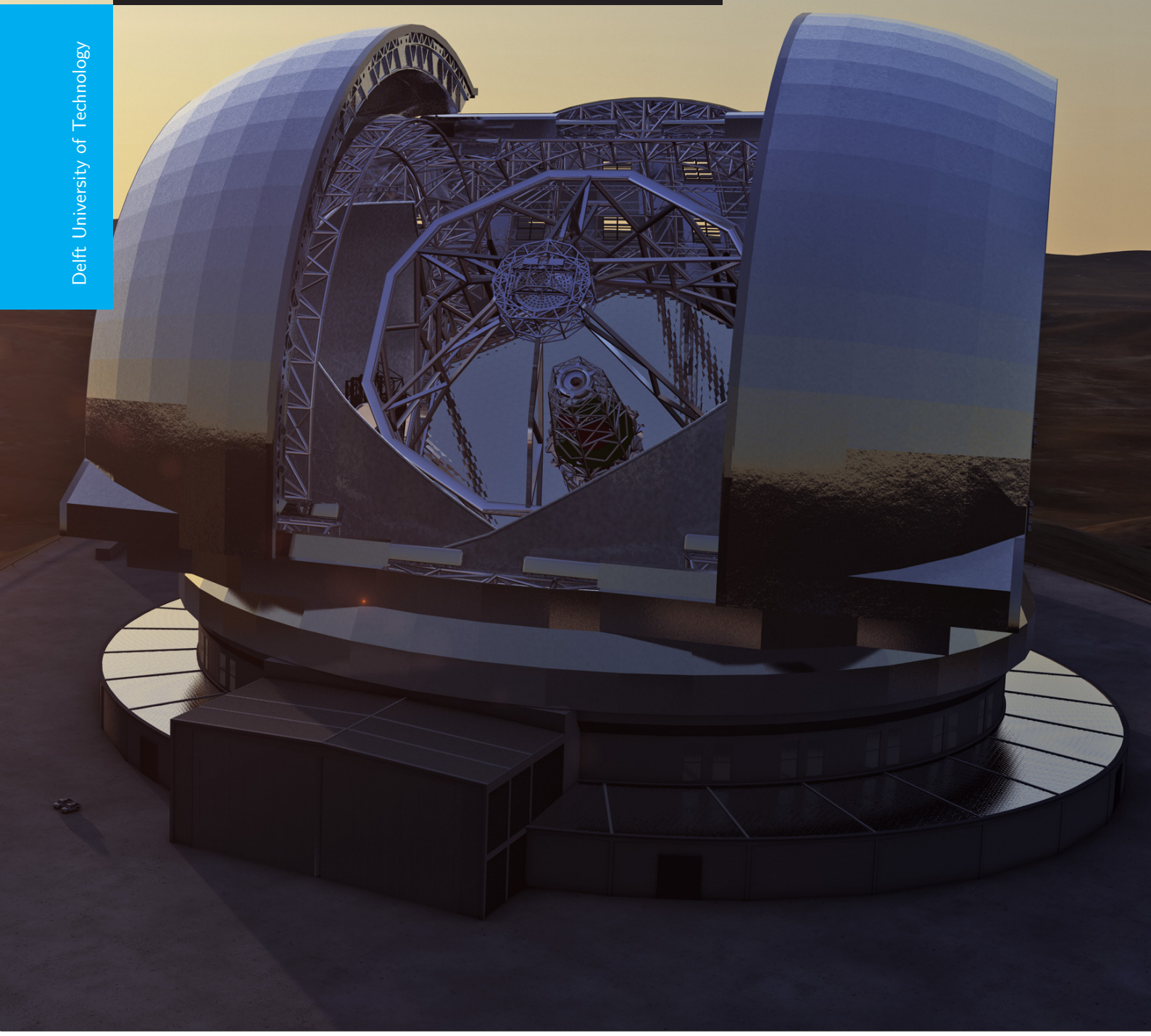
Subspace Identification of Roesser Models for large-scale adaptive optics

Mario Voorsluys

Delft University of Technology



TU Delft Delft University of Technology

Challenge the future

SUBSPACE IDENTIFICATION OF ROESSER MODELS FOR LARGE-SCALE ADAPTIVE OPTICS

by

Mario Voorsluys

in partial fulfillment of the requirements for the degree of

Master of Science
in Embedded Control Systems

at the Delft University of Technology,
to be defended publicly on Monday November 9, 2015 at 01:00 PM.

Student number:	4048482	
Project duration:	November 2014 – November 2015	
Thesis Committee:	Prof. dr. ir. M. Verhaegen,	DCSC TU Delft, Supervisor
	Prof. dr. ir. A. Schouten,	BME TU Delft,
	Prof. dr. ir. J.W. van Wingerden ,	DCSC TU Delft,
	B. Sinquin,	DCSC TU-Delft, Daily supervisor

This thesis is confidential and cannot be made public until June 31, 2016.

TU Delft Delft University of Technology

DCSC

PREFACE

This master thesis is the result of the master research I performed under supervision from Prof. Michel Verhaegen in the last year.

After two years following courses for the Master Embedded Systems and working part-time with the development of milking robots, I decided to stop working for a year to focus only on my master research. I chose to do the research at the university, to learn the most I could in my last year here. Initially I contacted Prof. Verhaegen to discuss the research options, aiming to do research on fault-tolerant control. After talking to him, the focus of the research was on the control-loop for adaptive optics. With the guidance of my daily supervisor Baptiste Sinquin and from Prof. Verhaegen, I had freedom to make my own choices and focus on what was more interesting for me.

During the literature survey, the possibility to use 2D models in large-scale adaptive optics was discussed, and a goal for the research was proposed. The goal was to study the application of 2D models in adaptive optics, but that seemed to be a project much larger than a master research.

So the focus went to the identification of 2D models, keeping in mind that the goal was to identify deformable mirrors in adaptive optics. The result of my research is here, and I hope you enjoy reading it!

ACKNOWLEDGEMENTS

I would like to express my gratitude to my daily supervisor Baptiste Sinquin and prof. dr. M. Verhaegen for their guidance, constructive criticism and their enthusiasm for my research, and special gratitude to Baptiste for the time spent in helping me and his very quick feedback.

I would like to thank also the other members of the research group. Here a special acknowledgement to Reinier Doelman, for sharing his results and allowing us to use his work on solvers for bilinear optimization problems.

I thank prof. J. Ramos and Guillaume Mercère for sharing the code of their experiments with us. Your work was really helpful for the research.

My deepest gratitude goes to my parents. They gave me the opportunity to study, supported me in my dream to study in the Netherlands, and made a really big effort to be present on the day of my defense.

And last, but not least, I would like to thank my fiancé, my family, roommates and friends for keeping me motivated and helping me were possible.

I would like to end this acknowledgements citing Johann Sebastian Bach: *"Soli Deo gloria"*.

Mario Voorsluys
Delft, November 2015

ABSTRACT

Current control algorithms for large-scale adaptive optics are computationally demanding and an accurate wavefront correction is hardly achieved within the time requirements. An idea is to exploit the local interactions between the wavefront sensor and the actuators and to develop compact models that describe the spatial dynamics of the mirror for future use in control.

The goal of this research is to study the subspace identification of 2D Roesser models to model the spatial dynamics of a deformable mirror. Two subspace identification algorithms for 2D Roesser models are presented. Both rely on the decomposition of the Roesser model into two 1D state-space models. The difference is how the models are connected. In the first case the 1D models are connected in a feed-back loop. In the second case the 1D models are connected in series.

The subspace identification using the feed-back decomposition provides good estimates of the system matrices, but can only be applied to a subclass of Roesser models. Is is also shown that using the estimates from the subspace method as an initial guess for a parametric identification allows to identify more general Roesser models. A subspace identification algorithm for the decomposition in series has been studied and handles a subset of Roesser models as well.

CONTENTS

1

INTRODUCTION

With new scientific goals in mind, astronomers need better telescopes to make sharper pictures of astronomical objects. In order to achieve this, new telescopes are being built such as the European Extremely Large Telescope (E-ELT) and the Thirty Meter Telescope (TMT).

The requirement for larger telescopes comes from 2 factors. The first is the image distortion introduced by the limited diameter of the telescope. A point light source in an ideal system generates an Airy pattern image, shown in Figure 1.1. From the Airy pattern a resolution criterion can be defined. This resolution criterion is called the *Rayleigh resolution criterion*, which states that two light spots can only be distinguished from each other if the angular distance between them is bigger than $1.22\lambda/D$, where λ is the light wavelength and D is the aperture diameter, in the case of the telescope that is the telescope's diameter.

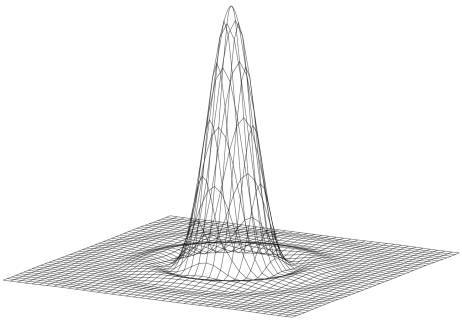


Figure 1.1: Intensity plot of the Airy Disk. Most of the energy is gathered in the central spot. Around the central spot there are dark rings, caused by light diffraction when light passes through a circular aperture.

The second factor is that a larger telescope is able to capture more light in a given time interval. This also increases the quality of the images obtained.

The image captured by a telescope is also influenced by the atmospheric turbulence. The turbulent air layers above the telescope introduce a disturbance on the incoming light wavefront. This disturbance reduces the telescopes resolution. One commonly used approach to compensate for the wavefront disturbance is to use an Adaptive Optics (AO) system.

In Figure 1.2 a schematic for an AO system is given. The main idea is to measure the incoming light wavefront distortion, and reshape it to a flat wavefront. This is necessary to make sharper pictures. The basic components are a measurement device, e.g. the Shack-Hartmann Sensor (SHS), a wavefront correction device, e.g. a Deformable Mirror (DM), and a computer that is able to determine the optimal control command for the mirror based on the measurements of the SHS, i.e. the commands that will result in the most flat wavefront possible.

For the E-ELT, the exact system specifications are not fully known, but approximate desired specifications can be given. One of the challenges in the development of the AO system is the required computational power. The control-loop for the E-ELT should run at a 1 kHz frequency. Considering the size of the mirror with 10^4

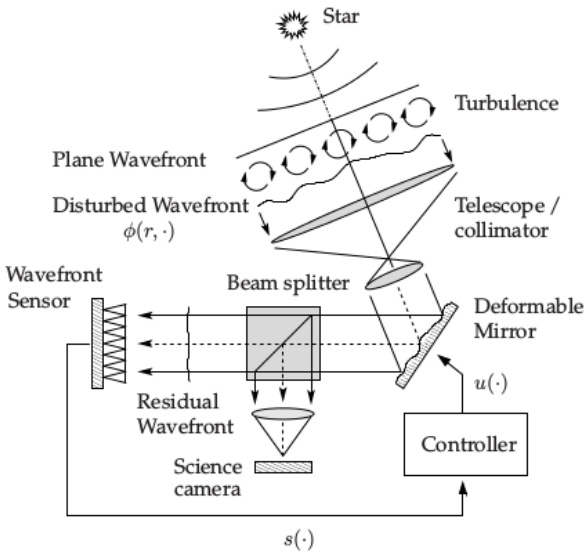


Figure 1.2: Schematic of an Adaptive Optics system for telescopes.

actuators and 4×10^4 measurements, that means that in less than 1 ms the data of 4×10^4 measurements have to be processed and the optimal value for 10^4 actuators have to be computed.

In traditional systems, this has been done with a Matrix Vector Multiplication (MVM) algorithm. In this algorithm, the relation between measurements and the optimal control sequence is given by a matrix-vector multiplication:

$$\hat{u} = Gy \tag{1.1}$$

where $\hat{u}$ is a vector containing the optimal commands that have to be sent to the actuators. y is a vector containing all the measurements and G is a matrix obtained in the mirror calibration, a process where the influence of each actuator on all the sensors is measured.

The matrix G is dense, i.e., most elements of the matrix are different from zero, and has a total size of $M \times N$, where M is the number of actuators and N the number of measurements. Three aspects of the required computational power are explained:

1. Number of computations per second, denoted by Floating Point Operations per Second (FLOPS).

2. Amount of memory required in bytes (B).

3. The required memory bandwidth, i.e., the amount of data that has to be transferred from the memory to the computational units, denote by Bytes per second (Bps).

The number of computations required on a matrix-vector multiplication is given by $2 \times N \times M$ Floating-point operations. For the E-ELT numbers mentioned before, the number of operations for one multiplication is 800×10^6 operations. With the specified control frequency of 1 kHz, the total number of operations is 800 GFLOPS.

The amount of memory required to store the matrix is the number of elements in the matrix times the size of each element. Real numbers in computers are usually given by single-precision or double-precision floating point numbers. The precision of single-precision numbers is lower, however they require less space, and the computation is usually faster. If single-precision numbers are used, the size of each element is 4 bytes. In the studied case, the amount of memory required is approximately 1.6 GB. The required size for double-precision is twice the size of the requirements for single-precision.

The required memory bandwidth is given by the size of the matrix (in bytes) times the control frequency. With the numbers considered so far, the memory bandwidth required is 1600 GBps for single-precision numbers.

The matrix-vector multiplication can be implemented to run efficiently on Graphical Processing Units (GPUs). Consider one of the most recent development in GPUs for scientific purposes, with the technical specification given in Table 1.1.

Feature	Unit	Value for K80	E-ELT requirements
Double-precision performance	TFLOPS	1.87	0.8
Single-precision performance	TFLOPS	5.6	0.8
Memory Bandwidth	GBytes/s	480	1600
Memory	GBytes	24	1.6

Table 1.1: Specifications of the most recent Kepler GPU from NVidia (K80), designed for scientific computing[1].

As it can be seen, the number of computations per second and the total amount of memory necessary are not a problem for this particular system. The bottleneck is the memory bandwidth. For less advanced GPU's, the number of computations and the amount of memory might also be a problem.

For this reason, much effort has been put into developing algorithms that require less computations, memory and memory bandwidth. Most of the effort has been put in wavefront reconstruction algorithms such as [1–3]. This algorithm has the task of determining what the current wavefront is, based on the measurements of a SHS. Less has been done on optimizing the computation of the optimal actuator commands.

One possible approach to reduce the requirements mentioned before is to compress the mirror spatial dynamics into a two dimensional state-space model, which will be introduced in the next chapter. In this thesis, the focus is on an identification algorithm for 2D Roesser models, and the possibility to apply these type of models to model the spatial dynamics of large-scale telescope mirrors.

The thesis starts with an introduction to Roesser models in Chapter 2. After that in Chapter 3 and Chapter 4 two subspace identification algorithms are derived, both based in the decomposition of a special class of Roesser models. In Chapter 5 future research work is suggested.

[1] Information available at `http://www.nvidia.com/object/tesla-servers.html`, acessed on 13-October-2015

2

2D ROESSER MODELS

2.1. INTRODUCTION TO 2D ROESSER MODELS

To describe the state behaviour for discrete 2 dimensional systems Roesser introduced a 2D state-space model in [4]. Initially this model was intended for imaging processing algorithms, but its application has also been studied in other multi-dimensional systems like magnetic maps and seismic models. It has also been used to describe systems with one time dimension and one spatial dimension, such as the modelling of heat-transfer systems [5].

In adaptive optics the model can be used to describe the mirror behaviour or to model the wavefront aberration. This makes it possible to create a compact model of the mirror, where the size of the mirror does not influence the size of the model. Important here is that the number of actuators on the mirror does influence the number of computations. A larger size is similar to a larger simulation time for a 1D state-space model.

The Roesser state-space model is given by the following equations:

$$\begin{aligned} \begin{bmatrix} x^h(i+1,j) \\ x^v(i,j+1) \end{bmatrix} &= \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u(i,j) \ , \\ y(i,j) &= \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} + Du(i,j) \ , \end{aligned} \tag{2.1}$$

where the indices i and j represent the horizontal and vertical indices, $x^h \in \mathbb{R}^{n_h}$ and $x^v \in \mathbb{R}^{n_v}$ represents the horizontal and vertical evolving state variable, $u \in \mathbb{R}^m$ represents the system's input and $y \in \mathbb{R}^l$ the system's output. And the system matrices are: $A_1 \in \mathbb{R}^{n_h \times n_h}$, $A_2 \in \mathbb{R}^{n_h \times n_v}$, $A_3 \in \mathbb{R}^{n_v \times n_h}$, $A_4 \in \mathbb{R}^{n_v \times n_v}$, $B_1 \in \mathbb{R}^{n_h \times m}$, $B_2 \in \mathbb{R}^{n_v \times m}$, $C_1 \in \mathbb{R}^{l \times n_h}$, $C_2 \in \mathbb{R}^{l \times n_v}$, and $D \in \mathbb{R}^{l \times m}$.

In this chapter, different concepts related to the Roesser models are explained. These concepts are important for understanding the identification algorithms presented later in the thesis.

2.2. COMPUTATION OF THE STATE-SPACE MODEL

The computation of the Roesser model can be done iteratively, similar to the 1D state-space model. It is also possible to compute the state at point (i_1, j_1) based on the initial state and the inputs, without the necessity to compute all the intermediate states.

First the initial-state is defined. The initial state can be separated in vertical and horizontal initial states, and are composed by:

$$x_0^h = \begin{bmatrix} x^h(0,0) & x^h(0,1) & x^h(0,2) & \cdots x^h(0,N_v) \end{bmatrix}^T \tag{2.2}$$

and

$$x_0^v = \begin{bmatrix} x^v(0,0) & x^v(1,0) & x^v(2,0) & \cdots x^v(N_h,0) \end{bmatrix}^T \tag{2.3}$$

where N_v is the number of vertical points of the 2D grid of the model, and N_h is the number of horizontal points. The initial conditions are also known as boundary conditions, and graphically represented in Figure 2.1.

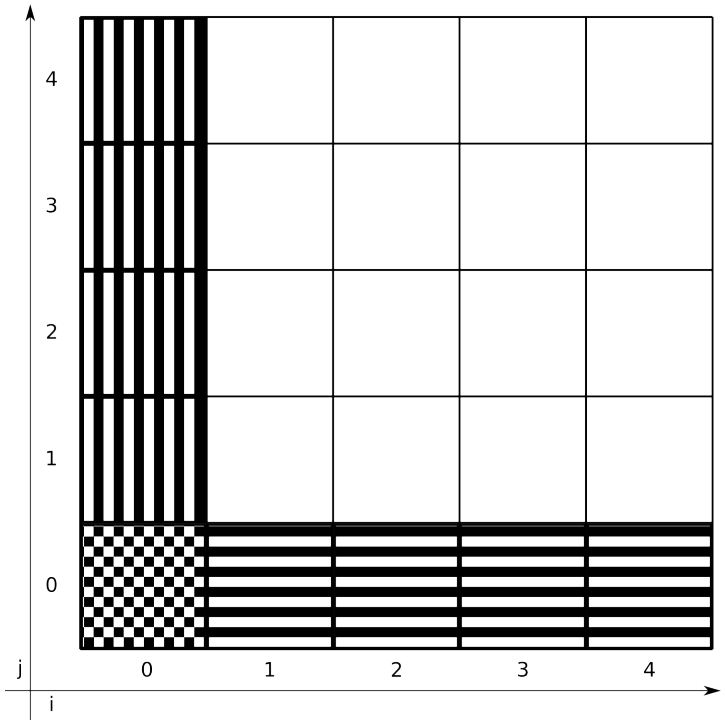


Figure 2.1: The model progress in horizontal (i) and vertical (j) directions. The blocks with horizontal lines have an initial vertical state $x^v(i,0)$ and the blocks with vertical lines have an initial horizontal state $x^h(0,j)$. The block at $(i,j)=(0,0)$ has both an initial vertical and horizontal state defined in the boundary conditions.

Roesser introduced the state-transition matrices $A^{i,j}$ for 2D systems, given by the following equations:

$$A^{0,0} = I \tag{2.4}$$

where I is the identity matrix of size $n_h + n_v$,

$$A^{0,1} = \begin{bmatrix} 0 & 0 \\ A_3 & A_4 \end{bmatrix} \quad , \quad A^{1,0} = \begin{bmatrix} A_1 & A_2 \\ 0 & 0 \end{bmatrix} \quad , \quad A^{1,1} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} , \tag{2.5}$$

$$A^{i,j} = A^{1,0}A^{i-1,j} + A^{0,1}A^{i,j-1} , \quad (i,j) > (0,0) \tag{2.6}$$

$$A^{-i,j} = A^{i,-j} = 0 , \quad \text{for} i \geq 1, j \geq 1. \tag{2.7}$$

If the system has no inputs, and the initial state is 0 except for $x^h(0,0)$ and $x^v(0,0)$, then the state at i,j is given by:

$$\begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} = A^{i,j} \begin{bmatrix} x^h(0,0) \\ x^v(0,0) \end{bmatrix} \tag{2.8}$$

One very important detail in this equation is that the matrix $A^{i,j}$ is given by the summation of multiple matrices. This is important in the creation of data-equations for sub-space identification.

The computation with different initial states and with inputs different from zero have been shown by Roesser as well:

$$\begin{aligned} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} &= \sum_{k=0}^{j} A^{i,j-k} \begin{bmatrix} x^h(0,k) \\ 0 \end{bmatrix} + \sum_{h=0}^{j} A^{i-h,j} \begin{bmatrix} 0 \\ x^v(h,0) \end{bmatrix} \\ &+ \sum_{h=0}^{j} \sum_{k=0}^{i} \left(A^{i-h-1,j-k} \begin{bmatrix} B_1 \\ 0 \end{bmatrix} + A^{i-h,j-k-1} \begin{bmatrix} 0 \\ B_2 \end{bmatrix} \right) u(h,k) \end{aligned} \tag{2.9}$$

The state at any point (i,j) is defined by multiple matrix summations. As mentioned before, this is very important to know when formulating the data-equations for subspace identification.

2.3. CONTROLLABILITY AND OBSERVABILITY

In 1D systems, the notions of observability and controllability are widely studied. See [11] for an introduction. Based on these concepts, it is possible to determine if a system is minimal or not.

For 2D systems, the same notions exist. Roesser introduced the concept of local observability and controllability [4].

Definition 2.3.1 (Local observability)**.** A Roesser model is locally observable if and only if for any given initial state $x(0,0) = \begin{bmatrix} x^h(0,0) & x^v(0,0) \end{bmatrix}^T$ and a set of inputs, the output of the model $y(i,j)$ with $(i,j) \geq (0,0)$ is not the same as when the initial state is different.

Definition 2.3.2 (Local controllability)**.** A Roesser model is locally controllable if and only if for a given initial state $x(0,0) = \begin{bmatrix} x^h(0,0) & x^v(0,0) \end{bmatrix}^T$ there exists an set of inputs and a pair $(i,j) \geq (0,0)$ such that any possible state $x(i,j)$ can be reached.

In [6] it has been shown that these concepts do not provide much information about the system minimality. A Roesser model can be locally controllable and observable without being minimal, but it can also be minimal without being locally controllable or observable.

One approach to study the system's observability and controllability is to decompose the model into two 1D systems, and apply the known concepts of observability and controllability to the 1D systems. In the next section two possible decompositions are demonstrated.

2.4. SEPARABILITY AND DECOMPOSITION OF 2D MODELS

Two dimensional models can also be represented in a transfer-function form given by:

$$H(z_h, z_v) = \frac{N(z_h, z_v)}{D(z_h, z_v)} \tag{2.10}$$

where z_h and z_v are the forward shift operators in each dimension, N and D are polynomials in both z_h and z_v.

The transfer function representation helps to understand the concept of separability of 2D systems. A Roesser model can be separable in denominator or separable in numerator.

The separable in numerator transfer-function is given by:

$$H(z_h, z_v) = \frac{N_h(z_h)N_v(z_v)}{D(z_h, z_v)}. \tag{2.11}$$

And the separable in denominator is given by:

$$H(z_h, z_v) = \frac{N(z_h, z_v)}{D_h(z_h)D_v(z_v)}. \tag{2.12}$$

Some of the separable models can be decomposed into two 1D systems, i.e., modelling separately the vertical and the horizontal behaviour.

2.4.1. SEPARABLE IN NUMERATOR

In [7] the decomposition of the Roesser model into two 1D state-space models interconnected in a feedback loop is studied. The model can be seen in Figure 2.2.

The horizontal **H** and vertical **V** systems are given by:

$$x_h(i+1,j) = A_h x_h(i,j) + B_h u_h(i,j)\,, \tag{2.13a}$$
$$y(i,j) = C_h x_h(i,j) + D_h u_h(i,j)\,, \tag{2.13b}$$

$$x_v(i,j+1) = A_v x_v(i,j) + B_v u_v(i,j)\,, \tag{2.13c}$$
$$y_v(i,j) = C_v x_v(i,j)\,. \tag{2.13d}$$

The relation between the decomposed model and the Roesser model is given by:

$$\begin{aligned} \begin{bmatrix} x^h(i+1,j) \\ x^v(i,j+1) \end{bmatrix} &= \begin{bmatrix} A_h & B_h C_v \\ B_v C_h & A_v + B_v D_h C_v \end{bmatrix} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} + \begin{bmatrix} B_h \\ B_v D_h \end{bmatrix} u(i,j) \\ y(i,j) &= \begin{bmatrix} C_h & D_h C_v \end{bmatrix} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} + D_h u(i,j) \end{aligned} \tag{2.14}$$

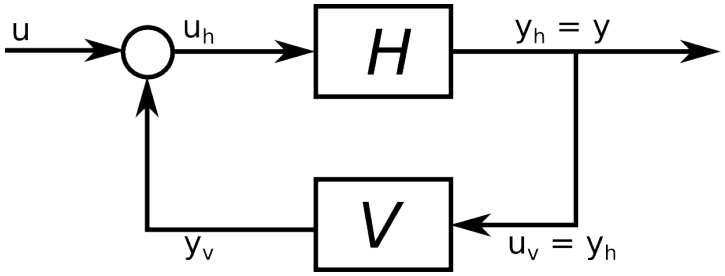


Figure 2.2: Decomposition of a Roesser model into two 1D systems.

Lemma 2.4.1. *A Roesser model as described in* (2.1) *can be decomposed as in Figure 2.2 with the system equations given in* (2.13) *if and only if:*

$$rank\begin{bmatrix} C_1 & D \end{bmatrix} = rank\begin{bmatrix} C_1 & D \\ A_3 & B_2 \end{bmatrix} \tag{2.15a}$$

and

$$rank\begin{bmatrix} B_1 \\ D \end{bmatrix} = rank\begin{bmatrix} B_1 & A_2 \\ D & C_2 \end{bmatrix}. \tag{2.15b}$$

Proof. See [7]. If the system in Figure 2.2 is a decomposition of a Roesser model given by (2.1), then the system matrices have to be identical, i.e.

$$\begin{gathered} A_h = A_1;\ \ B_h C_v = A_2;\ \ B_v C_h = A_3;\ \ A_v + B_v D_h C_v = A_4; \\ B_h = B_1;\ \ B_v D_h = B_2;\ \ C_h = C_1;\ \ D_h C_v = C_2;\ \ D_1 = D; \end{gathered} \tag{2.16}$$

For a given model A_1, A_2, A_3, A_4, B_1, B_2, C_1, C_2 and D it is possible to choose A_h, A_v, B_h, B_v, C_h, C_v and D_h if

$$B_v[C_1 \ \ D] = [A_3 \ \ B_2] \text{ and } \begin{bmatrix} B_1 \\ D \end{bmatrix} C_v = \begin{bmatrix} A_2 \\ C_2 \end{bmatrix}. \tag{2.17}$$

From the last equation, B_v and C_v can be computed if and only if (2.15a) and (2.15b) hold. □

Lemma 2.4.2. *The decomposition in Figure 2.2 leads to a separable in numerator transfer-function, i.e., a transfer function where $N(z_h, z_v)$ is separable in the variables z_h and z_v.*

Proof. This can be proved using the transfer functions from **H** and **V**:

$$H_h(z_h) = \frac{N_h(z_h)}{D_h(z_h)} \text{ and } H_v(z_v) = \frac{N_v(z_v)}{D_v(z_v)}. \tag{2.18}$$

The transfer-function of two systems connected in a feed-back is given by:

$$G(z_h, z_v) = \frac{H_h(z_h)}{1 + H_v(Z_v)H_h(z_h)} = \frac{N_h(z_h)D_v(z_v)}{D_h(z_h)D_v(z_v) + N_h(z_h)N_v(z_v)}. \tag{2.19}$$

□

Remark. Not all the separable in numerator models can be decomposed as in Figure 2.2. An example is given in [7]:

$$G(z_h, z_v) = \frac{(z_h + 1)(z_v + 1)}{z_h^2 z_v^2 + z_h z_v + 1}. \tag{2.20}$$

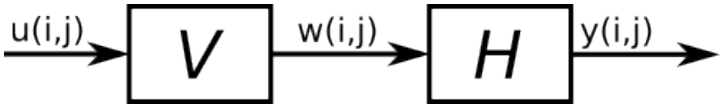


Figure 2.3: Decomposition of a CRSD model into two 1D systems.

2.4.2. SEPARABLE IN DENOMINATOR

In [8] another decomposition of the Roesser model is proposed. Instead of connecting the vertical and horizontal systems in a feedback loop, the systems are connected in series, as shown in Figure 2.3. Models that fit into this decomposition are a special class of Roesser models called Causal, Recursive and Separable in Denominator (CRSD) models.

The horizontal (**H**) and vertical (**V**) systems are given by:

$$x_v(i, j+1) = A_v x_v(i, j) + B_v u(i, j)\,, \tag{2.21a}$$
$$w(i, j) = C_v x_v(i, j) + D_v u(i, j)\,, \tag{2.21b}$$

$$x_h(i+1, j) = A_h x_h(i, j) + B_h w(i, j)\,, \tag{2.21c}$$
$$y(i, j) = C_h x_h(i, j) + D_h w(i, j)\,, \tag{2.21d}$$

where the intermediate signal $w \in \mathbb{R}^p$.

Lemma 2.4.3. *The Roesser model in* (2.1) *can be decomposed into two 1D systems as in Figure 2.3 with the system equations in* (2.21) *if the system matrix A_3 or A_2 is equal to zero.*

Proof. See [8]. To prove this, the two systems in Figure 2.3 are combined into one system:

$$x_v(i, j+1) = A_v x_v(i, j) + B_v u(i, j) \tag{2.22a}$$
$$w(i, j) = C_v x_v(i, j) + D_v u(i, j) \tag{2.22b}$$

And by replacing $w(i, j)$ in the horizontal equation with $C_v x_v(i, j) + D_v u(i, j)$:

$$x_h(i+1, j) = A_h x_h(i, j) + B_h C_v x_v(i, j) + B_h D_v u(i, j) \tag{2.22c}$$
$$y(i, j) = C_h x_h(i, j) + D_h C_v x_v(i, j) + D_h D_v u(i, j) \tag{2.22d}$$

And writing it in the Roesser model form:

$$\begin{aligned} \begin{bmatrix} x_h(i+1, j) \\ x_v(i, j+1) \end{bmatrix} &= \begin{bmatrix} A_h & B_h C_v \\ 0 & A_v \end{bmatrix} \begin{bmatrix} x_h(i, j) \\ x_v(i, j) \end{bmatrix} + \begin{bmatrix} B_h D_v \\ B_v \end{bmatrix} u(i, j) \\ y(i, j) &= \begin{bmatrix} C_h & D_h C_v \end{bmatrix} \begin{bmatrix} x_h(i, j) \\ x_v(i, j) \end{bmatrix} + \begin{bmatrix} D_h D_v \end{bmatrix} u(i, j) \end{aligned} \tag{2.23}$$

□

Remark. The proof shows that the resulting system has $A_3 = 0$. If you consider the horizontal system first as in Figure 2.4, the same proof can be used to show that $A_2 = 0$ in that case.

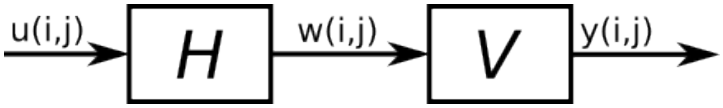


Figure 2.4: Decomposition of a CRSD model into two 1D systems. Horizontal system first.

Theorem 2.4.4. *A Roesser model can be decomposed into two interconnected 1D models as in Figure 2.3 if and only if the transfer-function is separable in denominator.*

Proof. To prove this theorem, the transfer function of a bivariate separable denominator function is given by:

$$H(z_h, z_v) = \frac{N(z_h, z_v)}{D_h(z_h)D_v(z_v)} = \frac{\sum_{j=0}^{J}\sum_{k=0}^{K} m_{jk} z_h^j z_v^k}{D_h(z_h)D_v(z_v)} = \frac{Z_h^J M Z_v^K}{D_h(z_h)D_v(z_v)} \tag{2.24}$$

where M is a matrix containing the coefficients m_{jk}, $Z_h^J = [1, z_h, \cdots, z_h^J]$, $Z_v^K = [1, z_h, \cdots, z_h^J]^T$ and $D_h(z_h)$ and $D_v(z_v)$ are univariate polynomials.

Now the decomposition is considered. The transfer-function for each 1D model is given by:

$$H_h(z_h) = \frac{Z_h^J F}{D_h(z_h)} \quad \text{and} \quad H_v(z_v) = \frac{G Z_v^K}{D_v(z_v)} \tag{2.25}$$

where $F = [F_0^T F_1^T \cdots F_J^T]^T$ and $G = [G_0 G_1 \cdots G_K]$ are matrices with sizes $(J+1) \times p$ and $p \times (K+1)$.

When connecting the two 1D systems, the transfer-function of the 2D model is given by the product between the transfer-functions from the 1D models:

$$H(z_h, z_v) = \frac{Z_h^J F}{D_h(z_h)} \frac{G Z_v^K}{D_v(z_v)} \tag{2.26}$$

The transfer function in (2.26) is equal to the transfer function in (2.24) when $FG = M$. Under that condition, any 2D CRSD model can be decomposed into a vertical and horizontal model. And the model in Figure 2.3 always leads to a CRSD model. □

Definition 2.4.1 (Minimal Decomposition)**.** As defined in [8], a *minimal decomposition* of the CRSD model is the decomposition with the lowest possible value of p, i.e. the lowest size for the intermediate signal $w(i, j) \in \mathbb{R}^p$.

Based on the definition, the following theorem is formulated in [8]:

Theorem 2.4.5. *The system given in Figure 2.3 is a minimal decomposition of a CRSD Roesser model if and only if:*

$$p = \text{rank} \begin{bmatrix} A_2 & B_1 \\ C_2 & D \end{bmatrix} \tag{2.27}$$

Proof. To prove this theorem, the relation between the original Roesser model and the decomposed system is written as:

$$A_1 = A_h \ , \ A_4 = A_v \ , \ C_1 = C_h \ , \ B_2 = B_v \tag{2.28}$$

and

$$\begin{bmatrix} A_2 & B_1 \\ C_2 & D \end{bmatrix} = \begin{bmatrix} B_h \\ D_h \end{bmatrix} \begin{bmatrix} C_v & D_v \end{bmatrix} \tag{2.29}$$

Using the Sylvester's rank inequality relation:

$$\text{rank} \begin{bmatrix} B_h \\ D_h \end{bmatrix} + \text{rank} \begin{bmatrix} C_v & D_v \end{bmatrix} - p \leq \text{rank} \begin{bmatrix} A_2 & B_1 \\ C_2 & D \end{bmatrix} \leq \min \left(\text{rank} \begin{bmatrix} B_h \\ D_h \end{bmatrix}, \text{rank} \begin{bmatrix} C_v & D_v \end{bmatrix} \right) \tag{2.30}$$

then, by choosing $\text{rank}[B_h \ \ D_h]^T = p$ and $\text{rank}[C_v \ \ D_v] = p$, no unnecessary terms are added to the intermediate signal w, and the decomposition is minimal. □

2.5. CAUSALITY FOR ROESSER MODELS

In 1D state-space models, causality is a well known concept. A system is said to be causal if the state at time instant k is only affected by past inputs. In a non-causal system, the state is also influenced by future inputs. This can be the case for data-compressing algorithms, and other types of signal processing.

In 2D systems however, causality is a more complex definition. Instead of defining past and future inputs, quarter-planes are defined as shown in Figure 2.5. And the terms causal and non-causal are replaced with the notion of plane causality, i.e., a system can have Quarter-Plane (QP), Half Plane (HP) or Full Plane (FP) causality.

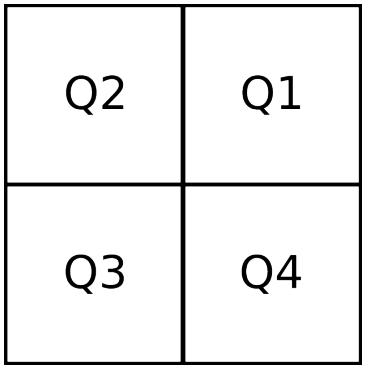


Figure 2.5: The 4 Quarter-planes for causality of 2D models. The causality of a model indicates which quarter-planes are influenced by the input at the middle of the complete square.

A 2D system is said to be Quarter-Plane causal, if the input $u(i,j)$ only has effect on the output $y(i',j')$ with $i' \geq i$ and $j' \geq j$, i.e, $(i',j') \in \mathbf{Q1}$. In a Half Plane system, the input $u(i,j)$ only has effect on the output $y(i',j')$ with $i' \geq i$, i.e., $(i',j') \in (\mathbf{Q1} \cup \mathbf{Q4})$. And for a Full-Plane case, the input $u(i,j)$ influences outputs in all the quadrants **Q1**, **Q2**, **Q3** and **Q4**.

The Roesser model described in (2.1) only describes a quarter-plane (QP) system.

2.6. IDENTIFICATION OF ROESSER MODELS

The use of Roesser models in the control of deformable mirrors requires some identification method that allows to create a model of the mirror based on input and output data. To model the mirror dynamics the inputs of the model are the voltages applied to the actuators, and the output is the wavefront phase introduced by the mirror.

Two identification approaches studied for 2D models are the subspace identification as studied in [9, 10] and parametric model estimation [5].

The subspace algorithm from [9, 10] is limited to CRSD models. One of the reasons is that for the full Roesser models, the formulation of the data-equations necessary in subspace identification is far from trivial.

When writing the data-equations, the first question is how to consider the two different dimensions in the Hankel matrices. One possible approach is to consider Hankel matrices for the columns:

$$\mathscr{H}_{s,N}(i,j) = \begin{bmatrix} u(i,j) & u(i,j+1) & \cdots & u(i,j+N-1) \\ u(i,j+1) & u(i,j+2) & \cdots & u(i,j+N) \\ \vdots & \vdots & \ddots & \vdots \\ u(i,j+s) & u(i,j+s+1) & \cdots & u(i,j+N+s-2) \end{bmatrix} \tag{2.31}$$

where N is the number of columns, and s the number of rows of the Hankel matrix.

Then, the row Hankel matrices are added to block Hankel matrices:

$$\mathscr{U}_{i,j,s,N} = \begin{bmatrix} \mathscr{H}_{s,N}(i,j) & \mathscr{H}_{s,N}(i+1,j) & \cdots & \mathscr{H}_{s,N}(i+N-1,j) \\ \mathscr{H}_{s,N}(i+1,j) & \mathscr{H}_{s,N}(i+2,j) & \cdots & \mathscr{H}_{s,N}(i+N,j) \\ \vdots & \vdots & \ddots & \vdots \\ \mathscr{H}_{s,N}(i+s,j) & \mathscr{H}_{s,N}(i+s+1,j) & \cdots & \mathscr{H}_{s,N}(i+N+s,j) \end{bmatrix} \tag{2.32}$$

One problem is that the influence of the input $u(i_1,j_1)$ on the output $y(i_2,j_2)$ cannot be expressed only by matrix multiplications, but it also requires matrix summations. This can be seen in (2.9). These summations make it difficult to estimate the system matrices, even when a data-equation is formulated.

SUB-SPACE IDENTIFICATION OF CRSD MODELS

In [9] an algorithm for the identification of 2D CRSD models has been described. The goal is to estimate approximate system matrices A_1, A_2, A_4, B_1, B_2, C_1, C_2 and D from an input-output dataset.

This algorithm relies on the fact that the influence from the input $u(i_1,j_1)$ on the output $y(i_2,j_2)$ can be described by matrix multiplications only in a CRSD model, and by the fact that it is possible to decouple vertical and horizontal information, and use the vertical information as an input of the horizontal data-equations.

One assumption made in order to decouple the vertical system is that the initial states, i.e. the boundary conditions for the vertical states is 0. By decoupling the systems, data-equations can be formulated, and the sub-space identification can be performed with known techniques available in Numerical algorithms for Subspace State-Space System IDentification (N4SID) [11].

PARAMETRIC IDENTIFICATION USING LFT

Another identification approach is to use parametric identification as described in [5]. The algorithm relies on the strong link between the Roesser model and the Linear Fractional Representation (LFR).

In the parametric approach, the system is parametrized as a function of the vector Θ. The goal is to choose an ideal Θ such that the error between the output of the estimated model and the output data is minimal.

The big drawbacks of this identification algorithm are two. The first one is that some initial knowledge about the system order is necessary. The second drawback is that it relies on a non-convex optimization problem. This can cause the optimization to fail to find the best solution due to local minima. A good initial guess of the system is necessary for a good system estimate.

This algorithm can be combined with subspace identification methods. In that case the subspace method defines the size of the model and an initial estimate of the system matrices, and the LFT algorithm improves the model fit.

3

SYSTEM IDENTIFICATION FOR SEPARABLE IN NUMERATOR MODELS

A subset of the class of Roesser models is considered in the present chapter. As seen in Section 2.4, a special case of separable in numerator models can be decomposed into two 1D systems interconnected in a feedback loop. In this chapter an identification algorithm for this class of Roesser models is derived.

It is also shown that the derived algorithm can be applied to a more general class of 2D Roesser models, by providing a good initial guess for the non-linear LFT algorithm described in [5].

The chapter starts with the formulation of the identification problem in Section 3.1. After that the data-equations for the sub-space identification algorithm are derived in Section 3.2. In Section 3.3 the minimization problem of the algorithm is formulated. After that, the methods for estimating the system matrices are explained in Section 3.4, and the best estimation is chosen according to the description in Section 3.5. A summary of the algorithm is given in Section 3.6, together with some implementation details. In Section 3.7, it is explained how the code was optimized by using another solver. The results are presented in Section 3.8 and discussed in Section 3.9.

3.1. FORMULATION OF THE IDENTIFICATION PROBLEM

THE CLASS OF ROESSER MODELS BEING CONSIDERED

The decomposed Roesser model is shown in Figure 3.1, and the decoupled model is given by:

$$x_h(i+1,j) = A_h x_h(i,j) + B_h u_h(i,j), \tag{3.1a}$$
$$y(i,j) = C_h x_h(i,j) + D_h u_h(i,j), \tag{3.1b}$$

$$x_v(i,j+1) = A_v x_v(i,j) + B_v u_v(i,j), \tag{3.1c}$$
$$y_v(i,j) = C_v x_v(i,j), \tag{3.1d}$$

$$u_h(i,j) = y_v(i,j) + u(i,j), \tag{3.1e}$$

with $x^h \in \mathbb{R}^{n_h}$, $x^v \in \mathbb{R}^{n_v}$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^l$, and the matrices A_h, A_v, B_h, B_v, C_h, C_v and D_h are real matrices of appropriate dimensions.

THE SIMILARITY TRANSFORMATION

Before defining the identification problem, the concept of similarity transformation is explained, as it is required to understand it in the identification algorithm.

There are different matrices that describe the same relation between input and output for the subsystems in (3.1), and all the models are valid. The similarity transformation for a 1D state-space model is a linear transformation of the state-variable. The similarity transformation for the horizontal subsystem is given:

$$x_h(i,j) = T_h x_{Th}(i,j), \tag{3.2}$$

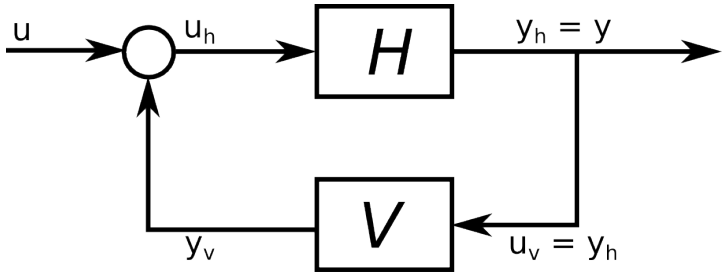


Figure 3.1: Decomposition of a Roesser model into two 1D systems. The system equations are given in (3.1).

where T_h is an invertible matrix.

By replacing the new state-variable in the state-space model, the relation between a system and a similar system is:

$$x_{Th}(i,j) = T_h^{-1} A_h T_h x_{Th}(i,j) + T_h^{-1} B_h u_h(i,j) \ , \tag{3.3}$$

$$y_h(i,j) = C_h T_h x_{Th}(i,j) + D_h u_h(i,j). \tag{3.4}$$

And now the system matrices A_{Th}, B_{Th}, C_{Th} and D_{Th} are defined by:

$$A_{Th} = T_h^{-1} A_h T_h,$$

$$B_{Th} = T_h^{-1} B_h,$$

$$C_{Th} = C_h T_h,$$

$$D_{Th} = D_h.$$

A very important property of this transformation, is that it describes the same system, but with new state-variables. Using this property it can be concluded that estimating a system up to a similarity transformation is sufficient to describe the system.

THE IDENTIFICATION PROBLEM

The identification problem can now be formulated.

Consider an input-output dataset:

$$\left\{u(i,j), y(i,j)\right\}_{(i,j)=(1,1)}^{(N_h,N_v)}$$

with $N_h > n_h$ and $N_v > n_v$. Further a SISO system is considered, thus $u(i,j) \in \mathbb{R}$ and $y(i,j) \in \mathbb{R}$.

Some conditions are required and assumptions made about the input-output dataset. The first criterion is related to the definition of persistence of excitation.

Definition 3.1.1. According to [11], a signal $u(k), k = 1,2,\ldots$ is persistently exciting of order n if and only if there is an integer N such that the matrix:

$$\mathbf{U}_{0,n,N} = \begin{bmatrix} u(0) & u(1) & \cdots & u(N-1) \\ u(1) & u(2) & \cdots & u(N) \\ \vdots & \vdots & \ddots & \vdots \\ u(n-1) & u(n) & \cdots & u(N+n-2) \end{bmatrix} \tag{3.5}$$

has full rank n.

For this identification problem, the signals u_v and u_h have to be persistently exciting. This conditions cannot be ensured in the experiment, but can be used to discard an input-output dataset.

Assuming the dataset was retrieved from an identification experiment with a Single-Input Single-Output (SISO) Roesser model decomposed as shown in Figure 3.1, the identification problem is to estimate the system orders n_v, n_h and the system matrices ($\hat{A}_{Th}$, $\hat{A}_{Tv}$, $\hat{B}_{Th}$, $\hat{B}_{Tv}$, $\hat{C}_{Th}$, $\hat{C}_{Tv}$ and $\hat{D}_{Th}$) that define the decomposed model:

$$\hat{x}_{Th}(i+1,j) = \hat{A}_{Th}\hat{x}_{Th}(i,j) + \hat{B}_{Th}\hat{u}_h(i,j), \tag{3.6a}$$

$$\hat{y}(i,j) = \hat{C}_{Th}\hat{x}_{Th}(i,j) + \hat{D}_{Th}\hat{u}_h(i,j), \tag{3.6b}$$

and

$$\hat{x}_{Tv}(i, j+1) = \hat{A}_{Tv}\hat{x}_{Tv}(i, j) + \hat{B}_{Tv}u_v(i, j), \tag{3.6c}$$

$$\hat{y}_v(i, j) = \hat{C}_{Tv}\hat{x}_{Tv}(i, j), \tag{3.6d}$$

such that

$$\hat{u}_h(i, j) = \hat{y}_v(i, j) + u(i, j). \tag{3.6e}$$

Remark. In this identification approach, it is also necessary to estimate the signals y_v and u_h, although this is not a primary goal of the algorithm.

Using the estimated matrices, an estimate of the Roesser model is given by:

$$\begin{aligned}\begin{bmatrix}\hat{x}_{Th}(i+1, j)\\ \hat{x}_{Tv}(i, j+1)\end{bmatrix} &= \begin{bmatrix}\hat{A}_{Th} & \hat{B}_{Th}\hat{C}_{Tv}\\ \hat{B}_{Tv}\hat{C}_{Th} & \hat{A}_{Tv} + \hat{B}_{Tv}\hat{D}_{Th}\hat{C}_{Tv}\end{bmatrix}\begin{bmatrix}\hat{x}_{Th}(i, j)\\ \hat{x}_{Tv}(i, j)\end{bmatrix} + \begin{bmatrix}\hat{B}_{Th}\\ \hat{B}_{Tv}\hat{D}_{Th}\end{bmatrix}u(i, j)\\ y(i, j) &= \begin{bmatrix}\hat{C}_{Th} & \hat{D}_{Th}\hat{C}_{Tv}\end{bmatrix}\begin{bmatrix}\hat{x}_{Th}(i, j)\\ \hat{x}_{Tv}(i, j)\end{bmatrix} + \hat{D}_{Th}u(i, j)\end{aligned} \tag{3.7}$$

3.2. DATA EQUATIONS FOR SUBSPACE IDENTIFICATION

The first step in the subspace identification algorithms is the formulation of the data equation. For the decomposed model being identified, two data equations are formulated. In this section the formulation of the data equation for the horizontal case is detailed, but the steps are similar for the vertical case.

The input/output data is structured in Hankel matrices, with s block rows, where $s > n$. The number of columns is chosen so that $N + s - 1 = N_h$ and let $j \in [1, N_v]$ be an index denoting the row where the data is selected from.

$$\mathbf{Y}^h_{j,s,N} = \begin{bmatrix} y_h(1, j) & y_h(2, j) & \cdots & y_h(N, j)\\ y_h(2, j) & y_h(3, j) & \cdots & y_h(N+1, j)\\ \vdots & \vdots & \ddots & \vdots\\ y_h(s, j) & y_h(s+1, j) & \cdots & y_h(N_h, j)\end{bmatrix}, \tag{3.8}$$

$$\mathbf{U}^h_{j,s,N} = \begin{bmatrix} u_h(1, j) & u_h(2, j) & \cdots & u_h(N, j)\\ u_h(2, j) & u_h(3, j) & \cdots & u_h(N+1, j)\\ \vdots & \vdots & \ddots & \vdots\\ u_h(s, j) & u_h(s+1, j) & \cdots & u_h(N_h, j)\end{bmatrix}. \tag{3.9}$$

The state sequence is given by:

$$\mathbf{X}^h_{j,N} = \begin{bmatrix} x_h(1, j) & x_h(2, j) & \cdots & x_h(N, j)\end{bmatrix}. \tag{3.10}$$

Then the extended observability matrix and the Toeplitz matrix $\mathscr{T}^h_s$ are defined as:

$$\mathscr{O}^h_s = \begin{bmatrix} C_h\\ C_hA_h\\ C_hA_h^2\\ \vdots\\ C_hA_h^{s-1}\end{bmatrix} \quad \text{and} \quad \mathscr{T}^h_s = \begin{bmatrix} D_h & 0 & 0 & \cdots & 0\\ C_hB_h & D_h & 0 & \cdots & 0\\ C_hA_hB_h & C_hB_h & D_h & & 0\\ \vdots & & \ddots & \ddots & \\ C_hA_h^{s-2}B_h & C_hA_h^{s-3}B_h & \cdots & C_hB_h & D_h\end{bmatrix}. \tag{3.11}$$

Then the relation between the matrices is derived similarly as done in [11] and is given by:

$$\mathbf{Y}^h_{j,s,N} = \mathscr{O}^h_s\mathbf{X}^h_{j,s,N} + \mathscr{T}^h_s\mathbf{U}^h_{j,s,N}. \tag{3.12}$$

As mentioned in [12], the data equation as formulated in (3.12) has some key structural properties worth emphasising:

1. The matrix product $\mathscr{O}^h_s\mathbf{X}^h_{i,j,s,N}$ is low rank since $s > n_h$.

2. The matrix $\mathscr{T}_s^h$ is block Toeplitz.

Similar to the Nuclear Norm Subspace IDentification (N2SID) algorithm in [12, 13], these properties will be used in the identification process proposed in this chapter.

The vertical system is very similar. The difference is how the data is indexed in the input and output Hankel matrices. To show the difference, the Hankel matrix $\mathbf{U}_{i,s,N}^v$ is shown:

$$\mathbf{U}_{i,s,N}^v = \begin{bmatrix} u_v(i,1) & u_v(i,2) & \cdots & u_v(i,N) \\ u_v(i,2) & u_v(i,3) & \cdots & u_v(i,N+1) \\ \vdots & \vdots & \ddots & \vdots \\ u_v(i,s) & u_v(i,s+1) & \cdots & u_v(i,N_v) \end{bmatrix}. \tag{3.13}$$

The data equation for the vertical system is given by:

$$\mathbf{Y}_{i,s,N}^v = \mathscr{O}_s^v \mathbf{X}_{i,s,N}^v + \mathscr{T}_s^v \mathbf{U}_{i,s,N}^v. \tag{3.14}$$

CONCATENATING DATASETS

In the identification problem, the equality constraint $y_v(i,j) + u(i,j) = u_h(i,j)$ has to hold.

When using the data-equations (3.12) and (3.14) for the subspace identification, the only point where this constraint can be applied is at (i,j) as can be seen in Figure 3.2.

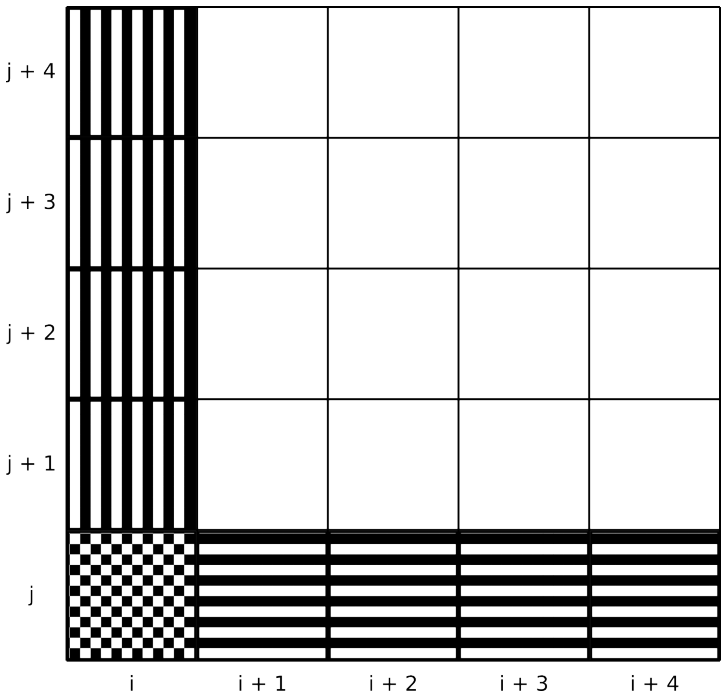


Figure 3.2: For an starting (i,j) coordinate, the data used in the vertical an horizontal Hankel matrices are different. For the horizontal system the row j is used, and for the vertical system the column i is used. The only point where the equality constraint $u_h(i,j) = y_v(i,j) + u(i,j)$ can be applied is at point (i,j). At all other points, the variables y_v and u_h are free to chose.

Considering the horizontal system, adding more rows to the data equation is similar to adding more input/output datasets to a 1D subspace identification algorithm like MOESP [11, 14]. The same holds for adding more columns to the vertical data equations.

Adding datasets to subspace identification algorithms is done by concatenating the Hankel matrix of each dataset. As an example, the data equation for the horizontal system is given:

$$\begin{aligned} \begin{bmatrix} \mathbf{Y}_{1,s,N}^h & \mathbf{Y}_{2,s,N}^h & \cdots & \mathbf{Y}_{N_v,s,N}^h \end{bmatrix} = \mathscr{O}_s^h \begin{bmatrix} \mathbf{X}_{1,s,N}^h & \mathbf{X}_{2,s,N}^h & \cdots & \mathbf{X}_{N_v,s,N}^h \end{bmatrix} + \\ \mathscr{T}_s^h \begin{bmatrix} \mathbf{U}_{1,s,N}^h & \mathbf{U}_{2,s,N}^h & \cdots & \mathbf{U}_{N_v,s,N}^h \end{bmatrix}. \end{aligned} \tag{3.15}$$

To denote the concatenated matrices the following notation is used:

$$\underline{\mathbf{U}}^h_{s,N} = \begin{bmatrix} \mathbf{U}^h_{1,s,N} & \mathbf{U}^h_{2,s,N} & \dots & \mathbf{U}^h_{N_v,s,N} \end{bmatrix}. \tag{3.16}$$

The data-equation for the horizontal case is then formulated as:

$$\underline{\mathbf{Y}}^h_{s,N} = \mathscr{O}^h_s \underline{\mathbf{X}}^h_{s,N} + \mathscr{T}^h_s \underline{\mathbf{U}}^h_{s,N}. \tag{3.17}$$

Note that in this new formulation of the data equation, the two structural properties still hold, i.e.:

1. The matrix product $\mathscr{O}^h_s \underline{\mathbf{X}}^h_{s,N}$ is low rank since $s > n_h$.
2. The matrix $\mathscr{T}^h_s$ is block Toeplitz.

A second important remark is on the size of the matrix $\underline{\mathbf{Y}}^h_{s,N}$. Keeping the notation used so far, the number of rows for SISO models is given by s, and the number of columns is given by $(s+N-1)N$.

3.3. THE MINIMIZATION PROBLEM

In the data-equations (3.12) and (3.14), the signals y_v and u_h are unknowns. But the relation between the unknown signals and the the known signals can be given by $u_h = u + y_v$.

Using the property that the matrix product $\mathscr{O}^h_s \underline{\mathbf{X}}^h_{s,N}$ is low rank, a minimization problem can be formulated to minimize the rank of $\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}^v_s \underline{\mathbf{U}}^v_{s,N}$ and the rank of $\underline{\mathbf{Y}}^h_{s,N} - \mathscr{T}^h_s \underline{\mathbf{U}}^h_{s,N}$

The minimization problem is given by:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N}, \underline{\mathbf{U}}^h_{s,N}, \mathscr{T}^v_s, \mathscr{T}^h_s} \quad & \operatorname{rank}\left(\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}^v_s \underline{\mathbf{U}}^v_{s,N}\right) + \operatorname{rank}\left(\underline{\mathbf{Y}}^h_{s,N} - \mathscr{T}^h_s \underline{\mathbf{U}}^h_{s,N}\right) \\ \text{subject to} \quad & y_v(i,j) + u(i,j) = u_h(i,j) \ . \end{aligned} \tag{3.18}$$

This optimization problem is not tractable, due to the rank operator. As seen in [12, 13, 15], a convex relaxation of the problem makes it tractable, and solvable with convex optimization algorithms.

The nuclear norm of a matrix, defined as the sum of the singular values of a matrix, and denoted by the operator $||M||_*$ is known to be a convex relaxation of the rank operator [12, 13, 15] and can be used to make the optimization problem in (3.18) convex.

Now the optimization problem can be written as:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N}, \underline{\mathbf{U}}^h_{s,N}, \mathscr{T}^v_s, \mathscr{T}^h_s} \quad & \left|\left|\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}^v_s \underline{\mathbf{U}}^v_{s,N}\right|\right|_* + \left|\left|\underline{\mathbf{Y}}^h_{s,N} - \mathscr{T}^h_s \underline{\mathbf{U}}^h_{s,N}\right|\right|_* \\ \text{subject to} \quad & y_v(i,j) + u(i,j) = u_h(i,j) \ . \end{aligned} \tag{3.19}$$

This optimization problem is bilinear, since the optimization variables $\mathscr{T}^h_s$ and $\underline{\mathbf{U}}^v_{s,N}$ appear as a multiplication.

The approach used to avoid the bilinear problem, was to consider the inverse of the horizontal model (**H**), given by:

$$\begin{aligned} x_I(i+1,j) &= A_I x_I(i,j) + B_I u_I(i,j) \\ y_I(i,j) &= C_I x_I(i,j) + D_I u_I(i,j) \end{aligned} \tag{3.20}$$

where $A_I = \left(A_h - B_h D_h^{-1} C_h\right)$, $B_I = B_h D_h^{-1}$, $C_I = -C_h D_h^{-1}$, $D_I = D_h^{-1}$, $u_I(i,j) = y_h(i,j)$, $y_I(i,j) = u_h(i,j)$ and $x_I = x_h$.

Note that the inversion is only possible if D_h is invertible. So besides the assumptions made before about the system, it is also assumed that $D_h \neq 0$.

When considering the inverse model, the identification problem becomes the identification of two models in parallel with the same input as seen in Figure 3.3.

The new optimization problem is given by:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N}, \underline{\mathbf{Y}}^I_{s,N}, \mathscr{T}^v_s, \mathscr{T}^I_s} \quad & \left|\left|\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}^v_s \underline{\mathbf{U}}^v_{s,N}\right|\right|_* + \left|\left|\underline{\mathbf{Y}}^I_{s,N} - \mathscr{T}^I_s \underline{\mathbf{U}}^I_{s,N}\right|\right|_* \\ \text{subject to} \quad & y_v(i,j) + u(i,j) = y_I(i,j) \ . \end{aligned} \tag{3.21}$$

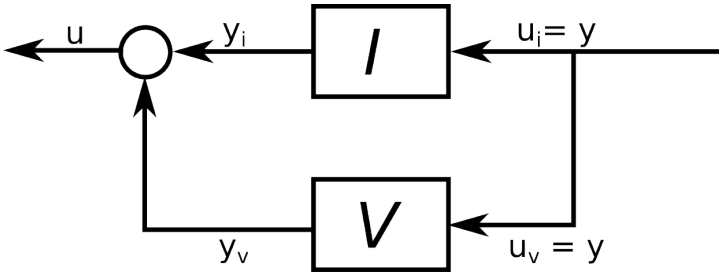


Figure 3.3: Block scheme for the identification of the inverse system.

The equality constraint has been enforce by adding a penalty to the minimization problem. By doing this the minimization problem becomes:

$$\min_{\underline{\mathbf{Y}}^{v}_{s,N},\underline{\mathbf{Y}}^{I}_{s,N},\mathscr{T}_s^{v},\mathscr{T}_s^{I}} \left|\left|\underline{\mathbf{Y}}^{v}_{s,N}-\mathscr{T}_s^{v}\underline{\mathbf{U}}^{v}_{s,N}\right|\right|_* + \left|\left|\underline{\mathbf{Y}}^{I}_{s,N}-\mathscr{T}_s^{I}\underline{\mathbf{U}}^{I}_{s,N}\right|\right|_* + \lambda(||y_I(i,j)-y_v(i,j)-u(i,j)||_2^2) \tag{3.22}$$

where $\lambda \in [0,\infty)$ is a regularization parameter.

The problem formulated in (3.22) is very similar to the optimization problem formulated in the N2SID algorithm in [12]. This problem is convex, and by searching through λ, a whole set of optimal solutions can be found.

Only the solution which best approximates the actual system is of interest. To test which solution is the best, it is first necessary to estimate the system matrices out of the optimal solution.

3.4. CALCULATION OF THE SYSTEM MATRICES

This section explains how the system matrices are computed from the solution of the optimization problem.

3.4.1. ESTIMATING THE SYSTEM MATRICES USING THE EXTENDED OBSERVABILITY MATRIX

In this subsection, the computation of the system matrices is described, based on N2SID and other subspace algorithms. This is done based on the data-equation for the inverse horizontal system. The same principles are applied to the vertical system.

As described in [12], the first step is to compute the Singular Value Decomposition (SVD) of the estimated matrix as follows:

$$\left[\underline{\mathbf{Y}}^{I}_{s,N}-\mathscr{T}_s^{I}\underline{\mathbf{U}}^{I}_{s,N}\right] = \begin{bmatrix} U_{\hat{n}_h} & \star \end{bmatrix} \begin{bmatrix} \Sigma_{\hat{n}_h} & 0 \\ 0 & \star \end{bmatrix} \begin{bmatrix} V^T_{\hat{n}_h} \\ \star \end{bmatrix} \tag{3.23}$$

where $\hat{n}_h$ is an integer that denotes the $\hat{n}_h$ largest singular values and $\star$ denotes a matrix which is compatible and the contents are of no interest here.

The value of $\hat{n}_h$ can be automatically computed. This is done by ordering the singular values in descending order and then select the index of the singular value which is the closest to logarithmic mean of the first and the last singular value, i.e., the highest and the lowest singular value.

Then the system matrices $\hat{A}_{TI}$ and $\hat{C}_{TI}$ can be extracted from $U_{\hat{n}_h}$, as shown in [11]. To compute the system matrices the notation $M(m:n,p:q)$ is used to specify the sub-matrix of M with rows m to n and columns p to q. To specify a complete row or column only : is necessary.

The matrix $U_{\hat{n}_h}$ can be considered an approximation of the extended observability matrix $\mathscr{O}_s^I$. The matrix $\hat{C}_{TI}$ corresponds to the first m rows of $U_{\hat{n}_h}$. The matrix $\hat{A}_{TI}$ can then be computed by solving an overdetermined set of equations given by:

$$U_{\hat{n}_h}(1:(s-1)m,:)\hat{A}_{TI} = U_{\hat{n}_h}(m+1:sm,:). \tag{3.24}$$

The matrix $\hat{B}_{TI}$ can be computed by solving another overdetermined set of equations given by:

$$U_{\hat{n}_h}(1:(s-1)m,:)\hat{B}_{TI} = \mathscr{T}_s^I(m+1:ms,1:l). \tag{3.25}$$

The system matrix $\hat{D}_{TI}$ can be extracted directly from the Toeplitz matrix $\mathscr{T}_s^I$.

$$\hat{D}_{TI} = \mathscr{T}_s^I(1:m, 1:l). \tag{3.26}$$

The same technique can be applied to estimate the matrices of the vertical system $(\hat{A}_{Tv}, \hat{B}_{Tv}, \hat{C}_{Tv})$.

3.4.2. ESTIMATING THE SYSTEM MATRICES USING THE TOEPLITZ MATRIX

The first column of $\mathscr{T}_s^I$ contains the so called Markov parameters. A Hankel matrix $\mathscr{H}_h$ with the Markov parameters, except the first one, i.e. D_h, is defined as:

$$\mathscr{H}_h := \begin{bmatrix} C_{TI}B_{TI} & C_{TI}A_{TI}B_{TI} & \dots & C_{TI}A_{TI}^{l_1}B_{TI} \\ C_{TI}A_{TI}B & C_{TI}A_{TI}^2B_{TI} & \dots & C_{TI}A_{TI}^{l_1+1}B_{TI} \\ \vdots & \vdots & \ddots & \vdots \\ C_{TI}A_{TI}^{l_2}B & C_{TI}A_{TI}^{l_2+1}B_{TI} & \cdots & C_{TI}A_{TI}^{l_1+l_2}B_{TI} \end{bmatrix} = \begin{bmatrix} C_{TI} \\ C_{TI}A_{TI} \\ \vdots \\ C_{TI}A_{TI}^{l_2} \end{bmatrix} \begin{bmatrix} B_{TI} & A_{TI}B_{TI} & \dots & A_{TI}^{l_1}B_{TI} \end{bmatrix} \tag{3.27}$$

where $l_1 + l_2 + 1 \leq s$. We prove with Sylvester's inequality that $rank(\mathscr{H}_h) = \hat{n}_h$ and therefore $l_1 > \hat{n}_h$.

A SVD of $\mathscr{H}$ give rise to the triplet (U, S, V). From the singular values in S we can determine the system order as done in subsection 3.4.1.

The matrices, known up to a similarity transformation, are computed with :

$$\begin{aligned} U_{\hat{n}_h} &= U(:, 1:\hat{n}_h), \\ \hat{C}_{TI} &= U_{\hat{n}_h}(1:p, :), \\ \hat{A}_{TI} &= U_{\hat{n}_h}(1:(s-1)p, :)^{-1} U_{\hat{n}_h}(p+1:sp, :), \\ \hat{B}_{TI} &= S(1:\hat{n}_h, 1:\hat{n}_h) V^T(1:\hat{n}_h, 1:m). \end{aligned}$$

The matrix $\hat{D}_{TI}$ is computed in the same way as in subsection 3.4.1.

One drawback of this approach is that the number of singular values of the matrix $\mathscr{H}$ is less or equal to half the value of s in the Hankel matrices $\underline{\mathbf{Y}}_{s,N}^v - \mathscr{T}_s^v \underline{\mathbf{U}}_{s,N}^v$ and $\underline{\mathbf{Y}}_{s,N}^v - \mathscr{T}_s^v \underline{\mathbf{U}}_{s,N}^v$. That means that in this approach the value chosen for s should be $s \geq 2n_h$.

Again, the approach describes the identification of the system matrices for the horizontal system, but the same technique can be applied for the vertical one.

3.5. SELECTING THE BEST SYSTEM ESTIMATION

The minimization problem (3.22) contain a regularization term λ. In the identification process multiple λ values are used, and the best result is selected as the final result of the algorithm.

To select the best result, a second dataset is considered. The model estimated with a specific value of λ is simulated with the input-data.

A common measurement for the difference between an estimated signal and the actual signal is the root-mean-square error (RMSE). For one experiment of a 2D system it is given by:

$$\text{RMSE}(y(i,j), \hat{y}(i,j)) = \sqrt{\frac{\sum_{(i,j)=(1,1)}^{(N_h,N_v)} (y(i,j) - \hat{y}(i,j))^2}{N_h N_v}} \tag{3.28}$$

where y is the original signal, $\hat{y}$ is the signal obtained from the estimated model, N_h is the number of columns in the experiment, and N_v the number of rows.

To select the ideal model, the RMSE is computed for each model, i.e., the model estimated for each λ value. The model with the lowest RMSE is considered the best one.

3.6. SUMMARY AND IMPLEMENTATION OF THE ALGORITHM

The implemented algorithm is fully described in Algorithm 1.

The algorithm has been implemented in MATLAB, with the use of CVX [16] to solve the optimization problem (3.22). The code is delivered on a CD along with the report.

The results of the algorithm will be discussed later in Section 3.8. The initial implementation with CVX was considered slow, taking approx. 1 hour for a single identification run for Hankel matrices of size $s = 12$ and $N = 12$, and 13 lambda values to be tested.

Algorithm 1 Identification algorithm for the feed-back decomposition of the Roesser model.

initialization:
  Define identification parameters s, N, and a set of λ values $\Lambda = \text{logspace}(\log(\lambda_{min}), \log(\lambda_{max}), steps)$.
  Create the Hankel matrices for the data-equations with dataset 1.
for each $\lambda \in \Lambda$ **do**
  Solve the optimization problem (3.22).
  Compute vertical and inverse system matrices using the algorithms in Section 3.4.
  Compute the inverse of system I, to get H.
  Compute Roesser model, from the estimated H and V system matrices.
  Simulate the estimated model using dataset 2.
  Compute the RMSE of the output of the simulation and the output of dataset 2.
end for
Select the estimated system with the lowest RMSE.

3.7. OPTIMIZING THE COMPUTATIONS

In the first tests of the algorithm, the CVX implementation was considered slow, taking more than one hour to estimate a system, depending on the sizes of the Hankel matrices. The experiments were performed with MATLAB 2015b on an Ubuntu 14.04 installation on a computer with an Intel® Core™ i7-3610QM CPU running at 2.3 GHz with 24 GBytes of memory.

In [15], the Alternating direction method of multipliers (ADMM) method has been applied to solve the minimization of a nuclear norm with a quadratic regularization term. It is known that ADMM is a method suited to solve large-scale optimization problems in a distributed way [17]. The minimization problem (3.22) can be adapted to fit into the problem solved in [15], so that algorithm can be applied here.

The minimization problem formulated by [15] was:

$$\min_x ||\mathscr{A}(x) - \mathbf{A}_0||_* + \frac{1}{2}(x-a)^T H(x-a) \tag{3.29}$$

where $x \in \mathbb{R}^k$ is the variable vector, the $\mathscr{A}(x)$ is a linear mapping $\mathscr{A} : \mathbb{R}^k \to \mathbb{R}^{p\times q}$, $A_0 \in \mathbb{R}^{p\times q}$ is a constant matrix and H is a symmetric matrix with k rows ans columns. The vector a is a constant vector with the same size as x.

To apply ADMM to (3.22), the first step is to define the variable vector x.

$$x = [\mathbf{t}_s^I \quad \mathbf{t}_s^v \quad \underline{y}_I \quad \underline{y}_v] \tag{3.30}$$

where: $\mathbf{t}_s^I$ are the Markov parameters $(D_I, C_I B_I, C_I A_I B_I, \cdots, C_I A_I^{s-2} B_I)$ representing the first column of $\mathscr{T}_s^I$, $\mathbf{t}_s^v$ are the Markov parameters for the vertical system, except $D_v = 0$, as it is not a variable, but a constant. The vectors $\underline{y}_v$ and $\underline{y}_I$ are given by:

$$\underline{y}_v = [y_v(1,1) \quad \cdots \quad y_v(1,N_h) \quad y_v(2,1) \quad \cdots \quad y_v(2,N_h) \quad y_v(N_v,1) \quad \cdots \quad y_v(N_v,N_h)] \tag{3.31}$$

and

$$\underline{y}_I = [y_I(1,1) \quad \cdots \quad y_I(N_v,1) \quad y_I(1,2) \quad \cdots \quad y_I(N_v,2) \quad y_I(1,N_h) \quad \cdots \quad y_v(N_v,N_h)]. \tag{3.32}$$

The linear mapping $\mathscr{A}(x)$ produces the matrix:

$$A = \begin{bmatrix} \underline{\mathbf{Y}}_{s,N}^v - \mathscr{T}_s^v \underline{\mathbf{U}}_{s,N}^v & 0 \\ 0 & \underline{\mathbf{Y}}_{s,N}^I - \mathscr{T}_s^I \underline{\mathbf{U}}_{s,N}^I \end{bmatrix}. \tag{3.33}$$

The nuclear norm of the block-diagonal matrix A is equal to the sum of the nuclear norms of each block. This means that by computing the nuclear-norm of A, the sum $||\underline{\mathbf{Y}}_{s,N}^v - \mathscr{T}_s^v \underline{\mathbf{U}}_{s,N}^v||_* + ||\underline{\mathbf{Y}}_{s,N}^I - \mathscr{T}_s^I \underline{\mathbf{U}}_{s,N}^I||_*$ is computed.

By fitting the minimization problem (3.22) into the form given in Equation 3.29, the optimization problem can be solved with the algorithm described in [15].

A big performance improvement was achieved by using ADMM. For the same identification experiment, i.e., the same I/O data, for the same system, with Hankel matrices of the same size, the CVX implementation takes over one hour, while the ADMM implementation requires less than a minute.

Next to using ADMM, another approach to speed up the computations is to run the optimization problem for multiple λ values in parallel. This can be done, as the results of one optimization problem do not interfere with the results of the other.

3.8. RESULTS

In this section the performed tests, and the results are shown. To quantify the quality of the identification algorithm, different quality criteria have been chosen.

Next to the quality criteria, some different test cases have been formulated for the algorithm. Although there are no special features in the algorithm to handle noise, or non-zero boundary conditions in the Roesser model, it is interesting to see how the algorithm performs in non-ideal cases.

3.8.1. QUALITY CRITERIA

To determine if the algorithm has a good quality or not, some performance indicators are necessary. In the performed tests, the following indicators were used.

NORMALIZED ROOT MEAN SQUARED ERROR (NRMSE)

The first criteria is the difference between the actual system output and the output of the estimated system for the same input.

As already mentioned before, the RMSE is a common measurement for the difference between an estimated and the actual signal.

To compare different I/O datasets, the RMSE should be normalized. This allows for a fair comparison of different datasets where the amplitudes of one signal are greater than the amplitudes of the other.

To normalize the RMSE, the signal range is used:

$$\text{NRMSE}(y(i,j),\hat{y}(i,j)) = \frac{RMSE(y(i,j),\hat{y}(i,j))}{y_{max} - y_{min}} \tag{3.34}$$

where y_{max} and y_{min} are the highest and the lowest value found in y.

For the normalized root-mean-square error (NRMSE), a low value represents a better estimated system. The lowest possible value is 0.

VARIANCE ACCOUNTED FOR (VAF)

According to [11], another scaled version for the prediction error commonly used is the Variance Accounted For (VAF):

$$\text{VAF}(y(i,j),\hat{y}(i,j)) = \max\left(0, \left(1 - \frac{\frac{1}{N}\sum_{(i,j)=(1,1)}^{(N_h,N_v)} ||y(i,j) - \hat{y}(i,j)||_2^2}{\frac{1}{N}\sum_{(i,j)=(1,1)}^{(N_h,N_v)} ||y(i,j)||_2^2}\right) \cdot 100\%\right). \tag{3.35}$$

The maximum value of the VAF is 100%, in which case the prediction error is zero, and the estimated model is perfect. The lowest possible value is zero, in which case there is no relation between the actual signal y and the estimated one $\hat{y}$.

EIGENVALUES OF THE A MATRIX

The last check is to compare the eigenvalues of the system matrix A:

$$A = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}. \tag{3.36}$$

As described before, the matrices for the horizontal and vertical systems are estimated up to similarity transformation. The A matrices of similar systems have the same eigenvalues. So to estimate the quality of the system, the eigenvalues of the A matrix are plotted, next to the eigenvalues of the actual matrix A.

Note that if the vertical and horizontal systems are estimated correctly (up to similarity transformation), the matrix A is also estimated correctly up to a similarity transformation.

If the matrix A is constructed according to (3.7) and the similarity transformations are made explicit:

$$\begin{bmatrix} T_h^{-1}A_hT_h & T_h^{-1}B_hC_vT_v \\ T_v^{-1}B_vC_hT_h & T_v^{-1}A_vT_v + T_v^{-1}B_vD_hC_vT_v \end{bmatrix} = \begin{bmatrix} T_h^{-1} & 0 \\ 0 & T_v^{-1} \end{bmatrix} \begin{bmatrix} A_h & B_hC_v \\ B_vC_h & A_v + B_vD_hC_v \end{bmatrix} \begin{bmatrix} T_h & 0 \\ 0 & T_v \end{bmatrix}. \tag{3.37}$$

This shows that if the subsystems are estimated up to similarity transformation, the matrix A is also estimated up to a similarity transformation.

3.8.2. IDEAL CASE

The first test has been done with ideal conditions given by:

- Noiseless I/O datasets
- A system which matches the requirements for the decomposition
- A system where $D \neq 0$

The used system is given by:

$$A_1 = \begin{bmatrix} -0.0200 & -0.1663 & -0.8796 \\ 2.0138 & 0.0039 & -0.2513 \\ -0.5713 & 0.0273 & -0.2203 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0.5522 & 0.5570 \\ 0.2243 & 0.2262 \\ 0.5397 & 0.5444 \end{bmatrix}, \tag{3.38a}$$

$$A_3 = \begin{bmatrix} -0.1825 & 0.0553 & 0.2127 \\ -0.3592 & 0.1087 & 0.4185 \end{bmatrix}, \quad A_4 = \begin{bmatrix} -0.5258 & 0.9507 \\ -0.2542 & -0.3923 \end{bmatrix}, \tag{3.38b}$$

$$B_1 = \begin{bmatrix} -1.4186 \\ -0.5761 \\ -1.3864 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -0.0159 \\ -0.0314 \end{bmatrix}, \tag{3.38c}$$

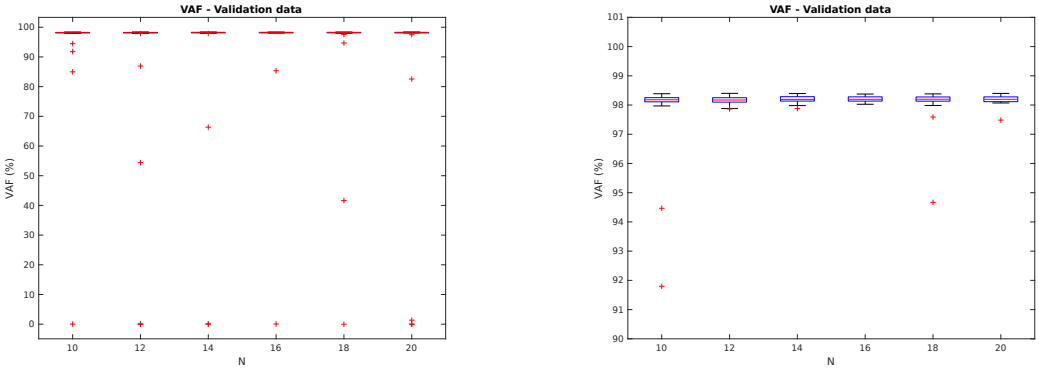
$$C_1 = \begin{bmatrix} 0.7144 & -0.2162 & -0.8322 \end{bmatrix}, \quad C_2 = \begin{bmatrix} -0.0243 & -0.0245 \end{bmatrix} \text{ and } D = \begin{bmatrix} 1 \end{bmatrix}. \tag{3.38d}$$

The identification has been done for 50 different input/output datasets randomly generated. All of the datasets have as boundary conditions $x_0^h = 0$ and $x_0^v = 0$.

The size of each Hankel matrix was $s = 12$ and N varying from 10 to 20. Remember that this means that the concatenated Hankel matrices have a total size of s rows and $(s+N-1)N$ columns. Varying N allows to study the effect and necessity of more data-points in the identification algorithm. The two different extraction methods are also compared to each other.

The VAF and NRMSE were computed with a validation dataset. The size of the validation datasets was $N_v = 40, N_h = 40$. And the input signal was a randomly generated Gaussian signal, with 0 mean and unit variance.

For the extraction algorithm in subsection 3.4.2, the plots can be found in Figure 3.5 and Figure 3.4. Although most identification runs seem to have a high VAF and a low NRMSE, there are still outliers with very low VAF, i.e. a VAF close to 0%. It is also possible to see that including more points does not necessarily mean a better system estimation. This is due to the fact that the estimation is already very close to 100%.

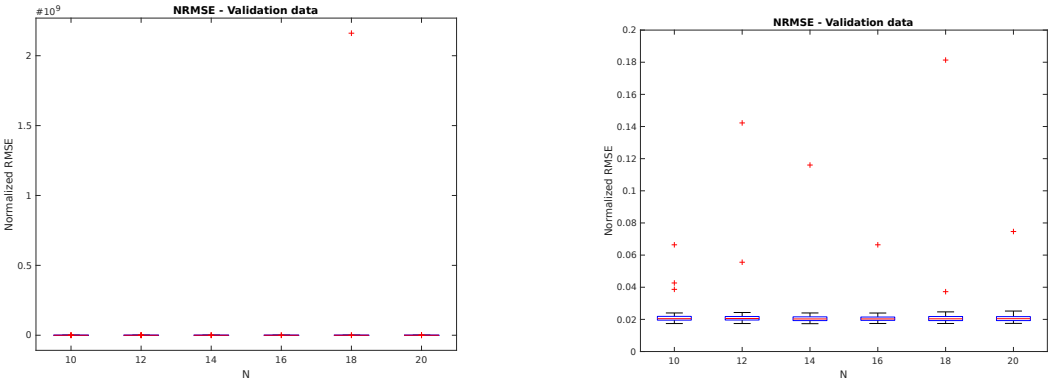


(a) Without zoom.

(b) With zoom.

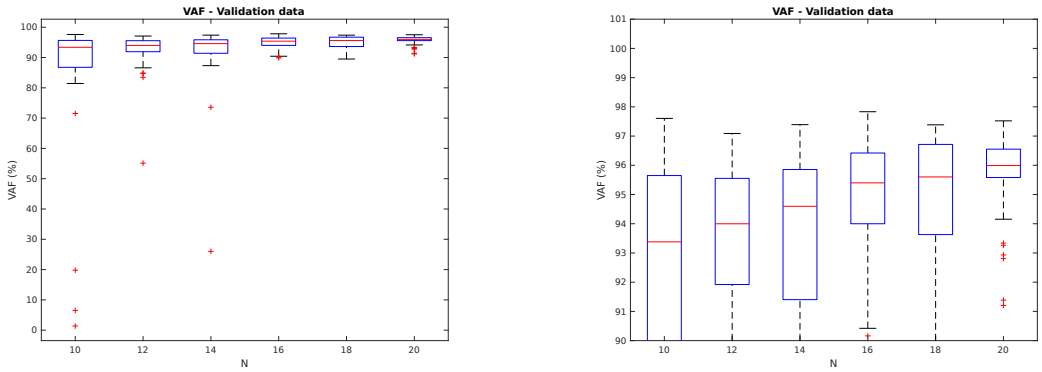
Figure 3.4: A box-plot for the VAF for 50 identification experiments with different values for N. The matrices are estimated with the algorithm in subsection 3.4.2.

For the extraction algorithm in subsection 3.4.1, the plots can be found in Figure 3.7 and Figure 3.6. Again the results have a high VAF and low NRMSE. In this matrix estimation algorithm N plays a more important role in the quality of the estimated system.



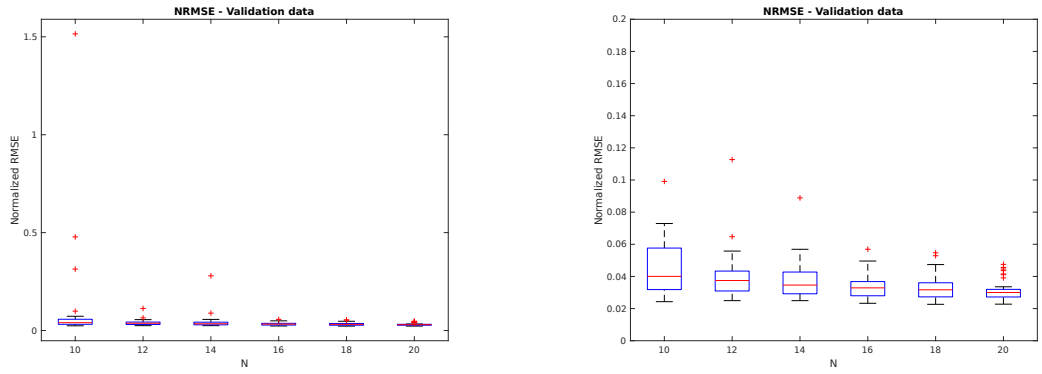
(a) Without zoom. (b) With zoom.

Figure 3.5: A box-plot for the NRMSE for 50 identification experiments with different values for N. The matrices are estimated with the algorithm in subsection 3.4.2.



(a) Without zoom. (b) With zoom.

Figure 3.6: A box-plot for the VAF for 50 identification experiments with different values for N. The matrices are estimated with the algorithm in subsection 3.4.2.



(a) Without zoom. (b) With zoom.

Figure 3.7: A box-plot for the NRMSE for 50 identification experiments with different values for N. The matrices are estimated with the algorithm in subsection 3.4.2.

The median of the VAF when estimating the system matrices out of the Toeplitz matrix is higher, but this approach also has more outliers around 0%. The same happens with the NRMSE, which in general is lower, but has some much bigger outliers.

And finally the eigenvalues of the estimated system matrix A can be seen in Figure 3.8.

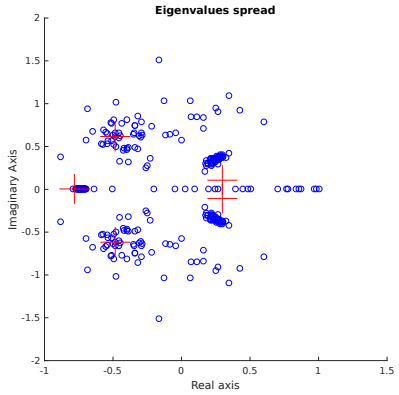


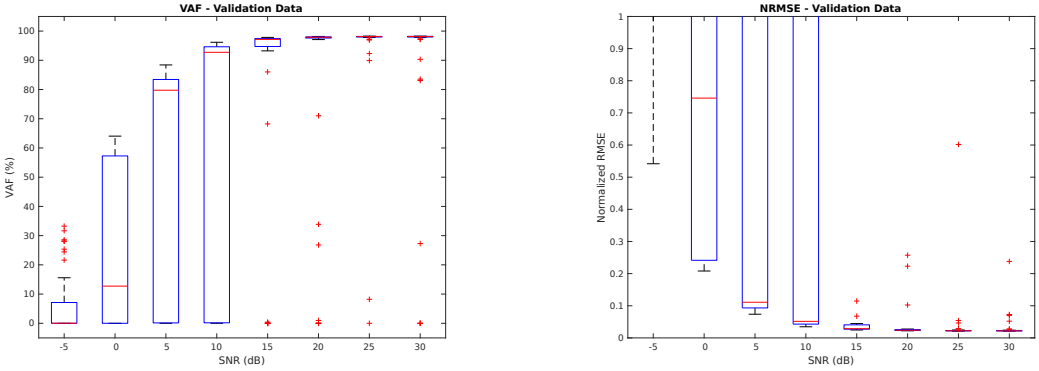
Figure 3.8: Eigenvalues of the system-matrix A, when it is extracted from the Toeplitz matrix.

3.8.3. NOISY SIGNALS

In a more realistic case, the input-output datasets are noisy. The algorithm has been tested with different noise levels with a Signal-to-Noise Ratio (SNR) varying from 30 to -5 dB. For each noise level 50 experiments were done, and the results are presented in this section. The system is the same as in the previous subsection.

In these experiments, the size of the Hankel matrices are given by $s = 12$ and $N = 10$. The system matrices are estimated from the Toeplitz matrices as described in subsection 3.4.2. The reason to chose this method is that it had better results in the noiseless case. The reason for choosing $N = 10$ is that the computations are faster with smaller values of N, while the quality of the estimated systems using the Toeplitz matrix was not improved by a higher value of N in the noiseless case.

The results can be seen in Figure 3.9.



(a) VAF (b) NRMSE

Figure 3.9: Box-plot with the VAF and the NRMSE of the estimated systems for different noise levels.

For the eigenvalues, the experiments with a SNR of 15 dB have been considered. Lower SNR already had a bad performance on the VAF, so no good estimation of the eigenvalues is expected.

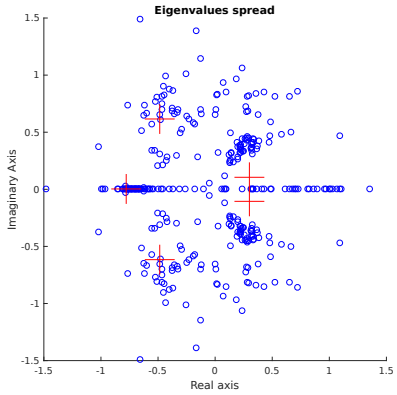


Figure 3.10: Eigenvalues distribution for the system matrix A, for 50 identification experiments with a SNR of 15 dB.

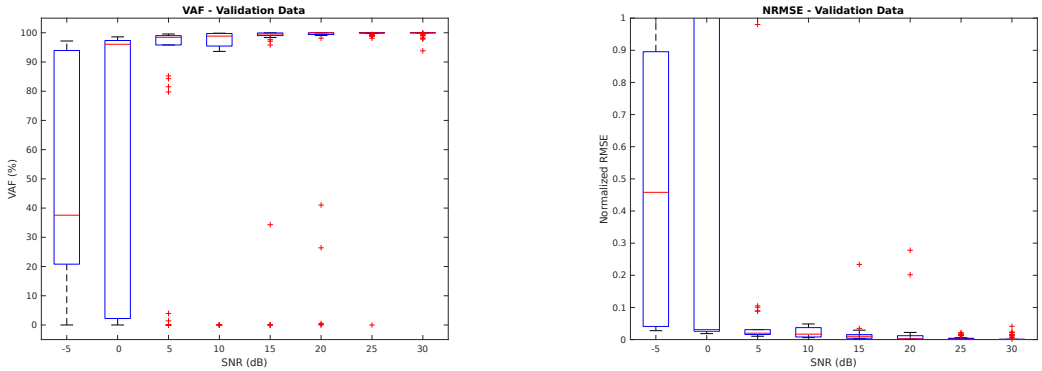
3.8.4. COMBINING WITH LFT

Subspace algorithms deliver an approximate system estimation. They can be combined with parametric algorithms for a better estimation of the system.

In this experiment, the algorithm has been used to provide the system size and an initial guess for the LFT algorithm described in [5]. The LFT is a parametric identification method. The system matrices are parametrized as a function of the vector Θ. The algorithm searches for the ideal Θ which minimizes the estimation error of the model. As mentioned [5], the algorithm is highly sensitive to the initial guess, as it might find a sub-optimal local minimum.

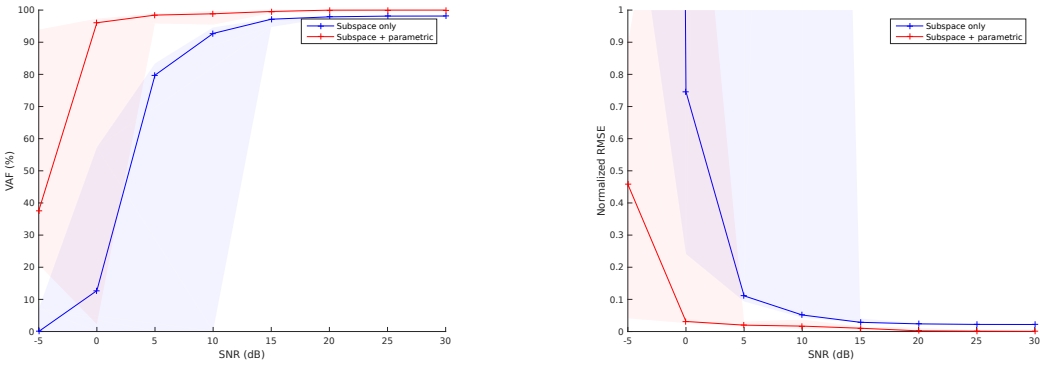
The experiments consisted of estimating the same system as previously mentioned, with noisy data, as described in the previous sub-section. The difference is that the LFT algorithm is used to provide a better approximation.

The results can be found in Figure 3.11 and in Figure 3.12 the results are compared with the results of subspace identification only.



(a) VAF (b) NRMSE

Figure 3.11: Box-plot with the VAF and the NRMSE of the estimated systems for different noise levels.



(a) VAF (b) NRMSE

Figure 3.12: Plots comparing the VAF and the NRMSE for the identification with the subspace algorithm only and for the identification combined with the parametric LFT algorithm. The lines are the median. The area around them represent the region between the 25% and 75% percentile, i.e., the region of the blue box in the box-plot.

3.8.5. ESTIMATING A SYSTEM THAT DOES NOT FIT IN THE REQUIRED CONDITIONS

The feedback model does not cover all types of Roesser models. But an interesting question to be answered is how well the identification algorithm would perform for a generic Roesser model.

For this case, we use a noiseless dataset for the Roesser model presented in [9], given by:

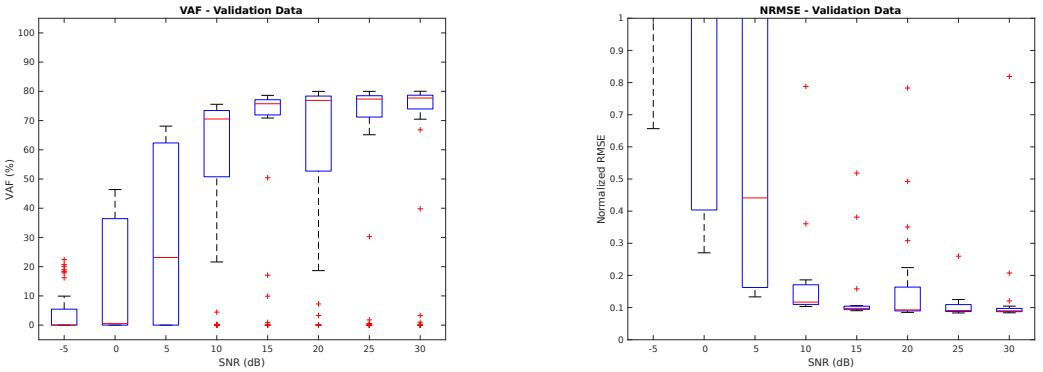
$$A_1 = \begin{bmatrix} -0.2589 & -0.4997 & -0.2581 \\ -0.4526 & 0.2943 & -0.0616 \\ -0.0736 & 0.0822 & -0.0858 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0.3967 & 0.1232 \\ -0.0459 & 0.2855 \\ 0.3396 & -0.0614 \end{bmatrix}, \tag{3.39a}$$

$$A_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad A_4 = \begin{bmatrix} -0.1175 & -0.1809 \\ 0.1257 & 0.3264 \end{bmatrix}, \tag{3.39b}$$

$$B_1 = \begin{bmatrix} 0.1006 \\ 0.1273 \\ -0.2313 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -1.0636 \\ -0.1246 \end{bmatrix}, \tag{3.39c}$$

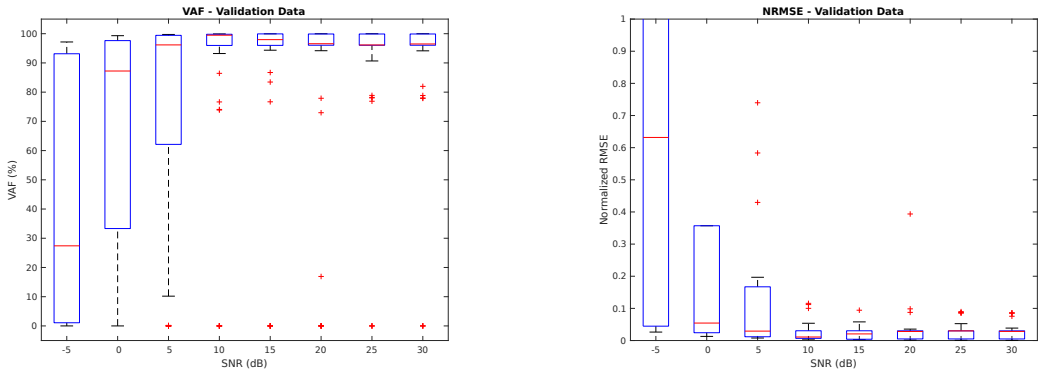
$$C_1 = \begin{bmatrix} 1.4578 & 1.7139 & 0.6824 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 0.3152 & 2.5705 \end{bmatrix} \text{ and } D = \begin{bmatrix} -0.2638 \end{bmatrix}. \tag{3.39d}$$

The results can be found in Figure 3.13. The combination with the LFT algorithm has also been explored, and the results can be found in Figure 3.14.



(a) VAF (b) NRMSE

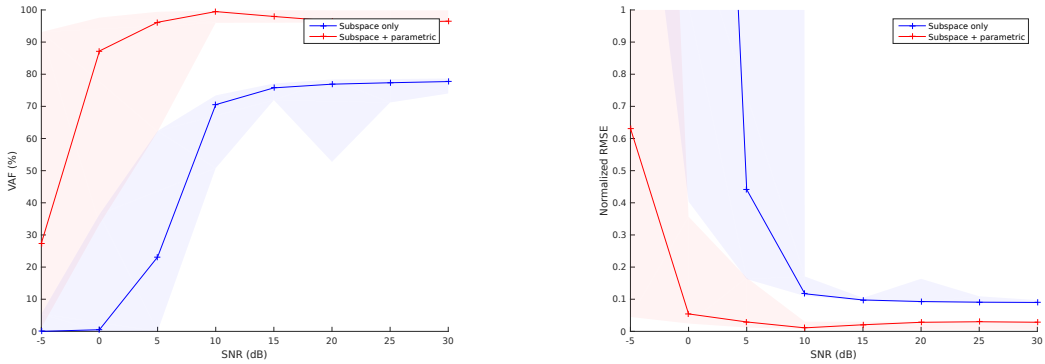
Figure 3.13: Box-plot with the quality criteria when the algorithm is used to estimate a CRSD model.



(a) VAF (b) NRMSE

Figure 3.14: Box-plot with the quality criteria when the algorithm is used to estimate a CRSD model, using the LFT algorithm to improve the quality of the estimated model.

For a better comparison, the results are plotted together in Figure 3.15.



(a) VAF (b) NRMSE

Figure 3.15: Plots comparing the VAF and the NRMSE for the identification of the CRSD model with the subspace algorithm only and for the identification combined with the parametric LFT algorithm. The lines are the median. The area around them represent the region between the 25% and 75% percentile, i.e., the region of the blue box in the box-plot.

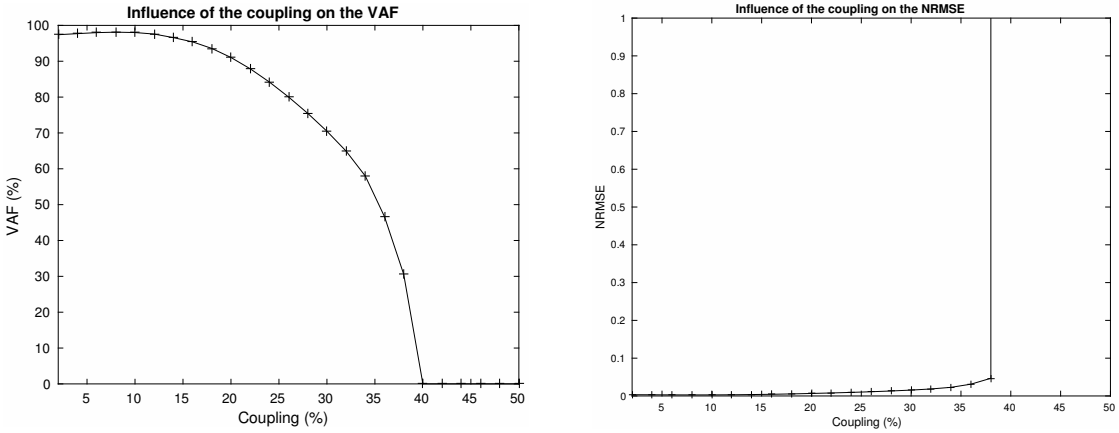
3.8.6. IDENTIFICATION OF MIRROR MODELS

The final goal of the algorithm is to identify mirrors used in Adaptive Optics, such that the Roesser model can be used in the control loop on Adaptive Optics (AO) systems.

For the experiment the shape of the mirror is given as the sum of the influence of all the actuators. By applying a voltage to one actuator, the mirror is deformed in a Gaussian like shape around that actuator. This Gaussian shape influences the height of the mirror at surrounding actuators. This influence is called coupling. A coupling of 5% means that the displacement of the mirror at neighbouring actuators of (i, j) is 5% of the displacement of the mirror at (i, j). For this experiment, the data was generated by using the Adaptive Optics Simulator YAO, found in [18]. The input of the model is the voltage applied to each actuator, and the output is the height of the mirror at the same points.

The identification was based on noiseless datasets, but with varying coupling between actuators.

For this case, it is not possible to plot the eigenvalues of the system matrix A, so only the VAF and the NRMSE were plotted. The graphs can be found in Figure 3.16.



(a) VAF (b) NRMSE

Figure 3.16: Quality of estimated mirrors for changing coupling between actuators.

3.9. DISCUSSION

3.9.1. ANALYSIS OF THE RESULTS

The algorithm was evaluated with three different indicators. The VAF and NRMSE are based on simulated data. By plotting the estimated eigenvalues, a check is performed on the estimated system itself, and not only on simulation data. In this section the results for each case are discussed.

NOISELESS CASE

For the noiseless case the algorithm presented good results with a VAF of more than 90% for most cases, and a NRMSE of less than 5%. But the algorithm was not able to estimate the system correctly with all the data-sets, generating statistical outliers in the quality data seen in Figure 3.4, Figure 3.5, Figure 3.6 and Figure 3.7. It is interesting to note that the median and the outliers depend on the method used to estimate the system matrices.

When estimating the system matrices from the Toeplitz matrix, the median of VAF for the estimated systems is higher, but the worse outliers have a VAF of 0%. For the other estimation method presented, the median for the estimated system was lower, but the number of outliers close to zero reduced when the value of N was increased.

The plots with the distribution of the eigenvalues seen in Figure 3.8 show that the eigenvalues of the estimated system from the Toeplitz matrix is biased.

NOISY CASE

For the noisy case only the estimation based on the Toeplitz matrix was used, due to the results on the noiseless cases. The VAF and the NRMSE showed a decrease in the quality of the estimated system. This is not surprising, as no technique has been used to compensate for noise. The plots in Figure 3.9 show the effect of the Signal to Noise Ration on the quality of the estimated systems. As expected, a lower SNR leads to worse estimates. A SNR higher than 15 dB is necessary for good estimates. The eigenvalues plot for this SNR level, seen in Figure 3.10 show the same bias as for the noiseless case.

COMBINATION WITH LFT

As an method to handle noise, a parametric identification method have been used. As expected, the results showed an significant improvement for cases with a low SNR. Note that there does not seem to be an significant reduction of the number of outliers. This is caused due to the fact that the approximation estimated by the sub-space algorithm is already really bad, and the parametric optimization is not able to correct that. The eigenvalues also are closer to the actual eigenvalues.

A CRSD MODEL

The most interesting result is that combined with LFT, the algorithm presented can also approximate Roesser models that do not fit into the required conditions for decomposition. The approximated model have a high

VAF and low NRMSE, and the eigenvalues are close to the actual ones. Again the quality of the estimated system depends on the dataset used for the estimation.

MIRROR IDENTIFICATION

For the identification of the spatial dynamics of deformable mirrors the algorithm provides a satisfactory result for low coupling between actuators. Increasing the coupling reduces the quality of the estimated model. Above 35% the algorithm is not able to estimate an approximate model. This can be due to the fact that even estimation of D cannot be done, due to the great influence of neighbouring actuators not considered in the model. The decreasing quality is expected as the mirror dynamics do not fit into the quarter-plane causality of the Roesser models being estimated. For a low coupling, estimating D, the direct influence of an actuator on the output at the same point, is enough to have a good estimation of the model. When the coupling is high, the role of neighbour actuators is higher, reducing the quality of the estimated model. To fully describe a mirror a full-plane Roesser model has to be estimated. See Figure 3.17.

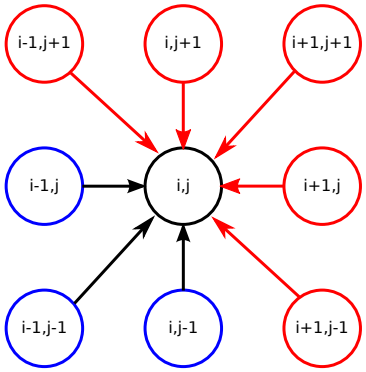


Figure 3.17: The point (i, j) of the mirror is influenced by the state of all its neighbours. The Quarter-Plane causal mirror however only models the influence of the three blue neighbours.

3.9.2. FUTURE WORK

The algorithm presented here can be improved in different ways. In this subsection, suggestions for future research work for this algorithm is given.

There are different points to improve the algorithm. One is to handle process and measurement noise. This reduces the necessity to use parametric identification algorithms, and improves the results of the estimated systems when the SNR is low. This can be done by using instrumental variables as done in PI-MOESP and PO-MOESP [11]. Another approach is to consider the bilinear problem. That allows an approach similar to N2SID, where the error between the model output and the actual output is included in the minimization problem. A possible formulation of the minimization problem then is given by:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N}, \hat{\underline{\mathbf{Y}}}^h_{s,N}, \underline{\mathbf{U}}^h_{s,N}, \mathcal{T}_s^v, \mathcal{T}_s^h} \quad & \left|\left|\underline{\mathbf{Y}}^v_{s,N} - \mathcal{T}_s^v \underline{\mathbf{U}}^v_{s,N}\right|\right|_* + \left|\left|\hat{\underline{\mathbf{Y}}}^h_{s,N} - \mathcal{T}_s^h \underline{\mathbf{U}}^h_{s,N}\right|\right|_* + \lambda ||\hat{y} - y||_2^2 \\ \text{subject to} \quad & y_v(i,j) + u(i,j) = u_h(i,j) \ . \end{aligned} \tag{3.40}$$

The algorithm derived in this chapter is limited to the SISO cases were $D \neq 0$. One area to improve the system is to handle MIMO systems. Extending the current algorithm is one option, but ideally the bilinear problem should be solved, instead of avoided. By solving the bilinear problem, there are no restrictions on D to be invertible. This facilitates the identification for MIMO cases.

And finally, using the algorithm for Adaptive Optics require some extra work to identify Full Plane 2D systems. As mentioned before, the mirror is a full-plane 2D system, while the identified system is only a Quarter-Plane model. To have a useful identification algorithm for adaptive optics, the Full-Plane model is required. One option is to consider non-causal 1D models for the vertical and horizontal case as mentioned in [7].

4

SYSTEM IDENTIFICATION FOR SEPARABLE IN DENOMINATOR MODELS

In this chapter a different subset of the class of Roesser models is considered. As seen in Section 2.4, separable in denominator models can be decomposed into two 1D systems interconnected in series. In this chapter an identification algorithm for this class of Roesser models is derived.

This algorithm has not yet been fully studied, so the chapter is structured explaining the problems encountered at each step done so far, and how the problem has been solved, or avoided.

The chapter starts with the formulation of the identification problem in Section 4.1. The data-equation and minimization problem formulation is given together in Section 4.2. The first problem encountered was the non-desired trivial solution of the optimization problem as seen in Section 4.3. As the algorithm requires a bi-linear minimization solver, one has been tested in Section 4.4. Problems with the optimization regarding the minimal decomposition are explained in Section 4.5. A regularization term was added to the minimization problem as explained in Section 4.6. A summary of the algorithm is given in Section 4.7, together with some implementation details. This algorithm has not yet been tested as the algorithm from Chapter 3. For this reason there are only a few results. The discussion about implementation and partial results when developing the algorithm are described in Section 4.8.

4.1. FORMULATION OF THE IDENTIFICATION PROBLEM

THE CLASS OF ROESSER MODELS BEING CONSIDERED

The decomposed Roesser model is shown in Figure 4.1, and the decoupled model is given by:

$$x_h(i+1,j) = A_h x_h(i,j) + B_h u_h(i,j), \tag{4.1a}$$
$$y(i,j) = C_h x_h(i,j) + D_h u_h(i,j), \tag{4.1b}$$

$$x_v(i,j+1) = A_v x_v(i,j) + B_v u(i,j), \tag{4.1c}$$
$$y_v(i,j) = C_v x_v(i,j) + D_v u(i,j), \tag{4.1d}$$

$$u_h(i,j) = y_v(i,j) = w(i,j). \tag{4.1e}$$

with $x^h \in \mathbb{R}^{n_h}$, $x^v \in \mathbb{R}^{n_v}$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^l$, $w \in \mathbb{R}^p$ and the matrices A_h, A_v, B_h, B_v, C_h, C_v, D_h and D_v are real matrices of appropriate dimensions.

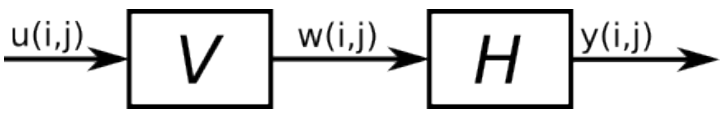


Figure 4.1: Decomposition of a Roesser model into two 1D systems. The system equations are given in (3.1).

THE IDENTIFICATION PROBLEM

The identification problem can now be formulated.

Consider an input-output dataset:

$$\left\{u(i,j), y(i,j)\right\}_{(i,j)=(1,1)}^{(N_h,N_v)}$$

with $N_h > n_h$ and $N_v > n_v$. Further it is known that $u(i,j) \in \mathbb{R}$ and $y(i,j) \in \mathbb{R}$.

Assuming the data-set was retrieved from an identification experiment with a SISO Roesser model decomposed as shown in Figure 4.1, the identification problem is to, given an intermediate signal size p, estimate the system orders n_v, n_h and the system matrices ($\hat{A}_{Th}$, $\hat{A}_{Tv}$, $\hat{B}_{Th}$, $\hat{B}_{Tv}$, $\hat{C}_{Th}$, $\hat{C}_{Tv}$, $\hat{D}_{Th}$ and $\hat{D}_{Tv}$) that define the decomposed model:

$$\hat{x}_{Th}(i+1,j) = \hat{A}_{Th}\hat{x}_{Th}(i,j) + \hat{B}_{Th}\hat{u}_h(i,j), \tag{4.2a}$$

$$\hat{y}(i,j) = \hat{C}_{Th}\hat{x}_{Th}(i,j) + \hat{D}_{Th}\hat{u}_h(i,j), \tag{4.2b}$$

and

$$\hat{x}_{Tv}(i,j+1) = \hat{A}_{Tv}\hat{x}_{Tv}(i,j) + \hat{B}_{Tv}u(i,j), \tag{4.2c}$$

$$\hat{y}_v(i,j) = \hat{C}_{Tv}\hat{x}_{Tv}(i,j) + \hat{D}_{Tv}u(i,j) \tag{4.2d}$$

such that

$$\hat{u}_h(i,j) = \hat{y}_v(i,j) = \hat{w}(i,j). \tag{4.3}$$

Remark. In this approach it is also necessary to estimate the signal w although this is not a primary goal of the algorithm.

Remark. Ideally the identification algorithm should also find the minimal value of p. But in the development so far, the size of the intermediate signal is a known parameter.

The estimated Roesser model is then given by:

$$\begin{aligned}\begin{bmatrix}\hat{x}_{Th}(i+1,j)\\ \hat{x}_{Tv}(i,j+1)\end{bmatrix} &= \begin{bmatrix}\hat{A}_{Th} & \hat{B}_{Th}\hat{C}_{Tv}\\ 0 & \hat{A}_{Tv}\end{bmatrix}\begin{bmatrix}\hat{x}_{Th}(i,j)\\ \hat{x}_{Tv}(i,j)\end{bmatrix} + \begin{bmatrix}\hat{B}_{Th}\hat{D}_{Tv}\\ \hat{B}_{Tv}\end{bmatrix}u(i,j)\\ y(i,j) &= \begin{bmatrix}\hat{C}_{Th} & \hat{D}_{Th}\hat{C}_{Tv}\end{bmatrix}\begin{bmatrix}\hat{x}_{Th}(i,j)\\ \hat{x}_{Tv}(i,j)\end{bmatrix} + \begin{bmatrix}\hat{D}_{Th}\hat{D}_{Tv}\end{bmatrix}u(i,j)\end{aligned} \tag{4.4}$$

4.2. DATA EQUATIONS FOR SUBSPACE IDENTIFICATION AND THE MINIMIZATION PROBLEM

The formulation of the data-equations in this algorithm is similar to the formulation in Section 3.2. One difference in the matrices is that the vertical system now can be a Single-Input Multiple-Output (SIMO) system and the horizontal a Multiple-Input Single-Output (MISO). That does change the sizes of the matrices, but the formulation is still the same as in the previous chapter.

Note that in order to formulate the data equations, a guess value for p is necessary, i.e., the size of the intermediate signal has to be given or guessed.

The minimization problem is again, similar to the minimization problem formulated in Section 3.3. Considering the nuclear norm as a relaxation of the rank operator, the minimization problem is now given by:

$$\begin{aligned}\min_{\underline{\mathbf{Y}}^v_{s,N},\underline{\mathbf{U}}^h_{s,N},\mathscr{T}_s^v,\mathscr{T}_s^h} \quad & \left|\left|\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}_s^v\underline{\mathbf{U}}^v_{s,N}\right|\right|_* + \left|\left|\underline{\mathbf{Y}}^h_{s,N} - \mathscr{T}_s^h\underline{\mathbf{U}}^h_{s,N}\right|\right|_*\\ \text{subject to} \quad & y_v(i,j) = u_h(i,j)\ .\end{aligned} \tag{4.5}$$

The minimization problem given is again a bilinear minimization problem, as $\mathscr{T}_s^h$ and $\underline{\mathbf{U}}^v_{s,N}$ appear as a multiplication.

The approach of inverting the horizontal system in Chapter 3 can be directly applied here when $p = 1$.

The chosen approach was to first only consider models where the minimal decomposition was 1, i.e., $p = 1$. By choosing $p = 1$ and inverting the horizontal system, the optimization problem can be written as:

$$\begin{aligned}\min_{\underline{\mathbf{Y}}^v_{s,N},\underline{\mathbf{Y}}^I_{s,N},\mathscr{T}_s^v,\mathscr{T}_s^I} \quad & \left|\left|\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}_s^v\underline{\mathbf{U}}^v_{s,N}\right|\right|_* + \left|\left|\underline{\mathbf{Y}}^I_{s,N} - \mathscr{T}_s^I\underline{\mathbf{U}}^I_{s,N}\right|\right|_*\\ \text{subject to} \quad & y_v(i,j) = y_I(i,j)\ .\end{aligned} \tag{4.6}$$

In this case, D_h of the system being estimated is assumed to be different from 0.

The estimation of the system matrices are similar as described in Section 3.4, and will not be described again here.

4.3. THE TRIVIAL SOLUTION TO THE MINIMIZATION PROBLEM

The first experiments to identify a CRSD model did not work, due to the fact that the optimization problem has an undesired trivial solution where the cost function is 0.

This comes due to the fact that nothing enforces the intermediate signal $y_v = w$ to be different from 0. And then all the Toeplitz matrices are filled with zeros. In that case, the optimal solution is zero.

To solve this problem, it has been considered that the value of $\hat{D}_{Th}$ can be enforced to a desired value without adding restrictions to the Roesser model being identified. In (4.4) the term $\hat{D}_{Th}$ only appears multiplying other terms, i.e., it appears in the multiplications $\hat{D}_{Th}\hat{C}_{Tv}$ and $\hat{D}_{Th}\hat{D}_{Tv}$.

To enforce an intermediate signal different from 0, the value of $\hat{D}_{Th}$ was enforced to 1, by adding a constraint to the minimization problem:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N},\underline{\mathbf{Y}}^I_{s,N},\mathscr{T}_s^v,\mathscr{T}_s^I} \quad & \left|\left|\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}_s^v \underline{\mathbf{U}}^v_{s,N}\right|\right|_* + \left|\left|\underline{\mathbf{Y}}^I_{s,N} - \mathscr{T}_s^I \underline{\mathbf{U}}^I_{s,N}\right|\right|_* \\ \text{subject to} \quad & y_v(i,j) = y_I(i,j), \\ & \hat{D}_{Th} = 1. \end{aligned} \tag{4.7}$$

Enforcing $\hat{D}_{Th} = 1$ can be done by enforcing it inside the Toeplitz matrix $\mathscr{T}_s^I$.

Applying this technique solved the problem with the trivial solution, and allowed systems with $p = 1$ to be identified with good accuracy. However, this restricts the number of systems that can be identified with this algorithm.

Consider the example given in [9]. The minimal decomposition of that system is $p = 3$. Estimating the system with the algorithm described so far in this chapter results in a VAF lower than 70% for noiseless datasets, using Hankel matrices with sizes $s = 12$ and $N = 20$. For this reason, solving the bilinear problem formulated in (4.5) is desired.

4.4. THE BILINEAR PROBLEM

Restricting the size of the intermediate signal add restrictions to the identified Roesser model. We are therefore interested in being able to estimate systems with a higher p.

By applying a bilinear solver, it is possible to keep the original data-equations, and no system inversion is necessary. This allows to estimate the vertical and horizontal system with $p \geq 1$. Using a bilinear solver it is also possible to handle noise in a similar approach as done in N2SID. The minimization problem for the noisy case would become:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N},\underline{\hat{\mathbf{Y}}}^h_{s,N},\underline{\mathbf{U}}^h_{s,N},\mathscr{T}_s^v,\mathscr{T}_s^h} \quad & \left|\left|\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}_s^v \underline{\mathbf{U}}^v_{s,N}\right|\right|_* + \left|\left|\underline{\hat{\mathbf{Y}}}^h_{s,N} - \mathscr{T}_s^h \underline{\mathbf{U}}^h_{s,N}\right|\right|_* + \lambda \left|\left|\hat{y} - y\right|\right|_2^2 \\ \text{subject to} \quad & y_v(i,j) = u_h(i,j) \ . \end{aligned} \tag{4.8}$$

A bilinear solver developed inside the research group [19] has been applied to the minimization problem for the noiseless case (4.5). The use of the algorithm in this optimization problem is described, but the theoretical explanation of the algorithm not. The reason is that the focus of the research was on the identification algorithm, and a good formulation of the minimization problem rather than the optimization problem. The solver allows to keep the data-equation structure, which is very important for estimating the system matrices. The next paragraphs describe the implementation of the algorithm.

First, the optimization problem (4.5) is reformulated:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N},\underline{\mathbf{U}}^h_{s,N},\mathscr{T}_s^v,\mathscr{T}_s^h} \quad & \left|\left|\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}_s^v \underline{\mathbf{U}}^v_{s,N}\right|\right|_* + \left|\left|\underline{\mathbf{Y}}^h_{s,N} - C\right|\right|_* \\ \text{subject to} \quad & C = \mathscr{T}_s^h \underline{\mathbf{U}}^h_{s,N} \\ & y_v(i,j) = u_h(i,j). \end{aligned} \tag{4.9}$$

Then the matrices $\mathscr{X}$ and $\mathscr{Y}$ are initialized with random values. The size of $\mathscr{X}$ is equal to the size of $\mathscr{T}_s^h$ and the size of $\mathscr{Y}$ is equal to the size of $\underline{\mathbf{U}}^h_{s,N}$.

Then the matrix M is defined as a function of $\mathscr{X}$ and $\mathscr{Y}$:

$$M(\mathscr{X},\mathscr{Y}) = \begin{bmatrix} C + \mathscr{X}\mathscr{Y} + \mathscr{X}\underline{\mathbf{U}}^h_{s,N} + \mathscr{T}_s^h\mathscr{Y} & \mathscr{T}_s^h + \mathscr{X} \\ \underline{\mathbf{U}}^h_{s,N} + \mathscr{Y} & I_n \end{bmatrix} \tag{4.10}$$

where I_n is the identity matrix of appropriate dimensions.

Now the following minimization problem is formulated:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N},\underline{\mathbf{U}}^h_{s,N},\mathscr{T}_s^v,\mathscr{T}_s^h,C} \quad & \left|\left|\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}_s^v\underline{\mathbf{U}}^v_{s,N}\right|\right|_* + \left|\left|\underline{\mathbf{Y}}^h_{s,N} - C\right|\right|_* + \lambda||M||_* \\ \text{subject to} \quad & C = \mathscr{T}_s^h\underline{\mathbf{U}}^h_{s,N} \\ & y_v(i,j) = u_h(i,j). \end{aligned} \tag{4.11}$$

After solving the minimization problem, new values for $\mathscr{X}$ and $\mathscr{Y}$ are defined:

$$\mathscr{X} = -\mathscr{T}_s^h \tag{4.12}$$

and

$$\mathscr{Y} = -\underline{\mathbf{U}}^h_{s,N} \tag{4.13}$$

The rank of the matrix M is then checked. If the rank is equal to the rank of I_n, then the optimal result has been found. Due to numerical tolerance, the rank operator has been substituted by the nuclear-norm and a small difference between the nuclear norms is allowed.

If that is not the case, the value of λ can be increased, and the steps repeated starting at solving the minimization problem (4.11).

A summary is given in Algorithm 2.

Algorithm 2 Bi-linear solver algorithm from [19] applied to the identification problem.

initialization:
  Generate a random $\mathscr{X}$ with the size of $\mathscr{T}_s^h$.
  Generate a random $\mathscr{Y}$ with the size of $\underline{\mathbf{U}}^h_{s,N}$.
while $||M||_* > ||I_n||_*$ **do**
  ▷ Where M is computed as defined in (4.10)
  Solve (4.11)
  $\mathscr{X} = -\mathscr{T}_s^h$
  $\mathscr{Y} = -\underline{\mathbf{U}}^h_{s,N}$
  if $||M||_*$ did not change in this iteration **then**
    $\lambda = \lambda \times 2$
  end if
end while

4.5. ENFORCING A FULL RANK W

A problem found in the first implementation of this algorithm was that the minimization problem lead to a trivial solution again. Enforcing $\hat{D}_{Th}(1) = 1$ solved the trivial solution of $w(i,j) = 0$, but resulted in a solution where all the values in the signal $w(i,j)$ were the same, i.e., if $w(i,j) = [w_1(i,j), w_2(i,j), \cdots, w_p(i,j)]^T$, then $w_1(i,j) = w_2(i,j) = \cdots = w_p(i,j)$ for all the values of i and j.

To study why this happens, it is necessary to study the minimal decomposition criteria described in Section 2.4.

One of the conditions imposed for the minimal decomposition in Section 2.4 was:

$$\text{rank}\begin{bmatrix} C_v & D_v \end{bmatrix} = p \tag{4.14}$$

This rank condition ensures that all the information contained in w is unique, i.e., a signal sequence $\mathbf{w} = [w(i,1), w(i,2), \cdots, w(i,N_v)]$ is full row rank, i.e., rank($\mathbf{w}$) $= p$, if the input signal $\mathbf{u} = [u(i,1), u(i,2), \cdots, u(i,N_v)]$ is persistently exciting. Note that it is not necessary to define here a 2D persistence of excitement, as the requirement applies to single columns of the 2D data which are treated as a 1D system.

If for a $p > 1$ all the values of $w(i,j) = [w_1(i,j), w_2(i,j), \cdots w_p(i,j)]^T$ are the same for all i and j, then the rank condition rank($\mathbf{w}$) $= p$ is not met as rank($\mathbf{w}$) $= 1$.

Lemma 4.5.1. *The rank condition in* (4.14) *ensures that the sequence* $\mathbf{w} = [w(i,1), w(i,2), \cdots, w(i,N_v)]$ *is full rank if the input sequence* $\mathbf{u} = [u(i,1), u(i,2), \cdots, u(i,N_v)]$ *is persistently exciting.*

Proof. The sequence **w** is given by the relation:

$$\mathbf{w} = \begin{bmatrix} C_v & D_v \end{bmatrix} \begin{bmatrix} \mathbf{X}^v_{j,s,N} \\ U^v_{j,1,N_v} \end{bmatrix} \tag{4.15}$$

The rank of **w** is given by Sylvester's rank inequality:

$$\operatorname{rank}\left(\begin{bmatrix} C_v & D_v \end{bmatrix}\right) + \operatorname{rank}\left(\begin{bmatrix} \mathbf{X}^v_{j,s,N} \\ U^v_{j,1,N_v} \end{bmatrix}\right) - (n_v+1) \leq \operatorname{rank}(\mathbf{w}) \leq \min\left(\operatorname{rank}\left(\begin{bmatrix} C_v & D_v \end{bmatrix}\right), \operatorname{rank}\left(\begin{bmatrix} \mathbf{X}^v_{j,s,N} \\ U^v_{j,1,N_v} \end{bmatrix}\right)\right). \tag{4.16}$$

As proven in Lemma 10.4 from [11], the rank:

$$\operatorname{rank}\left(\begin{bmatrix} \mathbf{X}^v_{j,s,N} \\ U^v_{j,1,N_v} \end{bmatrix}\right) = n_v + 1, \tag{4.17}$$

if the input signal $\mathbf{u} = [u(i,1), u(i,2), \cdots, u(i,N_v)]$ is persistently exciting of order $n_v + 1$.

This proves that $\operatorname{rank}(\mathbf{w}) = p$, if $n_v + 1 \geq p$. If the matrix $[C_v \;\; D_v]$ is full row rank, then $n_v + 1 \geq p$ also holds. □

The solution where all the values of w are the same, even for a persistently exciting signal is an undesired solution, as the full rank condition for the sequence **w** is not met.

To ensure the full rank condition, more equality constraints are added to the minimization problem.

Lemma 4.5.2. *For an intermediate signal of size* $p \geq 2$*, the full rank condition on* **w** *is achieved if the matrix:*

$$\begin{bmatrix} \hat{C}_{Tv}\hat{A}^{p-2}_{Tv}\hat{B}_{Tv} & \cdots & \hat{C}_{Tv}\hat{A}_{Tv}\hat{B}_{Tv} & \hat{C}_{Tv}\hat{B}_{Tv} & \hat{D}_{Tv} \end{bmatrix}$$

is full row-rank.

Proof. The relation between the sequence **w** and a Hankel matrix with the inputs is given by:

$$\mathbf{w} \approx \begin{bmatrix} \hat{C}_{Tv}\hat{A}^{p-2}_{Tv}\hat{B}_{Tv} & \cdots & \hat{C}_{Tv}\hat{A}_{Tv}\hat{B}_{Tv} & \hat{C}_{Tv}\hat{B}_{Tv} & \hat{D}_{Tv} \end{bmatrix} U^v_{j,p,N_v} \tag{4.18}$$

and for conciseness is written as

$$\mathbf{w} \approx C U^v_{j,p,N_v} \tag{4.19}$$

With the Sylvester's rank inequality condition:

$$\operatorname{rank}(C) + \operatorname{rank}(U^v_{j,p,N_v}) - p \leq \operatorname{rank}(\mathbf{w}) \leq \min(\operatorname{rank}(C), \operatorname{rank}(U^v_{j,p,N_v})) \tag{4.20}$$

With a persistently exciting system, $\operatorname{rank}(U^v_{j,p,N_v}) = p$. So if the rank of C is also p, the rank of **w** is also p. □

Ensuring the rank condition defined in Theorem 4.5.2 can be achieved by adding constraints to the minimization problem:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N}, \underline{\mathbf{U}}^h_{s,N}, \mathcal{T}_s{}^v, \mathcal{T}_s{}^h} \quad & \left\| \underline{\mathbf{Y}}^v_{s,N} - \mathcal{T}_s{}^v \underline{\mathbf{U}}^v_{s,N} \right\|_* + \left\| \underline{\mathbf{Y}}^h_{s,N} - \mathcal{T}_s{}^h \underline{\mathbf{U}}^h_{s,N} \right\|_* \\ \text{subject to} \quad & y_v(i,j) = u_h(i,j), \\ & \begin{bmatrix} \hat{D}_{Tv} & \hat{C}_{Tv}\hat{B}_{Tv} & \hat{C}_{Tv}\hat{A}_{Tv}\hat{B}_{Tv} & \cdots & \hat{C}_{Tv}\hat{A}^{p-2}_{Tv}\hat{B}_{Tv} \end{bmatrix} = I_p \end{aligned} \tag{4.21}$$

The new constraint can be enforced in the Toeplitz matrix for the vertical system.

4.6. WEIGHTING THE NUCLEAR NORMS

One assumption made so far is that the nuclear norm of $\underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}_s{}^v \underline{\mathbf{U}}^v_{s,N}$ is approximately equal to the nuclear norm of $\underline{\mathbf{Y}}^h_{s,N} - \mathscr{T}_s{}^h \underline{\mathbf{U}}^h_{s,N}$, but this is not necessarily the case.

To consider a possible difference in the nuclear norms, an extra weighting factor γ has been added to the minimization problem:

$$\begin{aligned} \min_{\underline{\mathbf{Y}}^v_{s,N}, \underline{\mathbf{Y}}^h_{s,N}, \mathscr{T}_s{}^v, \mathscr{T}_s{}^h} \quad & \left\| \underline{\mathbf{Y}}^v_{s,N} - \mathscr{T}_s{}^v \underline{\mathbf{U}}^v_{s,N} \right\|_* + \gamma \left\| \underline{\mathbf{Y}}^h_{s,N} - \mathscr{T}_s{}^h \underline{\mathbf{U}}^h_{s,N} \right\|_* \\ \text{subject to} \quad & y_v(i,j) = y_I(i,j), \\ & \begin{bmatrix} \hat{D}_{Tv} & \hat{C}_{Tv}\hat{B}_{Tv} & \hat{C}_{Tv}\hat{A}_{Tv}\hat{B}_{Tv} & \cdots & \hat{C}_{Tv}\hat{A}_{Tv}^{p-2}\hat{B}_{Tv} \end{bmatrix} = I_p \end{aligned} \tag{4.22}$$

The draw-back is that the computation time is increased, as the optimization problem has to be solved for multiple values of γ, but based on the few experiments performed, the quality of the estimated system increased with a $\gamma \neq 1$.

4.7. SUMMARY AND IMPLEMENTATION OF THE ALGORITHM

The implemented algorithm is fully described in Algorithm 3.

Algorithm 3 Identification algorithm for the CRSD decomposition of the Roesser model.

initialization:
  Define identification parameters s, N, and a set of γ values $\Gamma = \text{logspace}(\log(\gamma_{min}), \log(\gamma_{max}), steps)$.
  Create the Hankel matrices for the data-equations with dataset 1.
for each $\gamma \in \Gamma$ **do**
  Solve (4.22) using Algorithm 2. ▷ Solve the optimization problem
  Compute vertical and horizontal system matrices
  Compute Roesser model, from the estimated H and V system matrices.
  Simulate the estimated model using dataset 2.
  Compute the RMSE of the output of the simulation and the output of dataset 2.
end for
Select the estimated system with the lowest RMSE.

The code is delivered on a CD along with the report.

4.8. DISCUSSION

The algorithm presented in this chapter has not been thoroughly tested as the algorithm in Chapter 3. The main reason is that the there was not much available time before the end of the research. The algorithm itself, implemented with the bilinear optimization algorithm requires a full day to run, making the necessary time for testing longer than the available time.

One big drawback of the algorithm is the necessary computational time. Considering this is due to the iterative approach of the bilinear solver, optimizing the code of that algorithm can bring significant improvements to the identification algorithm. Two approaches are possible. One is reducing the number of necessary iterations. The second is to speed-up the code for a single iteration.

Two experiments were performed with the algorithm. The sizes of the Hankel matrices were $s = 10$ and $N = 10$. The input data was a Gaussian random signal with zero mean. The eigenvalues of the estimated A matrix match the eigenvalues of the actual system, but the NRMSE still indicates an error of 8%, and the VAF is lower than 50%. The reason is that the estimated A_2 matrix is close to zero. That would mean the algorithm was able to identify the vertical and horizontal dynamics, but failed to give a good estimate of the coupling between those two. The source of this problem is not known, and more research about it is necessary.

If the algorithm works for the noiseless case, a next step is to extend the algorithm to handle noise. The suggested approach already has been introduced in Section 4.4. The draw back is that the algorithm already has a regularization term γ between the 2 nuclear norms, and a second regularization term would increase even more the time necessary to perform a single identification.

Another field for improvement is on the bilinear solver. As mentioned before, the algorithm is really slow, taking approximately 20 hours for one identification run for small Hankel matrices. One possible approach is to adapt the ADMM algorithm to solve the optimization problem in (4.11).

Considering the existence of other subspace identification algorithms for 2D CRSD models such as the one presented in [9], it would be interesting to compare the results of both algorithms. It would also be interesting to know the effects of noise, and non-zero initial state on both algorithms.

The algorithm presented in the previous chapter had its results improved by combining it with a parametric identification algorithm. The same approach for this algorithm should be studied.

And finally, the use of this algorithm for mirror identification has to be studied. The problem of causality mentioned in the previous chapter should also be solved in here. In [20], the description of Full Plane separable in denominator models has been given.

5

CONCLUSION AND FUTURE WORK

In this final chapter, the conclusions about the proposed algorithm are written. It starts with a short description of the problem aimed to solve in Section 5.1, then discusses the approaches tried out in Section 5.2 and the results in Section 5.3. Finally future work is proposed in Section 5.4.

5.1. THE PROBLEM

The computational requirements in large-scale adaptive optics are not easily met. One of the problems is the memory bandwidth required due to large dense matrices used in the computation of the control signal for the deformable mirror.

One possible approach to reduce the requirements is to apply 2D state-space models, such as the Roesser model [4]. One of the first steps for the application of Roesser models to large-scale adaptive optics is to have an identification algorithm, that based on measurements is able to estimate a 2D model for the mirror.

The goal of the thesis was to develop an identification algorithm for Roesser models which can be applied to the identification of deformable mirrors.

Two available algorithms were studied. One is a subspace algorithm restricted to Causal Recursive Separable in Denominator models. This algorithm has as drawback that it does not handle the non-causal aspect of the mirror.

The second algorithm is a parametric identification based on the Linear Fractional Representation. The algorithm requires knowledge about the system size, and its results highly depend on the initial guess, as the algorithm relies on a non-convex optimization.

Another algorithm was considered necessary, to allow the identification of the non-causal aspect of the mirror.

5.2. SOLUTION APPROACH

The approach considered in this thesis was to consider the class of Roesser models which can be decomposed into two 1D systems. In the literature, two decompositions were found. The first one explored in this research is where the two 1D systems are connected in a feedback loop. The second algorithm considered the two systems connected in series. The advantage of the series connection is that it leads to CRSD models, which is a widely studied subclass of Roesser models.

By decomposing the Roesser model into two 1D systems, the identification problem can be reformulated to identify two 1D systems. This allows to apply identification techniques for 1D systems.

Both approaches developed consider a minimization problem, to minimize the rank of the subspace of interest. In the decomposed model, the Alternating Direction Method of Multipliers (ADMM) algorithm has been used to speed up the computations. For the models connected in series, the CVX toolbox [16] has been used, together with the bilinear solver described in [19].

No noise handling techniques have been applied to the identification algorithms so far, and the non-causal aspect has not been considered either. Both aspects will be considered for future research.

5.3. RESULTS

FEEDBACK DECOMPOSITION

The first identification algorithm developed uses the feedback loop decomposition of the Roesser model. The algorithm has been widely tested under difference scenarios.

First in the noiseless case, and with a Roesser model that fits into the decomposition. The results are good, with a high Variance Accounted For (VAF) and a low Normalized Root Mean Squared Error (NRMSE). Some of the datasets produced a bad system estimation but the overall performance was good.

When adding measurement noise to the dataset, the quality of the results decreases. But for a Signal to Noise Ratio of 15 dB, the results are still good, with a high VAF and a low NRMSE. One approach tested to handle noise, was to use the parametric LFT algorithm. First the system is estimated with the subspace algorithm described in the thesis. That result was used as an initial guess for the LFT algorithm. A performance increase can be seen for the combined algorithm.

The identification of systems that do not fit into the decomposed feedback model was also tested. The subspace algorithm itself does not work well. The VAF remained under 80%, and the NRMSE close to 10%. But when the algorithm is combined with the parametric LFT algorithm, the quality of the estimated systems is good, with a VAF higher than 90% for most cases.

The conclusion from the results is that this subspace method works well under ideal conditions. Combined with LFT it also works to estimate systems that do not fit into the required conditions for the feedback decomposition.

SERIES DECOMPOSITION

The second algorithm has been less tested, and still requires more work. The discussion is based on two runs, with noiseless data. The eigenvalues of the estimated A matrix are close to the eigenvalues of the actual A matrix, but the VAF is lower than 50% and the NRMSE is approximately 8%.

Looking at the estimated systems, this is due to the fact that the algorithm is able to estimate the dynamics in the horizontal and vertical system correctly, but fails to estimate the coupling between the two systems.

Another drawback of the algorithm is the time necessary to perform the identification. The two runs performed took 20 hours each.

5.4. FUTURE WORK

Both algorithms can be improved with more research. In this section future work for both algorithms is presented, with some ideas on how to approach the problems. Other ideas related to the use of Roesser models in adaptive optics are also presented.

FEEDBACK DECOMPOSITION

The first drawback of the algorithm is that it requires a system where D is invertible. That is not the case for all 2D systems. In order to avoid this requirement is to handle the bilinear minimization problem which has been avoided in this algorithm.

Another point for improvement is to handle noise in the algorithm. This could be achieved by considering the bilinear minimization problem, and adding the estimation error to the cost function. This approach has been used in N2SID for 1D systems [12, 13].

And a very important aspect for the use of the identification algorithm on large-scale adaptive optics, specifically on the deformable mirror, is that the model should be non-causal. As proven in [7], the decomposition is also valid for singular, i.e., non-causal models. So the algorithm has to identify two 1D non-causal systems.

SERIES DECOMPOSITION

This algorithm requires more study, and tests. The first big improvement is to optimize the computation of the optimization problem. A suggested approach is to use ADMM to solve the convex minimization problem inside the bilinear solver.

The results of the algorithm need further study. In further tests for the algorithm, the relation between the size of the datasets and the quality of the estimated model can studied. Larger datasets could improve the quality of the estimation of the coupling between the two 1D systems, which is the problem of the algorithm now.

More results on this algorithm are also interesting so that it can be compared to existing subspace identification algorithms for CRSD models.

Similar to the feedback decomposition, this algorithm can also be extended to handle noise by adding an estimation error to the cost function, similar to N2SID [12, 13]. This has already been mentioned during the development.

No tests have been performed in the identification of deformable mirrors. This algorithm still requires modifications in order to consider the non-causality of such systems.

MODELLING THE MIRROR WITH A 3D ROESSER MODEL

One assumption made so far is that the relation between measurements and actuators can be modelled only with spatial dynamics. This is not the case. The relation also changes in time, due to the movements a telescope has to make to follow a star.

When the time dynamics also play a role, the mirror could be modelled with a 3D Roesser model. In the 3D model, two dimensions are for the spatial dynamics of the mirror, and one dimension is for the time dynamics.

To make this possible, more insight on the application of the 2D models is necessary first. As mentioned in Chapter 3 and Chapter 4, the first step is to have Full-plane models for the mirrors.

MODELLING THE WAVEFRONT ABERRATION WITH ND ROESSER MODELS

Not only the mirror, but the atmospheric disturbances to the wavefront could also be modelled with a Roesser model. In [10] subspace identification of 2D Roesser models for stochastic input/output has been presented. The algorithm was used to compress information of a picture into a 2D model with lower dimensions. The technique could be applied for wavefront aberrations.

Using 2D models to compress the information of wavefront aberration would not require the model to be Full-Plane. The wavefront aberration can be seen as a picture, and the Roesser model is only used to compress the image information as done in [10]. However a Full-Plane model can be of lower dimension than a Quarter-Plane model [20].

The wavefront can also be modelled as a 3D system, with two spatial and one temporal dimension. By doing that, the aberration in future time-stamps can be predicted and corrected. Currently the correction is applied based on previously measured wavefronts.

BIBLIOGRAPHY

[1] C. C. de Visser and M. Verhaegen, *Wavefront reconstruction in adaptive optics systems using nonlinear multivariate splines,* J. Opt. Soc. Am. A **30**, 82 (2013).

[2] E. Thiébaut and M. Tallon, *Fast minimum variance wavefront reconstruction for extremely large telescopes,* J. Opt. Soc. Am. A **27**, 1046 (2010).

[3] M. Rosensteiner, *Cumulative reconstructor: fast wavefront reconstruction algorithm for extremely large telescopes,* J. Opt. Soc. Am. A **28**, 2132 (2011).

[4] R. P. Roesser, *A discrete state-space model for linear image processing,* Automatic Control, IEEE Transactions on **20**, 1 (1975).

[5] M. Farah, G. Mercere, R. Ouvrard, T. Poinot, and J. Ramos, *Identification of 2D Roesser models by using linear fractional transformations,* in *Control Conference (ECC), 2014 European* (2014) pp. 382–387.

[6] S.-Y. Kung, B. C. Levy, M. Morf, and T. Kailath, *New results in 2-d systems theory, part ii: 2-d state-space models—realization and the notions of controllability, observability, and minimality,* Proceedings of the IEEE **65**, 945 (1977).

[7] T. Kaczorek, *Decomposition of the Roesser model and new conditions for local controllability and local observability,* International Journal of Control **49**, 65 (1989).

[8] T. Lin, M. Kawamata, and T. Higuchi, *Decomposition of 2-d separable-denominator systems: Existence, uniqueness, and applications,* Circuits and Systems, IEEE Transactions on **34**, 292 (1987).

[9] J. Ramos, A. Alenany, H. Shang, and P. Lopes dos Santos, *A subspace algorithm for identifying 2-D CRSD systems with deterministic inputs,* in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on* (2011) pp. 6457–6462.

[10] J. Ramos and P. Lopes dos Santos, *Subspace system identification of separable-in-denominator 2-D stochastic systems,* in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on* (2011) pp. 1491–1496.

[11] M. Verhaegen and V. Verdult, *Filtering and System Identification: A Least Squares Approach* (Cambridge University Press, 2012).

[12] M. Verhaegen and A. Hansson, *N2SID: nuclear norm subsystem identification,* CoRR **abs/1501.04495** (2015).

[13] M. Verhaegen and A. Hansson, *Nuclear norm subspace identification (N2SID) for short data batches,* CoRR **abs/1401.4273** (2014).

[14] M. Verhaegen, A. Varga, and G. Grubel, *Some experience with the moesp class of subspace model identification methods in identifying the bo105 helicopter,* Institute for Robotics and System Dynamics, D–82230 Weßling, Techn. Rep. TR R165-94 (1994).

[15] Z. Liu, A. Hansson, and L. Vandenberghe, *Nuclear norm system identification with missing inputs and outputs,* Systems and Control Letters **62**, 605 (2013).

[16] M. Grant and S. Boyd, *CVX: Matlab software for disciplined convex programming, version 2.1,* `http://cvxr.com/cvx` (2014).

[17] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers,* Foundations and Trends® in Machine Learning **3**, 1 (2011).

[18] F. Rigaut, *yao::manual,* `http://frigaut.github.io/yao/manual.html` (2009), accessed in Februari 2015.

[19] R. Doelman and M. Verhaegen, *Sequential convex relaxation for convex optimization with bilinear matrix equalities,* (2016), submitted to the European Control Conference (ECC), 2016.

[20] S. Lele and J. Mendel, *Modeling and recursive state estimation for two-dimensional noncausal filters with applications in image restoration,* Circuits and Systems, IEEE Transactions on **34**, 1507 (1987).

ACRONYMS

ADMM Alternating direction method of multipliers. 20

AO Adaptive Optics. 1

CRSD Causal, Recursive and Separable in Denominator. 9, 33

DM Deformable Mirror. 1

E-ELT European Extremely Large Telescope. 1, 2

GPU Graphical Processing Unit. 2

LFR Linear Fractional Representation. 12

MISO Multiple-Input Single-Output. 32

MVM Matrix Vector Multiplication. 2

N2SID Nuclear Norm Subspace IDentification. 16, 33

N4SID Numerical algorithms for Subspace State-Space System IDentification. 11

NRMSE normalized root-mean-square error. 21, 22

RMSE root-mean-square error. 19, 21

SHS Shack-Hartmann Sensor. 1, 3

SIMO Single-Input Multiple-Output. 32

SISO Single-Input Single-Output. 14, 32

SNR Signal-to-Noise Ratio. 24

SVD Singular Value Decomposition. 18

TMT Thirty Meter Telescope. 1

VAF Variance Accounted For. 21, 22, 33