Topology detection using Bayesian statistics

Eduard-Edis, Răclaru

Award date:
2018

Technische Universiteit
Eindhoven
University of Technology

Graduation project

Topology detection using Bayesian statistics

Răclaru Eduard-Edis
Student Id: 0928569
MSc. Systems and Control

supervised by
prof.dr.ir. P.M.J. Van den Hof
ir. H.H.M Weerts

Control Systems,
Department of Electrical Engineering,
Eindhoven University of Technology.

October 30, 2018

TU/e Technische Universiteit Eindhoven University of Technology

Declaration concerning the TU/e Code of Scientific Conduct

I have read the TU/e Code of Scientific Conduct[i].

In carrying out research, design and educational activities, I shall observe the five central values of scientific integrity, namely: trustworthiness, intellectual honesty, openness, independence and societal responsibility, as well as the norms and principles which follow from them.

Date

30.10.2018

Name

RĂCLARU EDUARD-EDIS

ID-number

0928569

Signature

Răclaru

Submit the signed declaration to the student administration of your department.

[i] See: http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/
The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.
More information about scientific integrity is published on the websites of TU/e and VSNU

January 15 2016

Abstract

A new trend in automation is to develop complex systems that involve networks of interconnected subsystems which interact physically and exchange information though a communication network. Uncertain factors could disrupt the interconnections producing unknown topology changes. Since the operation of the complex system might rely on knowing the current topology, it is of interest to investigate the possibility of recovering it from data. Two possible procedures for detecting the topology are presented, one based on prediction error modeling and the other based on models obtained though the stable-spline with exponential hyperprior method. The preference for using stable-spline with exponential hyperprior for topology detection is provided through simulations comparing and evaluating the two methods.

Contents

Chapter 1

Introduction

1.1 Motivation

The presence of complex networks of dynamic systems in many fields of study such as econometrics, neuroscience and automation, have lead to a raised interest in the use of measurement data for determining the interconnections between the network components. For example, in neuroscience, measurement records are used to identify key brain structures underlying the organization of a given brain function, such as, determining the interactions that appear between different areas of the motor cortex, involved in motor maintenance [14]. Another example would be in automation, where measurements from a distributed configuration of sensors over a network of, e.g gas pipelines, could be used to detect faulty areas, due to leakages, reducing in this way the maintenance and downtime costs. Similarly, it could be applied in petrol industry, water distribution, in the power grid, for sensor arrays, or in large scale manufacturing processes. The list of applications is not limited to those just mentioned, as dynamic networks are ubiquitous in natural and engineering sciences.

In practice, there are multiple possible benefits for knowing the topology of a network of dynamic systems depending on the application. Knowledge of topology can help reduce maintenance and downtime costs, and increase safety through fault detection capabilities. Another possible benefit is that it provides an understanding of dynamic networks with fixed but unavailable switching rules. In this regard, it could help extend the scientific knowledge of natural sciences, through empirical studies of complex processes. It also helps in the design of new systems based on an already existing network systems. Topology knowledge could also ease the automatic operation of a plant with a network structure through an appropriate choice of a distributed control configuration. It could also help, with the identification of a particular module in the network, for example, by choosing an appropriate experiment design. It helps in achieving reliable operation of complex processes, or in other words helps in achieving stability by implementing appropriate control actions. In this regard, it could help with the control of hybrid dynamical systems, to make them resilient to unexpected switches.

The importance of topology detection is also determined by the increasing number of engineering systems that function as networks. In the modern society, we encounter many engineering systems upon which its population is relying as it directly affects its welfare. Developing new systems is always a challenge that must thus be confronted for improving the economic and social being of the society. One way of developing new systems could involve a process through which multiple other systems are combined together to realize emergent holistic behavior. As a result, the new complex system could thus have a network structure, the management of which is crucial for the functioning of the system. Related to our problem of interest, it may appear that the topology of engineer developed networks should always be known by the system designer, with the exception of situation when faults occur. The following example, of a three tank system, will reveal that in a engineer developed network, we could have uncertain factors, other than spontaneous faults, that drive its

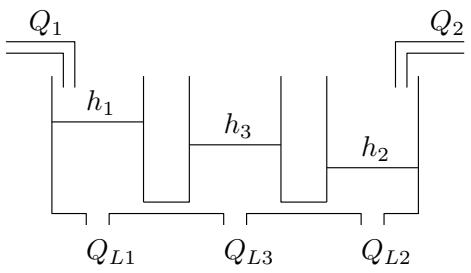


Figure 1.1: Three tank system.

topology, making it unpredictable. Besides providing a concrete application of topology detection, this example suggests the potential of topology detection for many other applications, which might involve similar uncertainty sources.

Three tank system

The importance of topology detection for engineering systems will now be illustrated through an example of a three tank system shown in Figure 1.1. The three tanks are interconnected through pipes at the bottom as shown in the figure. Substance can be introduced in tank 1 and 2 through actuated valves at the top of the two tanks, while at the bottom of each of the three tanks the substance is leaking. Q_1 and Q_2 denote the inlet flow rate of substance in the tanks, h_i , $i \in \{1, 2, 3\}$ represent the measured heights, and Q_{Li}, $i \in \{1, 2, 3\}$ are the leakage flow rates. The control objective of such a system is to maintain the heights h_1 and h_2 at specific values. To that end, by selecting $Q_1 - Q_{L1}$ and $Q_2 - Q_{L2}$ as controlled, input, variables, and h_1, h_2 as the process outputs, we define the dynamic model as a system with two inputs and two outputs. To solve the control problem, a multi-variable approach could be taken for controller design. Considerate study of the process, though, would reveal an alternative control strategy. If the heights are to be maintained at values such that $h_1 > h_3 > h_2$, then the substance will flow only in one direction in the connecting pipes, i.e. from tank 1 to 3 and from 3 to 2. Similarly, if $h_2 > h_3 > h_1$, then the flow direction will reverse. Also, if $h_1 > h_3$ and $h_2 > h_3$, then substance will pass from tank 1 to 3 and from 2 to 3. In this last situation, if h_3 is close to zero, we notice that tank 1 will not affect or be affected by tank 2. Thus, if we would be in this situation, we could apply two separate single input single output controllers for tanks 1 and 2 and avoid the more involved multi-variable approach. There are two ways of determining whether tank 1 is decoupled from 2, namely, through topology detection, or by verifying the state values h_1, h_2, h_3, and deducing from them the current operating structure. The last approach has, though, the inconvenient of requiring access to the state variable h_3, which might not be measured. In that case, a state observer based on Kalman filtering might be used to estimate the values of h_3, if the state h_3 is observable. In case it is not though, we would be unable to determine the current operating topology, based on the state variables. On top of that, since we don't know if the two systems are interconnected and we also don't have access to h_3, a full state feedback, multi-variable approach would have to be applied, but it would not be possible since it would require the full state values. Thus, if we would have knowledge that the two tanks are decoupled, from topology detection, we could apply individual single-input single-output controllers, without the need of knowing h_3. Generalizing, from this example, it might be beneficial to realize that more complex dynamic networks could similarly benefit from topology detection.

The diversity of applications and multitude of possible benefits of utilizing topology detection, as just presented, thus motivates us to investigate an effective procedure through which we could determine the network topology from measurement data.

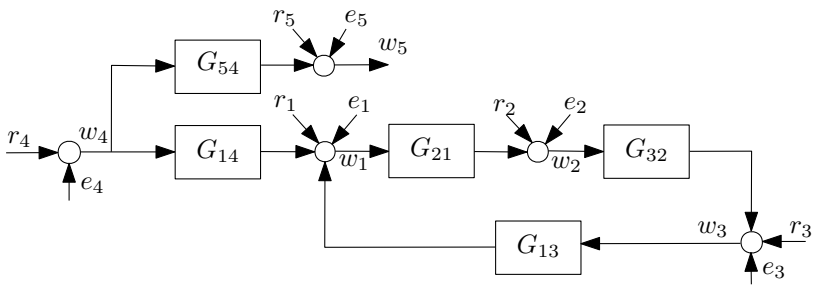


Figure 1.2: Network example. The circles indicate summation nodes.

1.2 Thesis problem statement

In this thesis, we will consider interconnections of dynamical systems, from which we gather measurements to be used for determining the topology of the considered network. The networks used in this thesis, are interconnections of discrete-time, linear-time-invariant, single-input single-output, dynamical systems called modules, along with a set of measurement points called nodes [4]. We define the nodes to be the signals that are inputs to at least one system. For illustration, an example is given in Figure 1.2. Then, the topology is defined as the set of all existing direct links between every pair of nodes.

An account of the main research question along with two other supportive ones, are given in diagram 1.3. The two annexed questions are used to provide the background useful for motivating and formulating the main question.

Given as an accepted fact that the topology is influencing the observed signals of the network in consideration, a rational first question is, could the interconnection structure be restored from the effect it has on these signals? To answer this question, we must ask ourselves beforehand, in what manner is the topology affecting the observed signals, or how could we measure this effect? Formulating the question for the simplest network, of two signals, we are interested in deriving a condition on signal measurements, for when a system is present between the two signals, with one of the signals being considered the input and the other the output. Then, extending, we would seek the conditions under which an intricate interconnection exists between the signals of a more complex network. This rational process is the subject of the first question in the diagram of Figure 1.3. In this report, the solution to the previously stated problems is provided through the concept of Granger causality [13], [14]. The key idea lies in the fact that for a two signal network, the presence or absence of a system between the two signals is a problem of determining the causal relationship between the respective signals. It is worth noting, that causality is referring to interconnections between signals, whereas topology is referring to interconnections between systems, and that the two representations are equivalent. The outcome of answering the first question in diagram 1.3 should be an effective strategy of detecting the topology of a network of dynamic systems based on Granger causality.

The second question in diagram 1.3, makes the transition from the first to the third, and main, question. It indicates the inconveniences of the topology detection procedure, suggested as an answer to the first question. In this report, Granger causality is tested, through the construction of predictive dynamic models meant to capture the signal transfer from one another. Hence, the problem of topology detection, or of testing causality, becomes a problem of dynamic modeling. In this situation, every pairwise signal relationship has to be modeled. The links between signals that are not present in the data generating network, should give dynamic models which would indicate very little influence or none at all, between the signals in question. To make the discrimination between present and absent connections plausible, we must express a relatively high confidence in our constructed models. Thus, topology detection through the suggested method, presents inconveniences that come with estimating predictive dynamic models.

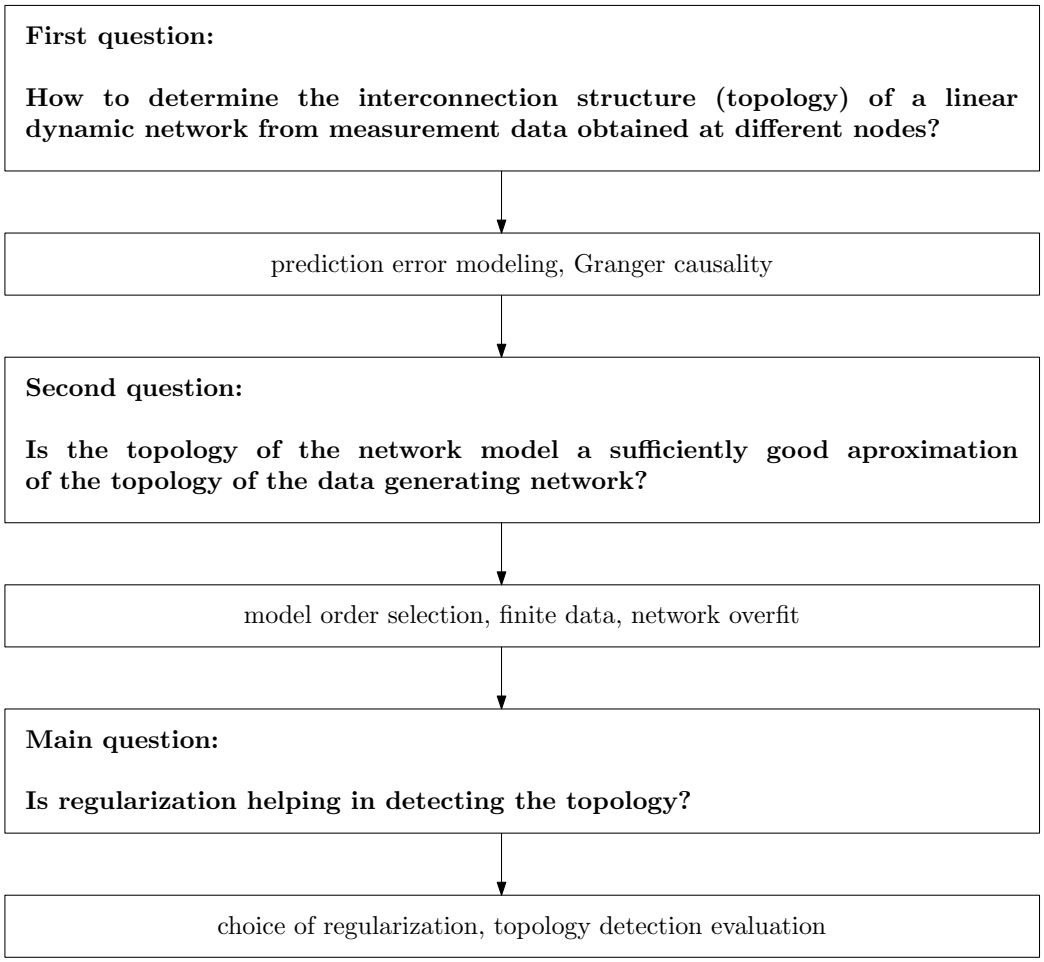


Figure 1.3: Thesis research questions.

Since the measured signals are affected by noise, the estimated module parameters will also be affected by it. Moreover, in a practical situation, working with finite time data sets adds to the parameter variance compared to the situation when even more data points are available. The parameter uncertainty, then, affects the predictions produced by our constructed models, thus decreasing our belief in the causal inference between the signals of our network, based on the estimates. In this regard, parsimonious model choices help in keeping low the noise effect on the estimates. However, model set specification, in dynamic networks, is an exhausting task as there are many module models to choose, and each model has to be tunned individually to a specific order. At this point, we understand the problem of model set selection for dynamic network models, as the problem of deciding a model set for each modeled module in the network.

The standard method for model selection used in single-input single-output system identification is to consider a model class and to tune the model order using a cross validation procedure based on residual analysis, i.e. for each different model order we apply the residual test to validate the model and we select the model with the lowest order that is still validated [1], [2]. Thus, in system identification, the model selection problem is to choose a model structure and a model order in such a way that we are able to capture all the relevant input-output relationships without fitting in the noise realization. In cases where the system is not in the model set, a trade-off between noise fitting and dynamics fitting has to be established. An example of this would be the case where we try to

obtain an approximate linear model for a non-linear data generating system.

In this thesis we aim to formulate automated mechanisms for model selection, for the purpose of obtaining low variance network models, which still capture all the important dynamics of the system under consideration. One of the most familiar automated mechanisms for model selection, in the literature, is based on information theoretic reasoning and uses Akaike's information criterion (AIC) [1], [2], [10]. These methods present the limitations of overfitting the models, and of increased computational complexity when multiple model orders have to be determined. We, thus, pursue another path to model order selection. An alternative and more recent procedure in the literature is based on using prior information in a Bayesian context to overcome the presented inconveniences of AIC methods. This brings us to the main research question, specified in diagram 1.3, which pushes us to investigate how regularization methods could benefit topology detection.

Besides aiding the model order selection, using prior information through Bayesian statistics for our network identification problem could also offer a leverage in the model quality even when the best model structure is already specified. This is a result of including correct prior information into our models which are stochastic in nature and might lead to a reduction of the uncertainty of our model parameters. Through the Bayesian approach, in contrast with the classic prediction error methods used in system identification, we could also trade bias for variance, reducing in this way the difference between the true and estimated parameters, expressed using the root-mean-squared-error [8]. Thus, increasing the model quality, through Bayesian statistics, by including additional information besides that which is present in the measurement data, would help to better asses the interconnections between the signals of our network. Using prior information to improve the estimates, seems to be useful for practical applications as the estimation is carried out using finite data sets which are affected by noise of a certain degree. For a given model structure, an estimation carried with less data points will increase the effect of the noise on the estimated parameters compared with an estimation done with a higher number of data points. Thus we shall try to reduce the noise effect from our models, using the prior information to improve on the estimates in the practical situation when the data sets are finite. The quality of our dynamic models is in their capability to make good predictions, i.e. reproduce the behavior of the data generating system, and reducing the uncertainty in the model will help making them better at prediction, which in turn will help to discriminated between absent and present connections.

Another important aspect that will be treated using Bayesian statistics is what we call the problem of network over-fit. This is the inconvenience of classic prediction error methods that they are not able to select which of the input signals affects the prediction of a given output, by setting the corresponding transfers to zero. In a Bayesian context, priors which produce sparse results ([9], [6], [16]) will be used to address this issue.

To answer the main question in diagram 1.3, we will analyze a choice of the Bayesian prior used to improve topology detection, named stable-splines with exponential hyperprior, introduced in [9]. In this thesis, we will try to give answers to all three questions of diagram 1.3, with the aim of developing an efficient topology detection procedure. To assess its performance we also provide a comparison with the results obtained through classic prediction error methods, by evaluating each topology detection procedure.

1.3 Thesis overview

The thesis will proceed as follows.

In chapter 2, we review the fundamentals of prediction error methods and we introduce the notation that will be used in this report. We express the problem of over-fit as a model selection problem that should be avoided to obtain models which are suited for prediction. The network

setup is described and afterwards we express the problem of model selection for network identification. Finally, we formulate our topology detection procedure based on Granger causality.

In chapter 3, we present the Bayesian alternative to single-input single-output system modeling, and its connection to regularized prediction error methods. We evaluate the stable-spline method, that we will apply for its automatic model order selection ability and good predictive performance. The used Bayesian estimation procedure comes with its own difficulties, in that it requires the solution to a non-convex optimization problem. In this regard, we provide two choices of initialization that help to successfully solve this issue.

In chapter 4, we provide our procedure of evaluating an estimator for its topology detection capabilities, and we firstly evaluate the classic prediction error method for a given data generating network example. Afterwards, we present the multi-variable Bayesian estimate known as the stable-spline with exponential hyperpriors. Similar to the single-input single-ouput case, a non-convex optimization problem needs to be solved. Appropriate choices of initializations are discussed and evaluated. A comparison of topology detection through classic prediction error methods, and stable-splines with exponential hyperpriors is provided along with a confirmation of the improvements.

Chapter 2

Topology detection of dynamic networks

In this chapter we build up from the fundamental results in system identification to the problem of topology detection. In the begining, we review the methodology and notations, behind prediction error methods for single-input single-output systems, and indicate the concrete estimator we will be using in the thesis. Thus, to define our choice of prediction error method, we specify the model structure, the criterion for selecting the parameters, and the predictor choice. Then, we move on to present the problem of over-fit when choosing the model orders. Afterwards, we introduce the considered network setup and associated identification problem indicating the issue of over-fit for networks. Finally, we reveal the topology detection approach and the use of network modeling in its derivation.

2.1 Predition error methods

One of the most used tools for modeling dynamic systems from measurement data, for a control systems engineer, is the prediction error method (PEM) [1], [2]. The underlying idea behind the method lies in the concept of prediction, which is used to derive models that able to reproduce closely the behavior of the true system. To illustrate the method, let us consider the discrete-time linear-time-invariant (LTI) single-input single-output system

$$y[t] = G(q^{-1};\theta_0)u[t] + e[t] \tag{2.1}$$

$$\mathbb{E}\{e[t]e^T[s]\} = \Lambda(\theta_0)\delta(t-s), \tag{2.2}$$

as our data generating system, with $y[t]$ the measured output, $u[t]$ the input and $e[t]$ a stationary white noise signal, and denote by $y^*[t] = G(q^{-1};\theta_0)u[t]$ the "noise-free" system output of the true system $G(q^{-1};\theta_0)$ which needs to be identified. Also, q^{-1} denotes the unit delay operator, $u[t-1] = q^1u[t]$, and $\mathbb{E}$ denotes the expectation operator. Λ indicates the parameterized covariance matrix of the noise e, and $\delta(t)$ is the Kronecker delta. The true plant $G(q^{-1};\theta_0)$ is part of a parameterized class of plants $G(q^{-1};\theta)$ with parameters θ. Here, θ_0 is indicating the parameters of the true plant. Notice that, throughout the thesis, we will assume the noise affecting the measurements to be white.

Given measurements of the inputs $u[t]$ and the outputs $y[t]$, we would like to model the data generating system using a parameterized general linear model structure of the same class as our data generating system,

$$\begin{aligned}\bar{y}[t] &= G(q^{-1};\theta)u[t] + \bar{e}[t],\\ \mathbb{E}\{\bar{e}[t]\bar{e}^T[s]\} &= \Lambda(\theta)\delta(t-s).\end{aligned} \tag{2.3}$$

Then the choice of parameters θ should be made such that the resulting model $G(q^{-1};\theta)$ is capable to make good predictions, i.e. reproduce the behavior of the data generating system.

To select a set of parameters $\hat{\theta}$ that make the model $G(q^{-1};\hat{\theta})$ good at predicting the behavior of the data generating systems, a predictor must first be defined. For this model structure, a predictor could generally be seen as a system able to reconstruct as good as possible the output $y[t]$ of our data generating system, using only the past measured values of the outputs and inputs, and the present input value. It can be expressed as

$$\hat{y}[t|t-1;\theta] = F(q^{-1};\theta)y[t] + L(q^{-1};\theta)u[t] \tag{2.4}$$

where $F(q^{-1};\theta)$ and $L(q^{-1};\theta)$ are parameterized filters in the parameter vector θ, used for specifying the predictor, with $F(q^{-1};\theta)$ having no feed-though term. Then, to measure the predictive performance of our model $G(q^{-1};\theta)$, we can define the prediction error as

$$\varepsilon[t;\theta] = y[t] - \hat{y}[t|t-1;\theta]. \tag{2.5}$$

The model parameters θ should be chosen as to give smallest prediction errors, without having access to neither $y[t]$ or $e[t]$ at the current time step t. The most used predictor in the literature [1], [2], and the one used in this thesis, is the one that minimizes the variance of the prediction error, out of all possible predictor defined by $\{F(q^{-1};\theta), L(q^{-1};\theta)\}$, i.e.

$$\hat{\theta}^{opt} = \arg\min_{\theta} \mathbb{E}\{\varepsilon^T(\theta)\varepsilon(\theta)\}, \tag{2.6}$$

and, for the considered situation of white output noise, is given by

$$\hat{y}[t|t-1;\theta] = G(q^{-1};\hat{\theta})u[t]. \tag{2.7}$$

The associated prediction error is given by

$$\varepsilon[t;\theta] = y[t] - G(q^{-1};\theta)u[t]. \tag{2.8}$$

Notice that, in defining the predictor, use was made of the assumption that the noise affecting the output is white, and in case it isn't, a predictor with a different structure should be used [1], [2].

For practical reasons, instead of optimizing the variance of the prediction error, we could search for the set of parameters $\hat{\theta}$ that minimize sample variance instead

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{N} \sum_{t=n_0}^{N} \varepsilon^2[t;\theta]. \tag{2.9}$$

where N is the number of input-output data samples and n_0 is a chosen starting point for evaluating the prediction error.

In this way, the resulting model $G(q^{-1};\hat{\theta})$ should be suitable for use in the recovery of the realization of the noise term. In other words, $\varepsilon[t;\theta]$ in ((2.5)) should reproduce the noise realization $e[t]$. This resemblance could be quantified, for example, by the functional distance

$$\sum_{t=n_0}^{N} (\varepsilon[t;\theta] - e[t])^2 \tag{2.10}$$

which is required to be as small as possible to make sure that we are not over-fitting. This ensures that our model has good predictive capabilities, i.e. it can finely reproduce the response of the system when tested against a new set of data.

It is worth reminding that the prediction error method has some interesting asymptotic properties, which makes it desirable as an estimator, namely consistency, asymptotic normality and statistical efficiency [1], [2]. The reason for considering these properties is that, in practice, if we could obtain sufficiently large data sets, we would have guarantees that the estimates should improve compared to the estimates obtained with less measurement samples.

In order to find the model parameters $\hat{\theta}$ in (2.9), we need to specify the structure in which $G(q^{-1};\theta)$ is parameterized in terms of θ. Two model classes will be used throughout the thesis, namely the finite impulse response (FIR) and the output error (OE) model.

The nonparametric finite impulse response model is a discrete model for a truncation of the impulse response. It can be represented as

$$y[t] = G(q^{-1})u[t-n_k] + e[t], \qquad G(q^{-1}) = g_1 + g_2 q^{-1} + \cdots + g_{n_g} q^{-n_g+1}, \tag{2.11}$$

where $G(q^{-1})u[t] = \sum_{i=0}^{n_g-1} g[t]u[t-i]$ is a convolution sum between the impulse response and the input and n_g is the order of the model, while n_k is the input time delay. It is considered as nonparametric since it is actually the discretization of the impulse response which is a continuous function. The estimated parameters in (2.9), for FIR models, could be determined using an analytical expression, since the predictor (2.7) is linear in the parameters and provides a cost function (2.9) which has a unique minimal point. The FIR model could be expressed in the linear regression form

$$y[t] = \varphi^T[t]\theta + e[t], \qquad \theta = \begin{pmatrix} g_1 & \dots & g_{n_g} \end{pmatrix}^T, \tag{2.12}$$

$$\varphi[t] = \begin{pmatrix} u[t-n_k] & \dots & u[t-n_k-n_g+1] \end{pmatrix}^T. \tag{2.13}$$

Or, when using $N \geq n_g + n_k$ measurement samples, as

$$Y_N = \Phi_N^T \theta + \Lambda_N, \tag{2.14}$$

$$Y_N = \begin{pmatrix} y[n_g+n_k] & y[n_g+n_k+1] & \dots & y[N] \end{pmatrix}^T, \tag{2.15}$$

$$\Phi_N = \begin{pmatrix} \varphi[n_g+n_k] & \varphi[n_g+n_k+1] & \dots & \varphi[N] \end{pmatrix}, \tag{2.16}$$

$$\Lambda_N = \begin{pmatrix} e[n_g+n_k] & e[n_g+n_k+1] & \dots & e[N] \end{pmatrix}^T. \tag{2.17}$$

Then, the analytical solution is given by the least squares solution

$$\hat{\theta}_N = (\Phi_N \Phi_N^T)^{-1} \Phi_N Y_N \tag{2.18}$$

for which efficient methods of solving it exist.

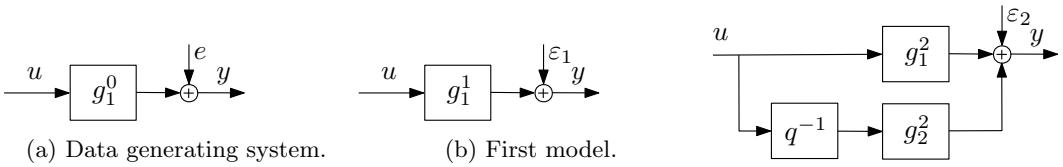
The output-error model, that will also be used in some instances in this thesis, is

$$y[t] = G(q^{-1})u[t-n_k] + e[t], \qquad B(q^{-1}) = b_1 + b_2 q^{-1} + \cdots + b_{n_b} q^{-n_b+1}, \tag{2.19}$$

$$\text{with } G(q^{-1}) = \frac{B(q^{-1})}{F(q^{-1})}, \qquad F(q^{-1}) = 1 + f_1 q^{-1} + \cdots + f_{n_f} q^{-n_f}. \tag{2.20}$$

with n_b and n_f the orders of the $B(q^{-1})$ and $F(q^{-1})$ polynomials, respectively, and n_k the input time delay. Differently from the estimates obtained when using the FIR model structure, the OE estimates don't have a closed form solution, therefore they have to be obtained by using search methods to solve the optimization problem (2.9). Furthermore, the criterion in (2.9), for the OE model, presents the difficulty of being non-convex, thus possibly leading to local optimal solutions.

We will be using prediction error methods to obtain models that are meant for prediction throughout the thesis. To model the network modules we will utilize the FIR model structure as it gives a closed form solution for which efficient methods for solving it exist. The choice of model orders needed for obtaining the estimates is not a trivial problem and will be discussed in the next section.



(a) Data generating system. (b) First model. (c) Second model.

Figure 2.1: Data generating system and model candidates for Example 2.2.1.

2.2 Over-fit and model selection

In practical situations we deal with finite data obtained in an experiment which runs a finite time. Sometimes the experiment time is costly and as such the number of samples acquired is relatively small. Thus, is some situations, the asymptotic properties mentioned earlier are mere unreachable ideal goals. If the model order is chosen too large it can happen that the residual ((2.5)) becomes zero. In this case, the noise realization is completely modeled by the module dynamics and we achieve a perfect fit of the model to the training data. The estimated model then is said to over-fit the data and will have poor predictive capabilities since when tested against another data set, corrupted by another noise realization, the model will offer a poor fit.

The model order thus should be kept small, and the model set should be chosen less flexible in order to prevent modeling the noise realization. The use of simpler models is also supported by "Principle of parsimony" because they are easier to use. On the other hand, we want a flexible model set with a reasonable model order, since we want to be able to capture all the relevant dynamics due to the input-output behavior seen in the data, making thus our model good for prediction.

In practice, we don't have access to the noise realization to verify whether it is reproduced by the obtained prediction error (2.8). We will instead determine the predictive quality of our model by using the fit measure

$$W = \left(1 - \frac{\sum_{t=nb+1}^{N}(y[t] - \hat{y}[t])}{\sum_{t=nb+1}^{N}(y[t] - \bar{y})}\right)100\%, \qquad \bar{y} = \frac{1}{N}\sum_{t=1}^{N} y[t], \tag{2.21}$$

also known as the coefficient of determination, where $\bar{y}$ is the sample mean of the output data $\{y[t]\}_{t=1}^{N}$. This measure expresses the amount of variance in the outputs predicted by the inputs. When we identify the model we choose the parameters to improve the fit. If we obtain 100% fit to the training data we usually will have a model that incorporates the noise realization. For this reason, cross-validation offers a better assessment of the model quality. It supposes that we determine the parameters on a training data set, then we validate the model by testing the fit on the model using another data set, called a validation data set. In this way we test to what extent the noise realization has affected the identified parameters.

In the following example, we will give an illustration of the connection between model selection and parameter over-fit. We consider as a data generating system an FIR model (2.11) of order 1, and we compare in terms of prediction two models, obtained from using the prediction error method (2.9) for two different parametrization choices, see figure (2.1). The first is an FIR model of order 1, thus, having the same structure as the data generating system. The second is an FIR model of order 2, thus representing an over-parameterized model for the data generating system. The predictive ability of the two models is compared by evaluating which of the two offers a better reproduction of the output noise $e[t]$ in (2.1) through their corresponding prediction errors (2.8). Actually, we will show that the first model reproduced better the realization of the noise $e[t]$ by demonstrating that $\sum_{t=n_0}^{N}(\varepsilon_1[t;\theta] - e[t])^2 \leq \sum_{t=n_0}^{N}(\varepsilon_2[t;\theta] - e[t])^2$, where ε_1 is the prediction error corresponding to the first model and ε_2 corresponds to the second one.

Example 2.2.1. Consider the gain system $G_0(q^{-1};\theta_0) = g_1^0$ with input u and measured output $y = g_1^0 u + e$, with e a white noise signal uncorrelated with u. Chose the FIR parameterization of the system as $G_1(q^{-1};\theta_1) = g_1^1$ with $\theta_1 = g_1^1$. Then the PEM solution, based on N input-output data samples, is (2.18)

$$\hat{\theta}_1 = \hat{g}_1^1 = (\Phi_N^1 {\Phi_N^1}^T)^{-1}\Phi_N^1 Y_N^1, \tag{2.22}$$

where $\Phi_N^1 = \begin{pmatrix} u[1] & u[2] & \dots & u[N] \end{pmatrix} \in \mathbb{R}^{1\times N}$ and $Y_N = \begin{pmatrix} y[1] & y[2] & \dots & y[N] \end{pmatrix}^T \in \mathbb{R}^{N\times 1}$. Replacing $Y_N = {\Phi_N^1}^T\theta_0 + \Lambda_N$, with $\Lambda_N = \begin{pmatrix} e[1] & e[2] & \dots & e[N] \end{pmatrix}^T \in \mathbb{R}^{N\times 1}$ the unknown realization of the noise, in the formula for $\hat{\theta}_1$, the estimate can be expressed as

$$\hat{\theta}_1 = \theta_0 + (\Phi_N^1 {\Phi_N^1}^T)^{-1}\Phi_N^1 \Lambda_N. \tag{2.23}$$

For the rest of the presentation we will use the following notation $U_1 = {\Phi_N^1}^T \in \mathbb{R}^{N\times 1}$, and in the new notation we have

$$\hat{\theta}_1 = \theta_0 + (U_1 U_1^T)^{-1} U_1^T \Lambda_N. \tag{2.24}$$

If we compute the residual $\mathcal{E}_1 = Y_N - \hat{Y}_N^1$ we will obtain

$$\begin{aligned} \mathcal{E}_1 =& Y_N - \hat{Y}_N^1 = [U_1\theta_0 + \Lambda_N] - [U_1(\theta_0 + (U_1^T U_1)^{-1}U_1^T\Lambda_N)] \\ =& \Lambda_N - U_1(U_1^T U_1)^{-1}U_1^T\Lambda_N = \Lambda_N - A_1\Lambda_N, \end{aligned} \tag{2.25}$$

with $A_1 = U_1(U_1^T U_1)^{-1}U_1^T$. Since e should be uncorrelated from u, if we pick N large enough, such that the correlation of Λ_N and U_1 is negligible, i.e. the convolution $\Lambda_N * U_1 \approx 0$, we can consider $\mathcal{E}_1 \approx \Lambda_N$, and thus we recover the output noise using the residual.

Consider again the same data generating system $G_0(q^{-1};\theta_0) = g_1^0$, but now we choose an FIR model $G_2(q^{-1};\theta_2) = g_1^2 + g_2^2 q^{-1}$ with parameters $\theta_2 = \begin{pmatrix} g_1^2 & g_2^2 \end{pmatrix}^T$. The PEM solution, given the same data set, is in this case

$$\hat{\theta}_2 = (\Phi_N^2 {\Phi_N^2}^T)^{-1}\Phi_N^2 Y_N, \tag{2.26}$$

where $\Phi_N^2 = U^T = \begin{pmatrix} U_1 & U_2 \end{pmatrix}^T \in \mathbb{R}^{2\times N}$, where $U_2 = \begin{pmatrix} 0 & u[1] & \dots & u[N-1] \end{pmatrix}^T \in \mathbb{R}^{N\times 1}$, and U_1 and Y_N are the same as defined before. Again, replacing $Y_N = U\tilde{\theta}_0 + \Lambda_N$, with $\tilde{\theta}_0 = \begin{pmatrix} \theta_0 & 0 \end{pmatrix}^T$, in the formula for $\hat{\theta}_2$, the estimate can be expressed as

$$\hat{\theta}_2 = \tilde{\theta}_0 + (U^T U)^{-1}U^T V. \tag{2.27}$$

The residual $\mathcal{E}_2 = Y_N - \hat{Y}_N^2$ will be

$$\begin{aligned} \mathcal{E}_2 =& Y_N - \hat{Y}_N^2 = [U\tilde{\theta}_0 + \Lambda_N] - [U(\tilde{\theta}_0 + (U^T U)^{-1}U^T\Lambda_N)] \\ =& \Lambda_N - U(U^T U)^{-1}U^T\Lambda_N. \end{aligned} \tag{2.28}$$

And again, since e is uncorrelated with u, if we pick N such that the correlation of Λ_N and U is negligible, i.e. $||U^T\Lambda_N||_2 \approx 0$, we can consider $\mathcal{E}_2 \approx \Lambda_N$, and thus we recover the output noise using the residual.

We will make a comparison of the two cases by looking at the obtained residuals, $\mathcal{E}_1$ and $\mathcal{E}_2$. Notice that the residual corresponding to the second case can be expressed as

$$\begin{aligned} \mathcal{E}_2 &= \Lambda_N - \begin{pmatrix} U_1 & U_2 \end{pmatrix} \left(\begin{pmatrix} U_1 & U_2 \end{pmatrix}^T \begin{pmatrix} U_1 & U_2 \end{pmatrix} \right)^{-1} \begin{pmatrix} U_1 & U_2 \end{pmatrix}^T \Lambda_N \\ &= \Lambda_N - \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} U_1^T U_1 & U_1^T U_2 \\ U_2^T U_1 & U_2^T U_2 \end{pmatrix}^{-1} \begin{pmatrix} U_1^T \\ U_2^T \end{pmatrix} \Lambda_N \qquad (2.29) \\ &= \Lambda_N - \frac{U_1 U_2^T U_2 U_1^T - U_2 U_2^T U_1 U_1^T - U_1 U_1^T U_2 U_2^T + U_2 U_1^T U_1 U_2^T}{U_1^T U_1 U_2^T U_2 - U_1^T U_2 U_2^T U_1}\Lambda_N \qquad (2.30) \\ &= \Lambda_N - A_2\Lambda_N \qquad (2.31) \end{aligned}$$

if we partition $U = \begin{pmatrix} U_1 & U_2 \end{pmatrix}$. Here A_2 is a simple notation for the complicated fraction given above. Notice that the inverted matrix is 2×2, and each of its entries is a scalar. Now we will try to show that

$$||\mathcal{E}_1 - \Lambda_N||_2 \leq ||\mathcal{E}_2 - \Lambda_N||_2 \tag{2.32}$$

where $||\cdot||_2$ is the vector 2-norm. This reduces to showing

$$||A_1\Lambda_N||_2 \leq ||A_2\Lambda_N||_2 \tag{2.33}$$

Or using the definition of the vector 2-norm, the following condition will be sufficient

$$A_2 - A_1 \geq 0 \tag{2.34}$$

where ≥ 0, in this inequality, indicates matrix positive semidefiniteness, and A_1 and A_2 are idempotent matrices.

$$A_2 - A_1 = \frac{U_1U_2^TU_2U_1^T - U_2U_2^TU_1U_1^T - U_1U_1^TU_2U_2^T + U_2U_1^TU_1U_2^T}{U_1^TU_1U_2^TU_2 - U_1^TU_2U_2^TU_1} - U_1(U_1^TU_1)^{-1}U_1^T \tag{2.35}$$

$$= \frac{(U_1^TU_2)(U_2^TU_1)U_1U_1^T - (U_1^TU_1)(U_2^TU_1)U_2U_1^T - (U_1^TU_1)(U_1^TU_2)U_1U_2^T + (U_1^TU_1)(U_1^TU_1)U_2U_2^T}{(U_1^TU_1)[(U_1^TU_1)(U_2^TU_2) - (U_1^TU_2)(U_2^TU_1)]} \tag{2.36}$$

Notice that since U_1 and U_2 are column vectors, the term $U_1^TU_1$ is an inner product, thus a scalar and it commutes with the matrices.

To show inequality ((2.34)), we thus need to show that the numerator is a positive semidefinite matrix, since the denominator, which is a scalar, is always positive because $U_1^TU_1$ is an inner product and $(U_1^TU_1)(U_2^TU_2) - (U_1^TU_2)(U_2^TU_1) \geq 0$ by Cauchy-Bunyakovsky-Schwarz inequality. The numerator can be written as

$$(U_1^TU_1)\left[\frac{(U_1^TU_2)(U_2^TU_1)}{U_1^TU_1}U_1U_1^T - (U_2^TU_1)U_2U_1^T - (U_1^TU_2)U_1U_2^T + (U_1^TU_1)U_2U_2^T\right] \tag{2.37}$$

$$= (U_1^TU_1)\left[\sqrt{U_1^TU_1}U_2^T - \frac{(U_1^TU_2)}{\sqrt{U_1^TU_1}}U_1^T\right]^T\left[\sqrt{U_1^TU_1}U_2^T - \frac{(U_1^TU_2)}{\sqrt{U_1^TU_1}}U_1^T\right] \tag{2.38}$$

Thus the numerator is positive semidefinite since it is the multiplication of a positive scalar and a quadratic form. From this, we conclude that, indeed, inequality ((2.34)) holds. This implies that inequality ((2.32)) also holds, and it shows that, for this example, adding more parameters than needed has the effect of increasing the difference between the residual and the true noise term.

2.3 Network identification setup

Throughout the thesis, we consider a dynamic network to be an interconnection of single-input single-output, discrete-time, linear-time-invariant systems also called modules, where the input to a certain system could be a summation of the outputs from some of the other systems of the network. A set of L scalar measured signals w_j, $j \in \{1, \ldots, L\}$, are taken at nodes such that each of these signals represents either the input to at least one system. Then, every node signal could be expressed as

$$w_j[t] = \sum_{k \in \mathcal{N}_j} G_{jk}^0(q^{-1})w_k[t] + r_j[t] + e_j[t], \tag{2.39}$$

with $G_{jk}^0(z)$ a proper rational transfer function and $\mathcal{N}_j$ the set of indices of signals w_k, $k \neq j$, for which $G_{jk}^0 \neq 0$. Also, e_j is a white Gaussian noise of zero mean and variance σ_e^2, which is not

measured and r_j is a known excitation signal.

In this thesis, we will assume that our networks are stable and also that each system $G^0_{jk}(z^{-1})$ is stable and has the following output-error structure

$$G^0_{jk}(z^{-1}) = z^{-1}\frac{B(z^{-1})}{F(z^{-1})} = z^{-1}\frac{b_0 + b_1 z^{-1} + \cdots + b_{nb} z^{-nb}}{1 + f_1 z^{-1} + \cdots + f_{na} z^{-nf}}, \quad j,k \in \{1,\ldots,L\}, j \neq k. \tag{2.40}$$

Furthermore, it is assumed that the excitation signals r_j, $j \in \{1,\ldots,L\}$, are mutually uncorrelated, white Gaussian noises of zero mean and variance σ_r^2. The noise signals e_j, $j \in \{1,\ldots,L\}$, are also assumed mutually uncorrelated, uncorrelated with the excitation signals and they are all considered always present.

2.4 Considered network identification problem

From a network as presented earlier, we acquire measurements by sampling the signals at the nodes with a sampling period of T_s. The measured signals are represented by the collection of samples $\{w_k[t]|1 \leq t \leq N\}$, $\forall\, k \in \{1,\ldots,L\}$, where N is the total number of samples.

Assuming the topology of the network is unknown, the network identification problem is then to identify the transfer functions between each pair of signals, from the data points $\{w_k[t]|1 \leq t \leq N\}$, $\forall\, k \in \{1,\ldots,L\}$. Thus, we must choose a set of parameterized models for each possible interconnection pair of signals $\{w_j, w_k\}$, $j,k \in \{1,\ldots,L\}$, $j \neq k$. For each such pair we choose two module models $G_{jk}(q^{-1};\theta^j)$ and $G_{kj}(q^{-1};\theta^k)$ to capture both signal flow directions, i.e. $w_k \rightarrow w_j$ and $w_j \rightarrow w_k$. Therefore, the network model used is

$$w_j[t] = \sum_{k \in \mathcal{X}_j} G_{jk}(q^{-1};\theta^j) w_k[t] + r_j[t] + e_j[t], \qquad j \in \{1,\ldots,L\}, \tag{2.41}$$

where X_j the set of indices of signals w_k, $k \neq j$. Notice the difference between $\mathcal{X}_j$ and $\mathcal{N}_j$ defined earlier. $\mathcal{X}_j$ is used in the model structure to select both present and absent systems in the data generating system, whereas $\mathcal{N}_j$ selects only the present systems in the data generating system (2.39). For each individual single-input single-output model $G_{jk}(q^{-1};\theta^j)$ we will use the FIR model parametrization given by (2.11). To obtain our network model, we then have to search for suitable model parameters θ^j, $j \in \{1,\ldots,L\}$, from the data.

An issue with the estimation the model parameters is that the networks we consider may contain closed-loops. We say that two signals w_j and w_k are in a closed-loop if there is a causal path, that could possibly go though other nodes, both from w_j to w_k as well as from w_k to w_j. Systems in closed-loops require additional attention since the input to the identified system gets correlated with the output additive noise, which leads to having no guarantee on the consistency of the prediction error method. To overcome this issue, we are going to identify our network modules through the direct approach for general network topology presented in [4]. Therefore, for each separate measured signal w_j, $j \in \{1,\ldots,L\}$, we will be using the the multi-input single-output predictor

$$\hat{w}_j(t|t-1;\theta^j) = \sum_{k \in \mathcal{X}_j} G_{jk}(q^{-1};\theta^j) w_k[t] + r_j[t] \tag{2.42}$$

to define the prediction error

$$\varepsilon_j[t] = w_j[t] - \hat{w}_j(t|t-1;\theta^j). \tag{2.43}$$

Then, we are going to estimate the parameters $\theta = \begin{pmatrix} \theta^{1T} & \cdots & \theta^{LT} \end{pmatrix}^T$ of the network model (2.41) by minimizing separately, the sample variance of the prediction error (2.43),

$$\hat{\theta}^j = \arg\min_{\theta^j} \frac{1}{N}\sum_{t=1}^{N} \varepsilon_j^2[t;\theta^j] = \arg\min_{\theta^j} \frac{1}{N}\sum_{t=1}^{N} (w_j[t] - \sum_{k \in \mathcal{X}_j} G_{jk}(q^{-1};\theta^{jk}) w_k[t] - r_j[t])^2, \tag{2.44}$$

where

$$\theta^j = \begin{pmatrix} {\theta^{jk_1}}^T & {\theta^{jk_2}}^T & \dots & {\theta^{jk_{L-1}}}^T \end{pmatrix}^T, \quad \text{with } k_i = i \text{ if } i < j \text{ and } k_i = i+1 \text{ if } i > j. \tag{2.45}$$

and when using the FIR model parameterization,

$$G_{jk}(q^{-1};\theta^{jk}) = q^{-1}g_1^{jk} + q^{-2}g_2^{jk} + \dots + q^{-n_g}g_{n_g}^{jk}, \quad \theta^{jk} = \begin{pmatrix} g_1^{jk} & g_2^{jk} & \dots & g_{n_g}^{jk} \end{pmatrix}^T. \tag{2.46}$$

The separation for estimating the parameters is possible because we assumed that the noise terms e_j in (2.39) are mutually uncorrelated. Proposition 2 from [4] establishes that consistent estimates are obtained directly from (2.44) if the following conditions are met:

1. The noise terms e_j, $j \in \{1, \dots, L\}$ are mutually uncorrelated, as well as uncorrelated with all the excitation signals r_k, $k \in \{1, \dots, L\}$.
2. For both the data generating system and the model, every loop through node j has a delay.
3. The predictor input signals w_k in (2.42) are persistently exiting of sufficient order.
4. the system is in the model set, i.e. there exists a θ^{0j} such that $G_{jk}(z;\theta^{0j}) = G_{jk}^0(z)$ for all $j, k \in \{1, \dots, L\}$, $j \neq k$.

Through the assumptions that we made in the previous section, all of the conditions are satisfied in our case except the last one. The modules of the data generating network (2.39) have an output-error structure whereas we use the FIR structure to model them. The output-error models could be exactly reproduced by using an infinite impulse response representation. Considering our assumption, that the models are stable, FIR representations could be used to give approximations of the output-error models for any accuracy, by using a suitable FIR model order. Thus, for a large enough FIR model order n_g we could say that there exists FIR parameters θ^{0j} such that $G_{jk}(z;\theta^{0j})$ gives a sufficiently accurate approximation of $G_{jk}^0(z)$. Then as the number of data points grows, we would want θ^j to converge to θ^{0j}. Furthermore, in any practical situation, the "system in the model set" condition is never satisfied exactly. For these reasons and the computational difficulty associated with deriving output-error models, we consider that using FIR models in our case is appropriate.

2.5 Network over-fit problem

The network identification problem introduces an over-fit notion specifically for networks with unknown topology. We will illustrate this through the following example.

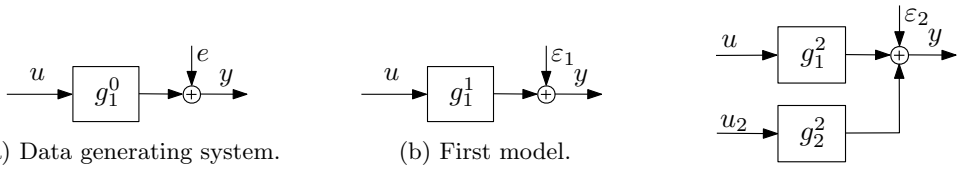
Example 2.5.1. Consider the dynamic network

$$w_1[t] = G_{12}(q^{-1};\theta^0)w_2[t] + e_1[t] \tag{2.47}$$
$$w_2[t] = G_{23}(q^{-1};\theta^0)w_3[t] + e_2[t] \tag{2.48}$$
$$w_3[t] = e_3[t] \tag{2.49}$$

where e_1, e_2, e_3 are mutually independent white noise Gaussian processes. In what follows we will examine only the multi-input single-output identification problem when w_1 is considered as output, and w_2 and w_3 are inputs. We wish to investigate two situations, one in which the true topology is known and one in which the topology is not known.

When the true topology is known, we could parameterize only the systems that actually exist $w_1 \leftarrow w_2$. Then the model set, corresponding to the identification problem with w_1 as output, would be composed of only one system

$$\mathcal{M}_1 = \{G_{12}(q^{-1};\theta) | \theta \in \mathbb{R}^p\} \tag{2.50}$$



(a) Data generating system. (b) First model. (c) Second model.

Figure 2.2: Data generating system and model candidates for Example 2.5.2.

where p is the number of parameters used for modeling G_{12}.

On the other hand, when the topology is unknown, we have to parameterize both interconnections $w_1 \leftarrow w_2$ and $w_1 \leftarrow w_3$. In this case, the model set would be composed of two systems

$$\mathcal{M}_2 = \{G_{12}(q^{-1};\theta_1), G_{13}(q^{-1};\theta_2) | \theta_1 \in \mathbb{R}^p, \theta_2 \in \mathbb{R}^p\}. \tag{2.51}$$

If we express the first model set as

$$\mathcal{M}_1 = \{G_{12}(q^{-1};\theta_1), G_{13}(q^{-1};\theta_2) | \theta_1 \in \mathbb{R}^p, \theta_2 = 0_{p\times 1}\}, \tag{2.52}$$

we could say that

$$\mathcal{M}_1 \subset \mathcal{M}_2, \tag{2.53}$$

or that the model set $\mathcal{M}_2$ covers the model set $\mathcal{M}_1$. Thus, we conclude that, in case knowledge on the topology is available, the model set could be reduced compared to the situation when such information is not available.

The dynamic network model set could be specified by first providing the topology, and then providing model orders for the links that are considered present. Then one model set is said to be included into another if, the topology of the latter model set covers the topology of the first model set, and the models orders of the first model set can be covered by their respective counterparts in the latter model set.

Example 2.5.2. Consider the situation from Example 2.2.1, but now in the second identification experiment, instead of choosing to increase the model order, we consider a MISO model with two inputs and one output and each of its SISO models has order 1 and is modeled using the gains g_1^2 and g_2^2, respectively, see Figure 2.2. Then $U = \begin{pmatrix} U_1 & U_2 \end{pmatrix}$, is composed of two different signals U_1 and U_2, with $U_2 = \begin{pmatrix} u_2[1] & u_2[2] & \dots & u_2[N] \end{pmatrix} \in \mathbb{R}^{N\times 1}$. The rest of the derivations are the same as in Example 2.2.1. Thus, when modeling modules that are not present in the MISO setup, we obtain an increase in the difference between the residual and the actual noise compared to the situation when the model order is chosen to match the true system order.

If we take an identification approach as presented in Section 2.4, where we try to model at the same time both existing and non-existing connections between the measurement nodes, we could end up with models for the absent connections that do not exactly give a zero transfer function. This will affect the topology detection as will be presented in the following section.

2.6 Topology detection approach

In this chapter we revealed what is the used approach in the thesis for modeling dynamic networks using the tools of system identification. We started by reviewing the PEM, and emphasized the importance of prediction as the most important measure of the quality of our models, and showed, through an example, the importance of model order selection for obtaining good predictors. We have, also, redefined the notion of model set for dynamic network models by emphasizing the role of the model topology. Next we wish to examine how can we determine the model topology from the data.

2.6.1 Granger causality approach to topology detection

Determining of the interconnection structure in the network could be expressed as checking if ordered pairs (w_i, w_j) of signals, for $i, j = \overline{1, nw}, i \neq j$, form causal relationships, i.e. w_i causes w_j. Such a problem has been considered in [13]. The concept of Granger causality relies on the fact that the cause should precede the effect and it states that signal w_i causes signal w_j if we can show/prove that w_i helps in making predictions of the future of w_j, compared with the situation where w_i would not be used to make those predictions. This indicates that the cause embeds unique information useful in the construction of the future effect, without which a decrease in prediction would be observed.

In this thesis the approach is to build predictive models, in order to determine the interconnection structure. Granger causality is determined by comparing the variance of the prediction error for two model choices. For the first model, w_i is not used to predict w_j, whereas for the second model, w_i is used in the prediction of w_j. Then the model that is best for prediction out of the two is the one that indicates causality. For a single-input single-output system of the form (2.1), Granger causality from the input to the output could be tested in one go by applying the prediction error method (2.9) to determine an FIR model (2.11). Provided that we have selected appropriately the model order, the prediction error method will select itself the most appropriate structure the gives the best variance of the prediction error. Thus, if the input would help to predict the output then the solver will give an FIR model indicating this. On the other hand, if the input is not connected to the output then the solver should give an FIR model which has a very small gain. It is interesting to notice that for our considered case, if the input is not connected to the output then the output is white noise, thus the outputs own past will also not help to improve the predictions of its future values.

By increasing the credibility in the predictive performance, for example by increasing the fit to estimation/validation data, we can actually assess better the effect that the cause has on its outcome, i.e. we can capture better and in more obvious way this effect. Since we capture more of the input-output dynamics, we also make the relationship less sensitive in face of errors. Thus, the better the model is in prediction the better we could assess the influence that the cause has on the future predictions of the effect. A model which is worse at prediction, on the other hand, will capture less from this influence. An important aspect for obtaining appropriate predictive models, for determining causality, is the correct use of model structure, as was discussed in section 2.2. In case the condition "system in the model set" is satisfied, by selecting model orders lower than the true system orders, we obtained biased estimates. Then the gain of the system will also be biased, and this might lead to errors when taking classification decisions.

2.6.2 Hypothesis testing approach to topology detection

The problem of causality between two signals could also be approached from a statistical perspective, which involves hypothesis testing [15]. We start from the assumption of the null hypothesis, that the directed connection is absent, and decide whether to reject it or not. This is done through a statistical test which should express the condition under which the null hypothesis will be rejected, while accepting the alternative hypothesis.

$$\begin{cases} \text{Null hypothesis:} & \text{Connection from } w_i \text{ to } w_j \text{ is absent. } W = 0. \\ \text{Alternative hypothesis:} & \text{Connection from } w_i \text{ to } w_j \text{ is present. } W \neq 0. \end{cases} \tag{2.54}$$

W is the coeficient of determination introduced in equation (2.21), which indicates the predictive power of the model. For linear regression models, with zero mean Gaussian noise, this can be tested through the t-test statistic

$$t_{obs} = \frac{\sqrt{W(N-n)}}{\sqrt{1-W}} \tag{2.55}$$

where N is the number of samples, n is the number of regressors, and t_{obs} has a Student t-distribution [15]. Then the rejection of the null hypothesis is done when $t_{obs} > t_{N-n}(\alpha/2)$, where $t_{N-n}(\alpha/2)$ is the evaluation of the Student t-distribution of $(N-n)$ degrees of freedom at the value of $\alpha/2$ which indicates the significance level (usually the 5% significance level). Notice that for $W = 0$ then $t_{obs} = 0$, and for $W \to 1$ then $t_{obs} \to \infty$. Thus, for models good at prediction, with W close to 1, then possibly, $t_{obs} > t_{N-n}(\alpha/2)$.

This approach is useful in situations where there is an expectation of very accurate prediction error modeling. In this thesis, we will not take this approach, and we will accept every model that gives reasonable fit according to (2.21) to estimation and validation data.

2.6.3 Determining causality between multiple measured signals

In dynamic networks we usually have more than one causal direction to test. In this case, applying the causality test to pairs of signals might lead to erroneous results as is illustrated in the following ([14]). Let there be three signals, w_i, w_j, w_k, connected as follows

$$w_j[t] = G^0_{jk}(q^{-1})w_k[t] + v_j[t], \quad w_k[t] = G^0_{ki}(q^{-1})w_i[t] + v_k[t], \quad w_i[t] = v_i[t]. \tag{2.56}$$

v_j, v_k, v_i are mutually uncorrelated, white noise, Gaussian processes of zero mean and fixed variance. By testing pairwise Granger causality, we would end up deducing a direct interconnection from w_i to w_j ($G_{ji} = G^0_{jk}G^0_{ki}$), which is actually not present. The solution to this problem, is to consider a multivariate prediction. So, if, in order to determine the connection from w_i to w_j we consider the MISO identification from $\begin{pmatrix} w_i & w_k \end{pmatrix}^T$ to w_j, then the estimate of G^0_{jk} should model both the transfer $v_k \to w_j$ and $G^0_{ki}v_i \to w_j$, which would predict most of the w_j signal power leaving no part of it left to be explained through a direct connection $w_i \to w_j$.

To detect the topology of the network, we will thus take a multi-input single-output approach for modeling the relationships between the measured signals as was described in Section 2.4.

2.6.4 Approach taken in this report

Let us consider that a network model has been obtained as described in Section 2.4, thus the estimated modules $G_{ji}(q^{-1};\theta_j)$, $\forall j, i \in \{1,\dots,L\}$, $j \neq i$, are available. The fit to estimation or validation data could be calculated for a single multi-input single-output model (2.41), with the formula (2.21). This measures the quality of prediction for the multi-input single-output model as a whole, and does not provide any information on the quality of prediction for each individual module involved. To determine which of the modules are effectively improving the total fit, the topology of the network will then be determined from the gains of the network module models $||G_{ji}||_{\mathscr{H}_2} = \sum_{t=0}^{\infty} h^2_{ji}[t]$, where $h_{ji}[t]$, $t \in \mathbb{N}$, are the impulse response coefficients of G_{ji}. Thus the topology of the network is determined by the following discrimination rules

$$\begin{cases} \text{if } ||G_{ji}||_{\mathscr{H}_2} < T & \text{then the connection from } i \text{ to } j \text{ is absent,} \\ \text{otherwise} & \text{the connection from } i \text{ to } j \text{ is present.} \end{cases} \tag{2.57}$$

where T is a selected threshold, which should indicate the significance level for the classification rule [18].

A noticeable difference between the method of hypothesis testing to topology detection and the one based on gain classification with a specific threshold is as follows. In hypothesis testing the quality of prediction of our model is playing a direct role in the classification of the connection, through the t-test statistic, where we would reject the null hypothesis if the model is sufficiently predictive given the significance level. In gain classification, on the other hand, the quality of prediction was not mention. There we build models of as best prediction quality as we can, by using model order selection mechanisms. Then no matter the exact prediction quality obtained, we classify them

using a fixed threshold. The hypothesis approach has the drawback that when good predictive models could not be obtained, according to the significance level, the classification might be misleading. The gain test approach, on the other hand, has the drawback that it does not take the quality of prediction into account. However, in this report we are going to take the gain classification approach to topology detection. To improve this strategy, in the remainder of the report, we will address two aspects. The first one is related to improving the predictive performance of our models. The second one will be to treat the problem of network overfit introduced in section 2.5.

2.7 Summary and conclusions

In this chapter, we have provided our answer to the first research question presented in diagram 1.3 from Section 1.2, by presenting an effective topology detection procedure based on Granger causality. The key element of this procedure is the concept of Granger causality which could be tested though the construction of dynamic models that are able to predict the values of the nodes from the network. Therefore, we began the chapter by presenting how will a single module be modeled, in this thesis, from input-output data through the PEM. In this regard we have indicated the type of data generating system, the predictor, the model structure and parameterizations, and the criterion for choosing the parameters used. We then established that the quality of our models is given by their ability to make predictions of the output of the data generating system. Through Example 2.2.1. we have shown that the choice of model order influences the ability, of the models, to make predictions. Thus, an appropriate choice of model order is also essential for determining the topology through Granger causality. Afterwards, the types of data generating networks considered were defined and the network identification procedure was expressed as multiple multi-input single-output identification problems solved using the direct method for closed-loop systems. A new problem was indicated in the network context differently from the single module case, which affects the ability in making predictions, of the module models, which was called the network over-fit problem. The similarity with the problem of model order selection was indicated through Example 2.5.2. The problem addresses the choice of model structure for multi-input single-output system identification, which as it turns out, is important for topology detection. The less is the noise affecting the models of the null connections, the better it is for topology detection as the discrimination threshold could be picked smaller. In this regard, the smaller the threshold could be picked, the easier is to discriminate between present and absent connections. Once the prerequisites have been established, the topology detection procedure based on Granger causality is stated. The approach uses the models obtained through the multi-input single-output method presented in the previous sections to determine the topology. Connections which are not present between signals, and are modeled, are affecting the topology detection mechanism. As such a discrimination has to be made in order to select the true connections. The use of model validation for discriminating between connections is not treated in this report. In conclusion, the topology detection procedure presented has two major inconveniences, that support the legitimacy of the second question in diagram 1.3. These are the choice of model order for each connection of the network, and the network over-fit problem. Solutions to these, will be addressed in the next two chapters.

Chapter 3

System modeling using regularized prediction error methods

3.1 Bayesian statistics

In this chapter, we will investigate the use of Bayesian estimation, as an alternative procedure, to the prediction error methods, for modeling dynamic systems from measurement data for the purpose of topology detection. The reason we would want to use a Bayesian approach to network modeling is twofold. On one hand, in order to make our topology assessment more plausible, we want to improve the predictive perfomance of our models, for which Bayesian methods have shown promising results [8], [10]. The main problem with prediction error methods indicated in the previous chapter, which affects their predictive capabilities, is that of choosing an appropriate model order. This problem becomes even more complicated for networks of systems, because it involves deciding simultaneously the model order of multiple modules. Recent developments in the literature [10], use a Bayesian approach for selecting appropriate model orders. In this thesis, we are going to study the effect of this new procedure on topology detection. The other reason for using a Bayesian approch to network modeling is that we want to solve the problem of network over-fit, expressed in the previous chapter. Doing so, would help us decide the causality between signals by selecting only the input signals that are contributing for the prediction of the considered output signal. In this chapter we only treat the approach to system modeling through Bayesian statistics, leaving the solution for the problem of network over-fit to chapter 4.

In the prediction error methods framework, we treat the parameters of the model as deterministic but unknown variables and we look for a tight confidence region inside which we believe that the true parameters might be. On the other hand, in the Bayesian perspective we start from a probabilistic model that expresses the intial belief we have about the system, then, using new observations or evidence we update the model, to make it suitable in explaining those observations. A new model is thus obtained which could, as well, be used as a prior for deriving a newer model based on newer observations, and so on. Thus as data becomes available, we try to pull our model from the previous belief to conform with the newly available evidence.

Formally, the Bayesian estimator is defined as the result of minimizing the Bayes risk, which is usually taken as the mean-square-error between the estimated parameters and the true parameters, i.e.

$$\hat{\theta}_{MMSE} = \arg\min_{\theta} \mathbb{E}\{(\theta(Y_N) - \theta_0)^2\}. \tag{3.1}$$

For this choice of Bayes risk, it is also known the minimum-mean-squared-error estimator (MMSE). In the formula, Y_N are the output observations which, along with the selected inputs U_N, represents the data used for estimation; θ_0 represents the true parameters assuming the system is in the

model set; and the expectation is taken over the joint distribution of Y_N and θ. The parameters $\theta(Y_N)$ implicitly depend also on the input data set U_N which is considered fixed and deterministic as opposed with Y_N. It can be shown, using manipulations of the conditional expectation, that formula (3.1) is equivalent with

$$\hat{\theta}_{MMSE} = \mathbb{E}(\theta|Y_N). \tag{3.2}$$

If this conditional expectation is taken over a probability distribution for which the expected value corresponds also to its mode (i.e. the maximum value of the distribution), such as for symmetrical unimodal distribution, e.g. the Gaussian distribution, the model parameters will be given by the maximum a posteriori estimator (MAP)

$$\hat{\theta}_N^{apost} = \arg\max_\theta p(\theta|Y_N) = \arg\max_\theta \frac{p(Y_N|\theta)p(\theta)}{p(Y_N)}. \tag{3.3}$$

The second equality follows from Bayes rule, and $p(Y_N)$ could be neglected as it does not affect the optimization since it does not depend on the θ. Here, $p(\theta)$ is called the prior and it is used to adjust the preference for solutions in the solution space $\theta \in R^n$, i.e. to influence the posterior probability of the parameters given the data.

Let's consider the linear regression model (given in equations (2.14), (2.16))

$$Y_N = \Phi_N^T\theta + \Lambda_N, \quad \theta \in \mathbb{R}^n, \quad e[t] \sim \mathcal{N}(0, \sigma^2), \tag{3.4}$$

where the noise term is Gaussian distributed. Also, we will assume the prior on the parameters to be Gaussian distributed

$$\theta \sim \mathcal{N}(0, P_n). \tag{3.5}$$

Notice that the prior is specified both by the distribution family, in this case Gaussian, but also through its moments, i.e. its mean and covariance.

In this thesis, we will consider the FIR model structure given in (2.14), (2.16) also for Bayesian estimation. Using the fact the logarithm is a strictly increasing function and thus does not affect the optimization, the Gaussian assumption and the FIR model structure for the predictor (2.7), we can write the posterior estimates as

$$\hat{\theta}_N^{apost} = \arg\max_\theta \log(p(Y_N|\theta)p(\theta)) = \arg\max_\theta \log((p(Y_N|\theta)) + \log(p(\theta)) \tag{3.6}$$

$$= \arg\min_\theta \sum_{t=n_g}^{N} (y[t] - \hat{y}[t|t-1;\theta])^2 + \theta^T P_n^{-1}\theta \tag{3.7}$$

$$= \arg\min_\theta \sum_{t=n_g}^{N} (y[t] - \varphi^T[t]\theta)^2 + \theta^T P_n^{-1}\theta \tag{3.8}$$

where the first term in the last equality of equation (3.6) is the maximum likelihood estimate defined as

$$\hat{\theta}_{ML} = \arg\max_\theta \log(p(Y_N|\theta)) \tag{3.9}$$

where for fixed Y_N, $\mathcal{L}_{Y_N}(\theta) = p(Y_N|\theta)$ represents the likelihood function. In the system identification literature [1], [2], the prediction error methods are also called maximum likelihood methods because of the equivalence between the two when the noise is Gaussian distributed and the data generating system is in the model set. The first term of the cost function (3.7) corresponds to the cost function used for prediction error methods. Thus the Bayes estimate could be seen as a penalized prediction error method. In fact, a penalized prediction error method is called a regularized prediction error

method [8], with $D_n = P_n^{-1}$ being called a regularization matrix. This matrix could be seen as a weighting matrix for the establishing the importance priority of the parameters when solving the optimization problem (3.7). Sometimes the covariance matrix P_n is be called a kernel. This is because the Bayes estimate, using a Gaussian prior, is shown to belong to a solution space which has the structure of a so called reproducing-kernel-Hibert-space (RKHS) defined by its covariance matrix/kernel P_n [12]. The RKHS is used in non-parameteric (function) estimation for obtaining smooth solutions. In our case, the FIR model is considered a non-parametric model since it represents the impulse response through a function of time.

Returning to equations (3.4) and (3.5), since θ and Y_N are jointly Gaussian distributed as

$$\begin{pmatrix} \theta \\ Y_N \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} P_n & P_n\Phi_N \\ \Phi_N^T P_n & \Phi_N^T P_n \Phi_N + \sigma^2 I_{N-n} \end{pmatrix} \right) \tag{3.10}$$

we deduce the posterior [8] to be

$$p(\theta|Y_N) \sim \mathcal{N}(\hat{\theta}_N^{apost}, P_N^{apost}), \tag{3.11}$$

$$\hat{\theta}_N^{apost} = (P_n\Phi_N\Phi_N^T + \sigma^2 I_{N-n})^{-1} P_n \Phi_N Y_N, \quad P_N^{apost} = P_n - P_n\Phi_N(\Phi_N^T P_n \Phi_N + \sigma^2 I_{N-n})^{-1} Y_N. \tag{3.12}$$

Notice that when the noise variance is zero the posterior mean of the parameters is given by the least squares solution given in equation (2.18). Using the formula above for the posterior computation leads to erroneous results when $\sigma^2 = 0$, and doesn't make use of the efficient numerical methods of solving least squares problems, as the kernel is still affecting the numerical results. In this report, we introduced and make use of another equivalent formula.

In the alternative formula

$$\hat{\theta}_N^{apost} = (R_N + \sigma^2 P_n^{-1})^{-1} R_N \hat{\theta}_N^{LS}, \qquad R_N = \Phi_N \Phi_N^T, \tag{3.13}$$

given in [8], with $\hat{\theta}_N^{LS}$ given in equation (2.18), if we apply the matrix inversion lemma

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1} \tag{3.14}$$

by making the following substitutions

$$A = R_N, C = \sigma^2 P_N^{-1}, U = I, V = I, \tag{3.15}$$

we obtain a new expression for the MAP estimator

$$\hat{\theta}_N^{apost} = \hat{\theta}_N^{LS} - \sigma^2(P_n R_N + \sigma^2 I)^{-1}\hat{\theta}_N^{LS}. \tag{3.16}$$

In this new form the regularized estimates are computed as an update of the least squares solution. When the noise variance is small we notice that the kernel matrix won't be affecting the estimate. Also when the noise variance is not small, the kernel matrix does not need to be inverted, which does represent a benefit for the kernel matrices which could become numerically singular (even though the kernel matrix should always be positive definite), such as the stable spline kernel presented in the next section.

As just explained, the system models are obtained using the Bayesian estimates for the model parameters given by the closed form solution (3.16). The covariance matrix P_n needed in calculations has to be determined though. The results in the literature [10] provide a choice for the covariance matrix P_n which automatically selects the model orders for the chosen model structure in an efficient manner. The procedure through which we can determine the P_n is provided in the next section.

3.1.1 Empirical Bayes

In the the previous section, we derived the solution for the Bayes estimate (3.1) when using Gaussian priors (3.5), to be given by formula (3.16). Thus, in order to compute the model parameters, the prior information has to be provided through the covariance P_n for the Gaussian distribution of the parameters (3.5). In practice, most of the time, the information of where the true parameters might be, expressed through a probabilistic model, is not readily available. One way of dealing with this problem is to pick a covariance matrix that has very large elements. This will not help very much the estimation though as we don't provide useful information about the parameters. It could be seen also from formula (3.7) that the regularization penalty will go to zero if the determinant of P_n becomes large. Thus, procuring a beneficial prior is an important element to obtaining improved estimation results.

In the literature [12], to overcome the inconvenience of not having a prior, the covariance matrix is given a fixed structure $P(\xi)$, parameterized in terms of a set of hyperparameters ξ. Then these hyperparameters are determined from data. They should be adjusted to improve the predictive ability of the model, by selecting the most appropriate covariance matrix from the parameterized class. This method is called Empirical Bayes.

The derivation of the hyperparameters comes from a statistical consideration that leads to a choice which is minimizing the original Bayes risk, i.e. the MSE, over the chosen class of priors. Since the output observations are distributed as (see formula (3.10))

$$Y_N \sim \mathcal{N}(0, \Phi_N^T P_n(\xi)\Phi_N + \sigma^2 I_{N-n}), \tag{3.17}$$

we can build the marginal likelihood [8] of Y_N given ξ as

$$\mathcal{L}_{Y_N}(\xi) = (\sqrt{2\pi}\det(V_N(\xi)))^{-1}\exp(Y_N^T V_N^{-1}(\xi)Y_N/2) \tag{3.18}$$

where $V_N(\xi) = \Phi_N^T P_n(\xi)\Phi_N + \sigma^2 I_{N-n}$. Then, to find the hyperparameters we solve the log marginal likelihood optimization problem

$$\hat{\xi} = \arg\min_{\xi} Y_N^T V_N^{-1}(\xi)Y_N + \log(\det(V_N(\xi))). \tag{3.19}$$

The term marginal comes from the fact that in the joint distribution (3.10), we marginalized out the parameters θ, from the joint distribution, to get the distribution of Y_N.

As a result, the prior in the Empirical Bayes approach is given by

$$\theta \sim \mathcal{N}(0, P_n(\hat{\xi})). \tag{3.20}$$

with $\hat{\xi}$ the solution to the marginal likelihood optimization problem (3.19).

3.1.2 System identification by stable splines

In the literature [8], [10], [12], there are different ways of choosing the covariance matrix (3.5), or (3.20), for benefiting impulse response models. In identification, we could use time functions to model the dynamics of a system through the impulse response in the convolution representation of the system (2.11). Such an approach to modeling is called non-parameteric. A specific application of regularization, in the literature [12], is for obtaining non-parametric models by requiring the solutions to have a certain degree of smoothness. In this case, we look for a function from a functional solution space. Regularization is used here to assign preference between the elements of the solution space by using prior information on the expected smoothness of the sought solution. The resulting restricted solution space forms a reproducing-kernel-Hilbert-space (RKHS) with a particular structure given by the kernel matrix [12]. Typical estimators are the smoothing splines such as the cubic smoothing splines, known to fit without bias linear functions [10]. However, because

in system identification usual stable impulse responses have an exponential decay, and as such there is an interest for obtaining estimators that are unbiased for exponential functions, the stable spline kernel was introduced [10]. The kernel obtained, i.e. P_n in formula (3.8), is

$$K_{n_g}^{ss}(k,j) = \frac{\lambda^2}{\sigma^2}\left(\frac{e^{-\beta(k+j)}e^{-\beta\max(k,j)}}{2} - \frac{e^{-3\beta\max(k,j)}}{6}\right), \quad i,k \in \{1,2,\ldots,n_g\}. \tag{3.21}$$

where the σ^2 is the output additive noise variance (see equation (3.4)), λ is a hyperparameter indicating the strength of regularization and β is a hyperparameter which captures the exponential decay of the stable impulse responses. Gathering together the hyperparameters, in the vector $\xi = \begin{pmatrix}\lambda & \beta\end{pmatrix}$, we will be using the Empirical Bayes to find them, as explained earlier. Also, σ^2 will be determined from an initial PEM estimate without regularization.

Conceptually, the stable spline models are regularized FIR models using the stable-spline regularization matrix (3.21). The exponential decay of the regularization matrix (3.21), provides an exponential increase in the penalty (3.8) for the FIR parameters. The largest penalty is put on the last FIR parameters corresponding to the tail of the impulse response, thus forcing them to zero, as wanted from stable impulse responses. In this way, starting with an FIR model with a large enough order, the effective model order could be seen as being selected automatically by setting the last FIR parameters to zero.

In this report we are using the stable-spline kernels for impulse response models, which besides improving the MSE for the estimates, also provide a way of automatically selecting the model order for obtaining predictive models. In the literature [10], evidence based on improvements in prediction is provided, favoring the use of stable-splines for avoiding model order selection in front of the classic PEM with model orders chosen based on methods of Akaike's information criterion type. It could be said that the hyperpameter optimization problem in the Empirical Bayes method, is the equivalent of model order selection for classic PEMs. Since the hyperparameters could be obtain from optimizing the marginal likelihood (3.19) using search methods, the Empirical Bayes approach to modeling could be seen as a method that automatically selects the model orders. Thus, to avoid the problem of model order selection we will be using stable-spline modeling for the modules of our networks.

3.2 Hyperparameter estimation for stable spline models

In this section, we are going to investigate how to obtain the hyperparameters for the stable-spline kernel (3.21), i.e. $\xi = \begin{pmatrix}\lambda & \beta\end{pmatrix}$, needed for obtaining the model parameter estimates (3.16). We have already established, that the hyperparameters will be estimated using the Empirical Bayes method, i.e. by optimizing the marginal likelihood (3.19), for the stable-spline kernel (3.21). The difficulty with this method is that the marginal likelihood (3.19) is not convex, no matter the choice of covariance matrix used. Thus, there are no guarantees that, using convex optimization tools, we would find the optimal hyperparameters, as the solver might give a local solution. We could ensure that we would find the global solution, if we could initialize the optimization routine in the basin of attraction of the global optimum. In general, such a step requires additional information that is not readily available. In our situation though, we know that the β hyperparameter for the stable-spline kernel (3.21) is meant to capture the rate decay of the impulse response of our stable system. We thus speculate, that the β hyperparameter could be related to the dominant pole of our system and therefore an initial guess of β could be derived from an estimate of the dominant pole. Consequently, we are going to investigate the appropriateness of such a choice of an initial guess.

In [7] it was shown, through simulation, that for a simplified version of the stable spline kernel, namely a diagonal exponential decaying kernel (see the diagonal kernel in [8]), the decay rate converges, under suitable assumptions, to two times the absolute value of the dominant pole (or to the square absolute value of the dominant pole, if a discrete version of the stable-spline kernel (3.21) is

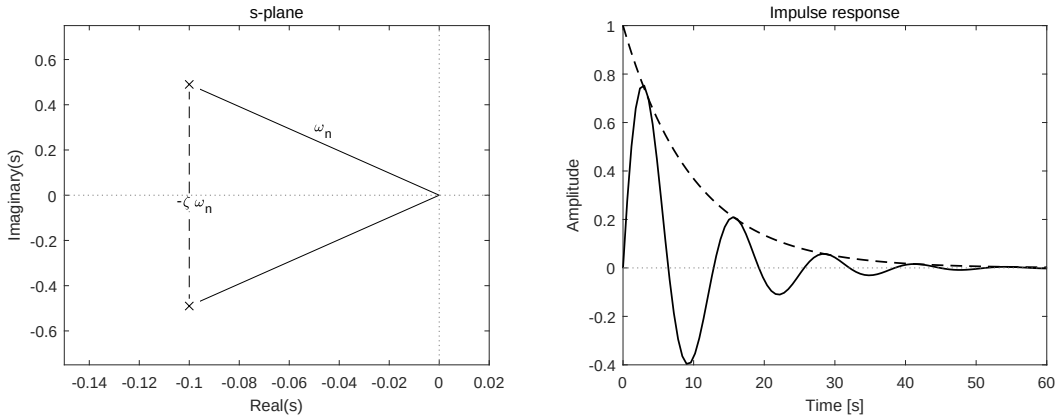


Figure 3.1: Pole location of the second order system in the s-plane (left), and the impulse response with continuous line (right). The dashed exponential in the right figure is the response of a first order system with pole at the dominant pole of the second order system.

used). A proof of the statement is not provided therein. We will try to show, through an example, how does a rate decay chosen as such compare with the rate decay chosen as the absolute value of the dominant pole.

Firstly, we define the dominant pole of a continuous time system to be the real part of the pole(s) with closest real part to the origin and the dominant pole of a discrete time system to be the absolute value of the pole with closest absolute value to the unit circle. For a first order system the dominant pole is also the only pole of the system while for a second order underdamped system the dominant pole of the system gives the envelope that bounds the impulse response as seen figure (3.1). Also for a continuous time, second order, underdamped ($0 < \zeta < 1$) system

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n + \omega_n^2} \tag{3.22}$$

the dominant pole is given by $-\zeta\omega_n$, as indicated in the left plot of figure (3.1).

Another argument for using the dominant pole in deriving the kernel rate decay is as follows. Every transfer function can be expressed using partial fraction expansions as a sum of first or second order transfer functions. By the linearity property of the system, the total impulse response is given by the sumation of individual impulse responses corresponding to the partial fraction expansion components. Then, the general decay trend of the systems is given by the dominant pole, if the impulse response of the partial fraction expansion component corresponding to the dominant pole has a large enough signal energy that would not make it insignificant to the total impulse response. The implication for regularization is that, on one hand, a kernel rate decay which is noticeably slower than the dominant pole, will not provide an appropriate regularization penalty as the resulting weight between the parameters, c.f. (3.8), will not force the tail of the impulse response to zero sufficiently fast. On the other hand, a kernel rate decay which is noticeably faster than the dominant pole, will not provide an appropriate regularization penalty as the resulting weight between the parameters will force the tail of the impulse response to zero faster than needed. Thus, in general, the dominant pole would be a good indicator of the kernel rate decay β, as it embeds information about the weighting between the impulse response parameters.

We will try to investigate, through experiments, whether a kernel rate decay choice based on the dominant pole of the system is truly providing the good results, in terms of maximizing the marginal likelihood, for stable spline identification. In the next example, we will compare the hyperparameter choice $\beta = 2|p_1|$ from [7] with $\beta = |p_1|$, where p_1 is the dominant pole of the system. Through

this example, we also illustrate the appropriateness of using the dominant pole for determining the stable-spline hyperparameters.

Example 3.2.1. Let's consider a continuous time, first order system with the pole at $p_1 = -0.2$, i.e. $G(s) = 1/(s + 0.2)$. Input-ouput data is generated from a zero-order-hold discretization of $G(s)$ with a sampling time $T_s = 2$ and output additive noise. We have chosen to do the simulations in the presence of output noise because this would resemble a practical application and because when the noise variance is zero the MSE estimator would give the same results as the least squares estimate and thus it does not make sense, in that case, to use regularization, as indicated also in formula (3.16). For the choice of kernel rate decay $\beta = 2|p_1| = 0.4$, and assumed known noise variance $\sigma_2 = 0.25$, we determine the regularization factor, by optimizing the marginal likelihood, to be $\lambda^2 = 65$, giving a marginal likelihood cost of $J_{\beta=2|p_1|} = 169$. Using a grid of 100 points between $[0, 0.6]$, for the β, we determine for each the corresponding optimal λ^2 as before. The results obtained are shown in Figure 3.2. There, we also indicate the point where $\beta = |p_1|$ giving $\lambda^2 = 4.2$, $J_{\beta=2|p_1|} = 160$, for comparison with $\beta = 2|p_1|$. It appears that $\beta = |p_1|$ would be a better choice than $\beta = 2|p_1|$. With the optimal point over the grid at $J_{opt} = 159.7$, $\beta = 0.224$, the following ratios are obtained

$$1 - \frac{|J_{\beta=2|p_1|} - J_{opt}|}{|J_{opt}|} = 0.944, \qquad 1 - \frac{|J_{\beta=|p_1|} - J_{opt}|}{|J_{opt}|} = 0.998, \tag{3.23}$$

which indicate the relative closeness to the optimal point in favor of $\beta = |p_1|$. Notice that the marginal likelihood cost for $\beta = 2|p_1|$ is the same as when $\beta = 0.09$, and lower in between $(0.09, 0.4)$, which might be considered a large interval for the expected choice of the kernel rate decay rate.

Another important aspect in choosing β is the corresponding optimal value for λ^2. Comparing the corresponding λ^2 for the two choices of β, namely $\beta = |p_1|$ and $\beta = 2|p_1|$, we see that λ^2 is smaller for $\beta = |p_1|$, which is closer to the optimal β. A smaller λ^2 corresponds to a stronger need for regularization, indicating possibly a larger contribution of the prior in the derivation of the estimated model. So, for this reason, a $(\beta; \lambda^2)$ pair with smaller values for λ^2 will indicate a possibly better hyperparameter pair in terms of regularization as it tries to use more the information from the prior in constructing the estimated model, or in other words the smaller the λ^2, the more effective the regularization.

The lower graphs in Figure 3.2 show the marginal likelihood cost and the λ^2 over a larger range of β. These plots are suggestive for the limiting behavior, when β would be chosen large. Notice that when $\beta \gg \beta^{opt}$, then the marginal likelihood cost deteriorates even further, and the corresponding optimally chosen λ^2 also grows in amplitude. The irregular behavior in the λ^2 when $\beta \in (1, 3)$ can be explained by the fact that for $\lambda^2 > 10^6$ the regularization penalty is so small that the optimally chosen λ^2 by optimizing the marginal likelihood for the given β becomes dependent on the data noise.

We repeat the same experiment for 50 realizations of the output noise and compute the $25 - 50 - 75$ percentiles. The results are shown in Figure 3.3, were we see that even for multiple experiments the choice of $\beta = |p_1|$ comes close to the global optimum. The shaded area represents the area between the 25 and 75 percentile, while the thick middle line is indicating the 50 percentile. In Figure 3.4 we see the contour plot of the marginal likelihood as a function of (β, λ^2) for one realization of the noise and a grid over the space of (β, λ^2). It appears that, in the region visible, we have only one minima which is near our choice of $\beta = |p_1|$.

The procedure for building Figure 3.2 reveals a reasoning useful in deciding the optimal regularization parameters. The reasoning is that it is more revealing to grid over the β and optimize over the λ^2 for the chosen grid, than do the other way round. This is because the β hyperparameter is a static parameter that does not change with the experimental conditions such as number of data points and noise variance. Its role is to embed the structural properties of the data generating system, i.e. the impulse response decay rate, into the prior. On the other hand the role of λ^2 parameter is to

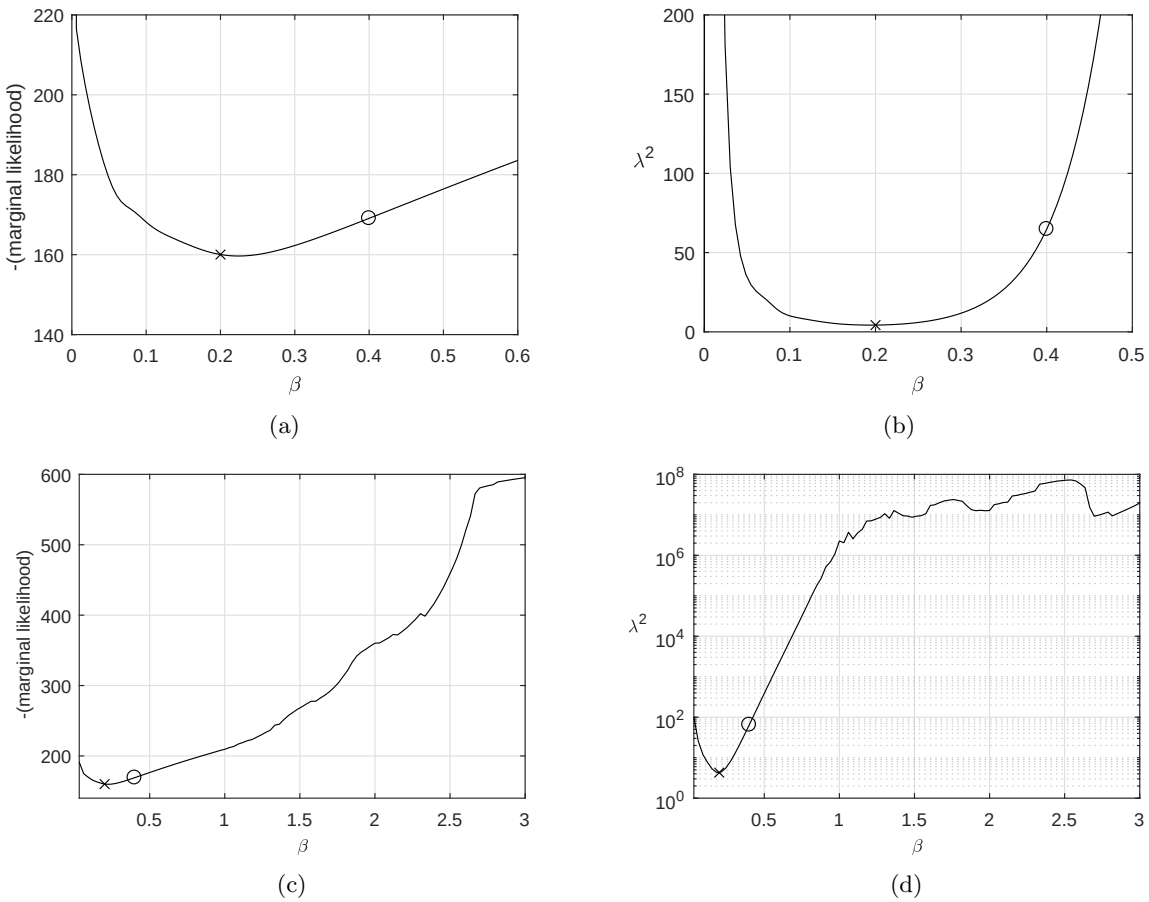


Figure 3.2: The marginal likelihood cost (left) for the $(\beta; \lambda^2)$ pair in the right figure. The circle indicates the point where $\beta = 2|p_1|$, while the cross indicates the point where $\beta = |p_1|$. The upper figures are zoomed in versions of the bottom ones.

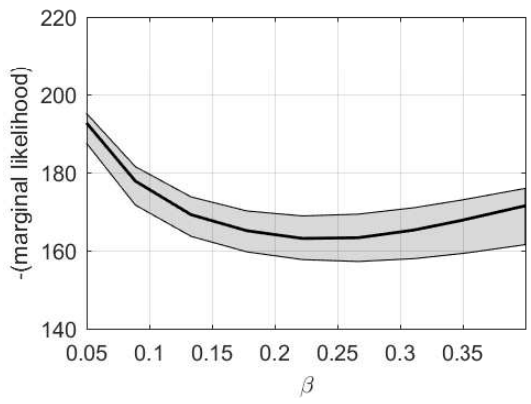


Figure 3.3: Monte Carlo simulation for calculating the marginal likelihood for a grid of β points and optimally chosen λ^2, for different noise realizations. The shaded region indicates the $25-50-75$ percentiles.

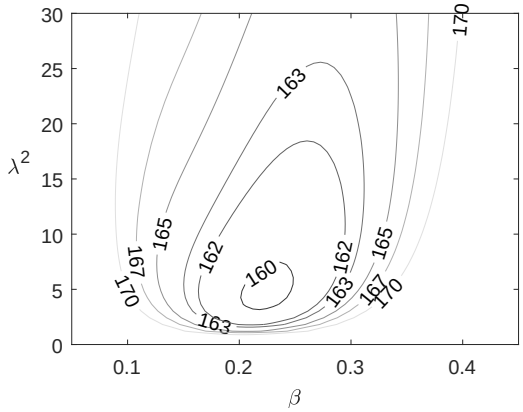


Figure 3.4: Contour plot of the -(marginal likelihood) as function of β and λ^2.

assess the amount by which the prior should affect the estimate and it does depend on experimental factors such as the number of data points and noise variance. Since the β depends mostly on the data generating system and should not change based on the experiments, it is much easier to choose a set of possible candidates for it, from which the optimal β is part, i.e. it is more intuitive to choose a griding over the β in the region we expect the optimal β to be, than to do the same for λ^2.

The previous observation leads to the following optimization strategy based on the gradient decent method.

Result: Find optimal hyperparameter values: (β, λ^2)
Initialization: β_0, λ_0, $k = 1$;
while *stopping condition is not satisfied* **do**
  make a gradient descent step: $\beta_k = \beta_{k-1} + \gamma \nabla J_1(\beta_k)$;
  where $J_1(\beta_k) = \min_{\lambda_k^2} J_2(\beta_{k-1}, \lambda_k^2)$, and γ is indicating here the step size;
  $k = k + 1$;
end

We try to optimize over the β, by calculating the marginal likelihood point-wise using the optimal λ^2 for that β. Notice that in the minimization of J_2, β_{k-1} is fixed, while when evaluating J_1 , λ^2 is fixed at the optimal value determined. The gradient could be calculated numerically by using the function evaluation of $J_1(\beta_k)$ near the point where the gradient needs to be evaluated. Now, if we happen to always initialize the optimization in the basin of attraction where the global optimum is found, then a direct application of the gradient methods could be done to obtain the global minimum.

In this example, we compared two β hyperparameter choices based on the dominant pole location of the system, namely we choose $\beta = 2|p_1|$ as was suggested in paper [7] and $\beta = |p_1|$. For the simplest system of only one pole, we compared the two options through the value of the marginal likelihood, revealing that $\beta = |p_1|$ gives a lower cost and thus a better result, actually the best result of all possible choices as it gives the actual minus marginal likelihood minimum. From the procedure used for the comparison, we have also noticed that the different roles of the hyperparameters λ and β could be used when optimizing the marginal likelihood. In this regard, an algorithm based on gradient descent was suggested, were the marginal likelihood is to be optimized with respect to β by finding the optimal λ for each β evaluation. The $\beta = |p_1|$ initialization choice and this algorithm might help in finding the global minimum of the non-convex marginal likelihood, as it did for our example.

3.2.1 Estimation of the dominant pole

We have shown, in the previous section, though an example, that the dominant pole of the system offers valuable insight in the location of the hyperparameter β. In practice, the location of the dominant pole of a system is usually unknown and has to be determined from data as well. One way of estimating the dominant pole from data is by modeling the data generating system, and afterwards selecting the dominant pole from the obtained model. Thus, we can consider the output-error model structure given by (2.19), (2.20) and use the prediction error method to obtain suitable model parameters from data by solving (2.9). Then, the dominant pole estimate will be given by the absolute value of the model pole(s) with closest absolute value to the unit circle, i.e.

$$\hat{p}_1 = \min_{p \in S_p} |1 - |p||, \tag{3.24}$$

where S_p is the set of all poles of the estimated output-error model. The limitation of this method is that it requires the output-error model orders to be specified. Yet, the whole purpose of estimating the dominant pole was for the its use in constructing stable-spline models, to avoid the problem of model order selection. This, illustrates the difficulty that we have in estimating the dominant pole, namely that we have to avoid the problem of model order selection. A solution to this inconvenience is provided next.

The approach that we take in this thesis, to estimate, from data, the dominant pole of a single-input single-output system, follows from performing a combination of approximations and applying realization theory to move from one representation to another. The method is summarized through the next steps:

1. Derive from data an FIR model $G^{FIR}(q^{-1};\hat{\theta})$ ((2.11)), calculating its parameters $\hat{\theta}$ using the prediction error method, given through the closed form solution (2.18).

2. Reduce the order of the obtained FIR model $G^{FIR}(q^{-1};\hat{\theta})$, by using the method of balanced truncation [3], to obtain a new model $G^{red}(q^{-1})$. Namely, we are going to truncate the state-space representation in balanced form, of $G^{FIR}(q^{-1};\hat{\theta})$, by choosing the truncation order based on the Hankel singular values. We then transform back the state-space representation of the reduced system to the transfer function form.

3. Select the dominant pole, according to (3.24), from the reduced model $G^{red}(q^{-1})$.

The idea behind this procedure comes from the FIR model representation ((2.11)), which can be expressed as

$$G(z) = \frac{g_{n_g-1} + g_{n_g-2}z^1 + \cdots + g_1 z^{n_g-1}}{z^{n_g-1}} \tag{3.25}$$

where z is a complex variable. Such a representation has n poles at zero and n zeros given by the impulse response coefficients. Notice that its dominant pole seems to be at zero even though such a representation could be a good approximation of a first order system with any stable pole location. To recover the original dominant pole of the system from the approximation, another approximation could be made on the FIR model by using model reduction methods. Thus, we will try to find a more simple model structure, in terms of number of poles and zeros, for the estimated FIR model, that will move the poles of the system away from zero.

This method of obtaining the dominant pole of a system could also be considered as a method of determining the model orders for a system. The results depend on choosing an appropriate truncation order. It is a less rigorous method for this purpose though, as the FIR model is an estimate affected by noise which might create difficulty in selecting the best model orders.

We will illustrate how the method performs on two practical examples: one in which all the poles are real, and another in which complex poles are present.

Example 3.2.2. Let's first consider a transfer functions with only real poles and zeros

$$G(s) = \frac{1}{s+0.2} + \frac{1}{s+0.3} + \frac{1}{s+0.4} + \frac{1}{s+1.7} + \frac{1}{s+2.3} \tag{3.26}$$

which is discretized with sampling period $T_s = 2$. Input-output data is obtained from the discretized system under the effect of output noise and FIR identification using PEM is done to obtain an estimate of it. Then, balanced truncation is performed to obtain a reduced order model that captures the dynamics due to the first most influential balanced states. Figure 3.5 shows the Hankel singular values for the discrete system as well as for its FIR estimate. Even though the data generating system has 5 poles, thus 5 minimal states, most of the state energy is concentrated in the first one. The estimated FIR model presents additional state contributions because of the noise that affected the estimate. A reduced first order model of the FIR estimate offers a good approximation of the original system as is demonstrated in Figure 3.6 and in Figure 3.8. The frequency response function indicates an small (possibly insignificant) bias in the low frequencies, where the dominant pole is, while some more considerate bias is noticed at higher frequencies, where the faster poles are. In practice, if there is distrust in the truncation order, we can use a cross-validation procedure to decide whether the reduced order model can predict the validation data set.

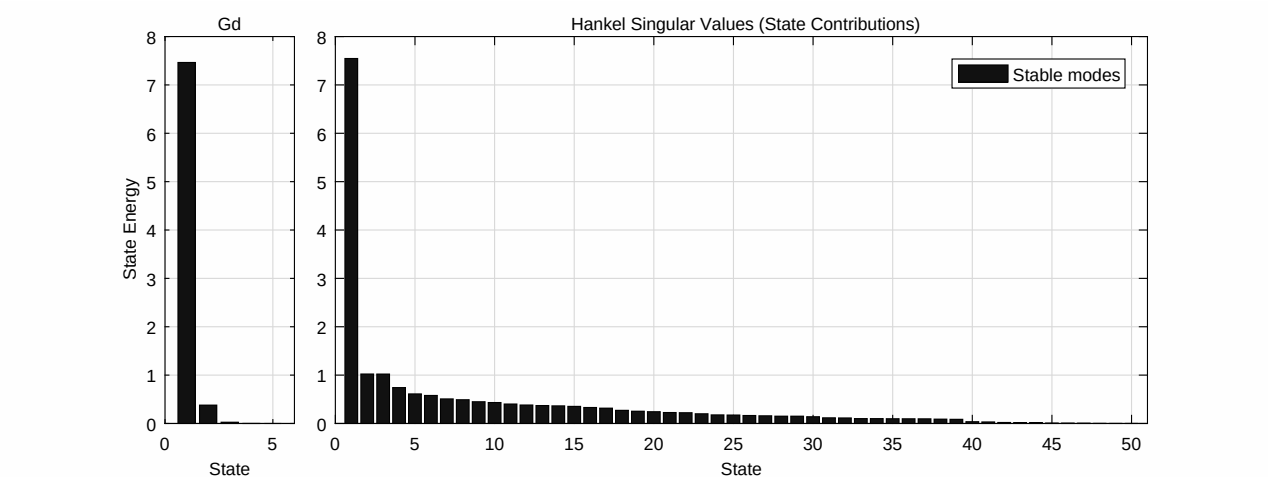


Figure 3.5: Hankel singular values of the data generating system (left) and of the FIR estimate (right).

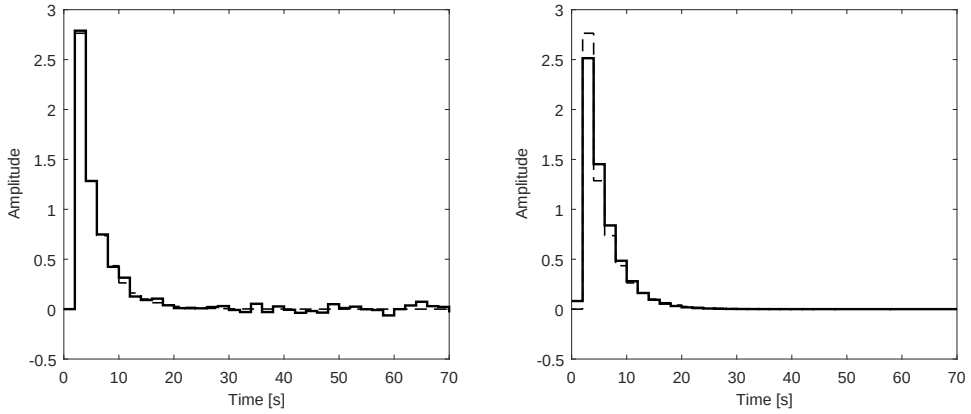


Figure 3.6: Impulse responses of the FIR estimate (left) and of the reduced order model (right). The impulse response of the true system is shown with dashed lines.

Inspecting the marginal likelihood cost in Figure 3.7 we notice that the estimated dominant pole is close to the value of the true dominant pole but we also notice that the dominant pole is no longer precisely indicating the minimum. This is due to the effect of the additional dynamics, thus indicating that the additional dynamics also influences the location of the optimal rate decay.

The results in this example demonstrate that a reduced order model does offer good results in estimating the dominant pole of the system and that a first order model is a good approximation in this case. From more extensive simulations it was noticed that a large class of high order transfer functions with real poles only can be finely approximated using first order models.

Example 3.2.3. In this example we consider a transfer function which has complex conjugate poles

$$G(s) = \frac{1}{s+0.2} + \frac{0.32}{s^2+0.12s+0.16} + \frac{0.72}{s^2+0.12s+0.36} \tag{3.27}$$

and we are going to see how well is the dominant pole estimated by repeating the simulations done in the preceding example. As before we use the discretized model as the data generating system and we identify an FIR model from the input output data, from which a reduced order model is

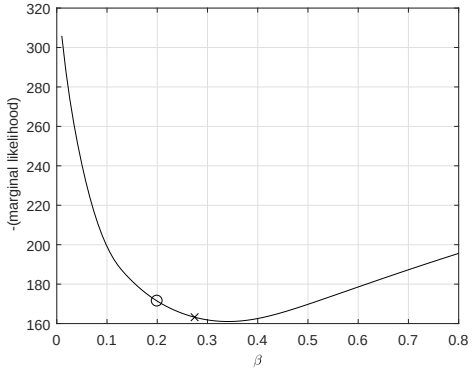


Figure 3.7: Marginal likelihood for a grid of β's and optimally chosen λ^2. The circle indicates the dominant pole of the true system, while the cross indicates the dominant pole of the reduced system.

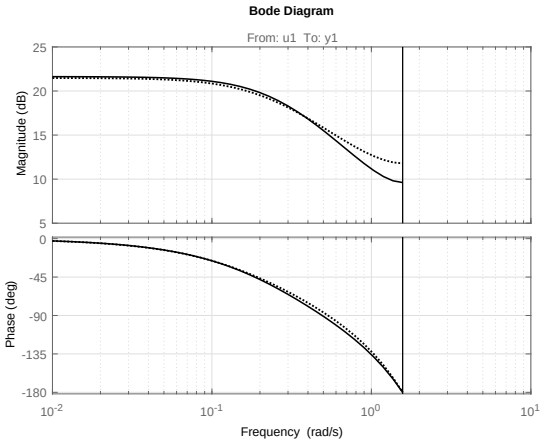


Figure 3.8: Bode diagram of the true system (dashed line) and of the reduced order system (continous line).

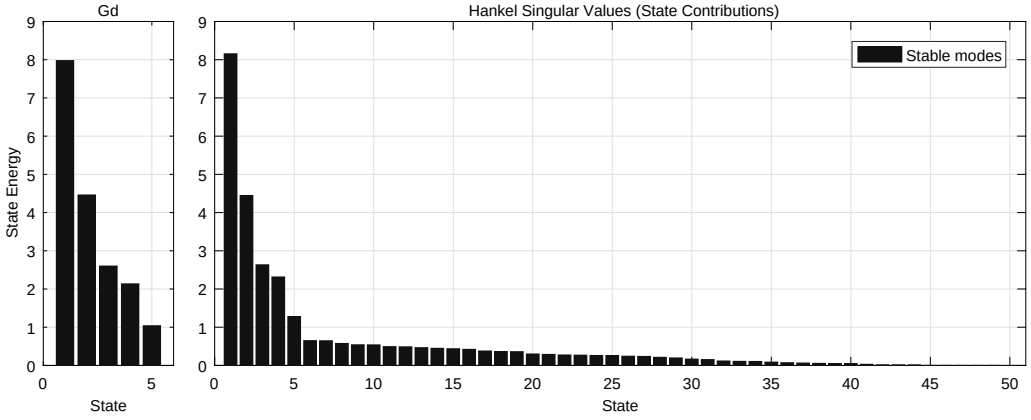


Figure 3.9: Hankel singular values of the data generating system (left) and of the FIR estimate (right).

obtained. In this case the data generating system has five significant states as the can be seen in the Hankel singular value plots in Figure 3.9 and in the impulse response (see Figure 3.11), which no longer can be approximated with sufficient accuracy by a first order model. The position of the estimated dominant pole is close to the true dominant pole as illustrated in figure 3.10. In the figure we notice the irregular shape of the marginal likelihood is present even for small values of the rate decay $\beta > 0.3$, which is accompanied by large values for λ^2 and a irregular shape as well. Also at the optimal point of the marginal likelihood, seen in the figure, the corresponding optimal λ^2 is large. This could be due to the fact that the inpulse response does not have an exponential decaying shape (c.f. see the Hankel singular values), and thus it would indicate that regularization in this case would not be so effective. In fact, the Hankel singular values could be a first indicator of the usefulness of the stable spline regularization. If the initial point of β would be chosen greater than 0.3, the optimization might return a local optimum. On the other hand, if β is chosen very small, because of the steep gradient, the optimization step performed might take the β outside of the global region of attraction. Thus a good initialization as the one provided would be beneficial to avoid such situations.

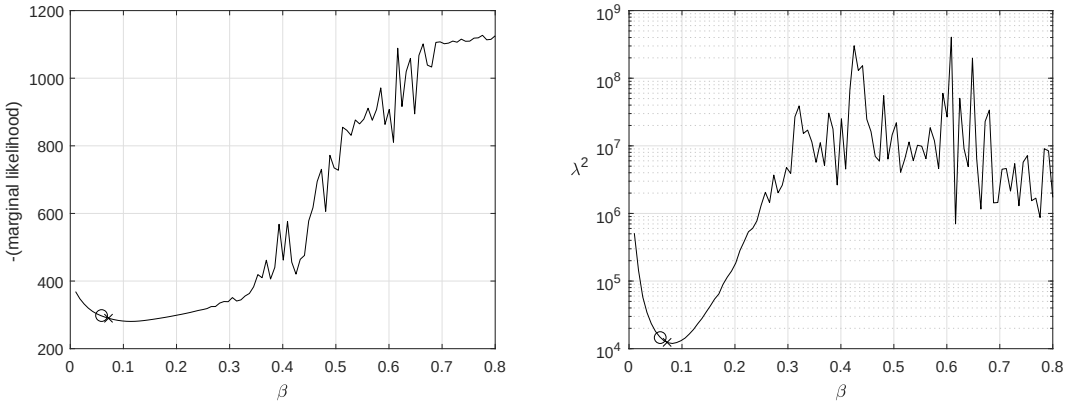


Figure 3.10: Marginal likelihood (left) for (λ^2, β) pair in the right figure. The circle indicates the dominant pole of the true system, while the cross indicates the dominant pole of the reduced system.

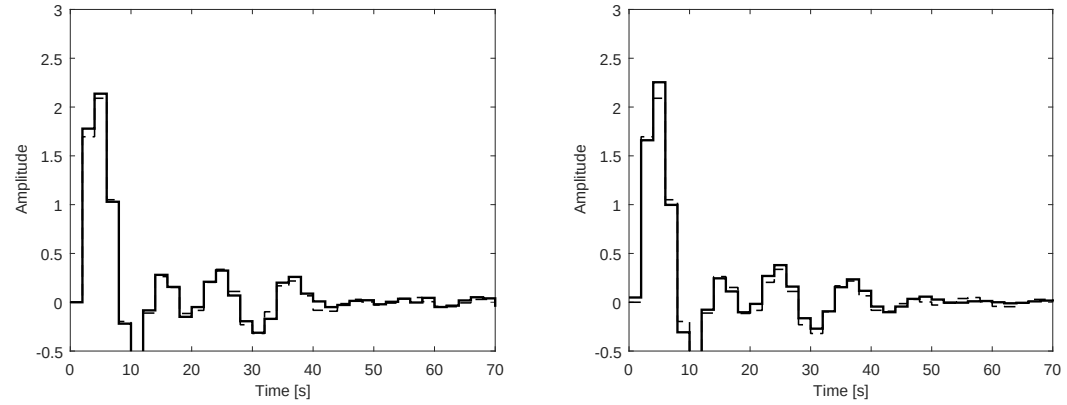


Figure 3.11: Impulse responses of the FIR estimate (left) and of the reduced order model (right). The impulse response of the true system is shown with dashed lines.

3.2.2 First order approximation of the kernel hyperparameter

An alternative way of choosing the kernel rate decay β could be to consider a first order output-error approximation of the system. Thus, we try to model our the data generating system using the output-error model structure $G^{OE}(q^{-1}; \hat{\theta})$ from formulas (2.19), (2.20) for which we fix the model orders to $n_b = 1$ and $n_f = 1$. The model parameters $\hat{\theta}$ are estimated from data by minimizing the prediction error as indicated in formula (2.9). Then, the kernel rate decay β is obtained from the only pole of the estimated output-error model $G^{OE}(q^{-1}; \hat{\theta})$.

With respect to the chosen approach an interesting question would be whether a first order approximation of the transfer function gives the minimal mean-square-error, with respect to the impulse response, out of all first order models. In other words, we want to verify whether $\arg\min_\theta \frac{1}{N}\sum_{t=1}^{N}(y[t] - \hat{y}[t;\theta])^2 = \arg\min_\theta \frac{1}{N}\sum_{t=1}^{N}(h[t] - \hat{h}[t;\theta])^2$, where $\hat{h}[t;\theta]$ is the impulse response of a first order model with parameters θ , and the model structure, used to make predictions, comprises of first order OE models.

Using the convolution representation of the output signals

$$y[t] = h[t] * u[t] + e[t], \qquad \hat{y}[t;\theta] = \hat{h}[t;\theta] * u[t], \tag{3.28}$$

and the distributivity of the convolution operation, we can write

$$\arg\min_{\theta} \frac{1}{N}\sum_{t=1}^{N}(y[t]-\hat{y}[t;\theta])^2 = \arg\min_{\theta} \frac{1}{N}\sum_{t=1}^{N}((h[t]-\hat{h}[t;\theta]) * u[t]+e[t])^2. \tag{3.29}$$

Notice that the input data $u[t]$ acts here as a weight adjusting the relevance of the components of the impulse response $h[t]$ to the total cost. It is known that, asymptotically,

$$\theta^* = \arg\min_{\theta} \bar{\mathbb{E}}\varepsilon^2[t;\theta] = \arg\min_{\theta} \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi_{\varepsilon}(\omega)d\omega, \tag{3.30}$$

where $\bar{\mathbb{E}}\varepsilon^2[t;\theta] = \lim_{N\to\infty}\frac{1}{N}\sum_{t=1}^{N}\mathbb{E}\varepsilon^2[t;\theta]$ is the expectation of the quasi-stationary process $\varepsilon^2[t;\theta]$, and $\Phi_{\varepsilon}(\omega)$ is the power spectral density of $\varepsilon[t;\theta]$. For our choice of first order OE model structure, the above expression can be written as

$$\theta^* = \arg\min_{\theta} \frac{1}{2\pi}\int_{-\pi}^{\pi} |G_0(e^{i\omega}) - \hat{G}(e^{i\omega};\theta)|^2\Phi_u(\omega)d\omega. \tag{3.31}$$

Here, $G_0(e^{i\omega})$ is the frequency response function corresponding to the true impulse response $h[t]$, while $\hat{G}(e^{i\omega};\theta)$ is the frequency response function corresponding to the impulse response of our first order OE model $\hat{h}[t;\theta]$. If $u[t]$ is white noise then it has a flat spectrum and the estimated parameters are chosen such that the sought first order OE model, or it's impulse response, approximates the best, in the frequency domain, the frequency response function of the data generating system. This result increases our motivation in using the first order output-error approximation as the a means for getting a initial kernel rate decay, since the pole of the estimated model captures the overall decay trend in the impulse response.

In the following example, we investigate how is the rate decay, chosen based on a first order approximation, performing in terms of maximizing the marginal likelihood (3.19).

Example 3.2.4. Consider the transfer function of a continuous time system

$$G(s) = \frac{1}{(s+0.2)(s^2+0.2s+1)} \tag{3.32}$$

from which we obtained the discretized data generating system using the zero-order hold discretization with sampling time $T_s = 2$. Identification data is obtained from the system under output additive noise of variance 0.09, using white noise input of variance 1. Based on 500 input-output data points collected with a sampling time of $T_s = 2$, a first order output-error model is estimated, the result being shown in figure (3.12). As before, a Monte Carlo simulation, depicted in figure (3.12), shows how is the β, chosen based on the estimated model, comparing with other β's over a grid, in terms of marginal likelihood cost. As can be seen, the use of a first order model provides a β not exactly at the optimum but very close to it.

3.2.3 Comparison between the methods

We saw earlier that both the dominant pole and the first order approximation are good candidates for initializing the optimization of the marginal likelihood (3.19). It is questionable at this point though, which of the two choices is better for use as a kernel rate decay. In the following example we will try to compare the two choices through an example.

Example 3.2.5. Let's consider the impulse responses studied in [7],

$$h_1[t] = 0.7^t + 3\cdot 0.3^t, \qquad h_2[t] = 0.7^t - 3\cdot 0.3^t, \qquad h_1[t] = 0.7^t\cos(\pi t/3),\ \ t\in\mathbb{N}. \tag{3.33}$$

For comparison, shown in Figure 3.13 are the true impulse responses along with the first order output-error approximation as well as the impulse response of first order models with the poles at

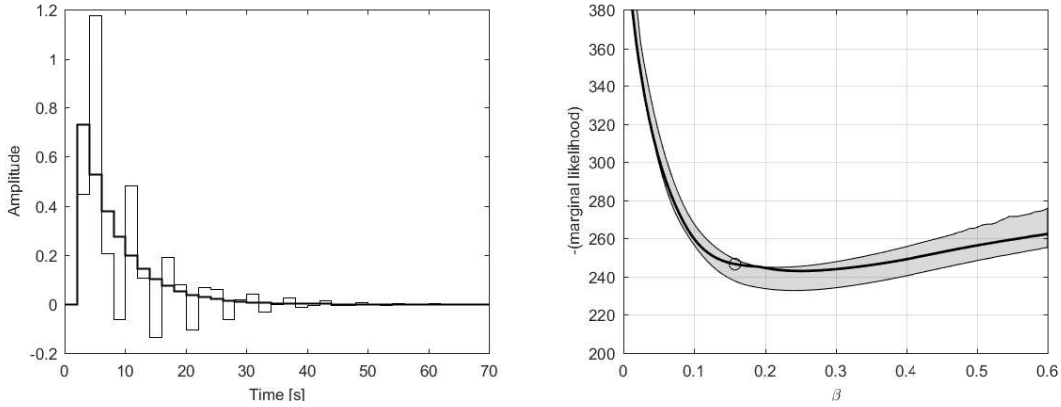


Figure 3.12: a) The impulse response of the data generating system and of the first order OE approximation (thick line). b) Monte Carlo simulation for different noise realizations, calculating the marginal likelihood over a grid of β's. The circle indicates the β selected through first order OE modeling. The shaded region indicates the $25 - 50 - 75$ percentiles.

the dominant poles of the true systems. The gain of the first order approximation with the pole at the dominant pole of the system was choosen to match the gain of the dominant pole in the impulse response representation, namely it was choosen 1 for all three cases. The Kronecker delta impulse, used as input for calculating the impulse response, is of length $T_s = 2$ and height $1/T_s$. From the figure, we see that the first order output-error approximation offers a good indication of the rate decay for the first impulse response, but performs slightly worse in capturing the decay of the second and third systems. For the second system, the first order output-error approximation gives a faster rate decay, while for the third system, which is second order with complex conjugate poles, it offers a slower rate decay than that of the true impulse response. On the other hand, the first order impulse responses using the dominant pole of the systems provides a rate decay that follows better tail of the true impulse response in all three cases.

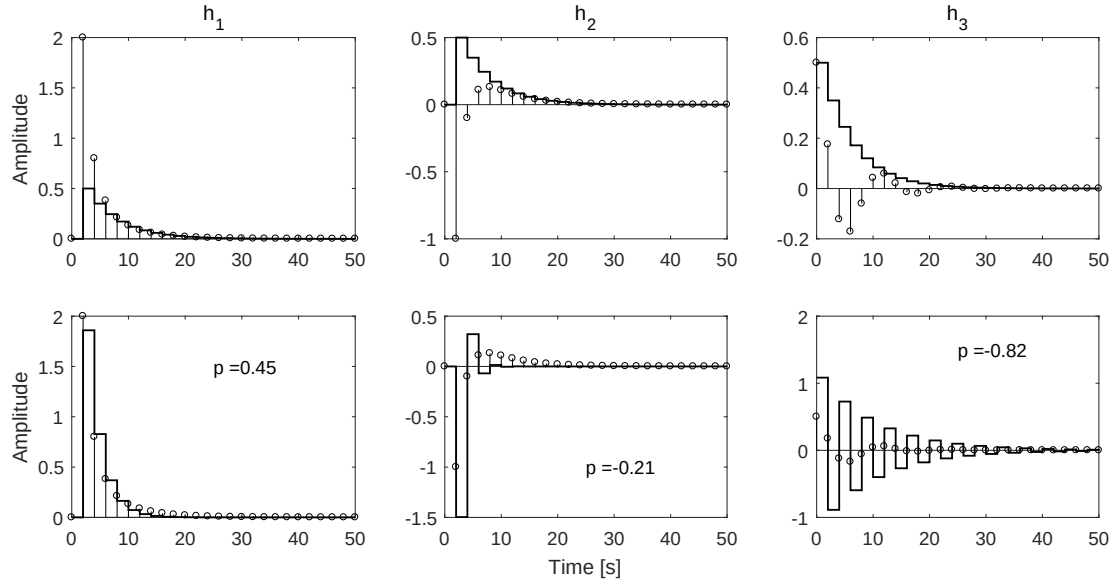


Figure 3.13: Impulse responses of the true systems (stem), their first order OE approximations (bottom plots, thick line) and the dominant pole (top plots, thick line). The p represents the estimated pole location.

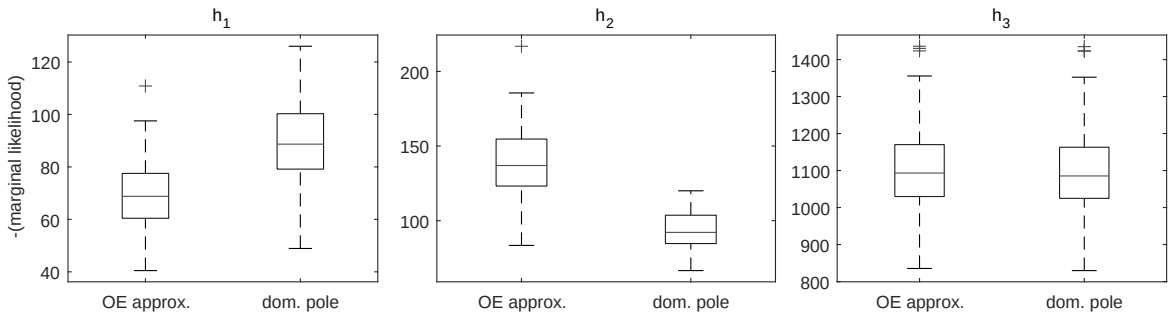


Figure 3.14: 50 Monte Carlo simulations indicating the marginal likelihood cost comparison between the two choices of β.

From these results we deduce that, apparently, the dominant pole of the system does give a slightly better approximation of the impulse response decay than the first order output-error approximation.

Figure 3.14 shows box plots indicating the marginal likelihood for 50 Monte Carlo simulations with different noise realizations for the three cases. For the first impulse response, we see that the output-error approximation offers the better cost, while for the second case, the dominant pole of the true system gives a better result. In the third situation, both give similar results. From Figures 3.13, 3.14 we thus deduce that, overall, both choices of kernel decay rate perform well and there is no definitive choice favoring one over the other. On the other hand, it has to be reminded that finding a first order output-error approximation involves solving a non-convex optimization problem, whereas estimating the dominant pole through the proposed method in this section, does not. In fact, the estimated dominant pole could serve as an initialization in finding the first order output-error approximation.

3.3 Summary and conclusions

In the present chapter, we have addressed the problem of model order selection, for a single-input single-output system, through the use of a Bayesian approach to dynamic modeling. The problem is essential for the topology detection approach expressed in the previous chapter. The solution presented, which was introduced in [10], is to use stable-spline models which use as prior information the decaying trend of the impulse response of stable systems, in an Empirical Bayes approach to non-parameteric modeling. Whereas the modeled order selection is avoided in this way, the new procedure comes with its own difficulty, namely in finding an appropriate prior, or more specifically, appropriate hyperparameters. The solution is to optimize a marginal likelihood function which is non-convex. To avoid obtaining local solutions, two initial choices for the kernel rate decay were indicated, namely one through the estimate of the dominant pole of the system and the other though a first order output-error estimate. The two options were evaluated and compared through experiments. The different role of the hyperparameters λ and β indicated the strategy of evaluating the two initial value options for β. The results revealed that both methods give accurate values of the kernel rate decay, close to the global solution, and based on the cost evaluation no preference is set for any of the two. The preference is mostly specified by the fact that the solution to a first-order output-error estimate involves a non-convex problem whereas the other option does not.

Chapter 4

Topology detection through regularized network modeling

4.1 Topology detection evaluation

In this section we will explore how to evaluate whether an estimator is good at detecting the topology of a dynamic network from measurement data. We will try to determine how to compare the topology detection results of two estimators, or in other words, how do we measure which of the two determined topologies is closer to the true topology and what would be measures of closeness between topologies.

In the literature on evaluation of binary classifiers, a common method of measuring whether a given result has been labeled correctly is through the confusion table [18]. By setting our topology detection problem in a binary classification format we could use the following table to compare two estimated topologies.

		True topology	
		Connection present	Connection absent
Estimated topology	Connection present detected	True positive	False positive
	Connection absent detected	False negative	True negative
		Sensitivity	Specificity

Table 4.1: Confusion table for topology detection.

True positive (TP) indicates the number of connections present in the data generating network that were detected as being present. False negative (FN) is the number of connections actually present but detected as absent. False positive (FP) indicates the number of connections absent in the data generating network that were detected as being present, and True negative (TN) is the number of absent connections in the true network that were determined as being present.

Two of the measures used to evaluate binary classifiers are the sensitivity and the specificity. The sensitivity (or true positive rate) indicates the proportion of the actual present connections that are detected correctly, while the specificity (or true negative rate) indicates the proportion of actual

		True topology	
		Connection present	Connection absent
Estimated topology	Connection present detected	TP = 4	FP = 1
	Connection absent detected	FN = 0	TN = 15
		Sensitivity = 1	Specificity = 0.94

Table 4.2: Confusion table for topology detection in example 4.1.1.

absent connections that are detected correctly. These two measure are calculated as

$$TPR = TP/P = TP/(TP + FN) \qquad TNR = TN/N = TN/(TN + FP) \tag{4.1}$$

where P is the total number of present connection of the data generating system, also called the number of positives, while N is the total number of absent connections of the data generating system, also called the number of negatives.

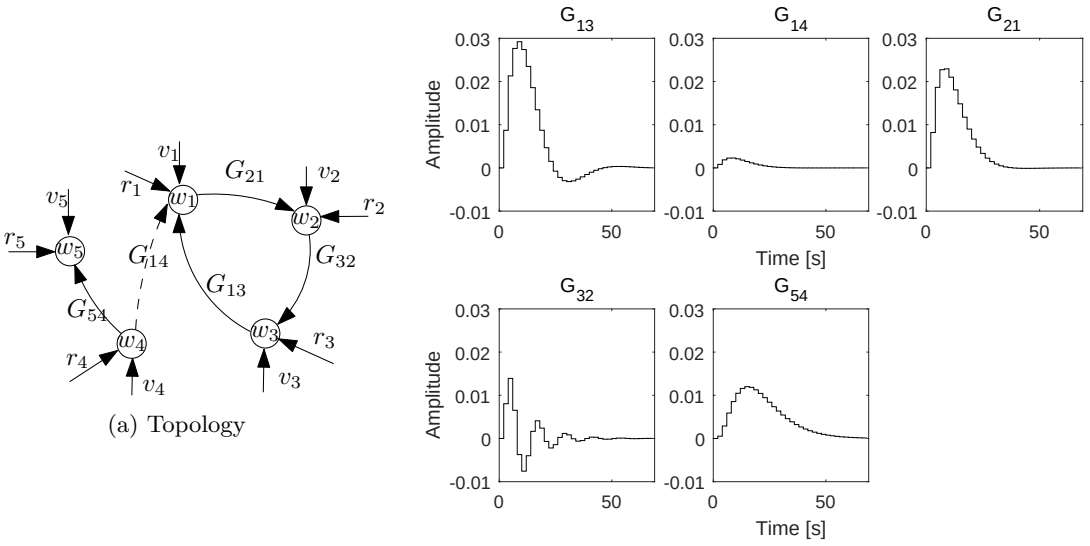
Example 4.1.1. We consider an example where the topology is determined from the gains of the estimated dynamic models for the modules in a network using the PEM. The interconnectivity of the network we are trying to identify is shown in Figure 4.1, along with the corresponding impulse responses. The topology is chosen to have just a few interconnections. In this example, the system G_{14} is not present in the data generating network. Each node signal w_j, $1 \leq j \leq L$, is perturbed by a white noise signal v_j of variance $\sigma_j^2 = 0.09$, which is mutually uncorrelated with the noise signals that affect the other nodes. We also have measured exitation signals r_j, at each node w_j, which are white noise of zero mean and variance $\sigma_r = 1$, mutually uncorrelated and uncorrelated with the noise signals v_j. Under these conditions, $N = 500$ data points are captured from each node of the network.

Estimating the dynamic models for each module and setting the threshold T to $T = 0.02$ we evaluate the detection of the links through the following confusion matrix. The $\mathscr{H}_2$ system norm of each identified module $\hat{G}_{j,i}$ represents the score and it is indicated in the matrix entry $K_{j,i}$. Based on whether $K_{j,i} > T$ or not we assign the label connection present or connection absent to the respective module.

$$K = \begin{pmatrix} 0 & 0.016 & 0.131 & 0.017 & 0.02 \\ 0.107 & 0 & 0.019 & 0.018 & 0.021 \\ 0.017 & 0.044 & 0 & 0.017 & 0.015 \\ 0.014 & 0.014 & 0.013 & 0 & 0.015 \\ 0.016 & 0.017 & 0.017 & 0.072 & 0 \end{pmatrix} \tag{4.2}$$

The topology assessment using the $\mathscr{H}_2$ -norm as score is dependent of the threshold choice. Figure 4.2a reveals the importance of it through a graph where the sensitivity and the false positive rate ($FPR = FP/N = 1 - TNR$) are evaluated for different threshold choices. Note that such a graphic cannot be used in deciding upon the right choice of T for an application since the measures used, compare the detected topology with the actual true topology. Thus knowing the true topology, from the figure, we see that the best choice of T is in the interval $T \in [0.022; 0.043]$ where $TPR = 1$ and $FPR = 0$. There, all the links that are present in the data generating network are detected as present as well as all the links that are absent in data generating network are detected as absent.

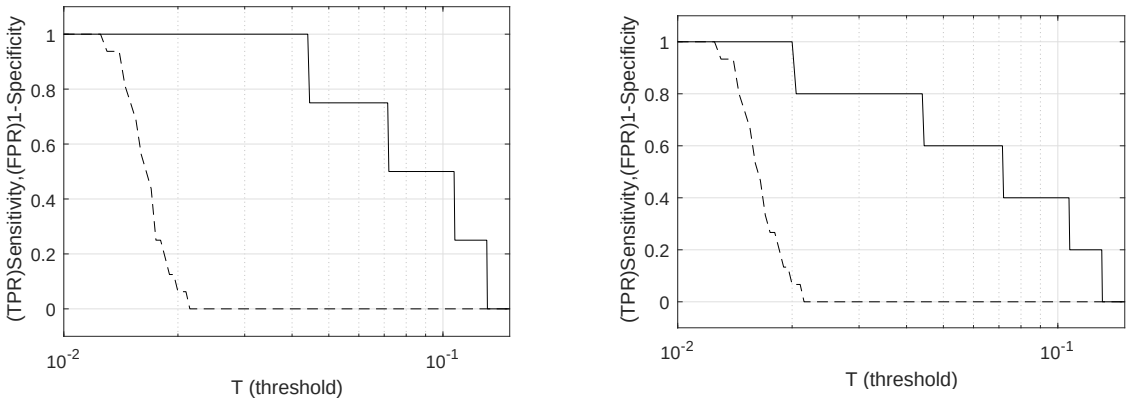
Example 4.1.2. Let's consider the same dynamic network as before but with an additional weak connection from w_4 to w_1. In this case, as can be seen in Figure 4.2b, where the TPR and FPR



(a) Topology

(b) Impulse responses.

Figure 4.1: Data generating network in example 4.1.1.



(a) (b)

Figure 4.2: Sensitivity (solid line) and false positive rate (dashed line) as we vary the threshold.

are plotted over the threshold, that there is no value of T for which we can obtain at the same time $TPR = 1$ and $FPR = 0$. So in this case, no matter which threshold value we choose, we won't obtain perfect topology detection as either some present links will be classified as being absent or some absent link will be classified as being present. Figure 4.3 presents the estimated impulse responses for some of the modules. Because of its relatively small gain, the added module G_{14}, is indistinguishable from the noise. Also, notice that the trade-off between TPR and FPR is unbalanced because of the imbalance in the number of true connections present and the number of true connections absent. Thus, we see that a decrease in TPR to 0.8 results in an increase in FPR of only 0.06.

Example 4.1.3. Since FIR models require a large number of parameters, thus leading to increased variance in their estimates, we will repeat the experiment in example 4.1.1, but now using output-error models. We consider from the beginning the network with the module G_{14} present. First, we

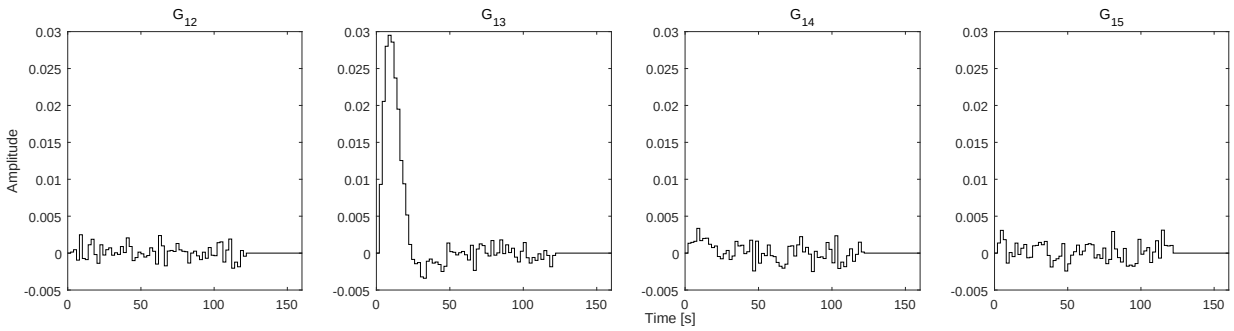


Figure 4.3: Impulse response estimates.

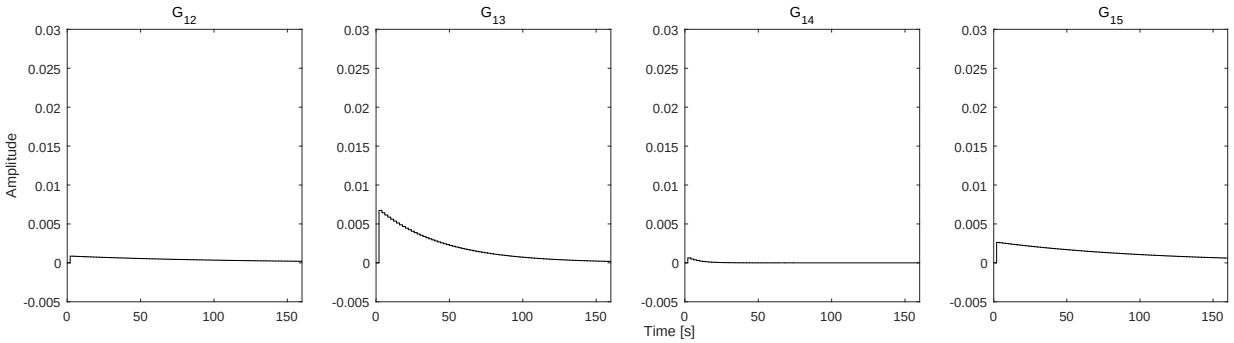


Figure 4.4: Impulse responses of the first order OE models.

try to model the modules using a model set, which is unable to cover the data generating system, by using output-error models

$$w_j[t] = \sum_{i=1, i\neq j}^{L} \frac{B_i(q^{-1})}{F_i(q^{-1})} w_i[t-k] + e_j[t] \tag{4.3}$$

with orders

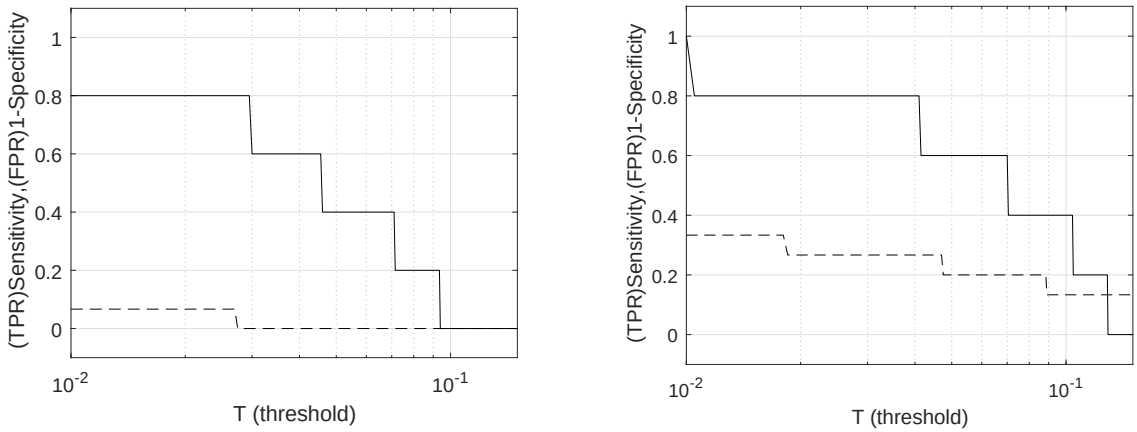
	u_1	u_2	u_3	u_4
n_b	1	1	1	1
n_f	1	1	1	1
n_k	1	1	1	1

The resulting impulse responses of some of the estimated models are shown in Figure 4.4. We see that first order models are unable to capture correctly the dynamic behavior connecting the signals, and because of that it offers a poor detection of the topology having a short interval where the threshold offers the best results (see Figure 4.5a).

Better models are obtained if we consider a more flexible model set given by the OE models with model orders

	u_1	u_2	u_3	u_4
n_b	5	5	5	5
n_f	5	5	5	5
n_k	1	1	1	1

The obtained models, shown in Figure 4.6, indicate better dynamic modeling than before, but the absent connections are still affected by the noise. A consequence of using parametric OE models,



(a) Under-fitted model.

(b) Over-fitted model.

Figure 4.5: Sensitivity (solid line) and false positive rate (dashed line) as we vary the threshold.

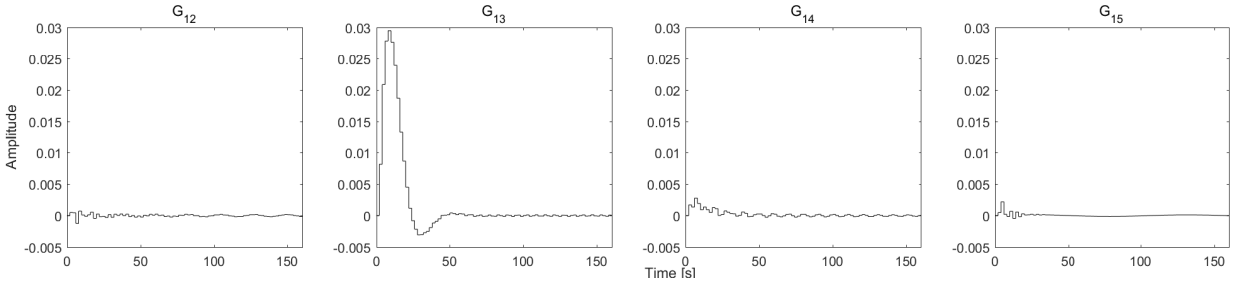


Figure 4.6: Impulse responses of the fifth order OE models.

compared with the nonparametric FIR models, is the improvement in variance we see in the impulse responses for higher lags. For lower lags the noise is still being present thus affecting topology detection. The drawback of using parameteric models is that the model orders affect the quality of the estimates. Despite the better modeling of the dynamics, the TPR, FPR characteristic in Figure 4.5b actually indicate a much worse result in terms of topology detection, especially in the detection of absent connections compared with the case in Figure 4.5a. It just might be that the better topology detection for the network with a worser model does not indicate a general rule but it just a particular result of this configuration. The bias due to undermodeling actually pushed the parameters, incidentally, such that a better topology can be determined.

To decide the threshold, in practice, one must compare the detected topology to the data, or in other words validate the topology. A way of doing so is to create a new network model based on the identified network model by setting the absent detected links to zero. Then we will verify whether the fit to validation data of the newly obtained model improves compared with the original identified network. We can apply this strategy to test how does the threshold choice affects topology detection without having access to the true topology.

4.2 Topology detection by stable splines with exponential hyperpriors

The approach to topology detection in this thesis, presented in section 2.6, requires network module modeling, which could be obtained using the PEM. In this section we take a Bayesian approach

to prediction and consequently to topology detection. We will address two problems related to the PEM, in the context of topology detection: the module model order selection problem and the network overfit problem, and provide solutions to them through Bayesian methods. The solutions provided for the two problems lead to the method for topology detection by stable splines with exponential hyperpriors (SSEH), introduced in [9].

In order to obtain predictive models that maintain minimal over-fit of the noise realization, as discussed in section 2.2, model orders have to be correctly specified. A typical solution to this problem is to use cross validation methods. However, the issue with these methods is that in our network context we will deal with MISO identification problems which require a more complex order selection problem as the orders of multiple modules have to be determined at the same time. The cross-validation procedure, in this case, would involve individually selecting the model orders for each module and then testing how different structures, chosen as such, compare with one another. Since this solution might be too complex to apply in practice, other automated solutions are preferable. A widely used automated method for model order selection is through Akaike's information criterion (AIC, see [1], [2]). However, in this report we take a different path and choose stable spline modeling for solving the model order selection problem, as it was shown in the literature (see [10], [9]) to provide better results in term of prediction compared with models obtained using PEM and AIC and to provide very close results to the ones using the PEM and the true model orders.

In section 2.5, the problem of network overfit was revealed, indicating a possible concern when the topology is to be determined by the chosen method explained in section 2.6. Namely, when network overfit occurs, the parameters of the connections that should be absent are different than zero which might make it difficult to discern whether the connection is present or not. The network overfit also makes the MISO estimates as a hole to be less predictive compared with the situation when the true topology is known and would be used for identification. In section 4.1 a few numerical examples were provided showing the mentioned issues related to the model order selection and the network overfit problem, when using classic PEM. That is, the classification threshold has to be chosen and it might be difficult to agree upon a suitable one. In the present section, we indicate a way of handling this problem. We, thus, turn once again to Bayesian statistics to provide the solution.

Since in practice it is common to have networks of interconnected modules, with each of the modules interconnected only with a few other modules, and since the network overfit problem is characteristic for such networks, estimators which provide sparse results will represent a solution (estimators which use the Bayesian framework to consider the sparsity structure as a prior will be tested). For linear regression models, such estimators try to select only a subset of the regressors, instead of using all of them, by setting the corresponding parameters to zero, in a way to improve prediction and provide interpretability. In the literature the most commonly used estimator to work as such, is the least-absolute-shrinkage-and-selection-operator (Lasso), introduced in [16].

For a linear regression model (c.f. equation (2.12))

$$y[t] = \varphi^T[t]\theta + e[t], \qquad e[t] \sim \mathcal{N}(0, \sigma^2), \tag{4.4}$$

the Lasso estimate is given by

$$\hat{\theta}^{lasso} = \arg\min_{\theta} \frac{1}{N} \sum_{t=nb+1}^{N} (y[t] - \varphi^T[t]\theta)^2 + \lambda ||\theta||_1, \tag{4.5}$$

where $|| \cdot ||_1$ is the vector 1-norm. The effect of using the 1-norm for regularization is known to provide sparse solutions ([16]). Here, the regularization parameter λ is a tunning parameter that should be adjusted, and the usual way to do this is through cross-validation. An interesting fact is that the Lasso solution could be interpreted as the convex relaxation of the l_0-penalized linear

regression problem

$$\hat{\theta}^{l_0} = \arg\min_{\theta} \frac{1}{N} \sum_{t=nb+1}^{N} (y[t] - \varphi^T[t]\theta)^2 + \lambda ||\theta||_0, \tag{4.6}$$

where the 0-norm, which is defined as the number of non-zero elements of the vector θ, is used to penalize the estimates [19]. The solution to this optimization problem, because of the 0-norm, involves an exhaustive search through all possible parameter subsets, which makes it impractical. Thus, the Lasso is instead used as an approximation to it and is solved using proximal gradient methods.

Consider the network setup of section 2.3 given in (2.39), with a number of L measured signals w_j, $1 \leq j \leq L$. Then the classic PEM estimates, for the FIR model structures (2.46), are given by (2.44). Then, we would want to select the best subset of input signals w_k, $k \in \mathcal{X}_j$, that should be used for predicting signal w_j. This could be done by using the SSEH estimates introduced in [9], given by

$$\hat{\theta}^j = \arg\min_{\theta^j} \frac{1}{N} \sum_{t=1}^{N} (w_j[t] - \sum_{k \in \mathcal{X}_j} G_{jk}(q^{-1}; \theta^{jk}) w_k[t] - r_j[t])^2 + \sum_{k \in \mathcal{X}_j} \frac{\sigma^2}{\lambda_k^2} ||\theta_{jk}||^2_{K_{ss}}. \quad (SSEH) \tag{4.7}$$

where K_{ss} is the stable-spline kernel from (3.21). In this setup, sparsity is introduced through the individual regularization factors λ_k using a hierarchical model as will be explained. Intuitively, the method works in the following way. When λ_k is close to zero or even zero, the penalty will be so large that the optimization will push the corresponding parameters θ^{jk} to zero. When λ_k is not close to zero, then the penalty term would allow for regularization using the stable spline kernel K_{ss}.

To provide sparse estimates the SSEH assumes a probabilistic model for the λ_k parameters, namely it assumes the λ_i's to be independent exponentially distributed random variables

$$p(\lambda_k) = \gamma \exp(-\gamma \lambda_k) \mathbf{1}(t) \tag{4.8}$$

where γ is a positive parameter of the hyperprior which needs to be determined from data, and $\mathbf{1}(t)$ is the Heaviside step function, i.e. $\mathbf{1}(t) = 1$ for $t \geq 0$ and $\mathbf{1}(t) = 0$, otherwise. It is also assumed that the terms $||\theta_{ji}||_{K_{ss}}$ are independent and normally distributed random variables with mean zero and variance λ_k^2, i.e.

$$p(||\theta^{jk}||_{K_{ss}} | \lambda_k) = \frac{1}{\sqrt{2\pi\lambda_k^2}} \exp - \frac{||\theta^{jk}||^2_{K_{ss}}}{2\lambda_k^2}. \tag{4.9}$$

Then the joint probability of all the terms $||\theta^{jk}||_{K_{ss}}$, for $k \in \mathcal{X}_j$ follows from marginalizing out the λ_k's and the independence assumption,

$$p\left(||\theta^{jk_1}||_{K_{ss}} \quad \dots \quad ||\theta^{jk_{L-1}}||_{K_{ss}}\right) = \prod_{k \in \mathcal{X}_j} \int p(||\theta^{jk}||_{K_{ss}} | \lambda_k) p(\lambda_k) d\lambda_k, \tag{4.10}$$

with $k_i = i$ if $k < j$ and $i_k = k + 1$ if $k \geq j$ for $i \in \mathcal{X}_j$. This representation is called a scale mixture of normals and it is known (see [9]) that if $p(\lambda_k)$ is exponentially distributed, then the joint distribution $p\left(||\theta^{jk_1}||_{K_{ss}} \quad \dots \quad ||\theta^{jk_{L-1}}||_{K_{ss}}\right)$ would be Laplace distributed. It is also a known fact that the Lasso estimate can be interpreted as the Bayesian estimate under the Laplace distributed prior [16]. Thus making the assumptions given through equations (4.8), (4.9), we give the sparse nature of the SSEH estimates.

Now the solution to the optimization problem (4.7) fits the Bayesian context presented in section 3.1, with the difference that there the results where provided for SISO systems, whilst here we

have a MISO problem. The regression model for the multi-input single-output case is then

$$y_j[t] = \varphi^T[t]\theta + e_j[t], \qquad \theta = \begin{pmatrix} \theta^{jk_1\,T} & \dots & \theta^{jk_{L-1}\,T} \end{pmatrix}^T, \tag{4.11}$$

$$\varphi[t] = \begin{pmatrix} \varphi^T_{jk_1}[t] & \dots & \varphi^T_{jk_{L-1}}[t] \end{pmatrix}^T. \tag{4.12}$$

with $k_i = i$ if $k < j$ and $k_i = i + 1$ if $k > j$ for $i \in \mathcal{X}_j$. Or, when using N measurement samples, as

$$Y_N = \Phi_N^T\theta + \Lambda_N, \qquad Y_N = \begin{pmatrix} y_j[n_g+1] & y_j[n_g+2] & \dots & y_j[N] \end{pmatrix}^T \tag{4.13}$$

$$\Phi_N = \begin{pmatrix} \varphi[n_g+1] & \varphi[n_g+2] & \dots & \varphi[N] \end{pmatrix} \quad \Lambda_N = \begin{pmatrix} e_j[n_g+1] & e_j[n_g+2] & \dots & e_j[N] \end{pmatrix}^T \tag{4.14}$$

and the posterior estimate is given by

$$\hat{\theta}_N^{apost} = \hat{\theta}_N^{LS} - \sigma^2(P_n R_N + \sigma^2 I)^{-1}\hat{\theta}_N^{LS}, \tag{4.15}$$

$$\hat{\theta}_N^{LS} = (\Phi_N\Phi_N^T)^{-1}\Phi_N Y_N. \tag{4.16}$$

The noise variance is determined from the least squares solution $\hat{\theta}_N^{LS}$.

4.2.1 Hyperparameter estimation for networks

In order to compute the posterior solution, the stable spline kernel matrix has to be provided. As discussed in section 3.1, the hyperparameters for the stable spline kernel, namely

$$\xi = \begin{pmatrix} \lambda_1 & \dots & \lambda_{L-1} & \beta & \gamma \end{pmatrix}, \tag{4.17}$$

will be determined from the data, through the marginal likelihood estimation (Proposition 4 from [9]):

$$\hat{\xi} = \arg\min_{\xi} J(\mathcal{D}, \xi), \qquad J(\mathcal{D}, \xi) = \frac{1}{2} Y_N^T V_N^{-1} Y_N + \frac{1}{2}\log(\det(2\pi V_N)) + \gamma \sum_{i=1}^{L-1} \lambda_i, \tag{4.18}$$

with $V_N = \sigma^2 I + \sum_{i=1}^{L-1} \lambda_i^2 \Phi_i K_{ss}(\beta)\Phi_i^T$, obtained from marginalizing out the parameters θ from the joint distribution of Y_N and θ.

Since the marginal likelihood is not guaranteed to be convex, using convex optimization tools to solve the problem at hand, might lead to local solutions. As discussed for the SISO identification case, in section 3.2, appropriate initial values for the convex optimization routine, in the basin of attraction of the global minimum, if possible, would allow for the finding of the global minimum. Two ways for initializing the optimization routine for the SISO case were provided in section 3.2. In the following, we aim at extending one of the presented SISO methods for obtain an initial value for the rate decay β to the network setup.

To simplify the reasoning, we consider that the impulse responses of the modules in our network have similar rate decays. In this case, the difference in hyperparameter estimation between the network situation and the SISO situation is that multiple regularization gains λ_i, one for each module as indicated in formula (4.17), have to be determined increasing the dimensions of the optimization problem, see (4.18). As was pointed out in [9], because of the increased complexity of the optimization problem, the choice of initial values provided to the optimization routine influences the success of obtaining the global solution.

A Bayesian forward selection algorithm for choosing initial values for individual λ_i hyparameters was presented in [9]. Because this algorithm will also be used in the simulations provided in this report, for ease of reference, it is explained below:

1. In a first stage, a solution to a simplified marginal likelihood cost is computed.

$$\begin{pmatrix}\hat{\lambda_1} & \dots & \hat{\lambda_{L-1}} & \hat{\beta}\end{pmatrix} = \arg\min_{\xi} J(\mathcal{D},\xi), \quad s.t.\ \beta > 0,\ \lambda_1 = \lambda_2 = \dots = \lambda_{L-1} \geq 0. \tag{4.19}$$

   This simplified version considers a single, common λ for all of the modules, i.e. $\lambda_1 = \lambda_2 = \dots = \lambda_L = \lambda \geq 0$ and fixes $\gamma = 0$.

2. Then, in a second stage, a input selection procedure is employed. Let I denote the set of indexes of selected inputs for prediction, i.e. $I = \{i | u_i \text{ is selected}, 1 \leq i \leq L-1\}$. Associated with this set, is the set of hyperparameters $\xi_I = \begin{pmatrix}\tilde{\lambda}_1 & \dots & \tilde{\lambda}_{L-1} & \hat{\beta} & \hat{\gamma}\end{pmatrix}$, with fixed $\hat{\gamma} = 1/\hat{\lambda}$ and $\tilde{\lambda}_i = \hat{\lambda}$ if $i \in I$, otherwise $\tilde{\lambda}_i = 0$. Also, denote by $J(\mathcal{D},\xi_I)$ the marginal likelihood cost evaluated for the hyperparameter values ξ_I, associated with the set I. Initially, none of the inputs are selected, $I = \emptyset$.

   (a) For each input that has not been selected yet, i.e. $k \in \{1,\dots,L-1\}/I$, construct a new index set $I_k = I \cup \{k\}$ with associated hyperparameters ξ_{I_k}, and marginal likelihood cost $J(\mathcal{D},\xi_{I_k})$ as defined earlier.

   (b) Choose, for $k \in \{1,\dots,L-1\}/I$, one k^* that gives the smallest value for $J(\mathcal{D},\xi_{I_k})$.

   (c) Check if the marginal likelihood cost becomes smaller when using hyperarameters $\xi_{I_{k^*}}$, compared with the cost when using ξ_I, i.e. see if $J(\mathcal{D},\xi_{I_{k^*}}) < J(\mathcal{D},\xi_I)$. If the condition is satisfied, select k^* by setting $I = I_{k^*}$ and repeat from a). If it is not, then it means that none of the remaining inputs $k \in \{1,\dots,L-1\}/I$ can be individually selected to give a smaller marginal likelihood cost. In this case we stop the iteration and the final hyperparameters are given by the last obtained ξ_I.

In this report, we provide an addition to the just presented algorithm by looking for initial values for the simplified marginal likelihood (4.19) from the first stage of the forward selection algorithm.

Effectively, the hyperparamter vector for the simplified problem (4.19) contains only two variables $\xi = \begin{pmatrix}\lambda & \beta\end{pmatrix}$. An initial value for the β could be obtained from a first order OE model. Since the impulse responses have similar rate decays, we will simplify the procedure by using a first order OE model , that has the same pole, for every transfer function:

$$y_j[t] = \sum_{i=1}^{L-1} \frac{b_{i,0}}{1+f_0 q^{-1}} u_i[t-1] + e_j[t] = \frac{1}{1+f_0 q^{-1}} \sum_{i=1}^{L-1} b_{i,0} u_i[t-1] + e_j[t]. \tag{4.20}$$

Then the first order OE estimate for the MISO setup is

$$\hat{\theta}^{OE} = \arg\min_{\theta} \sum_{t=1}^{N} \left[y_j[t] - \sum_{i=1}^{L-1} \frac{b_{i,0}}{1+f_0 q^{-1}} u_i[t-1] \right]^2, \tag{4.21}$$

and the rate decay β will be derived as the pole of the first order OE approximation.

Different rate decays

In the above derivations, we considered that every module of the network has the same rate decay, but we might come across networks which don't have this structure. In such a situation, considering individual rate decays for each module will lead to a further increase in complexity of marginal likelihood optimization problem. Based on the previous suggestion, for the simpler case with a common rate decay, namely, of using a first order OE model to determine an initial β, we extend the procedure to accommodate the new situation with different rate decays. We try to use a first order output error model for each module of the network. To reduce the bias that might appear due to under-modeling, we first estimate an FIR multi-input single-output model of our network without regularization. Then we try to fit, individually, first order OE models over the previously determined FIR models.

Hence, first, we define individual data sets $\mathcal{D}_{ji} = \{y = \hat{\theta}_j^{FIR}; u = \begin{pmatrix} 0 & 1/T_s & 0 & \cdots & 0 \end{pmatrix}\}$ for each module, where T_s is the sampling time and u is a unit impulse applied at time T_s. Then, we obtain the output error estimates as

$$\hat{\theta}^{OE} = \arg\min_{\theta} \sum_{t=1}^{N} \left[y[t] - \frac{b_0}{1 + f_0 q^{-1}} u[t] \right]^2, \tag{4.22}$$

and, as before, we determine initial values for the rate decay hyperparameters from the poles of the output error models. The complete strategy for hyperparameter estimation for different rate decays is presented below.

1. Obtain individual hyperparameters β_i for each module, as just presented.
2. Solve a simplified marginal likelihood over a common λ, only, with $\gamma = 0$ and individual β's fixed at the values determined in the previous step.
3. Do forward selection as presented before, with the only difference that $\hat{\beta}$ used here should be a vector $\hat{\beta} = \begin{pmatrix} \hat{\beta}_1 & \dots & \hat{\beta}_{L-1} \end{pmatrix}$.
4. Solve a complete marginal likelihood for $\xi = \begin{pmatrix} \lambda_1 & \dots & \lambda_{L-1} & \beta & \gamma \end{pmatrix}$, with $\beta = \begin{pmatrix} \beta_1 & \dots & \beta_{L-1} \end{pmatrix}$ and with initial conditions set at the values obtained in the last step.

In this subsection, we have presented two possible initial values for the kernel rate decays, to be used for optimizing the marginal likelihood (4.18), one for the networks with systems that have the same settling time, and the other for networks with systems that have different settling times. The forward selection mechanism from [9], might appear to provide the topology of the network. In this thesis, the approach to topology detection, as presented in Chapter 2, is through the use of dynamic models and we do not determine the topology only based on the values of λ_i, $i \in \{1, \dots, L-1\}$ obtained from the forward selection. In order to obtain the models through SSEH, we need to determine the hyperparameters from the non-convex marginal likelihood optimization (4.18). The hyperparameters that give the global minimum of the marginal likelihood will be used to construct the sparse models. When a given λ_i is close to zero, then the regularization penalty becomes large in (4.7) forcing the corresponding parameter values to zero. We will not make any deduction from the values of the hyperparameters λ_i, $i \in \{1, \dots, L-1\}$, about the topology of the network, and we will determine the topology only based on the resulting models obtained, since the λ indicates only the strength of regularization. Furthermore, the obtained λ_i, $i \in \{1, \dots, L-1\}$ from forward selection, are only used as initial values for minimizing the marginal likelihood, since it might happen that the optimal values of the marginal likelihood provide a different topology than the one deduced from the initial values.

4.3 Simulations

We will now determine how does the topology detection method through SSEH perform by designing different experiments. Let us denote this method by SSEH-TD.

As a reminder, for a given network, the topology is determined following the next steps:

1. Identify the modules using the chosen identification procedure, i.e. either the PEM or regularization, to obtain models suitable for prediction.
   (a) If regularization is chosen for modeling the network modules, then first determine the needed hyperparameters, and afterwards compute the estimates.
   (b) If the PEMs are used, then the order of each module has to be provided.
2. Evaluate the topology based on the estimated models.

4.3.1 Simulation 1

A first experiment is to compare the SSEH-TD with topology detection through classic PEM as was executed in section 4.1. The aim of the experiment is to investigate two aspects:

1. Whether the SSEH-TD improves the detection of the null connections, i.e. it improves the FPR.
2. Whether the SSEH-TD improves the detection of the present connections, i.e. it improves the TPR.

In both of the above cases, the referred improvement will be studied with respect to the topology obtained using classic PEM for network modeling.

Experimental setup

Let us reconsider the setup from example 4.1.2. The data generating network has the topology as in the graph 4.1a with connections represented by LTI systems with the impulse responses given in Figure 4.1b. For reference, we also add below the exact transfer functions used.

$$
\begin{aligned}
&G_{13}(z^{-1}) = z^{-1}\frac{0.01741 + 0.01523z^{-1}}{1 - 1.572z^{-1} + 0.6703z^{-2}}, \quad G_{14}(z^{-1}) = 0.1z^{-1}\frac{0.01638 + 0.0134z^{-1}}{1 - 1.459z^{-1} + 0.5488z^{-2}},\\
&G_{21}(z^{-1}) = z^{-1}\frac{0.01638 + 0.0134z^{-1}}{1 - 1.459z^{-1} + 0.5488z^{-2}}, \quad G_{32}(z^{-1}) = z^{-1}\frac{0.01609 + 0.01402z^{-1}}{1 - 0.8573z^{-1} + 0.6703z^{-2}},\\
&G_{54}(z^{-1}) = z^{-1}\frac{0.001067 + 0.003412z^{-1} + 0.00068z^{-2}}{1 - 2.222z^{-1} + 1.646z^{-2} - 0.4066z^{-3}}.
\end{aligned} \tag{4.23}
$$

Each node signal w_j, $j = \overline{1, nw}$, is perturbed by a white noise signal v_j of variance $\sigma_j^2 = 0.09$, which is mutually uncorrelated with the noise signals that affect the other nodes. We also have measured exitation signals r_j, at each node w_j, which are white noise of zero mean and variance $\sigma_r = 1$, mutually uncorrelated and uncorrelated with the noise signals v_j. Under these conditions, 500 data points are captured from each node of the network.

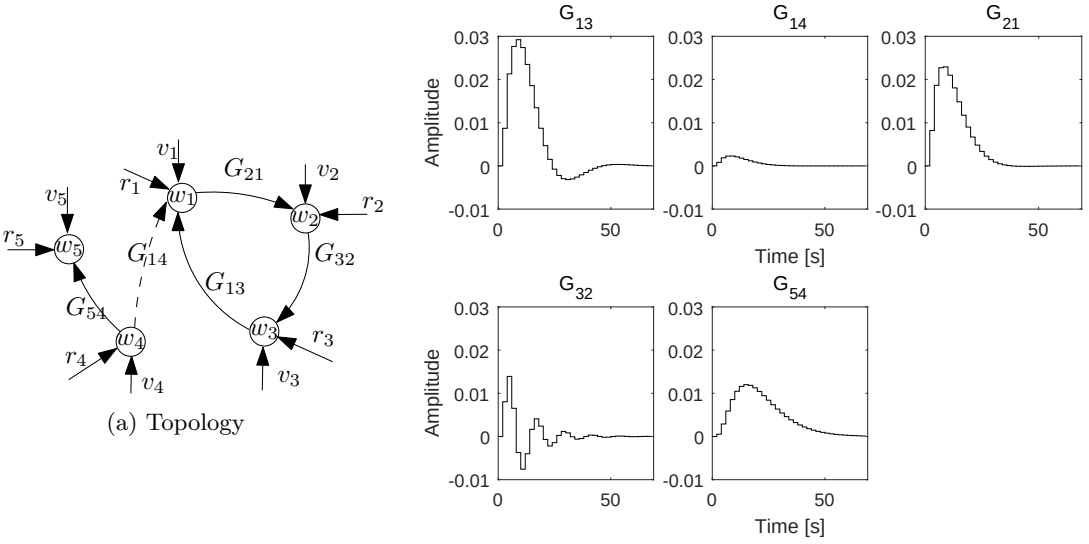
The obtained data set, $\mathcal{D} = \{(w_1[t], r_1[t], w_2[t], r_2[t], w_3[t], r_3[t], w_4[t], r_4[t], w_5[t], r_5[t]), 1 \leq t \leq 500\}$, will be used for topology detection. We thus aim to use the data set, first, to model the network, and then, to determine the topology by using the discrimination rule (2.57) for a suitable threshold. A discussion on the threshold choice will be provided.

PEM modeling

We begin with deriving the PEM models. The PEM models are obtained for the data set $\mathcal{D}$, following the procedure presented in section 2.1 and example 4.1.2, by minimizing the residual sum of squares (2.44), using the predictor (2.42) and the OE model parametrization

$$
G_{jk}(q^{-1}; \theta^{jk}) = \sum_{k \in \mathcal{X}_j} \frac{B_{jk}(q^{-1}; \theta^{jk})}{F_{jk}(q^{-1}; \theta^{jk})} w_k[t-1] \tag{4.24}
$$

with orders $(n_b = 1, n_f = 2)$ for each pair of B_{jk}, and F_{jk} polynomials respectively, along with a unit delay. The model structure and order choice is made to match the data generating network with module models given in the formulas in (4.23), were we notice that all the systems are of orders $(n_b = 1, n_f = 2)$ with the exeption of G_{54}. Since G_{54} has an impulse response that resembles that of a second order system (see Figure 4.1b), the choice of model orders as $(n_b = 1, n_f = 2)$ seems a reasonable one. The estimated models are displayed in Figure 4.8. We see that the estimated present connections resemble remarkably the true impulse response in Figure 4.7b. Figure 4.8b displays the estimated transfer functions for the first MISO identification problem with w_1 considered as the output signal, and it shows that the estimates of the null connections are affected by the noise present



(a) Topology

(b) Impulse responses.

Figure 4.7: Data generating network in example 4.1.1.

in the data. Thus, from a modeling perspective it seems that the newly obtained models from the data, capture most of the effects needed for prediction, while keeping the variance small, especially, for the null connections parameters, making them have little effect on predictions. **SSEH modeling**

We perform next, the SSEH modeling of the same data generating network using the same data set $\mathcal{D}$ as was used with the PEM. Differently from the PEM, we obtain the model parameters by minimizing the regularized version of the residual sum of squares cost function presented in formula (4.7).

$$\{\hat{\theta}_{ji}\}_{i=1,i\neq j}^{nw} = \underset{\{\theta_{ji}\}_{i=1,i\neq j}^{nw}}{\arg\min} \; ||Y_j - \sum_{i=1,i\neq j}^{nw} \Phi_{ji}^T \theta_{ji}||_2^2 + \sum_{i=1,i\neq j}^{nw} \frac{\sigma^2}{\lambda_i^2} ||\theta_{ji}||_{K_{ss}}^2. \quad (SSEH) \tag{4.25}$$

using the same predictor as for the PEM, but with models having an FIR structure of order 60:

$$y_j[t] = \sum_{i=1}^{L} G_i(q^{-1}) u_i[t] + e_j[t], \quad G(q^{-1}) = q^{-1} g_1 + q^{-2} g_2 + \cdots + q^{-60} g_{60}. \tag{4.26}$$

Hyperparameter estimation

Before solving (4.7), the hyperparameters $\xi = \begin{pmatrix} \lambda_1 & \cdots & \lambda_L & \beta & \gamma \end{pmatrix}$ have to be estimated from the marginal likelihood problem (4.18). Following the procedure in section 3.2, we extend it to the MISO case by modeling a first order OE model for each module.

A summary of the hyperparameter estimation steps used in the simulations is provided below:

1. Estimate a first order OE model with a common pole f_0 for all transfer functions, different gain $b_{i,0}$, and a unit delay (see equation (4.21)). Deduce the hyperparameter from the obtained model.
2. Solve a simplified marginal likelihood optimization problem with one β and one λ for all the module priors. Initialize the optimization routine at the previously determined β and the initialization for λ could be set at zero or a small number.

y	$\hat{\sigma}^2$	β^{oe}	$\hat{\beta}$	$\hat{\lambda}$
w_1	0.0891	0.1414	0.1462	0.0398
w_2	0.0889	0.1224	0.1775	0.0155
w_3	0.0927	0.6405	0.2157	0.1369
w_4	0.0966	13.3271	13.3271	0
w_5	0.0866	0.0502	0.0944	0.0047

Table 4.3: Simplified marginal likelihood optimization. Each row indicates the β^{oe} obtained from a first order approximation along with the hyperparameter values $(\hat{\beta}, \hat{\lambda})$ obtained from a simplified marginal likelihood using as initial values $(\beta^{oe}, 0)$. Each row indicates the results of a different MISO case with the output indicated by w_k in the first column.

		λ_1	λ_2	λ_3	λ_4	β	γ	$J(\mathcal{D}, \xi)$
w_1	ξ^{fwd}	0	0.0398	0.0398	0	0.1462	0	277
	ξ^{final}	1×10^{-6}	0.1312	3×10^{-4}	1×10^{-5}	0.1398	1×10^{-6}	249
w_2	ξ^{fwd}	0.0155	0	0	0	0.1775	0	232
	ξ^{final}	0.0488	1×10^{-5}	7×10^{-10}	1×10^{-5}	0.1705	8×10^{-4}	221
w_3	ξ^{fwd}	0	0.1369	0	0	0.2157	0	252
	ξ^{final}	4×10^{-11}	0.5619	2×10^{-6}	1×10^{-5}	0.1831	5×10^{-7}	236
w_4	ξ^{fwd}	0	0	0	0	13.3271	0	177
	ξ^{final}	0	0	0	0	13.3271	0	177
w_5	ξ^{fwd}	0	0	0	0.0047	0.0944	0	205
	ξ^{final}	6×10^{-12}	3×10^{-12}	6×10^{-12}	0.0128	0.0909	1×10^{-5}	199

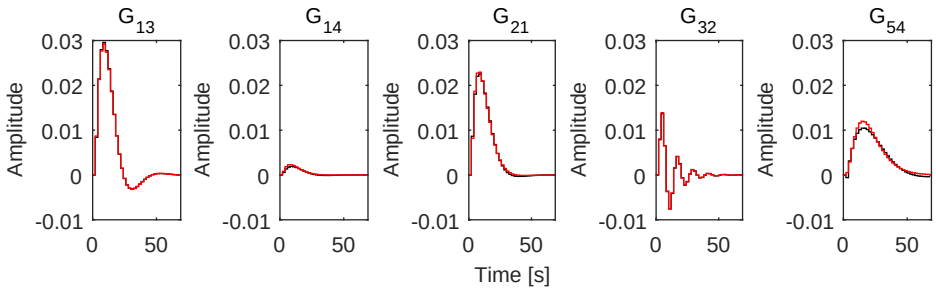
Table 4.4: Hyperparameter values obtained from the forward selection ξ^{fwd} and the marginal likelihood optimization ξ^{final} with initial values provided by ξ^{fwd}. Each row indicates the results of a different MISO case with the output indicated by w_k in the first column. The shaded boxes indicate the hyperparameters corresponding to the existing interconnections in the true network.

3. Do forward selection to select a different λ_i for each module prior.

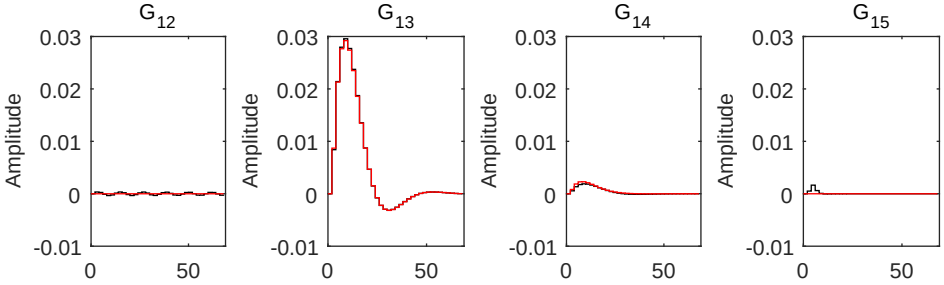
4. Solve the complete marginal likelihood with the previously determined hyperparameters as initial values.

Table 4.3 shows the estimated noise variance $\hat{\sigma}^2$, along with the first order OE estimate of the kernel rate decay β^{oe}, and the hyperparameters obtained from the simplified marginal likelihood optimization, $(\hat{\lambda}, \hat{\beta})$, for each MISO problem of the network. The noise variance looks to be estimated close to the true value of $\sigma^2 = 0.09$. We also notice that the values of the $\hat{\beta}$ from the simplified marginal likelihood seems close to the value of the β^{oe} estimated using the first order OE model, thus indicating that the first order OE provides the solution near the local optimum, which might as well be the global one. Initializing the β in a sufficiently large region around the found optimum, which covers a large band of candidate rate decays leads to the same results, suggesting that the found optimum might be the right one for regularization.

The hyperparameters resulting from the application of the forward selection algorithm to the data and the solutions from optimizing the marginal likelihood with the initialization provided by the forward selection results are provided in Table 4.4. For our considered experiment, the forward selection algorithm appears to select all of the true modules, and does not select any of the null connections, thus achieving the best results. Inspecting the final hyperparameters resulting from the complete marginal likelihood, i.e. ξ^{final}, we notice that the regularization parameters λ_i corresponding to the null connections are made small but not zero, even though the initial conditions were set at zero. Even though these results might contradict the data generating system, since the hyperparameters



(a) Estimated impulse responses using PEM for all of the existing connections in the true network.



(b) Estimated impulse responses for the models connected at the output to w_1.

Figure 4.8: Impulse responses of the estimated transfer functions using PEM (black) and true impulse responses (red) for different connections from the network.

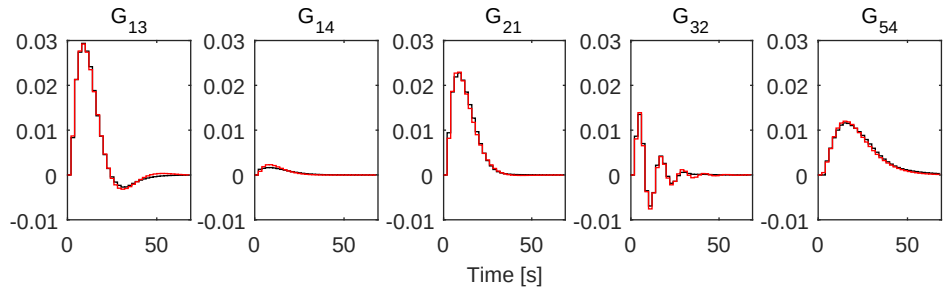
λ_i for the null connections should be set at zero to obtain the best regularization, the marginal likelihood value actually is smaller after running the complete optimization routine compared with the cost of at the initial points, as is indicated also in Table 4.4, i.e. $J(\mathcal{D}, \xi^{fwd}) \geq J(\mathcal{D}, \xi^{final})$. The non-sparse hyperparameter results might also explain the low values of the γ.

SSEH modeling results.

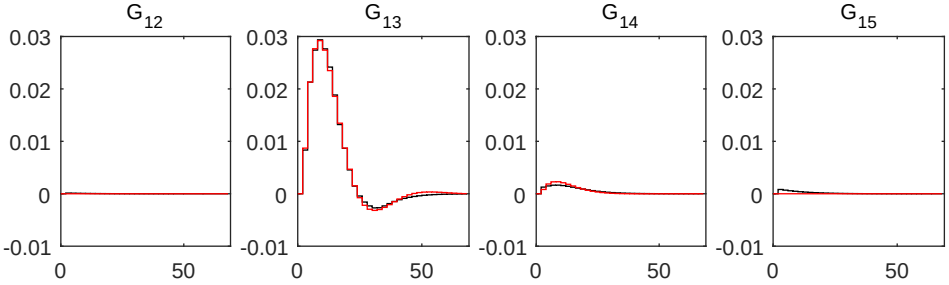
Dynamic models for each module of the network are obtained as previously explained using the determined hyperparameters ξ^{final}. The results are exposed in Figures 4.9. The first Figure 4.9a shows the impulse reponses of the models for the modules present in the network. Comparing these with the true impulse responses of the data generating system in Figure 4.1, we infer that the SSEH provides accurate models, with almost unnoticeable differences. The second Figure 4.9b presents the models when the signal w_1 is considered as an output. In this figure, we notice the relatively small gains of the null connections compared with those of the present modules. The impulse response models of G_{12} and G_{15} are almost set to zero and thus have little contribution the prediction of $y = w_1$, showing that the SSEH works as intended, that is, it tries to eliminate the effect the noise has on the estimation of the null connections. The corresponding values of the regularization gains λ_i used in obtaining the models for G_{12} and G_{15} are given in Table 4.4, on the row of w_1 and ξ^{final} and columns denoted by λ_1 and λ_4. The values of these hyperparameters, close to zero correspond to the strong regularization needed to push the model gains to zero as was seen in Figure 4.9b.

Model comparison

To better appreciate the choice of modeling method made for the purpose of topology detection, a comparison between the PEM and SSEH models will now be made. Thus, we will compare the predictive quality of the obtained models through both of the identification methods. First, a comparison in terms of the RMSE between the estimated ($\hat{h}$) and true (h^0) impulse responses is provided in Table 4.5, with $RMSE(\hat{h}, h^0) = (\frac{1}{nb}\sum_{i=1}^{nb}(\hat{h}[i-1] - h^0[i-1])^2)^{1/2}$. The shaded cells indicate the



(a) All the existing systems in the true network.



(b) System that have outputs that connect directly to node w_1.

Figure 4.9: Estimated impulse responses using SSEH (black) and the true impulse responses (red).

modules that are present in the data generating network. For reference the $\mathscr{H}_2$ norm calculated as the RMSE between the system impulse response and 0, i.e. $||h^0||_{\mathscr{H}_2} = (\frac{1}{nb}\sum_{i=1}^{nb}(h^0[i-1])^2)^{1/2}$, is given for G_{13}, the system with the largest effect, and for G_{14}, the system with the smallest effect (c.f Figure 4.1b). Thus, we notice that the errors obtained in Table 4.5 are small relative to the norm of G_{13}, both for modeling the present modules and the missing connections. It is worth mentioning that it comes without surprise that the models obtained using PEM perform so well, since the chosen model orders, for the present modules, were made to match that of the data generating network. It is remarkable, on the other hand, that the SSEH offers models that are able to compete with the PEM with optimal model order, given that SSEH does not require the specification of the model orders.

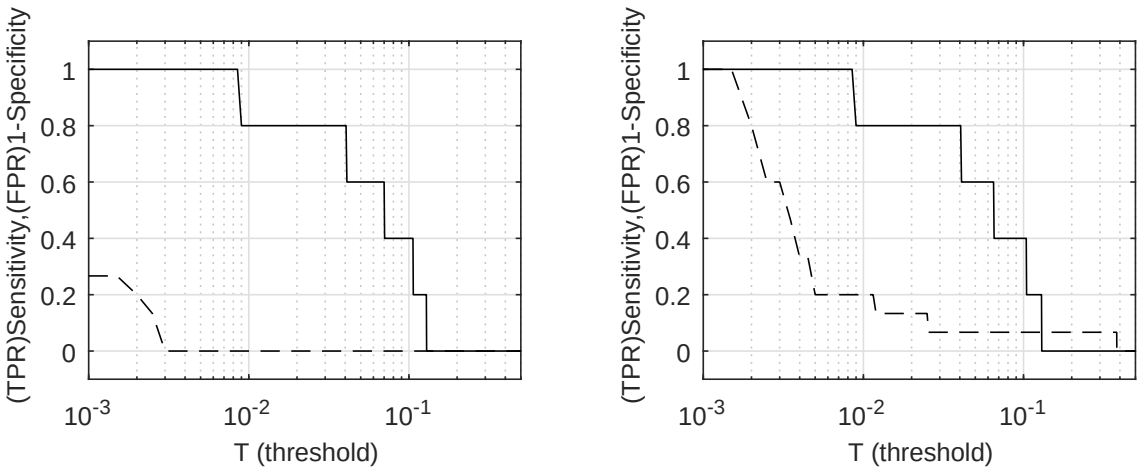
Inspecting the RMSE table, we could also determine how do PEM compare with SSEH in terms of topology detection. The best results are seen when using SSEH for $y = w_5$. In that case, all the models of the null connections are set to very small values, making clear the discrimination between the links.

Topology detection comparison

Now that we have obtained dynamic models for each of the modules, the network topology is determined from the discrimination rule (2.57). As explained in section 4.1, the evaluation of the topology detection for comparing the PEM and SSEH, is done through plots of the TPR, percentage of actual/present connections estimated correctly, and FPR, percentage of null connections estimated wrongly, as the discrimination threshold is varied. The results are shown in Figure 4.10, where we see that with SSEH we are able to detect the topology without missing any actual or null connections, i.e. we obtain $TPR = 1$, $FPR = 0$ concurrently, for a threshold $2 \times 10^{-3} < T < 8 \times 10^{-3}$, whereas when using PEM there is no threshold for which we can obtain the same results. We further notice that with SSEH we are able to detect all of the null connections correctly for a threshold $T > 2 \times 10^{-3}$, whereas with PEM this is obtained for a much larger threshold $T > 0.4$ for which all the actual connections are already set to zero by the discrimination rule. The reason for this

y		u_1	u_2	u_3	u_4
w_1	PEM	4.6	1.4	2	3.7
	SSEH	0.5	4.3	2.7	2.9
w_2	PEM	3.1	1.9	11.7	4.7
	SSEH	4.5	1.8	$\propto 10^{-5}$	2.6
w_3	PEM	1.8	0.9	382.5	2.1
	SSEH	$\propto 10^{-5}$	3.9	0.5	2.3
w_4	PEM	3.8	2	3	2.5
	SSEH	$\propto 10^{-16}$	$\propto 10^{-16}$	$\propto 10^{-16}$	$\propto 10^{-16}$
w_5	PEM	25.5	1.6	3.1	8
	SSEH	$\propto 10^{-7}$	$\propto 10^{-7}$	$\propto 10^{-6}$	3.9

Table 4.5: RMSE($\times 10^{-3}$) of the difference between the estimated and true impulse responses. For reference $||G_{13}||_{\mathscr{H}_2} = 128.5 \times 10^{-3}$ and $||G_{14}||_{\mathscr{H}_2} = 10.5 \times 10^{-3}$.



(a) SSEH (b) PEM

Figure 4.10: Sensitivity (solid line) and false positive rate (dashed line) as we vary the threshold.

happening is that, using PEM, the model of system G_{34} has been assigned two complex conjugate poles on the unit circle making the system marginally stable resulting in an undamped resonating impulse response of small magnitude. The exponentially decaying stable spline prior avoided this situation by selecting only stable system models. The TPR curve for both of the cases seems identical, because as we have seen previously, both PEM and SSEH are able to model very accurately the present connections.

Because the noise realization affecting the data might not be very representative of white noise, several Monte Carlo simulations are executed. By repeating the above simulation for 50 different data sets, each with a different noise and reference signal realization, we could make the evaluation of our estimators less sensitive to inconvenient experimental situations and avoid drawing conclusions based only on isolated results. Figure 4.11 shows TPR/FPR for various threshold values for 50 Monte Carlo simulations. Over the 50 simulations we calculated the 25, 50 and 75 percentiles for both TPR and FPR. The x percentile of, for example, the TPR is the value of TPR under which the evaluation of x percent of simulations fall. Comparing Figure 4.11 with 4.10a, we see that the figures are alike, thus the conclusions that we have drawn earlier are not only valid for a specific isolated experiment.

From the previously made observations, we conclude that SSEH offers better discrimination be-

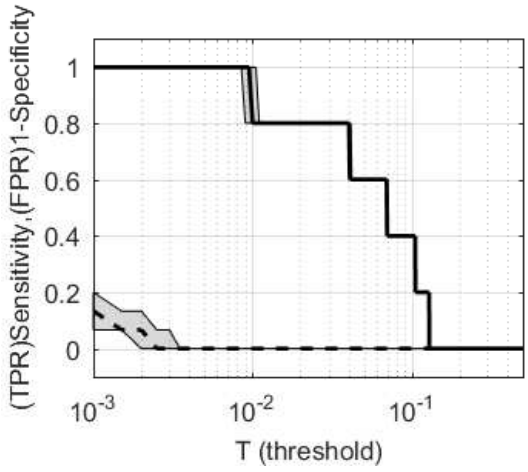


Figure 4.11: Sensitivity (solid line) and false positive rate (dashed line) as we vary the threshold, for the SSEH. 50 Monte Carlo simulations. The area between the 25 and 75 percentiles is shaded. The median is also shown in between the 25 and 75 percentile.

tween the links making it more suitable than PEM for topology detection. Even so, as we see in Figure 4.10a, when $T < 3 \times 10^{-3}$ the FPR is no longer set to zero. This indicates that SSEH is not entirely able to make automatic selection on the present connections since some of the models of the null connections still produce small effects, thus an appropriate choice of threshold has to be determined. The smaller the threshold for which the FPR is still zero, the better it is for topology detection. Thus, SSEH is better than PEM for topology detection in what concerns the choice of the discrimination threshold, but it does not eliminate the burden of having to choose one.

4.3.2 Heterogenous networks

Let us consider another simulation where the data generating network has modules with different settling times. As explained in section 4.2.1, we might want to consider separate rate decay hyperparameters for each module, increasing the complexity of the marginal likelihood. In this simulation we are going to evaluate how does the solution to this problem, introduced in section 4.2.1, perform on a test example.

To keep it simple, we will use the network from the previous simulation, i.e the one given in Figure 4.1, but with a modification to accommodate the new situation. We add the fast first order system G_{12} with impulse response which dies out in just over 10 seconds and transfer function

$$G_{12}(z^{-1}) = z^{-1}\frac{0.0583}{1 - 0.3012z^{-1}}. \tag{4.27}$$

From a newly collected data set $\mathcal{D}$ which includes the effect of the newly added module, first, the hyperparameters, $\xi_{final} = \begin{pmatrix}\lambda_{final} & \beta_{final} & \gamma\end{pmatrix}$, with $\lambda_{final} = \begin{pmatrix}\lambda_1 & \dots & \lambda_L\end{pmatrix}$ and $\beta_{final} = \begin{pmatrix}\beta_1 & \dots & \beta_L\end{pmatrix}$ now vectors of parameters, have been determined according to the strategy presented in section 4.3. The determined initialization values of the hyperameters are denoted by λ^2_{OE} and β_{OE}, where, first, β_{OE} was determined from a first order output error model on data set $\bar{\mathcal{D}}_{ji} = \{y = \hat{\theta}^{FIR}_j; u = \begin{pmatrix}0 & 1/T_s & 0 & \cdots & 0\end{pmatrix}\}$, and λ^2_{OE} was determined by forward selection. For comparison, we also estimated, from the same data set $\mathcal{D}$, a hyperparameter vector with a common β_{cmm} and individual λ^2_{cmm}. The results are displayed in Table 4.6 from where we notice that the marginal likelihood cost is the smallest when we use different β's for each module, $J_{final} < J_{cmm}$. Therefore, it seems that using individual rate decays for each module provides more accurate kernels. Also, we see that β_{final} has a larger value for the fast module G_{12} compared with that of G_{13},

$y = w_1$	β_{cmm}	β_{OE}	β_{final}	λ^2_{cmm}	λ^2_{OE}	λ^2_{final}	J_{cmm}	J_{OE}	J_{final}
$u_1 = w_2$	0.16	1.50	0.76	0.09	0.38	0.05	289	276	238
$u_2 = w_3$		0.19	0.13		0.38	0.14			
$u_3 = w_4$		0.20	0.27		0.38	0.02			
$u_4 = w_5$		0.49	7.22		0	7×10^3			

Table 4.6: Hyperparameter comparison for $y = w_1$, with J_{cmm}, J_{OE} and J_{final} marginal likelihood costs for the corresponding hyperparameter set.

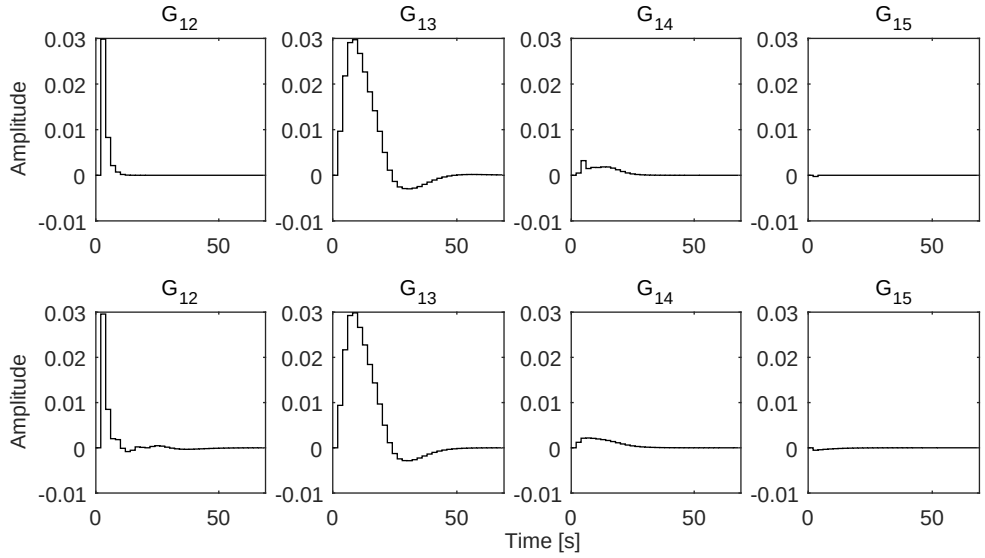


Figure 4.12: Impulse responses of the module estimates using SSEH when w_1 is considered the output signal, with different $\beta's$ (top) and with a common β (bottom).

indicating the choice of a faster decay. The estimated impulse responses using SSEH shown at the top of Figure 4.12 resemble very closely the true ones. The difference between the estimates using different β's and using a common β, could also be seen in Figure 4.12, where G_{12} seems to have more variance in it's tail in the second case, as a result of a inappropriate hyperparameter choice. As we could see in table 4.6, the common β_{cmm} is chosen towards the slower decay rate of the group corresponding to G_{13}. This is a result of G_{13} having a much larger $\mathscr{H}_2$ gain than the rest of the systems, thus dominating the effect that it has on the output y, and as such, also on the prediction of y.

W^{dif}_{est}	W^{cmm}_{est}	W^{dif}_{val}	W^{cmm}_{val}
79.41	79.42	78.82	78.60

Table 4.7: Fit to estimation (W_{est}) and validation (W_{val}) data for the estimates obtained using SSEH with a common β (W^{cmm}) and individual β's (W^{dif}).

A further comparison between the SSEH estimates using a common β and one that use individual β's is made in Table 4.7, based on the fit to validation and estimation data. The table presents the fit values calculated using (2.21) for two estimated models, namely a model which was obtained by using SSEH with a common rate decay, and another model obtained from SSEH with seperate rate decays for each module, both for estimation and validation data. The very small differences

between the two estimates, both in terms of validation as well as estimation data, indicate that there might not be a substantial need for using individual β's, as the estimates with the common β are as predictive as it's competitor. It has to be mentioned that a limitation of the fit measure (2.21) used, is that it could offer small fit values because of the noise affecting the data, even if the models are accurate enough. To justify that the obtained results are not only representative for the data set used, a Monte Carlo experiment is executed, consisting of generating the fit for 50 estimation and 50 validation data sets. The results are displayed in Figure 4.13. Since the difference $W^{cmm} - W^{dif}$ between the pair of fits (W^{cmm}, W^{dif}) is small for both validation and estimation data, then, it appears that there is no preference for using individual β's compared with a common one. Further, Figure 4.14 shows the fit to estimation data and the marginal likelihood cost over a grid of common β_{cmm} with the λ hyperparameters chosen through forward selection and optimizing afterwards a complete marginal likelihood. We see that for $0.08 \leq \beta_{cmm} \leq 1.3$ the fit to estimation data is maintained around 79%, c.f. Table 4.7. Whereas the marginal likelihood seems to have the optimal value at the same point where the fit is maximized, it shows significant changes in the cost for the same interval $0.08 \leq \beta_{cmm} \leq 1.3$. Hence, when choosing hyperparameters for SSEH models, it might be helpful to consider, besides the marginal likelihood cost, also the fit measure, as it might reveal that accurate hyperparameters are unnecessary. In conclusion, for this example, we deduce that even though the data generating system is having modules with different settling times, considering a common rate decay β for the module kernels gives network models with satisfactory predictive quality.

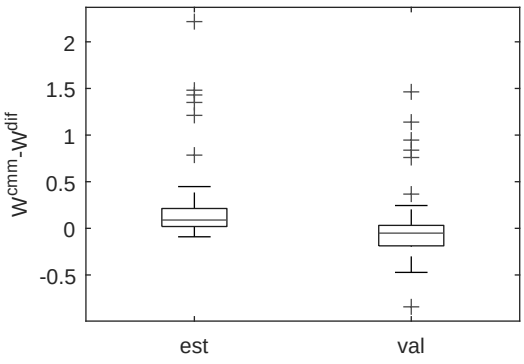


Figure 4.13: Box plot of the fit difference $W^{cmm} - W^{dif}$ between the fit of SSEH models with a common β and with different β for each module, both on estimation (left) and validation (right) data.

4.4 Summary and conclusions

In this chapter, we have presented the topology detection procedure though modeling using stable-splines with exponential hyperpriors and offered a comparison with the method of determining the topology based on classic prediction error methods. First, we have explained how we could evaluate a topology detection strategy based on the TPR, FPR over the discrimination threshold plot. Based on it, we could decide whether a topology detection method is better than another by comparing the TPR, FPR pair for corresponding threshold values. Another important aspect of a topology detection procedure that could be determined from such a plot is the range of the threshold values over which the method offers the best considered TPR, FPR pair. For a given topology detection procedure it is desirable to have this range larger. The lower the threshold could be picked the better it is, as the estimators used for modeling the dynamics, provide models for the null connections with smaller gains. Thus, improving the discrimination. In the first section, the evaluation method based on this plot, was used to indicate the problem of model order selection

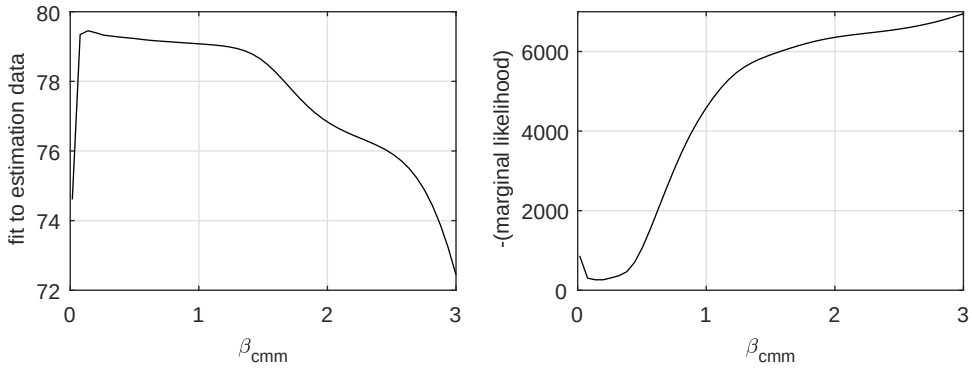


Figure 4.14: Fit to estimation data and marginal likelihood evaluated over a grid of β and optimally chosen λ.

and network over-fit in an experiment setup for the topology detection procedure based on classic prediction error methods. Afterwards, the approach to modeling using SSEH was presented along with the strategy for determining initial values for the marginal likelihood optimization based on first-order approximation of the rate decay. Simulations were provided showing a low RMSE between the true and modeled FIR parameters estimated using the SSEH. The choice of initial values for the hyperparameters, used in deriving the models, were close the optimal values, and thus helped in finding suitable hyperparameter values indicated by the good modeling results in terms of the RMSE. The comparison between the topology provided through SSEH modeling and through PEM, showed a preference towards the first, because better TPR, FPR pairs could be obtained, which could not be obtained for the second method, no matter the threshold value. Even though $SSEH$ performs better in topology detection, it is not able to automatically select all of the null connections, by pushing their model gains to zero. We have also analyzed the choice of the kernel rate decay for networks with multiple time constants, ending up with the conclusion that it might not be necessary to individually model the rate decays for each module. All things considered, regularization did help improve the topology detection, but it still could use some advancements in terms of automatically selecting the modules through sparse model representations.

Chapter 5

Conclusions and results

The objective of this thesis is to provide an efficient procedure for detecting the topology of a network of dynamical systems, by using Bayesian methods. We have reached this goal by dividing the work into several parts. First, we have investigated the use of dynamic system modeling as a means towards detecting the topology of a network. As a result, we have formulated in Section 2.6 the approach to topology detection based on Granger causality, which relies on the concept of prediction. Subsequently, we have investigated the difficulties of deriving the topology based on the prediction error methods, thus recognizing as main difficulties the model order selection and the problem of network over-fit. Afterwards, we have looked into the literature on Bayesian modeling for solutions to our identified problems related to the use of prediction error methods for topology detection, namely the problem of model order selection and network over-fit. To solve our problems we chose to use the SSEH method for modeling networks of dynamical systems. The SSEH method has its own difficulties, namely, in finding appropriate hyperparameters. In this regard, we have provided two ways of estimating appropriate initial kernel rate decays, namely though the estimation of the dominant pole and though the estimation of a first order model. Simulations were provided, showing that the results are satisfactory in both cases. The two methods for topology detection, namely through the classic prediction error method and through SSEH modeling were compared and analysed using practical examples. To compare and study the topology detection performance of the two methods, we used the confusion table along with the sensitivity-specificity plots. The provided simulations indicate that using SSEH gives a more accurate topology than the classic prediction error method. They also indicate that the threshold used to discriminate between an absent and a present connection is better defined when using the SSEH method. Even so, simulations revealed that even when using SSEH we can get errors in estimating the null connections, because SSEH was not able to automatically push to zero the parameter corresponding to the null connections. In the end, we also tested the SSEH on heterogeneous networks, from which we concluded that obtained models are relatively robust to inaccurate kernel decay rates.

Throughout the thesis, we have determined multiple factors influencing the topology detection, such as, the choice of predictor, of model structure and parametrization, of model order, the use of regularization methods, the procedure for validating a model.

The results presented on topology detection form a step towards the aspiration, stated in [17], of transitioning "from the traditional view of control systems as a single process with a single controller to recognizing control systems as a heterogeneous collection of physical and information systems, with intricate interconnections and interactions".

Bibliography

[1] T. Söderstrom and P. Stoica. *System identification*. Prentice-Hall, Inc., 1988.

[2] L. Ljung. *System identification - theory for the user*. Prentice-Hall, 1999.

[3] A.C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. SIAM, 2005.

[4] P.M.J. Van den Hof, A.G. Dankers, P.S.C. Heuberger, and X. Bambois. *Identification of dynamic models in complex networks with prediction error methods - basic methods for consistent module estimates*. *Automatica*, 49(10):2994-3006, 2013.

[5] H.H.M. Weerts. *Topology detection in dynamic networks*. Master's thesis, Eindhoven University of Technology, the Netherlands, 2014.

[6] M. Yuan and Y. Lin. *Model selection and estimation in regression with grouped variables*. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 68(1):49-67, 2006.

[7] F. Carli, T. Chen, A. Chiuso, L. Ljung, G. Pillonetto. *On the estimation of hyperparameters for Bayesian system identification with exponentially decaying kernels*. 51th IEEE Conference on Decision and Control December 10-13, 2012. Maui, Hawaii, USA.

[8] T. Chen, H. Ohlsson, L. Ljung. *On the estimation of transfer functions, regularizations and Gaussian processes - revisited*. Automatica, 2012.

[9] A. Chiuso, G. Pillonetto. *A Bayesian approach to sparse dynamic network identification*. Automatica 48: 1553-1565, 2012.

[10] G. Pillonetto, G. De Nicolao. *A new kernel-based approach for linear system identification*. Automatica 46: 81-93, 2010.

[11] G. Pillonetto, A. Chiuso, G. De Nicolao. *Prediction error identification of linear systems: A nonparametric Gaussian regression approach*. Automatica 47: 291-305, 2012.

[12] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, L. Ljung. *Kernel methods in system identification, machine learning and function estimation: A survey*. Automatica 50: 657-682, 2014.

[13] C.W.J. Granger. *Investigating causal relations by econometric models and cross-spectral methods*. Econometrica 37, 424-438, 1969.

[14] M. Ding, Y. Chen, S.L. Bressler. *Granger causality: Basic theory and application to neuroscience*. In S. Schelter, N. Winterhalder, J. Timmer. *Handbook of time series analysis*. Wiley, Wienheim, 2006.

[15] R.G.D. Steel, J.H. Torrie. *Principles and procedures of statistics with special reference to the biological sciences*. McGraw Hill, 1960, pg. 288.

[16] R. Tibshirani. *Regression shrinkage and selection via the LASSO*. Journal of the Royal Statistical Society, Series B, 58, 1996.

[17] R. Murray, K.J. Åström, S.P. Boyd, R.W. Brockett, G. Stein. *Future directions in control in an information-rich world.* Control Systems Magazine, 23(2), 20-33. DOI: 10.1109/MSC.2003.1188769, 2003.

[18] T. Fawcett. *An introduction to ROC analysis.* Pattern Recognition Letters. 27(8): 861-874, 2006.

[19] J. Mairal, F. Bach, J. Ponce. *Sparse modeling for image and video processing.* arXiv:1411.3230v2 [cs.CV], 2014.