_y + 1)n_y$.

To sum up, we give the four different methods together with their main characteristics in Table 8-1. Based on this table we assume that Approach 3 includes the assumptions of Approach 2 and Approach 4 includes the assumptions of Approach 3 and 2.

Table 8-1: Different approaches for the parametrization of the kernel for the LPV case

	Approach 1	Approach 2	Approach 3	Approach 4
Characteristics	Full parametr.	Common λ_t for all signals	Common σ_u (σ_y) for each input (output)	Common λ for all signals
# of hyperpar.	$2p(n_u+n_y)n_y$	$pn_y(n_u+n_y+1)$	$n_y(p+n_u+n_y)$	$n_y(1+n_u+n_y)$

Estimation the unknown hyperparameters

The estimation of the hyperparameters is not much different from what we have seen so far, e.g. (8-5). However, in this case a matrix formulation of the deterministic part of the data (which now contains only the past inputs and outputs but not the scheduling parameters) is not really useful. For this reason we will follow a different paradigm, using the following definition and lemma.

Definition 8.1. The matrix $\mathcal{K}$ is of dimensions $(N-p)\times(N-p)$ and its value at row i and column j is given by the expression

$$\begin{aligned}[\mathcal{K}]_{i,j} = &\sum_{t=1}^{p} u_{i+p-t}\,[K_t^u]_{i,j}\,u_{j+p-t} \\ &+\sum_{t=1}^{p} y_{i+p-t}\,[K_t^y]_{i,j}\,y_{j+p-t}.\end{aligned} \tag{8-22}$$

△

Based on this definition and with the use of (8-18) we can construct the matrix $\mathcal{K}$. Subsequently, we can describe the statistical properties of the outputs, given in the following lemma.

Lemma 8.3. The mean m_Y and the covariance Σ_Y of the output vector Y (assuming that h^u and h^y are also uncorrelated with the innovation sequence) are expressed as

$$Y \sim \mathcal{N}\left(0,\ \mathcal{K}+\sigma^2 I_{N-p}\right). \tag{8-23}$$

△

Proof 8.2. The proof follows from straightforward calculations, by taking into account the statistical properties of the noise variance and the coefficients $h^u(\mu_{k-t},\ldots,\mu_{k-1};t)$ and $h^y(\mu_{k-t},\ldots,\mu_{k-1};t)$, given in (3-11). □

Now that we have defined all the required quantities, we will determine the unknown hyperparameters via the minimization of minus the logarithm of the marginal likelihood function, which is given by

$$\begin{aligned}J(\eta) := -\log\left(p\left(Y|\eta\right)\right) = &\frac{N-p}{2}\log(2\pi) \\ &+\frac{1}{2}\log\left(\det\left(\Sigma_Y\left(\eta\right)\right)\right)+\frac{1}{2}Y\Sigma_Y^{-1}\left(\eta\right)Y^T.\end{aligned} \tag{8-24}$$

Computation of the value of the GP at the training points

The next step is to compute the values of the functions $h^u(\mu_{k-t},\dots,\mu_{k-1};t)$ and $h^y(\mu_{k-t},\dots,\mu_{k-1};t)$ for $t=\{1,\dots,p\}$, evaluated at the training points $k=\{p+1,\dots,N\}$. To do so, we are looking for the a posteriori estimates $p\left(h^y|Y,\eta\right)$ and $p\left(h^u|Y,\eta\right)$. We note that the correspondence of this method with the Tikhonov regularization in a Reproducing Kernel Hilbert Space can also be proven [5], which will actually reveal its usefulness in the next section.

The a posteriori estimate can be derived in an analytical way [54]. The value of a Gaussian process at a specific training point for h^u (similarly for h^y) is given by

$$\begin{aligned} &h^u(\mu_{k-t},\dots,\mu_{k-1};t) = \\ &\quad Y\left(\mathcal{K}+\sigma^2 I_{N-p}\right)^{-1}\begin{bmatrix} u_{k-t}\left[K_t^u\right]_{1,k-p} \\ u_{k-t+1}\left[K_t^u\right]_{2,k-p} \\ \vdots \\ u_{k-t+N-p-1}\left[K_t^u\right]_{N-p,k-p}\end{bmatrix}. \end{aligned} \tag{8-25}$$

With the use of (8-25) we can compute the value of h^u (similarly for h^y) for each $k=\{p+1,\ \dots,\ N\}$ and $t=\{1,\ \dots,\ p\}$.

Derivation of the LPV equivalent VARX coefficients

Following the estimation of the hyperparameters with the use of (8-24) and the impulse response functions at the training points based on (8-25), we now need to estimate the unknown coefficients which are contained in $\mathcal{L}_1^u,\mathcal{L}_1^y,\cdots,\mathcal{L}_p^u,\mathcal{L}_p^y$, defined in (3-7). In order to end up with a unique solution of the system matrices, the following condition should hold.

$$\operatorname{rank}\left(\underbrace{\begin{bmatrix}P_{t|p+1},\ P_{t|p+2},\ \cdots,\ P_{t|N}\end{bmatrix}}_{m^t\times N-p}\right) = m^t \text{ for each } t\in\mathbb{N}^+, \tag{8-26}$$

where $p,N\in\mathbb{N}^+$, $p<N$ and $N-p\geq m^t$. This condition can be perceived as a persistency of excitation condition and it is needed to uniquely determine the impulse response coefficients, evaluated at the training points.

Using the result of (8-26) we can compute a unique solution for each $\mathcal{L}_t^u$ (and similarly for each $\mathcal{L}_t^y$) with the use of the Least Squares (LS) method. The equality that corresponds to this problem is

$$\begin{bmatrix} h^u(\mu_{p+1-t},\dots,\mu_p;t) \\ h^u(\mu_{p+2-t},\dots,\mu_{p+1};t) \\ \vdots \\ h^u(\mu_{N-t},\dots,\mu_{N-1};t)\end{bmatrix}^T = \mathcal{L}_t^u\begin{bmatrix} P_{t|p+1} \\ P_{t|p+2} \\ \vdots \\ P_{t|N}\end{bmatrix}^T. \tag{8-27}$$

The LS problem, based on (8-27), has to be solved for each t (similarly for h^y). After having estimated the system parameters, the algorithm proceeds as it is explained in Appendix B.

Remark 8.2. In the specific case where all the states are measured then the state sequence can be directly derived by inverting the C matrix. For example, assume that we have a second order SISO LPV system, with $C = diag\,(0.1,\, 0.2)$. What we have estimated are the impulse response coefficients $h^u(k;t)$ and $h^y(k;t)$ for $k = \{p+1, \ldots, N\}$ and $t = \{1, \ldots, p\}$. Based on these coefficients, we can directly get an estimate of the output y_k without requiring the estimation of the LPV equivalent Markov parameters. By inverting the C matrix, an estimate of the states can be derived. This of course holds true for the LPV-PBSID$_{opt}$ algorithm, too, so the state estimation step via the extended observability times controllability matrix can be skipped. However, in the case of the LPV-K&PBSID$_{opt}$ algorithm, the estimated quantities are $p(N-p)(n_u+n_y)n_y$ (in the general MIMO case), while in the LPV-PBSID$_{opt}$ algorithm they are $\tilde{q}(n_u+n_y)n_y$. Therefore, in this specific case the LPV-K&PBSID$_{opt}$ is not suffering from the curse of dimensionality, therefore in cases where $\tilde{q} > (N-p)p$ the LPV-K&PBSID$_{opt}$ requires the estimation of less coefficients. △

Summary of the proposed algorithm - The LPV-K&PBSID$_{opt}$ algorithm

It is now time to summarize the proposed algorithm, which we will call it LPV Kernel and PBSID$_{opt}$, LPV-K&PBSID$_{opt}$.

Algorithm 8.1. LPV-K&PBSID$_{opt}$

1. Create the matrix Y and the kernel $\mathcal{K}$ following (3-12) and (8-22).
2. Determine the value of the hyperparameters that minimize the value of (8-24)
3. Estimate the value of the Gaussian processes h^u and h^y, evaluated at the training points $k = \{p+1, \ldots, N\}$ based on (8-26).
4. Solve the related Least Squares problems that correspond to (8-27) in order to estimate the system parameters $\mathcal{L}_t^u, \mathcal{L}_t^y,\ t = \{1, \cdots, p\}$.
5. Proceed as it is explained in sections 4.2 and 4.3 of [30] to estimate the state sequence and then the unknown matrices of the model (3-4)-(3-5) (see also Appendix B).

An improvement on the LPV-K&PBSID$_{opt}$ algorithm based on RKHS theory

In Section 8-3 we outlined a novel method for the estimation of an LPV state-space model in affine form. The extra step of estimating the MAP estimates of the impulse response coefficients, evaluated at the training points, is necessary for the subsequent estimation of the LPV equivalent Markov parameters. This actually reveals the inherent trade-off; by increasing the complexity of the algorithm, we expect that the accuracy of the estimated Markov parameters will be higher, due to the sophisticated regularization of the impulse response coefficients (when using a regularization point of view).

However, this expectation may never become a reality in some specific cases. In (8-26) we made a rather strong assumption regarding the scheduling parameter sequence matrix. Unfortunately, this assumption can be easily violated. This is for example always the case when we use a large past window, leading to $m^t > N$ for some $t = [1, \ldots, p]$. The same

undesirable result may also be observed for small past window values, when the number of local systems is large. In such cases the related matrices will never be full row rank, thus leading to an under-determined Least Squares (LS) problem, which will definitely lead to a suboptimal estimation of the Markov parameters.

Fortunately, a different approach could be followed, based on the Reproducing Kernel Hilbert Space (RKHS) theory, analysed in Appendix D. This is actually a different way of viewing the kernel based methods, instead of a fully stochastic framework that uses notions such as the MAP estimate. In the LTI case, the RKHS and the stochastic framework were leading to the same solution. However, in the LPV case there are a few differences between the two frameworks. In order to show how the estimates can be derived in the RKHS framework, we need to introduce some necessary definitions first. These definitions will be given for the input signal, but similar quantities have also to be defined for the output signal and, in general, for all the signals of a MIMO system, as long as we treat the information from each output signal separately.

Definition 8.2. The impulse response coefficients of instant t, evaluated at the training points are gathered in the row vector H_t^u, given by

$$H_t^u = \underbrace{\begin{bmatrix} h^u(\mu_{p+1-t}, \dots, \mu_p; t) \\ h^u(\mu_{p+2-t}, \dots, \mu_{p+1}; t) \\ \vdots \\ h^u(\mu_{N-t}, \dots, \mu_{N-1}; t) \end{bmatrix}^T}_{1 \times N-p} = \underbrace{C\theta_t^u}_{1 \times m^t} \underbrace{\begin{bmatrix} P_{t|p+1} \\ P_{t|p+2} \\ \vdots \\ P_{t|N} \end{bmatrix}^T}_{m^t \times N-p}. \tag{8-28}$$

△

With this definition, we can collect the impulse response coefficients for all $t = [1, \dots, p]$ as follows.

$$H^u = \begin{bmatrix} H_1^u & \cdots & H_p^u \end{bmatrix} \tag{8-29}$$

Based on the assumptions of the previous section, we can construct the related kernels with the use of (8-18). The total $\mathcal{K}^u$ matrix take the following block-diagonal form.

$$\mathcal{K}^u = \underbrace{\text{block-diag}\left(\underbrace{K_1^u}_{N-p \times N-p}, \dots, K_p^u \right)}_{p(N-p) \times p(N-p)} \tag{8-30}$$

Finally, we also need to modify the data matrices in order to keep consistency between the equations.

Definition 8.3. The modified data equations are given by (using Matlab notation inside the parentheses)

$$U_{mod} = \underbrace{\begin{bmatrix} \text{diag}\left(U_{1,p,N-p}\left(1,:\right)\right) \\ \text{diag}\left(U_{1,p,N-p}\left(2,:\right)\right) \\ \cdots \\ \text{diag}\left(U_{1,p,N-p}\left(p,:\right)\right) \end{bmatrix}}_{p(N-p)\times N-p}, \tag{8-31}$$

where the matrix $U_{1,p,N-p}$ is defined in (2-8). In other terms, in order to keep consistency we have to take each row of the data matrix and place its elements in a diagonal form. △

Lemma 8.4. The output row vector for a SISO system is described by

$$Y = H^u U_{mod} + H^y Y_{mod} + E \tag{8-32}$$

△

Proof 8.3. The proof follows by straightforward calculations, using equations (8-30) and (8-31). □

Finally, we are in position to formulate the corresponding Tikhonov regularization LS problem, following the RKHS theory. This is given by

$$\begin{aligned} J = ||Y - H^u U_{mod} - H^y Y_{mod}||_2^2 + \frac{\sigma^2}{\sigma_u^2}||h^u(\mu_{k-1};1)||_{\mathcal{H}_1^u}^2 + \ldots + \frac{\sigma^2}{\sigma_u^2}||h^u(\mu_{k-p},\ldots,\mu_{k-1};p)||_{\mathcal{H}_p^u}^2 \\ + \frac{\sigma^2}{\sigma_y^2}||h^y(\mu_{k-1};1)||_{\mathcal{H}_1^y}^2 + \ldots + \frac{\sigma^2}{\sigma_y^2}||h^y(\mu_{k-p},\ldots,\mu_{k-1};p)||_{\mathcal{H}_p^y}^2 \Leftrightarrow \\ ||Y - H^u U_{mod} - H^y Y_{mod}||_2^2 + \sigma^2 H^u(\mathcal{K}^u)^{-1}H^{uT} + \sigma^2 H^y(\mathcal{K}^y)^{-1}H^{yT}, \end{aligned} \tag{8-33}$$

where, in general, the notation $\mathcal{H}_i^j$ is used to denote a specific RKHS which corresponds to the impulse response function at instant i of the signal j. By solving (8-33) and making use of the matrix inversion lemma, we end up with the same solution as in (8-25), as expected. However, in this specific case we can make use of the analytic expression (8-28). By this way, we can skip step 4 in Algorithm 8.1. In order to show this in a clear way, let us first introduce one more matrix, $\mathcal{M}_t$, which is given by

$$\mathcal{M}_t = \underbrace{\begin{bmatrix} P_{t|p+1} \\ P_{t|p+2} \\ \vdots \\ P_{t|N} \end{bmatrix}^T}_{m^t\times N-p} \tag{8-34}$$

We also define the block diagonal matrix, given by

$$\mathcal{M} = \begin{bmatrix} \mathcal{M}_1 & & & \\ & \mathcal{M}_2 & 0 & \\ & & \ddots & \\ & 0 & & \ddots \\ & & & \mathcal{M}_p \end{bmatrix} \tag{8-35}$$

Now we can re-write (8-33) as follows.

$$J = \left\| Y - C\mathcal{K}^{(p)} \begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix} \begin{bmatrix} U_{mod} \\ Y_{mod} \end{bmatrix} \right\|_2^2 + \sigma^2 C\mathcal{K}^{(p)} \begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix} \begin{bmatrix} \mathcal{K}^u & 0 \\ 0 & \mathcal{K}^y \end{bmatrix}^{-1} \begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix}^T \mathcal{K}^{(p)^T} C^T \tag{8-36}$$

where $\mathcal{K}^{(p)}$ was defined in (3-15). It is also noteworthy that in (8-36) we have completely substituted the impulse response coefficients, evaluated at the training points, by their analytic expressions. Therefore we **can also skip Step 3 in Algorithm 8.1**. By requiring that

$$null\left(\left(\begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix} \begin{bmatrix} U_{mod} \\ Y_{mod} \end{bmatrix}\right)^T\right) \cap null\left(\left(\sigma \begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix} \begin{bmatrix} \mathcal{K}^u & 0 \\ 0 & \mathcal{K}^y \end{bmatrix}^{-1/2}\right)^T\right) = \emptyset \tag{8-37}$$

the solution can be directly computed and it is given by

$$\mathcal{K}^{(p)} = Y \begin{bmatrix} U_{mod} \\ Y_{mod} \end{bmatrix}^T \begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix}^T \left(\begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix} \begin{bmatrix} U_{mod} \\ Y_{mod} \end{bmatrix} \begin{bmatrix} U_{mod} \\ Y_{mod} \end{bmatrix}^T \begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix}^T + \begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix} \begin{bmatrix} \mathcal{K}^u & 0 \\ 0 & \mathcal{K}^y \end{bmatrix}^{-1} \begin{bmatrix} \mathcal{M} & 0 \\ 0 & \mathcal{M} \end{bmatrix}^T \right)^{-1} \tag{8-38}$$

However, for the equation (8-38) **we cannot use the matrix inversion lemma because the matrix product in the second line of the equation is not necessarily invertible**.

Remark 8.3. In case where not both B and K matrices are μ-dependent, then the two involved $\mathcal{M}$ will be different, following the backward substitution in the LPV innovation model (3-4) -(3-5). △

All in all, this new approach manages to avoid two steps, compared to the LPV-K&PBSID$_{opt}$ algorithm, but it comes with the cost of an extra inversion, namely of the block-diagonal matrix that contains the kernels $\mathcal{K}^u, \mathcal{K}^y$. Therefore, we expect that this method, which **we will call it the LPV-RKHS-PBSID$_{opt}$**, will partially face some numerical problems due to this inversion, but it will be able to estimate well the LPV equivalent Markov parameters even for large past window values, for which $m^t > N$, as long as (8-37) is satisfied.

Summary of the proposed algorithm - The LPV-RKHS-PBSID$_{opt}$ algorithm

Here we will give a summary of the proposed algorithm, following the discussion above.

Algorithm 8.2. LPV-RKHS-PBSID$_{opt}$

1. Create the matrix Y and the kernel $\mathcal{K}$ following (3-12) and (8-22).
2. Determine the value of the hyperparameters that minimize the value of (8-24).
3. Estimate the LPV equivalent Markov parameters using (8-38).
4. Proceed as it is explained in sections 4.2 and 4.3 of [30] to estimate the state sequence and then the unknown matrices of the model (3-4)-(3-5) (see also Appendix B).

Chapter 9

Simulations for the LPV case

In this chapter we will investigate the characteristics of the proposed algorithms by identifying various LPV models. By examining different identification setups, it is our intention to reveal the advantages and deficiencies of the proposed algorithms. The algorithms that will be investigated are the following.

1. Diagonal kernel + LPV-PBSID$_{opt}$ (LPV-Diag), presented in Section 8-3
2. LPV-K&PBSID$_{opt}$, presented in Section 8-3
3. LPV-RKHS-PBSID$_{opt}$, presented in Section 8-3
4. Ridge regularization with noise estimate + LPV-PBSID$_{opt}$ (NPridge-LPV)
5. Classical PBSID$_{opt}$ (no regularization) (LPV-PBSID$_{opt}$), presented in [30]
6. GCV ridge regularization + PBSID$_{opt}$ (GCV+LPV-PBSID$_{opt}$), presented in [30, 50]
7. Optimal regularization + LPV-PBSID$_{opt}$ (LPV MSEopt), presented in Section 8-3
8. True VARX coefficients + LPV-PBSID$_{opt}$ (Opt)

At this point some remarks are useful. The "Opt" algorithm is based on the direct use of the LPV-equivalent Markov parameters that correspond to the first p impulse instants. In practice, we saw that for the chosen simulation examples the "Opt" methods and the "LPV MSEopt" method deliver almost identical results (see also (5-16)), so we will only give the results for the "LPV MSEopt". For the methods "LPV-K&PBSID$_{opt}$" and " LPV-RKHS-PBSID$_{opt}$" we used the **Approach 3** (unless otherwise stated) for the parametrization kernel, which is summarized in Table 8-1. The "NPridge-LPV" method is assigning an identity prior to the LPV equivalent Markov parameters. In that sense, it could be seen as a special case of the LPV-Diag method. The "GCV-LPV-PBSID$_{opt}$" algorithm, discussed in [50] is one of the most well known techniques to perform SID of LPV systems, especially when the

number of coefficients to be estimated is larger than the available data points. Finally, we note that other kernel structures have been investigated (such as a block diagonal kernel, based on Figure 8-3) but the results were rather poor, as it was indeed expected following the discussion in Section 8-3. For this reason, we will not present any other kernel structures of this type except for the diagonal one.

Concerning the simulations, we performed 50 Monte Carlo simulations for each setting (except if otherwise stated), using a fresh noise and input sequence in each one of them. Following the remarks in Chapter 6 (referring to the LTI kernel-based identification), we define a specific SNR_{mod} value for each experiment, given in (6-2).

Moreover, we pre-estimated the noise variance σ, using the same approach as in the LTI case, that is to say, we estimated first the LPV equivalent Markov parameters using the LPV equivalent VARX formulation step, without any regularization and then we employed (5-19). However, in the LPV case it is much possible to end up with an under-determined Least Squares (LS) problem due to the curse of dimensionality, as it was discussed in the previous chapter and in Chapter 3. In this case, the pre-estimation of the noise will be inaccurate, as we experienced by investigating multiple cases [1]. For this reason, we decided to use the pre-estimation step, but we always treated it as an additional hyperparameters, while the pre-estimated value was used as the **initialization point** in the optimization routine. Nonetheless, if someone would prefer to avoid this increase in the complexity of the algorithm, another approach may be possible. More specifically, the maximum past window value could be used such that the VARX LS problem is not under-determined. With this approach the estimated σ will be as close as it can be to the actual noise variance. However, this does not guarantee that the estimation will be accurate, since this method is mainly empirical and so there does not exist a mathematical tool to explain the relation between the chosen past window value and the accuracy of the estimated σ.

9-1 A 2nd order SISO LPV system, described by 2 local systems

Simulations results for a well excited scheduling parameter sequence

In this example we consider a second order LPV model, which is a modified version of a model for the flapping dynamics of a wind turbine, also used as an example in [81]. The system

[1] This is actually the typical effect of over-parametrization, thus leading to the over-fitting problem. In other words, the model contains too many free variables to be estimated and so they adopt the noise characteristics. This is the reason why in these cases the estimated noise variance is much lower than the actual one. If we use, though, cross-validation to evaluate the accuracy of the estimated model the results will reveal the inaccuracy of the estimated model.

matrices of the data-generating LPV state-space model are given by

$$\begin{aligned}
\left[\begin{array}{c|c} A^{(1)} & A^{(2)} \end{array}\right] &= \left[\begin{array}{cc|cc} 0 & 0.0734 & -0.0021 & 0 \\ -6.5229 & -0.4997 & -0.0138 & 0.5196 \end{array}\right], \\
\left[\begin{array}{c|c} B^{(1)} & B^{(2)} \end{array}\right] &= \left[\begin{array}{c|c} -0.7221 & 0 \\ -9.6277 & 0 \end{array}\right], \\
\left[\begin{array}{c} C \end{array}\right] &= \left[\begin{array}{cc} 1 & 0 \end{array}\right], \\
\left[\begin{array}{c} D \end{array}\right] &= \left[\begin{array}{c} 0 \end{array}\right], \\
\left[\begin{array}{c|c} K^{(1)} & K^{(2)} \end{array}\right] &= \left[\begin{array}{c|c} 1 & 0 \\ 0 & 1 \end{array}\right].
\end{aligned}$$

The LPV system is excited by a white noise input signal u_k with $\mathbb{E}[u_k] = 0$ and $var(u_k) = 1$. The scheduling parameter is given by $\mu_k = \left[\begin{array}{cc} 1, & \cos(\frac{2\pi}{10}k) + 0.2 + 0.1v_k \end{array}\right]^T$, where v_k is a zero mean white noise sequence with auto-variance equal to one.

In this example we simulated the system using two different SNR_{mod} values, namely 1 and 10. The data length was chosen to be $N = 400$, while the input signal is a zero mean white noise sequence with variance 1. The past window f was kept constant, $f = 3$ in all the experiments. In Figure 9-1 we present the results for methods 1-7 with $\text{SNR}_{\text{mod}} = 1$, while in Figure 9-2 we present the VAF results for $\text{SNR}_{\text{mod}} = 10$ for the past window values $p = \{4, 5, 6, 7, 8\}$.

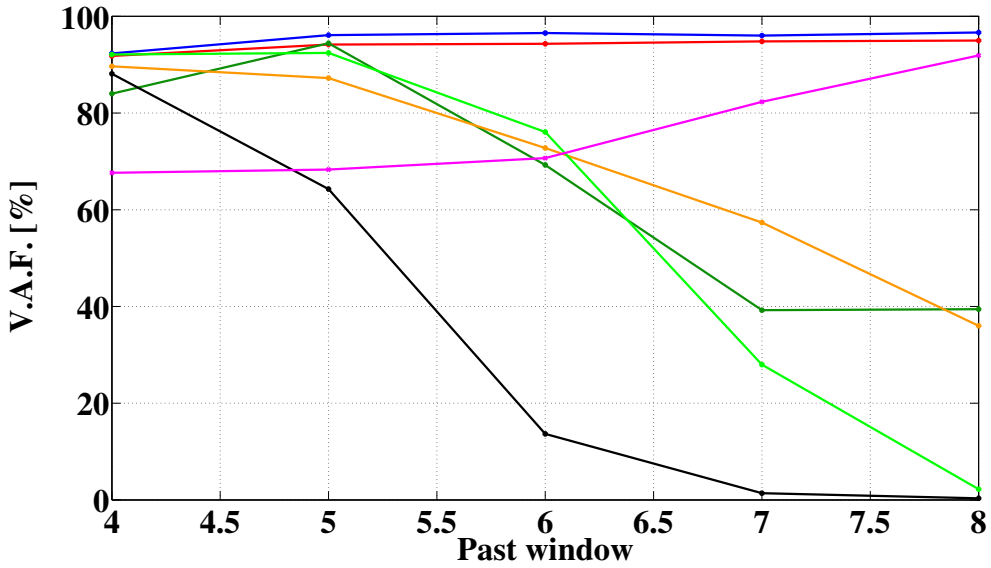


Figure 9-1: VAF results for the methods 1-6. The SNR_{mod} is 1. The blue line corresponds to "LPV-K&PBSID$_{opt}$", the red one to LPV-RKHS-PBSID$_{opt}$, the dark green to "LPV-Diag", the orange one to "NPridge-LPV", the black one to the "PBSID$_{opt}$", the light green to the "GCV+LPV-PBSID$_{opt}$" and the magenta one to the "LPV MSEopt".

For both SNR cases it is evident that the RBF based approaches (LPV-K&PBSID$_{opt}$ and LPV-RKHS-PBSID$_{opt}$) exhibit much better results. Moreover, it is important to notice that the LPV-MSEopt is not providing an upper limit for the VAF results, since the past window

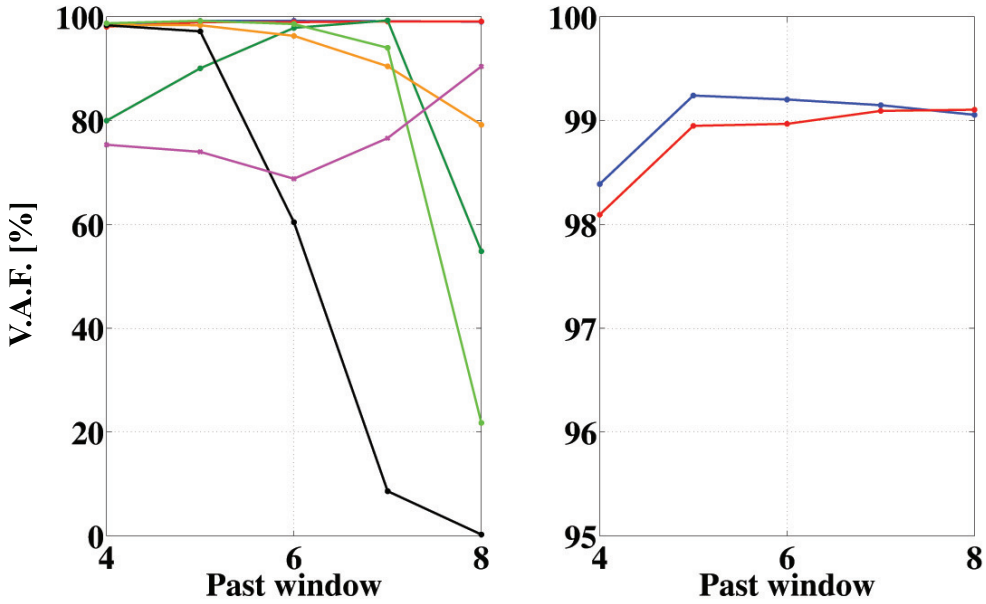


Figure 9-2: VAF results for the methods 1-6. The SNR_{mod} is 10. Left plot: the blue line corresponds to "LPV-K&PBSID$_{opt}$", the red one to LPV-RKHS-PBSID$_{opt}$, the dark green to "LPV-Diag", the orange one to "NPridge-LPV", the black one to the "PBSID$_{opt}$", the light green to the "GCV+LPV-PBSID$_{opt}$" and the magenta one to the "LPV MSEopt". Right plot: Comparison of "LPV-K&PBSID$_{opt}$" with "LPV-RKHS-PBSID$_{opt}$".

value is small and so the inherent assumption of Lemma 8.2 is not satisfied. Moreover, as it is expected, the GCV+LPV-PBSID$_{opt}$ algorithm offers better results compared to the standard LPV-PBSID$_{opt}$, but both fail to compete the RBF based approaches. Finally the NPridge-LPV method is delivering satisfying results only for the smallest past window $p = 4$. Following the remark in Chapter 6, this is an expected result, since this method is taking advantage of the pre-estimation of the noise variance σ, but it fails to deliver good results for large past window values because it assigns the same prior in all the impulse response coefficients, despite the fact that it is expected that their value will decay exponentially for a exponentially stable system. So, the larger the past window is, the more restrictive this assumption becomes.

Additionally, we see that the LPV-K&PBSID$_{opt}$ is delivering in general slightly better results, compared to LPV-RKHS-PBSID$_{opt}$. It is reasonable to assume that this result is directly related to the computational aspects of these two methods and specifically to the inversion of the $\mathcal{K}_i^u, \mathcal{K}_i^y$, $i \in [1, \ldots, p]$ kernels, as shown also in (8-38). Nonetheless, it is observed in Figure 9-2 that the VAF results for the LPV-RKHS-PBSID$_{opt}$ algorithm surpass LPV-K&PBSID$_{opt}$ when the past window value is large ($p = 8$). This is mainly related to a pitfall of the LPV-K&PBSID$_{opt}$, namely the derivation of the LPV equivalent Markov parameters from the impulse response coefficients, shown in (8-27). For $p = 8$ and for a μ dependent matrix, such as K in this example, the matrix $\mathcal{M}_8$ in (8-34) will be of dimensions $2^8 \times 392 = 256 \times 392$, so it may be possible that it will be ill-conditioned. For example, this would be definitely the case for $\mathcal{M}_9$. It is nonetheless important to clarify that even in the latter case the results will not be much worse because the algorithm will still be able to uniquely determine the

Markov parameters that correspond to the first 8 impulse response instants, since these can still be determined in a unique way (more discussion about this topic will follow in the next example.).

Another important remark is related to the scheduling parameter. In this example the μ sequence is a sinusoid, corrupted by additive noise. The latter characteristic is in fact very important for the RBF based methods. For both methods the good excitation of the scheduling parameter is important. As Rasmussen put it, "*... supervised learning algorithms are based on the idea that similar input patterns will usually give rise to similar outputs*". Consequently, the rich excitation of μ delivers the information needed for the correct estimation of the hyperparameters and the subsequent estimation of the impulse response coefficients and the LPV equivalent Markov parameters, a condition which is also reflected by the rank of the kernels in (8-18). However, for the LPV-K&PBSID$_{opt}$ algorithm, the good excitation of the scheduling parameters is necessary for an additional reason, namely the correct derivation of the LPV equivalent Markov parameters in (8-27). In total, we see that the LPV-K&PBSID$_{opt}$ algorithm requires more strict assumptions, as it was also discussed in Section 8-3.

Finally, a different point of view can be given by investigating the accuracy in the estimation of the pole locations. By comparing the best novel kernel based method, namely LPV-K&PBSID$_{opt}$, with the best method from the existing ones, namely GCV+LPV-PBSID$_{opt}$ for $p = 7$, $\text{SNR}_{\text{mod}} = 1$ and $N = 400$, we can indeed verify in Figure 9-3 that LPV-K&PBSID$_{opt}$ delivers much better estimates.

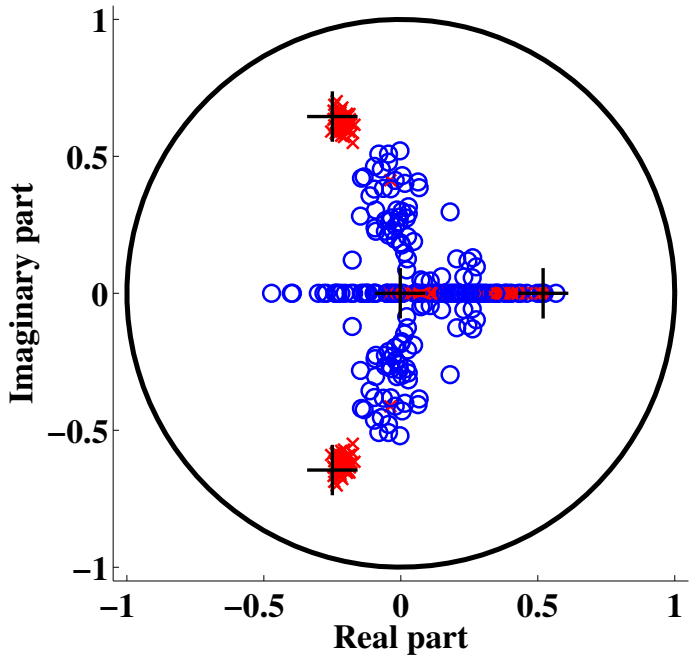


Figure 9-3: Pole location for the LPV-K&PBSID$_{opt}$ (red crosses) and GCV+LPV-PBSID$_{opt}$ (blue circles) for a past window $p = 7$ and $\text{SNR}_{\text{mod}} = 1$

Simulations results for a sinusoidal scheduling parameter sequence

Following the remarks of the previous section, it will be insightful to investigate the accuracy of the proposed methods for the case of a periodic μ sequence. For this purpose, we simulated the same system with exactly the same configurations, except for the fact that we removed the white noise part in the μ parameter and we chose an $\text{SNR}_{\text{mod}} = 1$.

In this case the RBF kernels will definitely be rank deficient, due to the periodicity of their input signal, which is the scheduling parameter. This subsequently entails that the μ data is not "rich-enough" and so we expect that this will be reflected in the accuracy of the RBF based methods. Indeed, this can be deduced from Figure 9-4.

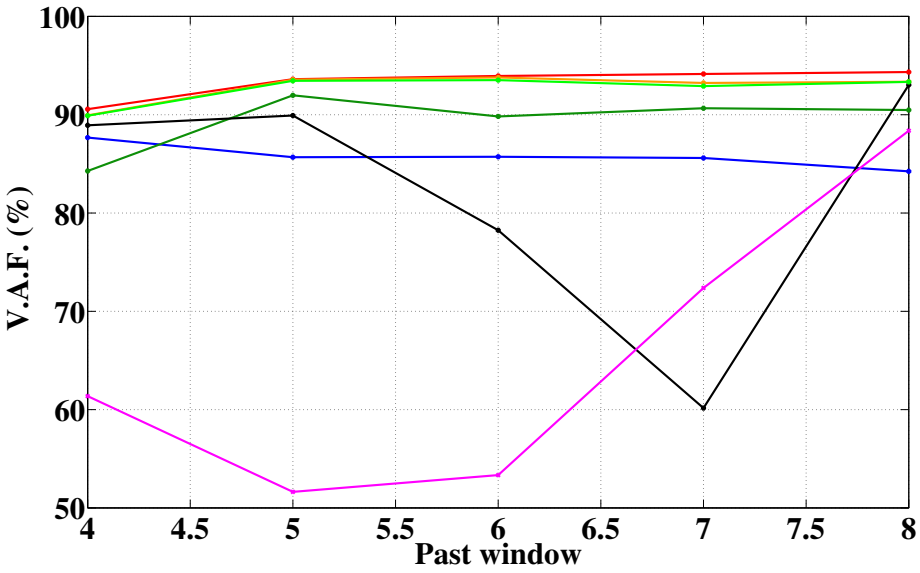


Figure 9-4: VAF results for the methods 1-6. The SNR_{mod} is 10. Left plot: the blue line corresponds to "LPV-K&PBSID$_{opt}$", the red one to LPV-RKHS-PBSID$_{opt}$, the dark green to "LPV-Diag", the orange one to "NPridge-LPV", the black one to the "PBSID$_{opt}$", the light green to the "GCV+LPV-PBSID$_{opt}$" and the magenta one to the "LPV MSEopt". Right plot: Comparison of "LPV-K&PBSID$_{opt}$" with "LPV-RKHS-PBSID$_{opt}$".

As far as the LPV-RKHS-PBSID$_{opt}$ method is concerned, Figure 9-4 reveals that the drop in its performance is not critical, namely around 0.5-1% less than the performance observed in Figure 9-1 and it is definitely showing the best performance among the examined methods. On the other hand, LPV-K&PBSID$_{opt}$ method is showing much worse results when μ is periodic. This is clearly due to rank conditions that are not fulfilled in the estimation step of the LPV equivalent Markov parameters from the impulse response coefficients, evaluated at the training points. In this case, the periodicity of μ leads to row rank deficiency many of the $\mathcal{M}_t$ matrices and so it leads to underdetermined LS problems (see also (8-28)). Finally, it is interesting to notice that the NPridge-LPV method is able to identify highly accurate models, while the expected drop in the performance as the past window value increases is still observed, but it is not as severe as shown in (9-1).

In total, the comparison of the results in Figure 9-4 with the ones in Figure 9-1 demonstrates in a clear way the main pitfall of most of the SID methods, namely the high dependency

on the numerical aspects of the algorithm. The many different involved signals, the large data matrices and the usually large number of coefficients to be estimated form an exquisite problem, in which a slight change in the configuration setting (as it is the removal of the white noise part in μ in this example) may lead to steep changes in the results.

Comparison of different kernel structures

In the beginning of this chapter we clarified that we use the Approach 3, given in Table 8-1, as the kernel structure for the RBF based methods. In this section we would like to evaluate the accuracy of Approach 4, compared to Approach 3. In general, Approach 3 has shown its worth in this example, while it was often observed that Approaches 1 and 2 suffer from local minima and so they were not further investigated.

We used as a past window the value $p = 8$ and an $\text{SNR}_{\text{mod}} = 1$, while the rest of the settings are the same as in Section 9-1. The noise variance σ^2 was treated as a hyperparameter. The results as given in Table 9-1.

Table 9-1: Average VAF results for Approach 3 and 4 in the RBF kernel, where σ is treated as a hyperparameter

Past Window p = 6		
	K&PBSID$_{opt}$	LPV-RKHS-PBSID$_{opt}$
Approach 3	90.30 %	88.70 %
Approach 4	54.15 %	54.77 %

As we can see, Approach 3 is delivering much more accurate models. The result is directly related to the fact that Approach 3 contains more hyperparameters and so the non-convex optimization problem has a bigger flexibility. In general, though, the balance between the flexibility of the kernel and the avoidance of local minima is far from trivial due to the nature of the optimization problem.

Estimation of singular values

It is expected that the accurate estimation of the VARX coefficients will be reflected on the gap of the singular values of the extended observability times controllability matrix (B-4). This was also observed in the LTI case. In order to investigate this, we used a data length of $N = 800$, the SNR_{mod} was set to 10 and the past window was set equal to the future window, $f = p = 6$. In Figure 9-5 we plotted the average singular values (using 10 Monte Carlo simulations) of this matrix for the algorithms LPV-RKHS-PBSID$_{opt}$, LPV-PBSID$_{opt}$ and GCV+LPV-PBSID$_{opt}$.

As we can see, the distance between the first two singular values and the rest of the singular values is obviously higher for the LPV-RKHS-PBSID$_{opt}$, while the singular values of the LPV-PBSID$_{opt}$ algorithm do not show at all any visible gap, thus making the estimation of the order of the system really difficult. This advantage of the kernel based methods will turn out to be extremely useful in real life applications, where the order of the system is usually not known.

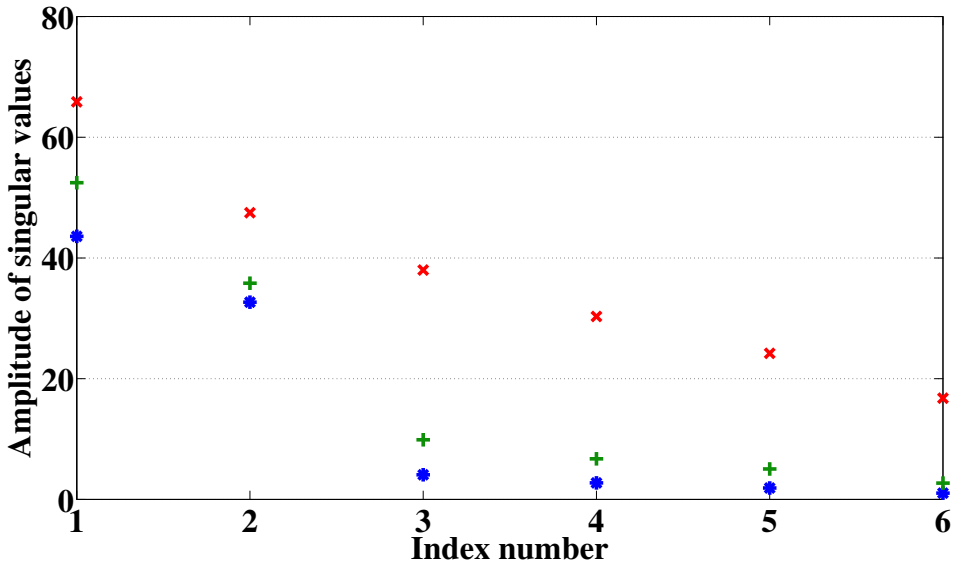


Figure 9-5: Singular values of the extended observability times controllability matrix. The blue stars correspond to LPV-RKHS-PBSID$_{opt}$, the dark green crosses "GCV+LPV-PBSID$_{opt}$" and the red ones to the "LPV-PBSID$_{opt}$"

9-2 A 3rd order SIMO LPV system, described by 2 local systems

In this example we investigated the accuracy of the proposed algorithms in a more complex system, characterized by 2 outputs and one input. The related system matrices are given by

$$
\begin{aligned}
\left[\begin{array}{c|c} A^{(1)} & A^{(2)} \end{array}\right] &= \left[\begin{array}{ccc|ccc} 0 & 0.9 & 0.2 & 0.6 & 0.5 & 0.5 \\ -0.9 & 0.5 & 0 & 0.5 & 0.6 & 0 \\ -0.2 & 0 & 0.2 & -0.5 & 0 & 0.6 \end{array}\right], \\
\left[\begin{array}{c|c} B^{(1)} & B^{(2)} \end{array}\right] &= \left[\begin{array}{c|c} 1 & 0.4 \\ 1 & 0.2 \\ 1 & 0.12 \end{array}\right], \\
\left[\begin{array}{c} C \end{array}\right] &= \left[\begin{array}{ccc} 0.2 & 1 & 0.5 \\ 0.2 & 0.1 & 1 \end{array}\right], \\
\left[\begin{array}{c} D \end{array}\right] &= \left[\begin{array}{c} 0 \\ 0 \end{array}\right], \\
\left[\begin{array}{c|c} K^{(1)} & K^{(2)} \end{array}\right] &= \left[\begin{array}{cc|cc} 0.013 & 0.0225 & 0 & 0 \\ 0.0089 & 0.006 & 0 & 0 \\ 0.0002 & -0.001 & 0 & 0 \end{array}\right].
\end{aligned}
$$

The LPV system is excited by a white noise input signal u_k with $\mathbb{E}[u_k] = 0$ and $var(u_k) = 1$. The scheduling parameter is given by $\mu_k = \left[\begin{array}{cc} 1, & 0.1v_k \end{array}\right]^T$, where v_k is a zero mean white noise sequence with auto-variance equal to one. We used a future window $f = 4$ and the data length was chosen to be $N = 400$. This model is an adaptation of the example in [86, p.99]. We

evaluated the accuracy of the proposed algorithms for different past window values, namely for $p = [4, \ldots, 10]$. The corresponding VAF results versus the past window values are given in Figure 9-6 for the case where $\text{SNR}_{\text{mod}} = 1$ and in Figure 9-7 for the case where $\text{SNR}_{\text{mod}} = 10$.

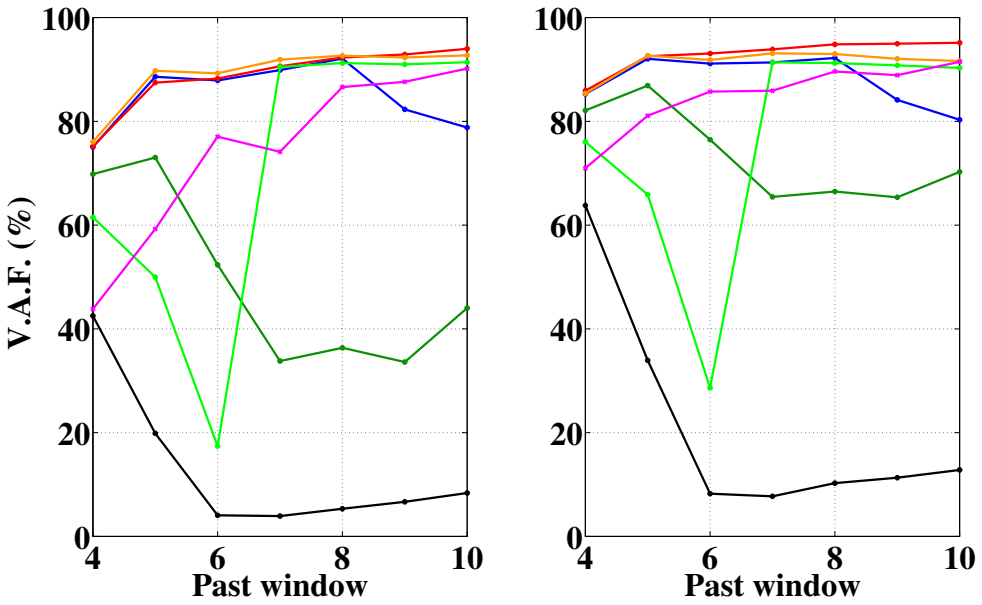


Figure 9-6: VAF results for the methods 1-6. The SNR_{mod} is 1. Left plot: VAF results for the first output. The blue line corresponds to "LPV-K&PBSID$_{opt}$", the red one to LPV-RKHS-PBSID$_{opt}$, the dark green to "LPV-Diag", the orange one to "NPridge-LPV", the black one to the "PBSID$_{opt}$", the light green to the "GCV+LPV-PBSID$_{opt}$" and the magenta one to the "LPV MSEopt". Right plot: VAF results for the second output (same colours used).

In general, it is expected that for a well excited system (e.g. with high SNR value) the merits and pitfalls of the kernel based methods will not be easily distinguishable. For example, this is the case for LPV-K&PBSID$_{opt}$, which shows a high accuracy when $\text{SNR}_{\text{mod}} = 10$ but in the case where $\text{SNR}_{\text{mod}} = 1$ the disadvantages of this method are clearly observed, especially as the past window value increases. However, since a non-convex problem is introduced in the kernel based methods, this relation will not be always visible due to the complexity of the algorithm. For example, in the previous example we observed the same drop of performance for LPV-K&PBSID$_{opt}$ in the case of $\text{SNR}_{\text{mod}} = 10$ and not in the case of $\text{SNR}_{\text{mod}} = 1$. In any case, it is clear that the LPV-K&PBSID$_{opt}$ faces more difficulties in delivering highly accurate models. The same analysis holds for the NPridge-LPV and the GCV+LPV-PBSID$_{opt}$ methods. For the case of $\text{SNR}_{\text{mod}} = 10$ their performance is very close to the LPV-RKHS-PBSID$_{opt}$ algorithm (but still slightly worse than the latter), but we have to notice that the GCV+LPV-PBSID$_{opt}$ algorithm is inconsistent with respect to the past window value, since it fails to deliver accurate models for $p < 7$. However, both methods show a much lower performance in VAF terms when $\text{SNR}_{\text{mod}} = 1$.

Finally, another point of view can again be offered by looking at the estimated eigenvalues. In Figure 9-8 we compared the estimated eigenvalues for $p = 10$ and $\text{SNR}_{\text{mod}} = 10$ for the methods LPV-RKHS-PBSID$_{opt}$ and GCV+LPV-PBSID$_{opt}$.

Following the results in the previous example, it is observed that the higher accuracy of the

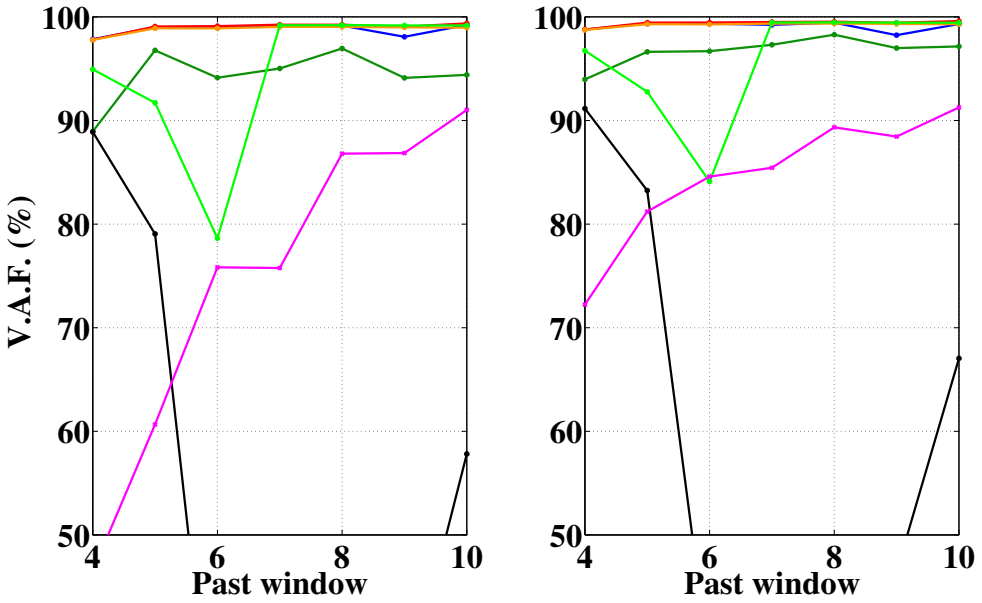


Figure 9-7: VAF results for the methods 1-6. The SNR_{mod} is 10. Left plot: VAF results for the first output. The blue line corresponds to "LPV-K&PBSID$_{opt}$", the red one to LPV-RKHS-PBSID$_{opt}$, the dark green to "LPV-Diag", the orange one to "NPridge-LPV", the black one to the "PBSID$_{opt}$", the light green to the "GCV+LPV-PBSID$_{opt}$" and the magenta one to the "LPV MSEopt". Right plot: VAF results for the second output (same colours used).

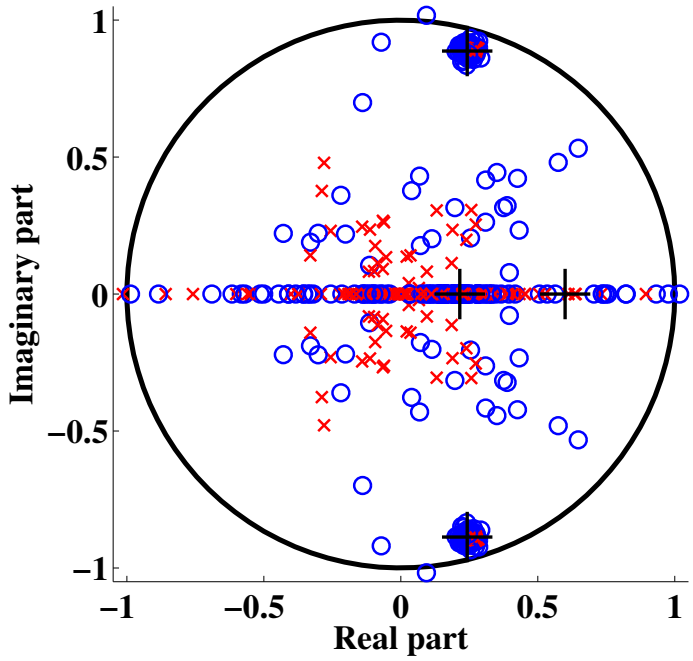


Figure 9-8: Pole location for the LPV-RKHS-PBSID$_{opt}$ (red crosses) and GCV+LPV-PBSID$_{opt}$ for a past window $p = 10$ and $\text{SNR}_{\text{mod}} = 10$

LPV-RKHS-PBSID$_{opt}$ in terms of VAF values is also reflected in the estimation of the poles of the systems. Nonetheless, the higher complexity of this system is clearly seen in the pole estimation. In reality the large number of parameters to be estimated means that there can still be a combination of A, B, C, K matrices that may not lead to an accurate estimation of the poles but still it can estimate the underlying system with high accuracy.

9-2-1 Comparison of different kernel structures

In order to validate the results concerning the Approaches 3 and 4 for the structure of the RBF, we repeated the simulations using these two approaches. The parameters were similar to the ones described above, while the SNR_{mod} was chosen to be 1. The results are given in Tables 9-2 and 9-3.

Table 9-2: Average VAF results for Approach 3 and 4 in the RBF kernel, output #1

Past Window p = 6		
	K&PBSID$_{opt}$	LPV-RKHS-PBSID$_{opt}$
Approach 3	92.04 %	93.21 %
Approach 4	93.11 %	93.57 %

Table 9-3: Average VAF results for Approach 3 and 4 in the RBF kernel, output #2

Past Window p = 6		
	K&PBSID$_{opt}$	LPV-RKHS-PBSID$_{opt}$
Approach 3	94.13 %	95.53 %
Approach 4	94.81 %	96.03 %

As we can see in this case, Approach 4 is able to identify the underlying model with high accuracy and actually it is about 0.5% more accurate than Approach 3. In general, Approach 4 is more attractive due to the fact that it requires less hyperparameters. In general, though, the example in Section 9-1 demonstrated in clear way that this is not always the case and in fact Approach 4 can lead to much worse results. Therefore, it is recommended to resort to Approach 4 only in cases where Approach 3 fails to deliver accurate models. This could be the case, for example, in some systems or some datasets, in which the local minima in the non-convex optimization routine create serious problems.

9-3 A 4th order MISO LPV system, described by 3 local systems

In order to highlight all the aspects of the proposed algorithms we proceeded to the identification of a 2-inputs 1-output LPV system, described by 3 local systems. The system matrices are described by

$$
\begin{aligned}
\left[\begin{array}{c} A^{(1)} \end{array}\right] &= \left[\begin{array}{cccc} -1.3 & -0.6325 & -0.1115 & 0.0596 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{array}\right], \\
\left[\begin{array}{c} A^{(2)} \end{array}\right] &= \left[\begin{array}{cccc} -0.51 & -0.1075 & -0.007275 & -6.25 \cdot 10^{-5} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{array}\right], \\
\left[\begin{array}{c} A^{(3)} \end{array}\right] &= \left[\begin{array}{cccc} 0.2 & 0 & 0 & 0 \\ 0 & 0.4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right], \\
\left[\begin{array}{c|c|c} B^{(1)} & B^{(2)} & B^{(3)} \end{array}\right] &= \left[\begin{array}{cc|cc|cc} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0.3 & 0.3 & 0.3 & 0.3 \end{array}\right], \\
\left[\begin{array}{c} C \end{array}\right] &= \left[\begin{array}{cccc} 1 & 0 & 0 & 0 \end{array}\right], \\
\left[\begin{array}{c} D \end{array}\right] &= \left[\begin{array}{cc} 0 & 0 \end{array}\right], \\
\left[\begin{array}{c} K \end{array}\right] &= \left[\begin{array}{c} 0.16 \\ 0.1 \\ 0.2 \\ 0.16 \end{array}\right].
\end{aligned}
$$

This system is a slight adaptation of the Example 20 in the PBSID toolbox [73]. The main attribute of this system, compared to the two previous ones, is the 3 local systems. We again simulated the system for different past window values. In order to highlight another characteristic of the kernel based methods, we used a relatively small number of data points, $N = 2000$. The future window was chosen to be $f = 5$ and the past window values $p = [5, 6, 7]$. The results are given in Figure 9-9.

First of all, we observe that for the specific system, the LPV-Diag kernel is showing a relatively high accuracy in VAF terms. The most important remark here, though, is related to the RBF based kernels. More specifically, we see that their accuracy is relatively low and it is **dropping** as the past window increases. For the explanation of the results for LPV-RKHS-PBSID$_{opt}$ and LPV-K&PBSID$_{opt}$ we have to resort to the Gaussian process theory and especially in the RBF characteristics. As it is well known from the related theory [5], the RBF kernel is a non-degenerate one. This practically means that the corresponding Gaussian process can be expressed using an infinite dimensional basis. In practice, though, the number of the basis coefficients is limited by the number of the available data points, which in this case is $N - p = 2000 - p$, depending on the specific choice of the past window.

The RBF kernels, as they were defined in (8-18), take as input arguments the μ coefficients. It is evident that the number of inputs is mt, where t corresponds to the impulse response instant, $t \in [1, p]$. Following a weight-space view [5, p.8] we could view the impulse response

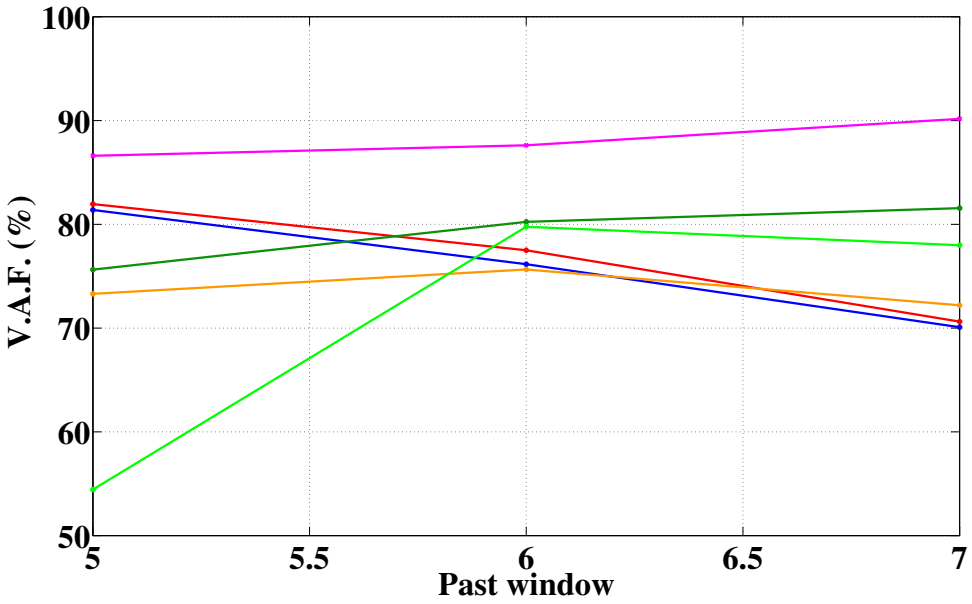


Figure 9-9: VAF results for the methods 1-6. The SNR_{mod} is 10. The blue line corresponds to "LPV-K&PBSID$_{opt}$", the red one to "LPV-RKHS-PBSID$_{opt}$", the dark green to "LPV-Diag", the orange one to "NPridge-LPV", the black one to the "PBSID$_{opt}$", the light green to the "GCV+LPV-PBSID$_{opt}$" and the magenta one to the "LPV MSEopt".

.

functions as mappings $h^u(\mu_{p+i-t}, \ldots, \mu_{p+i-1}; t) : \mathbb{R}^{mt} \rightarrow \mathbb{R}^N$ (similarly for the outputs), that is to say, functions that project the input space to a feature space. This point of view gives us the most appropriate perspective to understand the limitations of the RBF kernels. If the number of m is large, then the number of the available data points should also be large in order to make this mapping informative enough to describe the underlying system. Moreover, the same should also hold for p. In other words, as the past window value gets larger, the mapping will become less capable to describe the underlying impulse response instant. However, its effect is not as crucial as the value of m. In fact, if the data length is not too small, the feature spaces that correspond to the first impulse response instants, let's say t', will be informative enough to make accurate estimations and only the impulse response functions for $p \leq t < t'$ will start suffering from this limited expressiveness. Consequently, this analysis implies that in the RBF based methods we always have to pay high attention to the number of available data points, in cases where their results are not the expected ones.

Finally, in order to verify our claims, we compared the accuracy of the two RBF kernels for two different values of available data points, namely 2000 and 3000. The results are given in Table 9-4. From these results it becomes clear that the RBF based methods can indeed deliver accurate models. As we already explained, though, the number of the available data points should always be taken into consideration in cases of inaccurate identification.

Table 9-4: Average VAF results for Approach 3 and 4 in the RBF kernel, where σ is treated as a hyperparameter

Past Window p = 6		
	K&PBSID$_{opt}$	LPV-RKHS-PBSID$_{opt}$
N=2000	76.16 %	77.50 %
N=3000	89.18 %	91.64 %

Chapter 10

Conclusions and future work

10-1 Conclusions on the kernel based LPV SID

In Part III of this thesis we introduced a novel framework for the kernel based Subspace Identification (SID) of Linear Parameter Varying (LPV) systems. Two different paths were followed; in the first one we incorporated a Gaussian prior distribution on the LPV equivalent Markov parameters, while in the second one we modelled the time-varying impulse response coefficients as Gaussian processes. The first one led to the development of a new approach that makes use of a diagonal kernel (named LPV-Diag in Chapter 9). On the other hand, the second one gave rise to two new methods, namely the LPV-K&PBSID$_{opt}$ and the LPV-RKHS-PBSID$_{opt}$, where both of them used the Radial Basis Function (RBF) kernel to characterise the statistical properties of the related coefficients. Following the theoretical analysis of these methods in Chapter 8 and the simulation examples in Chapter 9, we are now in position to draw some important conclusions.

First of all, it is clear that the LPV-K&PBSID$_{opt}$ and LPV-RKHS-PBSID$_{opt}$ algorithms deliver much more accurate models than the standard approaches, namely the LPV-PBSID$_{opt}$ algorithm and the PBSID$_{opt}$ approach with Generalized Cross-Validation (GCV) ridge regularization, as well as the other investigated methods. The flexibility of the Gaussian processes, together with the sophisticated correlation of the system characteristics with the hyperparameters of the kernels played a major role to this result. Moreover, despite the introduced non-convex optimization for the determination of these hyperparameters, we clearly observed that the derived results do not suffer from the local minima (one should take into account that we avoided any multi-start approach of this optimization step).

Regarding the past window value, it is well known that the LPV-PBSID$_{opt}$ algorithm suffers from the curse of dimensionality, which becomes more severe for larger past window values p. Moreover, it is well known that a small number of data points is expected to reduce the accuracy of the SID algorithms. Nonetheless, when it comes to the RBF kernel based methods (LPV-K&PBSID$_{opt}$ and LPV-RKHS-PBSID$_{opt}$ algorithms), there is one more reason why the number of data points is crucial for the success of the algorithms. As we explained

in Section 9-3, another limitation arises in the RBF kernel based methods in cases where the number of available data points is rather small. In such cases, the functions used to approximate the impulse responses $h^u(\mu_{p+i-t}, \ldots, \mu_{p+i-1}; t)$, where $t = \{1, \ldots, p\}$, will show a reduced accuracy for larger impulse response instants t, due to the small number of basis functions in the feature space, which is limited by the number of available data points. The same consideration should also be taken into account when the number of local systems m is large, as we explained in the same section. Therefore, for systems where m is very large it may be the case that this will have severe effects on the RBF kernel based algorithms. On the other hand, the diagonal kernel presented in Section 8-3 does not suffer from this specific limitation, because its only "input" signal is the impulse response instant $t = [1, \ldots, p]$.

The comparison of the LPV-K&PBSID$_{opt}$ and LPV-RKHS-PBSID$_{opt}$ algorithms is definitely a significant result of this thesis. Following the analysis in Section 8-3, it was observed that in some cases the LPV-K&PBSID$_{opt}$ shows a slightly better performance than the LPV-RKHS-PBSID$_{opt}$ algorithm. This result was mainly attributed to the numerical aspects of the latter one and more specifically to the additional step of the inversion of the kernels. However, the LPV-K&PBSID$_{opt}$ is more dependent on the past window value as well as on the excitation conditions of the scheduling parameter. These remarks were indeed verified mainly in the first two examples of Chapter 9, in which we confirmed that the performance of the LPV-K&PBSID$_{opt}$ algorithm drops in cases where the scheduling parameters are sinusoidal or when the past window value is getting large. Therefore, it is rather fair to qualify the LPV-RKHS-PBSID$_{opt}$ algorithm as the best proposed method, since it makes the least assumptions regarding the scheduling parameter and it is more consistent with respect to the past window value.

The kernel structure is also an important aspect of the RBF based methods. In Table 8-1 we introduced four different approaches for the construction of the kernels that differ on how strong is the correlation that we establish between the impulse response coefficients and the kernel's hyperparameters. As we saw in the examples of Chapter 9, the Approach 3 in Table 8-1 shows the most consistent results. On the other hand, Approach 4 also showed some promising but inconsistent results (as we saw in the first two simulation examples), thus rendering its direct use rather insecure. However, the fact that it shows a much smaller computational burden, compared to Approach 3, implies that it can still be useful in some applications (e.g. when there are time limitations or when Approach 3 is highly vulnerable to suboptimal solutions).

As far as the diagonal kernel is concerned, presented in Section 8-3, we experienced in a clear way the limitations of this approach. By assigning the same prior to all the LPV equivalent Markov parameters that refer to the same impulse response instant, it is obvious that, when the local systems are described by much different dynamics, this kernel will be very restrictive. It is therefore recommended that this method is used only when there is prior knowledge asserting that the local models have very close characteristics. On the other hand, the PBISD$_{opt}$ algorithm with the use of the estimated noise variance σ^2 ("NPridge-PBSID"), although it was not expected to be a superior technique, it still managed to deliver decent models in many cases. This observation, together with the fact that it does not include any non-convex optimization step (in cases where we pre-estimated σ^2 instead of treating it as a hyperparameter) means that it could may be an alternative in cases where the RBF based kernels cannot be used (e.g. time limitations or very few available data points), but this should always be done with consideration.

All in all, it is rather fair to conclude that the LPV-RKHS-PBSID$_{opt}$ exhibits the best performance compared to the other presented approaches. However, this does not comes without any cost. More specifically, in order to enjoy these improved results, we had to sacrifice some interesting characteristics of the standard LPV-PBSID$_{opt}$: the direct treatment of MIMO systems and the avoidance of non-convex optimization routines. Moreover, based on our experience with the code, we have to make clear that the computational time required for the RBF based methods was very high. However, we confidently state that it was a well deserved effort, since we managed to deliver highly accurate models in a variety of identification examples.

10-2 Future work for the kernel based LPV SID

The main purpose of this part of the thesis was to suggest a new framework for the kernel based SID of LPV systems, intrigued by the relevant developments for LTI systems. To this end, two different paths were investigated.

Concerning the first proposed path, investigated in Section 8-3, the case of the diagonal kernel was investigated, due to the complications of a full parametrization approach, as they were discussed throughout this part of the thesis. However, in cases where the local models are characterized by "close" dynamics, then the full parametrization of the kernel may deliver more accurate models compared to the diagonal one. This extension necessarily requires two steps: the justification of the term "close dynamics" through the investigation of different LPV systems and the adaptation of the kernels for LTI systems to the LPV systems, possibly by following a procedure similar to the one performed for the diagonal kernel.

Among the two proposed paths, the methods that introduce a prior in the impulse response coefficients (the LPV-RKHS-PBSID$_{opt}$ and LPV-K&PBSID$_{opt}$ algorithms) were shown to offer superior results compared to the other investigated methods. However, the proposed framework is far from being regarded as "complete". In fact, there are multiple aspects of the proposed algorithms that should be further examined and can be possibly improved.

First of all, the structure of the RBF kernel is still an open question. The balance between a too flexible kernel (that is highly susceptible to local minima, such as Approach 1 in Section 8-3) and a too rigid one (that cannot adapt well to the current model, which was partially the case for Approach 4) has to be further investigated.

More generally, the computational aspects of the proposed approaches require an analytic investigation. Especially for the RBF kernel based approaches (introduced in Section 8-3) it was observed that they show a heavy computational load, which hinders the investigation of multiple simulation paradigms. This is directly related to the complexity of all the involved steps and especially of the non-convex optimization routine (maximization of marginal likelihood). It is characteristic that some investigated identification examples (involving 50 Monte Carlo identification loops) required more than 3 days in 16-core processors of the 3ME computer cluster. It therefore becomes apparent that ways have to be sought such that the computational burden is alleviated, always keeping a balance between optimal solutions and computational time. For example, this investigation can start from the special case where the scheduling parameter is periodic, since there are already proposed methods for the reduction of the computational complexity [30, 81].

At this point it is important to make another remark regarding the RBF kernel. We already saw that the zero mean assumption of the Gaussian processes does not imply that the a posterior estimate will also be zero mean. However, one should pay attention to the fact that one of the inputs of the impulse responses, which are in practice the scheduling parameters, is always one due to the affine assumption on the structure of the state space model. Therefore, the terms which are explicitly related to this input are not "visible" in the kernel because the difference between two of these inputs at any time instants is always zero. These terms are the utmost left elements in each $\mathcal{L}_t^u, \mathcal{L}_t^y$ with $t \in \{1, \dots, p\}$, defined in (3-6) and (3-7). The RBF kernel approaches 3 and 4 of Table 8-1 unfortunately do not have the required flexibility to capture this characteristic. This can be deduced by noticing that for constantly zero scheduling parameters, the auto-variances of all the impulse response instants of a signal are the same, e.g. equal to σ_u^2 for the input signal. In other words, we assign the same prior to the coefficients $C\tilde{A}^{(1)}B^{(1)}$, $C\tilde{A}^{(1)^2}B^{(1)}$ etc. As a solution, we could follow two different paths. One way is by introducing a base-line model, which practically means that we introduce a deterministic term for each past input and output term in order to capture these utmost left elements that we mentioned above. However, in this way we do not introduce any regularization for these elements. As an alternative, we can assign a normal prior distribution on them, too, but this means that they have to be treated as hyperparameters, thus increasing the computational complexity of the marginal likelihood maximization problem.

Moreover, the RBF kernel selection is itself debatable. The examination of other kernel structures can reveal new, improved ways to perform kernel based SID for the LPV systems. One new approach can be followed by implementing automatic relevance determination among the various inputs of the RBF kernels [5, Section 5.1]. In other words, one hyperparameter is assigned to each input (which in our case are the scheduling parameters) to determine the characteristic length scale. In this way, the inputs that have the slightest contribution will be neglected and therefore a more accurate estimation may be achieved. Additionally, the use of polynomial kernels may be an interesting approach, due to the products of the scheduling parameters that arise when we construct the predictor, as shown in (3-8).

The parametrization of the off-block-diagonal terms is also important for this type of kernels. In this thesis we decided to assign a prior only on the elements that refer to the same impulse response instant, as shown in (8-30). However, the optimal kernel is a full matrix, as it was discussed in Section 8-3. Therefore, finding a way to fully parametrize the kernel is expected to lead to better results. This procedure however is not trivial. In fact, by treating the impulse response coefficients as functions defined over the domain $\mathbb{R}^{m^t}$, as shown in (8-18), means that the construction of the RBF kernel for the off-diagonal elements is not feasible in a straightforward way, since these terms correspond to different impulse response instants t and so they correspond to different functions. Consequently, a different approach should be pursued in order to assign a meaningful prior in these terms.

All in all, the proposed algorithms formulate an interesting approach for the SID of LPV systems. The superior results (compared to the other investigated methods) together with the many yet unexplored aspects render them an interesting topic, whose further investigation can offer new insights and many improvements. We hope that this thesis managed to shed light on this new path, where only the first few steps are taken but there are still many new direction to explore.

Appendix A

The PBSID$_{opt}$ algorithm for Linear Time Invariant (LTI) systems

In Chapter 2 we introduced some notation related to the PBSID$_{opt}$ algorithm and we analysed the steps to be taken till the VARX formulation. In this appendix we will enumerate the rest of the steps that have to be taken, following the notation in [23]. To this end, let us first define two useful quantities.

Definition A.1. The extended observability matrix and the extended controllability matrices are given in (A-1) and (A-2), respectively.

$$\Gamma^{(f)} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{f-1} \end{bmatrix} \tag{A-1}$$

$$\mathcal{K}^{(p)} = \begin{bmatrix} A^{p-1}\bar{B} & ... & A\bar{B} & \bar{B} \end{bmatrix}, \tag{A-2}$$

where $\bar{B} = [B\ K]$.

We will also introduce the vector $z_k = \begin{bmatrix} u_k^T & y_k^T \end{bmatrix}^T \in \mathbb{R}^{n_u+n_y}$ and the stacked vector

$$z_k^{(p)} = \begin{bmatrix} z_{k-p}^T & z_{k-p+1}^T & \cdots & z_{k-1}^T \end{bmatrix}^T \tag{A-3}$$

Moreover, we will define the Hankel matrices in analogy with the Toeplitz matrix, defined in (2-8). The Hankel matrix is given by

$$U_{i,s,N}^{H} = \begin{bmatrix} u_i & u_{i+1} & \dots & u_{i+N-1} \\ u_{i+1} & u_{i+2} & \dots & u_{i+N} \\ \vdots & \vdots & \ddots & \vdots \\ u_{i+s-1} & u_{i+s} & \dots & u_{i+N+s-2} \end{bmatrix} \tag{A-4}$$

Given a future window f and a past window p with $p \geq f \geq n$, we can construct the extended observability times controllability matrix $\Gamma^{(f)}\mathcal{K}^{(p)}$.

$$\tilde{\Gamma}^{(f)}\tilde{\mathcal{K}}^{(p)} = \begin{bmatrix} C\tilde{A}^{p-1}\bar{B} & C\tilde{A}^{p-2}\bar{B} & \cdots & \cdots & C\bar{B} \\ C\tilde{A}^{p}\bar{B} & C\tilde{A}^{p-1}\bar{B} & \cdots & \cdots & C\tilde{A}\bar{B} \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ C\tilde{A}^{p+f-2}\bar{B} & C\tilde{A}^{p+f-3}\bar{B} & \cdots & \cdots & C\tilde{A}^{f-1}\bar{B} \end{bmatrix} \tag{A-5}$$

In the PBSID algorithm the matrix is (A-5) is used without any modification. On the other hand, in the PBSID$_{opt}$ algorithm the approximation of $\tilde{A}^{p+i} \approx 0$ for $i \geq 0$ is used, while it is also shown in [22] that the PBSID$_{opt}$ algorithm shows lower variance than PBSID. The latter matrix quantity can be totally constructed with the use of the Markov parameters that were identified in the VARX step and it is given by (the case where $p = f$ is presented here)

$$\tilde{\Gamma}^{(f)}\tilde{\mathcal{K}}^{(p)} \approx \begin{bmatrix} C\tilde{A}^{p-1}\bar{B} & C\tilde{A}^{p-2}\bar{B} & \cdots & \cdots & C\bar{B} \\ 0 & C\tilde{A}^{p-1}\bar{B} & \cdots & \cdots & C\tilde{A}\bar{B} \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & C\tilde{A}^{f-1}\bar{B} \end{bmatrix} \tag{A-6}$$

Moreover, if we multiply $\tilde{\Gamma}^{(f)}\tilde{\mathcal{K}}^{(p)}$ on the right with $Z_{1,p,N-p}$ it can be shown that the following equivalence holds [23].

$$\tilde{\Gamma}^{(f)}\tilde{\mathcal{K}}^{(p)} Z_{1,p,N-p} = \Gamma^{(f)} X_{p+1,1,N-p} \tag{A-7}$$

Now we have to make some necessary assumptions. We will assume that the pair (A, C) is observable and the pair $\left(A, \begin{bmatrix} B & KW^{1/2} \end{bmatrix}\right)$ is reachable, where $\mathbb{E}\left[e_j e_k^T\right] = W\delta_{jk}$ and δ denotes the kronecker delta. Moreover, we will assume that the state sequence $X_{p+1,1,N-p}$ has full row rank equal to n.

With theses assumptions we can estimate the state sequence with the use of the rank revealing SVD decomposition.

$$\tilde{\Gamma}^{(f)}\tilde{\mathcal{K}}^{(p)} Z_{1,p,N-p} = \begin{bmatrix} U_n & U_n^{\perp} \end{bmatrix} \begin{bmatrix} \Sigma_n & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_n^T \\ V_n^{T\perp} \end{bmatrix} \tag{A-8}$$

Ideally, the quantity Σ_2 is zero but due to the noise this is not the case. However, by detecting a gap in the singular values we can keep only the n most dominant singular values. Consequently, we can estimate the state sequence as $X_{p+1,1,N-p} = \Sigma_n V_n^T$.

Having estimated the state sequence, it is straightforward to estimate the unknown matrices. As a first step, we compute the following Least Squares (LS) problem based on the output equation.

$$||Y_{p+1,1,N-p} - CX_{p+1,1,N-p}||_F^2 \tag{A-9}$$

With this LS problem we estimate the matrix C, while the residual can be used as an estimation of $E_{p+1,1,N-p}$. Finally, we can estimate the matrices A, B and K by solving the LS problem

$$||X_{p+2,1,N-p-1} - AX_{p+1,1,N-p-1} - BU_{p+1,1,N-p-1} - KE_{p+1,1,N-p-1}||_F^2. \tag{A-10}$$

Appendix B

Subspace Identification (SID) of LPV systems: the LPV-PBSID$_{opt}$ algorithm

In Chapter 3 we described the first steps of the LPV-PBSID$_{opt}$ algorithm, up to the LPV equivalent VARX formulation step. The related quantities (such as Z, $\mathcal{K}^{(p)}$) were defined in a way that enables the direct incorporation of the kernel methods, that is to say, the coefficients that correspond to each signal were brought together.

In order to keep consistency, thought, with the related literature, we will describe the next steps in the LPV-PBSID$_{opt}$, as it is explained in [30] and also implemented in the PBSID toolbox [52] [1]. In order to do so, we have to redefine the matrices Z and $\mathcal{K}^{(p)}$.

More specifically, the new Z_{new} and $\mathcal{K}^{(p)}_{new}$ are given by

$$Z_{new} = \begin{bmatrix} P_{p|p+1}u_1 & P_{p|p+2}u_2 & \cdots & P_{p|N}u_{N-p} \\ P_{p|p+1}y_1 & P_{p|p+2}y_2 & \cdots & P_{p|N}y_{N-p} \\ P_{p-1|p+1}u_2 & P_{p-1|p+2}u_3 & \cdots & P_{p-1|N}u_{N-p-1} \\ P_{p-1|p+1}y_2 & P_{p-1|p+2}y_3 & \cdots & P_{p-1|N}y_{N-p-1} \\ \vdots & \vdots & \vdots & \vdots \\ P_{1|p+1}u_p & P_{1|p+2}u_{p+1} & \cdots & P_{1|N}u_{N-1} \\ P_{1|p+1}y_p & P_{1|p+2}y_{p+1} & \cdots & P_{1|N}y_{N-1} \end{bmatrix}, \quad Z \in \mathbb{R}^{\tilde{q}\times N}, \tag{B-1}$$

$$\mathcal{K}^{(p)}_{new} = \begin{bmatrix} \mathcal{L}_p \cdots & \mathcal{L}_1 \end{bmatrix}, \tag{B-2}$$

where

$$\begin{aligned} \mathcal{L}_1 &= \begin{bmatrix} B^{(1)}, & B^{(2)}, & \cdots, B^{(m)}, & K^{(1)}, & \cdots & K^{(m)} \end{bmatrix}, \\ \mathcal{L}_j &= \begin{bmatrix} \tilde{A}^{(1)}\mathcal{L}_{j-1} & \cdots & \tilde{A}^{(m)}\mathcal{L}_{j-1} \end{bmatrix}. \end{aligned} \tag{B-3}$$

[1] The only difference between these two approaches is in the construction of the matrix $\mathcal{L}_1$. More specifically, we will adopt the one in the PBSID toolbox, that is to say, we will define $\mathcal{L}_1 = \begin{bmatrix} B^{(1)}, & B^{(2)}, & \ldots, B^{(m)}, & K^{(1)}, & \ldots & K^{(m)} \end{bmatrix}$

The next steps of the algorithm are the following. First, similar to the LTI case, we will construct the extended observability times controllability matrix. In the LPV case, we use the extended observability matrix of the first local system. By keeping in mind the assumption that $\phi_{k,j} \approx 0$ for $j > p$, we derive the matrix (here we show the case where $f = p$)

$$\Gamma^p \mathcal{K}^{(p)}_{new} = \begin{bmatrix} C\mathcal{L}_p & C\mathcal{L}_{p-1} & \cdots & C\mathcal{L}_1 \\ 0 & C\tilde{A}^{(1)}\mathcal{L}_{p-1} & \cdots & C\tilde{A}^{(1)}\mathcal{L}_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & C\left(\tilde{A}^{(1)}\right)^{p-1}\mathcal{L}_1 \end{bmatrix}, \tag{B-4}$$

where the rest $\mathcal{L}_t$, $t = \{2, \ldots, p\}$ are constructed similar to the procedure shown in (3-7). Also, with Γ^p we denote the observability matrix of the first local system, given by

$$\Gamma^p = \begin{bmatrix} C \\ C\tilde{A}^{(1)} \\ \vdots \\ C\left(\tilde{A}^{(1)}\right)^{p-1} \end{bmatrix}, \tag{B-5}$$

The extended observability matrix is assumed to be full column rank. Now, the state sequence is defined as

$$X = \begin{bmatrix} x_{p+1} & x_{p+2} & \cdots & x_N \end{bmatrix}. \tag{B-6}$$

and it is assumed also to be full rank. Based on these assumptions, we can finally estimate the state sequence (up to a similarity transformation) by using an Singular Value Decomposition (SVD) decomposition of (B-4).

$$\Gamma^p \mathcal{K}^{(p)}_{new} Z_{new} = \begin{bmatrix} \mathcal{U}_1 & \mathcal{U}_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}, \tag{B-7}$$

$$X = \Sigma_1 V_1. \tag{B-8}$$

From this point, the estimation of the unknown matrices is straightforward. First, we solve a LS problem to derive the matrix C as well as the noise sequence e_k for $k = \{p+1, \ldots, N\}$ and then the noise is treated as a deterministic signal and it is used in the state equation to compute the matrices $A^{(i)}, B^{(i)}$ and $K^{(i)}$ for $i = \{1, \ldots, m\}$.

The algorithm can now be summarized as follows:

1. Compute the quantities Y and Z based on (3-12) and (3-14) respectively.
2. Solve the LS problem (3-16).
3. Create the matrix (B-4).
4. Estimate the state sequence based on (B-8).
5. Estimate the unknown matrices by solving the LS problems based on (3-4)-(3-5).

Appendix C

Bayesian framework for the kernel based Subspace Identification (SID) methods

C-1 Introduction to the bayesian framework and properties of normally distributed random variables

In Section 4-2 we already saw how we can compute the required distributions for the case where the model is completely treated as non-parametric. The only difference in the kernel based system identification is that the latter contains an additional parametric part due to the past input and output data. Apart from this, the analysis and the computations from Section 4-2 can also be applied in system identification with only a few modifications. In order to show this, let us assume first that we have a data generating model, described by the equation

$$Y = \theta Z + E \tag{C-1}$$

where $Y = [y_1\ y_2\ \dots\ y_N]$ denotes the output, $E = [e_1\ e_2\ \dots\ e_N]$ denotes the noise, $\theta \in \mathbb{R}^p$ are the unknown coefficients to be estimated and Z is a known (deterministic) matrix of dimensions $p \times N$. We also assume that the noise is a zero mean white noise sequence with normal distribution and variance equal to σ^2.

If we treat θ as random variables, we can assign a prior distribution to them.

$$\theta \sim \mathcal{N}(0\ ,\ P) \tag{C-2}$$

Usually we let P be parametrized by some (unknown) hyper-parameters, which encode information about the system. For this reason, we will write the covariance matrix of θ as

$\mathbb{E}\left(\theta^T\theta\right) = P(\eta)$ to make clear that it is a function of the hyper-parameters $\eta \in \mathbb{R}^d$ The normality assumption is not in general necessary, but it is required in order to end up with an analytic expression of the quantities to follow. Due to this assumption, the output signals are themselves normally distributed random variables. Nonetheless, due to the assumption that θ are random variables, the output signals are dependent random variables.

Before we proceed to the description of the Marginal Likelihood (MargLik) and the Maximum a Posteriori (MAP) estimates, we will describe the joint statistical properties of Y and θ.

$$\begin{aligned}[Y,\ \theta]^T &\sim \left(\begin{bmatrix} \mathbb{E}\,(Y)^T \\ \mathbb{E}\,(\theta)^T \end{bmatrix},\ \begin{bmatrix} \mathbb{E}\left(Y^TY\right) & \mathbb{E}\left(Y^T\theta\right) \\ \mathbb{E}\left(\theta^TY\right) & \mathbb{E}\left(\theta^T\theta\right) \end{bmatrix}\right) \\ &\sim \left(\begin{bmatrix} 0_{N\times 1} \\ 0_{p\times 1} \end{bmatrix},\ \begin{bmatrix} Z^TP(\eta)Z+\sigma^2I & Z^TP(\eta) \\ P(\eta)Z & P \end{bmatrix}\right)\end{aligned} \tag{C-3}$$

C-2 Marginal Likelihood and a Posteriori estimates

The marginal likelihood of the output signals is given as [87]

$$p(Y|Z,\eta) = \int p(Y,\theta|Z,\eta)d\theta = \int p(Y|\theta,Z,\eta)p(\theta|Z,\eta)d\theta \tag{C-4}$$

In the Gaussian case, the quantity in (C-4) can be computed analytically, using the following expression.

$$p(Y|Z,\eta) = \frac{1}{(2\pi)^{\frac{N}{2}}\left(\det(Z^TP(\eta)Z+\sigma^2I)\right)^{\frac{1}{2}}}\exp\left(-\frac{1}{2}Y\left(Z^TP(\eta)Z+\sigma^2I\right)^{-1}Y^T\right) \tag{C-5}$$

Maximization of the MargLik (or more often, minimization of the minus logarithm of MargLik) is a common way to estimate the values of unknown quantities such as the hyperparameters in $P(\eta)$. Taking the minus logarithm of (C-5) we find the expression

$$-\log p(Y|Z,\eta) = \frac{N}{2}\log(2\pi) + \frac{1}{2}\log\left(\det\left(Z^TP(\eta)Z+\sigma^2I\right)\right) + \frac{1}{2}Y\left(Z^TP(\eta)Z+\sigma^2I\right)^{-1}Y^T \tag{C-6}$$

Another useful quantity is the so-called posterior estimate. Based on the Bayes rule, the general expression for the posterior distribution is the following (e.g. [5]):

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{marginal likelihood}}. \tag{C-7}$$

Based on the data generating system in (C-1), the quantity in (C-7) can be written as

$$p(\theta|Y,Z,\eta) = \frac{p(Y|\theta,Z,\eta)p(\theta|Z,\eta)}{p(Y|Z,\eta)}. \tag{C-8}$$

If the assumptions about the normal distribution of the random variables hold, then the posterior estimate can be computed in an analytical way. For the model in (C-1), the 1st and 2nd order statistical properties of the posterior estimate of θ are given by [54]

$$\mathbb{E}\left(\theta|Y,\eta,Z\right) = Y\left(Z^T P(\eta) Z + \sigma^2 I\right)^{-1} Z^T P(\eta) = Y Z^T \left(Z Z^T + \sigma^2 P(\eta)^{-1}\right)^{-1} \tag{C-9}$$

$$\mathbb{E}\left(\theta^T\theta|Y,\eta,Z\right) = P(\eta) - P(\eta) Z \left(Z^T P(\eta) Z + \sigma^2 I\right)^{-1} Z^T P(\eta) \tag{C-10}$$

Writing the solution as shown in the second part of (C-9) is in general more convenient, especially when $p >> N$ due to the smaller size of the matrix that has to be inverted. On the other hand, the most right part of this equation is more appealing at cases where $p << N$ but it may lead to numerical problems due to the inversion of the P matrix, so in general it is avoided.

Moreover, it can be directly concluded that this analysis becomes identical with the one presented in Section 4 if we substitute the product θZ with a function, let's say f. In other words, the selection of where a prior distribution will be assigned depends on what we want to achieve. For example, in the kernel based SID methods for Linear Time Invariant (LTI) systems the coefficients θ of (C-1) are the impulse response coefficients, whose domain are the impulse response instants $t \in \mathbb{R}$.

Another useful remark is that, due to the normality assumptions, the mean of the posterior estimate of θ given in (C-8) is also its mode and so the posterior is also the MAP estimate of θ [5]. Finally, it is worth noticing that the solution we derived with this Bayesian framework is identical to the one being derived with the use of the Reproducing Kernel Hilbert Space (RKHS) theory, which is analysed in Appendix D.

Appendix D

Reproducing Kernel Hilbert Spaces and Regularization

D-1 Theory of Reproducing Kernel Hilbert Spaces

In this Appendix we will give an insight into the theory of Reproducing Kernel Hilbert Space (RKHS) and we will show its link with the Tikhonov based regularization. The theory of RKHS was developed by Aronszajn [45] and since then it has been applied in many fields, such as Machine Learning [5].

To introduce this theory, let us first define an index set $\mathcal{X}$ and assume that a set of functions f are defined over that set. We will assume that these functions belong to a Hilbert space $\mathcal{H}$, that is to say, to a complete vector space with an inner product $< f, g >$ and its norm is defined as $||f|| = \sqrt{< f, f >}$.

The function $k(x, y) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is called a Reproducing Kernel Hilbert Space if the following conditions hold:

1. For every y, $k(x, y)$ as a function of x belongs to $\mathcal{H}$.
2. k satisfies the reproducing property: $f(y) =< f(x), k(x, y) >$, where the inner product is defined as a function of x.

As it is proven in [45], the RKHS uniquely determines the function $k(x, y) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ if $k(x, y)$ is **positive semi-definite**. Moreover, the converse property also holds, that is to say, to every positive semi-definite function $k(x, y)$ there corresponds a unique RKHS.

Finally, a necessary and sufficient condition for the RKHS to exist is that the function $f(x)$ is a continuous functional of f in $\mathcal{H}$ for every $x \in \mathcal{X}$.

D-2 Representer theorem and regularization

The representer theorem is very useful tool for the theory of RKHS, since it provides a basis for the functions f.

In order to explain this theorem, let us assume that $f \in \mathbb{R}^m$, where m is finite, then we can express f as follows [5]:

$$f(x) = \sum_{i=1}^{m} \alpha_i k(x, x_i), \quad N \in \mathbb{N}, x_i \in \mathcal{X}, \alpha_i \in \mathbb{R} \tag{D-1}$$

Now, we can define the representer theorem [67, 88]:

Definition D.1. Given a set $\mathcal{X}$, a positive semi-definite kernel $k(x, y) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, the data set $\{(x_1, y_1), \cdots (x_N, y_N)\} \in \mathcal{X} \times \mathbb{R}$ and a strictly monotonically increasing real-valued function g on $[0, \infty)$, an arbitrary cost function $c : (\mathcal{X} \times \mathbb{R}^2)^m \to \mathbb{R} \cup \{\infty\}$ and a class of functions

$$\mathcal{F} = \Big\{ f(z) = \sum_{i=1}^{\infty} \beta_i k(z, x_i), \quad x_i \in \mathcal{X}, \beta_i \in \mathbb{R}, ||f|| < \infty \Big\} \tag{D-2}$$

Then, any $f(z)$ that minimizes a functional

$$J(f) = c\Big((x_1, y_1, f(x_1)), \cdots, (x_N, y_N)\Big) + g\left(||f||\right) \tag{D-3}$$

has an optimal solution which can be expressed in the following form:

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x, x_i), \quad N \in \mathbb{N}, x_i \in \mathcal{X}, \alpha_i \in \mathbb{R} \tag{D-4}$$

This is actually a powerful result. It means that the minimization over a possibly infinite space can be expressed as a minimization problem over $\mathbb{R}^N$. Therefore, every solution in RKHS can be expressed as a function of a finite basis, even when the function $f(x)$ is itself infinite dimensional.

D-3 Correspondence between Tikhonov based regularization and Maximum a Posteriori estimate

RKHS and Tikhonov regularization

In order to make clear why and how the theory of RKHS can be used for the optimal derivation of some unknown estimates, let us consider the functional

$$J(f) = \sum_{i=1}^{N} (y_i - f(x_i)) + \sigma^2 ||f||_{\mathcal{H}}^2, \tag{D-5}$$

where $||\cdot||^2_{\mathcal{H}}$ denotes the regularization in the Hilbert space $\mathcal{H}$.

Using the expression $f(z) = \sum_{i=1}^{N} \alpha_i k(z, x_i)$ and the property 1 of the RKHS, namely $< k(z, x_i), k(z, x_j) >_{\mathcal{H}} = k(x_i, x_j)$, we find that the term $||f||^2_{\mathcal{H}}$ is equal to the following quantity:

$$||f||^2_{\mathcal{H}} = \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_N \end{bmatrix} \begin{bmatrix} k(x_1,x_1) & k(x_1,x_2) & \cdots & k(x_1,x_N) \\ \vdots & \ddots & \ddots & \vdots \\ k(x_N,x_1) & k(x_N,x_2) & \cdots & k(x_N,x_N) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix} = \alpha K \alpha^T \tag{D-6}$$

Based on (D-6), we express (D-5) as follows:

$$J(\alpha) = \sigma^2 \alpha K \alpha^T + ||\mathbf{y} - \alpha K||_2^2 \tag{D-7}$$

where $\mathbf{y} = [y_1,\, y_2 \cdots y_N]^T$. It is obvious that (D-7) corresponds to a Least Squares problem with Tikhonov regularization [26]. Therefore, the solution can be computed analytically and it is given by the following equation:

$$\alpha = \mathbf{y} \left(\sigma^2 I_N + K\right)^{-1} \tag{D-8}$$

Substituting (D-8) to (D-4), we derive the optimal solution for f namely:

$$\begin{aligned} \begin{bmatrix} f(x_1 & f(x_2) & \cdots & f(x_N) \end{bmatrix} &= \alpha K \Rightarrow \\ \begin{bmatrix} f(x_1 & f(x_2) & \cdots & f(x_N) \end{bmatrix} &= \mathbf{y} \left(\sigma^2 I_N + K\right)^{-1} K \end{aligned} \tag{D-9}$$

Maximum a Posteriori estimate

Let us now assume that a system is described by the following equation:

$$y_i = f(x_i) + e_i \tag{D-10}$$

where $e_i \sim \mathcal{N}\left(0, \sigma^2\right)$ and moreover, $\mathbb{E}\left(e_i e_j\right) = \sigma^2 \delta_{i-j}$, where δ represents the kronecker delta.

Let us again assume that we have collected N data points. If we model f as a zero-mean Gaussian Process with $\mathbb{E}\left(f(x_i) f(x_j)\right) = k(x_i, x_j)$, we can compute the mean value of the Maximum a Posteriori estimate of f_i, $\mathbb{E}\left(f_i|\mathbf{y}\right)$, $i \in \{1, \cdots, N\}$. It is straightforward to show that the derived solution coincides with the solution that was derived in the previous section. To do so, let us first compute the mean and covariance of the vector that contains all the $f(x_i)$, y_i:

$$\begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_N) \\ y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \sim \left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} K & K \\ K & K + \sigma^2 I_N \end{bmatrix} \right) \tag{D-11}$$

Using the properties of Gaussian distributed random variables, we can compute the a Posteriori distribution $p\left(f\left(x_i\right) | f\left(x_1\right), \cdots, f\left(x_{i-1}\right), f\left(x_{i+1}\right), f\left(x_N\right), y_1, \cdots, y_N\right)$ (e.g. Appendix A in [5]). In total, we derive the following result:

$$\left.\begin{array}{c} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_N) \end{array}\right| \left.\begin{array}{c} y_1 \\ y_2 \\ \vdots \\ y_N \end{array}\right. \sim \mathcal{N}\left(K(\sigma^2 I_N + K)^{-1}\mathbf{y}^T, K - K(\sigma^2 I_N + K)^{-1}K\right) \tag{D-12}$$

Consequently, by taking the transpose of the mean value in (D-12), we find that the Maximum a Posteriori estimate of f, evaluated at the training points yields the same result as in (D-9).

Bibliography

[1] M. Merriman, "On the history of the method of least squares," *The Analyst*, pp. 33–36, 1877.

[2] M. Lovera, M. Bergamasco, and F. Casella, "LPV modelling and identification: An overview," in *Robust Control and Linear Parameter Varying Approaches*, pp. 3–24, Springer, 2013.

[3] R. Tóth, *Modeling and identification of linear parameter-varying systems*, vol. 403. Springer Science & Business Media, 2010.

[4] T. Freeth, Y. Bitsakis, X. Moussas, J. Seiradakis, A. Tselikas, H. Mangou, M. Zafeiropoulou, R. Hadland, D. Bate, A. Ramsey, *et al.*, "Decoding the ancient greek astronomical calculator known as the antikythera mechanism," *Nature*, vol. 444, no. 7119, pp. 587–591, 2006.

[5] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[6] L. Ljung, H. Hjalmarsson, and H. Ohlsson, "Four encounters with system identification," *European Journal of Control*, vol. 17, no. 5, pp. 449–471, 2011.

[7] G. Pillonetto, A. Chiuso, and G. De Nicolao, "Predictor estimation via gaussian regression," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pp. 744–749, IEEE, 2008.

[8] A. Chiuso, G. Pillonetto, and G. De Nicolao, "Subspace identification using predictor estimation via gaussian regression," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pp. 3299–3304, Dec 2008.

[9] A. Golabi, N. Meskin, R. Tóth, and J. Mohammadpour, "A bayesian approach for estimation of linear-regression LPV models," in *Decision and Control (CDC), 2014 53th IEEE Conference on*, pp. 2555–2560, IEEE, 2014.

[10] L. Ljung, *System identification - Theory for the User*. Prentice-Hall, 1999.

[11] M. Verhaegen and V. Verdult, *Filtering and System Identification. A Least Squares Approach*. Cambridge University Press, 2011.

[12] H. Akaike, "Stochastic theory of minimal realization," *Automatic Control, IEEE Transactions on*, vol. 19, no. 6, pp. 667–674, 1974.

[13] P. V. Overschee and B. D. Moor, "N4sid: Subspace algorithms for the identification of combined deterministic-stochastic systems," *Automatica*, vol. 30, no. 1, pp. 75 – 93, 1994. Special issue on statistical signal processing and control.

[14] M. Verhaegen, "Identification of the deterministic part of mimo state space models given in innovations form from input-output data," *Automatica*, vol. 30, no. 1, pp. 61 – 74, 1994. Special issue on statistical signal processing and control.

[15] W. Larimore, "Canonical variate analysis in identification, filtering, and adaptive control," in *Decision and Control, 1990., Proceedings of the 29th IEEE Conference on*, pp. 596–604 vol.2, Dec 1990.

[16] P. V. Overschee and B. D. Moor, "A unifying theorem for three subspace system identification algorithms," *Automatica*, vol. 31, no. 12, pp. 1853 – 1864, 1995. Trends in System Identification.

[17] P. V. Overschee and B. D. Moor, *Subspace Identification for Linear Systems*. Kluwer Academic Publishers, 1996.

[18] T. Katayama, *Subspace Methods for System Identification*. Springer, 2005.

[19] S. J. Qin and L. Ljung, "Parallel qr implementation of subspace identification with parsimonious models," in *Proceedings of the 13th IFAC SYSID Symposium*, pp. 1631–1636, 2003.

[20] L. Ljung, *System Identification Toolbox for Use with MATLAB*. The MathWorks, Inc., 2007.

[21] T. McKelvey, "Frequency domain system identification with iv based subspace algorithm," 1995.

[22] A. Chiuso, "The role of vector autoregressive modeling in predictor-based subspace identification," *Automatica*, vol. 43, no. 6, pp. 1034–1048, 2007.

[23] G. van der Veen, J.-W. van Wingerden, M. Bergamasco, M. Lovera, and M. Verhaegen, "Closed-loop subspace identification methods: an overview," *IET Control Theory & Applications*, vol. 7, no. 10, pp. 1339–1358, 2013.

[24] L. Ljung and T. McKelvey, "Subspace identification from closed loop data," *Signal Processing*, vol. 52, no. 2, pp. 209–215, 1996.

[25] L. Ljung and B. Wahlberg, "Asymptotic properties of the least-squares method for estimating transfer functions and disturbance spectra," *Advances in Applied Probability*, vol. 24, no. 2, pp. pp. 412–440, 1992.

[26] G. Golub and C. V. Loan, *Matrix Computations, Fourth Edition*. The Johns Hopkins University Press, 2013.

[27] J. Mohammadpour and C. W. Scherer, *Control of linear parameter varying systems with applications*. Springer Science & Business Media, 2012.

[28] D. J. Leith and W. E. Leithead, "Survey of gain-scheduling analysis and design," *International journal of control*, vol. 73, no. 11, pp. 1001–1025, 2000.

[29] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, no. 10, pp. 1401 – 1425, 2000.

[30] J.-W. van Wingerden and M. Verhaegen, "Subspace identification of bilinear and lpv systems for open-and closed-loop data," *Automatica*, vol. 45, no. 2, pp. 372–381, 2009.

[31] A. Bachnas, R. Tóth, J. Ludlage, and A. Mesbah, "A review on data-driven linear parameter-varying modeling approaches: A high-purity distillation column case study," *Journal of Process Control*, vol. 24, no. 4, pp. 272 – 285, 2014.

[32] R. Tóth, F. Felici, P. S. Heuberger, and P. M. Van den Hof, "Discrete time lpv i/o and state space representations, differences of behavior and pitfalls of interpolation," in *Proc. of the European control conf*, pp. 5418–5425, 2007.

[33] R. Tóth, H. Abbas, and H. Werner, "On the state-space realization of LPV input-output models: Practical approaches," *Control Systems Technology, IEEE Transactions on*, vol. 20, no. 1, pp. 139–153, 2012.

[34] R. Tóth, J. C. Willems, P. S. Heuberger, and P. M. Van den Hof, "The behavioral approach to linear parameter-varying systems," *Automatic Control, IEEE Transactions on*, vol. 56, no. 11, pp. 2499–2514, 2011.

[35] L. Lee and K. Poolla, "Identification of linear parameter-varying systems via lfts," in *Decision and Control, 1996., Proceedings of the 35th IEEE Conference on*, vol. 2, pp. 1545–1550 vol.2, Dec 1996.

[36] V. Verdult, *Non linear system identification: a state-space approach*. Twente University Press, 2002.

[37] V. Verdult and M. Verhaegen, "Subspace identification of multivariable linear parameter-varying systems," *Automatica*, vol. 38, no. 5, pp. 805–814, 2002.

[38] M. Lovera and G. Mercere, "Identification for gain-scheduling: a balanced subspace approach," in *American Control Conference 2007, ACC'07*, p. CDROM, 2007.

[39] J.-W. Van Wingerden, *Control of wind turbines with'Smart'rotors: proof of concept & LPV subspace identification*. TU Delft, Delft University of Technology, 2008.

[40] W. J. Rugh, *Linear system theory*, vol. 2. prentice hall Upper Saddle River, NJ, 1996.

[41] J.-W. van Wingerden and M. Verhaegen, "Closed loop identification of mimo hammerstein models using ls-svm," in *System Identification*, vol. 15, pp. 1650–1655, 2009.

[42] R. Tóth, V. Laurain, W. X. Zheng, and K. Poolla, "Model structure learning: A support vector machine approach for lpv linear-regression models," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 3192–3197, IEEE, 2011.

[43] J. V. Beck and K. J. Arnold, *Parameter estimation in engineering and science*. John Wiley & Sons Inc, 1977.

[44] T. B. Schön, A. Wills, and B. Ninness, "System identification of nonlinear state-space models," *Automatica*, vol. 47, no. 1, pp. 39–49, 2011.

[45] N. Aronszajn, "Theory of reproducing kernels," *Transactions of the American mathematical society*, pp. 337–404, 1950.

[46] M. J. Orr *et al.*, "Introduction to radial basis function networks," 1996.

[47] C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (gpml) toolbox," *The Journal of Machine Learning Research*, vol. 11, pp. 3011–3015, 2010.

[48] C. E. Rasmussen and Z. Ghahramani, "Occam's razor," *Advances in neural information processing systems*, pp. 294–300, 2001.

[49] P. M. Van den Hof, *Course SC4110: System Identification*. Delft University of Technology, 2006.

[50] V. Verdult and M. Verhaegen, "Kernel methods for subspace identification of multivariable {LPV} and bilinear systems," *Automatica*, vol. 41, no. 9, pp. 1557 – 1565, 2005.

[51] G. H. Golub, M. Heath, and G. Wahba, "Generalized cross-validation as a method for choosing a good ridge parameter," *Technometrics*, vol. 21, no. 2, pp. 215–223, 1979.

[52] I. Houtzager, P. Gebraad, J. van Wingerden, and M. Verhaegen, "Predictor-based subspace identification toolbox version 0. 4," 2010.

[53] B. D. Anderson and J. B. Moore, *Optimal filtering*. Courier Dover Publications, 2012.

[54] T. Chen, H. Ohlsson, and L. Ljung, "On the estimation of transfer functions, regularizations and gaussian processes—revisited," *Automatica*, vol. 48, no. 8, pp. 1525–1535, 2012.

[55] G. Pillonetto and G. De Nicolao, "Pitfalls of the parametric approaches exploiting cross-validation for model order selection," in *System Identification*, vol. 16, pp. 215–220, 2012.

[56] P. Stoica and R. L. Moses, "On biased estimators and the unbiased cramér-rao lower bound," *Signal Processing*, vol. 21, no. 4, pp. 349–350, 1990.

[57] L. Ljung and T. Chen, "What can regularization offer for estimation of dynamical systems?," in *Adaptation and Learning in Control and Signal Processing*, vol. 11, pp. 1–8, 2013.

[58] G. Pillonetto, F. Dinuzzo, T. Chen, G. D. Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, no. 3, pp. 657 – 682, 2014.

[59] G. Pillonetto, A. Chiuso, and G. De Nicolao, "Prediction error identification of linear systems: a nonparametric gaussian regression approach," *Automatica*, vol. 47, no. 2, pp. 291–305, 2011.

[60] A. Aravkin, J. V. Burke, A. Chiuso, and G. Pillonetto, "On the estimation of hyper-parameters for empirical bayes estimators: Maximum marginal likelihood vs minimum mse," in *16th IFAC Symposium on System Identification*, vol. 16, pp. 125–130, 2012.

[61] R. Carli, T. Chen, A. Chiuso, L. Ljung, and G. Pillonetto, "On the estimation of hyper-parameters for bayesian system identification with exponentially decaying kernels.," in *51st IEEE Conference on Decision and Control*, pp. 5260–5265, 2012.

[62] G. Pillonetto and G. De Nicolao, "Kernel selection in linear system identification part i: A gaussian process perspective," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 4318–4325, IEEE, 2011.

[63] T. Chen, H. Ohlsson, G. C. Goodwin, and L. Ljung, "Kernel selection in linear system identification part ii: A classical perspective," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 4326–4331, IEEE, 2011.

[64] T. Chen, M. Andersen, L. Ljung, A. Chiuso, and G. Pillonetto, "System identification via sparse multiple kernel-based regularization using sequential convex optimization techniques," *Automatic Control, IEEE Transactions on*, vol. 59, pp. 2933–2945, Nov 2014.

[65] F. P. Carli, T. Chen, and L. Ljung, "Maximum entropy kernels for system identification," *arXiv preprint arXiv:1411.5620*, 2014.

[66] T. Chen and L. Ljung, "Constructive state space model induced kernels for regularized system identification⋆," in *Proceedings of the 19th IFAC World Congress*, 2014.

[67] F. Dinuzzo, "Kernels for linear time invariant system identification," *arXiv preprint arXiv:1203.4930*, 2012.

[68] G. Goodwin, M. Gevers, and B. Ninness, "Quantifying the error in estimated transfer functions with application to model order selection," *Automatic Control, IEEE Transactions on*, vol. 37, no. 7, pp. 913–928, 1992.

[69] A. Aravkin, J. V. Burke, A. Chiuso, and G. Pillonetto, "Convex vs nonconvex approaches for sparse estimation: Lasso, multiple kernel learning and hyperparameter lasso," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 156–161, IEEE, 2011.

[70] G. Wahba, *Spline models for observational data*, vol. 59. Siam, 1990.

[71] G. Pillonetto and G. De Nicolao, "A new kernel-based approach for linear system identification," *Automatica*, vol. 46, no. 1, pp. 81–93, 2010.

[72] A. Chiuso, T. Chen, L. Ljung, and G. Pillonetto, "Regularization strategies for nonparametric system identification," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pp. 6013–6018, IEEE, 2013.

[73] I. Houtzager, J. van Wingerden, and M. Verhaegen, "Predictor-based subspace identification toolbox version 0.4, 2010," *URL http://www. dcsc. tudelft. nl/˜ datadriven/pbsid.*

[74] K. J. Åström and B. Wittenmark, *Computer-controlled Systems (3rd Ed.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997.

[75] D. Bauer, "Order estimation for subspace methods," *Automatica*, vol. 37, no. 10, pp. 1561–1573, 2001.

[76] T. Chen and L. Ljung, "Implementation of algorithms for tuning parameters in regularized least squares problems in system identification," *Automatica*, vol. 49, no. 7, pp. 2213–2220, 2013.

[77] F. P. Carli, A. Chiuso, and G. Pillonetto, "Efficient algorithms for large scale linear system identification using stable spline estimators," in *System Identification*, vol. 16, pp. 119–124, 2012.

[78] M. Lovera, T. Gustafsson, and M. Verhaegen, "Recursive subspace identification of linear and non-linear wiener state-space models," *Automatica*, vol. 36, no. 11, pp. 1639–1650, 2000.

[79] I. Houtzager, J. van Wingerden, and M. Verhaegen, "Recursive predictor-based subspace identification with application to the real-time closed-loop tracking of flutter," *Control Systems Technology, IEEE Transactions on*, vol. 20, no. 4, pp. 934–949, 2012.

[80] M. Huber, "Recursive gaussian process regression," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 3362–3366, May 2013.

[81] F. Felici, J.-W. Van Wingerden, and M. Verhaegen, "Subspace identification of mimo lpv systems using a periodic scheduling sequence," *Automatica*, vol. 43, no. 10, pp. 1684–1697, 2007.

[82] J. van Wingerden and M. Verhaegen, "Subspace identification of bilinear systems using a dedicated input sequence," in *Decision and Control, 2007 46th IEEE Conference on*, pp. 4962–4967, IEEE, 2007.

[83] V. Verdult and M. Verhaegen, "A kernel method for subspace identification of multivariable bilinear systems," in *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, vol. 4, pp. 3972–3977, IEEE, 2003.

[84] P. Gebraad, J. Van Wingerden, G. Van der Veen, and M. Verhaegen, "Lpv subspace identification using a novel nuclear norm regularization method," in *American Control Conference (ACC), 2011*, pp. 165–170, IEEE, 2011.

[85] V. Verdult, *Non linear system identification: a state-space approach*. Twente University Press, 2002.

[86] J.-W. van Wingerden, *Control of wind turbines with 'Smart' rotors: proof of concept & LPV subspace identification*. PhD thesis, 2008.

[87] A. Papoulis and S. U. Pillai, *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education, 2002.

[88] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in *Computational learning theory*, pp. 416–426, Springer, 2001.

Glossary

List of Acronyms

ARX	Auto-Regressive with Exogenous Input
CV	Cross-Validation
FIR	Finite Impulse Response
GCV	Generalized Cross-Validation
GP	Gaussian Process
LPV	Linear Parameter Varying
LS	Least Squares
LS-SVM	Least Squares Support Vector Machines
LTI	Linear Time Invariant
MAP	Maximum a Posteriori
MargLik	Marginal Likelihood
MCMC	Markov Chain Monte Carlo
ML	Maximum Likelihood
MOESP	Multivariable Output- Error State-sPace
MSE	Mean Squared Error
PEI	Prediction Error Identification
RBS	Random Binary Sequence
RBF	Radial Basis Function
RKHS	Reproducing Kernel Hilbert Space

SID	Subspace Identification
SISO	Single Input Single Output
SNR	Signal-to-Noise Ratio
SVD	Singular Value Decomposition
SysID	System Identification
VAF	Variance Accounted For
VARX	Vector ARX