MIMO SYSTEM IDENTIFICATION
FOR H_∞ ROBUST CONTROL

A FREQUENCY DOMAIN APPROACH
WITH MINIMUM ERROR BOUNDS

Ton van den Boom

MIMO SYSTEM IDENTIFICATION
FOR H_∞ ROBUST CONTROL

A FREQUENCY DOMAIN APPROACH
WITH MINIMUM ERROR BOUNDS

MIMO SYSTEM IDENTIFICATION FOR H_∞ ROBUST CONTROL

A FREQUENCY DOMAIN APPROACH
WITH MINIMUM ERROR BOUNDS

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van
de Rector Magnificus, prof. dr. J.H. van Lint,
voor een commissie aangewezen door het College
van Dekanen in het openbaar te verdedigen op
dinsdag 5 januari 1993 om 16.00 uur

door

Antonius Josephus Johannes van den Boom

geboren te Bergen op Zoom

Dit proefschrift is goedgekeurd
door de promotoren

prof. dr. ir. P. Eykhoff
en
prof. ir. O. H. Bosgra

Copromotor: dr. ir. A. A. H. Damen

Voorwoord

De laatste 'loden' wegen het zwaarst. Nog wat kleine dingetjes, maar dan is het proefschrift toch helemaal af. Ik wil prof. Pieter Eykhoff, prof. Okko Bosgra, dr. Ad Damen, prof. Ton Backx, prof. Malo Hautus en dr. Siep Weiland graag bedanken voor het doorlezen van de eerste versies van dit proefschrift. Het belangrijke commentaar, dat zij gaven, heeft op het uiteindelijke resultaat een zeer positieve uitwerking gehad.

Voor mij markeert dit proefschrift de overgang van de ene naar de andere technische universiteit, namelijk van die van Eindhoven naar die van Delft. De mensen die ik wil vermelden komen dan ook voornamelijk uit deze twee 'kampen'.

Allereerst het Eindhovense. Ruim vier jaar geleden bood prof. Eykhoff mij aan om na mijn afstudeerwerk bij de vakgroep Meten en Regelen ook mijn vervangende dienstplicht te vervullen aan de TUE. Dat was voor mij het begin van een onderzoek dat nu in dit proefschrift is uitgemond.

In die vier jaar heb ik met plezier in Eindhoven gewerkt en veel discussies gevoerd met Ad Damen, Ton Backx, Ad van den Boom, Dolf van Rede, Siep Weiland en Ruud van Vliet. Deze discussies waren vaak fel, meestal gezellig maar bovenal zeer vruchtbaar voor mijn werk. Voor de noodzakelijke ontspanning zorgden de ontmoetingen in de Engelse pub, het zaalvoetballen tussen de middag (soms op kop, soms onderaan in de competitie, maar altijd met volle inzet) en het zeilen op de Grevelingen.

Natuurlijk vernoem ik ook mijn collega-promovendi, Martin Klompstra, Marc Keulers, Heinz Falkus en Jozef Mazak. Verder was het met Jobert Ludlage en Yu-cai Zhu van IPCOS vaak goed discussiëren. Ook leverden Huub Joosten, Frank Fokkelman, Leon Ariaans, Wim Liebregts en Wim Beckers, als stagiairs en afstudeerders een bijdrage aan mijn onderzoek. En natuurlijk waren de secretariële ondersteuning van Barbara Cornelissen en Muriel Simon, en ook de hulp van Udo Bartzke met betrekking tot problemen met de computers onontbeerlijk. Al deze mensen in Eindhoven wil ik van harte bedanken.

In het Delftse kamp vertoef ik pas sinds kort. Wat niet wegneemt dat velen al behulpzaam zijn geweest bij het tot stand komen van dit proefschrift. Bij name wil ik noemen Norbert Hofmeester en Ardjan Krijgsman die mede zorgden dat dit proefschrift uiteindelijk uit de printer rolde. Prof. Paul van den Bosch en Robert Jan Gorter waren zo aardig om mij zo af en toe vrij te stellen van verplichtingen met betrekking tot het college, waardoor ik de tijd vond om ook de laatste zinnen van mijn proefschrift op papier te zetten.

v

I also like to thank prof. Keith Glover and Dr. Jan Maciejowski, who gave me the opportunity to stay for three months at the Control Group of the University of Cambridge and who provided a lot of inspiration for my research. Thanks are due to all the people of the Control group, who were very helpful and gave me a nice time in England.

Tenslotte noem ik natuurlijk mijn vriendin Truus, die mij de laatste tijd vaak wel thuis zag komen maar niet echt het idee had dat ik aanwezig was. Toch kon zij hiervoor begrip opbrengen en als dat nodig was gaf zij mij weer een stimulans om verder te gaan.

Breda, 17 oktober 1992.

Abstract

A frequency domain system identification procedure for MIMO-systems is presented that yields a model with a bounded unstructured model error.

The disturbances on the input and output of the process are assumed to be bounded in the frequency domain, which implies that there exist bounding functions for the absolute value of the discrete Fourier transform of the noise signals.

First the identification of a SISO-process is considered with an additive or multiplicative model error structure. We derive uncertainty regions for the process dynamics and find an optimal model by H_∞-fitting.

Subsequently the identification method is extended to MIMO systems and various model error structures are considered, like additive model error, input and output multiplicative model error, reverse type model errors and coprime factor model errors. All these model error structures fit into a basic scheme with coprime factors.

We show that for a fixed model the true model error can be expressed as a function of a known matrix $G(z)$ and an unknown diagonal matrix $Q_t(z)$. This matrix $G(z)$ is built up from known information, such as the model, the model error structure, the measured data and the noise bounding functions. The diagonal matrix $Q_t(z)$ stands for the true scaled noise, which is unknown, but is assumed to be inside the unit ball.

An upper bound for the largest singular value of the model error is derived. We make use of the theory of structured singular values (μ-analysis) and of the so-called Redheffer star-products. The upper bounds for the model error are minimized in some norm-sense (H_∞, H_2 or a combination).

The choice of a linear parametrization leads to a convex optimization problem and the algorithms are robustly convergent.

Finally a case study is presented, where a laboratory process, consisiting of some connected water vessels, with 2 inputs and 2 outputs is identified and model with a bound on the model error is derived.

Contents

PART A: PRELIMINARIES

1

Introduction

System identification is an approximate modelling technique, using measured data. The models that are derived applying these system identification techniques are used in a wide field of applications, such as control design (e.g. industrial processes), diagnosis (e.g. medical science) and prediction (e.g. weather forcast).

In this thesis we are especially interested in the first application, control design. To be more specific, we wish to find an approximate model that is suited for H_∞ robust control design.

Recently techniques were developed to derive controllers that ensure robust performance. This means that the true plant will be stabilized and specified signal levels will be achieved, in the presence of plant uncertainties (Doyle et al. [12], [13]). A mathemetically convenient way to formalize the problem of robust control is to describe the system by a nominal model with a model error that is bounded in H_∞-sense. Bounds for these model errors, structured or unstructured, can be found by modelling the true plant. If physical modelling is not accurate enough (or even impossible), we can apply system identification to find better models with sufficiently small error bounds.

Most identification methods are based on the minimization of measures of residuals, error-signals like input error, output error or equation error (Eykhoff [16], Ljung [39]). The models can be parametrized in various ways, like impulse response models, difference equation models or state space models. The parameters are optimized by means of some estimation method (e.g. least squares).
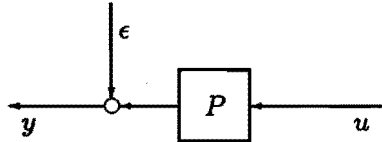
Figure 1.1: Conventional set up

The conventional set up of figure 1.1 has some drawbacks. Disturbances and noise signals are generally assumed to be Gaussian distributed and white, or coloured by some filter. Further they are assumed not to be correlated with the input signal. These assumptions are often violated. Furthermore, all disturbances are encaptured in an artificial signal ϵ acting on the output of the system. No distinction is made between different sources of disturbances, like sensor noise, actuator noise or process disturbances. Also the true process is often assumed to be an element of a model set. In practice this will usually not be the case, so that model errors and disturbance signals are minimized together. For that reason it is very hard to find bounds for the model errors.

Over the past years a lot of work has been done, in order to overcome the drawbacks of conventional identification, and to derive bounds for the model error. Of course there are many possible starting points, which all have their own advantages and drawbacks. Goodwin and Salgado [24], Gevers [20] and Zhu [68] have a stochastical setting, leading to so called 1-σ or 3-σ bounds on the model error. Others have a deterministic approach, either with bounded noise in the time domain (Wahlberg and Ljung [66], Helmicki et al. [30]), or with bounded noise in the frequency domain (LaMaire et al. [35], Gu and Khargonekar [26], Van den Boom et al. [60], [59], [56], [57], [58]).
It is our strong belief that any model identification method should come with explicit descriptions of the accuracy of the model. In this thesis we look for an identification procedure that not only gives a nominal model, but also gives a minimum upper bound for the model error.

In this chapter we give the problem statement for the work in this thesis. First, however, we will give some basic theoretical background for discrete

time systems, we discuss the notion of model error and we take a short look at robust control theory. The section 1.1 to 1.3 can be seen as a presentation of the definitions and notions that have to be known apriori before the problem statement can be given. Readers that are familiar with the concepts of sections 1.1 to 1.3 can proceed with section 1.4.

1.1 Linear time invariant systems

In this section we will consider some basic notions and definitions from system theory of discrete time systems. Frequency signals will be denoted as $x(z)$, time signals will be denoted with a bar, $\bar{x}(k)$, except for where the interpretation is clear from the context.

Consider the complex plane: Throughout this thesis

The unit circle will be denoted as C.
The closed unit disk will be denoted as $\mathcal{D}$.

Consider a time vector signal $\bar{x}(k)$, $k = -\infty, \ldots, \infty$,

The space $\mathbb{L}_2(-\infty, \infty)$: The space $\mathbb{L}_2(-\infty, \infty)$ is the space of all real vector valued signals $\bar{x}(k)$, $k = -\infty, \ldots, \infty$, for which

$$\sum_{k=-\infty}^{\infty} \bar{x}(k)^T \bar{x}(k) < \infty \tag{1.1}$$

In this thesis we will consider time vector signals $\bar{x}(k)$ to be an element of $\mathbb{L}_2(-\infty, \infty)$.
The subset of signals in $\mathbb{L}_2(-\infty, \infty)$ that are zero for all $k < 0$, is denoted as $\mathbb{L}_2[0, \infty)$. The subset of signals in $\mathbb{L}_2(-\infty, \infty)$ that are zero for all $k > 0$, is denoted as $\mathbb{L}_2(\infty, 0]$. The subset of signals in $\mathbb{L}_2(-\infty, \infty)$ that are zero for all $k < 0$ and all $k > M$, is denoted as $\mathbb{L}_2[0, M]$.

The z-transformation : Consider a time signal $\bar{x}(k) \in \mathbb{L}_2(-\infty, \infty)$. The z-transform $x(z)$ of the signal $\bar{x}(k)$ is defined as

$$x(z) = \sum_{k=-\infty}^{\infty} \bar{x}(k) z^{-k} \quad \text{for } z \in C \tag{1.2}$$

Consider the Markov parameters $\bar{p}(k)$, $k = 0, \ldots, \infty$ of a linear time invariant causal system.

The support of a Markov chain : The support of $\bar{p}(k)$ is the smallest interval $[0, l]$ such that $\bar{p}(k) = 0$, for all $k \notin [0, l]$.

The length of a Markov chain : Let $[0, l]$ be the support of a Markov chain. The length of the Markov chain is then the number $l - 1$.

Transfer function : For Markov parameters $\bar{p}(k)$ we can compute the z-transform

$$P(z) = \sum_{k=0}^{\infty} \bar{p}(k) z^{-k} \quad \text{for } z \in C \tag{1.3}$$

The function $P(z)$ is called the transfer function of the system.

Consider the following function spaces:

The space $R\!I\!P$:   The space $R\!I\!P$ is the space of all real rational proper matrices $P(z)$.

The space $R\!I\!L_{\infty}$: The space $R\!I\!L_{\infty}$ is the subspace of $R\!I\!P$, consiting of all elements $P(z)$ of $R\!I\!P$, that have no poles on the unit circle.

The space $R\!I\!H_{\infty}$: The space $R\!I\!H_{\infty}$ is the subspace of $R\!I\!L_{\infty}$, consiting of all elements $P(z)$ of $R\!I\!L_{\infty}$, that have all poles inside the unit disk.

The elements of $R\!I\!P$ describe all finite dimensional linear time invariant causal sampled systems. In this thesis we will consider all systems to be an element of the set $R\!I\!P$.

Let $\bar{u}(k) \in I\!L_2(-\infty, \infty)$ be the input vector signal and $\bar{y}(k) \in I\!L_2(-\infty, \infty)$ be the output vector signal of a system with its transfer function in $R\!I\!H_{\infty}$ with p inputs and q outputs. The input and output relation of this system can be given by a **difference equation:**

$$\bar{y}(k) + A_1\bar{y}(k-1) + \cdots + A_n\bar{y}(k-n) =$$

$$= B_0\bar{u}(k) + B_1\bar{u}(k-1) + \cdots + B_m\bar{u}(k-m)$$

where A_i are $p \times q$ real matrices, B_i are $p \times p$ real matrices and $m \leq n$. We introduce the **backward shift operator** τ by putting

$$\tau \, \bar{y}(k) = \bar{y}(k-1)$$

so that

$$(\, I \, + \, A_1 \tau \, + \, \cdots \, + \, A_n \tau^n \,) \, \bar{y}(k) \, =$$

$$= (\, B_0 \, + \, B_1 \tau \, + \, \cdots \, + \, B_m \tau^m \,) \, \bar{u}(k)$$

or

$$A(\tau) \, \bar{y}(k) \; = \; B(\tau) \, \bar{u}(k)$$

where $A(\tau) = I \, + \, A_1 \tau \, + \, \cdots \, + \, A_n \tau^n$
and $B(\tau) = B_0 \, + \, B_1 \tau \, + \, \cdots \, + \, B_m \tau^m$.

Let the z-transform of $\bar{u}(k)$ be given by $u(z)$ and let the z-transform of $\bar{y}(k)$ be given by $y(z)$, then

$$A(z) \, y(z) \; = \; B(z) \, u(z)$$

where $A(z) = (\, I \, + \, A_1 z^{-1} \, + \, \cdots \, + \, A_n z^{-n} \,)$
and $B(z) = (\, B_0 \, + \, B_1 z^{-1} \, + \, \cdots \, + \, B_m z^{-m} \,)$.

If $A(z)$ is invertible we can define: $P(z) = A^{-1}(z)B(z)$ and we get

$$y(z) \; = \; P(z) \, u(z)$$

If $A(\tau)$ is invertible we can define $P(\tau) = A^{-1}(\tau)B(\tau)$.
When we will use the symbolic notation

$$\bar{y}(k) \; = \; A(\tau)^{-1} B(\tau) \, \bar{u}(k) \; = \; P(\tau) \, \bar{u}(k)$$

we mean that $A(\tau) \, \bar{y}(k) \; = \; B(\tau) \, \bar{u}(k)$ and $y(z) \; = \; P(z) \, u(z)$.

Matrix norm : Let X be a complex valued matrix. The matrix norm of X is defined as the largest singular value of that matrix:

$$\|X\| \overset{def}{=} \sigma_{max}(X) \tag{1.4}$$

The infinity norm : The infinity norm of the matrix function $X(z) \in R\!L_\infty$ is defined as

$$\|X(z)\|_\infty \overset{def}{=} \max_{z \in C} \|X(z)\| \tag{1.5}$$

where $\|X(z)\|$ denotes the matrix norm.

The two norm: The two norm of the matrix function $X(z) \in R\!I\!\!L_\infty$ is defined as

$$\|X(z)\|_2 \overset{def}{=} \left(\frac{1}{2\pi} \int_{\omega=0}^{2\pi} \|X(e^{j\omega})\|^2 \, d\omega \right)^{\frac{1}{2}} \tag{1.6}$$

where $\|X(e^{j\omega})\|$ denotes the matrix norm.
Note that this two-norm differs from the H_2-norm that is often used in robust control theory (Doyle et al. [13]).

Norm error bounds for truncated system : Consider the Markov parameters $\bar{p}(k)$, $k = 0, \dots, \infty$ of a linear time invariant causal system with transfer function $P(z)$. Define a truncated system with Markov parameters

$$\bar{p}_T(k) \;=\; \left\{ \begin{array}{ll} \bar{p}(k), & \text{for } k = 0, \dots, l \\ 0 & \text{for } k \geq l+1 \end{array} \right.$$

and let $P_T(z)$ be the corresponding transfer function.
Further let the 'tail' of the Markov chain of $\bar{p}(k)$ be bounded with

$$\sum_{k=l+1}^{\infty} \| \bar{p}(k) \| \leq \epsilon_\Delta$$

Then we can give a bound for the matrix-norm of $\big(P(z) - P_T(z) \big)$ for all $z \in C$ as follows:

$$\| P(z) - P_T(z) \| \;=\; \| \sum_{k=0}^{\infty} \bar{p}(k)z^{-k} \;-\; \sum_{k=0}^{l} \bar{p}(k)z^{-k} \| \;=\;$$

$$\| \sum_{k=l+1}^{\infty} \bar{p}(k)z^{-k} \| \;\leq\; \sum_{k=l+1}^{\infty} \| \bar{p}(k) \| \;\leq\; \epsilon_\Delta$$

With this matrix-norm bound we can derive a bound for the infinity norm

$$\| P(z) - P_T(z) \|_\infty \;=\; \sup_{z \in C} \| P(z) - P_T(z) \| \;\leq\; \epsilon_\Delta$$

and a bound for two norm as well:

$$\|P(z) - P_T(z)\|_2 = \left(\frac{1}{2\pi} \int_{\omega=0}^{2\pi} \|P(e^{j\omega}) - P_T(e^{j\omega})\|^2 \, d\omega \right)^{\frac{1}{2}} \le$$

$$\le \left(\frac{1}{2\pi} \int_{\omega=0}^{2\pi} \|\epsilon_\Delta\|^2 \, d\omega \right)^{\frac{1}{2}} = \left(\frac{1}{2\pi} (\epsilon_\Delta^2) 2\pi \right)^{\frac{1}{2}} = \epsilon_\Delta$$

The discrete Fourier transform : Let $\bar{x}(k)$ be a time signal with k in the time interval $k = 0, \ldots, 2N - 1$. The discrete Fourier transform is then defined as

$$x(z) = \sum_{k=0}^{2N-1} \bar{x}(k) z^{-k} \quad \text{with} \quad z = e^{\frac{j2\pi n}{2N}}, \quad n = 0, \ldots, 2N - 1 \quad (1.7)$$

It is clear that, if the signal $\bar{x}(k) = 0$ for $k < 0$ and $k \ge 2N$, and if the domain of the definition of (1.7) is extended so as to include the unit circle, the discrete Fourier transform is equal to the z-transform.

Set of observation frequencies : We define the set of observation frequencies

$$\Omega = \{z_1, z_2, \ldots, z_N\} \quad \text{where} \quad z_n = e^{\frac{j\pi(n-1)}{N}}, \quad n = 1, \ldots, N$$

and so we consider samples in the frequency domain at equi-interval points on the unit circle.

The infinity norm for sampled frequency domain : Given Ω, the infinity norm of the matrix function $X(z)$ for the sampled frequency domain is defined as

$$\|X(z)\|_{s\infty} \overset{def}{=} \max_{z_i \in \Omega} \|X(z_i)\| \quad (1.8)$$

where $\|X(z_i)\|$ denotes the matrix norm.

The two norm for sampled frequency domain : Given Ω, the two norm of the matrix function $X(z)$ for the sampled frequency domain is defined as

$$\|X(z)\|_{s2} \overset{def}{=} \frac{1}{N} \left(\sum_{i=1}^{N} \|X(z_i)\|^2 \right)^{\frac{1}{2}} \quad (1.9)$$

where $\|X(z_i)\|$ denotes the matrix norm.

The ordinary infinity-norm and the infinity-norm for the sampled frequency domain are closely related. Define a weighting filter

$$W_\Omega(z) = \sum_{n=1}^{N} \delta(z - z_n) \quad \text{with } z_n \in \Omega \, , \, z \in C$$

where $\delta(z)$ denotes the Kronecker delta.
Then

$$
\begin{aligned}
\|W_\Omega(z)\,X(z)\|_\infty &= \max_{z \in C} \|W_\Omega(z)\,X(z)\| = \\
&= \max_{z_i \in \Omega} \|X(z_i)\| = \\
&= \|X(z)\|_{s\infty}
\end{aligned}
\tag{1.10}
$$

We see that if $X(z)$ is a smooth function and if N is large, then the ordinary infinity-norm and the infinity-norm for the sampled frequency domain are nearly equal. Let $X(z)$ be in $R\!L_\infty$. Then $\|X(z)\|$ will be a continuous function over the unit circle and we can give the following results:

$$\lim_{N \to \infty} \|X(z)\|_{s\infty} = \|X(z)\|_\infty$$

and

$$\lim_{N \to \infty} \|X(z)\|_{s2} = \|X(z)\|_2$$

We will now summarize some basic concepts of algebra and functional analysis that are used in literature.

Inner or all-pass functions: A matrix function $H(z)$ is called an inner or all-pass function if $H(z) \in R\!I\!H_\infty$ and $H^*(z)\,H(z) = I$ for all $z \in C$.

Coinner functions: A matrix function $H(z)$ is called a coinner function if $H(z) \in R\!I\!H_\infty$ and $H(z)\,H^*(z) = I$ for all $z \in C$.

Outer or minimum-phase functions: A matrix function $H(z)$ is called an outer or minimum-phase function if $H(z)$ is in $R\!I\!H_\infty$ and has full rank for all $|z| \geq 1$.

Unimodular functions: A square matrix function $H(z)$ is called a unimodular function in $R\!I\!H_\infty$ if $H^{-1}(z)$ exists and $H(z)$ and $H^{-1}(z)$ are both in $R\!I\!H_\infty$.

Right coprime factorization: The pair $N(z)$, $M(z)$ is called a right coprime factorization over $R\!I\!H_\infty$ of the system $P(z) \in R\!I\!P$ if

1. $P(z) = N(z) M^{-1}(z)$

2. $N(z) \in R\!H_\infty$ and $M(z) \in R\!H_\infty$

3. there exist two matrices $X(z) \in R\!H_\infty$ and $Y(z) \in R\!H_\infty$ such that $X(z) N(z) + Y(z) M(z) = I$.

Left coprime factorization: The pair $N(z)$, $M(z)$ is called a left coprime factorization over $R\!H_\infty$ of the system $P(z) \in R\!P$ if

1. $P(z) = M^{-1}(z) N(z)$

2. $N(z) \in R\!H_\infty$ and $M(z) \in R\!H_\infty$

3. there exist two matrices $X(z) \in R\!H_\infty$ and $Y(z) \in R\!H_\infty$ such that $N(z) X(z) + M(z) Y(z) = I$.

Normalized coprime factorization:

A right coprime factorization $P(z) = N(z) M^{-1}(z)$ is called normalized if $N^*(z)N(z) + M^*(z)M(z) = I$.
A left coprime factorization $P(z) = M^{-1}(z) N(z)$ is called normalized if $N(z)N^*(z) + M(z)M^*(z) = I$.
Here $*$ denotes complex conjugate.

Internal stability   (Francis [18]):

Consider the feedback system in figure 1.2 where $P(z) \in R\!P$ and $K(z) \in R\!P$. The feedback system is said to be internally stable if the four transfer matrices from v_1 and v_2 to u_1 and u_2 are all stable.
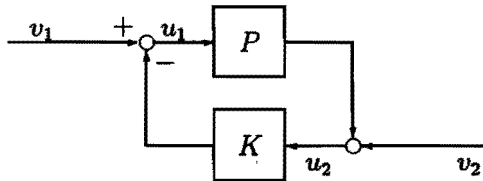
Figure 1.2: Feedback system

Stability criterion   (Vidyasagar [62], Vidyasagar et al. [64]):

Consider the Feedback system of figure 1.2. Let $P(z) = \tilde{M}^{-1}(z)\tilde{N}(z) = N(z) M^{-1}(z)$ be a left and a right coprime factorization of a process, and let $K(z) = \tilde{M}_C^{-1}(z)\tilde{N}_C(z) = N_C(z) M_C^{-1}(z)$ be a left and a right coprime factorization of a controller. Then the following statements are equivalent:

(1). The closed loop of $P(z)$ and $K(z)$ is internally stable.

(2). $\tilde{N}_C(z)N(z) + \tilde{M}_C(z)M(z)$ is unimodular.

(3). $\tilde{N}(z)N_C(z) + \tilde{M}(z)M_C(z)$ is unimodular.

1.2   Model error structures

When modelling an industrial process, either on the basis of physical laws, or by system identification, we will never be able to find an exact description of this process. There will always be a discrepancy between the true process $P_t(z)$ and its model $P(z)$. This model error $\Delta(z)$ can be thought of as an unknown transfer function that indicates the difference between the true process and the model. The model error may have various kinds of structures. In this section some model error structures will be considered. Throughout, P and Δ are supposed to have proper dimensions.

Additive model error (see figure 1.3):
The most common form to represent uncertainty in a model is the additive model error structure (Doyle & Stein [15], Cruz et al. [7], Freudenberg et al. [19]). Here the uncertainty $\Delta(z)$ is thought to be additive to the nominal model $P(z)$, so that the true process is described by

$$P_t(z) \;=\; P(z) \;+\; \Delta(z)$$

The corresponding configuration is given in figure 1.3
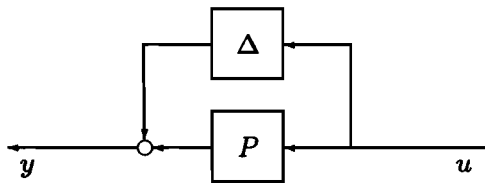
Figure 1.3: Additive model error structure

Another easy way to represent model errors is using a multiplicative model error representation (Doyle & Stein [15], Cruz et al. [7], Freudenberg et al. [19]). In the multivariable case we will have to distinguish between input and output multiplicative model errors.

Input multiplicative model error (see figure 1.4):
In this model error representation the uncertainty is thought to be multiplicative on the input of the nominal model, so

$$P_t(z) \;=\; P(z)\,(\,I \;+\; \Delta(z)\,)$$

The configuration will be as in figure 1.4.
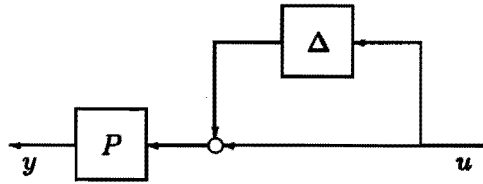
Figure 1.4: Input multiplicative model error structure

Output multiplicative model error (see figure 1.5):
In this model error representation the uncertainty is thought to be multiplicative on the output of the nominal model, so

$$P_t(z) = (I + \Delta(z)) P(z)$$

The configuration will be as in figure 1.5.
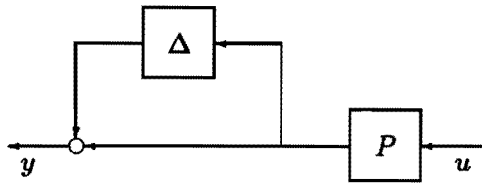
Figure 1.5: Output multiplicative model error structure

In control theory there has been a lot of effort in studying stability with additive and multiplicative model error structures. However, in the literature one generally assumes that the nominal model $P(z)$ and the true process $P_t(z)$ have the same number of unstable poles. If this condition can not be guaranteed, we can use other representations, which have an uncertainty model in a feedback loop (Postlethwaite and Foo [48],[49], Lunze [40]). For this kind of representation the unstable-pole-condition is replaced by the condition that the nominal model $P(z)$ and the true process $P_t(z)$ have the same number of finite zeros outside the unit disk. We discuss this type of model errors next.

Reverse additive model error (see figure 1.6):
We can represent the uncertainty of the model as a feedback over the nominal model, so

$$P_t(z) = P(z)(I + \Delta(z) P(z))^{-1}$$

as in the configuration of figure 1.6:
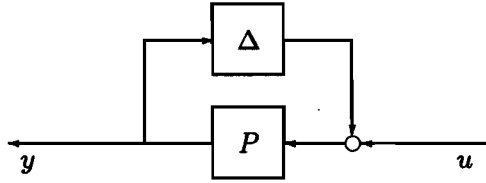
Figure 1.6: Reverse additive model error structure

Input reverse multiplicative model error (see figure 1.7):
In this model error representation the uncertainty is thought to be represented as a feedback, multiplicative on the input of the nominal model, so

$$P_t(z) = P(z)(I + \Delta(z))^{-1}$$

The configuration will be as in figure 1.7.
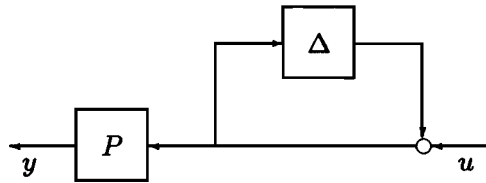
Figure 1.7: Input reverse multiplicative model error structure

Output reverse multiplicative model error (see figure 1.8):
In this model error representation the uncertainty is thought to represented as a feedback, multiplicative on the output of the nominal model, so

$$P_t(z) = (I + \Delta(z))^{-1} P(z)$$

The configuration will be as in figure 1.8.
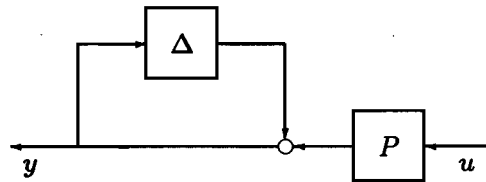
Figure 1.8: Output reverse multiplicative model error structure

While studying stability with the reverse kind of model errors, we need the additional assumption that the number of zeros, outside the unit disk of

the model and the perturbed plant are equal. If we do not like the condition concerning the number of zeros and poles outside the unit disk, or if we do not have enough information about it, we can make use of a different kind of model error structure, the coprime factor model error structure. This uncertainty representation starts from a coprime factorization of the plant as well as from the true process, and defines the model error as the difference between the coprime factors of the model and the coprime factors of the true plant (Vidyasagar & Kimura [63], McFarlane [43], Glover & McFarlane [22]). Of course, in the MIMO-case we can distinguish left and coprime factorizations:

Left coprime factor model error (see figure 1.9):
In this model error representation we choose a left coprime factorization of the plant $P(z) = M(z)^{-1}N(z)$ and the true process $P_t(z) = M_t(z)^{-1}N_t(z)$ and define the coprime factor model errors as the difference:

$$\Delta_N(z) = N_t(z) - N(z)$$
$$\Delta_M(z) = M_t(z) - M(z)$$

The matrices $\Delta_N(z)$ and $\Delta_M(z)$ will both be in $R\!H_\infty$ because $N(z), M(z)$, $N_t(z)$ and $M_t(z)$ all are in $R\!H_\infty$. This results in:

$$P_t(z) = (M(z) + \Delta_M(z))^{-1}(N(z) + \Delta_N(z))$$

We can represent this by the configuration as in figure 1.9
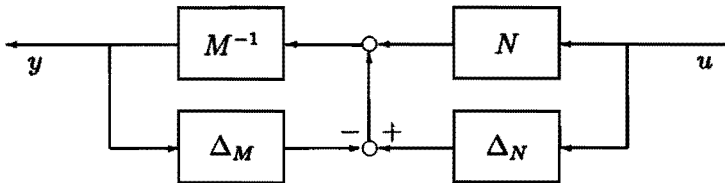
Figure 1.9: Left coprime factor model error

Right coprime factor model error (see figure 1.10):
In this model error representation we choose a right coprime factorization of the plant $P(z) = N(z)M(z)^{-1}$ and the true process $P_t(z) = N_t(z)M_t(z)^{-1}$ and define the coprime factor model errors as the difference:

$$\Delta_N(z) = N_t(z) - N(z)$$
$$\Delta_M(z) = M_t(z) - M(z)$$

The matrices $\Delta_N(z)$ and $\Delta_M(z)$ will both be in $R\!H_\infty$ because $N(z), M(z)$, $N_t(z)$ and $M_t(z)$ all are in $R\!H_\infty$. This results in:

$$P_t(z) = (\, N(z) \, + \, \Delta_N(z) \,)(\, M(z) \, + \, \Delta_M(z) \,)^{-1}$$

We can represent this by the configuration as in figure 1.10
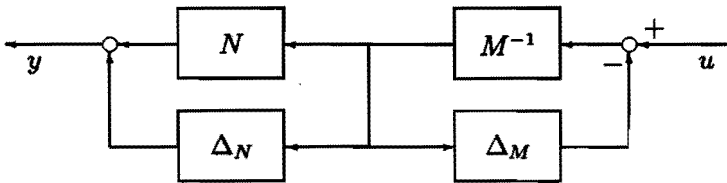
Figure 1.10: Right coprime factor model error

Of course many other configurations can be thought of, but the model error structures as mentioned above are the most important model error descriptions for unstructured model errors, used in the literature. A common assumption in literature is that $\Delta(z)$ should be in the $R\!L_\infty$ space. In this thesis we restrict ourselves to model errors $\Delta(z)$ that are in the $R\!H_\infty$ space, so they are stable. We will comment on that at the end of chapter 5.

1.3   Robust Control

In this section we will discuss robust control theory very briefly. Consider the configuration of figure 1.11, where a plant $P(z)$ is in closed loop with a controller $K(z)$. Plant $P(z)$ and controller $K(z)$ may be multivariable systems. $r(k)$ is the reference tracking signal, $u(k)$ is the plant input signal and $y(k)$ is the plant output signal. The plant input and output signals are perturbed by the disturbance signals $d(k)$ and $e(k)$ respectively.
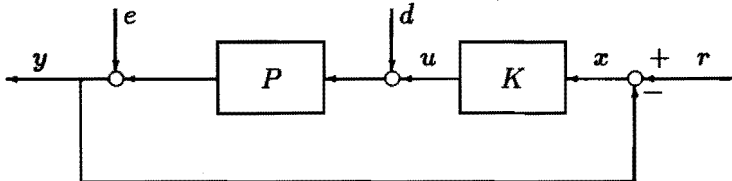
Figure 1.11: Plant and controller in closed loop

For this set up we define **the sensitivity function** as:

$$S(z) = (I + P(z) K(z))^{-1}$$

Performance of a closed loop system can be specified in different ways. One might be interested in optimal tracking or in minimum control power dissipation. The nominal performance of a closed loop system is the performance in the case of exact modelling, so $P_t(z) = P(z)$ in figure 1.1.11.

Another issue is that of the robust stability of the closed loop. Suppose, we have a set $\tilde{\mathbf{P}}$, with $P_t(z) \in \tilde{\mathbf{P}}$. Then the configuration of figure 1.11 is called **robustly stable** for the set $\tilde{\mathbf{P}}$ if it is stable for all $P(z) \in \tilde{\mathbf{P}}$. A controller $K(z)$ is called a **robustly stabilizing controller** for the set $\tilde{\mathbf{P}}$ if $K(z)$ stabilizes all $P(z) \in \tilde{\mathbf{P}}$.

Finally, if we want to achieve **robust performance**, we like to achieve a specified performance level in the presence of perturbations. We will now consider these three topics in more detail:

Nominal performance:
We consider the set up of figure 1.11 where $P_t(z) = P(z)$. If we are interested in **optimal signal tracking** we want the difference $(y - r)$ to be small with respect to the reference signal r. We know that

$$y(z) - r(z) = x(z) = (I - P(z) K(z) S(z)) r(z) = S(z) r(z)$$

Therefore, we are interested in getting the sensitivity function smaller than a pre-specified level $W_s(z)$ ($W_s(z)$ is unimodular in $RI\!H_\infty$), and we can write (for the SISO-case):

$$\| S(z) \| \leq \| W_s(z) \| \quad \text{for all } z$$

This is the same as

$$\| W_s^{-1}(z) S(z) \|_\infty \leq 1$$

If we are interested in a **limited control power dissipation** in the closed loop system, we want to achieve a small control signal $u(z)$ with respect to the reference signal $r(z)$. From figure 1.11 we see that

$$u(z) = K(z) S(z) r(z)$$

and so we can search for a controller that makes $K(z) S(z)$ smaller than a specified level W_{ks} ($W_{ks}(z)$ is unimodular in $RI\!H_\infty$). We can write (for the SISO-case):

$$\| K(z) S(z) \| \leq \| W_{ks}(z) \| \quad \text{for all } z$$

This leads to the condition

$$\| W_{ks}^{-1}(z) K(z) S(z) \|_\infty \leq 1$$

We see that the two performance requirements can be expressed by the conditions

$$\| W_{p1}(z) S(z) \|_\infty \leq 1 \quad \text{and} \quad \| W_{p2}(z) K(z) S(z) \|_\infty \leq 1$$

where in this case $W_{p1}(z) = W_s^{-1}(z)$ and $W_{p2}(z) = W_{ks}^{-1}(z)$. If the performance levels for tracking and limited control power have to be achieved simultaneously, the conditions can be combined as:

$$\left\| \begin{bmatrix} W_{p1}(z) S(z) \\ W_{p2}(z) K(z) S(z) \end{bmatrix} \right\|_\infty \leq 1 \tag{1.11}$$

Consider figure 1.12
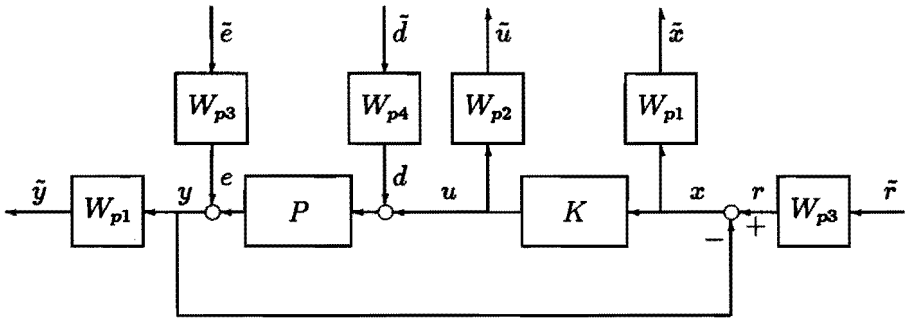
Figure 1.12: Feedback loop with weighting filters

For the configuration of figure 1.12 we can derive

$$\begin{bmatrix} x \\ u \end{bmatrix} = \begin{bmatrix} I \\ K \end{bmatrix} S [I \ P] \begin{bmatrix} r \\ d \end{bmatrix}$$

and

$$\begin{bmatrix} y \\ u \end{bmatrix} = \begin{bmatrix} I \\ K \end{bmatrix} S [I \ P] \begin{bmatrix} e \\ d \end{bmatrix}$$

By introducing the weighting filters $W_{p1}(z)$, $W_{p2}(z)$, $W_{p3}(z)$ and $W_{p4}(z)$ (all unimodular in RH_∞) , we can define performance requirements in terms of the transfers from the weighted exciting signals $\tilde{r}$, $\tilde{d}$, $\tilde{e}$ to the weighted measured signals $\tilde{y}$, $\tilde{x}$ and $\tilde{u}$. We get a more general condition, that is also applicable to the MIMO-case:

$$\left\| \begin{bmatrix} W_{p1} & 0 \\ 0 & W_{p2} \end{bmatrix} \begin{bmatrix} I \\ K \end{bmatrix} S \begin{bmatrix} I & P \end{bmatrix} \begin{bmatrix} W_{p3} & 0 \\ 0 & W_{p4} \end{bmatrix} \right\|_{\infty} \leq 1 \qquad (1.12)$$

We can achieve all kinds of other performance requirements by choosing apropriate weighting filters, and finding a controller $K(z)$ such that the condition as in (1.11) will hold. The choice of another kind of set up will also lead to a condition as in (1.11). This will be the case when we consider the performance of a closed loop configuration, where the reference signal affects the closed loop after the control operation.

Robust stability:
In the previous section we introduced various kinds of model error structures. For all these model error structures the true process can be written in the following form (Lunze [40], McFarlane [43]):

$$P_t = \Gamma_{22} + \Gamma_{21} \Delta_t (I - \Gamma_{11} \Delta_t)^{-1} \Gamma_{12} \overset{def}{=} \mathcal{F}_u(\Gamma, \Delta_t)$$

where the matrix Γ reflects both the nominal model as well as the model error structure. Γ will be called **the structure matrix** of the model and is partitioned as

$$\Gamma = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \end{bmatrix}$$

and $\Delta_t(z)$ is the true model error. The notation $\mathcal{F}_u(\Gamma, \Delta_t)$ is called an **upper linear fractional transformation.**
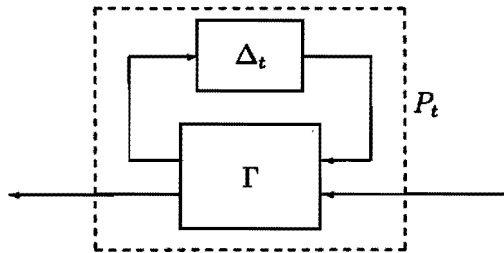
Figure 1.13: True model as an upper linear fractional transformation

As examples we can give Γ for

$$An\ additive\ model\ error:\quad \Gamma = \begin{bmatrix} 0 & I \\ I & P \end{bmatrix}$$

An input reverse multiplicative model error : $\Gamma = \begin{bmatrix} -I & I \\ -P & P \end{bmatrix}$

A left coprime factor model error:

$$\Gamma = \left[\begin{bmatrix} 0 \\ M^{-1} \\ M^{-1} \end{bmatrix} \begin{bmatrix} I \\ P \\ P \end{bmatrix} \right] \quad \text{with} \quad \Delta = [\Delta_N \ \Delta_M]$$

Now suppose we know that the H_∞-norm of the true model error $\Delta_t(z)$ is bounded by some constant ϵ:

$$\| \Delta_t(z) \|_\infty \leq \epsilon$$

Then we know that the true process is in the set of systems

$$\tilde{\mathbf{P}} = \left\{ \tilde{P}(z) \in R\!P \mid \tilde{P}(z) = \mathcal{F}_u(\Gamma(z), \Delta(z)) \text{ with } \| \Delta(z) \|_\infty \leq \epsilon \right\}$$

where Γ depends on the chosen model error structure.
The nominal model $P(z)$ is given by

$$P(z) = \mathcal{F}_u(\Gamma(z), 0) = \Gamma_{22}(z)$$

We can now consider the closed loop of $\tilde{P}(z) \in \tilde{\mathbf{P}}(z)$ with some controller $K(z)$ as in figure 1.14
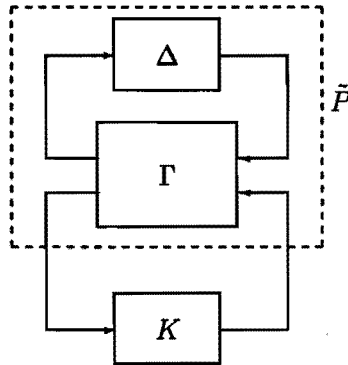
Figure 1.14: Nominal model with model error and feedback controller

For this configuration, McFarlane ([43]) gives the following theorem:

<u>Theorem 1:</u>
$K(z)$ stabilizes $P_\Delta(z) \overset{def}{=} \mathcal{F}_u(\Gamma(z), \Delta(z))$ for all $\| \Delta(z) \|_\infty \leq \epsilon$ if and only if

1. $K(z)$ stabilizes the nominal model $P(z) = \mathcal{F}_u(\,\Gamma(z), 0\,)$.

2. $\|\,\Gamma_{11} + \Gamma_{12}K(\,I - \Gamma_{22}K\,)^{-1}\Gamma_{21}\,\|_\infty \leq \epsilon^{-1}$

$\square$

(The proof is in [43]).

Condition 2 of this theorem can always be rewritten as a condition:

$$\left\|\begin{bmatrix} W_{r1} & 0 \\ 0 & W_{r2} \end{bmatrix}\begin{bmatrix} I \\ K \end{bmatrix} S[\,I \ \ P\,]\begin{bmatrix} W_{r3} & 0 \\ 0 & W_{r4} \end{bmatrix}\right\|_\infty \leq 1 \qquad (1.13)$$

for an appropriate choice of weighting filters $W_{r1}(z)$, $W_{r2}(z)$, $W_{r3}(z)$ and $W_{r4}(z)$ (all unimodular in $R\!H_\infty$).

Robust Performance:

If we have to ensure a specified performance level in the presence of model uncertainty we want to achieve robust performance. We will only focus on a special case of robust performance, namely the case of a SISO-system where we have a nominal performance condition

$$\|\,W_p(z)\,S(z)\,\|_\infty \leq 1$$

and a robust stability condition

$$\|\,W_r(z)\,K(z)\,S(z)\,\|_\infty \leq 1$$

for an additive model error structure $\Delta(z) \in R\!L_\infty$ such that

$$\|\,W_r^{-1}(z)\,\Delta(z)\,\|_\infty \leq 1$$

Doyle et al. ([12]) show that in that case a necessary and sufficient condition for robust performance will be:

$$\|\,|\,W_p(z)\,S(z)\,| + |\,W_r(z)\,K(z)\,S(z)\,|\,\|_\infty \leq 1$$

So if this condition is satisfied, we will have satisfied the performance condition even for the perturbed process.

1.4   Motivation and problem statement

In this section the problem statement will be constructed by considering the following aspects:

Models with bounded model error : Inaccuracy in modeling industrial processes causes problems in controller design. Controllers may destabilize the true process or do not meet the performance specifications. If bounds are available on the magnitudes of the model error, recently developed techniques for robust control design show that for an (inaccurate) nominal model controllers can be designed that ensure stability and performance.

A new identification method : As was already mentioned, conventional identifiaction methods only give a model without bounds on the magnitudes of the model error. New identification techniques must be developed to derive these bounds.

Limited number of assumptions on true process: The only assumption that will be made about the true process is that it is linear time invariant finite dimensional and proper (so $P_t(z) \in R\!I\!P$). No further assumptions will be made about the structure or the order of the process.

Better disturbance descriptions : If we want to derive bounds on the model error we will have to realize that this bound is partly due to disturbances and partly due to undermodelling. Therefore all disturbances are investigated separately and more intensively. Disturbances caused by the sensors and the actuators will be examined separately and uncertainty in the dynamics of the actuator will be modelled. Disturbances will be assumed to be bounded in the frequency domain by a known weighting function.

A worst case approach leading to a min-max problem: In the noiseless case we will be able to give an exact description of the true process, and in the case of undermodelling this leads to an exact description of the difference between the model and the true process. In the presence of input and output disturbances we are only able to derive a set of estimations of the true process. For every model we can derive a guaranteed upper bound for the model error by considering the worst case estimation of the true process. We can minimize the upper bound of the model error by choosing the model that gives a minimum worst case model error. This will lead to a min-max optimization problem.

A deterministic setting : The identification will be placed in a deterministic setting. Disturbances are assumed to be bounded by a known weighting function in the frequency domain. The fact that only a bounding function is required means that exact knowledge about colouring of the noise is not necessary. The identification will be done in the frequency domain. We aim at minimum bounds for the model error in the frequency domain. It is very convenient to work in one domain, where both the noise bounding filters and the optimization criterion are defined.

MIMO-systems and different model error structures :   Most industrial processes are multivariable. We like to find identification methods that are applicable to MIMO processes. In section 1.3 the necessity of various model error structures was mentioned. For all structures we like to be able to find bounds on the model error.

It is clear that the requirements are very different from those of the conventional identification methodologies. We are not looking for the best estimate of the model in a probabilistic sense. Instead we look for the model that minimizes the worst case estimate of the model error. The new approach proceeds along the following lines:

1. We collect measured data from input and output signals, in a finite time interval $k = 1, \ldots, 2N$. This results in the data set $\{ u(k), y(k) \}$.

2. We have apriori knowledge about bounds on the disturbances, additive on the input and output signals. Disturbance bounds will be specified in the frequency domain.

3. We choose a model set $\mathbf{P}$ with parametrized models $P(\theta, z)$ and parameter vectors $\theta \in \Theta$, i.e. $\mathbf{P} = \{ P(\theta, z) \in R\!I\!P \mid \theta \in \Theta \}$, where Θ is a set.

4. We choose a structure for our model error $\Delta(P(\theta, z), P_t(z))$.

5. We choose a deterministic model error criterion $J(\Delta)$, typically a norm.

6. We have a stabilizing controller $K(z) \in R\!I\!P$ for the true process $P_t(z)$.

A very important notion we use in this thesis will be that of the set of unfalsified systems $\tilde{\mathbf{P}}$. This set is defined as the set of all systems $\tilde{P}(z) \in R\!I\!P$ that do not falsify the data set and the noise bounds. Of course the true process will be an element of this set.

We will consider the model error criterion $J\big(\Delta(P(\theta, z), \tilde{P}(z))\big)$ for all elements $\tilde{P}(z)$ in $\tilde{\mathbf{P}}$ and $\theta \in \Theta$. Since $P_t(z) \in \tilde{\mathbf{P}}$, the worst-case (so largest)

model error criterion will always be worse (larger) than the criterion for the true process:

$$\forall \theta \in \Theta, \quad \max_{\tilde{P} \in \tilde{\mathbf{P}}} J\Big(\Delta(P(\theta, z), \tilde{P}(z))\Big) \geq J\Big(\Delta(P(\theta, z), P_t(z))\Big)$$

In this way we can derive an upper bound for the model error criterion of the true process.
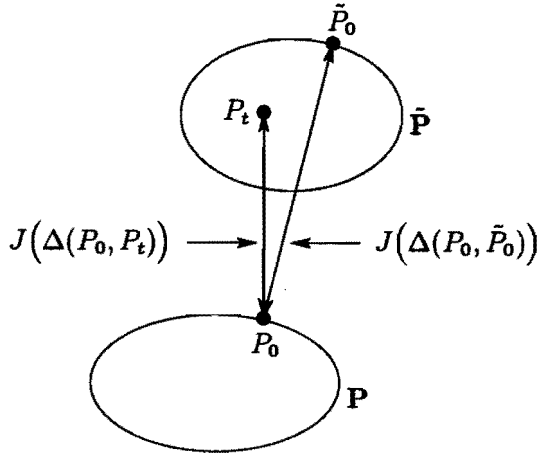
Figure 1.15: Visualization of optimizing a worst case model error

We can visualize the concept of minimizing the worst case model error in figure 1.15. We consider the model error between models in the model set $\mathbf{P}$ and systems in the set of unfalsified systems $\tilde{\mathbf{P}}$. For simplicity we take the euclidian distance for $J\Big(\Delta(P, \tilde{P})\Big)$. So in the figure we consider the distance between points in the set $\mathbf{P}$ and points in the set $\tilde{\mathbf{P}}$. We look for a model P in $\mathbf{P}$ with a minimum worst case distance with respect to a system in the set $\tilde{\mathbf{P}}$. Clearly in this figure the optimal model will be P_0 and the corresponding worst case system in the set $\tilde{\mathbf{P}}$ will be $\tilde{P}_0$.

Motivated by the foregoing, we can formulate the problem statement for the work in this thesis as follows:

Problem statement:

Given a model set **P** and the set of unfalsified systems $\bar{\mathbf{P}}$, determined by the data and a priori knowledge of the signal errors. Find a model $P \in \mathbf{P}$ that minimizes the maximum model error criterion $J(\Delta)$, over all model errors $\Delta(P, \bar{P})$ (so within the adopted model error structure) between that model P and the worst-case system $\bar{P} \in \bar{\mathbf{P}}$.

Formally, we wish to find a model $P_{opt} \in \mathbf{P}$ that satisfies

$$P_{opt} = \arg \inf_{P \in \mathbf{P}} \sup_{\bar{P} \in \bar{\mathbf{P}}} J\left(\Delta(P, \bar{P})\right)$$

This gives us an explicit model error bound

$$\gamma = \sup_{\bar{P} \in \bar{\mathbf{P}}} J\left(\Delta(P_{opt}, \bar{P})\right)$$

1.5 Organization of the thesis

A brief description will be given of what is contained in this thesis:
The succesion of the chapters is represented by figure 1.16.

The thesis consists of four main parts:

A. Preliminaries : In this part the problem area is explored, and important concepts are studied.

B. Formalization : In this part the original problem is formalized and transformed into an optimization problem, that is numerically solvable.

C. Optimization : In this part the optimization itself is considered and the evaluation of the final model is discussed.

D. Evaluation : The proposed methods are evaluated by a case study, resulting in some conclusions and remarks.

PART A: Preliminaries:

In chapter 2 some practical considerations are done and preliminary steps for the identification are discussed. A detailed experimental set up will be introduced, that can be reduced to a basic experimental set up. The disturbance signals will be characterized in terms of noise sets and frequency domain bounding functions will be defined. Finally, we discuss the transformation of the measured data from the time domain into the frequency domain. We will consider the errors that may occur, if we do not set up the experiments properly.

Chapter 3 gives a first attempt to derive model error bounds. For reasons of simplicity we restrict ourselves to the identification of a SISO-system with an additive or a multiplicative model error structure. Uncertainty regions are derived for the process dynamics in the complex plane, and an upper bound for the model error by H_∞-fitting of the model is minimized. The chapter concludes with a simulation study.

In chapter 4 we discuss how experiments should be set up if we want to identify multivariable systems.

In chapter 5 we look at various kinds of model error structures. We will show how these model error structures fit in a basic scheme with coprime factor models.

PART B: Formalization:

In chapter 6 the results of chapter 1 to 5 are combined. We show, that for a fixed model the true model error can be expressed systematicaly as a function of the model error structure, the chosen model ,the measured data, the noise bounding functions and some unknown scaled true noise matrix.

In chapter 7 bounds are derived for a general model error, using knowledge of the noise sets.

In chapter 8 we will consider the parametrization of the model, and its relation to optimality and optimization techniques.

PART C: Optimization:

In chapter 9 the optimization of the model in the case of a linear parametrization will be considered. We will show that this leads to a convex optimization problem.

Chapter 10 treats the evaluation of the models.

PART D: Evaluation:
In chapter 11 we will consider a case study. The identification of a multi-variable laboratory process (a water vessel process) will be done. A linear parametrization (chapter 9) will be used and minimum model error bounds will be derived.

Conclusions and remarks are given in chapter 12.
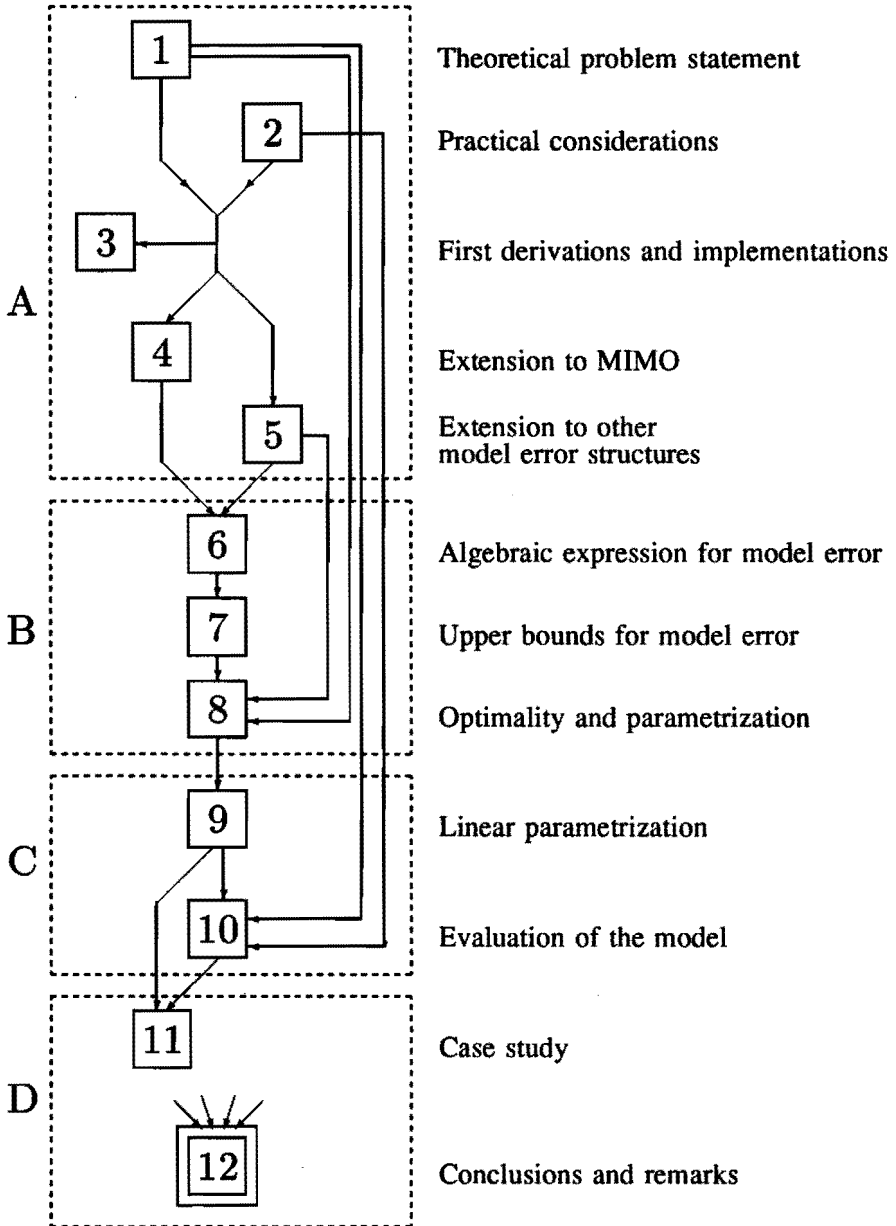
Figure 1.16: Succession of the chapters of this thesis

2

Practical considerations

2.1 Introduction

In this chapter we will consider some preliminaries for the identification and we will make some practical considerations before we go on with a theoretical analysis. We start with a detailed set up where most disturbance sources are discussed. Disturbances caused by the sensors and actuators will be examined separately and uncertainty in the dynamics of the actuator will be modelled (Van den Boom et al. [59]). Next, this detailed set up will be reduced to a basic set up. The identification will be placed in a deterministic setting and the noise and disturbance signals will be assumed to be bounded in the frequency domain by a known bounding function. The fact that only a bounding function is required means that exact knowledge about colouring of the disturbance signals is not necessary. We will consider the transform from the time domain into the frequency domain and the errors that can occur.

2.2   Plant environment

In this section we consider the industrial environment in which a plant is usually situated. Figure 2.1 gives a block scheme with 5 components: Plant, acturator, controller, input sensor and output sensor.
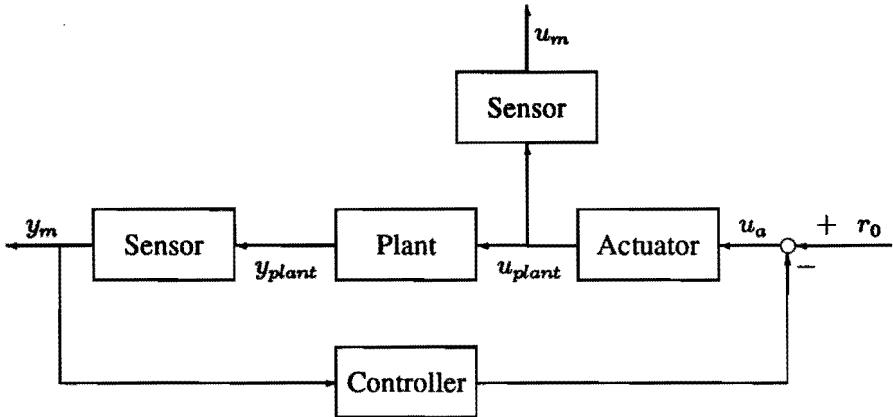
Figure 2.1: Plant with actuator, sensors and controller

The plant under study may be a multi-input multi-output system. The signal vector u_{plant} will be the true physical input signal and allows us to control the plant. The physical output signal of the plant, in which we are interested is the signal vector y_{plant}. The signal vectors y_{plant} and u_{plant} are measured by sensors, devices that transform measured physical quantities into electrical signals. They are used to measure quantities like temperature, pressure, displacement and others. The actuator transform the electrical signal into a physical signal.

The plant: We assume the plant to be a linear time-invariant finite dimensional system. We will call this system the true process and we denote its transfer function by $P_t(z)$. In practice, the plant may show some smooth non-linear behaviour or the dynamics may change in time. The part of the output signal that is caused by non-linear or time-varying behaviour, is seen as a part of the process disturbance. Consider figure 2.2

The true process $P_t(z)$ is excited by the input signal u_{plant}, resulting in an output signal y_t. This signal y_t will be denoted as the true output signal. It is the part of the output signal y_{plant}, that is due to convolution of the input signal u_{plant} and the impulse response of the linear time-invariant true process $P_t(z)$:
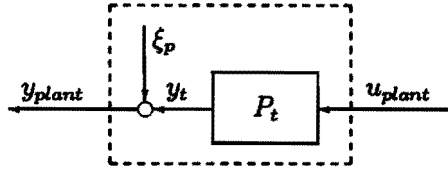
Figure 2.2: Detailed description of the plant

$$y_t(k) = P_t(\tau)u_{plant}(k)$$

The part of the physical output signal y_{plant} that is not explained by
the true output signal y_t is denoted by the signal ξ_p. It represents the
process disturbance and error-signals due to non-linear and time varying
behaviour.

The true process $P_t(z)$ is the system that we will focus on. The trans-
fer function $P_t(z)$ is unknown and in this thesis we consider a system
identification method that yields a nominal model for the plant with
minimum bounds on the model error.

Actuator : The input signal u_{plant} is generated by an actuator. The input u_a
of the actuator is known, because the reference signal and the output
signal of the controller are available.
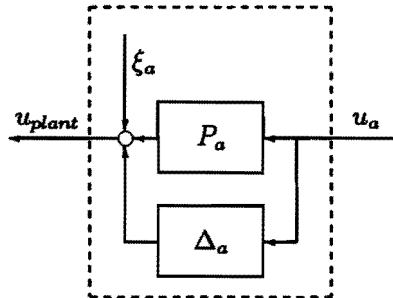
Figure 2.3: Detailed description of the actuator

The actuator can show a lot of dynamics within the bandwidth of the
process. Usually the actuator can be studied in advance, and we can
model it as a linear time-invariant system with a transfer function $P_a(z)$.
If there is some uncertainty in this actuator model $P_a(z)$, we can rep-
resent this uncertainty in some model error structure, e.g. an additive
uncertainty description as in figure 2.3. with an additive uncertainty
$\Delta_a(z)$. An actuator disturbance signal ξ_a may act on the output of the
actuator. If the actuator is very fast and its dynamics can mostly be

neglected compared to the process dynamics, the transfer functions of P_a and Δ_a will become constant matrices.

Input and output sensors: For the input and output sensors we can use the same modelling as for the actuator.
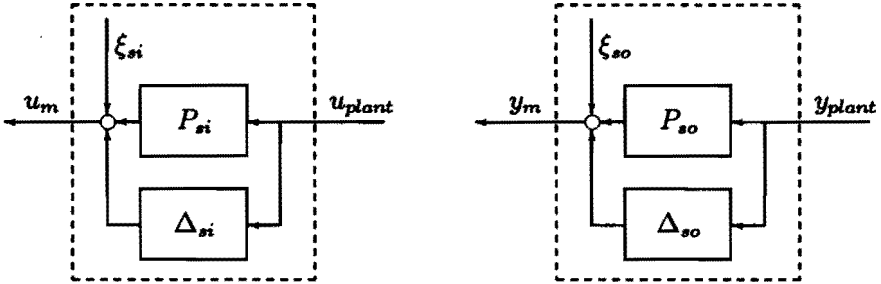
Figure 2.4: Detailed description of the sensors

For the input sensor transfer we have a linear time-invariant model P_{si} with the possibillity for an additive uncertainty Δ_{si}, for the output sensor transfer we have a linear time-invariant model P_{so} with the possibillity for an additive uncertainty Δ_{so}. Finally we contribute the additive measurement noise signals ξ_{mi} and ξ_{mo}, resulting in the measured input signal u_m and the measured output signal y_m.

Controller : The process is stabilized by a controller $K_s(z)$. In the case of a stable process this controller may be $K_s(z) = 0$.

Reference signal : The closed loop is excited by a reference signal r_0. Note that the reference signal affects the closed loop after the control operation. If this is not actually the case, we can easily transform it to the scheme of figure 2.1 because the controller and reference signal are known: the reference signal then is filtered by $K_s(z)$ or a part of it.

The set up, as presented here has the advantage that sensor and actuators are not necessarily incorporated in the plant. They are involved in the scheme as separate components and all error sources have their proper place. The choice of such a detailed set up will have some consequences. We will have to obtain more information about the noise and disturbances than in the conventional stochastic identification methods, where a data set $\{u_m(k), y_m(k)\}$ is sufficient. Sensors and actuators will have to be examined separately to determine the characteristics of the disturbances and to find the actuator transfer with its uncertainty.

A nice way to describe the noise and disturbance signals is by using norm-bounded sets, as will be discussed in section 2.4 .

2.3 Experimental set up

In the previous section a detailed experimental set up was given. Parts of this detailed concept will now be reduced to a so called **basic set up**, as given in figure 2.5. This basic set up will be used in this thesis for the theoretical analysis and the identification method.
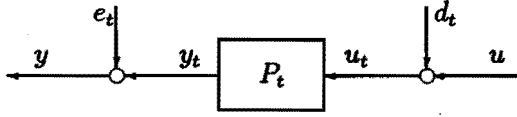
Figure 2.5: Basic experimental set up

The process $P_t(z)$ is excited by a disturbed input signal $u_t(k)$, resulting in an output signal $y_t(k)$. These signals are measured in the input signal $u(k)$ and output signal $y(k)$, with

$$u_t(k) = u(k) + d_t(k)$$
$$y_t(k) = y(k) - e_t(k)$$
$$y_t(k) = P_t(\tau)u_t(k)$$

The true process $P_t(z)$ is assumed to be linear time-invariant. The true noise signals $d_t(k)$ and $e_t(k)$ are unknown, but will be assumed to belong to a certain class (this will be discussed in the following section).

The signals in the set up of figure 2.1 can be translated into the signals in the basic set up of figure 2.5, using the detailed descriptions of the components as in the figures 2.2, 2.3 and 2.4.

About the output disturbance signal e_t :
The measurement of the true output $y_t(k)$ is the signal $y(k)$. From figure 2.2 and figure 2.4 we can derive

$$y_m = P_{so} y_t + (P_{so} + \Delta_{so})\xi_p + \Delta_{so} y_t + \xi_{so}$$

If the transfer function P_{so} is unimodular we can define

$$y = P_{so}^{-1} y_m$$
$$e_t = (I + P_{so}^{-1} \Delta_{so})\xi_p + P_{so}^{-1}\Delta_{so} y_t + P_{so}^{-1} \xi_{so}$$

The signal $e_t(k)$ is not known, but is assumed to belong to a certain set $\tilde{\mathbf{E}}$. We now define the set

$$\tilde{\mathbf{Y}} = \left\{ \tilde{y}(k) \in \mathbb{L}_2[0, 2N-1] \,|\, \tilde{y}(k) = y(k) - \tilde{e}(k) \,,\; \tilde{e}(k) \in \tilde{\mathbf{E}} \right\}$$

So $y_t(k)$ is in the set $\tilde{\mathbf{Y}}$.

Van den Boom et al. [59] show how $\mathbf{Y}$ can be found for an infinity-norm bounded set $\mathbf{E}$.

About the input disturbance signal d_t :

The true input signal $u_t = u_{plant}$ is generated by the actuator, and with figure 2.3 we derive

$$u_t(k) = P_a(\tau) u_a(k) + \Delta_a(\tau) u_a(k) + \xi_a(k)$$

and from figure 2.4 we see that the measurement of this input signal is equal to

$$u_m(k) = P_{si}(\tau) u_t(k) + \Delta_{si}(\tau) u_t(k) + (P_{si}(\tau) + \Delta_{si}(\tau))\xi_{si}(k)$$

The interpretation of the input noise signal $d_t(k)$ depends on the question, which signal measurements are available, either $u_m(k)$ or $u_a(k)$, or both.

(1). Consider the case, where we only have the actuator input signal $u_a(k)$ available. We see from figure 2.3 that we can define

$$\begin{aligned} u &= P_a u_a \\ d_{ta} &= \Delta_a u_a + \xi_a \end{aligned}$$

The signal $d_{ta}(k)$ is not known, but is assumed to belong to a certain set $\tilde{\mathbf{D}}_a$. We now define a set

$$\tilde{\mathbf{U}}_a = \left\{ \tilde{u}_a(k) \in \mathbb{L}_2[0, 2N-1] \,|\, \tilde{u}_a(k) = u(k) + \tilde{d}_a(k) \,,\, \tilde{d}_a(k) \in \tilde{\mathbf{D}}_a \right\}$$

So $u_t(k)$ is in the set $\tilde{\mathbf{U}}_a$.

(2). Next consider the case where we only have the input measurement signal $u_m(k)$ available. If we assume the transfer function P_{si} to be unimodular, we derive:

$$\begin{aligned} u &= P_{si}^{-1} u_m \\ d_{ts} &= -P_{si}^{-1} \Delta_{si} u_t - (I + P_{si}^{-1} \Delta_{si})\xi_{si} \end{aligned}$$

The signal $d_{ts}(k)$ is not known, but is assumed to belong to a certain set $\tilde{\mathbf{D}}_s$. We now define a set

$$\tilde{\mathbf{U}}_s = \left\{ \tilde{u}_s(k) \in \mathbb{L}_2[0, 2N-1] \,|\, \tilde{u}_s(k) = u(k) + \tilde{d}_s(k) \,,\, \tilde{d}_s(k) \in \tilde{\mathbf{D}}_s \right\}$$

So $u_t(k)$ is in the set $\bar{U}_s$.

(3). In the final case we have available the actuator signal $u_a(k)$ as well as the measurement signal $u_m(k)$. We know that $u_t(k)$ is in the intersection of the sets $\bar{U}_a$ and $\bar{U}_s$, defined by the set $\bar{U} = \bar{U}_a \cap \bar{U}_s$. If we choose some fixed signal $u(k)$ then we can define the set

$$\bar{D} = \left\{ \bar{d}(k) \in L_2[0, 2N-1] \, | \, \bar{d}(k) = \bar{u}(k) - u(k), \, \bar{u}(k) \in \bar{U} \right\}$$

We know that $u_t(k)$ is in the set $\bar{U}$ and $d_t(k) = u_t(k) - u(k)$ is in the set $\bar{D}$. For the fixed signal $u(k)$ we can make various choices. We can either choose $u(k) = P_{si}^{-1}(\tau) u_m(k)$ or $u(k) = P_a(\tau) u_a(k)$ or some interpolation between these two signals. If we fit the signal $u(k)$ in 'the center' of the set $\bar{U}$ we are sure that the set $\bar{D}$ is 'centered' around zero. This can be a nice feature if we want to describe the set $\bar{D}$ in terms of norm-bounded sets, as will become clear in the next section. Van den Boom et al. [59] describe the set $\bar{D}$ for infinity-norm bounded sets.

2.4 Assumptions on the disturbances

In an industrial environment many disturbance sources can be recognized. They can be modelled in two distinct ways:

Stochastic disturbance : The disturbance is assumed to have a stochastic character and it can be described by a probability density function. This description can be very accurate if, for example, we consider noise due to natural processes, e.g. thermal noise, nuclear radiation noise.

Deterministic disturbance : The disturbance is assumed to be fully determined by some non-stochastic cause. The causing signal may or may not be known. As an example we consider quantization errors (due to AD and DA converters), errors due to non-linearities in the sensors and actuators (like dead-zone in DC-motors, hysteresis) and errors with an oscillating character (like 50Hz interference of a power supply, fluctuations in a flow due to a periodic pump).

The choice of how to characterize the disturbance highly depends on the purpose of the identification. If we want a model of our process that is best in some probability sense, we can use a stochastic description of the disturbance. If we want to derive hard bounds on the model error, it is not convenient to choose a stochastic characterization of the disturbance, especially if we are

not sure that the noise signals will indeed satisfy the stochastic assumptions. In that case we can better use a deterministic description of the noise.

Suppose the z-transforms of the measured time signals $u(k)$, $y(k)$ and the true noise signals $d_t(k)$, $e_t(k)$ are given by the spectra $u(z)$, $y(z)$ and $d_t(z)$, $e_t(z)$ respectively. In the SISO-case, we can calculate the true process $P_t(z)$ by dividing the true output signal $y_t(z)$ by the true input signal $u_t(z)$, leading to:

$$P_t(z) = \frac{y_t(z)}{u_t(z)} = \frac{y(z) - e_t(z)}{u(z) + d_t(z)}$$

An estimator for the true process is given by:

$$\hat{P}(z) = \frac{y(z) - \tilde{e}(z)}{u(z) + \tilde{d}(z)}$$

where $\tilde{d}(z)$ and $\tilde{e}(z)$ are in the sets $\tilde{\mathbf{D}}$ and $\tilde{\mathbf{E}}$. It is clear that, if we do not assume the magnitude of the signals $d_t(z)$ and $e_t(z)$ to be bounded, the estimate $\hat{P}(z)$ for a specific $z \in C$ can take any value in the complex plane, so that whatever model error is considered, it will be unbounded.

Considering this, a straightforward way of characterizing the disturbance is to assume the z-transforms of the noise signals to be bounded:

$$|d_t(z)| \leq |W_d(z)| \quad \text{and} \quad |e_t(z)| \leq |W_e(z)| \quad \text{for} \quad z \in C$$

where $W_d(z)$ and $W_e(z)$ are supposed to be known bounding functions (figure 2.6).
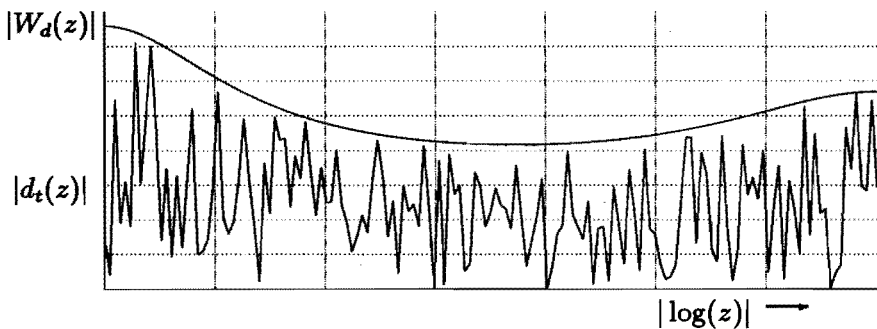
Figure 2.6: True noise signal and bounding function

Now we can define two noise sets $\tilde{\mathbf{D}}$ and $\tilde{\mathbf{E}}$:

$$\tilde{\mathbf{D}} = \left\{ \tilde{d}(k) \in \mathbb{L}_2[0, 2N - 1] \,|\, |\tilde{d}(z)| \leq |W_d(z)| \,,\ \text{for}\ z \in C \right\}$$

$$\tilde{\mathbf{E}} = \{ \tilde{e}(k) \in \mathbb{L}_2[0, 2N-1] \mid |\tilde{e}(z)| \leq |W_e(z)| \ , \ \text{for} \ z \in \mathcal{C} \}$$

It is clear that $d_t(z) \in \tilde{\mathbf{D}}$ and $e_t(z) \in \tilde{\mathbf{E}}$.

Finally we have to make a few remarks:

- We only use an upper bound for the z-transform of the disturbance, so we do not need to know the colouring of the noise exactly.

- The energy of the disturbance signals that are considered, depends on the length of the observation interval $(2N)$. Consequently the functions $W_d(z)$ and $W_e(z)$ will depend on the number N.

- If we have chosen for modelling the noise using Gaussian distributions, there is, strictly speaking, no upper bound on the z-transform of noise signal. However, in practice we can use a confidence interval for the magnitude in the frequency domain as a bound for the noise (e.g. 3σ-bound, or if we need a very secure bound we can take a 6σ-bound).

2.5 Signals and systems in the frequency domain

Signals in the frequency domain :
Experimental data is obtained by collecting a finite number of samples at $t = kT$ from the reference signal $r(k)$, the input signal $u(k)$ and the resulting output signal $y(k)$. So the data $\{r(k), u(k), y(k)\}, k = 0, .., 2N-1$, is given in the time-domain. However, we would like to derive bounds on the model error in the frequency domain. Therefore the identification will be done in the frequency domain.
For the transformation from the discrete time domain into the frequency domain we would like to use the z-transform. The z-transform however, is defined for time signals of infinite length, where as our signals are only given in the observation interval $k = 0, .., 2N-1$. If we pad the signals $r(k)$, $u(k)$ and $y(k)$ with zeros for $k < 0$ and $k \geq 2N$, we can compute the z-transform. However, this can be a source of transformation errors, if the signals are not really zero outside the observation interval. We like to keep these errors as small as possible. To achieve this we have to design a suitable reference signal $r(k)$. Furthermore we have to make careful choices for the sampling time T and the length of the observation interval $2N$.
We consider the set up in figure 2.7
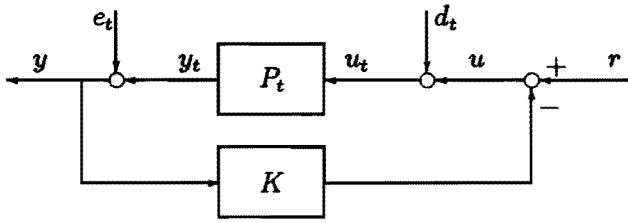
Figure 2.7: Experimental set up

First the noiseless case ($d_t = 0$ and $e_t = 0$) :

$$y_t(k) = P_t(\tau)(I + K(\tau)P_t(\tau))^{-1}r(k)$$
$$u_t(k) = (I + K(\tau)P_t(\tau))^{-1}r(k)$$

Assume that ω_B is the bandwidth of the original continuous time system $P_{ct}(j\omega)$ and that we have chosen a sampling time $T \ll \frac{\pi}{\omega_B}$. Further, let $l = \max(l_1, l_2)$, where l_1 is the length of the impulse response $h_1(k)$ of $P_t(z)(I + K(z)P_t(z))^{-1}$ and l_2 is the length of the impulse response $h_2(k)$ of $(I + K(z)P_t(z))^{-1}$. It is a well-known fact that systems with a finite bandwidth as well as a finite impulse response length do not exist. In practice, however, we can determine ω_B in such a way that the magnitudes of the transfer function $P_{ct}(j\omega)$ of the system is attenuated sufficient enough for higher frequencies ($\omega > \omega_B$) and we can determine l in such a way that the impulse response will be approximately zero for $k > l$. Then we choose the length of the observation interval $2N \gg l$.

We make a reference signal with $r(k) = 0$ for $k < 0$ and $r(k) = 0$ for $k \geq 2N - l$, so the z-transform of $r(k)$ will be exact. By choosing this reference signal we make sure that in the noiseless case $u(k) = 0$ and $y(k) = 0$ for $k < 0$ and for $k \geq 2N$, so that the z-transform does not introduce any errors (figure 2.8).

If noise is present, the signals $d_t(k)$ and $e_t(k)$ will introduce errors at the begin and at the end of the observation interval. In the configuration of figure 2.8 the signals $u(k)$ and $y(k)$ can be derived:

$$y(k) = S_1(\tau)P_t(\tau)r(k) + S_1(\tau)P_t(\tau)d_t(k) + S_1(\tau)e_t(k)$$
$$u(k) = S_2(\tau)r(k) + S_2(\tau)K(\tau)P_t(\tau)d_t(k) + S_2(\tau)K(\tau)e_t(k)$$

where

$$S_1(\tau) = (I + P_t(\tau)K(\tau))^{-1}$$
$$S_2(\tau) = (I + K(\tau)P_t(\tau))^{-1}$$
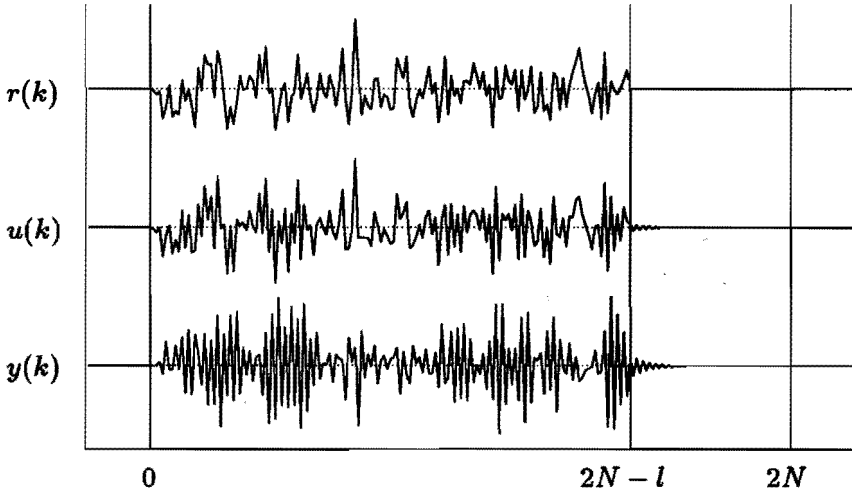
Figure 2.8: Observation interval for signals $r(k)$, $u(k)$ and $y(k)$

Now define

$h_1(k)$	is the impulse response of	$S_1(z)P_t(z)$	with length l_1
$h_2(k)$	is the impulse response of	$S_1(z)$	with length l_2
$h_3(k)$	is the impulse response of	$S_2(z)K(z)P_t(z)$	with length l_3
$h_4(k)$	is the impulse response of	$S_2(z)K(z)$	with length l_4

The input and output signal at time $t = kT$ with $k \geq 0$ are partly due to the noise signals at time $t = kT$ with $k \leq 0$. We denote these parts of $y(k)$ and $u(k)$ by $y_b(k)$ and $u_b(k)$ respectively, and we derive

$$y_b(k) = \sum_{n=k}^{l_1} h_1(n)\, d_t(k-n) + \sum_{n=k}^{l_2} h_2(n)\, e_t(k-n)$$

$$u_b(k) = \sum_{n=k}^{l_3} h_3(n)\, d_t(k-n) + \sum_{n=k}^{l_4} h_4(n)\, e_t(k-n)$$

In an analogous way, the noise signals at time $t = (2N+k)T$ with $k \leq 0$ have a response in the input and output signals at time $t = (2N+k)T$ with $k \geq 0$. We denote these parts of $y(k)$ and $u(k)$ by $y_e(k)$ and $u_e(k)$ respectively, and we derive

$$y_e(2N + k) = \sum_{n=k}^{l_1} h_1(n)\, d_t(2N + k - n) + \sum_{n=k}^{l_2} h_2(n)\, e_t(2N + k - n)$$

$$u_e(2N + k) = \sum_{n=k}^{l_3} h_3(n)\, d_t(2N + k - n) + \sum_{n=k}^{l_4} h_4(n)\, e_t(2N + k - n)$$

So that the errors in z-transform for the input and output signal become:

$$\epsilon_y(z) = \sum_{k=1}^{max(l_1,l_2)} \left(y_b(k)\, z^{-k} + y_e(2N + k)\, z^{-(2N+k)} \right) , \qquad z \in \mathcal{C}$$

$$\epsilon_u(z) = \sum_{k=1}^{max(l_3,l_4)} \left(u_b(k)\, z^{-k} + u_e(2N + k)\, z^{-(2N+k)} \right) , \qquad z \in \mathcal{C}$$

With some additional a priori knowledge we could compute an upper bound for these errors. However we will assume that the noise will not be too big and that $N \gg l$, so that the error in the z-transforms of $u(k)$ and $y(k)$ due to $d_t(k)$ and $e_t(k)$ will be negligible.

The assumptions about the length of impulse response of the various transfers can always be verified afterwards. See chapter 11. We substitute the final model $P(z)$ for the true process $P_t(z)$ and determine the lengths l_1, l_2, l_3 and l_4. Of course this check will only be valid if the final model is close enough to the true process.

Systems in the frequency domain :

- We are interested in bounds on the model error $\Delta(z)$ in some model error structure. In this thesis we will derive an upper bound for the matrix norm of the model error at specific frequencies:

$$\| \Delta(z_i) \| \leq \gamma(z_i) \quad \text{where} \quad z_i \in \Omega$$

However, the goal of the work is to find an upper bound for the continuous frequency domain:

$$\| \Delta(z) \| \leq \gamma_c(z) \quad \text{for all} \quad z \in \mathcal{C}$$

Now let the impulse response of $\Delta(z)$ be denoted by $h_\Delta(k)$, $k = 0, .., \infty$. If we can guarantee that $h_\Delta(k) = 0$ for $k > l_\Delta$ where $l_\Delta \ll N$, then we know that $\Delta(z)$ will be a smooth function over $\Delta(z_i)$, $z_i \in \Omega$. If we define a smooth interpolated function $\gamma_c(z)$ for $\gamma(z_i)$ it is clear that exist a very small $\eta \geq 0$ such that

$$\| \Delta(z) \| \leq \gamma_c(z) + \eta \quad \text{for all} \ z \in \mathcal{C}$$

This phenomenon can be explained by the duality of frequency domain and time domain. If we sample a continuous time signal with a sampling frequency that is much larger than the bandwidth, then a smooth interpolation of the sampled signal will return the original continuous time signal very accurately. In the same way, if we sample a continuous frequency response in $2N$ points over the unit circle, where $2N$ is much larger than the length of the corresponding impulse response, then a smooth interpolation of the sampled frequency response will return the original continuous frequency response very accurately.

- In the case of an additive model error structure the finite length constraint on the model error impulse response $h_\Delta(k) = 0$ for $k > l_\Delta$, $l_\Delta \ll N$ can be translated in a constraint on the impulse response $h_t(k)$ of the true process and the impulse response $h(k)$ of the model: A sufficient condition for

$$h_\Delta(k) = 0 \ \text{for} \ k > l_\Delta, \ l_\Delta \ll N$$

is the condition

$$h_t(k) = 0 \ \text{and} \ h(k) = 0 \ \text{for} \ k > l_\Delta \ \text{where} \ l_\Delta \ll N$$

For the other model error structures equivalent conditions on a smooth interpolation can be derived.

- If we have a bound on the tail of the impulse response of the model error, we can find bounds on errors in the infinity-norm and two-norm (see section 1.1). If we have the assumption

$$\sum_{k=l_\Delta}^{\infty} \| h_\Delta(k) \| \leq \epsilon_\Delta$$

then the errors in the infinity-norm and the two-norm, caused by neglecting the tail are always smaller than this value ϵ_Δ .

So, even if the impulse response of the model error is not exactly zero for $k > l_\Delta$, but the tail contribution of the impulse responses is small enough, we can neglected the error for practical use.

Resuming : When we choose the length $2N$ of the observation interval for the measurements, we will have to satisfy

$$
\begin{aligned}
l_1 &\ll 2N \\
l_2 &\ll 2N \\
l_\Delta &\ll 2N
\end{aligned}
$$

where

l_1 is the length of the impulse response

$$h_1(k) \quad \text{of} \quad P_t(z)(I + K(z)P_t(z))^{-1}$$

l_2 is the length of the impulse response

$$h_2(k) \quad \text{of} \quad (I + K(z)P_t(z))^{-1}$$

l_Δ is the length of the impulse response

$$h_\Delta(k) \quad \text{of} \quad \Delta(z)$$

In chapter 10 we will discuss how to get an indication whether the assumptions on the length of the impulse responses are realistic or not.

3

Identification of SISO-systems

3.1 Introduction

In this chapter an exploratory study will be done to get some insight into the identification techniques, which result in a bound on the model error. For simplicity we will restrict ourselves to SISO-systems, and we will only consider an additive and a multiplicative model error structure.

Consider the set up of figure 3.1 where a SISO linear time-invariant 'true' process $P_t(z)$ is excited by an unknown true input signal $u_t(z)$ that results in an unknown true output signal $y_t(z)$. The true input and output signals are perturbed by additive noise, $d_t(z)$ and $e_t(z)$ respectively.

We have a dataset $\{ u(z) , y(z) \}$ with $z \in C$ and we have descriptions of two sets for the noise signals in the frequency domain:

$$\tilde{\mathbf{D}} = \{ \tilde{d}(k) \in \mathbb{L}_2[0, 2N-1] \big| \; |\tilde{d}(z)| \leq |W_d(z)| \, , \, z \in C \}$$

$$\tilde{\mathbf{E}} = \{ \tilde{e}(k) \in \mathbb{L}_2[0, 2N-1] \big| \; |\tilde{e}(z)| \leq |W_e(z)| \, , \, z \in C \}$$
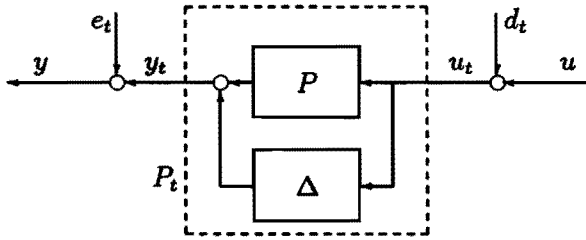
Figure 3.1: Basic experimental set up with additive model error

and we assume that $d_t(z) \in \bar{\mathbf{D}}$ and $e_t(z) \in \bar{\mathbf{E}}$.

Now we concentrate on two problems, the additive model error optimization problem and the multiplicative model error optimization problem:

Additive model error optimization problem:
Find a model $P(z)$ in a given model set $\mathbf{P}$ such that the H_∞-norm of the weighted additive model error

$$\Delta_{a,t}(z) = P_t(z) - P(z)$$

is minimized, so:

$$\inf_{P \in \mathbf{P}} \| W_\Delta(z) \Delta_{a,t}(z) \|_\infty$$

for some given weighting filter $W_\Delta(z)$.
To make sure that the additive model error is stable, we assume that the true process $P_t(z)$ and the model $P(z)$ are stable.
Because of the disturbance signals $d(z)$ and $e(z)$, we will not be able to determine the true process $P_t(z)$ exactly and thus we cannot compute the H_∞-norm of the model error. We will show that with the use of the noise sets $\bar{\mathbf{D}}$ and $\bar{\mathbf{E}}$ we are able to calculate an upper bound for the H_∞-norm of the model error. Instead of minimizing the H_∞-norm itself we will minimize the upper bound for the H_∞-norm of the model error.

Multiplicative model error optimization problem:
Find a model $P(z)$ in a given model set $\mathbf{P}$ such that the H_∞-norm of the weighted multiplicative model error

$$\Delta_{m,t}(z) = (\, P_t(z) - P(z) \,) P^{-1}(z) = P_t(z) P^{-1}(z) - 1$$

is minimized, so:

$$\inf_{P \in \mathbf{P}} \| W_\Delta(z) \Delta_{m,t}(z) \|_\infty$$

To make sure that the multiplicative model error is stable, we assume that the true process $P_t(z)$ and the model $P(z)$ are both stable and minimum-phase. In the same way as in the additive model error case, we will minimize an upper bound for the H_∞-norm of the model error, in stead of minimizing the H_∞-norm itself.

The first step in the identification procedure is to derive uncertainty regions for the system dynamics in the complex frequency plane. The second step will be to find an approximate model that is optimal in the sense that the upper bound for the weighted additive model error or the weighted multiplicative model error is minimized.

3.2 Derivation of uncertainty regions

From figure 3.1 we can easily see that

$$u_t(z) = u(z) + d_t(z) \quad \text{and} \quad y_t(z) = y(z) - e_t(z)$$

However, we do not know the signals $d_t(z)$ and $e_t(z)$, we only know that they belong to $\tilde{\mathbf{D}}$ and $\tilde{\mathbf{E}}$ respectively. Therefore we define the following sets:

$$\tilde{\mathbf{U}} = \left\{ \tilde{u}(k) \in I\!\!L_2[0, 2N - 1] \; \middle| \; \tilde{u}(z) = u(z) + \tilde{d}(z) \; , \; \tilde{d} \in \tilde{\mathbf{D}} \; , \; z \in \mathcal{C} \right\}$$
$$\tilde{\mathbf{Y}} = \left\{ \tilde{u}(k) \in I\!\!L_2[0, 2N - 1] \; \middle| \; \tilde{y}(z) = y(z) - \tilde{e}(z) \; , \; \tilde{e} \in \tilde{\mathbf{E}} \; , \; z \in \mathcal{C} \right\}$$

For all $\tilde{u} \in \tilde{\mathbf{U}}$ we have

$$|\tilde{u}(z) - u(z)| \le |W_d(z)| \; , \; z \in \mathcal{C}$$

and for all $\tilde{y} \in \tilde{\mathbf{Y}}$ we have

$$|\tilde{y}(z) - y(z)| \le |W_e(z)| \; , \; z \in \mathcal{C}$$

Note that $u_t \in \tilde{\mathbf{U}}$ and $y_t \in \tilde{\mathbf{Y}}$.

Now we like to have an estimate for the true process:

$$P_t(z) = y_t(z) \, / \, u_t(z)$$

Therefore we define the following set:
The set of unfalsified systems is the set with systems $\tilde{P} \in R\!I\!P$, that do not falsify the measured data and the noise bounds. So we consider the functions $\tilde{P}(z) = \tilde{y}(z)/\tilde{u}(z)$ for all $\tilde{u}(z) \in \tilde{\mathbf{U}}$ and $\tilde{y}(z) \in \tilde{\mathbf{Y}}$ (where we assume that $\tilde{u}(z) \ne 0$, for all $z \in \mathcal{C}$). This means that we deal with a persistently exciting input $u_t(z)$ and a sufficiently small input noise signal $\tilde{d}(z)$. Note that the true

process $P_t(z)$ is an element of the set $\tilde{\mathbf{P}}$.

Remark: From now on in this section we will consider all signals and functions for a specific frequency. Therefore we will use a simplified notation (i.e. $\tilde{u}$ in stead of $\tilde{u}(z)$, $\tilde{y}$ in stead of $\tilde{y}(z)$). Now the sets $\tilde{\mathbf{D}}$, $\tilde{\mathbf{E}}$, $\tilde{\mathbf{Y}}$ and $\tilde{\mathbf{U}}$ will be restricted to one disk in the complex plane for that specific frequency.

To derive a representation of the set $\tilde{\mathbf{P}}$ we will need an auxilary set $\tilde{\mathbf{X}}$ with signals $\tilde{x} = \tilde{u}^{-1}$ for all $\tilde{u} \in \tilde{\mathbf{U}}$.

Theorem 3.1:
Suppose $\tilde{\mathbf{U}}$ is the set of signals $\tilde{u}$ satisfying $\mid \tilde{u} - u \mid \leq \mid W_d \mid$, where we assume $\mid u \mid > \mid W_d \mid > 0$.
Then the set $\tilde{\mathbf{X}}$ with signals $\tilde{x} = \tilde{u}^{-1}$ for all $\tilde{u} \in \tilde{\mathbf{U}}$ is given by the disk:

$$\tilde{\mathbf{X}} = \left\{ \tilde{x} \mid \left| \tilde{x} - \frac{u^*}{uu^* - W_d W_d^*} \right| \leq \frac{\mid W_d \mid}{uu^* - W_d W_d^*} \right\} \qquad \square$$

The proof of this theorem is in appendix A1.

This means that the set $\tilde{\mathbf{X}}$ is represented by a disk in the complex plane, like the sets $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{Y}}$. For an easier notation we use:

$$x \stackrel{def}{=} \frac{u^*}{uu^* - W_d W_d^*} \quad \text{and we get} \quad \mid \tilde{x} - x \mid \leq \mid x \mid \frac{\mid W_d \mid}{\mid u \mid}$$

In figure 3.2 the sets $\tilde{\mathbf{U}}$, $\tilde{\mathbf{Y}}$ and $\tilde{\mathbf{X}}$ are shown in the complex plane for one frequency sample. As an example we can take d and e as white Gaussian noises and let the bound be given by the 3σ-bound (see section 2.4). One thousand realizations of these disturbances have been presented by points in figure 3.2. In that way the point density indicates the probability of the expected signal values. Note that the point density in the set $\tilde{\mathbf{X}}$ is not concentrated around the center x, but around a point, that is closer to the origin. This specific point is u^{-1}.
    So we now have the sets $\tilde{\mathbf{Y}}$ and $\tilde{\mathbf{X}}$ as

$$\tilde{\mathbf{Y}} = \left\{ \tilde{y} = y(1 + \alpha_y e^{j\phi}), \ 0 \leq \phi < 2\pi, \ 0 \leq \alpha_y \leq \frac{\mid W_e \mid}{\mid y \mid} \right\}$$

$$\tilde{\mathbf{X}} = \left\{ \tilde{x} = x(1 + \alpha_u e^{j\psi}), \ 0 \leq \psi < 2\pi, \ 0 \leq \alpha_u \leq \frac{\mid W_d \mid}{\mid u \mid} \right\}$$
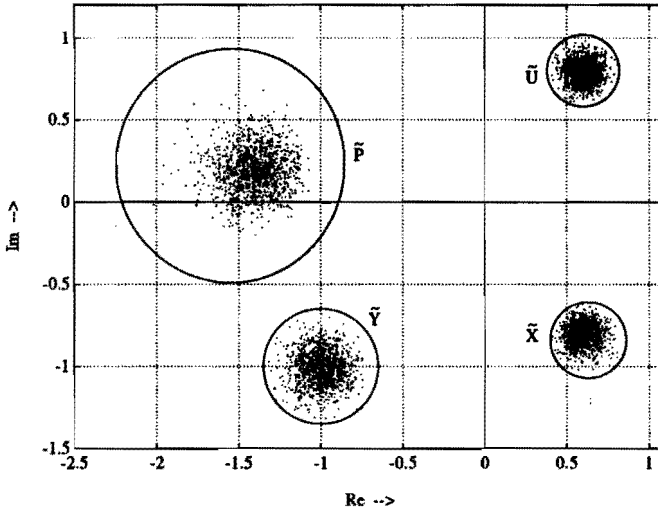
Figure 3.2: The sets $\tilde{U}$, $\tilde{Y}$, $\tilde{X}$ and $\tilde{P}$ in the complex plane.

We can derive a representation for the set $\tilde{P}$ with transfer functions $\tilde{P}$ by multiplying all elements $\tilde{y} \in \tilde{Y}$ with all elements $\tilde{x} \in \tilde{X}$. This results in

$$\tilde{P} = \{ \ \tilde{P} = yx(1 + \alpha_y e^{j\phi})(1 + \alpha_u e^{j\psi}),$$

$$0 \le \phi < 2\pi, \ 0 \le \psi < 2\pi, \ 0 \le \alpha_y \le \frac{|W_e|}{|y|}, \ 0 \le \alpha_u \le \frac{|W_d|}{|u|} \ \}$$

This is a region which typically is shaped like a bean, as will be analysed by means of figure 3.4. The next step will be to calculate a boundary function for the region.

First note that $\tilde{P}(\alpha_y, \alpha_u, \phi, \psi)$ for specific values of α_y, α_u, ϕ and ψ will only be a boundary point if $\tilde{y}(\alpha_y, \phi)$ and $\tilde{u}(\alpha_u, \psi)$ are also both boundary points. So we have to fix

$$\alpha_y = r \ \overset{def}{=} \ \frac{|W_e|}{|y|} \quad \text{and} \quad \alpha_u = s \ \overset{def}{=} \ \frac{|W_d|}{|u|}$$

and we obtain

$$\tilde{P}(r, s, \phi, \psi) = y\,x\,(1 + re^{j\phi})(1 + se^{j\psi}) = y\,x\,h(\phi, \psi)$$

so the problem reduces into finding the boundary function of the function $h(\phi, \psi) = (1 + re^{j\phi})(1 + se^{j\psi})$.

Theorem 3.2:

Let r and s be real constants with $0 < r < 1$ and $0 < s < 1$.
The boundary of the region with points $h(\phi, \psi) = (1 + re^{j\phi})(1 + se^{j\psi})$,
where $0 \le \phi < 2\pi$, $0 \le \psi < 2\pi$, is given by the function:

$$h_b(\psi) = 1 + s\,e^{j\psi} \;+\; \frac{r^2}{2}(\,e^{2j\psi} - 1\,) \;+$$

$$+\; r\,e^{j\psi}\sqrt{-r^2\sin^2\psi + 2s\cos\psi + 1 + s^2}$$

where $0 \le \psi < 2\pi$.   □

The proof of this theorem is given in appendix A2.

The boundary of region $\tilde{\mathbf{P}}$ becomes: $\tilde{P}_b(\psi) = y\,x\,h_b(\psi)$.
In figure 3.2 the region $\tilde{\mathbf{P}}$ is given. Also shown in the figure is the result of
the 1000 original random realizations of d and e, concentrated around y/u.

The boundary function $\tilde{P}_b(\psi)$ is not a nice function to work with. Therefore
we will try to derive a disk in the complex plane that encloses the region $\tilde{\mathbf{P}}$.
A simple circular bound (not the smallest) for the set $\tilde{\mathbf{P}}$ can be derived
very easily.
Define:

$$P_{c1} \;=\; y\,x \quad \text{and}$$

$$r_{c1} \;=\; |\,P_{c1}\,|\left(\frac{|W_d|}{|u|} + \frac{|W_e|}{|y|} + \frac{|W_dW_e|}{|uy|}\right)$$

$$\;=\; |\,P_{c1}\,|(\,r + s + rs\,)$$

Then the following will hold for all ϕ, ψ, α_y and α_u

$$|\,\tilde{P} - P_{c1}\,| \;=\; |y\,x\,(1 + \alpha_y e^{j\phi})(1 + \alpha_u e^{j\psi}) - y\,x\,| \;=$$

$$=\; |y\,x\,(\alpha_y e^{j\phi}) + \alpha_u e^{j\psi} + \alpha_y\alpha_u e^{j(\phi+\psi)})\,| \;\le$$

$$\le\; |\,P_{c1}\,|\left(\frac{|W_d|}{|u|} + \frac{|W_e|}{|y|} + \frac{|W_dW_e|}{|uy|}\right) \;=\; r_{c1}$$

So an enclosing set $\tilde{\mathbf{P}}_{c1} \supseteq \tilde{\mathbf{P}}$ with elements $\tilde{P}_{c1}$ can be given as:

$$\tilde{\mathbf{P}}_{c1} = \{\;\tilde{P}_{c1} \;=\; P_{c1} + \alpha_c\,e^{j\theta},\; 0 \le \theta < 2\pi,\; 0 \le \alpha_c \le r_{c1}\;\}$$

The set $\tilde{\mathbf{P}}_{c1}$ will enclose the set $\tilde{\mathbf{P}}$ very tightly as long as

$$\frac{|W_d|}{|u|} = s \ll 1 \quad \text{and} \quad \frac{|W_e|}{|y|} = r \ll 1$$

If these values increase then the enclosing will be less tight.

This 'simple' enclosing set is easy to calculate and will be satisfactory in most cases. If this enclosing set is too conservative, we will have to find the smallest circular enclosing set $\tilde{P}_{c2}$ for the set $\tilde{P}$. In other words, we like to find the smallest enclosing circle for the set $\tilde{P}$.

Before we derive this smallest enclosing circle we will consider a special point on the boundary, namely the point where $\text{Im}\big(h(\phi,\psi)\big)$ reaches its maximum value.

Because that point is a boundary point, we know from appendix A2 that the following will hold:

$$\frac{\partial h(\phi,\psi)}{\partial \phi} = \frac{\partial h(\phi,\psi)}{\partial \psi} C_0, \quad \text{with} \quad C_0 \in \mathbb{R}$$

In the case of a maximum of $\text{Im}\big(h(\phi,\psi)\big)$ the vectors in Fig.A.1 (Appendix A2) will be parallel to the real axis, and so:

$$\frac{\partial h(\phi,\psi)}{\partial \phi} \in \mathbb{R} \quad \text{and} \quad \frac{\partial h(\phi,\psi)}{\partial \psi} \in \mathbb{R}$$

leading to

$$\text{Im}\left(\frac{\partial h(\phi,\psi)}{\partial \phi}\right) = r \cos \phi + rs \cos(\phi+\psi) = 0 \quad \text{and}$$

$$\text{Im}\left(\frac{\partial h(\phi,\psi)}{\partial \psi}\right) = s \cos \psi + rs \cos(\phi+\psi) = 0$$

This results in the condition

$$r \cos \phi = s \cos \psi = -rs \cos(\phi+\psi)$$

Suppose (ϕ_0, ψ_0) satisfies the condition and let it be the solution corresponding to the maximum. Then, because of the symmetry around the real axis, $(-\phi_0, -\psi_0)$ is the solution corresponding to the minimum. Define:

$$c_0 = \text{Re}\big(h(\phi_0,\psi_0)\big) \in \mathbb{R} \quad \text{and} \quad \rho_0 = \text{Im}\big(h(\phi_0,\psi_0)\big) \in \mathbb{R}$$

then we can formulate the following theorem:

Theorem 3.3:
Let ϕ_0, ψ_0, c_0 and ρ_0 be defined as above.
Then

1. The circle, with centre point c_0 and radius ρ_0, is the smallest enclosing circle of $h(\phi, \psi)$.

2. An enclosing set $\tilde{\mathbf{P}}_{c2} \supseteq \tilde{\mathbf{P}}$ with elements $\tilde{P}_{c2}$ can be given as:

$$\mathbf{P}_{c2} = \{\ \tilde{P}_c\ =\ P_{c2}\ +\ \alpha_c\, e^{j\theta}\ ,\ 0 \le \theta < 2\pi,\ \ 0 \le \alpha_c \le r_{c2}\ \}$$

where $P_{c2} = yx\, c_0$ and $r_{c2} = |yx|\, \rho_0$.

$\square$

The proof of the first part of the theorem is in appendix A3. The second part is obvious with the result of the first part.
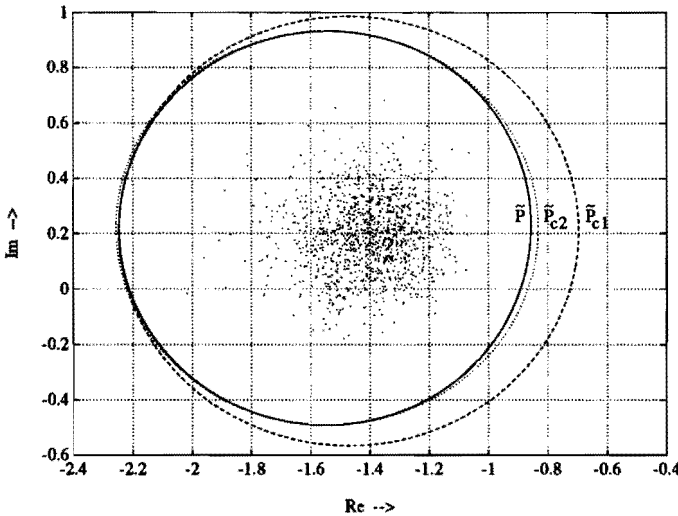
Figure 3.3: The sets $\tilde{\mathbf{P}}$, $\tilde{\mathbf{P}}_{c1}$ *and* $\tilde{\mathbf{P}}_{c2}$

In figure 3.3 the set $\tilde{\mathbf{P}}$ from figure 3.2 is given with its enclosing sets $\tilde{\mathbf{P}}_{c1}$ and $\tilde{\mathbf{P}}_{c2}$. It is clear that the results of the random realizations do not lead to a clustering around the centre of the region (P_{c1} or P_{c2}). Consequently in expectation, the centre of the region is not necessarily close to the real system transfer for the specific frequency.

In figure 3.4 the bounding function $P_b(\psi)$ is given for different values of r and s (where we fixed $y = 1$ and $u = 1$), together with the enclosing circles $P_{c1} + r_{c1}e^{j\theta}$ and $P_{c2} + r_{c2}e^{j\theta}$.
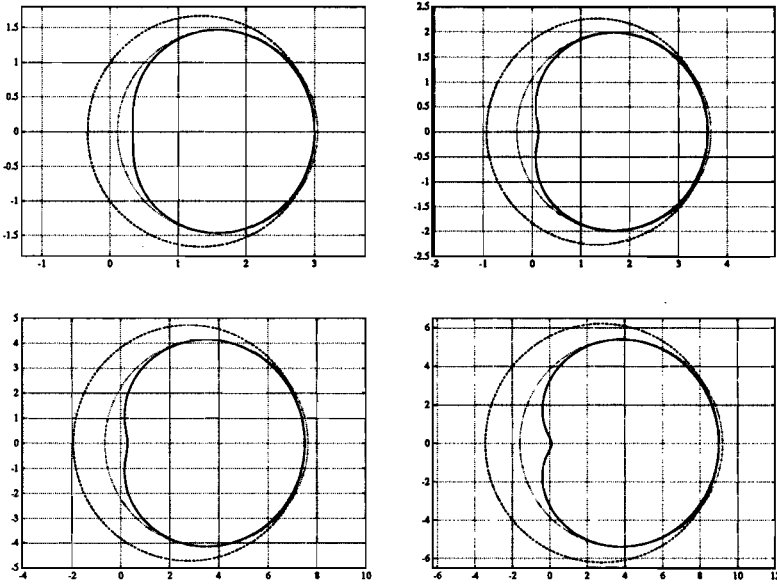
Figure 3.4: The sets $\tilde{\mathbf{P}}$ $(---)$, $\tilde{\mathbf{P}}_{c1}$ $(---)$ *and* $\tilde{\mathbf{P}}_{c2}$ $(\cdots)$
for various r and s :
    Left upper: $r = 0.5$, $s = 0.5$.
    Right upper: $r = 0.8$, $s = 0.5$.
    Left lower: $r = 0.5$, $s = 0.8$.
    Right lower: $r = 0.8$, $s = 0.8$.

3.3   Optimal H_∞-fitting

So far we only derived uncertainty regions for the true process. In this section we will look for an optimal nominal model that fits in the uncertainty regions, and we will consider the optimization of a parametric model in the sense that it minimizes the upper bound of the H_∞-norm of the weighted additive model error or the weighted multiplicative model error.

We define a model set **P** with the models $P(\theta, z)$, where $\theta \in \Theta$ is a vector with the model parameters. Of course we can choose many different types of models like ARMA, state space models, minimum polynomial models etc.

When we are interested in **additive model errors** we define a set of all candidate model errors

$$\tilde{\Delta}_a = \left\{ \tilde{\Delta}_a(z) \in R\!H_\infty \mid \tilde{\Delta}_a(z) = \tilde{P}(z) - P(z), \; \tilde{P} \in \tilde{\mathbf{P}}, \; P \in \mathbf{P} \right\}$$

Because $P_t(z) \in \tilde{\mathbf{P}}$ we note that the true model error $\Delta_{a,t}(z)$ will be in this set $\tilde{\Delta}_a$. In fact we would like to minimize the H_∞-norm of the true model

error ($\|\Delta_{a,t}(z)\|_\infty$) over all admissible models in the model set $\mathbf{P}$. However in the presence of input and output noise we can only give an upper bound:

$$\inf_{P \in \mathbf{P}} \|\Delta_{a,t}(z)\|_\infty \leq \inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \|\tilde{\Delta}_a(z)\|_\infty = \inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \|\tilde{P}(z) - P(z)\|_\infty$$

The McMillan degree of the model $P(\theta, z)$ is usually fixed, whereas the set $\tilde{\mathbf{P}}$ contains high order systems. So, in a sense, we solve a parametrized model approximation problem.

When we are interested in **multiplicative** model errors we define a set of all candidate model errors

$$\tilde{\mathbf{\Delta}}_m = \left\{ \tilde{\Delta}_m(z) \in RH_\infty \,|\, \tilde{\Delta}_m(z) = \frac{\tilde{P}(z) - P(z)}{P(z)}, \; \tilde{P} \in \tilde{\mathbf{P}}, \; P \in \mathbf{P} \right\}$$

Because $P_t(z) \in \tilde{\mathbf{P}}$ we note that the true model error $\Delta_{m,t}(z)$ will be in this set $\tilde{\mathbf{\Delta}}_m$. Again, in the presence of input and output noise we can give an upper bound:

$$\inf_{P \in \mathbf{P}} \|\Delta_{m,t}(z)\|_\infty \leq \inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \|\tilde{\Delta}_m(z)\|_\infty = \inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \| \frac{\tilde{P}(z) - P(z)}{P(z)} \|_\infty$$

To emphasize specific frequency ranges we can introduce a (stable and minimum phase) weighting filter $W_\Delta(z)$ and we can minimize the H_∞-norm of the weighted model error:

$$\inf_{P \in \mathbf{P}} \|W_\Delta(z)\Delta_{a,t}(z)\|_\infty \leq \inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \|W_\Delta(z)\tilde{\Delta}_a(z)\|_\infty =$$

$$= \inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \|W_\Delta(z)\big(\tilde{P}(z) - P(z)\big)\|_\infty$$

or

$$\inf_{P \in \mathbf{P}} \|W_\Delta(z)\Delta_{m,t}(z)\|_\infty \leq \inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \|W_\Delta(z)\tilde{\Delta}_m(z)\|_\infty =$$

$$= \inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \|W_\Delta(z)\big(\tilde{P}(z) P^{-1}(z) - 1 \big)\|_\infty$$

Now the problem of deducing upper bounds of the model error is reduced to a min-max problem, but with the use of the approximate set $\tilde{\mathbf{P}}_c$ (either $\tilde{\mathbf{P}}_{c1}$ or $\tilde{\mathbf{P}}_{c2}$) we can write:
For the **additive** model error:

$$\inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \|W_\Delta(z)\big(\tilde{P}(z) - P(z)\big)\|_\infty \leq$$

$$\inf_{P \in \mathbf{P}} \sup_{\tilde{P}_c \in \tilde{\mathbf{P}}_c} \|W_\Delta(z)\big(\tilde{P}_c(z) - P(z)\big)\|_\infty =$$

$$\inf_{P \in \mathbf{P}} \sup_{\tilde{P}_c \in \tilde{\mathbf{P}}_c} \| |W_\Delta(z)| \cdot |\tilde{P}_c(z) - P(z)| \|_\infty =$$

$$\inf_{P \in \mathbf{P}} \| |W_\Delta(z)| \big(|P_c(z) - P(z)| + r_c(z)\big) \|_\infty$$

For the **multiplicative** model error:

$$\inf_{P \in \mathbf{P}} \sup_{\tilde{P} \in \tilde{\mathbf{P}}} \|W_\Delta(z)\big(\tilde{P}(z) P^{-1}(z) - 1 \big)\|_\infty \leq$$

$$\inf_{P \in \mathbf{P}} \sup_{\tilde{P}_c \in \tilde{\mathbf{P}}_c} \|W_\Delta(z)\big(\tilde{P}_c(z) P^{-1}(z) - 1 \big)\|_\infty =$$

$$\inf_{P \in \mathbf{P}} \sup_{\tilde{P}_c \in \tilde{\mathbf{P}}_c} \| |W_\Delta(z)| \cdot | \tilde{P}_c(z) P^{-1}(z) - 1 | \|_\infty =$$

$$\inf_{P \in \mathbf{P}} \| |W_\Delta(z)| \big(|P_c(z) P^{-1}(z) - 1| + r_c(z) |P^{-1}(z)| \big) \|_\infty$$

Then we define the upper bound of the model error for a model with parameter vector θ and for a frequency $z \in \mathcal{C}$ as

$$\gamma_{a,max}(\theta, z) = |P_c(z) - P(\theta, z)| + r_c(z)$$
$$\gamma_{m,max}(\theta, z) = |P_c(z) P^{-1}(\theta, z) - 1| + r_c(z) |P^{-1}(\theta, z)|$$

Now the final problem we like to solve becomes:
For the **additive** model error:

$$\inf_{P \in \mathbf{P}} \|W_\Delta(z)\Delta_{a,t}(z)\|_\infty \leq$$

$$\inf_{\theta \in \Theta} \| |W_\Delta(z)| \big(|P_c(z) - P(\theta, z)| + r_c(z)\big) \|_\infty =$$

$$\inf_{\theta \in \Theta} \| W_\Delta(z) \gamma_{a,max}(\theta, z) \|_\infty$$

For the **multiplicative** model error:

$$\inf_{P \in \mathbf{P}} \|W_\Delta(z)\Delta_{m,t}(z)\|_\infty \leq$$

$$\inf_{\theta \in \Theta} \, \| \, |W_\Delta(z)| \left(\, |P_c(z) \, P^{-1}(\theta, z) - 1| \, + \, r_c(z) \, |P^{-1}(\theta, z)| \right) \, \|_\infty \, =$$

$$\inf_{\theta \in \Theta} \, \| \, W_\Delta(z) \, \gamma_{m,max}(\theta, z) \, \|_\infty$$

The problem turns out to be the minimization of the H_∞-norm of a function $W_\Delta(z) \gamma_{a,max}(\theta, z)$ or $W_\Delta(z) \gamma_{m,max}(\theta, z)$ over all admissible θ. Here we find the major drawback of using an H_∞-norm, namely that the cost-criteria $W_\Delta(z) \gamma_{a,max}(\theta, z)$ and $W_\Delta(z) \gamma_{m,max}(\theta, z)$ are not differentiable. This means that we can not directly use a gradient method to search for the minimum. We can solve the problem by using methods which do not need a gradient, e.g. simplex methods. The problem, however, with these methods is that convergence is not guaranteed if the initial value of θ is far from the optimal value. In that case we can use estimations from preliminary identifications as initial values.

3.4   Simulation example

In this section we present a simulation example.
A second order simulation model

$$P_t = \frac{z^2 - 1.1z + 0.24}{z^2 - 1.6z + 0.68}$$

in a configuration of figure 3.1 is excited by an input signal $u(k)$ and we measure the output $y(k)$. A Bode-plot of $P_t(e^{j\omega})$ is given in figure 3.5.

Input signal $u(k)$ is generated for 1024 samples and approximately has a flat spectrum $|u(z)| \approx 30$. Care has been taken that $-4 \leq u(k) \leq 4$ and that $u(k) = 0$ for $k = 1, .., 100$ and $k = 924, .., 1024$. (The length of the impulse response is less than 100 samples, see remarks in section 2.4). The control input and measured output signal are corrupted by additive white Gaussian noise $d(k)$ respectively $e(k)$. $W_d(z)$ and $W_e(z)$ are the 3σ-bounds in the frequency domain, and so they are constants. This results in the following values for the noise to signal ratios:

$$\frac{|W_d|}{|u(z)|} \leq 0.11 \quad \text{and} \quad \frac{|W_e|}{|y(z)|} \leq 0.16$$

We do a simulation experiment and obtain a data-set $\{ u(k) \, , \, y(k) \}$. For these noise-levels and for this choice of control signal $u(k)$ the errors due to the z-transform of $u(k)$ and $y(k)$ are negligible. The computations for
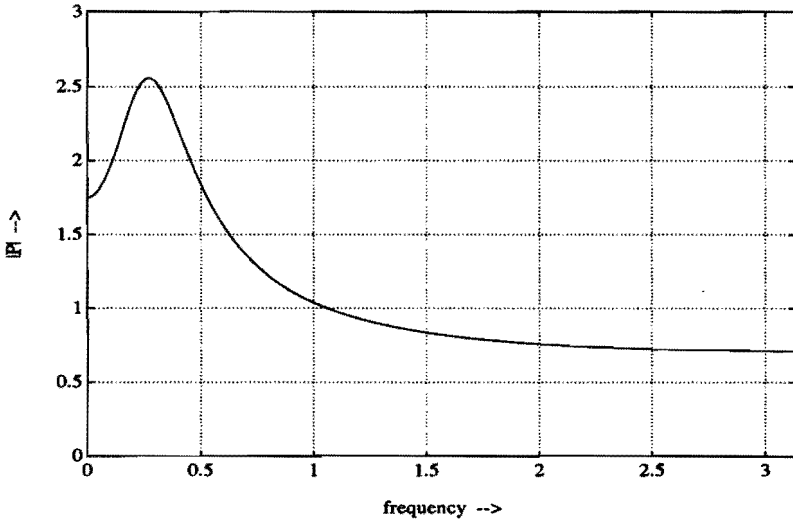
Figure 3.5: Bode plot of true process $P_t(e^{j\omega})$

the model error bounds will only be done on a limited number of frequency points z_i on the unit circle. We choose a frequency set $\Omega = \{z_1, z_2, ..z_N\}$ with $z = e^{j\omega_i}$ where $\omega_i = i\pi / 512$, $i = 1, ..512$.

For all frequencies z_i we calculate $P_C(z_i)$ and $r_C(z_i)$, using the simple circular bounds C_1, and we get the regions as in figure 3.6.

We will first consider an additive model error structure and so we must optimize the function

$$\gamma_{opt} = \inf_{\theta \in \Theta} \| W_\Delta(z) \gamma_{a,max}(\theta, z) \|_\infty$$

where the weighting filter is chosen $W_\Delta(z) = 1$.

In a first run we choose as a model set **P** all first order functions

$$P(z) = \frac{\theta_2 z + \theta_3}{z + \theta_1}$$

so $P_t(z)$ is not in the model set **P** .

We find an optimal $\theta_{opt} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} -0.275 \\ 0.895 \\ 0.757 \end{bmatrix}$ where $\gamma_{opt} = 0.936$.

In figure 3.7 the plots of the true process $P_t(z)$, the optimal model $P(z)$ and the region centre points $P_c(z)$ are given in the complex plane.
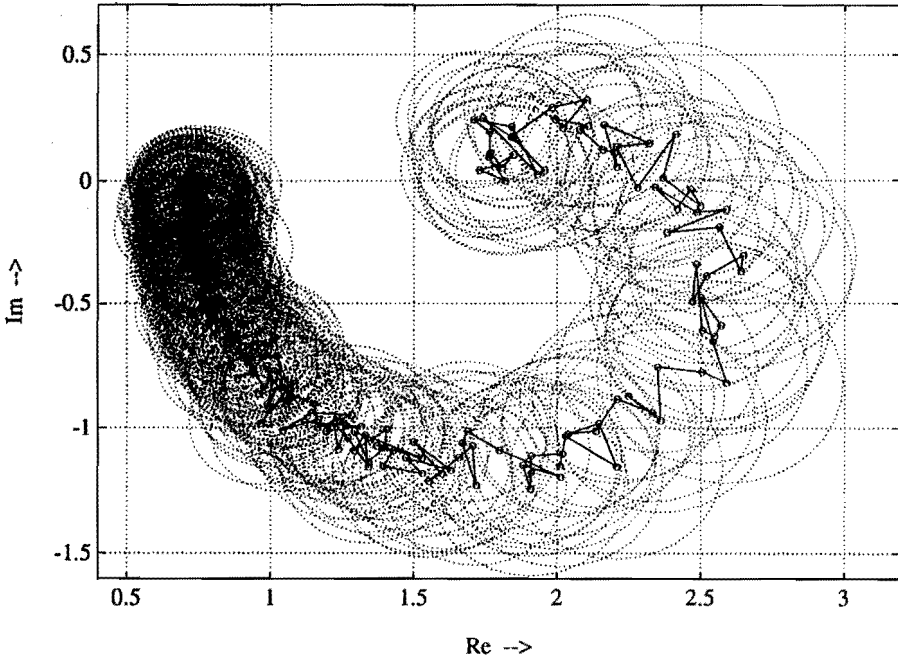
Figure 3.6: Uncertainty regions in the complex plane.

Now we define three functions:

$$\gamma_{a,max}(\theta, z) = |P_c(z) - P(\theta, z)| + r_c(z)$$

$$\gamma_{a,med}(\theta, z) = |P_c(z) - P(\theta, z)|$$

$$\gamma_{a,min}(\theta, z) = \max\left(0, |P_c(z) - P(\theta, z)| - r_c(z)\right)$$

The function $\gamma_{a,max}(\theta, z)$ gives an upper bound for the model error, the function $\gamma_{a,min}(\theta, z)$ is the minimum distance between $P(z)$ and $\tilde{P}(z)$ and so gives a lower bound. The function $\gamma_{a,med}(\theta, z)$ gives the distance of $P(z)$ to the centre of the region $\tilde{P}(z)$ and is centered between the upper and lower bounds. In figure 3.8 the functions $\gamma_{a,max}(\theta, z)$, $\gamma_{a,med}(\theta, z)$, $\gamma_{a,min}(\theta, z)$ and the true model error $|\Delta_{a,t}(z)|$ are plotted for the estimated model. In this example the lower bound $\gamma_{a,min}(\theta, z)$ is for nearly all frequencies larger than zero, which indicates, that the nominal model, that is found, cannot describe the system accurately. Note that these 'estimates $\gamma_{a,max}$, $\gamma_{a,med}$ and $\gamma_{a,min}$ can always be calculated and be used for defining a weighting filter W_Δ in a next iteration. If we want a better model for higher frequencies we choose a filter that will emphasize the error in the higher frequencies, so $W_\Delta(z)$ is large for higher frequencies, and will be small for the lower frequencies. Therefore we define a highpass filter as a weighting filter

$$W_\Delta(z) = \frac{z + 0.16}{z + 0.7}$$

For this choice of weighting filter we find $\theta_{opt} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} -0.402 \\ 1.240 \\ 0.190 \end{bmatrix}$

where $\gamma_{opt} = 1.031$.

In figure 3.7 and figure 3.8 give the results for the estimated model. Comparing the curves of the unweighted case in figure 3.8 with the weighted case in figure 3.9 we can see that the model error decreased very much for the higher frequencies, with the cost of a small increase at the lower frequencies.

If we want a better model for the lower frequencies we can choose a lowpass filter as a weighting filter

$$W_\Delta(z) = \frac{z + 0.7}{z + 0.16}$$

For this choice of weighting filter we find $\theta_{opt} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} -0.226 \\ 0.810 \\ 0.962 \end{bmatrix}$

where $\gamma_{opt} = 1.165$.

In figure 3.7 and figure 3.10 give the results for the estimated model. It is clear that the model did not improve very much for the lower frequencies, on the other hand it got much worse for the higher frequencies. We conclude that approximating the true process by a low order model is much more difficult for the lower frequencies than for the higher frequencies.

In a second run we choose as a model set **P** all second order functions

$$P(z) = \frac{\theta_3 z^2 + \theta_4 z + \theta_5}{z^2 + \theta_1 z + \theta_2}$$

so now $P_t(z)$ is in the model set **P**, and as a weighting filter $W_\Delta(z) = 1$.

We find an optimal $\theta_{opt} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \end{bmatrix} = \begin{bmatrix} -1.589 \\ 0.679 \\ 1.021 \\ -1.116 \\ 0.259 \end{bmatrix}$ where $\gamma_{opt} = 0.536$.

Figure 3.7 and figure 3.11 give the results for the estimated model. The lower bound $\gamma_{a,min}(\theta, z)$ in this example is exactly zero, which indicates, that the found nominal model might indeed describe the system exactly.

In a third run we choose as a model set **P** all third order functions

$$P(z) = \frac{\theta_4 z^3 + \theta_5 z^2 + \theta_6 z + \theta_7}{z^3 + \theta_1 z^2 + \theta_2 z + \theta_3}$$

so $P_t(z)$ is in the model set **P**, and as a weighting filter $W_\Delta(z) = 1$.

We find an optimal $\theta_{opt} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \end{bmatrix} = \begin{bmatrix} -1.610 \\ 0.703 \\ -0.0101 \\ 1.038 \\ -1.157 \\ 0.261 \\ 0.0063 \end{bmatrix}$ where $\gamma_{opt} = 0.535$.

Figure 3.7 and figure 3.12 give the results for the estimated model. Also in this case the lower bound $\gamma_{a,min}(\theta, z)$ in this example is exactly zero. However, the upper bound for the model error is not decreased much, so it looks as if a second order model will satisfy in this case (as could be expected). The parameters θ_3 and θ_7 are both nearly zero, resulting in a nearly pole-zero cancellation at $z = 0$.

Finally we consider a multiplicative model error structure and we optimize the function

$$\gamma_{opt} = \inf_{\theta \in \Theta} \| W_\Delta(z) \gamma_{m,max}(\theta, z) \|_\infty$$

where we choose a low pass filter as weighting filter

$$W_\Delta(z) = \frac{(z + 0.16)^2}{(z - 0.5)^2}$$

As a model set we choose a second order model

$$P(z) = \frac{\theta_3 z^2 + \theta_4 z + \theta_5}{z^2 + \theta_1 z + \theta_2}$$

and we find the optimal $\theta_{opt} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \end{bmatrix} = \begin{bmatrix} -1.629 \\ 0.702 \\ 0.985 \\ -1.122 \\ 0.266 \end{bmatrix}$ where

$\gamma_{opt} = 1.285$

In figure 3.7 and figure 3.13 give the results for the estimated model, where we defined the three functions:

$$\gamma_{m,max}(\theta, z) = |P_c(z) P^{-1}(\theta, z) - 1| + r_c(z)|P^{-1}(\theta, z)|$$

$$\gamma_{m,med}(\theta, z) = |P_c(z) P^{-1}(\theta, z) - 1|$$

$$\gamma_{m,min}(\theta, z) = \max\left(0, |P_c(z) P^{-1}(\theta, z) - 1| - r_c(z)|P^{-1}(\theta, z)|\right)$$

Again, the function $\gamma_{m,max}(\theta, z)$ gives an upper bound for the model error, the function $\gamma_{m,min}(\theta, z)$ gives a lower bound, and the function $\gamma_{m,med}(\theta, z)$ is centered between these bounds.

3.5 Conclusions

In this section we considered the identification of SISO-systems in terms of a minimum additive and multiplicative error bound. We calculated uncertainty regions and fitted the model in H_∞-norm sense. We have to make sure that the signal-to-noise ratio is sufficiently small, otherwise the model error bounds will become very large, or the proposed method will even fail.

Minimum, maximum and medium errors give an indication about the adaptation of the weighting filter $W_\Delta(z)$ in a next iteration step, and whether the model could represent the system.

With respect to the choice for an additive or a multiplicative model error structure, we like to make a remark: An additive model error can easily be transformed into a multiplicative model error, and vice versa, by choosing $\Delta_a(z) = P(z) \Delta_m(z)$, where $P(z)$ is the nominal model. So by choosing $W_{\Delta,m} = W_{\Delta,a} P(z)$ we get:

$$W_{\Delta,a}\Delta_a(z) = W_{\Delta,a}P(z)\Delta_m(z) = W_{\Delta,m}\Delta_m(z)$$

However, a minimization with a multiplicative error structure we will generally give another optimal model $P(z)$ than if we minimize with an additive model error structure.

The model error bounds are computed on a finite number of frequencies. However, true process P_t and the considered models P had all an impulse response with length smaller than 100 samples. We can use a simple interpolation technique to find a bound over all $z \in C$.
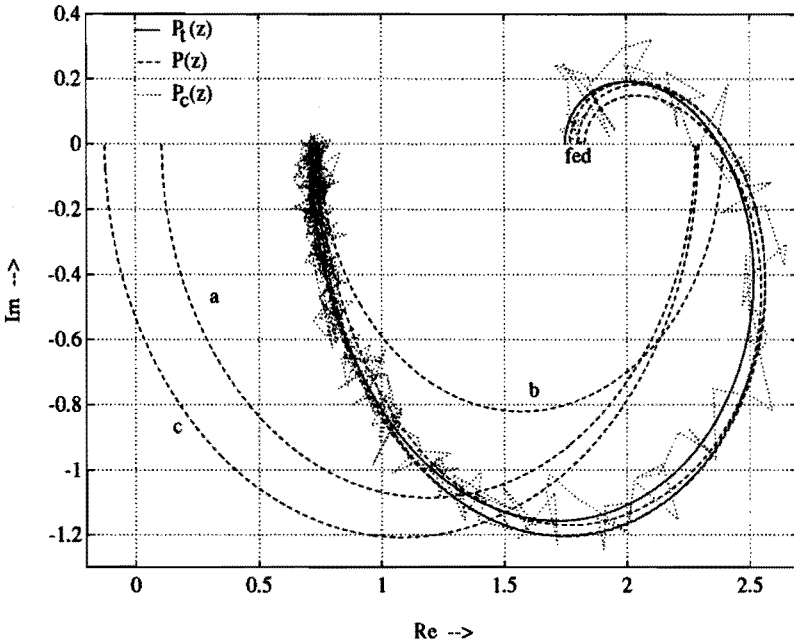
Figure 3.7: $P_t(z)$, $P(z)$ *and various models* $P_c(z)$ *in the complex plane:*

a : Additive model error:
  1st order model without weighting.

b : Additive model error:
  1st order model with weighting on higher frequencies.

c : Additive model error:
  1st order model with weighting on lower frequencies.

d : Additive model error:
  2nd order model without weighting.

e : Additive model error:
  3rd order model without weighting.

f : Multiplicative model error:
  2nd order model with weighting on lower frequencies.
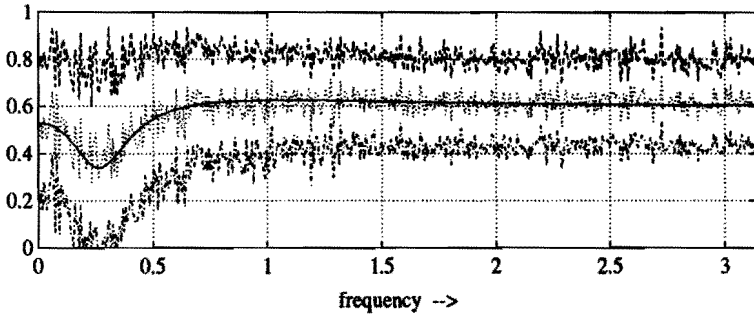
Figure 3.8: True additive model error with bounds (1st order, no weight)
Δ_t (——), γ_{max} (– – –), γ_{med} ($\cdots$), γ_{min} (– · –).
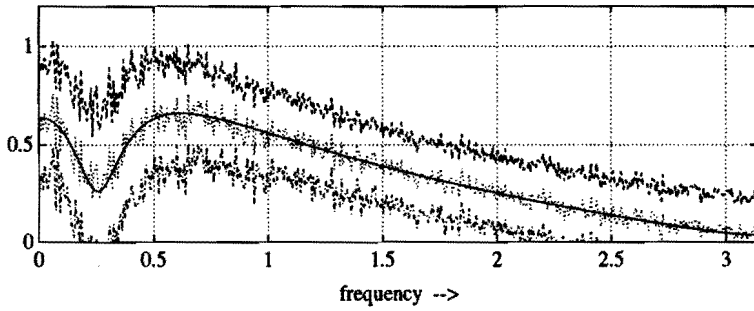
Figure 3.9: True additive model error with bounds (1st order, high freq. weight) Δ_t (——), γ_{max} (– – –), γ_{med} ($\cdots$), γ_{min} (– · –).
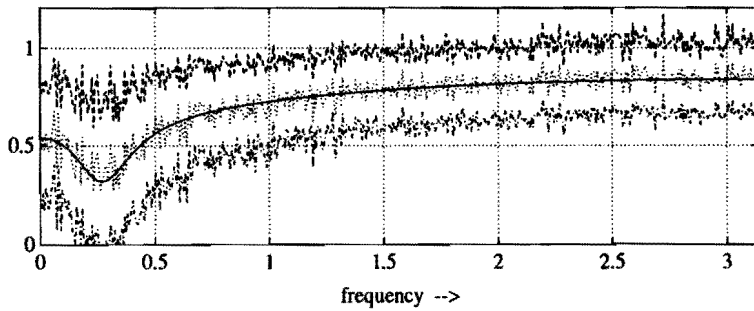
Figure 3.10: True additive model error with bounds (1st order low freq. weight) Δ_t (——) , γ_{max} (– – –) γ_{med} ($\cdots$) γ_{min} (– · –) .
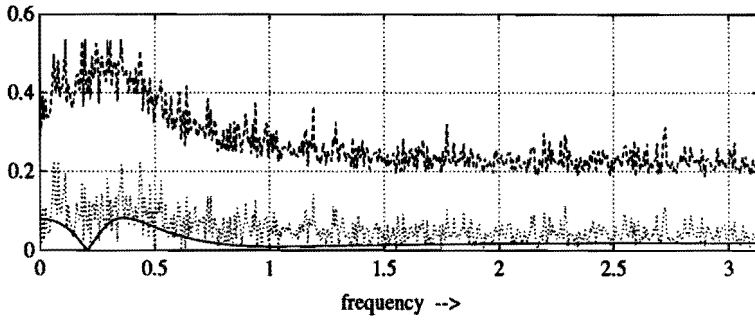
Figure 3.11: True additive model error with bounds (2nd order)
Δ_t (——), γ_{max} (———), and γ_{med} ($\cdots$).
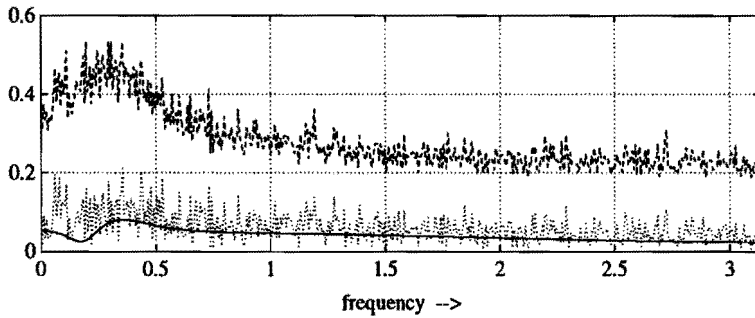
Figure 3.12: True additive model error with bounds (3rd order)
Δ_t (——), γ_{max} (———), and γ_{med} ($\cdots$).
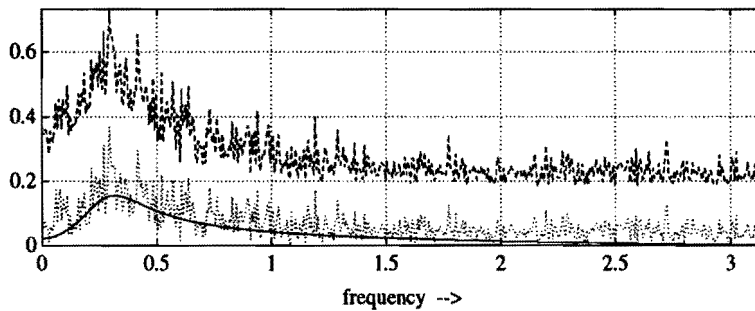
Figure 3.13: True multiplicative model error with bounds (2nd order)
Δ_t (——), γ_{max} (———), and γ_{med} ($\cdots$).

4

Multivariable systems

4.1 Introduction

In the previous chapter an identification procedure was presented for SISO-systems that resulted in a nominal model with bounds on the additive or multiplicative model error. In practice, however, most industrial processes are multivariable systems with various input and output signals. In this chapter we will discuss how to set up experiments for the identification of MIMO-systems.
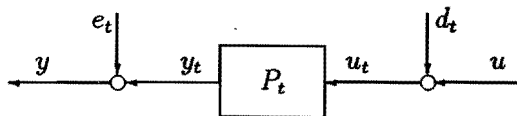
Figure 4.1: MIMO experimental set up

63

In figure 4.1, the process $P_t(z)$, with p inputs and q outputs, is excited by a true input signal $u_t(z)$, resulting in a true output signal $y_t(z)$. These signals $u_t(z)$ and $y_t(z)$ are corrupted by $d_t(z)$ and $e_t(z)$ respectively and measured in the input signal $u(z)$ and output signal $y(z)$, with

$$u_t(z) = u(z) + d_t(z)$$
$$y_t(z) = y(z) - e_t(z)$$
$$y_t(z) = P_t(z)u_t(z)$$

So $u(z)$, $u_t(z)$ and $d_t(z)$ are $p \times 1$ vectors, $y(z)$, $y_t(z)$ and $e_t(z)$ are $q \times 1$ vectors, $P_t(z)$ is a $q \times p$ matrix. The true process $P_t(z)$ is assumed to be linear time-invariant. The true noise signals $d_t(z)$ and $e_t(z)$ are unknown, but will be assumed to belong to a certain class. We consider all signals for all $z \in \mathcal{C}$.

4.2   Multiple Experiments

In the SISO-case we only did one experiment. We measured the input and output data and we applied the discrete Fourier transform to obtain the frequency signals $y(z)$ and $u(z)$. Together with the input and output disturbance signals $d_t(z)$ and $e_t(z)$ we derived:

$$P_t(z) \;=\; \frac{y_t(z)}{u_t(z)} \;=\; \frac{y(z) - e_t(z)}{u(z) + d_t(z)}$$

In the MIMO-case however the input signal $u_t(z)$ will be a $p \times 1$ vector, and so we can not just divide $y_t(z)$ by the signal $u_t(z)$ to obtain the true process $P_t(z)$. The input signal $u_t(z)$ contains some 'directional' information. By only exciting the system $P_t(z)$ with one signal vector $u_t(z)$, we will not be able to determine all dynamics of the system with all internal interactions by only observing the output. This can be understood by the following.

Suppose $P_t(z) \;=\; P_1(z) \;+\; P_2(z)$, where we choose this partitioning such that $u_t(z)$ in in the null-space of $P_2(z)$, so $P_2(z)u_t(z) = 0$. There is no response in $y_t(z)$ due to the part $P_2(z)$, so by only exciting the process by the signal $u_t(z)$ we will not be able to give an estimate of the magnitude of $P_2(z)$. So even if we find a bound for the model error of the part $P_1(z)$, we still do not have a bound for the total model error.

One way to solve the problem is to fix the structure of the model and assume that this structure is valid for $P_t(z)$. For most practical cases this can only be done in an approximate way or it is even impossible.

Another solution, we introduce here, is to do multiple experiments, where we change the 'direction' of the input signal $u_t(z)$ for every experiment. To excite all dynamics of the system $P_t(z)$ we will have to do a number of

experiments that is equal to the number of inputs:

We do p experiments where we measure the input and output signal. Now index the signal vectors, corresponding to the i-th experiment, as $u_i(z)$, $u_{ti}(z)$, $d_{ti}(z)$, $y_i(z)$, $y_{ti}(z)$ and $e_{ti}(z)$. Then we can construct the following matrices:

$$
\begin{array}{rcl}
U(z) & = & [\quad u_1(z) \quad u_2(z) \quad \ldots \quad u_p(z) \quad] \\
U_t(z) & = & [\quad u_{t1}(z) \quad u_{t2}(z) \quad \ldots \quad u_{tp}(z) \quad] \\
Y(z) & = & [\quad y_1(z) \quad y_2(z) \quad \ldots \quad y_p(z) \quad] \\
Y_t(z) & = & [\quad y_{t1}(z) \quad y_{t2}(z) \quad \ldots \quad y_{tp}(z) \quad] \\
D_t(z) & = & [\quad d_{t1}(z) \quad d_{t2}(z) \quad \ldots \quad d_{tp}(z) \quad] \\
E_t(z) & = & [\quad e_{t1}(z) \quad e_{t2}(z) \quad \ldots \quad e_{tp}(z) \quad]
\end{array}
$$

The matrices $U(z)$, $U_t(z)$ and $D_t(z)$ are $p \times p$ matrices, the matrices $Y(z)$, $Y_t(z)$ and $E_t(z)$ are $q \times p$ matrices. We have:

$$
\begin{array}{l}
U_t(z) = U(z) + D_t(z) \\
Y_t(z) = Y(z) - E_t(z) \\
Y_t(z) = P_t(z)U_t(z)
\end{array}
$$

We assume that the vectors $u_1(z)$, $u_2(z),\ldots,u_p(z)$ are linearly independent for all $z \in C$ and that the vectors $u_{t1}(z)$, $u_{t2}(z),\ldots,u_{tp}(z)$ are linearly independent for all $z \in C$, so that $U(z)$ and $U_t(z)$ are invertible for all $z \in C$. This results in:

$$
P_t(z) = Y_t(z)U_t(z)^{-1} = (Y(z) - E_t(z))(U(z) + D_t(z))^{-1}
$$

So by constructing a square matrix $U_t(z)$ with linear independent vectors we are sure that $U_t(z)$ excites in all 'directions', and so we will not miss any dynamics of $P_t(z)$.

Of course the noise matrices $D_t(z)$ and $E_t(z)$ are unknown. However, like in the SISO-case, we will consider them to belong to certain sets. These noise sets will be discussed in the next section.

Remark : As was already mentioned in the sections 2.4 and 2.5, the matrices $U(z)$, $Y(z)$, $U_t(z)$, $Y_t(z)$, $D_t(z)$ and $E_t(z)$ can be considered as matrices in $R H_\infty$ representing systems with a finite number of Markov parameters $(l \leq 2N)$.

4.3   Structured and unstructured noise sets

In this section the assumptions on the noise will be discussed for the multivariable case. The input noise matrix $D_t(z)$ is a $p \times p$ matrix, the output noise matrix $E_t(z)$ is a $q \times p$ matrix. Define:

$$F_t(z) \overset{def}{=} \begin{bmatrix} D_t(z) \\ E_t(z) \end{bmatrix} \quad , \text{ a } (p+q) \times p \text{ matrix.}$$

Assume that for the every experiment, the noise on the k-th input is bounded by a known filter $W_{dk}(z)$, and that the noise on the l-th output is bounded by a known filter $W_{el}(z)$ with

$$|d_{tk}(z)| \leq |W_{dk}(z)| \quad \text{and} \quad |e_{tl}(z)| \leq |W_{el}(z)| \quad \text{for } z \in \mathcal{C}$$

Then we can construct the diagonal matrices

$$W_d(z) = \begin{bmatrix} W_{d1}(z) & 0 & \cdots & 0 \\ 0 & W_{d2}(z) & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & W_{dp}(z) \end{bmatrix}$$

$$W_e(z) = \begin{bmatrix} W_{e1}(z) & 0 & \cdots & 0 \\ 0 & W_{e2}(z) & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & W_{eq}(z) \end{bmatrix}$$

and

$$W(z) = \begin{bmatrix} W_d(z) & 0 \\ 0 & W_e(z) \end{bmatrix}$$

Further we define

$$Q_{ut}(z) \overset{def}{=} W^{-1}(z) \begin{bmatrix} D_t(z) \\ E_t(z) \end{bmatrix}$$

(the subscript u denotes 'unstructured', as will be made clear soon. The subscript t denotes 'true').

This matrix $Q_{ut}(z)$ is a $(p+q) \times p$ matrix and is in fact the scaled version of the noise matrix $F_t(z)$. For $1 \leq i \leq (p+q)$ and $1 \leq j \leq p$ we have:

$$| \, [\, Q_{ut}(z)\,]_{ij} \, | \leq 1$$

The scaled noise matrix $Q_{ut}(z)$ is unknown, but we are sure that all entries of the matrix are on the unit disk. If we now introduce a set with matrices with all entries on the unit disk, then $Q_{ut}(z)$ will be in this set. There are two ways to describe such a set for the matrix $Q_{ut}(z)$, an unstructured or a structured way.

Unstructured set: If we want to describe $Q_{ut}(z)$ in an unstructured way, we like to find an ϵ_u such that

$$\| \, Q_{ut}(z) \, \| \; \leq \; \epsilon_u \; \text{ for all } z \in \mathcal{C}$$

This ϵ_u is very easy to derive. All entries of the matrix $Q_{ut}(z)$ are on the unit disk. The maximum singular value of a matrix filled with elements that are on the unit disk, is found if all elements are 1. In that case $\epsilon_u = \sqrt{(p+q)p}$. Using this ϵ_u, we can define the unstructured scaled noise set $\tilde{\mathbf{Q}}_u$ as follows:

$$\tilde{\mathbf{Q}}_u = \{ \, \tilde{Q}_u(z) \mid \| \, \tilde{Q}_u(z) \, \| \; \leq \; \sqrt{(p+q)p} \text{ for all } z \in \mathcal{C} \, \}$$

We define the set $\tilde{\mathbf{F}}_u$ as follows:

$$\tilde{\mathbf{F}}_u = \{ \, \tilde{F}_u(z) \; = \; W(z) \, \tilde{Q}_u(z) \text{ for all } \tilde{Q}_u(z) \in \tilde{\mathbf{Q}}_u \, \}$$

Note that $Q_{ut}(z) \in \tilde{\mathbf{Q}}_u$ and so $F_t(z) \in \tilde{\mathbf{F}}_u$.

Also note that the matrix $\tilde{Q}_u(z) \; = \; (\sqrt{(p+q)p}) \, I$ is in the set $\tilde{\mathbf{Q}}_u$. However, we are sure that $Q_{ut} \neq (\sqrt{(p+q)p}) \, I$, for all elements of the matrix Q_{ut} are on the unit disk. So, the description in an unstructured way, using only the largest singular value, generally will not be very tight. A better description can be given using a structured set.

Structured set: We can also describe $Q_{ut}(z)$ in a structured way.
First we construct a diagonal matrix $Q_{st}(z)$ where all elements of the matrix $Q_{ut}(z)$ are put on the diagonal of $Q_{st}(z)$:

$$[Q_{st}]_{(i-1)q+k,(i-1)q+k} \; = \; [Q_{ut}]_{ki} \text{ with } 1 \leq k \leq p+q \, , \; 1 \leq i \leq p$$

Now define

$$1_p = \underbrace{[1 \; 1 \ldots 1]}_{p \; times} \qquad , \qquad I_p \text{ is a } p \times p \; identity \text{ matrix}$$

$$1_q = \underbrace{[1 \; 1 \ldots 1]}_{q \; times} \qquad , \qquad I_q \text{ is a } q \times q \; identity \text{ matrix}$$

and

$$V_1 = \begin{bmatrix} (I_p \otimes 1_p) & 0 \\ 0 & (I_q \otimes 1_p) \end{bmatrix} = \begin{bmatrix} 1_p & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 1_p & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 1_p & \cdots & 0 \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 1_p \end{bmatrix}$$

$$V_2 = (I_p \otimes [\,1_p\ 1_q\,]^T) \;=\; \underbrace{[I_p\ I_p \cdots I_p\ I_p]}_{p\ times}^T$$

V_1 is a $(p+q) \times p(p+q)$ matrix, V_2 is a $p(p+q) \times p$ matrix.
With these definitions, the following holds

$$Q_{ut}(z) \;=\; V_1\, Q_{st}(z)\, V_2$$

All elements of the matrix $Q_{st}(z)$ are in the unit disk. Because the matrix is
diagonal, we find for the norm:

$$\| \, Q_{st}(z) \, \| \leq 1$$

So by constructing the diagonal matrix, we can define the structured scaled
noise set $\tilde{\mathbf{Q}}_s$ as follows:

$$\tilde{\mathbf{Q}}_s = \{\ \tilde{Q}_s(z) \text{ is diagonal and } \| \, \tilde{Q}_s(z) \, \| \ \leq\ 1 \ \text{ for all } z \in C\ \}$$

We define the set $\tilde{\mathbf{F}}_s$ as follows:

$$\tilde{\mathbf{F}}_s = \{\ \tilde{F}_s(z) \;=\; W(z)\, V_1\, \tilde{Q}_s(z)\, V_2 \text{ for all } \tilde{Q}_s(z)\ \in \tilde{\mathbf{Q}}_s$$

Note that $Q_{st}(z) \in \tilde{\mathbf{Q}}_s$ and so $F_t(z) \in \tilde{\mathbf{F}}_s$. However the structured set $\tilde{\mathbf{F}}_s$ is
smaller than the unstructured set $\tilde{\mathbf{F}}_u$, and so

$$\tilde{\mathbf{F}}_s \subset \tilde{\mathbf{F}}_u$$

If we use a structured description instead of an unstructured description for
the noise we generally will find smaller bounds on the model error.

Example: We will construct the matrices for the 2×2 case.
Suppose $D_t(z)$ and $E_t(z)$ are given as

$$D_t(z) = \begin{bmatrix} d_1(z) & d_2(z) \\ d_3(z) & d_4(z) \end{bmatrix} \quad , \quad E_t(z) = \begin{bmatrix} e_1(z) & e_2(z) \\ e_3(z) & e_4(z) \end{bmatrix}$$

For this $D_t(z)$ and $E_t(z)$ we can derive:

$$F_t(z) = \begin{bmatrix} d_1(z) & d_2(z) \\ d_3(z) & d_4(z) \\ e_1(z) & e_2(z) \\ e_3(z) & e_4(z) \end{bmatrix} \quad , \quad Q_{ut}(z) = \begin{bmatrix} W_{d1}^{-1}d_1 & W_{d2}^{-1}d_2 \\ W_{d1}^{-1}d_3 & W_{d2}^{-1}d_4 \\ W_{e1}^{-1}e_1 & W_{e2}^{-1}e_2 \\ W_{e1}^{-1}e_3 & W_{e2}^{-1}e_4 \end{bmatrix}$$

and

$$Q_{st}(z) =$$

$$= \begin{bmatrix} W_{d1}^{-1}d_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & W_{d2}^{-1}d_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & W_{d1}^{-1}d_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & W_{d2}^{-1}d_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & W_{e1}^{-1}e_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & W_{e2}^{-1}e_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & W_{e1}^{-1}e_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & W_{e2}^{-1}e_4 \end{bmatrix}$$

$$V_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad , \quad V_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

So far we assumed that the noise bounds remained the same for all experiments. In practical situations, however, it can happen that for every experiment the character of the noise changes, and therefore the noise bounds. For that case we will consider the structured set.

Assume that for the i-th experiment the noise on the k-th input ($[d_{ti}]_k = D_{tki}$) is bounded by a known filter $\bar{W}_{dki}(z)$ with $|D_{tki}| \leq |W_{dki}(z)|$, $z \in \mathcal{C}$, and assume that for the i-th experiment the noise on the l-th input ($[e_{ti}]_l = E_{tli}$) is bounded by a known filter $\bar{W}_{eli}(z)$ with $|E_{tli}| \leq |W_{eli}(z)|$, $z \in \mathcal{C}$. Now construct the diagonal matrices $\bar{W}_d$, $\bar{W}_e$, $\bar{Q}_t$ with

$$
\begin{array}{lll}
[\bar{W}_d]_{(i-1)q+k,(i-1)q+k} & = \bar{W}_{dki} & \text{with} \quad 1 \leq k \leq q \, , \, 1 \leq i \leq q \\
[\bar{W}_e]_{(j-1)q+l,(j-1)q+l} & = \bar{W}_{eli} & \text{with} \quad 1 \leq l \leq p \, , \, 1 \leq j \leq p \\
[\bar{Q}_{td}]_{(i-1)q+k,(i-1)q+k} & = \bar{W}_{dki}^{-1} D_{dki} & \text{with} \quad 1 \leq k \leq q \, , \, 1 \leq i \leq q \\
[\bar{Q}_{te}]_{(j-1)q+l,(j-1)q+l} & = \bar{W}_{eli}^{-1} E_{eli} & \text{with} \quad 1 \leq l \leq p \, , \, 1 \leq j \leq p
\end{array}
$$

and construct the diagonal matrices

$$\bar{W} = \begin{bmatrix} \bar{W}_d & 0 \\ 0 & \bar{W}_e \end{bmatrix} \qquad \bar{Q}_{st} = \begin{bmatrix} \bar{Q}_{td} & 0 \\ 0 & \bar{Q}_{te} \end{bmatrix}$$

$$F_t = V_1 \, \bar{W} \, \bar{Q}_{st} \, V_2$$

For this case we can define the structured scaled noise set $\bar{\mathbf{Q}}_{\pmb{s}}$ as follows:

$$\bar{\mathbf{Q}}_{\pmb{s}} = \{ \ \bar{Q}_{\pmb{s}}(z) \text{ is diagonal and } \| \ \bar{Q}_{\pmb{s}}(z) \ \| \ \leq \ 1 \ \text{ for all } z \in \mathcal{C} \ \}$$

We define the set $\bar{\mathbf{F}}_{\pmb{s}}$ as follows:

$$\bar{\mathbf{F}}_{\pmb{s}} = \{ \ \bar{F}_{\pmb{s}} \ = \ V_1 \, \bar{W} \, \bar{Q}_{\pmb{s}}(z) \, V_2 \text{ for all } \bar{Q}_{\pmb{s}}(z) \ \in \bar{\mathbf{Q}}_{\pmb{s}} \ \}$$

Note that $\bar{Q}_{\pmb{s}t}(z) \in \bar{\mathbf{Q}}_{\pmb{s}}$ and so $F_t(z) \in \bar{\mathbf{F}}_{\pmb{s}}$.

In the following chapters we will consider the unstructured set $\bar{\mathbf{Q}}_{\pmb{u}}$ and the structured set $\bar{\mathbf{Q}}_{\pmb{s}}$. In the case where we need $\bar{\mathbf{Q}}_{\pmb{s}}$ the formulae are easily adapted.

5

parametrization R-

5.1 Introduction

In chapter 3, a methodology was given to identify systems using an additive or multiplicative model error structure. In this chapter we will introduce a coprime factor model error structure that has advantages over additive and multiplicative model errors. The interpretation of coprime factor model errors, which is not so obvious, will be discussed in the sections 5.1 and 5.2. In section 5.3 we will introduce a simpler scheme and in section 5.4 we will show that this scheme is also applicable to all other model error structures.

In chapter 3 bounds were derived for a model error $\Delta_t(z)$, that was assumed to be stable. A stable model error $\Delta_t(z)$ in an additive or a multiplicative model error structure means that either the true plant $P_t(z)$ and the model $P(z)$ are both stable or that we can obtain an exact description of the unstable part of $P_t(z)$.

In the case where the number of unstable poles of the true plant is not exactly known, we cannot use an additive or multiplicative model error structure

any more. Then it will be necessary to use another model error description, like a reverse type structure or coprime factor structure. If we do not know the exact number of the true plant zeros that are outside the unit disk we will have to use a coprime factor plant description. We then consider the errors on the coprime factors of the process, which are required to be stable.
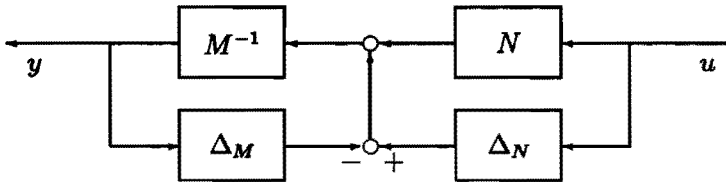
Figure 5.1: Left coprime factor model error

A simple example where the number of unstable poles of the true plant is not exactly known, is the case where we have the true plant

$$P_t(z) = \frac{az - b}{z - 1 + \epsilon}$$

where the sign of ϵ is unknown. If we take a model

$$P(z) = \frac{\hat{a}z - \hat{b}}{z - 1 + \hat{\epsilon}}$$

we can not guarantee that $P_t(z)$ has the same number of unstable poles as the model $P(z)$, and so we can not apply the results from robust control for additive or multiplicative model errors. In that case we can use the coprime factor plant description (Figure 5.1), where we assume the errors on the co-prime factors to be bounded.

For the example we can choose:

$$N_t(z) = \frac{az - b}{z - c_1} \quad , \quad M_t(z) = \frac{z - 1 + \epsilon}{z - c_1}$$

where we fix c_1 with $|c_1| < 1$ and as a model

$$N(z) = \frac{\hat{a}z - \hat{b}}{z - c_2} \quad , \quad M(z) = \frac{z - 1 + \hat{\epsilon}}{z - c_2}$$

where we fix c_2 with $|c_2| < 1$. It is clear that the factor model errors

$$\Delta_N(z) = N_t(z) - N(z) \quad , \quad \Delta_M(z) = M_t(z) - M(z)$$

are both stable because N_t, M_t, N and M are stable.

As should be clear from above the coprime factor model error is not only depending on the model $P(z)$ and the true process $P_t(z)$, but also on the choice of coprime factorization $P(z) = M(z)^{-1}N(z)$ and coprime factorization $P_t(z) = M_t(z)^{-1}N_t(z)$. In the example these factorizations are chosen by fixing c_1 and c_2. A different choice of these constants leads to a different coprime factor model error $[\, \Delta_N(z) \;\; \Delta_M(z) \,]$. For example, a scaling with a non-zero constant of the factors N_t, M_t, N and M will not change their coprimeness. The coprime factor model error however will be scaled with the same constant. In this way we can make the model error as small as we like. If we define a model error criterion $J(\Delta)$, for which $J(0) = 0$, we can make $J(\Delta)$ smaller than any $\eta > 0$, which is unrealistic in the light of e.g. the frequency behaviour of P and P_t.
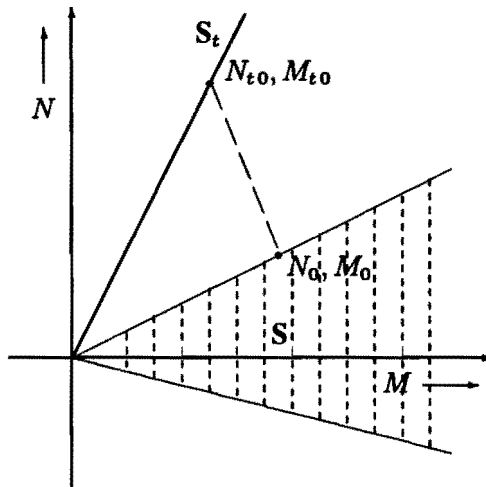
Figure 5.2: Coprime factors of true process and model

We can visualize this non-uniqueness problem with a simple example where we only consider constant real scalar valued systems. In figure 5.2 we show the coprime factors of the true process and all models in a model set. We put the M-factor on the horizontal axis and the N-factor on the vertical axis. A single line through the origin represent all coprime factorizations of one specific system, and the gradient of the line is equal to the value of that function. The vertical axis is excluded, because $M = 0$ is not allowed. As an example we have chosen the 'constant transfers' $P_t = 2$ and $P = \beta$ in a model set $\mathbf{P}$, where β is restricted to $-\frac{1}{4} \leq \beta \leq \frac{1}{2}$ (so P_t is not in the model set). The line $\mathbf{S}_t$ gives all possible coprime factor pairs (N_t, M_t) of the true process $P_t = N_t M_t^{-1}$. The dashed region $\mathbf{S}$ gives all possible coprime factor

pairs (N, M) for the models $P = NM^{-1}$ in the model set **P**. We choose a particular coprime factorization of P_t, denoted by (N_{t0}, M_{t0}), and we choose a particular coprime factorization for one choice of a model $P \in \textbf{P}$, denoted by (N_0, M_0). The coprime factor model error is now given by

$$\| [\, N_{t0} - N_0 \quad M_{t0} - M_0 \,] \| \; = \; \sqrt{(N_{t0} - N_0)^2 + (M_{t0} - M_0)^2}$$

and is equal to the Euclidean distance between the points (N_{t0}, M_{t0}) and (N_0, M_0) in the figure. Although $P_t \notin \textbf{P}$ we can make this distance and so the coprime factor model error as small as we like by shifting (N_{t0}, M_{t0}) and (N_0, M_0) towards the origin (the origin itself is excluded). In this way the coprime model error does not give any indication about how far away the model is away from the true process.

It is clear that we will have to fix at least one of the coprime factorizations, either of the model or of the true process. We will choose for fixing the coprime factorization of the model, because then we can restrict the structure and order of the nominal model coprime factors. This is important if we want to use the model coprime factors for robust control later on. We need some constraints on the choice of the model coprime factors in such a way that the model coprime factors become uniquely associated with a model in the model set **P**.

In order to achieve that we introduce a bijective mapping π from the model to the model coprime factors

$$\pi(P(z)) \; = \; [\, N(z) \quad M(z) \,]$$

Here, $N(z)$ and $M(z)$ are specific coprime factors in $R\!H_\infty$ such that $P(z) = M(z)^{-1} N(z)$. Therefore π is bijective and there holds

$$\pi^{-1} ([\, N(z) \quad M(z) \,]) \; = \; P(z)$$

In chapter 9 we will consider a specific choice for this mapping π. The choice of π will influence the optimization procedure to find the optimal model with minimum error bounds (see chapters 8 and 9).

With the choice of the mapping π we have chosen the coprime factorization of the model in a unique way. If we consider all models in the model set **P** we can define the set of coprime factor models:

$$\textbf{S} \; = \; \{ [\, N(z) \quad M(z) \,] \; = \; \pi(P(z)), \text{ where } P(z) \in \textbf{P} \}$$

In short: $\textbf{S} \; = \; \pi(\textbf{P})$.

Also for the true process $P_t(z)$, although it is unknown, we can fix a coprime factorization by a bijective mapping π_t from $R\!P$ to $R\!H_\infty \times R\!H_\infty$

$$[\ N_{to}(z)\ \ M_{to}(z)] \ = \ \pi_t(\ P_t(z)\ )$$

where $P_t(z) \ = \ M_{to}(z)^{-1} N_{to}(z)$. Usually this mapping π_t will not be the same as the mapping π. The mapping π is only defined on the model set $\mathbf{P}$, where as π_t will be defined on all the functions in $R\!P$.

If we consider the robustness criteria concerning coprime factor model errors (Vidyasagar [63] and McFarlane [43]) then there is no constraint on the choice of coprime factorization of the perturbed plant. The true plant will be in the set of perturbed plants, so a fixed coprime factorization for the process is not desirable. We will describe a way to find all coprime factorizations of the true process, starting from the mapping π_t. For that purpose we formulate the following lemma:

Lemma 5.1:
Let $P_t(z) \ = \ M_{to}(z)^{-1} N_{to}(z)$ be a left coprime factorization of $P_t(z)$ in $R\!H_\infty$.
Then
$P_t(z) \ = \ M_t(z)^{-1} N_t(z)$ will also be a left coprime factorization of $P_t(z)$ in $R\!H_\infty$ if and only if there exists a unimodular matrix function $A(z)$ in $R\!H_\infty$ such that

$$[\ N_t(z)\ \ M_t(z)\ ] \ = \ A(z)[\ N_{to}(z)\ \ M_{to}(z)\ ]$$

$\square$

The proof of this lemma is in Vidyasagar ([62], page 75).

Using Lemma 5.1, we can now define a set $\mathbf{S}_t$ with all possible coprime factorizations of the true process:

$$\mathbf{S}_t \ = \ \{[\ N_t(z)\ \ M_t(z)\ ] \ = \ A(z)\,\pi_t(\ P_t(z)\ ), \text{ where } A(z) \in \mathbf{A}\ \}$$

where the set $\mathbf{A}$ is defined as the set of unimodular functions in $R\!H_\infty$ with the same dimensions as M_t.

We now define optimality as follows: The coprime factor model error is given

$$\Delta(z) \ = \ [\ N_t(z) - N(z)\ \ M_t(z) - M(z)\ ]$$

Consider a model error criterion $J(\Delta)$. A model $P(z)$ is called optimal if the model error criterion $J(\Delta)$ is minimal, where we fix the coprime factorization of model $P(z)$ and where we have chosen the best possible coprime factorization of the true plant $P_t(z)$. The aim is to find the optimal model $P(z)$ with optimal coprime factors

$$[N(z) \ M(z)] = \pi(P(z))$$

so we like to solve the problem:

$$\inf_{[N \ M] \in S} \inf_{[N_t \ M_t] \in S_t} J([N_t(z) - N(z) \ M_t(z) - M(z)]) =$$

$$= \inf_{[N \ M] \in S} \inf_{A \in \mathbf{A}} J([A(z) N_{t0}(z) - N(z) \ A(z) M_{t0}(z) - M(z)])$$

where $[N_{t0}(z) \ M_{t0}(z)] = \pi_t(P_t(z))$. We will denote this optimization problem as the coprime factor approximation problem. We have to optimize over the model set **P** and the unimodular functions $A(z)$ for a fixed coprime factorization of the true plant $P_t(z)$.

In figure 5.3 we fixed the mapping π for the coprime factorization of the model. This is the solid line through the point (N_0, M_0). This line represents the set **S**.
Now we like to find the minimum distance between the element $(N_0(z) , M_0(z))$ in the set **S** and an element in the set S_t. We compute the fixed coprime factors of the true process $[N_{t0}(z) \ M_{t0}(z)] = \pi_t(P_t(z))$. Note that the set S_t is parametrized by multiplying any coprime factorization of $P_t(z)$ with a unimodular function $A(z)$, in our case a scalar α.

$$([N_t(z) \ M_t(z)]) = ([\alpha N_{t0}(z) \ \alpha M_{t0}(z)])$$

The set S_t is represented by the line through the origin and the point $(N_{t0}(z) , M_{t0}(z))$.
For all elements $(N(z) , M(z))$ in **S** we can compute this distance. In figure 5.3 the optimal model coprime factors in the set **S** will be the pair $(N_0(z) , M_0(z))$.
In the figure the element in S_t with minimum distance to $(N_0(z) , M_0(z))$. is denoted by $(N_t^*(z) , M_t^*(z))$.

Remark: Note the conceptual difference between the sets **S** and S_t: For the set **S** we fixed the coprime factorization and we vary the model $P(z)$ in **P**. For the set S_t we vary the coprime factorization for a fixed true process P_t.

Of course the next problem is that we do not know $P_t(z)$. What we do know is the set of unfalsified systems $\tilde{\mathbf{P}}$, that do not falsify the data. There holds $P_t(z) \in \tilde{\mathbf{P}}$. For every element $\tilde{P}(z) \in \tilde{\mathbf{P}}$ we can now compute a coprime factorization by the bijective mapping π_t. This mapping is defined on the true system, but will be extended to the set $\tilde{\mathbf{P}}$. Thus, for any $\tilde{P} \in \tilde{\mathbf{P}}$ we consider

$$[\tilde{N}_0(z) \ \tilde{M}_0(z)] = \pi_t(\tilde{P}(z))$$
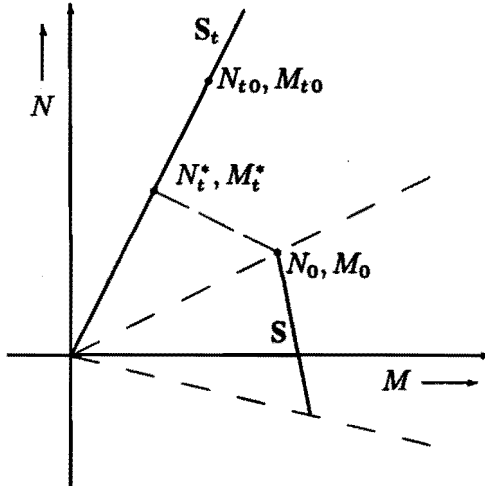
Figure 5.3: Coprime factors of true process and model

and so $\tilde{P}(z) = \tilde{M}_0(z)^{-1}\tilde{N}_0(z)$.

With this definition we can define a set $\tilde{\mathbf{S}}_0$ with a specific coprime factorization for all possible systems in the set of unfalsified systems $\tilde{P}(z) \in \tilde{\mathbf{P}}$:

$$\tilde{\mathbf{S}}_0 = \left\{ \left[\tilde{N}_0(z) \ \tilde{M}_0(z) \right] = \pi_t(\tilde{P}(z)), \text{ where } \tilde{P}(z) \in \tilde{\mathbf{P}} \right\}$$

In short: $\tilde{\mathbf{S}}_0 = \pi_t(\tilde{\mathbf{P}})$.

We can also define a set $\tilde{\mathbf{S}}$ with all possible coprime factorization for all possible systems in the set of unfalsified systems $\tilde{P}(z) \in \tilde{\mathbf{P}}$:

$$\tilde{\mathbf{S}} = \left\{ \left[\tilde{N}(z) \ \tilde{M}(z) \right] = A(z)\pi_t(\tilde{P}(z)), \ A \in \mathbf{A} \ , \ \tilde{P} \in \tilde{\mathbf{P}} \right\}$$

Note that

$$\pi_t(P_t) \in \tilde{\mathbf{S}}_0 \subset \tilde{\mathbf{S}}$$
$$\pi_t(P_t) \in \mathbf{S}_t \subset \tilde{\mathbf{S}}$$
$$\pi_t(P_t) = \tilde{\mathbf{S}}_0 \cap \mathbf{S}_t$$

Now we can redefine the notion of an optimal model: A model $P(z) \in \mathbf{P}$ will be optimal if the model error criterion $J(\Delta)$ is minimal, where we have chosen the best possible coprime factorization of the worst case system $\tilde{P}(z)$ in the set of unfalsified systems $\tilde{\mathbf{P}}$. So we have to solve the problem:

$$\inf_{[N\ M]\in S}\ \sup_{[\tilde{N}_0\ \tilde{M}_0]\in\tilde{S}_0}\ \inf_{A\in A}\ J\left(\left[A(z)\,\tilde{N}_0(z) - N(z)\quad A(z)\,\tilde{M}_0(z) - M(z)\right]\right)$$

We will denote this optimization problem as the min-max coprime factor approximation problem.
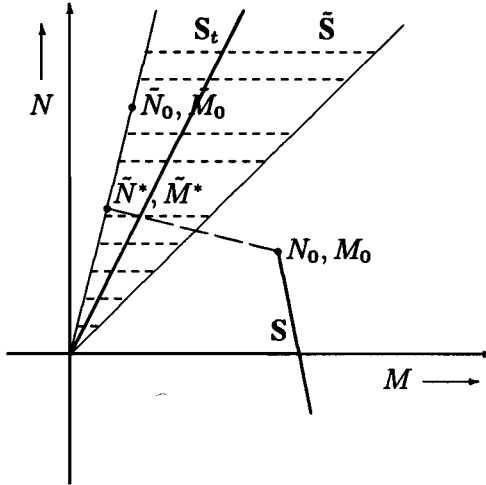
Figure 5.4: Coprime factors of true process and model

In order to illustrate the idea, we continue in figure 5.4 the scalar system example from figure 5.2 and figure 5.3. We consider a set $\tilde{P}$ of unfalsified systems with elements $\tilde{P} = \gamma$ where $1 \leq \gamma \leq 4$, and so $P_t \in \tilde{P}$. For all elements in $\tilde{P}$ we can compute all coprime factorizations and we obtain a set $\tilde{S}$. Now we consider the element $\tilde{P} = 4$ in $\tilde{P}$ and we compute the factors $\left[\tilde{N}_0\ \tilde{M}_0\right] = \pi_t(\tilde{P})$. This coprime factor pair can be scaled with a non-zero scalar α and we obtain $\left[\tilde{N}\ \tilde{M}\right] = \left[\alpha\,\tilde{N}_0\ \alpha\,\tilde{M}_0\right]$.
Consider the pair (N_0 , M_0), which is an element of the set S. We can measure the distance between that pair (N_0 , M_0) and the best possible coprime factorization of the worst case element $\tilde{P}$ in the set $\tilde{P}$. In the case of figure 5.4 the best chosen coprime factors of the worst case element of $\tilde{P}$ is obviously the pair ($\tilde{N}^*$, $\tilde{M}^*$). The distance between the pairs ($\tilde{N}^*$, $\tilde{M}^*$) and (N_0 , M_0) is always larger than or equal to the distance between the pair (N_0 , M_0) and the optimal choice of coprime factorization of the true process.
In the next section we will introduce a way to define the mapping π_t, using the so called R-parametrization from Hansen et al. ([27], [28]). We

will formulate the optimization problem, as mentioned above, in terms of this parametrization. This parametrization, and thus the mapping π_t, will depend on the model $P(z)$. This is allowed because during the optimization over $\tilde{P}$ the model $P(z)$ will be fixed.

In section 5.3 we will simplify the set up of section 5.2, and give a motivation for this simplification. Finally in section 5.4 it will be shown that many model error structures fit in the simplified scheme of section 5.3.

5.2   R-parametrization

We will assume that we have the knowledge of a controller $K(z)$ that stabilizes the true process $P_t(z)$ and which has a left coprime factorization

$$K(z) = M_C^{-1}(z)N_C(z)$$

This stabilizing controller $K(z)$ will be used during the analysis of the true process. It is not necessarilly the same controller as the controller $K_{exp}(z)$ that is used for the experiments.

We will now assume that the model $P(z)$ will also be stabilized by the controller $K(z)$, and so we restrict the model set **P** to those models $P(z)$ that are stabilized by the controller $K(z)$.

We now interconnect the coprime factors $N(z)$, $M(z)$, $N_C(z)$ and $M_C(z)$ with some function $R(z)$ as in the scheme of figure 5.5.
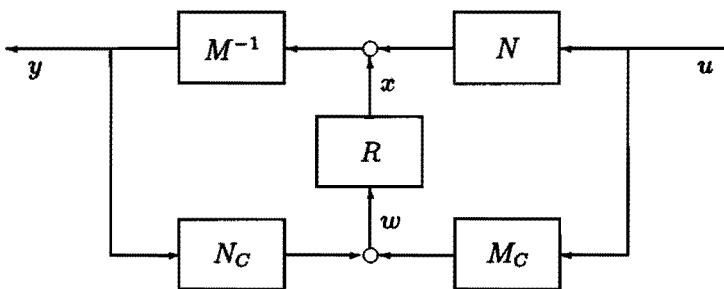
Figure 5.5: R-parametrization with left coprime factors

For this scheme we can formulate the following theorem

Theorem 5.1:
Let $K(z)$ be a controller with a left coprime factorization

$$K(z) = M_C^{-1}(z)N_C(z)$$

that stabilizes a model $P(z)$ with a left coprime factorization

$$P(z) = M^{-1}(z)N(z)$$

Then $K(z)$ stabilizes a process $P_0(z)$ if and only if there exists a function $R(z)$ in $R I\!H_\infty$ such that

$$P_0(z) = M_0^{-1}(z)N_0(z) \tag{5.1}$$

where

$$\begin{aligned} N_0(z) &= N(z) + R(z)M_C(z) \\ M_0(z) &= M(z) - R(z)N_C(z) \end{aligned} \tag{5.2}$$

Moreover, for every such $R(z) \in R I\!H_\infty$, the pair $N_0(z)$ and $M_0(z)$ will be left coprime and $R(z)$ uniquely defines the function P_0 via equation(5.1) and equation(5.2).

□

The proof of this Theorem is in Schrama[54]).

This way of parametrizing all plants stabilized by controller $K(z)$ was introduced by Hansen et al. ([27], [28]) and is called the R-parametrization.

The theorem claims that if the true process is stabilized by the controller $K(z)$, then there exists a stable function $R_t(z)$, that results in a left coprime factorization of the true process. So given any model $P(z) = M(z)^{-1}N(z)$, there exists a unique function $R_t(z) \in R I\!H_\infty$, such that the functions

$$\begin{aligned} N_{t0}(z) &\stackrel{def}{=} N(z) + R_t(z)M_C(z) \\ M_{t0}(z) &\stackrel{def}{=} M(z) - R_t(z)N_C(z) \end{aligned}$$

are coprime and

$$P_t(z) = M_{t0}^{-1}(z)N_{t0}(z)$$

For fixed model coprime factors $M(z)$ and $N(z)$ and a fixed controller $K(z)$ the function $R_t(z) \in R I\!H_\infty$ defines in this way a unique left coprime factorization of the true process $P_t(z)$ with coprime factors $M_{t0}(z)$ and $N_{t0}(z)$. In this way we have defined a bijective mapping π_t such that

$$[N_{t0}(z) \ M_{t0}(z)] = \pi_t(P_t(z))$$

Note that the mapping π_t depends on the choice of the model coprime factors $M(z)$ and $N(z)$ and the choice of the stabilizing controller $K(z)$, so

$$\pi_t\big(P_t(z)\big) \;=\; \pi_{t,(N,M,K)}\big(P_t(z)\big)$$

Because $R_t(z)$ is unique for fixed $M(z)$, $N(z)$, $M_C(z)$ and $N_C(z)$, there is also a bijective mapping ν_t from the true process to the stable function

$$\boxed{R_t(z) = \nu_t\,(P_t(z))}$$

Just like the mapping ν_t depends on the choice of the model coprime factors $M(z)$ and $N(z)$ and the choice of the coprime factors $M_C(z)$, $N_C(z)$ of the stabilizing controller.

$$\nu_t\big(P_t(z)\big) \;=\; \nu_{t,(N,M,N_C,M_C)}\big(P_t(z)\big)$$

We can parametrize all left coprime factorizations of the true process $P_t(z)$ by multiplying one pair of coprime factors with an arbitrary unimodular function $A(z)$ in $R I\!H_\infty$ and this yields the set S_t:

$$\mathbf{S}_t \;=\; \big\{\,[\,N_t(z)\;\;M_t(z)\,]\;=$$

$$= \; A(z)\,[\,N(z)+R_t(z)\,M_C(z)\;\;M(z)-R_t(z)\,N_C(z)\,],$$

$$\text{where } R_t(z) = \nu_t\,(\,P_t(z)\,)\,, \text{ and } A(z)\in\mathbf{A}\,\big\}$$

where $\mathbf{A}$ is the previously defined set of unimodular functions.

The coprime factor model error can now be expressed as:

$$[\,\Delta_N \quad \Delta_M\,] \;=\; [\,N_t - N \quad M_t - M\,] \;=$$

$$= \; [\,A\,(N+R_t\,M_C)-N \quad A\,(M-R_t\,N_C)-M\,] \;=$$

$$= \; [\,(A-I)\,N + A\,R_t M_C \quad (A-I)\,M - A\,R_t N_C\,]$$

where $R_t(z) = \nu_t\,(P_t(z))$ is a specific stable function and $A(z)$ is an arbitrary unimodular function in $R I\!H_\infty$.

Remark: By parametrizing the coprime factors of $P_t(z)$ in terms of the functions $R_t(z)$ and $A(z)$, we are sure that $M_t(z)$ and $N_t(z)$ are coprime, and so we compare all coprime factorizations of the true process with a specific coprime factorization of the model.

With the function $R_t(z)$ we can redefine the coprime factor approximation problem as:

$$\inf_{[N\ M]\in S}\ \inf_{[N_t\ M_t]\in S_t}\ J\big(\,[\ N_t - N\ \ M_t - M\ ]\,\big)\ =$$

$$\inf_{[N\ M]\in S}\ \inf_{A\in A}\ J\big(\,[\,(A-I)\,N + A\,R_t\,M_C\ \ (A-I)\,M - A\,R_t\,N_C\,]\,\big)$$

where $R_t(z) = \nu_t(\,P_t(z)\,)$. So we optimize over the model $P(z)$ and the unimodular function $A(z)$. One should keep in mind that the function $R_t(z)$ is itself a function of the model $P(z)$ and the initially chosen controller $K(z)$.

The true process $P_t(z)$ is unknown. However, we can derive a set of unfalsified systems $\tilde{P}$ with the use of the observed data $\{\,u\,,\,y\,\}$ and the noise bounding filters $W_d(z)$ and $W_e(z)$, as was already shown in chapter 3 for the SISO-case. If we now fix the model coprime factors $(\,N(z)\,,\,N(z)\,)$ and the controller coprime factors $(\,M_C(z)\,,\,N_C(z)\,)$, we can define the set

$$\tilde{R} = \Big\{\,\tilde{R}(z) \in RH_\infty \mid \tilde{R}(z) = \nu_t\big(\tilde{P}(z)\big),\ \ \tilde{P}(z) \in \tilde{P}\,\Big\}$$

In short: $\tilde{R} = \nu_t\big(\tilde{P}\big)$.

This results in a specific coprime factorization for every element $\tilde{P}(z) \in \tilde{P}$ using the mapping π_t as introduced in Theorem 5.1.

$$\big[\ \tilde{N}_0(z)\ \ \tilde{M}_0(z)\,\big] \ = \ \pi_t(\,\tilde{P}(z)\,)$$

with

$$\tilde{N}_0(z) \overset{def}{=} N(z) + \tilde{R}(z)\,M_C(z)$$
$$\tilde{M}_0(z) \overset{def}{=} M(z) - \tilde{R}(z)\,N_C(z)$$

and $\tilde{R}(z) = \nu_t\big(\tilde{P}(z)\big)$.

We can now redefine the min-max coprime factor approximation problem as follows:

$$\inf_{[N\ M]\in S}\ \sup_{[\tilde{N}_0\ \tilde{M}_0]\in S_o}\ \inf_{A\in A}\ J\Big(\,\big[\ A\,\tilde{N}_0 - N\ \ A\,\tilde{M}_0 - M\ \big]\,\Big)\ =$$

$$\inf_{[N\ M]\in S}\ \sup_{\tilde{R}\in \tilde{R}}\ \inf_{A\in A}\ J\Big(\,\big[\,(A-I)\,N + A\,\tilde{R}\,M_C\ \ (A-I)\,M - A\,\tilde{R}\,N_C\,\big]\,\Big)$$

In the following chapters we will show how to compute the set $\tilde{R}$. However, the functions $\tilde{R}(z)$ will not be given as real rational functions. We will
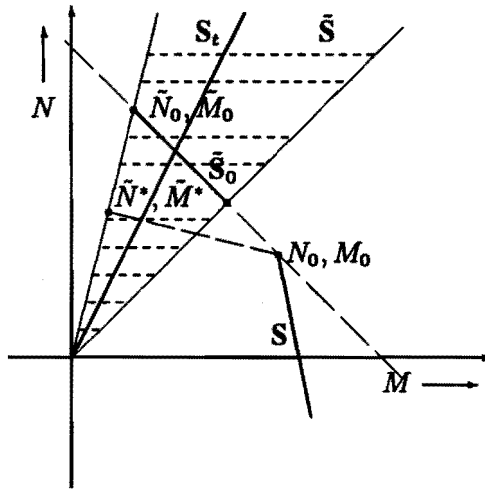
Figure 5.6: Coprime factors of true process and model

only be able to derive bounds for $\bar{R}(z)$ on specific frequencies z_i. This makes the min-max coprime factor optimization problem very difficult to solve, for we have to explicitly optimize over all unimodular functions $A(z)$ for all $\bar{R}(z)$.

In figure 5.6 we continue the scalar system example from figure 5.4. The controller $K = 1$ is stabilizing for the true process $P_t = 2$, because $(1 + P_t K)^{-1} = \frac{1}{3}$, and also for all models in the model set $\frac{2}{3} \leq (1 + P K)^{-1} \leq \frac{4}{3}$ with $-\frac{1}{4} \leq P \leq \frac{1}{2}$. We choose $M_C = 1$ and $N_C = 1$. We choose the pair (N_0 , M_0) in the model set **P** and consider the R-parametrization line with all pairs ($N + R M_C$, $M - R N_C$) for all $R \in \mathbb{R}$. In figure 5.6 this is the line through (N_0 , M_0) and ($\bar{N}_0$, $\bar{M}_0$). The set $\tilde{S}_0$ is the intersection of the R-parametrization line and the set $\tilde{S}$.

Again, the best chosen coprime factors of the worst case element of $\tilde{P}$ will be the pair ($\bar{N}^*$, $\bar{M}^*$), like in figure 5.4.

In this thesis we will not discuss the optimization over the unimodular function $A(z)$, because this optimization problem is too complex at this stage. We will choose $A(z)$ to be a fixed: $A(z) = I$. This choice can be motivated by the notion of consistency:
We will call our optimization method **consistent** if the chosen model error criterion $J(\Delta)$ will go to zero, when the model is chosen to be equal to the true process while at the same time the noise level is going to zero.
Thus consistency means that

$$\left. \begin{array}{r} P \to P_t \\ W_d, W_e \to 0 \end{array} \right\} \implies J(\Delta) \to 0$$

So for the min-max approximation problem:

$$\inf_{[N\ M]\in S}\ \sup_{\tilde{R}\in\tilde{\mathbf{R}}}\ J\left(\left[\ (A-I)\,N + A\,\tilde{R}\,M_C \quad (A-I)\,M - A\,\tilde{R}\,N_C\ \right]\right)$$

To be consistent we should necessarily have that $J(0) = 0$, which is typically the case if J is a norm.

We observe that $J(0) = 0$ and study what will happen if $P(\theta, z) \to P_t(z)$ and $W_d, W_e \to 0$.

We will assume that the true process $P_t(z)$ is in the model set $\mathbf{P}$ and we choose the model $P(\theta, z) = P_t(z)$. We parametrize the true process coprime factors:

$$\begin{aligned}
N_{t0}(z) &= N(z) + R_t(z)\,M_C(z) \\
M_{t0}(z) &= M(z) - R_t(z)\,N_C(z)
\end{aligned}$$

This gives the true process coprime factors for $R_t(z) = 0$. During an identification we will not be able to find $R_t(z) = 0$, because of the disturbances. We will only be able to derive the set $\tilde{\mathbf{R}}$ which will contain $R_t(z)$. As we will show in chapters 6 and 7, we can derive an H_∞ norm bound on all the elements of $\tilde{\mathbf{R}}$:

$$\|\,\tilde{R}(z)\,\|_\infty \leq \epsilon$$

This ϵ is only depending on the noise bounding filters $W_d(z)$ and $W_e(z)$. If we approach the noiseless case, so if $W_d(z), W_e(z) \to 0$ then we find $\epsilon \to \|\,R_t(z)\,\|_\infty$.

If $P(\theta, z) = P_t(z)$, then $R_t(z) = 0$ and we find that $W_d(z), W_e(z) \to 0$ leads to $\epsilon \to 0$.

For $P(\theta, z) = P_t(z)$ and $\epsilon \to 0$ the model error bound $J(\Delta)$ becomes

$$J(\Delta) = \lim_{\epsilon \to 0}\ \sup_{\|\tilde{R}\|\leq\epsilon}\ J([\ \tilde{N} - N \quad \tilde{M} - M\ ]) =$$

$$= \lim_{\epsilon \to 0}\ \sup_{\|\tilde{R}\|\leq\epsilon}\ J([\ (A-I)\,N + A\,\tilde{R}\,M_C \quad (A-I)\,M - A\,\tilde{R}\,N_C]) =$$

$$= J([\ (A-I)\,N \quad (A-I)\,M])$$

The choice $(A-I) = 0$ will make this model error Δ equal to zero, and thus $J(\Delta) = 0$.

So, if $J(0) = 0$ and $A(z) = I$ we find

$$\left. \begin{array}{l} P \rightarrow P_t \\ W_d, W_e \rightarrow 0 \end{array} \right\} \Longrightarrow J(\Delta) \rightarrow 0$$

which yields consistency of the optimization method.

In this thesis we therefore will only consider the optimization problem with fixed unimodular function

$$A(z) = I$$

5.3   Δ-parametrization

In this section we show that for a fixed unimodular function $A(z) = I$ the optimization problem will become easier.

As was mentioned in the previous section, we assume that we have the knowledge of a controller $K(z)$ that stabilizes the true process $P_t(z)$. Now we choose our model $P(z) = M^{-1}(z)N(z)$ such that it is also stabilized by this controller $K(z)$. We will denote the stable function $R_t(z)$ by the stable function $\Delta_t(z)$. The reason to change notation is the following: By fixing $A(z) = I$, the function $R_t(z)$ will behave as a model error, as will become clear in the next section. Therefore we introduce the notation $\Delta_t(z) = R_t(z)$. We now can give the interconnected scheme of figure 5.7:
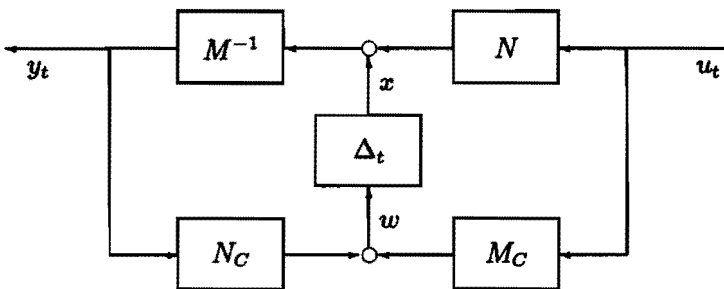
Figure 5.7: Δ-parametrization with left coprime factors

In this way we parametrized the true plant $N_t(z)$ and $M_t(z)$, with the left coprime factors of the model $N(z)$ and $M(z)$, the left coprime factors of the controller $N_C(z)$ and $M_C(z)$, and a stable function $\Delta_t(z)$:

$$P_t(z) = (M(z) - N_C(z)\Delta_t(z))^{-1}(N(z) + M_C(z)\Delta_t(z))$$

For this model $P(z)$ we can give an expression for the coprime factor model error:

$$[\; \Delta_N(z) \quad \Delta_M(z)\;] =$$

$$= [N_t(z) - N(z) \quad M_t(z) - M(z)\,] =$$

$$= [(N(z) + \Delta_t(z)M_C(z)) - N(z) \quad (M(z) - \Delta_t(z)N_C(z)) - M(z)] =$$

$$= [\; \Delta_t(z)\,M_C(z) \quad -\Delta_t(z)\,N_C(z)\;] =$$

$$= \Delta_t(z)[\; M_C(z) \quad -N_C(z)\,] \tag{5.3}$$

and so the coprime factor model errors can be expressed in terms of an unknown stable function $\Delta_t(z)$ multiplied by the known controller coprime factors $[\; M_C(z) \quad -N_C(z)\,]$. In the chapters 6 and 7 we will show how to find an estimate of this function $\Delta_t(z)$ and how to derive bounds on the model error.

So far we showed that coprime factor model errors can be discribed by the left coprime factor structure of figure 5.7. In the following section we will show that also other model error structures can be put in the configuration of figure 5.7. In the same way as we derived the scheme of figure 5.7 with left coprime factors we can give a scheme with right coprime factors (see figure 5.8), and analoguous results will hold.
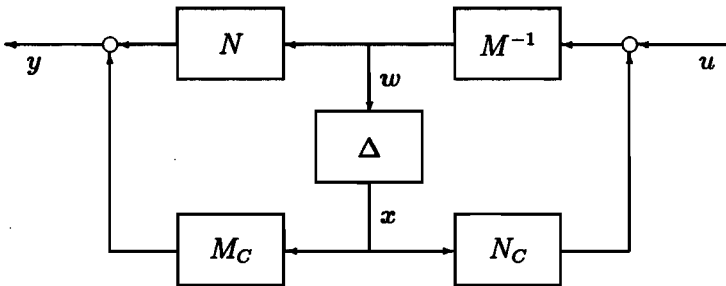
Figure 5.8: Δ-parametrization with right coprime factors

5.4 Model error structures

In this section we will show that many model error structures can be described by the left coprime factor structure of figure 5.7 and the right coprime factor structure of figure 5.8.

First we will consider the setup of figure 5.7. We have a model with a left coprime factorization $P(z) = M(z)^{-1}N(z)$ and a controller with a left coprime factorization $K(z) = M_C(z)^{-1}N_C(z)$. If this controller $K(z)$ stabilizes the model as well as the true process, then by Theorem 5.1 we can write the true process as: $P_t(z) = (M(z) - N_C(z)\Delta_t(z))^{-1}(N(z) + M_C(z)\Delta_t(z))$, where $\Delta_t(z)$ is stable. Many kinds of model errors can be described with this setup:

Additive model error (see figure 1.3):
We assume $P(z)$ and $P_t(z)$ to be stable so that $K(z) = 0$ is a stabilizing controller for both systems.
By choosing

$$ N = P \;, \quad M = I \;, \quad N_C = 0 \;, \quad M_C = I $$

we find $P_t(z) = P(z) + \Delta(z)$, which is the additive model error structure. The model error $\Delta(z) = P_t(z) - P(z)$ is in $R\!I\!H_\infty$.

Input multiplicative model error (see figure 1.4):
We assume $P(z)$, $P(z)^{-1}$, $P_t(z)$ and $P_t(z)^{-1}$ to be square and stable. Then $K(z) = 0$ is a stabilizing controller for both systems.
We now choose

$$ N = I \;, \quad M = P^{-1} \;, \quad N_C = 0 \;, \quad M_C = I $$

and we find $P_t(z) = P(z)\big(I + \Delta(z)\big)$, which is the multiplicative model error structure. The model error $\Delta(z) = P(z)^{-1}(P_t(z) - P(z))$ is in $R\!I\!H_\infty$.

Reverse additive model error (see figure 1.6):
We assume $P(z)^{-1}$ and $P_t(z)^{-1}$ to be square and stable. Then $K(z) = \infty$ (see remark 1 at the end of this section) is a stabilizing controller for both systems.
We choose

$$ N = I \;, \quad M = P^{-1} \;, \quad N_C = I \;, \quad M_C = 0 $$

and we find $P_t(z) = P(z) \left(I + \Delta(z) P(z) \right)^{-1}$, which is the reverse additive model error structure. The model error $\Delta(z) = P_t(z)^{-1} - P(z)^{-1}$ is in $R\!I\!H_\infty$.

Output reverse multiplicative model error (see figure 1.8):
We assume $P(z)$, $P(z)^{-1}$, $P_t(z)$ and $P_t(z)^{-1}$ to be stable. Then $K(z) = \infty$ is a stabilizing controller for both systems. We choose

$$N = P \,, \quad M = I \,, \quad N_C = I \,, \quad M_C = 0$$

and we find $P_t(z) = \left(I + \Delta(z) \right) P(z)$, which is the output reverse model error. The model error $\Delta(z) = (P_t(z) - P(z)) P_t(z)^{-1}$ is in $R\!I\!H_\infty$.

Left coprime factor model error (see figure 1.9):
We assume $P(z)$ and $P_t(z)$ to be stabilized by $K(z)$, and we choose

$$\Delta_N = M_C \Delta \,, \quad \Delta_M = -N_C \Delta$$

We find $P_t(z) = \left(M(z) + \Delta_M(z) \right)^{-1} \left(N(z) + \Delta_N(z) \right)$, which is the left coprime factor model error structure. The model error

$$[\, \Delta_N(z) \;\; \Delta_M(z) \,] = (MP_t - N)(M_c + N_cP_t)^{-1} [\, M_C(z) \;\; -N_C(z) \,]$$

is in $R\!I\!H_\infty$.

Now consider the setup of figure 5.8 where we have a model with a right coprime factorization $P(z) = N(z)M(z)^{-1}$ and a controller with a right coprime factorization $K(z) = N_C(z)M_C^{-1}$. If this controller $K(z)$ stabilizes the model as well as the true process, then we can write the true process as:

$$P_t(z) = (\, N(z) + \Delta_t(z)) M_C(z)\,)(\, M(z) - \Delta_t(z) N_C(z)\,)^{-1}$$

where $\Delta_t(z)$ is stable. Also with this setup many kinds of model errors can be described:

Additive model error (see figure 1.3):
We assume $P(z)$ and $P_t(z)$ to be stable. Then $K(z) = 0$ is a stabilizing controller for both systems.
By choosing

$$N = P \,, \quad M = I \,, \quad N_C = 0 \,, \quad M_C = I$$

we find $P_t(z) = P(z) + \Delta(z)$, which is the additive model error structure. The model error $\Delta(z) = (P_t(z) - P(z))$ is in $R\!I\!H_\infty$.

Output multiplicative model error (see figure 1.5):
We assume $P(z)$, $P(z)^{-1}$, $P_t(z)$ and $P_t(z)^{-1}$ to be square and stable. Then $K(z) = 0$ is a stabilizing controller for both systems. We choose

$$N = I \,, \quad M = P^{-1} \,, \quad N_C = 0 \,, \quad M_C = I$$

and we find $P_t(z) = \big(I + \Delta(z) \big) P(z)$, the multiplicative model error structure. The model error $\Delta(z) = (P_t(z) - P(z)) P(z)^{-1}$ is in RH_∞.

Reverse additive model error (see figure 1.6):
We assume $P(z)^{-1}$ and $P_t(z)^{-1}$ to be square and stable. Then $K(z) = \infty$ is a stabilizing controller for both systems.
We choose

$$N = I \,, \quad M = P^{-1} \,, \quad N_C = I \,, \quad M_C = 0$$

and we find $P_t(z) = P(z) \big(I + \Delta(z) P(z) \big)$, which is the reverse additive model error structure. The model error $\Delta(z) = P_t(z)^{-1}(P_t(z) - P(z))$ is in RH_∞.

Input reverse multiplicative model error (see figure 1.7):
We assume $P(z)$, $P(z)^{-1}$, $P_t(z)$ and $P_t(z)^{-1}$ to be square and stable. Then $K(z) = \infty$ is a stabilizing controller for both systems.
By choosing

$$N = P \,, \quad M = I \,, \quad N_C = I \,, \quad M_C = 0$$

we find $P_t(z) = P(z) \big(I + \Delta(z) \big)^{-1}$, which is the input reverse multiplicative model error structure. The model error $\Delta(z) = P_t(z)^{-1}(P_t(z) - P(z))$ is in RH_∞.

Right coprime factor model error (see figure 1.10):
We assume $P(z)$ and $P_t(z)$ to be stabilized by $K(z)$ and we choose

$$\Delta_N = \Delta M_C \,, \quad \Delta_M = \Delta N_C$$

We find $P_t(z) = \big(N(z) + \Delta_N(z) \big)\big(M(z) + \Delta_M(z) \big)^{-1}$, the right coprime factor model error structure. The model error

$$\begin{bmatrix} \Delta_N(z) \\ \Delta_M(z) \end{bmatrix} = \begin{bmatrix} M_C(z) \\ -N_C(z) \end{bmatrix} (M_c + P_t N_c)^{-1}(N - P_t M)$$

is in RH_∞.

In table 5.1 all considered model error structures are summarized with the corresponding coprime factor choices $N(z)$, $M(z)$, $N_C(z)$ and $M_C(z)$.

P		K		left/right	model error structure
N	M	N_C	M_C	coprime	
$P(z)$	I	0	I	left/right	additive
I	$P^{-1}(z)$	0	I	left	input multiplicative
I	$P^{-1}(z)$	0	I	right	output multiplicative
$N(z)$	$M(z)$	0	I	left/right	coprime factor ($\Delta_M = 0$)
I	$P^{-1}(z)$	I	0	left/right	reverse additive
$P(z)$	I	I	0	left	reverse output multipl.
$P(z)$	I	I	0	right	reverse input multipl.
$N(z)$	$M(z)$	I	0	left/right	coprime factor ($\Delta_N = 0$)
$N(z)$	$M(z)$	$N_C(z)$	$M_C(z)$	left/right	coprime factor

Table.5.1: Model error structures and corresponding coprime factor choices.

<u>Remark 1:</u> For some model error structures we like to choose $M_C(z) = 0$. This corresponds with a controller $K(z) = \infty$. We will explain how such a controller can be interpreted.

Suppose a square and invertible system $P(z)$ with a stable inverse $P^{-1}(z)$ is in closed loop with a controller $K(z) = \beta K_0(z)$ where $K_0(z)$ and $K_0^{-1}(z)$ are stable. We choose as left coprime factors of $P(z)$ and $K(z)$:

$$M(z) = P(z) \, , \;\; N(z) = I \, , \;\; M_C(z) = \beta^{-1} K_0^{-1}(z) \, , \;\; N_C(z) = I$$

The closed loop will now be stable if and only if $\Lambda = M(z)M_C(z) + N(z)N_C(z)$ is unimodular, so if and only if Λ and Λ^{-1} are both in $R H_\infty$ (Vidyasagar [62]). We find that $\Lambda = \beta^{-1}K_0^{-1}(z)P^{-1}(z) + I$ is in $R H_\infty$ because $K_0(z)$ and $P^{-1}(z)$ are in $R H_\infty$. Furthermore if we let β go to infinity we find

$$\lim_{\beta \to \infty} \Lambda^{-1}(z) = \lim_{\beta \to \infty} (\beta^{-1}K_0^{-1}(z)P^{-1}(z) + I)^{-1} = I$$

which is in $R H_\infty$. So for $\beta \to \infty$ the controller $K(z)$ will stabilize $P(z)$.

If we write $K(z) = \infty$ we mean that we are dealing with a stable minimum phase controller $K(z) = \beta K_0(z)$ of unbounded gain $\beta \to \infty$, with left coprime factors $M_C(z) = 0$ and $N_C(z) = I$.

The same holds for the right coprime case.

<u>Remark 2:</u> In this section we showed that many model error structures fit in the scheme of figure 5.7 and figure 5.8. However, on the model error structure there are some constraints that are not necessary for the use of robust control. For instance, in the case of an additive model error, we require the process to be stable. In the case of a multiplicative model error we even require the transfer matrix to be unimodular.

The reason to introduce these requirements is purely technical and guarantees solvability of our identification problem. A first constraint is that the model error must have the same size as the true process. A second constraint is that the model error must be stable. By formulating the problem in the scheme of figure 5.7 and figure 5.8. we can guarantee that these requirements are fulfilled.

Remark 3: To get the scheme as in figure 5.7 and figure 5.8 we have to calculate two coprime factorizations, one for the model $P(z)$ and one for the controller $K(z)$. The problem, however, is that a coprime factorization is not unique and can be obtained in many different ways.

The choice of the coprime factorization of the model may depend on some physical insight in the process, or can be motivated by the parameter estimation (see chapter 8). We can also choose a normalized coprime factorization, so that the final model will have some nice properties ([43]). The choice of coprime factorization of the model is highly correlated with the parametrization of the model and the choice of weighting filters for the model error. We will comment on this problem in chapter 8.

Remark 4: A excellent choice for the representation of the controller is the normalized coprime factorization. By formula 5.3 in that case there holds that the norm of the model error that is chosen is equal to the norm of the stable function $\Delta(z)$ in figure 5.7 and figure 5.8. For the additive, multiplicative, reverse additive and reverse multiplicative model errors, one of the coprime factors of the controller will be zero, so that the other coprime factor can be chosen as the identity matrix. For that choice the function $\Delta(z)$ in figure 5.7 and figure 5.8 represents the model error, and we have

$$
\begin{array}{lll}
\text{additive :} & \Delta_a(z) & = & \Delta(z) \\
\text{multiplicative :} & \Delta_m(z) & = & \Delta(z) \\
\text{inverse additive :} & \Delta_{ia}(z) & = & \Delta(z) \\
\text{inverse multiplicative :} & \Delta_{im}(z) & = & \Delta(z)
\end{array}
$$

Simularly, in the case of a left coprime model error structure we obtain the model error

$$
[\, \Delta_N(z) \; \Delta_M(z) \,] \; = \; \Delta(z)[\, M_C(z) \; - N_C(z) \,]
$$

If we choose a normalized left coprime factorization for the controller the matrix $[\, M_C(z) \; - N_C(z) \,]$ will be a coinner function and the norm of the function $\Delta(z)$ will be equal to the norm of the coprime factor model error

$$
\| [\Delta_N(z) \; \Delta_M(z)] \| =
$$

$$
= \| \Delta(z)[\, M_C(z) \; - N_C(z) \,] \| =
$$

$$= \parallel \Delta(z) \parallel \quad \text{for all } z \in \mathcal{C}$$

The same holds in the case of a right coprime model error structure.

So instead of considering the norm of the model error that is chosen, we can just consider the norm of the stable function $\Delta(z)$. Another choice than a normalized coprime factorization for the controller leads to a kind of weighting of the model error.

In chapter 7, where we consider the norm of the model error, we will introduce weighting filters. Since we have seen that all model error structures can be put in the framework of coprime model errors there is no need to further distinguish them. Motivated by Remark 4 we assume in the remainder of this thesis

- We will only consider a <u>normalized</u> coprime factorization of the controller $K(z)$.

- As a consequence the stable function $\Delta(z)$ can be denoted as <u>the model error</u>.

PART B:
FORMALIZATION

6

Star products

In the previous chapters we considered several topics, like the true process, the model, the model error with its structure, the R-parametrization framework, disturbance signals with their bounding functions, sets for the scaled true noise.

So far we discussed the following:

- The true model error $\Delta_t(z)$ in a specific structure is a function of the true process and the chosen nominal model.

- The true process can be expressed as a function of the measured signals $u(z)$ and $y(z)$ and the true noise signals $d_t(z)$ and $e_t(z)$.

- The true noise signals can be scaled by the disturbance bounding filters $W_d(z)$ and $W_e(z)$ and we obtain the true scaled noise matrix $Q_t(z)$.

- The true scaled noise matrix $Q_t(z)$ is in a set $\tilde{Q}$, either the structured scaled noise set $\tilde{Q}_s$ or the unstructured scaled noise set $\tilde{Q}_u$.

In this section we will introduce the notion of the Redheffer star product (Redheffer [50]), that describes the interconnection of two or more systems in a

95

structured way. The star product has already shown to be a usefull tool in robust control theory (Doyle et al. [14]). The results of this chapter can be achieved without the use of the Redheffer star product. However, we will use the star product because it gives a good insight in the interaction between model error, model error structure, model coprime factors, measured data, the noise bounds and the true scaled noise matrix. Finally, the star product description makes it very easy to link our identification problem to results from the robust control optimization, such as the computation of the structured singular value μ (we will discuss this computation in chapter 7).

We will first introduce the mathematical concepts of a star product. Then we will use the star product for the true process, the model error structure and for the model error itself.

6.1   Notation and properties of star products

Suppose we have two systems $A \in R\!P$ and $B \in R\!P$ which are partitioned as

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$

Suppose that

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = A \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$\begin{bmatrix} y_3 \\ y_4 \end{bmatrix} = B \begin{bmatrix} v_3 \\ v_4 \end{bmatrix}$$

and consider the interconnection as in figure 6.1 where $v_2 = y_3$ and $v_3 = y_2$.

If the matrix of $I - A_{22}B_{11}$ is well-defined and invertible, we can connect the lower outputs of A with the upper inputs of B, and connect the upper outputs of B with the lower inputs of A, as shown in figure 6.1. The interconnected system again is a map and is described by

$$\begin{bmatrix} y_1 \\ y_4 \end{bmatrix} = S(A, B) \begin{bmatrix} v_1 \\ v_4 \end{bmatrix}$$

where $S(A, B)$ is called the Redheffer star product [50], with
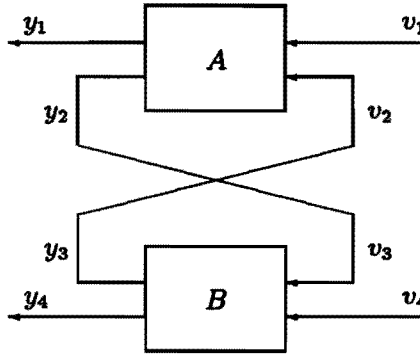
Figure 6.1: Interconnected system

$$S(A,B) =$$
$$= \begin{bmatrix} A_{11} + A_{12}B_{11}(I - A_{22}B_{11})^{-1}A_{21} & A_{12}(I - B_{11}A_{22})^{-1}B_{12} \\ B_{21}(I - A_{22}B_{11})^{-1}A_{21} & B_{22} + B_{21}A_{22}(I - B_{11}A_{22})^{-1}B_{12} \end{bmatrix}$$

If A_{11}, A_{12} and A_{21} are empty then

$$S(A,B) = B_{22} + B_{21}A_{22}(I - B_{11}A_{22})^{-1}B_{12} = F_u(B, A_{22})$$

is an upper linear fractional transformation.

If B_{22}, B_{12} and B_{21} are empty then

$$S(A,B) = A_{11} + A_{12}B_{11}(I - A_{22}B_{11})^{-1}A_{21} = F_l(A, B_{11})$$

is a lower linear fractional transformation.

If $S(A,B)$, $S(B,C)$ and $S(S(A,B),C)$ are well-defined, it is immediate that

$$S(S(A,B),C) = S(A,S(B,C)) \overset{def}{=} S(A,B,C)$$

If $S(A,B)$ is well-defined, but $S(B,C)$ is ill-defined (so $\det(I - B_{22}C_{11}) = 0$), it can happen that $S(S(A,B),C)$ is well-defined.

Example: Suppose the constant matrices:

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 0 & -1/2 \\ 1 & 1 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 3 \\ 1 & 0 \end{bmatrix}$$

Then $S(B,C)$ does not exist because $\det(I - B_{22}C_{11}) = 0$. However, if we first compute $S(A,B)$ and then $S\left(S(A,B),C\right)$ we get:

$$S\Big(S(A,B),C\Big) = S\left(\begin{bmatrix} 2 & -1/2 \\ 1 & 1/2 \end{bmatrix}, \begin{bmatrix} 1 & 3 \\ 1 & 0 \end{bmatrix}\right) = \begin{bmatrix} 1 & -3 \\ 2 & 3 \end{bmatrix}$$

Finally we give the following lemma:

Lemma 6.1:

Suppose a matrix A is partitioned as $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$ with A_{22} invertible.
Let

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

where y_1, y_2, x_1 and x_2 are vectors, compatible with the partitioning of A.
Then we can interchange the input vector x_2 with output vector y_2 and the
following will hold

$$\begin{bmatrix} y_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} A_{11} - A_{12} A_{22}^{-1} A_{21} & A_{12} A_{22}^{-1} \\ -A_{22}^{-1} A_{21} & A_{22}^{-1} \end{bmatrix} \begin{bmatrix} x_1 \\ y_2 \end{bmatrix} \tag{6.1}$$

$\square$

Proof:
We have

$$\begin{aligned} y_1 &= A_{11} x_1 &+ A_{12} x_2 \\ y_2 &= A_{21} x_1 &+ A_{22} x_2 \end{aligned}$$

So from the last equation it follows

$$x_2 = -A_{22}^{-1} A_{21} x_1 + A_{22}^{-1} y_2$$

Substitution into the first equation gives

$$y_1 = A_{11} x_1 + A_{12} \left(-A_{22}^{-1} A_{21} x_1 + A_{22}^{-1} y_2 \right)$$

This results in:

$$\begin{bmatrix} y_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} A_{11} - A_{12} A_{22}^{-1} A_{21} & A_{12} A_{22}^{-1} \\ -A_{22}^{-1} A_{21} & A_{22}^{-1} \end{bmatrix} \begin{bmatrix} x_1 \\ y_2 \end{bmatrix} \tag{6.2}$$

$\square$

Remark: In this chapter the Redheffer star product is used in a rather informal
way. The partitioning of the various matrices is considered to be known and
we assume the interconnections to be well-defined.

6.2 The true process and the noise as a star product

In this section we are going to write the true process as a star product of two matrices, where the first matrix consists of the sampled data and the second matrix is the scaled noise matrix $Q_t(z)$. In chapter 4 we derived that in the MIMO-case there holds

$$P_t = (Y - E_t)(U + D_t)^{-1}$$

under the assumption that the matrix $U_t(z)$ is invertible. We will also assume the matrix $U(z)$ to be invertible. (The case where U_t or U is not invertible will be treated in section 6.5).

First we claim that we can express the true process as a star product

$$P_t(z) = S\Big(T(z), F_t(z)\Big)$$

where the matrix $T(z)$ is defined as

$$T(z) = \begin{bmatrix} T_{11}(z) & T_{12}(z) \\ T_{21}(z) & T_{22}(z) \end{bmatrix} =$$

$$= \begin{bmatrix} Y(z)U^{-1}(z) & \vdots & -Y(z)U^{-1}(z) & -I \\ \cdots\cdots & & \cdots\cdots & \cdots \\ U^{-1}(z) & \vdots & -U^{-1}(z) & 0 \end{bmatrix} \tag{6.3}$$

and the matrix

$$F_t = \begin{bmatrix} D_t \\ E_t \end{bmatrix}$$

Indeed:

$$\begin{aligned}
P_t &= (Y - E_t)(U + D_t)^{-1} \\
&= (Y - [0\ I]F_t)(U + [I\ 0]F_t)^{-1} \\
&= \{Y + ([YU^{-1}\ 0] + [-YU^{-1}\ -I])F_t\}(U + [I\ 0]F_t)^{-1} \\
&= \{YU^{-1}(U + [I\ 0]F_t) + [-YU^{-1}\ -I]F_t\}(U + [I\ 0]F_t)^{-1} \\
&= YU^{-1} + [-YU^{-1}\ -I]F_t(U + [I\ 0]F_t)^{-1} \\
&= YU^{-1} + [-YU^{-1}\ -I]F_t(I + [U^{-1}\ 0]F_t)^{-1}U^{-1} \\
&= T_{11} + T_{12}\ F_t\ (I - T_{22}\ F_t)^{-1}\ T_{21} \\
&= S(T, F_t)
\end{aligned}$$

So the true process can be written as a star product of the matrix $T(z)$ with the true noise matrix $F_t(z)$ (see figure 6.2). Note that we treat the true noise matrix $F_t(z)$ and the signal matrices $U(z)$ and $Y(z)$ as transfer matrices instead of signal matrices. That is, the matrices $F_t(z)$, $U(z)$ and $Y(z)$ are interpreted as transfer functions in the space RH_∞ with the original time signals in $\mathbb{L}_2[0, 2N - 1]$ as Markov parameters.
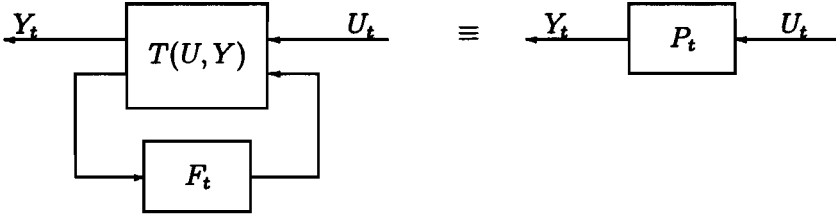
Figure 6.2: True process as a star product $P_t(z) = S(T(z), F_t(z))$.

We do the same manipulation for the true noise matrix $F_t(z)$ and write this matrix as a star product of a weighting filter and a scaled noise matrix.

- If we consider the <u>unstructured</u> noise matrix Q_{ut} with the corresponding bounding filter $W(z)$, (where Q_{ut} and $W(z)$ as defined in section 4.3), we derive

$$F_t = \begin{bmatrix} D_t \\ E_t \end{bmatrix} = W Q_{ut} = S(V, Q_t)$$

where the matrices $V(z)$ and $Q_t(z)$ are defined as

$$V(z) = \begin{bmatrix} 0 & W(z) \\ I & 0 \end{bmatrix}$$

$$Q_t(z) = Q_{ut}(z)$$

(6.4)

- If we consider the <u>structured</u> noise matrix Q_{st} with the corresponding filters $W(z)$, V_1 and V_2 (where Q_{ut}, $W(z)$, V_1 and V_2 as defined in section 4.3), we derive

$$F_t = \begin{bmatrix} D_t \\ E_t \end{bmatrix} = W V_1 Q_{st} V_2 = S(V, Q_t)$$

where the matrices $V(z)$ and $Q_t(z)$ are defined as

$$V(z) = \begin{bmatrix} 0 & W(z)V_1 \\ V_2 & 0 \end{bmatrix}$$

(6.5)

$$Q_t(z) = Q_{\bullet t}(z)$$

So $F_t = S(V, Q_t)$ where $V(z)$ and $Q_t(z)$ are defined either as in equation 6.4 for the unstructured case, or as in equation 6.5 for the structured case. If we combine these results $P_t = S(T, F_t)$ and $F_t = S(V, Q_t)$ we get:

$$P_t = S(T, F_t) = S(T, S(V, Q_t)) = S(T, V, Q_t)$$

which can be represented by a scheme as in figure 6.3
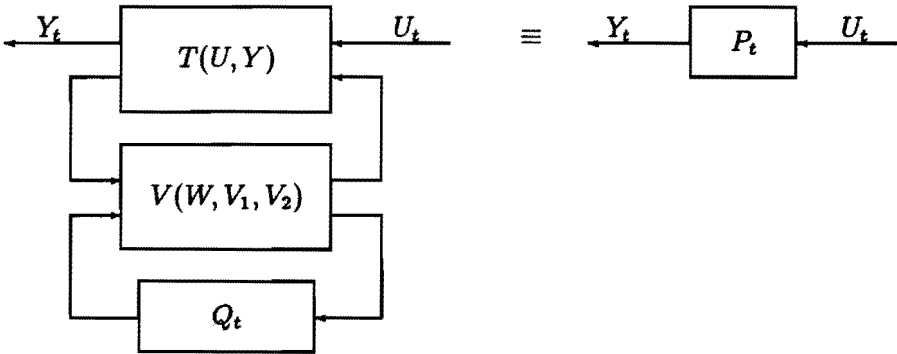
Figure 6.3: True process as a star product $P_t(z) = S(T(z), V(z), Q_t(z))$

6.3   Standard form

In this section we introduce a standard form in which the true process is written as a star product of the model coprime factors $N(z)$ and $M(z)^{-1}$, the true model error $\Delta_t(z)$ and a standard matrix $H(z)$, containing the coprime factors $N_C(z)$ and $M_C(z)$ of the stabilizing controller (see chapter 5). We assume $N(z)$, $M(z)$ and $\Delta_t(z)$ to be stable.
Consider figure 6.4:

We introduce a matrix $H(z)$ such that

$$\begin{bmatrix} \Phi_1 \\ \Psi_2 \\ W_t \\ Y_t \end{bmatrix} = H(z) \begin{bmatrix} \Phi_2 \\ \Psi_1 \\ X_t \\ U_t \end{bmatrix}$$
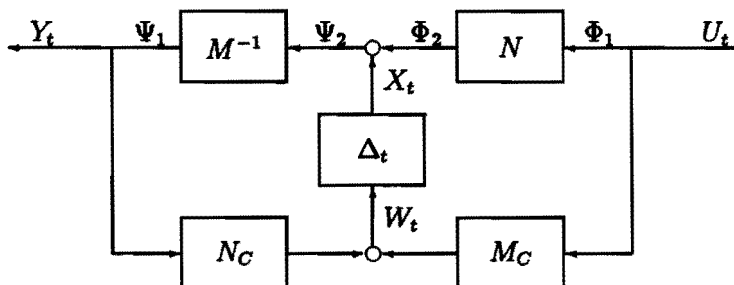
Figure 6.4: Model error scheme with left coprime factors

It is easily derived from figure 6.4 that $H(z)$ is given by

$$H(z) = \begin{bmatrix} 0 & 0 & 0 & I \\ I & 0 & I & 0 \\ 0 & N_C(z) & 0 & M_C(z) \\ 0 & I & 0 & 0 \end{bmatrix}$$

The function $H(z)$ is clearly in $R I\!H_\infty$. Now we close the upper loops from W_t, Ψ_2 and Φ_1 to X_t, Ψ_1 and Φ_2 with the transfer functions $\Delta_t(z)$, $M^{-1}(z)$ and $N(z)$ respectively as in figure 6.5.
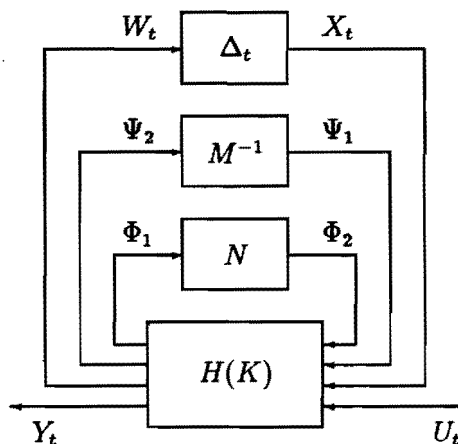
Figure 6.5: Standard form

We obtain the transfer function from U_t to Y_t, which is equal to the transfer function of the true process. We can write $P_t(z)$ as the star product:

$$P_t(z) = S\left(\begin{bmatrix} N(z) & 0 & 0 \\ 0 & M(z)^{-1} & 0 \\ 0 & 0 & \Delta_t(z) \end{bmatrix}, H(z) \right) \tag{6.6}$$

Remark: Of course one can think of model error structures that do not fit into the scheme of figure 5.7, but can be described in terms of a transfer function $H(z)$, with model coprime factors $N(z)$ and $M(z)$ and the true model error $\Delta_t(z)$.

6.4   Model error as a star product

In the previous section we introduced the standard form

$$P_t(z) = S\left(\begin{bmatrix} N(z) & 0 & 0 \\ 0 & M(z)^{-1} & 0 \\ 0 & 0 & \Delta_t(z) \end{bmatrix}, H(z) \right)$$

where $P_t(z)$ is given as a function of the model coprime factors $N(z)$ and $M(z)$, the true model error $\Delta_t(z)$ and the standard matrix $H(z)$. By manipulating this matrix $H(z)$ we can transform the interconnections in such a way that we obtain the model error as a function of the model coprime factors, the true process and a manipulated matrix $\hat{H}(z)$.

Consider figure 6.4 where we introduced the matrix $H(z)$. We can partition $H(z)$ in such a way that:

$$\begin{bmatrix} \Phi_1 \\ \Psi_2 \\ W_t \\ Y_t \end{bmatrix} = \begin{bmatrix} H_{11}(z) & H_{12}(z) & H_{13}(z) & H_{14}(z) \\ H_{21}(z) & H_{22}(z) & H_{23}(z) & H_{24}(z) \\ H_{31}(z) & H_{32}(z) & H_{33}(z) & H_{34}(z) \\ H_{41}(z) & H_{42}(z) & H_{43}(z) & H_{44}(z) \end{bmatrix} \begin{bmatrix} \Phi_2 \\ \Psi_1 \\ X_t \\ U_t \end{bmatrix}$$

Using Lemma 6,1, we will now interchange the lower output vector $\begin{bmatrix} \Psi \\ W \\ Y \end{bmatrix}$ with the lower input vector $\begin{bmatrix} \Psi \\ X \\ U \end{bmatrix}$.

There are three reasons to do so :

1. We like to work with $M(z)$ instead of $M^{-1}(z)$.

2. We like to close the loop from U_t to Y_t with the transfer function of the true process $P_t(z)$.

3. We like to obtain W_t to become the input and X_t the output of the interconnected system, so that this system will describe the transfer function of the true model error $\Delta_t(z)$.

We define the matrix

$$H_{rl}(z) = \begin{bmatrix} H_{22}(z) & H_{23}(z) & H_{24}(z) \\ H_{32}(z) & H_{33}(z) & H_{34}(z) \\ H_{42}(z) & H_{43}(z) & H_{44}(z) \end{bmatrix}$$

For the set up of figure 6.4 the matrix $H_{rl}(z)$ will be

$$H(z) = \begin{bmatrix} 0 & I & 0 \\ N_C(z) & 0 & M_C(z) \\ I & 0 & 0 \end{bmatrix}$$

We assume the matrix $H_{rl}(z)$ to be invertible, which is the case if $M_c(z)$ is invertible. (The case where $M_c(z)$ and H_{rl} are not invertible will be discussed in section 6.5).

If H_{rl} is invertible we can derive with equation (6.1):

$$\begin{bmatrix} \Phi_1 \\ \Psi_1 \\ X_t \\ U_t \end{bmatrix} = \hat{H} \begin{bmatrix} \Phi_2 \\ \Psi_2 \\ W_t \\ Y_t \end{bmatrix}$$

where

$$\hat{H}(z) = \begin{bmatrix} H_{11} - [H_{12}\ H_{13}\ H_{14}]H_{rl}^{-1} \begin{bmatrix} H_{21} \\ H_{31} \\ H_{41} \end{bmatrix} & [H_{12}\ H_{13}\ H_{14}]H_{rl}^{-1} \\ -H_{rl}^{-1} \begin{bmatrix} H_{21} \\ H_{31} \\ H_{41} \end{bmatrix} & H_{rl}^{-1} \end{bmatrix}$$

Using this matrix $\hat{H}(z)$ we get a scheme as in figure 6.6 where we opened the loop from X_t to W_t and where we closed the lower loop with $Y_t(z) = P_t(z)U_t(z)$. The input of this interconnected system is $W_t(z)$, and the output is $X_t(z)$ as desired. This means that this system is equal to the true model error

$$\Delta_t(z) = S(\begin{bmatrix} N(z) & 0 \\ 0 & M(z) \end{bmatrix}, \hat{H}(z), P_t(z)) \tag{6.7}$$

For example, if we consider the matrix $H(z)$ corresponding to figure 6.4 and we assume $\hat{M}_C$ to be invertible, then $\begin{bmatrix} 0 & I & 0 \\ N_C & 0 & M_C \\ I & 0 & 0 \end{bmatrix}$ is invertible and we obtain:
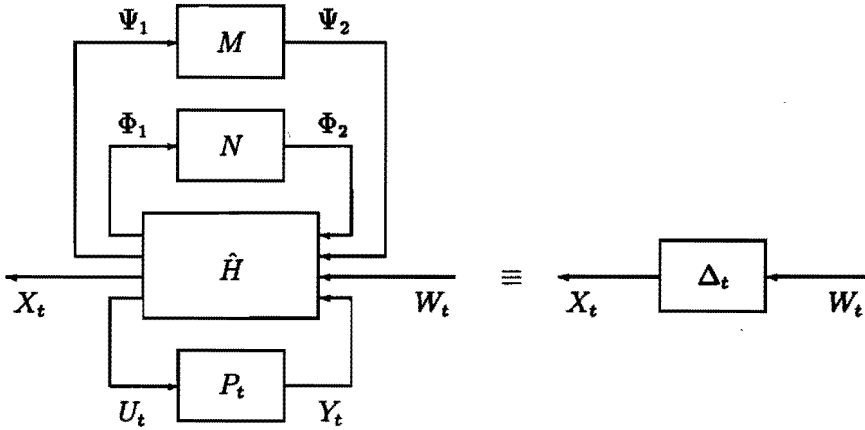
Figure 6.6: True model error as a star product

$$
\hat{H}(z) = \begin{bmatrix} 0 & 0 & M_C(z)^{-1} & -K(z) \\ 0 & 0 & 0 & I \\ -I & I & 0 & 0 \\ 0 & 0 & M_C(z)^{-1} & -K(z) \end{bmatrix}
$$

We already derived $P_t = S(T, V, Q_t)$. When we substitute this in equation (6.7) for the true model error we get

$$
\Delta_t(z) = S\left(\begin{bmatrix} N(z) & 0 \\ 0 & M(z) \end{bmatrix}, \hat{H}(z), T(z), V(z), Q_t(z) \right) \tag{6.8}
$$

which is illustrated in figure 6.7 .

Define

$$
G(z) = S\left(\begin{bmatrix} N(z) & 0 \\ 0 & M(z) \end{bmatrix}, \hat{H}(z), T(z), V(z) \right) . \tag{6.9}
$$

Then $\Delta_t(z) = S(G(z), Q_t(z))$. Note that $G(z)$ is built up with known information, $\{N(z), M(z)\}$ is the chosen model, $\hat{H}(z)$ is determined by the chosen model error structure, $T(z)$ is built up with the data $\{U(z), Y(z)\}$ and $V(z)$ contains the knowledge of the noise bounds $\{W_d(z), W_e(z)\}$.

For the set up of figure 6.4 we derive

$$
G(z) =
$$

$$
= \begin{bmatrix} \hat{X}\hat{W}^{-1} & \vdots & -\hat{X}\hat{W}^{-1}M_C V_{11} - N V_{11} & \hat{X}\hat{W}^{-1}N_C V_{12} - M V_{12} \\ \cdots\cdots & & \cdots\cdots\cdots\cdots\cdots & \cdots\cdots\cdots\cdots\cdots \\ V_2\hat{W}^{-1} & \vdots & -V_2\hat{W}^{-1}M_C V_{11} & V_2\hat{W}^{-1}N_C V_{12} \end{bmatrix} \tag{6.10}
$$

$$\Rightarrow \text{model}$$

$$\Rightarrow \text{model error structure}$$

$$\Rightarrow \text{data}$$

$$\Rightarrow \text{noise bounds}$$

$$\Rightarrow \text{scaled true noise}$$
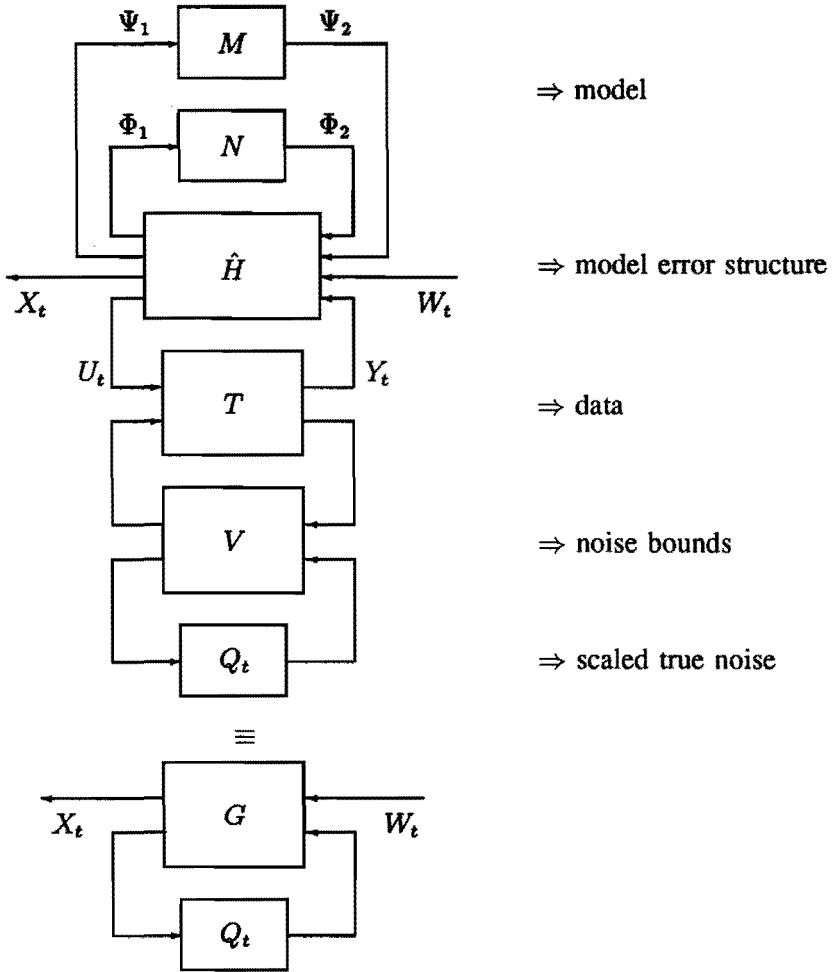
Figure 6.7: True model error as a star product

where $\hat{W}(z) \overset{def}{=} M_c U + N_c Y$ and $\hat{X}(z) \overset{def}{=} MY - NU$. These signal matrices $\hat{W}$ and $\hat{X}$ are related with the signal matrices W_t and X_t in figure 6.4 for the noiseless case, so for $Q_t = 0$.

We can now compute the model error from (6.10) by:

$$\Delta_t(z) = S\big(G(z), Q_t(z)\big) =$$

$$= \left(\hat{X} + [-N \ -M]V_1 Q_t V_2 \right)\left(\hat{W} + [M_c \ -N_c]V_1 Q_t V_2 \right)^{-1} =$$

$$= (MY - NU - ND_t - ME_t)(N_c Y + M_c U + M_c D_t - N_c E_t)^{-1} =$$

$$= X_t(z)\, W_t^{-1}(z)$$

which corresponds to figure 6.4.

6.5   Generalization

In section 6.2 and section 6.4 we made the assumptions that

1. The matrix $U(z)$ or $U_t(z)$ is invertible.

2. The matrix $H_{rt}(z)$ is invertible.

In this section these conditions are relaxed and we show that we can still compute the matrix $G(z)$.

1. The matrix $U(z)$ or $U_t(z)$ is not invertible:
The matrix $U_t(z)$ will become singular if the true process $P_t(z)$ has a zero on the unit circle. If z_0 is a zero of $P_t(z)$ (with $|z_0| = 1$), the feedback will make the matrix $U_t(z_0)$ singular in z_0 and the matrix $U(z_0)$ will be singular or close to singularity.

If the matrix $U(z)$ or the matrix $U_t(z)$ is not invertible, we can still solve the problem if the following conditions are satisfied:
The signal $\hat{W}(z) \stackrel{def}{=} M_C(z)U(z) + N_C(z)Y(z)$ is invertible.

The signal $\hat{W}_t(z) \stackrel{def}{=} M_C(z)U_t(z) + N_C(z)Y_t(z)$ is invertible.
where $M_C(z)$ and $N_C(z)$ are the left coprime factors of a stabilizing controller. This assumption is reasonable, for if we use the controller $K(z) = M_C^{-1}(z)N_C(z)$ for the experiments, we find that $\hat{W}(z) \stackrel{def}{=} M_C(z)U(z) + N_C(z)Y(z) = M_C^{-1}(z)R(z)$, where the reference signal matrix $R(z)$ is choosen invertible and where $M_C(z)$ is invertible. If the noise is not to big $\hat{W}_t(z)$ will also be invertible.

Now instead of looking at the true process we consider the transfer matrix

$$\bar{P}_t \stackrel{def}{=} Y_t\,\hat{W}_t^{-1} = Y_t(\, M_C(z)U_t(z) + N_C(z)Y_t(z)\,)^{-1}$$

We derive:
$$\bar{P}_t =$$

$$= Y_t(\, M_C(z)U_t(z) + N_C(z)Y_t(z)\,)^{-1}$$

$$= (Y - E_t)(M_C(z)U(z) + N_C(z)Y(z) + M_C(z)D_t(z) - N_C(z)E_t(z))^{-1}$$

$$= (Y - [0 \; I]F_t)(\hat{W} + [M_C \quad -N_C]F_t)^{-1}$$

$$= \{Y + ([Y\hat{W}^{-1}M_C \quad -Y\hat{W}^{-1}N_C] + [-Y\hat{W}^{-1}M_C \; Y\hat{W}^{-1}N_C - I])F_t\} \cdot$$

$$\cdot (\hat{W} + [M_C \quad -N_C]F_t)^{-1}$$

$$= \{Y\hat{W}^{-1}(\hat{W} + [M_C \quad -N_C]F_t) + [-Y\hat{W}^{-1}M_C \; Y\hat{W}^{-1}N_C - I]F_t\} \cdot$$

$$\cdot (\hat{W} + [M_C \quad -N_C]F_t)^{-1}$$

$$= Y\hat{W}^{-1}(\hat{W} + [M_C \quad -N_C]F_t) + [-Y\hat{W}^{-1}M_C \; Y\hat{W}^{-1}N_C - I]F_t \cdot$$

$$\cdot (\hat{W} + [M_C \quad -!N_C]F_t)^{-1}$$

$$= Y\hat{W}^{-1} + [-Y\hat{W}^{-1}M_C \; Y\hat{W}^{-1}N_C - I]F_t \cdot$$

$$\cdot (I + [-\hat{W}^{-1}M_C \quad -\hat{W}^{-1}N_C]F_t)^{-1}\hat{W}^{-1}$$

$$= S(T_{w2}, F_t)$$

where the matrix T_{w2} is defined as

$$T_{w2} = \begin{bmatrix} Y\hat{W}^{-1} & \vdots & -Y\hat{W}^{-1}M_C & Y\hat{W}^{-1}N_C - I \\ \cdots\cdots & & \cdots\cdots\cdots & \cdots\cdots\cdots \\ \hat{W}^{-1} & \vdots & \hat{W}^{-1}M_C & -\hat{W}^{-1}N_C \end{bmatrix} \qquad (6.11)$$

By definition of T_{w2} it will hold:

$$Y_t(z) = S(T_{w2}, F_t)(M_C U_t + N_C Y_t)$$

Now we introduce the matrix T_{w1} as

$$T_{w1} \stackrel{def}{=} \begin{bmatrix} 0 & I \\ M_C & N_C \end{bmatrix} \qquad (6.12)$$

and it will hold:

$$\begin{bmatrix} Y_t \\ \hat{W}_t \end{bmatrix} = T_{w1} \begin{bmatrix} U_t \\ Y_t \end{bmatrix}$$

The previously defined $G(z)$ is now replaced by

$$G_u(z) = S\left(S\left(\begin{bmatrix} N & 0 \\ 0 & M \end{bmatrix}, \hat{H}, T_{w1} \right), S\left(T_{w2}, V \right) \right) \qquad (6.13)$$
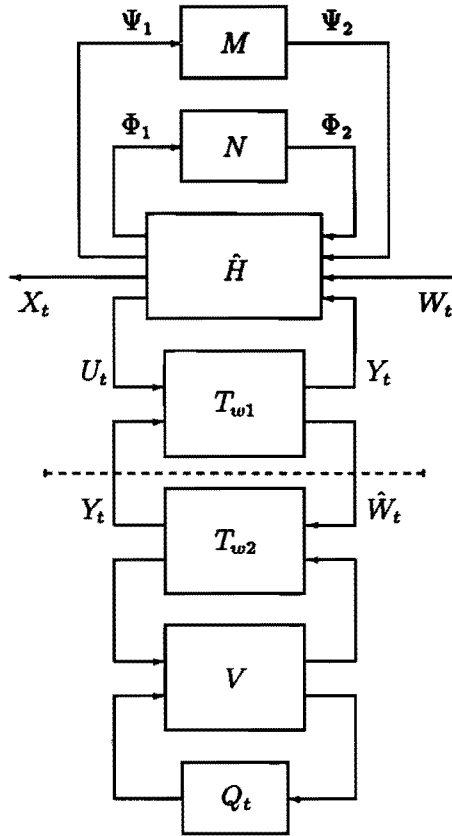
Figure 6.8: True model error as a star product for U not invertible

and we get the scheme as shown in figure 6.8

Remark: If we choose for $H(z)$ the set up with left coprime factors as in figure6dpar, then $W_t(z) = \hat{W}_t(z)$, and the final $G_u(z)$ will be equal to the result of equation (6.10).

So, even if $S(T_{w1}, T_{w2})$ does not exist, we can still compute $G_u(z)$ by splitting $T(z)$ in two parts. Note that if $U(z)$ and $U_t(z)$ are invertible there holds $T(z) = S(T_{w1}, T_{w2})$ and $G(z) = G_u(z)$

2. The matrix $H_{rl}(z)$ is not invertible:

If the matrix $H_{rl}(z)$ is not invertible, we can still solve the problem if the three following conditions are satisfied:

1. The matrix $H_{cen}(z) = \begin{bmatrix} H_{22}(z) & H_{23}(z) \\ H_{32}(z) & H_{33}(z) \end{bmatrix}$ is invertible.

2. The process $P_t(z)$ is sqaure and invertible for all $z \in C$.

3. The signal matrices $Y_t(z)$ and $Y(z)$ are invertible.

The matrix $H_{rt}(z)$ will not be invertible if we choose for a reverse type of model error structure because then $M_C(z) = 0$ for the controller controller $K(z) = \infty$ (See remark 1 of section 5.4).

However for a reverse type of model error structure we get a controller co-prime factor and $N_C(z) = I$. This makes the matrix $H_{cen}(z)$ invertible. Further for all the reverse type model errors structures we have the condition that the process $P_t^{-1}(z) \in R I\!H_\infty$, and so there are no zeros outside the unit disk. If we make sure that the reference signal $R(z)$ is invertible we will find that $Y_t(z)$ and $Y(z)$ are invertible.

If H_{cen} is invertible we can derive from Lemma 6.1:

$$\begin{bmatrix} \Phi_1 \\ \Psi_1 \\ X_t \\ Y_t \end{bmatrix} = \hat{H}_c \begin{bmatrix} \Phi_2 \\ \Psi_2 \\ W_t \\ U_t \end{bmatrix}$$

where

$$\hat{H}_c(z) = \begin{bmatrix} H_{11} - H_{123} H_{cen}^{-1} H_{231} & H_{123} H_{cen}^{-1} & H_{14} - H_{123} H_{cen}^{-1} H_{234} \\ -H_{cen}^{-1} H_{231} & H_{cen}^{-1} & -H_{cen}^{-1} H_{234} \\ H_{41} - H_{423} H_{cen}^{-1} H_{231} & H_{423} H_{cen}^{-1} & H_{44} - H_{423} H_{cen}^{-1} H_{234} \end{bmatrix}$$

with

$$H_{123} = [H_{12} \ H_{13}] \quad , \quad H_{423} = [H_{42} \ H_{43}]$$

$$H_{231} = \begin{bmatrix} H_{21} \\ H_{31} \end{bmatrix} \quad , \quad H_{234} = \begin{bmatrix} H_{24} \\ H_{34} \end{bmatrix}$$

Using this matrix $\hat{H}_c(z)$ we get a scheme as in figure 6.9 where we opened the loop from X_t to W_t and where we closed the lower loop with $U_t(z) = P_t^{-1}(z)Y_t(z)$. Note that in comparison with figure 6.6 the signals $U_t(z)$ and $Y_t(z)$ are not interchanged, so that we close the loop with $P_t^{-1}(z)$ instead of $P_t(z)$. The input of this interconnected system is $W_t(z)$, and the output is $X_t(z)$.

This means that this system is equal to the true model error
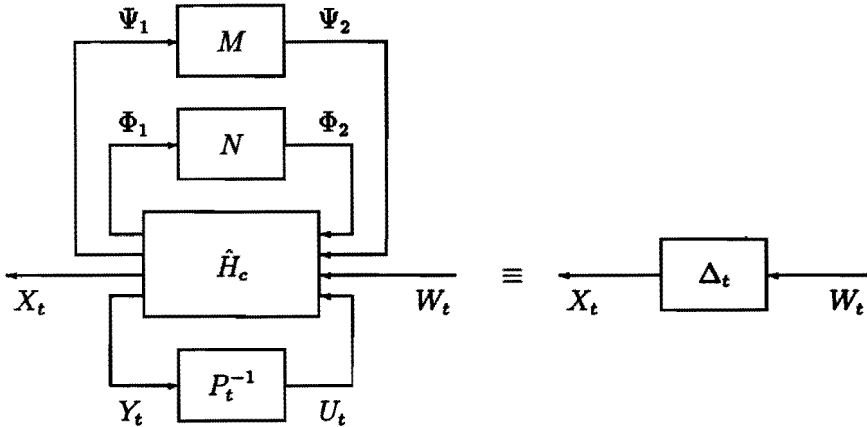
Figure 6.9: True model error as a star product

$$\Delta_t(z) = S(\begin{bmatrix} N(z) & 0 \\ 0 & M(z) \end{bmatrix}, \hat{H}_c(z), P_t^{-1}(z)) \tag{6.14}$$

Consider the $H(z)$ corresponding to figure 6.4 with $M_C = 0$ and N_C is invertible, then $\begin{bmatrix} 0 & I \\ N_C & 0 \end{bmatrix}$ is invertible and we obtain:

$$\hat{H}_c(z) = \begin{bmatrix} 0 & 0 & 0 & I \\ 0 & 0 & N_C^{-1}(z) & 0 \\ -I & I & 0 & 0 \\ 0 & 0 & N_C^{-1}(z) & 0 \end{bmatrix}$$

In section 6.2 we derived the matrix $T(z)$ to get the relation between $P_t(z)$ and $F_t(z)$. In the same way we can compute a matrix $T_c(z)$, that gives the relation between $P_t^{-1}(z)$ and $F_t(z)$. We find

$$T_c = \begin{bmatrix} UY^{-1} & \vdots & I & UY^{-1} \\ \dots\dots & & \dots\dots & \dots\dots \\ Y^{-1} & \vdots & 0 & Y^{-1} \end{bmatrix}$$

so that $P_t^{-1} = S(T_c, V, Q_t)$. When we substitute this in the equation for the true model error we get

$$\Delta_t(z) = S(G_c, Q_t)$$

where $G_c(z)$ is defined as

$$G_c(z) = S(\begin{bmatrix} N(z) & 0 \\ 0 & M(z) \end{bmatrix}, \hat{H}_c(z), T_c(z), V(z)) \qquad (6.15)$$

Note that for $H_{rl}(z)$ is invertible, we indeed find $G(z) = G_c(z)$.

7

Model error bounds

7.1 Introduction

In the previous chapter we showed that for a fixed model with a specific model error structure, the true model error can be written as a starproduct of some known matrix function $G(z)$ and a matrix $Q_t(z)$ with the scaled true noise parameters. For every frequency $z \in \mathcal{C}$, we can compute the matrix norm of the modelerror, and we get:

$$\|\Delta_t(z)\| = \|S(G(z), Q_t(z))\|$$

If we want to emphasize the norm at certain frequencies, or if we want to emphasize some input-'directions' or output-'directions', we can introduce weighting matrices $W_{\Delta 1}(z)$ and $W_{\Delta 2}(z)$ and calculate the weighted matrix norm of the model error

$$\| W_{\Delta 1}(z)\, \Delta_t(z)\, W_{\Delta 2}(z) \| =$$

$$= \| W_{\Delta 1}(z)\, S(G(z), Q_t(z))\, W_{\Delta 2}(z) \| =$$

$$= \| S(G_W(z), Q_t(z)) \|$$

where

$$G_W(z) \stackrel{def}{=} \begin{bmatrix} G_{W11}(z) & G_{W12}(z) \\ G_{W21}(z) & G_{W22}(z) \end{bmatrix} = \begin{bmatrix} W_{\Delta 1} G_{11} W_{\Delta 2} & W_{\Delta 1} G_{12} \\ G_{21} W_{\Delta 2} & G_{22} \end{bmatrix}$$

and so we incorporate the weighting filters in G_W.

The matrix $Q_t(z)$, however, is unknown. What we do know is that $Q_t(z)$ (either Q_{ut} or Q_{st}), belongs to some unstructured or structured set $\mathbf{Q}$ (either $\mathbf{Q}_u$ or $\mathbf{Q}_s$).

For every frequency $z \in \mathcal{C}$, we can bound the weighted matrix norm of the model error by:

$$\| W_{\Delta 1}(z) \Delta_t(z) W_{\Delta 2}(z) \| =$$

$$= \| S(G_W(z), Q_t(z)) \| \le$$

$$\le \max_{Q(z) \in \mathbf{Q}} \| S(G_W(z), Q(z)) \|$$

Now we will derive bounds for the matrix norm at one specific frequency z. For this specific frequency $G_W(z)$ and $Q(z)$ are constant complex matrices and will be denoted by G_W and Q.

We will first consider the structured case. Later we will show that the unstructured case sometimes has some advantages.

7.2   An upper bound using structured noise sets

In the case where we consider structured noise sets, we can derive non-conservative bounds, using the so called μ-techniques (Van den Boom [56]). We have

$$Q_t = Q_{st} \quad \text{is a diagonal matrix with} \begin{cases} |Q_{ij}| \le 1 \text{ for } i = j \\ |Q_{ij}| = 0 \text{ for } i \ne j \end{cases}$$

$$V = \begin{bmatrix} 0 & W V_1 \\ V_2 & 0 \end{bmatrix}$$

$$G_W = W_{\Delta 1} S(\begin{bmatrix} \tilde{N} & 0 \\ 0 & \tilde{M} \end{bmatrix}, \hat{H}, T, V) W_{\Delta 2} = \begin{bmatrix} G_{W11} & G_{W12} \\ G_{W21} & G_{W22} \end{bmatrix}$$

$$W_{\Delta 1}\,\Delta_t\,W_{\Delta 2} \;=\; S(G_W, Q_t)$$

and we consider the structured noise set $\mathbf{Q}_s$ with diagonal matrices Q_s satisfying:

$$\| Q_s \| \le 1$$

Then $Q_t \;=\; Q_{st} \in \mathbf{Q}_s$.

We consider the weighted model error

$$W_{\Delta 1}\,\Delta_t\,W_{\Delta 2} \;=\; S(G_W, Q_t)$$

For the matrix norm of the weighted model error there holds

$$\max_{Q_s \in \mathbf{Q}_s} \|S(G_W, Q_s)\| \;\le\; \gamma \quad \text{iff} \quad \mu(G_\gamma) \le 1$$

where we defined:

$$G_\gamma \;=\; \begin{bmatrix} \gamma^{-1}G_{W11} & \gamma^{-1/2}G_{W12} \\ \gamma^{-1/2}G_{W21} & G_{W22} \end{bmatrix}$$

This $\mu(G_\gamma)$ denotes the structured singular value of the matrix G_γ and is defined by Doyle ([10],[11]):

$$\mu(G_\gamma) \;\stackrel{def}{=}\; \left(\min_{\Delta_\mu} \{\sigma_{max}(\Delta_\mu) : \det(I + G_\gamma\Delta_\mu) = 0\} \right)^{-1} \tag{7.1}$$

where

$$\Delta_\mu \;=\; \begin{bmatrix} \Delta_1 & 0 \\ 0 & Q_s \end{bmatrix}$$

and Δ_1 is in the set of $q \times p$ matrices with $\|\Delta_1\| \le 1$ and Q_s is in the set of $p(p+q) \times p(p+q)$ diagonal matrices Q_s with $\|Q_s\| \le 1$ (so $Q_s \in \mathbf{Q}_s$).

If no Δ_μ makes $\det(I + G_\gamma\Delta_\mu)$ singular, then $\mu(G_\gamma) = 0$.

We can adjust the γ in an iterative way until we find $\mu(G_\gamma) = 1$.
    The value μ is difficult to compute, but we can give an upper bound:

$$\mu(G_\gamma) \;\le\; \sigma_{max}(G_\gamma)$$

We can even refine this bound with:

$$\mu(G_\gamma) \;\le\; \inf_{D \in \mathbf{D}} \bar{\sigma}(D G_\gamma D^{-1}) \tag{7.2}$$

where $\mathbf{D}$ is the set of matrices

$$D = \begin{bmatrix} I & 0 \\ 0 & D_2 \end{bmatrix}$$

with D_2 diagonal, and the partition of D comformably to the partition of G_γ.

The matrix D_2 that gives the minimum in equation (7.2) can be calculated using a converging algorithm (Packard et al. [47]). The upper bound will usually be very close, so that for practical use we will assume that the upper bound is equal to $\mu(G_\gamma)$. In this way we can find the bound γ for the norm of the model error.

7.3   An upper bound using unstructured noise sets

In the case where we consider unstructured noise sets, we have

$$Q_t \;\; = \;\; Q_{ut} \quad \text{is a full matrix with } |Q_{ij}| \le 1 \text{ for all } i,j$$

$$V \;\; = \;\; \begin{bmatrix} 0 & W \\ I & 0 \end{bmatrix}$$

$$G_W \;\; = \;\; W_{\Delta 1} \, S(\begin{bmatrix} \tilde{N} & 0 \\ 0 & \tilde{M} \end{bmatrix}, \hat{H}, T, V) \, W_{\Delta 2} \;\; = \;\; \begin{bmatrix} G_{W11} & G_{W12} \\ G_{W21} & G_{W22} \end{bmatrix}$$

$$W_{\Delta 1} \, \Delta_t \, W_{\Delta 2} \;\; = \;\; S(G_W, Q_t)$$

and we consider the unstructured noise set $\mathbf{Q}_u$ with matrices Q_u satisfying:

$$\| \, Q_u \, \| \;\; \le \;\; \sqrt{(p+q)p} \;\; \overset{def}{=} \;\; \lambda_0$$

(where p denotes the number of inputs and q denotes the number of outputs of the process). Then $Q_{ut} = Q_t \in \mathbf{Q}_u$.

If $\| \, G_{W22} \, \| \; < \; \lambda_0^{-1}$, so if the signal-to-noise ration is sufficiently large, we can find a bound for the weighted matrix norm:

$$\| \, W_{\Delta 1} \, \Delta_t \, W_{\Delta 1} \, \| \;\; = \;\; \| \, S(G_W, Q_{ut}) \, \| \;\; \le \;\; \max_{\|Q_u\| \le \lambda_0} \| \, S(G_W, Q_u) \, \|$$

The problem is a two full-block μ-problem. For this problem we can calculate the exact bound using μ-techniques. There holds:

$$\max_{\|Q_*\|\leq\lambda_0} \| S(G_W, Q_u) \| = \gamma$$

iff

$$\min_{d>0} \sigma_{max}\left(\begin{bmatrix} G_{W11}\,\gamma_u^{-1} & d\,\lambda_0^{1/2}\,G_{W12}\,\gamma_u^{-1/2} \\ d^{-1}\,\lambda_0^{1/2}\,G_{W21}\,\gamma_u^{-1/2} & \lambda_0\,G_{W22} \end{bmatrix}\right) = 1$$

An important advantage of this bound, compared with the bound for the structured set, is that we can compute the bound exactly (Packard et al. [5]). This can be important if we like to use convex optimization techniques, as will be discussed in chapter 9. Another advantage is that we only have to optimize over one scalar d instead of $(p+q)p$ scalars from the diagonal matrix D_2 in the case of a structured set.

A disadvantage is that the bound can become very conservative, compared to the approach with a structured set.
Another disadvantage is that we still need an iteration (although it is only over a scalar d). We will show that for the SISO-case we can find the bound in an explicit way without iteration. To find this explicit bound we will transform the expression

$$\Delta = S(G_W, Q_u) = G_{W11} + G_{W12}\,Q_u\,(\,I - G_{W22}\,Q_u\,)^{-1}G_{W21}$$

which is not affine in the variable Q_u, into an affine form.
We will now give four lemmas that provide the basic tools to transform the afore mentioned expression into an affine form (This will be formulated in Theorem 7.1) :

Lemma 7.1:
Suppose that an orthogonal matrix $E(z) \in I\!\!RL_\infty$ that is partitioned as

$$E(z) = \begin{bmatrix} E_{11}(z) & E_{12}(z) \\ E_{21}(z) & E_{22}(z) \end{bmatrix} \tag{7.3}$$

where we assume that E_{12} and E_{21} are square and invertible for $z \in \mathcal{C}$. Define the matrix

$$E_-(z) = \begin{bmatrix} E_{22}^*(z) & E_{12}^*(z) \\ E_{21}^*(z) & E_{11}^*(z) \end{bmatrix} \tag{7.4}$$

Then there will hold:

$$\begin{aligned} E^*(z)E(z) &= I \;, \quad z \in \mathcal{C} \\ E_-^*(z)E_-(z) &= I \;, \quad z \in \mathcal{C} \end{aligned} \tag{7.5}$$

$$S(E, E_-) = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$$

$$\hspace{8cm} (7.6)$$

$$S(E_-, E) = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$$

$$\square$$

Proof:
Property (7.5) is obvious because E and E_- are orthogonal. Property (7.6) is easily found by substitution of (7.3) and (7.4) in the starproduct.

An important consequence of the last property stated in Lemma 7.1 is now given in the following lemma.

Lemma 7.2:
Let E and E_- satisfy the conditions of Lemma 7.1.
Then for all $X, Y \in \mathbb{RL}_\infty$ there holds :

$$Y = F_l(E, X) \quad \Leftrightarrow \quad X = F_l(E_-, Y)$$

$$\square$$

Proof:

$$F_l(E_-, Y) = F_l\Big(E_-, F_l(E, Y)\Big) = F_l\Big(S(E_-, E), Y)\Big) =$$

$$= F_l\left(\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}, X\right) = X$$

and

$$F_l(E, X) = F_l\Big(E, F_l(E_-, X)\Big) = F_l\Big(S(E, E_-), Y)\Big) =$$

$$= F_l\left(\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}, Y\right) = Y$$

Lemma 7.3:
Let $E(z) \in R\mathbb{L}_\infty$ be an orthogonal matrix, so $E(z)E^*(z) = I$.
Further let $X(z)$ be any matrix in $R\mathbb{L}_\infty$ and define $Y = S(E, X)$.
Then

$$\|Y\| \leq 1 \quad \Leftrightarrow \quad \|X\| \leq 1 \qquad \text{for all } z \in C$$

$$\square$$

The proof of this Lemma is in appendix A4.

We now introduce two important orthogonal matrices J and J_- as follows:
Let $A(z) \in R\!L_\infty$ with $\| A(z) \|_\infty < 1$.
Define the matrices

$$J(A) = \begin{bmatrix} A^*(z) & \tilde{\Lambda}^*(z) \\ \Lambda^*(z) & -\Lambda^*(z)\, A(z)\, \tilde{\Lambda}^{-1}(z) \end{bmatrix} \tag{7.7}$$

$$\begin{aligned} J_-(A) &= \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} \left(J(A)^{-1} \right) \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} = \\ &= \begin{bmatrix} -(\tilde{\Lambda}^*(z))^{-1}\, A^*(z)\, \Lambda(z) & \tilde{\Lambda}(z) \\ \Lambda(z) & A(z) \end{bmatrix} \end{aligned} \tag{7.8}$$

where $\Lambda(z)$ and $\tilde{\Lambda}(z)$ are in $R\!H_\infty$ with

$$\Lambda(z)\,\Lambda^*(z) = (I - A(z)\, A^*(z)) \quad \text{and}$$

$$\tilde{\Lambda}^*(z)\tilde{\Lambda}(z) = (I - A^*(z)\, A(z))$$

The matrices $\Lambda(z)$ and $\tilde{\Lambda}(z)$ can be found with a spectral factorization (Anderson ??). The matrices $J(A)$ and $J_-(A)$ are both orthogonal.

Lemma 7.4:
Let $B(z) \in R\!L_\infty$ be partioned as:

$$B(z) = \begin{bmatrix} B_{11}(z) & B_{12}(z) \\ B_{21}(z) & B_{22}(z) \end{bmatrix}$$

and define

$$C(z) = \begin{bmatrix} C_{11}(z) & C_{12}(z) \\ C_{21}(z) & C_{22}(z) \end{bmatrix} = S\left(B(z),\, J\big(B_{22}(z)\big) \right)$$

with $J(\cdot)$ as in equation (7.7).
Then

$$C_{22}(z) = 0$$

$\square$

Proof:

$$
\begin{aligned}
C_{22} &= J_{22} + J_{21} B_{22} (I - J_{11} B_{22})^{-1} J_{12} \\
&= -\Lambda^* B_{22} \tilde{\Lambda}^{-1} + \Lambda^* B_{22} (I - B_{22}^* B_{22})^{-1} \tilde{\Lambda}^* = \\
&= \Lambda^* \left(-B_{22} (\tilde{\Lambda}^* \tilde{\Lambda})^{-1} + B_{22} (I - B_{22}^* B_{22})^{-1} \right) \tilde{\Lambda}^* \\
&= 0
\end{aligned}
$$

$\square$

With the aid of the Lemmas 7.1 to 7.4 we can transform the expression

$$
\Delta = S(G_W, Q_u) = G_{W11} + G_{W12} Q_u (I - G_{W22} Q_u)^{-1} G_{W21}
$$

into an affine form in the variable Q_u.
This is the formulated in Theorem 7.1:

Theorem 7.1:
Define the matrix X_u as:

$$
X_u = S(J_-(\lambda_0 G_{W22}), \lambda_0^{-1} Q_u)
$$

for some $(p+q) \times p$ matrix Q_u and with $J(\cdot)$ as in equation (7.7).
Further define the matrix

$$
\begin{aligned}
G_J(z) &= \begin{bmatrix} G_{J1} & G_{J2} \\ G_{J3} & G_{J4} \end{bmatrix} = \\
&= S \left(\begin{bmatrix} G_{W11} & \lambda_0^{1/2} G_{W12} \\ \lambda_0^{1/2} G_{W21} & \lambda_0 G_{W22} \end{bmatrix}, J(\lambda_0 G_{W22}) \right)
\end{aligned}
$$

Then

$$
\max_{\|Q_*\| \le \lambda_0} \| S(G_W , Q_u) \|_\infty = \max_{\|X_*\| \le 1} \| G_{J1} + G_{J2} X_u G_{J3} \|_\infty
$$

$\square$

Proof:
First from Lemma 7.2 we find: $Q_u = \lambda_0 S(J, X_u)$. Since J is an orthogonal
matrix, Lemma 7.3 yields

$$
\| X_u \| \le 1 \Leftrightarrow \| Q_u \| \le \lambda_0
$$

So now we have:

$$
\max_{\|Q_*\| \le \lambda_0} \| S(G_W , Q_u) \|_\infty =
$$

$$= \max_{\|\lambda_0^{-1} Q_u\| \leq 1} \left\| S \left(\begin{bmatrix} G_{W11} & \lambda_0^{1/2} G_{W12} \\ \lambda_0^{1/2} G_{W21} & \lambda_0 G_{W22} \end{bmatrix}, \lambda_0^{-1} Q_u \right) \right\|_\infty =$$

$$= \max_{\|X_u\| \leq 1} \| S(G_J, X_u) \|_\infty$$

From Lemma 7.3 we infer that by choosing $\lambda_0 J(G_{W22})$ we find $G_{J4} = 0$ and so

$$\max_{\|X_u\| \leq 1} \| S(G_J, X_u) \|_\infty = \max_{\|X_u\| \leq 1} \| G_{J1} + G_{J2} X_u G_{J3} \|_\infty$$

$\square$

A next bound is then given by:

$$\gamma = \max_{\|X_u\| \leq 1} \| G_{J1} + G_{J2} X_u G_{J3} \| \leq \bar{\sigma}(G_{J1}) + \bar{\sigma}(G_{J2}) \bar{\sigma}(G_{J3})$$

In the case where we deal with a SISO process P_t, the matrix G_W will be a 2×3 matrix, and G_{J1} and G_{J2} will be scalar functions. The last inequality then becomes an equality:

$$\boxed{\gamma = \sigma_{max}(G_{J1}) + \sigma_{max}(G_{J2}) \sigma_{max}(G_{J3})}$$

In that case we can calculate the bound without iteration (Van den Boom [57]).

7.4 Norms in the frequency domain

In the previous sections it was shown that an upper bound γ (either γ for the structured or for the unstructured case) can be derived for the matrix norm of the model error for each frequency $z \in \mathcal{C}$. This results in a frequency dependent bound $\gamma(z)$, with

$$\gamma(z) = \max_{Q(z) \in \mathbf{Q}} \| S(G_W(z), Q(z)) \| , \quad z \in \mathcal{C} \tag{7.9}$$

where $\| \cdot \|$ denotes the matrix norm as defined in section 1.1 .
Now we formulate four model error bound criteria $J(\gamma(z))$:
   Two norm:

$$J_2(\gamma(z)) = \|\gamma(z)\|_2 \tag{7.10}$$

Infinity norm:

$$J_\infty(\gamma(z)) = \|\gamma(z)\|_\infty \qquad (7.11)$$

Weighted sum of norms:

$$J_s(\gamma(z)) = \alpha\|\gamma(z)\|_2 + \beta\|\gamma(z)\|_\infty \qquad (7.12)$$

where α and β are positive real scalar constants.

Square root of the weighted sum of the squared norms:

$$J_{ss}(\gamma(z)) = \left(\alpha^2\|\gamma(z)\|_2^2 + \beta^2\|\gamma(z)\|_\infty^2\right)^{\frac{1}{2}} \qquad (7.13)$$

where α and β are positive real scalar constants.

Maximum of weighted norms:

$$J_{max}(\gamma(z)) = \max\left(\alpha\|\gamma(z)\|_2 , \beta\|\gamma(z)\|_\infty\right) \qquad (7.14)$$

where α and β are positive real scalar constants.

All these criteria $J(\gamma)$ are norms because they satisfy the four conditions:

1. $J(\gamma) \geq 0$

2. $J(\gamma) = 0$ iff $\gamma = 0$

3. $J(a\,\gamma) = |a|\,J(\gamma)$ where a is a scalar constant.

4. $J(\gamma_1 + \gamma_2) \leq J(\gamma_1) + J(\gamma_2)$

We can use these criteria to formulate an optimization problem, leading to an optimal nominal model in the sense of one of the norms mentioned above. This will be discussed in the next chapter.

Computation in the sampled frequency domain :
In this chapter we considered the infinity-norm and the two-norm in the frequency domain. In practice, however, we will compute the infinity-norm and the two-norm for the sampled frequency domain.
If we have chosen the length of the observation interval $2N$ large enough, so $N \gg l_\Delta$, where l_Δ is the length of the impulse response $h_\Delta(k)$ of the model error $\Delta(z)$, then we know from section 1.1 that

$$\lim_{N\to\infty} \| \Delta(z) \|_{s\infty} = \| \Delta(z) \|_\infty$$

$$\lim_{N\to\infty} \| \Delta(z) \|_{s2} = \| \Delta(z) \|_2$$

In applications we therefore expect that the error we make will be negligible if the observation interval is large enough (note that we assume $\Delta(z) \in R\!I\!H_\infty$). Even if l_Δ is not finite and we assume that the tail-contribution of $h_\Delta(k)$ is bounded by

$$\sum_{k=l_\Delta}^{\infty} h_\Delta(k) \leq \epsilon_\Delta$$

we derived in section 1.1 that the error in the two-norm and the infinity-norm will be smaller than ϵ_Δ.

8

Parametrization and Optimality

8.1 Introduction

We derived upper bounds for various model error criteria for a given model. Now we are going to parametrize the model coprime factors and we will optimize the parameters in the sense of minimizing the model error bound. We will have a discussion about quantities, except for the model-parameters, that influence the final optimal model, like applied controllers, factorizations and weighting filters.

We parametrize the model coprime factors $N(z)$ and $M(z)$ using an $n \times 1$ parameter-vector θ belonging to a set Θ. The corresponding dependence of the coprime factors of θ will be denoted as $N(\theta, z)$ and $M(\theta, z)$. Also, the upper bound $\gamma(z)$, as defined in chapter 7, will be denoted as $\gamma(\theta, z)$ and so we obtain the criterion $J(\gamma(\theta, z))$, that we wish to minimize over all θ in Θ:

$$\boxed{\min_{\theta \in \Theta} J\left(\gamma(\theta, z)\right)} \qquad\qquad (8.1)$$

For the criterion J we can make several choices, like J_∞, J_2, J_s or J_{ss}, as defined in section 7.4. ote that the criterion J depends on the model coprime factors, the controller coprime factors and the weighting filters:

$$J = J\Big(N(\theta, z),\, M(\theta, z),\, N_C(z),\, M_C(z),\, W_{\Delta 1}(z),\, W_{\Delta 2}(z) \Big)$$

<u>Remarks:</u>

- The first item we discuss is the choice of the model error structure. In chapter 5 we showed that the stabilizing controller indicates which model error structures can be used. If we have $K(z) = 0$ we can choose for an additive model error, input/output multiplicative model error or a coprime factor model error structure with fixed $\Delta_M(z) = 0$. For $K(z) = \infty$ we can choose one of the reverse type model error structures or a coprime factor model error structure with fixed $\Delta_N = 0$. For $K(z) \neq 0$ and $K(z) \neq \infty$ we deal with a coprime factor model error structure.

  The only choice that may be left is that of a left or a right coprime factorization of model and controller. In this thesis we assume that this choice is already made and so we consider the model error structure to be fixed.

- For the model $P(\theta, z)$ we have to select a coprime factorization. If we made a choice for a particular coprime factorization we obtain a parametrization of the model coprime factors $N(\theta, z)$ and $M(\theta, z)$. We will discuss this topic in section 8.2.

- Also for the stabilizing controller $K(z)$ we have to choose a coprime factorization. Further, if we have more than one stabilizing controller, we have to make a choice between these stabilizing controllers. We will discuss this topic in section 8.3.

- Finally we select the weighting filters $W_{\Delta 1}(z)$ and $W_{\Delta 2}(z)$ to be motivated by control requirements. We will discuss this topic in section 8.4.

8.2   Parametrization of the model coprime factors

In this section we will shortly discuss the choice of parametrization of the model coprime factors and the consequences of this choice. The controller

and its coprime factorization are assumed to be fixed. We like to obtain a nominal model that is suited for the design of a robust controller. We look for a nominal model with a low order, an easy structure. For the sake of a reliable and fast estimation of the parameters, it is recommendable to use a minimum number of parameters.

Before we can parametrize the model coprime factors we have to fix the coprime factorization of the model. For this, we introduce, as in section 5.2, a bijective mapping π:

$$\pi\,(P(z))\;=\;[\,N(z)\;\;M(z)\,]$$

where N and M constitute a specific coprime factorization of P.

Choice of coprime factorization of the model

We will work out what happens if we compare the resulting true model error for two different choices of left coprime factorization:

Consider two different coprime factorizations of a model $P(z) \in \mathbf{P}$:

$$P(z)\;=\;M_1^{-1}(z)\,N_1(z)$$

$$P(z)\;=\;M_2^{-1}(z)\,N_2(z)$$

and a specific coprime factorization of a system $\tilde{P}(z) \in \tilde{\mathbf{P}}$:

$$\tilde{P}(z)\;=\;\tilde{M}^{-1}(z)\,\tilde{N}(z)$$

This results in the model errors

$$[\,\Delta_N(z)\;\;\Delta_M(z)\,]_1\;=\;[\,\tilde{N}-N_1\;\;\tilde{M}-M_1\,]$$

$$[\,\Delta_N(z)\;\;\Delta_M(z)\,]_2\;=\;[\,\tilde{N}-N_2\;\;\tilde{M}-M_2\,]$$

In chapter 5 we proved that there will exist a unimodular matrix $A(z)$ such that:

$$N_2(z)\;=\;A(z)\,N_1(z)$$

$$M_2(z)\;=\;A(z)\,M_1(z)$$

We will show that the model errors for these choices of coprime factorizations are related as:

$$[\,\Delta_N(z)\;\;\Delta_M(z)\,]_2\;=\;A(z)[\,\Delta_N(z)\;\;\Delta_M(z)\,]_1$$

From equation (5.3) we deduce that

$$[\ \Delta_N(z)\ \ \Delta_M(z)\ ]_1\ =\ \bar{\Delta}_1(z)[\ M_C(z)\ \ -N_C(z)\ ]\ =$$

$$=\ \Big(M_1(z)\tilde{Y}(z)\ -\ N_1(z)\tilde{U}(z)\Big)\tilde{W}^{-1}(z)[\ M_C(z)\ \ -N_C(z)\ ]$$

where $\tilde{W}(z)\ =\ (\ M_C(z)\tilde{U}(z)\ +\ N_C(z)\tilde{Y}(z)\ )$ and where $\tilde{Y}(z)$ and $\tilde{U}(z)$ are choosen

$$\begin{bmatrix} \tilde{Y}(z) \\ \tilde{U}(z) \end{bmatrix}\ =\ \begin{bmatrix} Y(z) \\ U(z) \end{bmatrix}\ +\ \tilde{F}(z)$$

for an element $\tilde{F}(z)$ in a set $\mathbf{F}_s$ or a set $\mathbf{F}_u$ (The sets $\mathbf{F}_s$ and $\mathbf{F}_u$ are introduced in section 4). The signal $\tilde{W}(z)$ is independent of the model coprime factors (see example section 6.4).

The true model error for the second choice of the left coprime factors is given by

$$[\ \Delta_N(z)\ \ \Delta_M(z)\ ]_2\ =$$

$$=\ \Big(M_2(z)\tilde{Y}(z)\ -\ N_2(z)\tilde{U}(z)\Big)\tilde{W}^{-1}(z)[\ M_C(z)\ \ -N_C(z)\ ]\ =$$

$$=\ \Big(A(z)M_1(z)\tilde{Y}(z)\ -\ A(z)N_1(z)\tilde{U}(z)\Big)\tilde{W}^{-1}(z)[M_C(z)\ \ -N_C(z)]\ =$$

$$=\ A(z)[\ \Delta_N(z)\ \ \Delta_M(z)\ ]_1$$

where $\tilde{Y}$, $\tilde{U}$ and $\tilde{W}$ are the same matrices as mentioned above.
We see that by scaling the left coprime factors by a matrix $A(z)$ we also scale the model error by matrix $A(z)$. So, the choice of another coprime factorization can be viewed as choosing another weighting filter $W_{\Delta 1}(z)$. This result is illustrated in figure 8.1. For the right coprime factor case the situation is depicted in figure 8.2.

    Remark: In the case of a normalized coprime factorization the model co-prime factors $N(z)$ and $M(z)$ are fixed, up to an inner or coninner matrix multiplication.
This inner matrix, however, has no influence on the matrix-norm of the model error $\tilde{\Delta}(z)$.

Parametrization: ·
Suppose we have a canonical parameter set for all models $P(z)$. Then we can introduce a mapping ϕ from the parameter set Θ to the model set $\mathbf{P}$, such that

$$\phi\,(\,\Theta\,)\ =\ \mathbf{P}$$
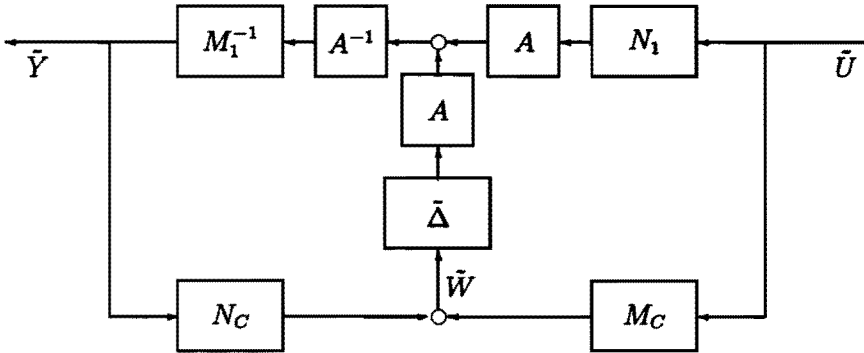
Figure 8.1: Model error scheme with scaled left coprime factors
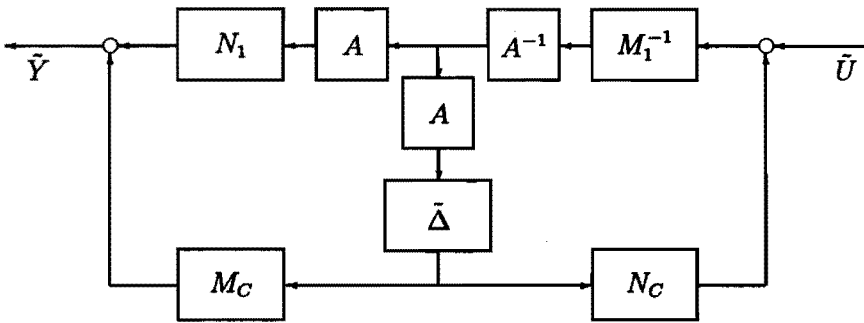
Figure 8.2: Model error scheme with scaled right coprime factors

If we choose the mapping

$$\phi(\theta) = P(\theta, z)$$

in such a way that for all $P(z) \in \mathbf{P}$ there is only one parameter vector $\theta \in \Theta$ such that $P(z) = P(\theta, z)$, we have a bijective mapping from Θ to $\mathbf{P}$ and each $\theta \in \Theta$ gives a canonical representation for $P(\theta, z) \in \mathbf{P}$. Next consider the bijective map π defining the coprime factorizations of elements $P \in \mathbf{P}$. By defining the mapping

$$\psi = \pi \phi$$

we get a parametrization of the model coprime factors with

$$\psi(\theta) = [N(\theta, z) \ M(\theta, z)]$$

Since both π and ψ are bijective, also ψ is bijective and so

$$\psi\,(\,\Theta\,) \,=\, \mathbf{S} \quad \text{and} \quad \psi^{-1}\,(\,\mathbf{S}\,) \,=\, \Theta$$

The relation between the mappings ϕ, π and ψ is visualized in figure 8.3.
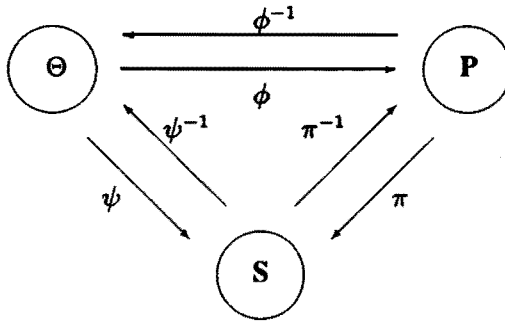
Figure 8.3: Bijective mappings

In section 7.4 we introduced different model error criteria $J\big(\gamma(\theta,z)\big)$. This leads to an optimization problem where we wish to minimize over all $\theta \in \Theta$, as in equation (8.1). The properties of the optimization of $\theta \in \Theta$ to minimize a model error criterion J is depending on the choice of the parametrization of $N(\theta,z)$ and $M(\theta,z)$.

We can choose the parametrization of N and M to be linear in the parameters θ. This can be done by 'fixing the poles' of the model coprime factors N and M. The main reason for doing so is that for this linear parametrization the optimization problem becomes a convex optimization problem. This will be discussed in chapter 9.

There are situations where we can not fix the poles of the model coprime factors. For example, if we consider an additive model error structure and we make $M(z) = I$, then we do not like to fix the poles of $N(z) = P(z)$. Another example is where we like to work with normalized coprime factors N and M. Models with a normalized coprime factorization are of special interest in control theory, because the problem of robust stability with these models has explicit solutions and controllers can be computed in a relatively simple way (McFarlane [43], Glover and McFarlane [22]). In those cases, however, the optimization problem (minimizing the funtional J over the parameters space Θ) will become very 'messy'. A parametrization using normalized coprime factors is shortly discussed in Van den Boom ([56]).

Of course many other parametrizations are possible, but we will restrict ourselves to the linear parametrization as mentioned above.

8.3   The stabilizing controller and its coprime factorization

Suppose we have a set of stabilizing controllers $\{ K_1(z) , K_2(z) , .., K_k(z) \}$. For all these controllers we can compute various left or right coprime factorizations which lead to different model error structures. In Chapter 5 we fixed the coprime factorization of the controller to be normalized. In this section we will release this choice and study the influence of different choices of coprime factorizations. Finally we consider what to do if we have more than one stabilizing controller.

Choice of the coprime factorization of the controller:
We will first consider the case where we have only one fixed controller $K(z)$. The first important remark is that the choice between a left or right coprime factorization of the controller is determined by the choice of coprime factorization of the model. If we choose a left coprime factorization for the model we have to choose a left coprime factorization for the controller as well, because the R-parametrization should result in a consistent scheme. The same holds for the right coprime case.

Now suppose that we choose a left coprime factorization for the model. Then we can compute different left coprime factorizations $K(z) = M_C^{-1}(z) N_C(z)$ for the controller. We will compare the resulting true model error for two different choices of left coprime factorization of the controller: Suppose these factorizations of the controller are given, i.e:

$$K(z) = M_{C1}^{-1}(z) N_{C1}(z)$$

$$K(z) = M_{C2}^{-1}(z) N_{C2}(z)$$

This results in the model errors

$$[\Delta_N(z) \ \Delta_M(z)]_1 = \tilde{\Delta}_1(z) [M_{C1}(z) \ -N_{C1}(z)]$$

$$[\Delta_N(z) \ \Delta_M(z)]_2 = \tilde{\Delta}_2(z) [M_{C2}(z) \ -N_{C2}(z)]$$

There will exist a unimodular matrix $B(z)$ such that:

$$N_{C2}(z) = B(z) N_{C1}(z)$$

$$M_{C2}(z) \;=\; B(z)\,M_{C1}(z)$$

We will show that the model errors for both choices of coprime factorizations are the same.

For the first choice of the left coprime factors $N_{C1}(z)$ and $M_{C1}(z)$ we can derive

$$[\,\Delta_N(z)\;\;\Delta_M(z)\,]_1 \;=\; \tilde{\Delta}_1(z)[\,M_{C1}(z)\;\;-N_{C1}(z)\,] \;=$$

$$=\; \tilde{X}(z)\left(\,N_{C1}(z)\tilde{Y}(z) + M_{C1}(z)\tilde{U}(z)\,\right)^{-1}[\,M_{C1}(z)\;\;-N_{C1}(z)\,]$$

where $\tilde{X}(z) \;=\; \left(\,M\,\tilde{Y}(z) - N\,\tilde{U}(z)\,\right)$ is independent of the controller coprime factors (see example section 6.4) and $\tilde{U}$ and $\tilde{Y}$ are introduced in the previous section.

For the second choice of coprime factorization the true model error is given by

$$[\,\Delta_N(z)\;\;\Delta_M(z)\,]_2 \;=\; \tilde{\Delta}_2(z)[\,M_{C2}(z)\;\;-N_{C2}(z)\,] \;=$$

$$=\; \tilde{X}(z)\left(\,N_{C2}(z)\tilde{Y}(z) + M_{C2}(z)\tilde{U}(z)\,\right)^{-1}[\,M_{C2}(z)\;\;-N_{C2}(z)\,] \;=$$

$$=\; \tilde{X}\left(\,B(z)\,N_{C1}(z)\,\tilde{Y}(z) + B(z)\,M_{C1}(z)\,\tilde{U}(z)\,\right)^{-1}\cdot$$

$$[\,B(z)\,M_{C1}(z)\;\;-B(z)\,N_{C1}(z)\,] \;=$$

$$=\; \tilde{X}(z)\left(\,N_{C1}(z)\tilde{Y}(z) + M_{C1}(z)\tilde{U}(z)\,\right)^{-1}\cdot$$

$$B^{-1}(z)\,B(z)\,[\,M_{C1}(z)\;\;-N_{C1}(z)\,] \;=$$

$$=\; \tilde{X}(z)\left(\,N_{C1}(z)\tilde{Y}(z) + M_{C1}(z)\tilde{U}(z)\,\right)^{-1}[\,M_{C1}(z)\;\;-N_{C1}(z)\,] \;=$$

$$=\; [\,\Delta_N(z)\;\;\Delta_M(z)\,]_1$$

and so the model error is not influenced by the choice of the coprime factorization of the stabilizing controller. This is illustrated by figure 8.4. Figure 8.5 illustrates the right coprime case.

Remark: We derived that the final model error is not influenced by the choice of coprime factorization of the controller. As was already motivated at the end of chapter 5, we will choose a **normalized coprime** Factorization For **the controller**. In that case there holds that the norm of the model error is
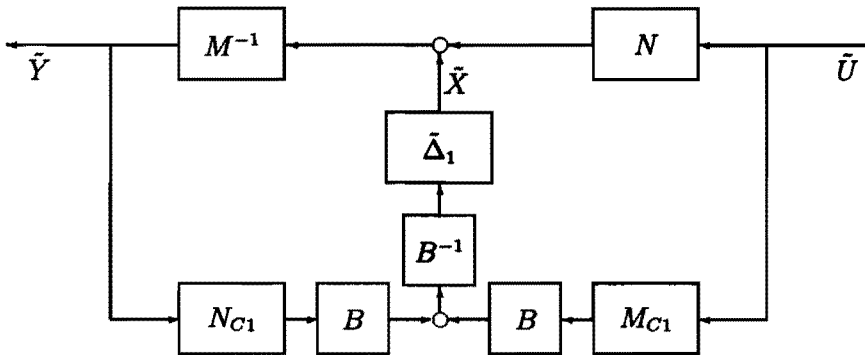
Figure 8.4: Model error scheme with scaled left coprime factors
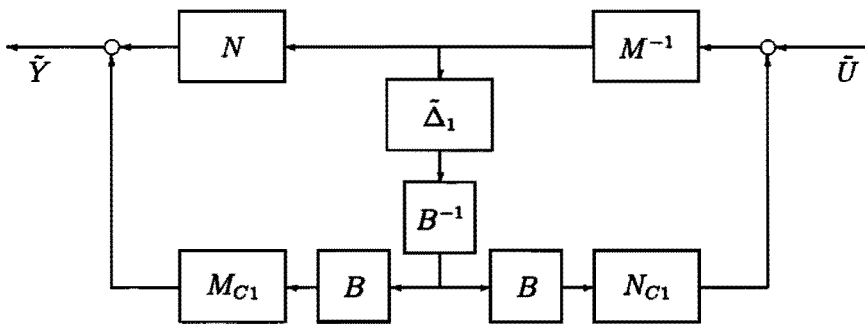
Figure 8.5: Model error scheme with scaled right coprime factors

equal to the norm of the stable function $\tilde{\Delta}(z)$ in figure 8.1/8.5.

Choice between various stabilizing controllers:

If we only know one stabilizing controller, we only have to choose for either a left or a right coprime factor description, which is determined by the choice of coprime factorization of the model.

If we have the access to more stabilizing controllers we can choose the controller that results in a minimum error bound.

For example, suppose we have a model set with left coprime factors $N(\theta, z)$ and $M(\theta, z)$ and we have a normalized left coprime factorization of the i-th controller: $K_i(z) = M_{Ci}^{-1}(z)N_{Ci}(z)$, then we search for the true process coprime factors in the set

$$\tilde{N}(z) = N(\theta, z) + \Delta_N(z)$$

$$\bar{M}(z) = M(\theta, z) - \Delta_M(z)$$

where

$$\Delta_N(z) = \Delta(z) M_{Ci}(z)$$

$$\Delta_M(z) = -\Delta(z) N_{Ci}(z)$$

This means that, although $[\Delta_N(z) \quad \Delta_M(z)]$ will be in different spaces for different choice of controller $K_i(z)$, we can compare the bounds for the various controllers a postriori, and we can select the one that results in a minimum bound for the model error.

8.4   Weighting filters and optimization criterion

We like the weighting filters W_{Δ_1} and $W_{\Delta 2}$ to be motivated by control requirements and robustness constraints. This implies that W_{Δ_1} and $W_{\Delta 2}$ will become large for some frequency-range and in some input/output 'direction', if we have high control demands for that particular frequency-range or input/output 'direction'.

Before the choice of the weighting filters is discussed we will mention the most important ingredients of the problem so far.

Nominal model: The nominal model $P(\theta, z)$ is parametrized by parameter vector $\theta \in \Theta$. The nominal model coprime factors become $[N(\theta, z) \quad M(\theta, z)] = \pi\left(P(\theta, z)\right)$. The parametrization is already discussed in section 8.2.

Controllers: For the identification and control of the process we use 3 different types of controllers:

1. Controller K_{exp} for the experiments, as discussed in chapter 2.
2. Controller K_{ana} for the analysis, as discussed in chapter 5.
3. Controller K_{opt} for optimal performance, as discussed in section 1.3.

The controllers K_{exp} and K_{ana} may be the same controllers.

Bounded model error: The discrepancy between the nominal model $P(\theta, z)$ and the true process $P_t(z)$, is described by the model error $\Delta_t(z)$. The true process $P_t(z)$ will be in a set $\bar{\mathbf{P}}$, which is given as:

$$\bar{\mathbf{P}} = \left\{ \bar{P} = F_u\left(\Gamma_P(z), \tilde{\Delta}(z)\right) \right\}$$

where F_u denotes the upper linear fractional transformation and the matrix $\Gamma_P(z)$ is given by

$$\Gamma_P(z) \;=\; S\left(\begin{bmatrix} N(\theta,z) & 0 \\ 0 & M(\theta,z) \end{bmatrix}, \hat{H} \right)$$

Here the matrix $\hat{H}$ is defined as in section 6.4 and $[\,N(\theta,z)\;\;M(\theta,z)\,]$ are the coprime factors of the nominal model $P(\theta,z)$. We will assume the model error $\tilde{\Delta}(z)$ to be in the set

$$\tilde{\Delta} \;=\; \left\{ \tilde{\Delta}(z) \mid \; \| \, W_{\Delta 1}(z)\,\tilde{\Delta}(z)\, W_{\Delta 2}(z) \,\|_\infty \leq \gamma \right\}$$

When we start with the modelling of an unknown process using identification the afore mentioned basic ingredients will be unknown in general. By doing identification and designing a controller we will gradually find a nominal model with a bounded model error and an optimal controller for this model. The main problem is that the three aspects (model, model error and controller) are related very closely. To see that, we consider the three problems: The H_∞-identification problem, the robust control problem and the model error set determination problem. We will discuss these problems and we will show that we can solve the identification/control problem in an iterative procedure, involving the three mentioned problems:

Problem 1: H_∞-Identification problem:

> **Given:**
> > - Data set $\{\,u\,,\,y\,,\,r\,\}$ and the disturbance bounding filters W_d, W_e.
> > - Stabilizing controllers K_{exp} and K_{ana}.
> > - Weighting filters $W_{\Delta 1}$ and $W_{\Delta 2}$ and upper bound γ.
> > - Model set with model coprime factors $[\,M(\theta,z),\,N(\theta,z)\,]$, $\theta \in \Theta$.
>
> **Find :**
> > - Optimal parameter vector $\theta_{opt} \in \Theta$ such that $\gamma(\theta) = J_\infty\big(\gamma(\theta,z)\big)$ is minimized.
>
> **Result:**
> > - Optimal model $P(\theta_{opt}, z)$ and $\gamma_{opt} = \gamma(\theta_{opt})$.

Problem 2: Robust control problem:

Given:

- Nominal model $P(\theta_0, z)$ and Γ_0.
- A set $\tilde{\mathbf{P}} = \{\ \tilde{P} = F_u(\Gamma_{P_0}, \tilde{\Delta}),\ \| W_{\Delta 1} \tilde{\Delta} W_{\Delta 2} \|_\infty \leq \gamma_0 \}$.
- Weighting filters W_{p1}, W_{p2}, W_{p3} and W_{p4} for the performance constraint.

Find :

- Controller K_{opt} such that

$$\eta(K) = \max_{\tilde{P}} \|\ \begin{bmatrix} W_{p1} \\ W_{p2}K \end{bmatrix} (I + \tilde{P}K)^{-1} \begin{bmatrix} W_{p3} & W_{p4}\tilde{P} \end{bmatrix} \|_\infty$$

is minimized, and K_{opt} stabilizes all $\tilde{P}(z) \in \tilde{\mathbf{P}}$.

Result:

- Optimal controller $K_{opt}(z)$ and $\eta_{opt} = \eta(K_{opt})$.

Problem 3: Model error set determination problem:

Given:

- Nominal model $P(\theta_0, z)$ and Γ_o.
- Controller $K_0(z)$.
- Weighting filters W_{p1}, W_{p2}, W_{p3} and W_{p4} for the performance constraint.

Find :

- Weighting filters $W_{\Delta 1}(z)$, $W_{\Delta 2}(z)$ such that

$$\| W_{\Delta 1}(z)\,\tilde{\Delta}(z)\,W_{\Delta 2}(z) \|_\infty \leq 1 \quad \Longrightarrow$$

$$\|\ \begin{bmatrix} W_{p1} \\ W_{p2}K_0 \end{bmatrix} (I + \tilde{P}K_0)^{-1} \begin{bmatrix} W_{p3} & W_{p4}\tilde{P} \end{bmatrix} \|_\infty \leq 1$$

and $K_0(z)$ stabilizes $\tilde{P}(z) = F_u(\Gamma_0(z), \tilde{\Delta}(z))$

Result:

- Weighting filters $W_{\Delta 1}(z)$ and $W_{\Delta 2}(z)$.

Problem 1 is discussed in the previous chapters.

Problem 2 is a robust control problem that can be solved using standard H_∞ or μ-techniques (Doyle et al. [13], Doyle [11] and Balas et al. [5]).

How to solve the third problem will be discussed now:

Consider the performance function

$$\begin{bmatrix} W_{p1} \\ W_{p2} K_0 \end{bmatrix} (I + \tilde{P} K_0)^{-1} \begin{bmatrix} W_{p3} & W_{p4} \tilde{P} \end{bmatrix} = S(K_0, \Sigma, \tilde{P})$$

where Σ is given by:

$$\Sigma = \begin{bmatrix} 0 & W_{p3} & 0 & I \\ 0 & W_{p1} W_{p3} & 0 & W_{p1} \\ W_{p2} & 0 & 0 & 0 \\ -I & 0 & W_{p4} & 0 \end{bmatrix}$$

For the nominal model P_0 we compute

$$\alpha = \| S(K_0, \Sigma, P_0) \|_\infty$$

If $\alpha > 1$ the performance constraints are too tight and no controller can do the job for the nominal model P_0. Therefore we will assume that $\alpha < 1$, so that there some 'room' to allow perturbations on the model P_0.

Now suppose Γ_0 is the structure matrix for the model P_0, then $\tilde{P} = S(\Gamma_0, \tilde{\Delta})$ and we get the performance constraint:

$$\| S(K_0, \Sigma, \tilde{P}) \|_\infty = \| S(K_0, \Sigma, \Gamma_0, \tilde{\Delta}) \|_\infty \leq 1 \qquad (8.2)$$

We define the matrix

$$R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = S(K_0, \Sigma, \Gamma_0)$$

and the performance constraint turns into

$$\| S(R, \tilde{\Delta}) \|_\infty \leq 1$$

Thus we are concerned about the magnitude of $\sigma_{max}\big(\tilde{\Delta}(z)\big)$ with z on the unit circle. We can transform this constraint into an affine form in $\tilde{\Delta}(z)$. Therefore define:

$$\begin{aligned} R_{12}^l &= (R_{12}^T R_{12})^{-1} R_{12}^T \\ C_0 &= (R_{22} R_{12}^l R_{11} - R_{21}) \\ A &= C_0^T (C_0 C_0^T)^{-1} R_{22} R_{12}^l \\ T &= \begin{bmatrix} T_0 & T_1 \\ T_2 & T_3 \end{bmatrix} = S\big(J_-(A), R \big) \end{aligned}$$

where $J_-(A)$ is defined as in equation (7.8). Finally we have assumed that the matrix R_{12} has full column-rank and the matrix $\big(R_{22} R_{12}^l R_{11} - R_{21}\big)$ has full row-rank.

With these definitions we can formulate theorem 8.1:

Theorem 8.1:
Suppose the matrix R_{12} has full column-rank and the matrix $\left(R_{22}R_{12}^l R_{11} - R_{21}\right)$ has full row-rank.
Let $T(z)$ be defined as above.
Then there holds

$$\| S(K_0, \Sigma, \Gamma_0, \tilde{\Delta}) \|_\infty \leq 1 \quad \Leftrightarrow \quad \| T_0 + T_1 \tilde{\Delta} T_2 \|_\infty \leq 1$$

$$\square$$

The proof of this theorem is in appendix A5.

The theorem claims that for some fixed P_0 and K_0 we can transform the performance constraint into a constraint affine in the model error. Except for the extra term T_0 this constraint yields two weighting filters T_1 and T_2 for the optimization criterion of the model error. We will assume that for the nominal model (so $\tilde{\Delta}(z) = 0$) the constraint will be satisfied, so $\| T_0(z) \|_\infty \leq 1$. Now we can approximate the criterion by

$$\| T_1 \Delta_t T_2 \| \leq 1 - \| T_0(z) \| \quad \text{for all } z \in C$$

If $\| T_0(z) \|_\infty \ll 1$ the approximation will be very close. Next we apply an inner-outer factorization on $T_1(z)$ and $T_2(z)$:

$$T_1(z) = T_{1i}(z) T_{1o}(z) \quad \text{and} \quad T_2(z) = T_{2o}(z) T_{2i}(z)$$

where T_{1i} is an inner function, T_{2i} is a coinner function and $T_{1o} \, T_{2o}$ are outer functions. Define

$$W_{\Delta 1} = T_{1o}(1 - \| T_0(z) \|)^{-1} \quad \text{and} \quad W_{\Delta 2} = T_{2o}$$

so that the filters $W_{\Delta 1}$ and $W_{\Delta 2}$ become square and stable. We get the criterion:

$$\| W_{\Delta 1} \tilde{\Delta} W_{\Delta 2} \|_\infty = \| T_1 \tilde{\Delta} T_2 (1 - \| T_0(z) \|)^{-1} \|_\infty \leq 1 \qquad (8.3)$$

We derived filters $W_{\Delta 1}$ and $W_{\Delta 2}$ that give an indication in which frequency-ranges and in which input/output 'directions' the minimization of the model error must be emphasized to satisfy the control demands.

We solved the three mentioned problems and we find that the results of the first problem are necessary to solve the second problem, the results of the second problem are necessary to solve the third problem, and finally, the results of the third problem are necessary to solve the first problem.
One way to solve all three problems is to use an iterative scheme:

Iterative algorithm:
We consider the 3 problems:

1.) The H_∞ identification problem

$$W_{\Delta 1}, \; W_{\Delta 2} \;\; \Longrightarrow \;\; P_0, \; \gamma_0$$

2.) The robust control problem

$$W_{\Delta 1}, \; W_{\Delta 2}, \; P_0, \; \gamma_0 \;\; \Longrightarrow \;\; K_0$$

3.) The model error set determination problem

$$P_0, \; K_0 \;\; \Longrightarrow \;\; W_{\Delta 1}, \; W_{\Delta 2}$$

We can start with a first estimate for filters $W_{\Delta 1}(z)$ and $W_{\Delta 2}(z)$ (For example $W_{\Delta 1}(z) = I$ and $W_{\Delta 2}(z) = I$), and then solve the H_∞ identification problem. The optimal nominal model P_0 and the model error bound γ that are found can be used as the input parameters for the robust control problem, where we will find an optimal controller K_0. For this K_0 and P_0 we can compute the model error set that satisfies the performance criterion and we obtain an new estimate for the filters $W_{\Delta 1}(z)$ and $W_{\Delta 2}(z)$.
We can repeat these 3 steps untill we are satisfied about the performance properties of the model with controller and model error bounds.
If the performance requirements cannot be met we have to do measurements (perhaps with improved sensors and actuators), or we have to relax the performance requirements.
If the performance requirements are met very easily and we do not want to tighten these requirements, we have 'room' to optimize another identification criterion, like J_2, J_s, J_{ss} or J_{max}, as long as after the identification it will hold that $J_\infty < 1$, as in equation (8.3).
   It is very difficult to get insight in the convergence of the iterative scheme. The three sub-problems are well defined optimization problems and will converge. The interaction between model, model error bound and optimal controller are very complex. However, to our believe the iterative scheme will converge if we deal with reasonable circumstances, like a good signal to noise ratio during the experiments, a wel chosen model set, and control requirements that are not too high.

PART C: OPTIMIZATION

9

Convex optimization

Part C is about optimization. The aim of identification is to be optimal in the sense of some model error criterion. This usually result in an optimization procedure using an iterative algorithm. During this iteration we hope to reach the optimal parameter values. For the majority of parametrizations and optimization algorithms there is no theoretical evidence that the algorithm will ever reach the optimum or will at least get very close to the optimum. In this Chapter we will therefore choose a special parametrization, the linear parametrization of the model coprime factors, i.e. a parametrization where the coprime factors are chosen linear in the parameter vector θ. We will show that, under some conditions, the optimization criterion will become convex over θ for this choice of parametrization.

Two optimization algorithms, the cutting-plane algorithm and the ellipsoid algorithm, will shortly be discussed. We choose for these algorithms because these algorithms are easily implemented and they provide simple stopping criteria that guarantee us that the optimum is found up to a prespcified accuracy. Furthermore the algorithms only make use of subgradients rather than gradients.

A subgradient function of four model error criteria J will be derived.

9.1   Linear parametrization

In this section the following parametrization of the coprime factors $N(\theta, z)$ and $M(\theta, z)$ will be considered:

$$\begin{bmatrix} N(z) & 0 \\ 0 & M(z) \end{bmatrix} = \begin{bmatrix} N_0(z) & 0 \\ 0 & M_0(z) \end{bmatrix} + \sum_{i=1}^{n} \theta_i \begin{bmatrix} N_i(z) & 0 \\ 0 & M_i(z) \end{bmatrix} \quad (9.1)$$

where $\theta \stackrel{def}{=} [\theta_1 \ \theta_2 \ ... \ \theta_n]^T$. In this way $N(\theta, z)$ and $M(\theta, z)$ are both affine in the parameter θ. To guarantee coprimeness of $N(\theta, z)$ and $M(\theta, z)$ we will restrict Θ to those parameter vectors θ that makes these factors coprime.

Remark: In equation 9.1 we choose the block-structure for the parametrization so that we can use the results for left coprime factors as well as for right coprime factors.

Examples:
ARMA-parametrization with fixed AR parameters:
Let:

$$N(\theta, z) = C(z)^{-1} B(\theta, z) \quad \text{and} \quad M(\theta, z) = C(z)^{-1} A(\theta, z)$$

where we denote

$$A(\theta, z) = A_0(\theta) + A_1(\theta) z^{-1} + .. + A_n(\theta) z^{-n}$$

$$B(\theta, z) = B_0(\theta) + B_1(\theta) z^{-1} + .. + B_n(\theta) z^{-n}$$

and we fix

$$C(z) = C_0 + C_1 z^{-1} + .. + C_n z^{-n}$$

such that $\det C(z) \neq 0$ for $\text{Re}(z) > 1$.

For the parametrization of $A_i(\theta)$ and $B_i(\theta)$ we can choose various canonical forms, as long as $A_i(\theta)$ and $B_i(\theta)$ are linear in the parameter θ (see Kailath [34]). So $N(\theta, z)$ and $M(\theta, z)$ are modelled with ARMA parameters where we fixed the AR parameters. (They are both affine in the parameter θ.)

In the case of an additive model error, we have $M = I$, and so we choose $A_i = C_i$. In the case of an input multiplicative model error, we have $N = I$, and so we choose $B_i = C_i$.

To avoid a trivial optimal solution $\theta_{opt} = 0$ in the case where $N \neq I$ and $M \neq I$, we must guarantee that

$$\text{rank} \left\{ \begin{bmatrix} N_0 & 0 \\ 0 & M_0 \end{bmatrix} \right\} \geq \dim(M_0)$$

We can achieve that for example by fixing $M_0 = I$.

Parametrization with orthogonal functions:

Next we mention the parametrization with orthogonal functions (Heuberger [31]). For a SISO-model we choose

$$\left. \begin{array}{ll} N_0(z) = 0 & N_i(z) = f_{1i}(z) \\ M_0(z) = 1 & M_i(z) = f_{2i}(z) \end{array} \right\} \quad i = 1, \ldots, n$$

The functions $f_{1i}(z)$ and $f_{2i}(z)$ are from two separate orthogonal sets with

$$\oint f_{1i}(z) f_{1j}(z) \, dz = \left\{ \begin{array}{ll} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{array} \right. \quad i, j > 0$$

$$\oint f_{2i}(z) f_{2j}(z) \, dz = \left\{ \begin{array}{ll} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{array} \right. \quad i, j, > 0$$

An advantage of the parametrization of the model coprime factors with orthogonal functions is that the optimization will take place in an orthogonal domain, and may be numerically better conditioned.

Parametrization with Laguerre polynomials:

In the SISO case we can choose a parametrization with Laguerre polynomials (Wahlberg [65]), which is a special case of parametrization with orthogonal functions:

$$N_0 = 0 \quad M_0 = 1$$

$$N_i(z) = \frac{\sqrt{1 - a_1^2}}{z - a_1} \left(\frac{1 - a_1 z}{z - a_1} \right)^{i-1} \quad \text{where} \quad -1 < a_1 < 1 \text{ and } 1 \leq i \leq n$$

and

$$M_i(z) = \frac{\sqrt{1 - a_2^2}}{z - a_2} \left(\frac{1 - a_2 z}{z - a_2} \right)^{i-1} \quad \text{where} \quad -1 < a_2 < 1 \text{ and } 1 \leq i \leq n$$

$N(\theta, z)$ and $M(\theta, z)$ will be Laguerre-models.

9.2   Convexity of the criterion

In this section we consider the convexity. If we like to obtain convexity we have to extend the parameter set Θ to a convex set. We choose for this extented convex set the entire $I\!\!R^n$ and so we optimize over $\theta \in I\!\!R^n$. Later, after the optimization is done we can check if the optimal θ^* is in the set Θ, and so if $N(\theta^*, z)$ and $M(\theta^*, z)$ are coprime and if $P(\theta^*, z)$ is stabilized by the controller $K(z)$.

We show that, if we choose a linear parametrization and if the matrix $\hat{H}(z)$ (as introduced in section 6.4) satisfies some conditions, we can transform the optimization procedure for particular model error structures into a convex optimization problem. This is formulated in the following theorem:

<u>Theorem 9.1:</u>

Suppose $\hat{H}$ can be partioned as:

$$\hat{H} = \begin{bmatrix} \hat{H}_{11} & \hat{H}_{12} & \hat{H}_{13} & \hat{H}_{14} \\ \hat{H}_{21} & \hat{H}_{22} & \hat{H}_{23} & \hat{H}_{24} \\ \hat{H}_{31} & \hat{H}_{32} & \hat{H}_{33} & \hat{H}_{34} \\ \hat{H}_{41} & \hat{H}_{42} & \hat{H}_{43} & \hat{H}_{44} \end{bmatrix}$$

Now let:

$$1.) \quad \begin{bmatrix} \hat{H}_{11} & \hat{H}_{12} \\ \hat{H}_{21} & \hat{H}_{22} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$2.) \quad \begin{bmatrix} \hat{H}_{41} & \hat{H}_{42} \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} \hat{H}_{14} \\ \hat{H}_{24} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$3.) \quad \begin{bmatrix} N & 0 \\ 0 & M \end{bmatrix} = \begin{bmatrix} N_0 & 0 \\ 0 & M_0 \end{bmatrix} + \sum_{i=1}^{n} \theta_i \begin{bmatrix} N_i & 0 \\ 0 & M_i \end{bmatrix}$$

Then there will exist functions $L_0(Q, z)$ and $L_i(Q, z)$ such that $S(G_W(\theta), Q)$ can be written as

$$S(G_W(\theta), Q) = L_0(Q, z) + \sum_{i=1}^{n} \theta_i L_i(Q, z) \tag{9.2}$$

and the function

$$J\big(\gamma(\theta, z)\big) = J(\max_{Q \in \mathbf{Q}} \|S(G_W(\theta), Q)\|) =$$

$$= J \left(\max_{Q \in \mathbf{Q}} \| L_0(Q, z) + \sum_{i=1}^{n} \theta_i L_i(Q, z) \| \right) \tag{9.3}$$

is a convex function over θ, where $J(\gamma)$ is a norm.

□

The proof of this theorem is in appendix A6.

From the example in section 6.4 we can see that for the left coprime set up corresponding to figure 6.4 the matrix $\hat{H}$ satisfies the conditions 1 and 2. Also for the right coprime set up we can derive that the conditions 1 and 2 will be satisfied.

9.3 Cutting-plane and ellipsoid algorithms

In the previous section it was shown that under certain conditions $J(\gamma(\theta, z))$ is a convex function over all $\theta \in \mathbb{R}^n$. We can solve the problem of minimization by making use of special convex optimization algorithms. We will shortly discuss two methods, the cutting plane algorithm ([6],[9]) and the ellipsoid algorithm ([6],[25]) , which both use stopping criteria that provide an optimum to a certain accuracy.

The gradients with respect to θ for the function $J(\gamma(\theta, z))$ can be discontinuous, because the gradient of the function $\gamma(\theta, z)$ is discontinuous. We are interested in the mentioned methods, because both algorithms make use of subgradients rather than gradients.

A function $g(\hat{\theta})$ is called a subgradient in $\hat{\theta}$ if for all θ there holds

$$J\big((\gamma(\theta, z)\big) \geq J\big((\gamma(\hat{\theta}, z)\big) + g^T(\hat{\theta})(\theta - \hat{\theta}) \tag{9.4}$$

We will derive a subgradient for all criteria J in the next section.

First the cutting plane algorithm will be discussed. Suppose, for k parameter vectors $\theta_{(1)}, .., \theta_{(k)}$ we have calculated the norms $J(\theta_{(1)}), .., J(\theta_{(k)})$ and for the subgradients $g_{(1)}, .., g_{(k)}$ (where $g_{(1)}$ denotes $g(\theta_{(1)})$). Now we know:

$$J(\theta) \geq J(\theta_{(i)}) + g_{(i)}^T(\theta - \theta_{(i)}) \quad \text{for all } i = 1, .., k$$

So for all θ we have

$$J(\theta) \geq \max_{i=1,..,k} \left(J(\theta_{(i)}) + g_{(i)}^T(\theta - \theta_{(i)}) \right)$$

and so for θ_{opt} there will hold

$$J(\theta_{opt}) \geq \min_{\theta} \max_{i=1,..,k} \left(J(\theta_{(i)}) + g_{(i)}^T(\theta - \theta_{(i)}) \right) \overset{def}{=} L_k$$

To calculate this minimum, define the following matrices:

$$w = \begin{bmatrix} \theta \\ L_k \end{bmatrix} \quad A = \begin{bmatrix} g_{(1)}^T & -1 \\ \vdots & \vdots \\ g_{(k)}^T & -1 \end{bmatrix}$$

$$b = \begin{bmatrix} g_{(1)}^T \theta_{(1)} - J(\theta_{(1)}) \\ \vdots \\ g_{(k)}^T \theta_{(k)} - J(\theta_{(k)}) \end{bmatrix} \quad c = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

and solve the linear programming problem

$$w_k^* = \arg \min_{Aw \leq b} c^T w$$

where we denote $w_k^* \in I\!\!R^{n+1}$ as the optimizer of this problem and define $\theta_k^* = [I \ 0] w_k^*$ and $L_k^* = [0 \ 1] w_k^*$.
Define $U_k^* = J(\theta_k^*)$ and we find the following bounds

$$U_k^* \geq J(\theta_{opt}) \geq L_k^*$$

If the interval size $(U_k^* - L_k^*)$ is sufficiently small, we can take θ_k^* as a reasonable estimate for θ_{opt}. If not, we define $\theta_{(k+1)} = \theta_k^*$ and put an extra row to the matrix A and an extra element to vector b. We can calculate a new upper bound U_{k+1}^* and lower bounds L_{k+1}^*, where there will hold:

$$U_k^* \geq U_{k+1}^* \geq J(\theta_{opt}) \geq L_{k+1}^* \geq L_k^*$$

We can iterate until the stopping criterion is sufficiently small.

Next we discuss the ellipsoid algorithm:
Suppose we know that the minimum θ_{opt} is in the ellipsoid

$$E_0 = \{\theta \mid (\theta - \theta_{(0)})^T A_0^{-1}(\theta - \theta_{(0)}) \leq 1\}$$

where A_0 is some square and non-singular matrix in $I\!\!R^{n \times n}$.
We can calculate a subgradient $g_{(0)}$ in $\theta_{(0)}$. $J(\theta)$ is convex, so θ_{opt} will be in the half-plane

$$H_0 = \{\theta \mid g_{(0)}^T(\theta - \theta_{(0)}) \leq 0\}$$

We now know that the half-ellipsoid $H_0 \cap E_0$ will contain the minimizer θ_{opt}. We can construct a new ellipsoid

$$E_1 = \{\theta \mid (\theta - \theta_{(1)})^T A_1^{-1} (\theta - \theta_{(1)}) \leq 1\}$$

such that $E_1 \supseteq (H_0 \cap E_0)$ and where the volume of E_1 is less than the volume of E_0.

This can be repeated and we get an iterative procedure. Now suppose after k iterations we know that the minimum θ_{opt} is in the ellipsoid

$$E_k = \{\theta \mid (\theta - \theta_{(k)})^T A_k^{-1} (\theta - \theta_{(k)}) \leq 1\}$$

Let $g_{(k)}$ be the subgradient in $\theta_{(k)}$ and define the normalized subgradient as

$$\tilde{g}_{(k)} = \frac{g_{(k)}}{\sqrt{g_{(k)}^T A_k g_{(k)}}}$$

Then we can calculate the $(k+1)$th ellipsoid

$$E_{k+1} = \{\theta \mid (\theta - \theta_{(k+1)})^T A_{k+1}^{-1} (\theta - \theta_{(k+1)}) \leq 1\}$$

where $\theta_{(k+1)}$ and A_{k+1} are given by

$$\theta_{(k+1)} = \theta_{(k)} - \frac{A_k \tilde{g}_{(k)}}{n+1}$$

$$A_{k+1} = \frac{n^2}{n^2 - 1}(A_k - \frac{2}{n-1} A_k \tilde{g}_{(k)} \tilde{g}_{(k)}^T A_k)$$

The volume of these subsequential ellipsoids will decrease to 0 for $k \to \infty$. Further there holds

$$J(\theta_{(k)}) - J(\theta_{opt}) \leq \sqrt{g_{(k)}^T A_k g_{(k)}}$$

Remark: For both methods we assume that $(\gamma(\theta, z))$ is a convex function over θ for every z. Convexity could be lost if the approximation for

$$\gamma(\theta, z) = \max_{Q \in \mathbf{Q}} \| S(G_W(\theta, z), Q(z)) \|$$

using the D-iteration (see section 7.2), is not good enough.

In the cutting-plane algorithm we might find that $U_i < L_i$. In that case a new set of starting vectors $\theta_{(1)}, .., \theta_{(k)}$ should be choosen in the neighbourhood of the last-found θ_i, and we can restart the optimization.

The ellipsoid algorithm, however, will converge, but we might find a final θ^*, that is not really the optimal θ_{opt}. If one of the ellipsoids E_i was chosen too small (so θ_{opt} is not in E_i) then θ_j, $j > i$ tends to the boundary of the ellipsoid E_i. We can check that by increasing the size of the final ellipsoid, and see if a new optimization will move out of the final ellipsoid.

9.4 Subgradients

In this section we will derive subgradients for all criteria J which were introduced in section 7.4. In particular we consider the infinity-norm and the 2-norm in the sampled frequency domain, so $z \in \Omega$.

By definition, a subgradient $g(\hat{\theta})$ will satisfy:

$$J\big((\gamma(\theta, z)\big) \geq J\big((\gamma(\hat{\theta}, z)\big) + g^T(\hat{\theta})(\theta - \hat{\theta}) \quad \text{for all } \theta \in \mathbb{R}^n$$

Before we give the subgradients we have to introduce some definitions:

$$Q_0(\hat{\theta}, z) \stackrel{def}{=} \arg\left(\max_{Q \in \mathbf{Q}} \|L_0(Q, z) + \sum_{i=1}^{n} \hat{\theta}_i L_i(Q, z)\|\right)$$

$$z_0(\hat{\theta}) \stackrel{def}{=} \arg\left(\max_{z \in \Omega} \|\gamma(\hat{\theta}, z)\|\right)$$

where $L_0(Q, z)$ and $L_i(Q, z)$ are the functions from theorem 9.1.
Suppose

$$L_0(Q_0, z) + \sum_{i=1}^{n} \hat{\theta}_i L_i(Q_0, z) = u(z)\,\Sigma(z)\,v^*(z)$$

is the singular value decomposition for all $z \in \Omega$, and let $u_0(z)$ and $v_0(z)$ be the column of resp. $u(z)$ and $v(z)$ corresponding to the maximum singular value $\sigma_{max}(z)$. Then

$$\|\gamma(\hat{\theta}, z)\|_\infty = u_0^*(z_0)\Big(L_0(Q_0, z_0) + \sum_{i=1}^{n} \hat{\theta}_i L_i(Q_0, z_0)\Big) v_0(z_0)$$

Finally define:

$$p(z) \stackrel{def}{=} \begin{bmatrix} u_0(z)^* L_1(Q_0, z) v_0(z) \\ u_0(z)^* L_2(Q_0, z) v_0(z) \\ \vdots \\ u_0(z)^* L_n(Q_0, z) v_0(z) \end{bmatrix}$$

With these definitions we can give subgradients for all four criteria J:

$$g_2 = \left(\sum_{z \in \Omega} p(z) \gamma(\hat{\theta}, z) \right) / J_2\left(\gamma(\hat{\theta}, z)\right)$$

$$g_\infty = p(z_0)$$

$$g_s = \alpha g_2 + \beta g_\infty$$

$$g_{ss} = \left(\alpha^2 \sum_{z \in \Omega} p(z) \gamma(\hat{\theta}, z) + \beta^2 p(z_0) \gamma(\hat{\theta}, z_0) \right) / J_{ss}\left(\gamma(\hat{\theta}, z)\right)$$

$$g_{max} = \begin{cases} \alpha g_2 & \text{for } \alpha J_2 \geq \beta J_\infty \\ \beta g_\infty & \text{for } \alpha J_2 < \beta J_\infty \end{cases}$$

The derivation of these subgradients is given in appendix A7.

10

Evaluation of the model

In the previous chapters we discussed the optimization of a model error criterion. When we have found the optimimum for this criterion

$$J_{opt} = J(\theta_{opt}) = \min_{\theta} J(\theta)$$

we need to evaluate this result.

In the first place we have made a lot of assumptions in the beginning of the identification procedure. We need to indicate if these assumptions are realistic or not. In the second place we like to know if the optimized criterion is really optimal.

10.1 Evalution of the assumptions

We will have to check whether the assumptions have proven to be true during the identification:

Finite length of impulse responses.

Let the observation interval for the measurements be of length $2N$. In chapter 2 we made the following assumptions:

$$l_1 \ll 2N \qquad l_2 \ll 2N \qquad l_\Delta \ll 2N$$

where

$\quad l_1 \quad$ is the impulse response length of $\quad (I + P_t K_{exp})^{-1} P_t$
$\quad l_2 \quad$ is the impulse response length of $\quad (I + P_t K_{exp})^{-1}$
$\quad l_\Delta \quad$ is the impulse response length of $\quad \Delta_t$

The function K_{exp} is known, because this is the controller we use during the experiments. The true process $P_t(z)$ and the true model error $\Delta_t(z)$, corresponding to the final optimal model $P_{opt}(z)$ are (of course) unknown, so it is very difficult to varify these assumptions afterwards. In this section we will see what happens in the case where the optimal model $P_{opt}(z)$ is close to the true process $P_t(z)$ (We mean close in the sense that the lengths l_1 and l_2 do not change too much).

If $P_{opt}(z)$ is close to $P_t(z)$ we can substitute $P_{opt}(z)$ for $P_t(z)$ and check if impulse responses corresponding to the functions

$$(I + P_{opt} K_{exp})^{-1} P_{opt} \quad \text{and} \quad (I + P_{opt} K_{exp})^{-1}$$

have both a length much smaller than $2N$. If this is true, we may assume that also for the true lengths l_1 and l_2 there holds:

$$l_1 \ll 2N \qquad \text{and} \qquad l_2 \ll 2N$$

It is more difficult, if not impossible to check the assumption on the length l_Δ of the impulse response of the true model error. The maximum that will be feasible is to get an indication whether the assumption about l_Δ is realistic or not. Compared to the check on l_1 and l_2 we get an extra problem, for a simple substitution of P_{opt} for $P_t(z)$ will not be enough. For example, if we consider an additive model error and we substitute $P_{opt}(z)$ for $P_t(z)$ we will be left with an estimated model error $\Delta(z) = P_{opt} - P_{opt} = 0$. A better way to get an idea of the impuls response length of $\Delta_t(z)$ will be by considering the impulse response of P_{opt} and P_t separately. If the length of impulse response of $P_{opt}(z)$ will be much smaller than $2N$ and we assume that $P_{opt}(z)$ is close to $P_t(z)$ we may assume that $P_t(z)$ will also have an impulse response with length smaller than $2N$, without considering $P_{opt} = P_t$. This will result in a (non-zero) model error with an impulse response length that is smaller than $2N$.

Next we consider the more general case with a coprime factor model error structure. Suppose the coprime factors corresponding to the optimal model are $N_{opt}(z)$ and $M_{opt}(z)$. The model error can now be given by:

$$\Delta_t = (M_{opt} P_t - N_{opt})(N_C P_t + M_C)^{-1}$$

Substituting $M_{opt}^{-1} N_{opt}$ for $P_t(z)$ gives us a model error $\Delta_t(z) = 0$. Therefore we better look to two different terms

$$T_1 = M_{opt} P_t (N_C P_t + M_C)^{-1} \quad \text{and}$$

$$T_2 = N_{opt} (N_C P_t + M_C)^{-1}$$

resulting

$$\Delta_t(z) = T_1(z) - T_2(z)$$

We substitute P_{opt} for P_t in $T_2(z)$ and we compute the length of the impulse response of the estimate

$$\hat{T}_2 = N_{opt} (N_C P_{opt} + M_C)^{-1}$$

We can reason in the same way as for the additive model error as follows: if the coprime factors of P_{opt} is close to the coprime factors of P_t and we find that the impulse response length of $\hat{T}_2$ is much smaller than $2N$, we can assume that the impulse response lengths of T_1 and T_2 and thus of Δ_t will be smaller than $2N$.

Of course this reasoning completely depends on the assumption that P_{opt} is close to P_t. If this is true we can get an indication if the assumptions are realistic with the described method. If we find that the length of the impulse response of $\hat{T}_2$ is larger than $2N$ we can nearly be certain that the assumption on l_Δ may be too severe.

If P_{opt} is not close to P_t we can not get an indication whether the the assumption on l_Δ is realistic or not.

The procedure as sketched above is a very inaccurate method. It may happen that we find that the estimated $\hat{l}_1$, $\hat{l}_2$ and $\hat{l}_\Delta$ do not satisfy the conditions, where the true l_1, l_2 and l_Δ are all smaller than $2N$. Also the opposite case may happen. The checks on the assumptions have to be accepted with some reserve.

Stabilizing controller:

We assumed the model to be stabilized by the controller $K(z)$ (see section 5.1). We have to check if our optimal model P_{opt} is indeed stabilized by this controller. If $P_{opt} = M_{opt}^{-1} N_{opt}$ is not stabilized by the controller $K(z)$ we have the problem that we can not guarantee that factorization of the true process $P_t = (M_{opt} - \Delta_t N_C)^{-1}(N_{opt} + \Delta_t M_C)$ will be a coprime factorization of the true process.

In that case we will have to restart the identification procedure with a new coprime factorization of the true process

$$N_t(z) = N_0(z) + \Delta_{0,t}(z)\, M_c(z) \quad \text{and} \quad M_t(z) = M_0(z) - \Delta_{0,t}(z)\, N_c(z)$$

where N_0 and M_0 are the coprime factors of a fixed system that is stabilized by the controller $K(z)$. For this new coprime factorization of the true process, the model error is given by

$$\Delta_N(z) = N_t(z) - N(z) = N_0(z) - N(z) + \Delta_{0,t}(z)\, M_c(z) \quad \text{and}$$
$$\Delta_M(z) = M_t(z) - M(z) = M_0(z) - M(z) - \Delta_{0,t}(z)\, N_c(z)$$

For this parametrization however we will find that

$$\| [\, \Delta_N(z) \quad \Delta_M(z) \,] \| \neq \| \Delta_{0,t} \| \quad \text{for } z \in \mathcal{C}$$

In other words, the matrix norm of the model error $[\, \Delta_N \quad \Delta_M \,]$ will not be equal to the norm of the stable function $\Delta_{0,t}(z)$. The expressions as derived in the chapters 6-10 have to be adapted. The essence of the minimization will remain the same, e.g. a linear parametrization will still lead to a convex optimization problem. The condition that P_{opt} needs to be stabilized by controller $K(z)$ is not necessary any more. The choice, however, of the system $M_0^{-1} N_0$ will be crucial to get acceptable bounds.

Finally we have to check if the factors N_{opt} and M_{opt} are really coprime. This will generically be the case. If the factors M_{opt} and N_{opt} have right-half plane zeros that are very close to each other and so if the coprimeness is very 'weak' we may have chosen a too high model order.

10.2 Evalution of the model error criterion

If no assumptions are violated, we can evaluate if the model error criterion $J(\theta)$ is as small as we aimed it to be. If this is small enough we can design a controller on the basis of this optimal nominal model with its error bounds. If the theoretical result and the actual implementation of the optimal controller will match the specifications, the job is done. We already discussed this procedure in chapter 8.

As was mentioned in section 8.4, the iterative scheme might not converge in some cases. Restricting ourselves to the identification part of the iterative procedure, there are several remedies that may improve the procedure:

- We can choose another model set **P**. The model can be parametrized in another way, the order may be increased, and for MIMO systems we may choose another structure.

- We can do another experiment to extend the data set. We may choose another reference signal that excites the process in some 'critical directions or frequency ranges'. In this way we can increase the information about the process.

- We can try to find better bounding functions for the noise. This can be done either by extending the dataset, by more intensive study of the sensors and actuators, or by improving or replacing the sensors and actuators.

Approximate modelling and H_∞-identification :
Finally a remark should be made about the concept of approximate modelling by H_∞-identification. The distance between a model $P(z)$ and the true process $P_t(z)$ can be measured in many ways. In this thesis we focussed on the minimization of an upper bound for the H_∞ norm of the model error. This implies that we choose twice for a worst-case approach. For each frequency we have a worst case bound over the scaled noise $Q_{(z)}$, given the upper bound $\gamma(\theta, z)$. Next we have a worst-case criterion over all frequencies (H_∞). The choice for a worst-case approach is only motivated by robust control. If we wish to have an approximate model that is optimal in the sense that the model matches the average behaviour of the true process as good as possible we should not use a worst-case approach, but e.g. minimize the H_2 norm of an estimate of the model error.
On the other hand, the H_∞ minimization of an upper bound of the model error may still lead to a very good approximate model in sense of a good average behaviour (Liebregts [37]).

PART D: EVALUATION

11

Case study

In this chapter a case study is presented. We consider the identification of a laboratory process, using the techniques as derived in this thesis. The process under study is the so called 'Water vessel process'. The water vessel process is a smoothly non-linear process with two inputs and two outputs. A SISO-version of this process has been studied by Liebregts (,[36],[37]) using the identification methods as described in Chapter 3. The actuators and sensors that are applied to the process are the same as in [36], [37], and are studied extensively in these reports.

11.1 The water vessel process

Process description
The water vessel process is a laboratory process, that consists of four vessels, two roller pumps as actuators and two level sensors as shown in the configuration of figure 11.1. The water vessels are positioned in two layers of two vessels each. Two roller pumps pump water from a reservoir tank into
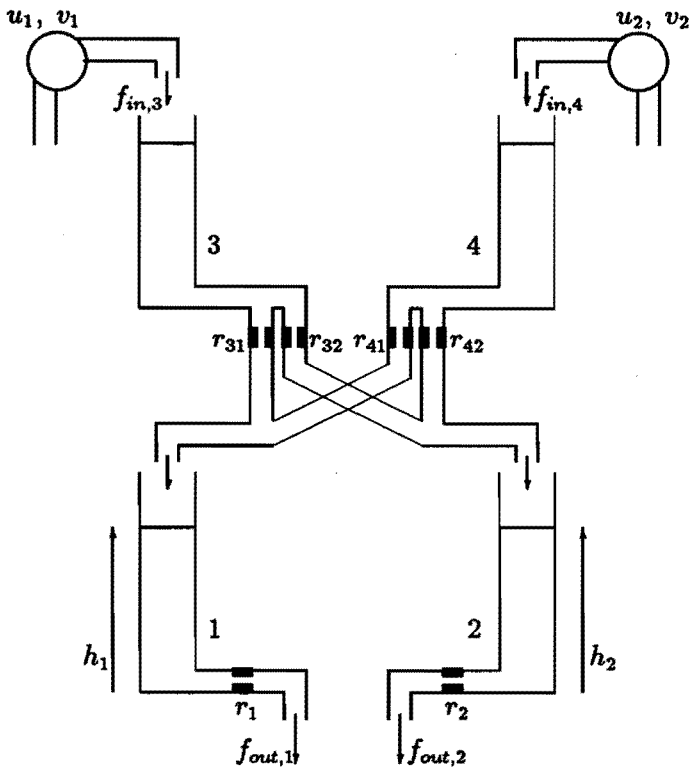
Figure 11.1: The water vessel process

the top vessels. The input voltages of the two pumps are denoted by v_1 and v_2. The rotation speed of the rollers of the pumps are denoted by u_1 and u_2. From each of the top vessels (3 and 4) the water flows via restrictions (r_{31}, r_{32}, r_{41}, r_{42}) into the two lower vessels. From the lower vessels (1 and 2) the water flows via restrictions (r_1, r_2) back into a reservoir. The rotation speeds u_1 and u_2 of the roller pumps will be considered as the inputs of the system. The flow, caused by the pump, is proportional to the rotation speed of the pump. Since the transfer of the control voltage v_i to the rotation speed u_i of the pump shows a saturation effect, we prefer the rotation speeds u_i rather than the control voltages of the roller pumps as input signals. The heights h_1 and h_2 of the water level in the two lower vessels will be considered as outputs. This two input two output system is smoothly non linear and we will study the dynamic behaviour in some working point. A complete modelling of this process using physical laws has shown to be very difficult (Liebregts [37]). Therefore we will only consider physical modelling in a very simple

and approximate manner.

Simplified model:
Consider figure 11.2 where water is flowing in a piece of a tube from R to S. We consider the density of the water to be constant $\rho = 1$, and the pressure p_1 to be equal to p_2. The area of the tube at R is denoted by A_1 and at S by A_2. The height of the water level in tube at R is denoted by h_1 and at S by h_2. The water velocity in the tube at R is denoted by w_1 and at S by w_2.
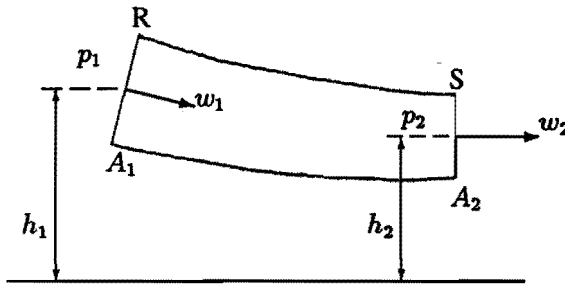
Figure 11.2: Water-flow in a tube

We assume the flow to be laminar and the velocity uniform. Two contributions to change in energy are considered:
(1). The energy per second due to kinetic energy:

$$\Delta E_k = \frac{1}{2}(A_1\,w_1\,\rho)\,w_1^2 - \frac{1}{2}(A_2\,w_2\,\rho)\,w_2^2 \tag{11.1}$$

(2). The energy per second due to potential energy:

$$\Delta E_p = h_1\,(A_1\,w_1\,\rho)\,g - h_2\,(A_2\,w_2\,\rho)\,g \tag{11.2}$$

According to the laws of hydrodynamics:

$$\Delta E_k + \Delta E_p = 0 \tag{11.3}$$

Finally the in coming mass must be equal to the out going mass (mass balance), which yields

$$A_1\,w_1\,\rho = A_2\,w_2\,\rho \tag{11.4}$$

Combining equations (11.1), (11.2), (11.3) and (11.4) results in:

$$\frac{1}{2}\rho w_1^2 + h_1 \rho g = \frac{1}{2}\rho w_2^2 + h_2 \rho g \tag{11.5}$$

Consider figure 11.3 with a single water vessel.
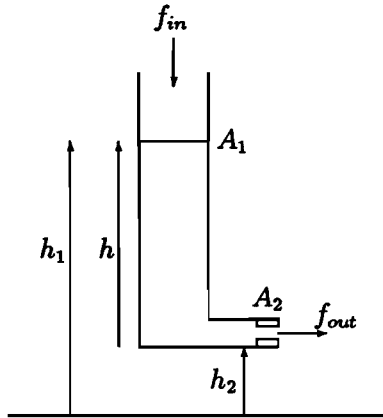
Figure 11.3: A water vessel

A water vessel with area A_1 and an in coming flow f_{in}. First consider the incoming flow $f_{in} = 0$. At the bottom the water flows through a restriction with area A_2 resulting in a out going flow $f_{out} = w_2 A_2 = w_1 A_1$. We assume $A_1 \gg A_2$, so that $w_2 \gg w_1$. Further we define $h = h_1 - h_2$, and equation (11.5) becomes

$$h\rho g = \frac{1}{2}\rho \frac{f_{out}^2}{A_2^2}$$

This results in

$$\boxed{f_{out} = A_2 \sqrt{2 h g}} \tag{11.6}$$

Now we consider the input flow $f_{in} \neq 0$. The increas of the water level is proportional to the difference between in coming flow and out going flow

$$\boxed{\frac{dh}{dt} = \frac{1}{A_1}(f_{in} - f_{out})} \tag{11.7}$$

The equations (11.6) and (11.7) together describe the non linear dynamic behaviour of one water vessel.

Linearized model:
We will consider the behaviour of a vessel about some working point. Let the in coming flow at the working point be given by $f_{in,0}$. In stationary operation we find

$$\frac{dh}{dt} = 0 \quad \text{and so} \quad f_{out,0} = f_{in,0}$$

The corresponding water level is found by (11.6):

$$f_{in,0} = f_{out,0} = A_2 \sqrt{2 h_0 g}$$

and so

$$h_0 = \frac{f_{in,0}}{2 g A_2^2}$$

We denoted the variations around the working point by:

$$f_{in} = f_{in,0} + \tilde{f}_{in}$$

$$f_{out} = f_{out,0} + \tilde{f}_{out}$$

$$h = h_0 + \tilde{h}$$

We derive

$$
\begin{aligned}
f_{out} &= f_{out,0} + \tilde{f}_{out} \\
&= A_2 \sqrt{2 (h_0 + \tilde{h}) g} \\
&= A_2 \sqrt{2 h_0 g} \sqrt{1 + \frac{\tilde{h}}{h_0}} \\
&\approx A_2 \sqrt{2 h_0 g} (1 + \frac{\tilde{h}}{2 h_0})) \\
&= f_{out,0} + A_2 \sqrt{\frac{g}{2 h_0}} \, \tilde{h}
\end{aligned}
$$

and so

$$\tilde{f}_{out} = A_2 \sqrt{\frac{g}{2 h_0}} \, \tilde{h}$$

With $f_{in} - f_{out} = \tilde{f}_{in} - \tilde{f}_{out}$ and

$$\frac{dh}{dt} = \frac{d\tilde{h}}{dt} = \frac{1}{A_2} \sqrt{\frac{2h_0}{g}} \frac{d\tilde{f}_{out}}{dt}$$

we derive

$$\frac{d\tilde{f}_{out}}{dt} = \frac{A_2}{A_1} \sqrt{\frac{g}{2h_0}} (\tilde{f}_{in} - \tilde{f}_{out})$$ (11.8)

Thus

$$\frac{\tilde{f}_{out}(s)}{\tilde{f}_{in}(s)} = \frac{\alpha}{s + \alpha} \quad \text{where} \quad \alpha = \frac{A_2}{A_1} \sqrt{\frac{g}{2h_0}}$$

From here we will only consider the linearized equations and use the notation $f_{out}(s)$ for $\tilde{f}_{out}(s)$ and $f_{in}(s)$ for $\tilde{f}_{in}(s)$.

The Multivariable linear model:
In the water vessel process as in figure 11.1 we have two layers with each layer consisting of two vessels. The out going flow $f_{out,3}$ of the vessel 3 is split into two fractions. A flow $f_{in,31} = \beta_1 f_{out,3}$ from vessel 3 to vessel 1 and a flow $f_{in,32} = (1 - \beta_1) f_{out,3}$ from vessel 3 to vessel 2. The same holds for the out going flow $f_{out,4}$ of the vessel 4, which is split into two fractions. A flow $f_{in,41} = (1 - \beta_2) f_{out,4}$ from vessel 4 to vessel 1 and a flow $f_{in,42} = \beta_2 f_{out,4}$ from vessel 4 to vessel 2. The in going flow at vessel 1 becomes

$$f_{in,1} = \beta_1 f_{out,3} + (1 - \beta_2) f_{out,4}$$

and the in going flow at vessel 2 becomes

$$f_{in,2} = (1 - \beta_1) f_{out,3} + \beta_2 f_{out,4}$$

We can find the overall model for the linearized process by considering zero-order hold transformation of the interconnection of the four water vessels. However if the sampling frequency is sufficiently large we may approximate the overall model by the interconnections of the zero-order transforms of the individual vessels.

The zero-order hold z-transformation of the i-th vessel with transfer function

$$H_i(s) = \frac{\alpha_i}{s + \alpha_i} \quad \text{is given by} \quad H_i(z) = \frac{1 - a_i}{z - a_i} \quad \text{where} \quad a_i = e^{-\alpha_i T}$$

We obtain a total transfer matrix

$$\begin{bmatrix} y_1(z) \\ y_2(z) \end{bmatrix} = \begin{bmatrix} \frac{K_1}{(z-a_1)(z-a_3)} & \frac{K_2}{(z-a_1)(z-a_4)} \\ \frac{K_3}{(z-a_2)(z-a_3)} & \frac{K_4}{(z-a_2)(z-a_4)} \end{bmatrix} \begin{bmatrix} u_1(z) \\ u_2(z) \end{bmatrix} \qquad (11.9)$$

This can be written as

$$K_0 \begin{bmatrix} K_4(z-a_1)(z-a_3) & -K_2(z-a_1)(z-a_4) \\ -K_3(z-a_2)(z-a_3) & K_1(z-a_2)(z-a_4) \end{bmatrix} \begin{bmatrix} y_1(z) \\ y_2(z) \end{bmatrix} =$$

$$= \begin{bmatrix} u_1(z) \\ u_2(z) \end{bmatrix}$$

where $K_0 = (K_1 K_4 - K_2 K_3)^{-1}$.
We multiply left and right by

$$\begin{bmatrix} K_1 & K_2 \\ K_3 & K_4 \end{bmatrix}$$

and we obtain

$$\begin{bmatrix} z^2 + \theta_1 z + \theta_5 & \theta_2 z + \theta_6 \\ \theta_3 z + \theta_7 & z^2 + \theta_4 z + \theta_8 \end{bmatrix} \begin{bmatrix} y_1(z) \\ y_2(z) \end{bmatrix} =$$

$$= \begin{bmatrix} \theta_9 & \theta_{10} \\ \theta_{11} & \theta_{12} \end{bmatrix} \begin{bmatrix} u_1(z) \\ u_2(z) \end{bmatrix}$$

where

$$\theta_1 = -a_3 + (K_2 K_3 a_2 - K_1 K_4 a_1) K_0$$

$$\theta_2 = K_1 K_2 (a_1 - a_2) K_0$$

$$\theta_3 = K_3 K_4 (a_2 - a_1) K_0$$

$$\theta_4 = -a_4 + (K_2 K_3 a_1 - K_1 K_4 a_2) K_0$$

$$\theta_5 = a_3 (K_1 K_4 a_1 - K_2 K_3 a_2) K_0$$

$$\theta_6 = K_1 K_2 a_4 (a_2 - a_1) K_0$$

$$\theta_7 = K_3 K_4 a_3 (a_1 - a_2) K_0$$

$$\theta_8 = a_4 (K_1 K_4 a_2 - K_2 K_3 a_1) K_0$$

$$\theta_9 = K_1$$

$$\theta_{10} = K_2$$

$$\theta_{11} = K_3$$

$$\theta_{12} = K_4$$

The final model is described with 12 parameters, where the model in equation (11.9) only had 8 parameters. The increase of parameters is necessary because we want the map

$$\Psi : \Theta \longrightarrow \mathbf{P}$$

to be linear (See Chapter 9).

11.2 Bounds on the disturbances

Disturbances on the input:
The input sensor measures the rotation frequency of the roller pump. For that purpose we measure the angular position of the shaft of the roller pump by a potential devider by means of a potentiometer that is connected to the shaft of the roller pump. The output voltage of the potentiometer is sampled by the computer, resulting in a saw-tooth signal for a constant rotation speed of the pump. Suppose the time interval between the first measurement v_f and the last measurement v_l of the saw-tooth signal is T and we measure n_T drops in the saw-tooth signal. A rough estimate for the average rotation speed in the interval is found by n_T/T. The number of drops can be detected without an error. A better estimate for the average rotation speed can be found by

$$u_m = \frac{1}{T} \left(n_T + \frac{v_l - v_f}{v_{tot}} \right)$$

where v_{tot} is the total range of the saw-tooth signal. The factor $(v_l - v_f)/v_{tot}$ can be measured with an error of about 10%. Because $|(v_l - v_f)/v_{tot}| \leq 1$ the maximum absolute error that can occur is 0.1. The total relative error that can occur is $0.1/n_T$. During the experiments there holds $n_T \gg 10$ and so the relative error we will make will always be less than 1% of the true rotation speed $u(k)$. We assume the error to be random over all frequencies and so we will consider as an upper bound for the input disturbance

$$|d_t(z)| \leq |0.01\, u(z)| \overset{def}{=} |W_d(z)|$$

If we make the magnitude of $u(z)$ to be constant over the frequency range the noise bounding filter $W_d(z)$ will become $W_d(z) = 0.01 \|u(z)\|_\infty$.

Disturbances on the output:
The output sensors measure the water levels in the lower vessels (1 and 2). The sensor consists of two plan parallel plates at a constant distance. An AC voltage with a frequency of $1kHz$ is applied to the plates. The impedance

between the plates will be inversely proportional to the water level. So a constant AC-voltage will result in an AC current, proportional to the water level. The carrier of the AC current is demodulated to a low frequency voltage and sampled by the computer. The main error sources of the output sensor are the following:

- Process disturbances and noise due to undulatory motion.

- Error due to adhesion of the water to the plates. This will give a hysteresis effect.

- Error due to the demodulation.

First consider the demodulation. Liebregts ([37]) shows that the demodulation error is negligible if the level sensor is calibrated before the data acquisition.

Next we consider the hysteresis error due to the adhesion of the water to the plates. Liebregts ([37]) shows that the hysteresis effect is smaller than $2\,mm$ for this level sensor. The problem is that this error is a bound in the time domain where we need a bound in the frequency domain. To transform this bound into the frequency domain we follow the method that is described in [37]. We first estimate a preliminary model P_{est} of the process. Note that we only need a rough estimation of the important frequency ranges instead of a real model. In our case the rough estimate is found by first computing $P_1(z) = Y(z)U(z)^{-1}$ where $U(z)$ and $Y(z)$ are the frequency domain data matrices with input and output measurements. Subsequently we compute the Markov chain $\bar{P}_1(k)$ by applying the inverse discrete Fourier transform. We truncate the Markov chain at a certain $l_k \ll 2N$ and we obtain the estimated Markov chain $\bar{P}_{est}(k)$. The value of l_k should be chosen equal to the assumed length of the impulse response of the true process. With this $\bar{P}_{est}(k)$ we can compute an estimation of the output signals

$$\begin{bmatrix} \bar{y}_{est,1}(k) \\ \bar{y}_{est,2}(k) \end{bmatrix} = \bar{P}_{est}(k) * \begin{bmatrix} \bar{u}_1(k) \\ \bar{u}_2(k) \end{bmatrix}$$

where $\bar{u}_1(k)$ and $\bar{u}_2(k)$ are the same process inputs as during the experiments. The signals $\bar{y}_{est,i}(k)$ represent the true water level before the hysteresis effects the measurements. We simulate the hysteresis by the worst possible hysteresis function as in figure 11.4 (see also Liebregts [37]).

We apply this hysteresis function to the signal $\bar{y}_{est,i}(k)$ and we obtain a signal $\bar{z}_i$ for the measured signal. We now hope that the quantity $\bar{e}_{hyst,i}(k) = \bar{y}_{est,i}(k) - \bar{z}_{est,i}(k)$ will give a good indication of the maximum error due to the hysteresis. We compute the discrete Fourier transform of $\bar{e}_{hyst,i}(k)$ and determine an upper bound for $|e_{hyst,i}(z)|$ and take this as a frequency domain
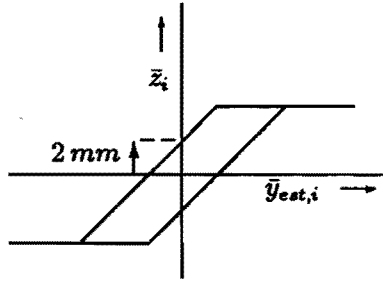
Figure 11.4: The worst case hysteresis

upper bound for the error due to the hysteresis effect.

Finally we discuss the process disturbances and noise due to undulatory motion of the water in the vessels. The effect can be measured by doing a free run experiment. We fix the input signal $u(k) = 0$ and measure the magnitude of the output signal $y(k)$. The main problem is that we can not eliminate the effect of the hysteresis and we will get the hysteresis error in our measured output signal. We will find in the next section that the measured disturbance during the free run experiment will be about the same as the estimation of the upper bound to due the hysteresis. We will therefore neglect the process disturbances.

11.3   Experiments and identification

From Liebregts ([37]) we know that the process about a working point with a high water level can be linearized very good. Therefore we choose a range of 2 Volts to 3 Volts for the pump voltages. We will derive a linear model about the offset voltages $v_{1,0} = v_{2,0} = 2.5$ Volts.
We find a stabilizing controller for the process with a transfer

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = K \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

where the signal u_i is the rotation speeds of a roller pump ($[Hz]$), the signal y_i is the water levels in one of the lower vessels ($[cm]$) and where we have taken the controller $K = 0.01\, I\, [Hz/cm]$.
The coprime factors $N_C(z)$ and $M_C(z)$ of the controller allow us to scale the effect of the disturbances on the denominator of the expression for the model error as derived in section 6.4

$$\Delta_t(z) = (MY - NU - ME_t - ND_t)(N_C Y + M_C U + M_C D_t - N_C E_t)^{-1}$$

By choosing the magnitude of the controller smaller than 1 we make $M_C(z)$ larger than $N_C(z)$ and so the influence of $E_t(z)$, which is larger than $D_t(z)$, in the denominator will be reduced.

The controller has shown to stabilize the process during a closed loop experiment. We will use this controller for the analysis as discussed in chapter 5. The experiments to obtain the datasets $\{u, y\}$ will be done without a controller.

Orientation experiments:

We first consider some orientation experiments that intend to obtain preliminary information about the process. We follow the procedure from Backx and Damen ([4]) which exists of three experiments:

- Free run experiment.
- Staircase experiment.
- Fast PRBNS experiment.

We will discuss the first three experiments and their results:

During the free run experiment a constant voltage of 2.5 Volts is applied to both the roller pumps. This results in a constant water level in all vessels up to the process disturbance that is still present. This process disturbance exists of a low frequency trend and higher frequency noise. The low frequency trend can not be modelled and is filtered out. The remaining high frequency disturbance appears to be smaller than the $2\,mm$ of the hysteresis error. We will therefore neglect the process disturbance and assume that it is incorporated in the bound due to the hysteresis error.

During the staircase experiment we apply a staircase test signals as input voltage to the roller pumps of the process. We get an idea about the non linearity of the process and we get a rough estimation of the largest and smallest time constant. Although the process shows some non linearities we will neglect these effects. The dynamical behaviour does not change very much over the complete range.

The fast PRBNS (pseudo random binary noise sequence) experiment is meant to find the bandwidth of the process. In Liebregts [37] a SISO version of the water vessel process was identified. During a fast PRBNS experiment a bandwidth was found of $B_{SISO} = 0.02\,Hz$. We do not expect that the bandwidth will differ very much for the MIMO case but to check this bandwidth we apply a fast PRBNS signal to the process. We take a sampling frequency of $1\,Hz$, which is 50 times the expected bandwidth.

We find that the bandwidth of process $B = 0.04\,Hz$ so that we may conclude that a for the final experiment a sampling frequency $f_s = 1/6\,Hz$ is enough to excite all relevant dynamics of the process.

Now the preliminary experiments are done we consider the final experiment.

Input design for final experiment:
The process has two input signals and according to chapter 4 we will have to do 2 experiments. The input voltages of the roller pumps, that are used as reference signals, can be arranged in a matrix

$$\bar{V}(k) = \bar{V}_0(k) + \bar{V}_{os}(k)$$

where

$$\bar{V}_{os}(k) = 2.5 \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

is a constant matrix, due to the offset voltage, and

$$\bar{V}_0(k) = \begin{bmatrix} \bar{v}_{11}(k) & \bar{v}_{12}(k) \\ \bar{v}_{21}(k) & \bar{v}_{22}(k) \end{bmatrix}$$

is the variantion around this offset voltage. The reference signal $\bar{V}_0(k)$ should be designed in a proper way, i.e. a good reference signal will excite the system with a maximum amplitude and the resulting frequency domain matrix $V_0(z)$ will have a nice condition number $\sigma_{max}/\sigma_{min} \approx 1$. A set of reference signals that satisfy this conditions is given by

$$\bar{V}_0(k) = \begin{bmatrix} 0.5 & -0.5 \\ 0.5 & 0.5 \end{bmatrix} \bar{v}_0(k)$$

Here $\bar{v}_0(k)$ is a scalar valued signal with a length of $2N = 5000$ samples that satisfies the following conditions:

- $|\bar{v}_0(k)| \approx 1$ in the interval $k = 100, \ldots, 4900$.
- $\bar{v}_0(k) \approx 0$ for $k = 1, .., 100$ and $k = 4900, .., 5000$.
- The frequency signal $v_0(z)$ has approximately an flat spectrum.
   (so $|v_0(z)| \approx$ constant).

A reference signal that satisfies these conditions will have maximum energy within the limits of the maximum amplitude. Further in section 2.5 we showed that the errors in the discrete Fourier transforms are negligible if the impulse responses length l_1 of the function $(I + P_t K_{exp})^{-1}P_t$ and the impulse responses length l_2 of the function $(I + P_t K_{exp})^{-1}$ are both much smaller than the length of the observation interval $(2N)$. From the staircase experiments we found $l_1 \approx 100$ and because $K_{exp} = 0$ the impulse response length l_2 is not important. Finally the frequency signal $V_0(z)$ will be invertible for all $z \in C$ and because the rotation speed will be approximately proportional to

the input voltage (up to the saturation effect), we hope that the matrix $U(z)$ will be invertible for all $z \in C$.

We do the final experiments with these signals of input voltages of the roller pumps. For both experiments we measure the roller pump rotation speeds and the water levels in the lower vessels. We obtain two datasets $\{u_1, y_1\}$ and $\{u_2, y_2\}$. We apply a discrete Fourier transform and construct the frequency domain matrices:

$$U(z) = [u_1(z) \ u_2(z)]$$

$$Y(z) = [y_1(z) \ y_2(z)]$$

We parametrize the model coprime factors with the ARMA models:

$$N(z) = C(z)^{-1}B(z) \ \text{and} \ M(z) = C(z)^{-1}A(z)$$

where

$$A(z) = \begin{bmatrix} z^2 + \theta_1 z + \theta_5 & \theta_2 z + \theta_6 \\ \theta_3 z + \theta_7 & z^2 + \theta_4 z + \theta_8 \end{bmatrix}$$

$$B(z) = \begin{bmatrix} \theta_9 & \theta_{10} \\ \theta_{11} & \theta_{12} \end{bmatrix}$$

$$C(z) = z^2 I$$

with $\theta = [\theta_1, \cdots, \theta_{12}]^T$. The denominator polynomial $C(z)$ has two roots for $z = 0$ so that the coprime factors $N(z)$ and $M(z)$ will be stable.

We compute the bounding filters $W_d(z)$ and $W_e(z)$ for the input and output disturbances as mentioned in the previous section. The disturbances on both inputs have the same bounding filters, so that $W_d(z) = w_d(z) I$ with $w_d(z)$ a scalar function. Because the signals $u(z)$ have an approximately flat spectrum we take an upper bound $w_d(z) = 0.29 \approx 0.01 |u(z)|$. The same holds for the output disturbances where we find $W_e(z) = w_e(z) I$ with $w_e(z)$ a scalar function. A bounding filter for the hysteresis error is found

$$w_e(z) = \frac{40 z^2}{(z - 0.8)^2}$$

For the disturbance description we have two options: An unstructured or a structured set description as discussed in Chapter 4. The main difference between the two sets is that the optimization of the model error criterion will be a convex optimization problem for the unstructured case or an approximately convex optimization problem for the structured case. The optimization in the structured case is theoretically convex, however, the structured singular value μ can only be approximated by an upper bound. This approximation may lead

to a loss of convexity (See discussion in Chapter 7).

The normalized coprime factors of the controller $K(z) = 0.01 I$ are given by $M_C(z) = 0.99995 I \approx I$ and $N_C(z) = 0.0099995 I \approx 0.01 I$.

Because the controller $K(z) \neq 0$ and $K(z) \neq \infty$ we can choose for a coprime factor model error structure.

We will minimize the model error criterion $J_\infty(\gamma(\theta, z))$, the infinity norm of the upper bound for the coprime factor model error. We first start the optimization of using an unstructured description of the disturbances.

Optimal model for an unstructured noise description:
We now minimize the criterion

$$J_\infty^u(\theta_{opt}) = \min_\theta J_\infty(\gamma(\theta, z)) \mid_{unstructured\ noise}$$

using a convex cutting plane algorithm. The algorithms are implemented in MATLAB, and we make use of the μ algorithm from the μ-tools software package [5].

An optimum is found for

$$\theta_{opt} = \begin{bmatrix} -1.7634 \\ -0.1011 \\ -1.4481 \\ -1.6472 \\ 0.8237 \\ 0.1178 \\ 0.6934 \\ 0.7190 \\ -3.9571 \\ -2.2058 \\ 0.7629 \\ -0.0529 \end{bmatrix} \quad \text{where} \quad J_\infty^u(\theta_{opt}) = 1.05$$

The poles of the optimal model $P(\theta_{opt}, z) = M^{-1}(\theta_{opt}, z) N(\theta_{opt}, z)$ are equal to the poles of $M^{-1}(\theta_{opt}, z)$. These poles are the roots of the polynomial

$$(z^2 + z\theta_1 + \theta_5)(z^2 + z\theta_4 + \theta_8) - (z\theta_2 + \theta_6)(z\theta_3 + \theta_7)$$

For θ_{opt} we find the roots:

$$z_{1,2} = 1.1432 \pm j\,0.2154$$

$$z_{3,4} = 0.5621 \pm j\,0.2476$$

The first two poles are outside the unit disk, which means that the optimal model is unstable. Furthermore, the model P_{opt} in closed loop with $K = 0.01\,I$ gives rise to an unstable closed loop. We know that the true process is stable by itself and that it is stabilzed by $K = 0.01\,I$ as well. For these parameters θ_{opt} we can not guarantee that the model coprime factors $N(\theta_{opt}, z)$ and $M(\theta_{opt}, z)$ were compared with coprime factors of the true process (see section 5.2 and 5.3) and so we can not guarantee that the bound J_∞^u is an upper bound for the coprime factor model error.

We now could use the procedure that was discussed in chapter 10 and take another coprime factorzation of the true process. However, we are more interested how the structured disturbance description can improve the coprime factor model error. We will only use the 'optimal' parameters θ_{opt} as initial values for the optimization using a structured disturbance description.

Optimal model for a structured noise description:
We now minimize the criterion

$$J_\infty^s(\theta_{opt}) = \min_\theta J_\infty(\gamma(\theta, z)) \mid \textit{structured noise}$$

using a convex ellipsoid algorithm. The algorithms are again implemented in MATLAB. The optimum is not reached, because of convergence problems, due to the fact that the structured singular value can only be approximated by an upper bound using the D-iteration. By using this upper bound instead of the real value μ the convexity of the optimization problem is lost.

The optimization has not reached its minimum but we do not find much improvement of the criterion after 176 iterations. We will denote the last estimate (after 176 iterations) with parameters θ_{last} as the final (suboptimal) model.

$$\theta_{last} = \begin{bmatrix} -1.8291 \\ -0.0192 \\ -0.0989 \\ -0.0474 \\ 0.8401 \\ 0.0229 \\ -0.0980 \\ 0.0078 \\ 0.1263 \\ -0.0925 \\ -0.0008 \\ -0.9498 \end{bmatrix} \quad \text{where} \quad J_\infty^s(\theta_{last}) = 0.33$$

We compute the poles of $M^{-1}(\theta_{last}, z)$ and find:

$$z_{1,2} = 0.9150 \pm j\,0.0611$$

$$z_{3,4} = 0.0232 \pm j\,0.0996$$

All poles are inside the unit disk, which means that the sub optimal model is stable. For the closed loop with the controller $K = 0.01\,I$ we find the poles:

$$z_{1,2} = 0.9151 \pm j\,0.0498$$

$$z_{3,4} = 0.0232 \pm j\,0.1397$$

Also these poles are inside the unit disk, so that we may conclude that an upper bound for the coprime factor model error with respect to the coprime factors $N(\theta_{last}, z)$ and $M(\theta_{last}, z)$ is given by $J_\infty^s(\theta_{last})$. It can be motivated from physical considerations that we expect four real valued poles. The poles $p_{1,2}$ as well as the poles $p_{3,4}$ are complex conjugated pairs, but the imaginary parts are indeed very small.
Note that $J_\infty^s(\theta_{last}) < J_\infty^u(\theta_{opt})$, which proofs that the unstructured description of the noise sets is more conservative than the structured description of the noise sets.

In figure 11.5 the upper bound for the coprime factor model error is given together with the largest singular values of the coprime factors $N(\theta_{last}, z)$ and $M(\theta_{last}, z)$. In figure 11.6 the entries of the coprime factors $N(\theta_{last}, z)$ (solid line) and $M(\theta_{last}, z)$ (dashed line) are plotted.
We find that the bound $\gamma(\theta_{last}, z)$ is larger than $\sigma_{max}\big(N(\theta_{last}, z)\big)$ and $\sigma_{max}\big(M(\theta_{last}, z)\big)$ for almost all $z \in C$. This means that for robust control design the model error bound γ is far to large. First we can consider the coprime factors errors separately. The error in the M-factor can be bounded as follows

$$\|\Delta_M\| = \|N_C\,\Delta\| \leq \|N_C\|\,\|\Delta\| \leq 0.01\,0.03 = 3\,10^{-4}$$

This means that the M-factor is estimated with an error smaller than 1.5% for the lower frequencies and an error smaller than 0.1% for the higher frequencies. So the poles of the process (the poles of M^{-1}) are likely to be estimated accurately. The error in the N-factor can be bounded

$$\|\Delta_N\| = \|M_C\,\Delta\| \leq \|M_C\|\,\|\Delta\| \leq 0.03$$

So the uncertainty in the N-factor is larger than the magnitude of the factor itself. This means that the gain of the process (the N-factor) is estimated very unreliably.
One reason may be that the choice of the stabilizing controller was not an appropriate one. The controller determines in which coprime factor the error

will appear. A controller $K = \alpha I$ with $\alpha < 0.01$ would probably have given a better estimate of the N-factor, with a larger error in the M-factor.

But even for the applied controller $K = 0.01\,I$ the error is very big. This might be due to the simplified model or maybe the sampling frequency of 1/6 Hz was still to high. Further study must be done to examine the effects of different choices of controllers, sampling frequencies and model sets.

Evaluation of the assumptions:
We follow the procedure of section 10.1. Because $K_{exp} = 0$ in our case we only compute the impulse response length of the the model $P_{last} = M^{-1}(\theta_{last}, z)N(\theta_{last}, z)$. We find $\hat{l}_1 = 75 < 100 = l_1$. This gives us some confidence that the assumption l_1 may be a realistic assumption.
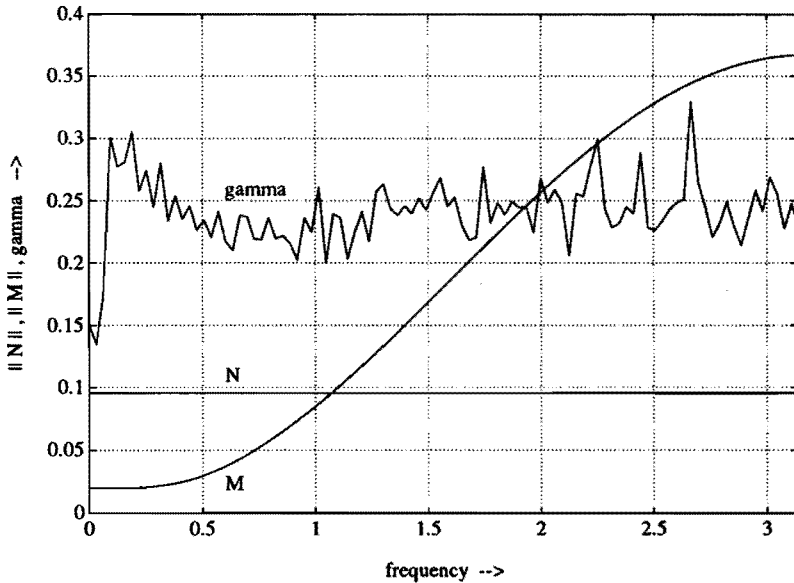

Figure 11.5: Upper bound for the coprime factor model error and largest singular values of the coprime factors.

11.4 Conclusions

In this chapter we presented a case study. The system identification procedure as introduced in this thesis has been applied to a water vessel process, a MIMO
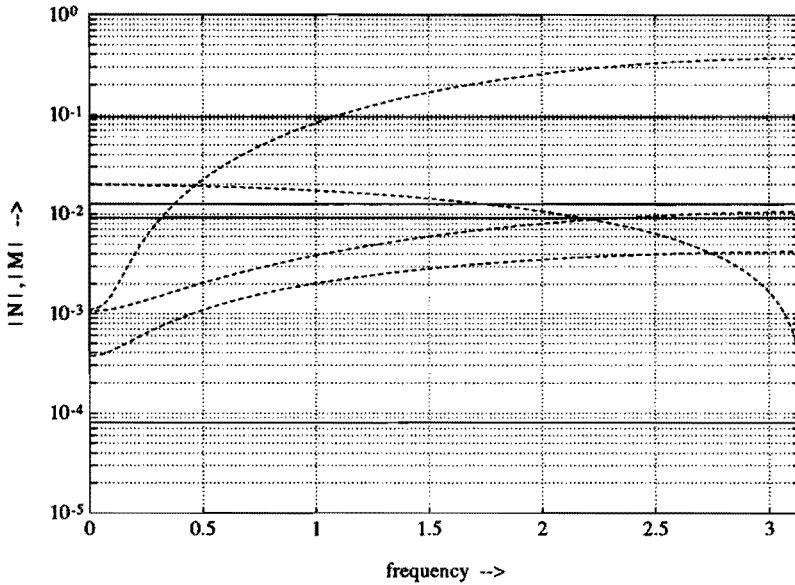
Figure 11.6: The entries of the coprime factors $|N_{ij}|$ (solid lines) and $|M_{ij}|$ (dashed lines).

system with two inputs and two outputs.

Bounds for the disturbances on the input and output sensors were derived from measured data and from a theoretical analysis of the sensors.

A controller $K = 0.01\,I$ was tested on the true process resulting in a stable closed loop. We considered a coprime factor model error structure and a linear parametrization for the model coprime factors, ARMA models with fixed AR parameters, was introduced.

Two experiments were done to obtain the final data set. With the data sets and with the derived disturbance bounds we optimized the model error crtiterion $J_\infty(\gamma(\theta, z))$. In the case of a unstructured disturbance set description the optimization converged to an optimum, resulting in model error criterion $J_\infty^u = 1.05$. The optimal model however was not stabilized by the controller $K = 0.01\,I$ and so the model error criterion had no meaning in the sense of an upper bound for the coprime factor model error.

We used the 'optimal' parameter vector of this experiment as a initial value for the optimization using a structured disturbance set description. The algorithm did not converge and after 176 iterations the procedure was aborted. The parameters corresponding to the last iteration gave rise to a model that was stabilized by $K = 0.01\,I$.

The model error criterion $J_\infty^s = 0.33$ can therefore be used as a bound for the coprime factor model error with respect to the model coprime factors corresponding to the parameters of the 176th iteration.

This bound however is very large and therefore not suitable for robust control design. Most likely a better choice of controller $K = \alpha I$, with $\alpha < 0.01$ will give a smaller coprime factor model error bound, with an increased error in the M-factor, but with a smaller error in the N-factor.

12

Conclusions and Remarks

In this thesis a procedure has been described for the identification of linear time invariant systems with a bounded model error.

An important difference between the identification methods, described in this thesis, and the conventional identification methods is that there has been chosen for a approach in which an optimal model is defined by a minimum H_∞-norm upper bound on the model error. The description of the disturbances on input and output is done using bounding filters in the frequency domain.

The apriori knowledge we need is

- A data set $\{ u(k), y(k) \}$.
- Frequency domain bounds $W_d(z)$ and $W_e(z)$ on the disturbances.
- A stabilizing controller K_{exp} for the experiments.
- A stabilizing controller K_{ana} for the analysis.
- A bound on the impulse response length of the closed loop.
- A bound on the impulse response length of the true model error.

The identification of SISO-systems with additive or multiplicative model error

181

structures can be done using graphical methods. These methods give a lot of insight in the problems that may arise but they are very limited if we want to extend the problem to multivariable systems and other model error structures. For the identifiaction of multivariable systems we need to do multiple experiments to guarantee excitation of all dynamics and their internal interactions. The modelling of the disturbance for MIMO-systems can be done in a structured or unstructured way. In both cases we end up with a scaled noise matrix $Q_t(z)$ that is in the unit ball.

The estimations of bounds for models with a coprime factor model error structure is non-unique and the coprime factorization of the model should be fixed. The R-parametrization has given us a useful tool to bring all model error structures in one basic scheme, in which the model error is represented by a stable function $\Delta_t(z)$.

The concept of Redheffer star products has given a good insight in the interactions between model error, model error structure, model coprime factors, measured data and the disturbances. It gives us the possibility to study the problem in a more algebraic way.

A upper bound on the model error is minimized in the sense of some norm (H_∞, H_2 or a combination). The use of μ-techniques gives the tightest upper bound for the model error.

With the derived techniques we can compute upper bounds for various model error criteria for a fixed model. We can parametrize the model and minimize the model error upper bound.

In the case of a linear parametrization the optimization problem will become convex and the optimization will be easy.

The identification procedures were tested on a laboratory process, a water vessel process with two inputs and two outputs. Bounds on the coprime model errors were derived, but they were far too large for H_∞ control design. An important reason for this large bound might be a bad choice for the controller $K(z)$, used in the analysis. Another problem was that the final parameter optimization algorithms did not converge. This was caused by the approximation of the structured singular value by its upper bound, so that convexity was lost.

This thesis gave the first steps into an new identification area, that of system identification for robust control. The developed techniques are the first attempts to derive minimum model error bounds. One single path has been paved, but it will most probably not be the optimal one. There are still some interesting problems left:

- Choice of controller $K_{ana}(z)$. In Chapter 11 the bound on the coprime model error was too large to be used for H_∞ control design. Probably

another controller would have given a better result.

- In chapter 5 a unimodular funtion $A(z)$ was introduced, which was fixed to the identity matrix in the sequel of the thesis. The optimization over this matrix $A(z)$ may be a difficult task, but it is plausible that such an optimization will improve the resulting model and further decrease the model error bound.

- The interpretation of the coprime factor model error structure is not an obvious one. Not only the input-output behaviour of the process is important, but also the realization. The choice of coprime factorization plays an important role in the magnitude of the resulting model and its model error bound. An interesting point of further research could be the study of the interaction between choice of model coprime factorization (and so of the choice of the model error structure), the final coprime factor model error bound and the resulting optimal controller.

- In section 7.4 the model error criteria J_∞, J_2, J_s, J_{ss}, J_{max} were discussed. It may be interesting to study the minimization of one of these criteria (J_m) with a specified bound on one or more other criteria $(J_{c,1}, \cdots, J_{c,k})$. This will lead to a constrained optimization problem

$$\min_{\theta \in \Theta} J_m$$

where

$$J_{c,1} \le \beta_1 , \cdots , J_{c,k} \le \beta_k$$

The values β_i, $i = 1, \ldots, k$ are prespecified levels. For example, if we are satisfied with an ∞-norm that is smaller than some β_∞ we can minimize the 2-norm criterion J_2 under the constraint that $J_\infty \le \beta_\infty$. In this way we can get a model with a specified worst case bound and a good average behaviour.

- If a linear parametrization is chosen, the model error criterion as introduced in chapter 9 will become convex in $\theta \in I\!\!R^n$. The main problem of the model error criterion is the computation of the structured singular value μ, which gives rise to a loss of convexity. This was found in the case study in chapter 11. A closer look should be taken at the algorithms to get the optimization algorithm more robust.

- The linear parametrization has the big advantage that it leads to a convex optimization problem. However, other parametrizations are possible as

well (parametrization of normalized coprime factors, see Van den Boom [56]). Such kind of parametrizations need more attention in further research.

- Because of the assumptions on the disturbances we can compute the model error bound frequency by frequency. This makes the optimization very easy, for we can make use of the μ-techniques. On the other hand it makes the resulting bound very conservative.

  By incorporating more apriori knowledge about the interaction of the model error at different frequencies, we can improve the bound on the model error. As an example of a frequency interaction we mention the assumption of a finite length of the model error impulse response. The final optimization with such an extra frequency interaction assumption will usually become more difficult.

Appendix A:
Proofs

Appendix A1

Proof of theorem 3.1:

We can give the sets $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{Y}}$ for one specific frequency as:

$$\tilde{\mathbf{U}} = \left\{ \tilde{u} = u(1 + \alpha_u e^{j\theta}) , \; 0 \leq \theta < 2\pi , \; 0 \leq \alpha_u \leq \frac{|W_d|}{|u|} \right\}$$

$$\tilde{\mathbf{Y}} = \left\{ \tilde{y} = y(1 + \alpha_y e^{j\phi}) , \; 0 \leq \phi < 2\pi , \; 0 \leq \alpha_y \leq \frac{|W_e|}{|y|} \right\}$$

To derive the set $\mathbf{P}$ we will need an auxilary set $\tilde{\mathbf{X}}$ (see Fig.2.2) with signals $\tilde{x} = \tilde{u}^{-1}$ for all $\tilde{u} \in \tilde{\mathbf{U}}$.

Under the assumption that $|u| > |W_d| > 0$ so $(uu^* - W_d W_d^*) > 0$
it holds:

$$|\tilde{u} - u| \leq |W_d| \quad \Leftrightarrow \quad (\tilde{u} - u)(\tilde{u} - u)^* \leq W_d W_d^* \quad \Leftrightarrow$$

185

$$(\tilde{u}\tilde{u}^* - \tilde{u}u^* - u\tilde{u}^* + uu^* - W_dW_d^*) \leq 0 \quad \Leftrightarrow$$

$$\frac{1}{\tilde{x}\tilde{x}^*} - \frac{u^*}{\tilde{x}} - \frac{u}{\tilde{x}^*} + (uu^* - W_dW_d^*) \leq 0 \quad \Leftrightarrow$$

$$1 - u\tilde{x}^* - u^*\tilde{x} + (uu^* - W_dW_d^*)\tilde{x}\tilde{x}^* \leq 0 \quad \Leftrightarrow$$

$$\tilde{x}\tilde{x}^* - \frac{u\tilde{x}^* + u^*\tilde{x}}{uu^* - W_dW_d^*} + \frac{uu^*}{(uu^* - W_dW_d^*)^2} - \frac{W_dW_d^*}{(uu^* - W_dW_d^*)^2} \leq 0 \quad \Leftrightarrow$$

$$\left[\tilde{x} - \frac{u^*}{(uu^* - W_dW_d^*)}\right]\left[\tilde{x} - \frac{u^*}{(uu^* - W_dW_d^*)}\right]^* \leq \frac{W_dW_d^*}{(uu^* - W_dW_d^*)^2} \quad \Leftrightarrow$$

$$\left|\tilde{x} - \frac{u^*}{(uu^* - W_dW_d^*)}\right| \leq \frac{|W_d|}{(uu^* - W_dW_d^*)}$$

So the set $\bar{\mathbf{X}}$ is given by the ball:

$$\bar{\mathbf{X}} = \left\{ \tilde{x} \mid \left| \tilde{x} - \frac{u^*}{(uu^* - W_dW_d^*)} \right| \leq \frac{|W_d|}{(uu^* - W_dW_d^*)} \right\} =$$

$$\left\{ \tilde{x} = \frac{u^*}{(uu^* - W_dW_d^*)}(1 + \alpha_u e^{j\psi}), \, 0 \leq \psi < 2\pi, \, 0 \leq \alpha_u \leq \frac{|W_d|}{|u|} \right\}$$

$\square$

Appendix A2

Proof of theorem 3.2:

We have the region $h(\phi, \psi) = (1 + re^{j\phi})(1 + se^{j\psi})$, where $0 \le \phi < 2\pi$ and $0 \le \psi < 2\pi$. If $h(\phi, \psi)$ is a boundary point of the region for some ϕ and ψ, then it holds for the derivatives (see Fig.A1):

$$\frac{\partial h(\phi, \psi)}{\partial \phi} = \frac{\partial h(\phi, \psi)}{\partial \psi} C_0 \quad \text{with} \quad C_0 \in \mathbb{R}$$
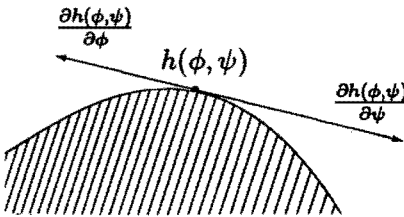
Fig.A1: Derivatives of $h(\phi, \psi)$ at a boundary point

This leads to

$$\text{Im}\left(\frac{\partial h(\phi, \psi)/\partial \phi}{\partial h(\phi, \psi)/\partial \phi}\right) = 0$$

and therefore:

$$\text{Im}\left(\frac{\partial h(\phi, \psi)/\partial \phi}{\partial h(\phi, \psi)/\partial \phi}\right) = \text{Im}\left(\frac{j\, r\, e^{j\phi}(1 + s\, e^{j\psi})}{j\, s\, e^{j\psi}(1 + r\, e^{j\phi})}\right) =$$

$$= \text{Im}\left(\frac{(r\, e^{j\phi} + rs\, e^{j(\phi+\psi)})(e^{-j\psi} + r\, e^{-j(\phi+\psi)})}{(s\, e^{j\psi} + rs\, e^{j(\phi+\psi)})(e^{-j\psi} + r\, e^{-j(\phi+\psi)})}\right) =$$

$$= \text{Im}\left(\frac{r(e^{j(\phi-\psi)} + s\, e^{j\phi} + r\, e^{-j\psi} + rs)}{s(1 + r\, e^{j\psi} + r\, e^{-j\psi} + r^2)}\right) =$$

$$= \frac{r\left(\sin(\phi - \psi) + s\, \sin\phi - r\, \sin\psi\right)}{s(1 + 2r\, \cos\phi + r^2)} = 0$$

so we find the condition

$$\boxed{\sin(\phi - \psi) + s\, \sin\phi - r\, \sin\psi = 0}$$

or

$$\frac{e^{j(\phi-\psi)} - e^{-j(\phi-\psi)}}{2j} + \frac{s\,e^{j\phi} - s\,e^{-j\phi}}{2j} - \frac{r\,e^{j\psi} - r\,e^{-j\psi}}{2j} = 0$$

Now we define $a = e^{j\phi}$ and $b = e^{j\psi}$ and we get the condition

$$\frac{a}{b} - \frac{b}{a} + sa - \frac{s}{a} + \frac{r}{b} - rb = 0$$

and because $a \neq 0$ and $b \neq 0$ we can write:

$$a^2 - b^2 + sa^2b - sb + ra - rab^2 =$$

$$= a^2(1 + sb) + a(r - rb^2) + (-b^2 - sb) = 0$$

We get the solutions

$$a_{1,2} = \frac{r(b^2 - 1) \pm \sqrt{r^2(1 - b^2)^2 + 4(1 + sb)(b^2 + sb)}}{2(1 + sb)} =$$

$$= \frac{r(b^2 - 1) \pm b\sqrt{r^2(b^{-1} - b)^2 + 4(1 + sb)(1 + sb^{-1})}}{2(1 + sb)} =$$

$$= \frac{r(e^{2j\psi} - 1) \pm e^{j\psi}\sqrt{r^2(e^{-j\psi} - e^{j\psi})^2 + 4(1 + se^{j\psi})(1 + se^{-j\psi})}}{2(1 + se^{j\psi})} =$$

$$= \frac{r(e^{2j\psi} - 1) \pm 2e^{j\psi}\sqrt{-r^2\sin^2\psi + 2s\cos\psi + 1 + s^2}}{2(1 + se^{j\psi})}$$

We substitute this solution for a in $h(\phi,\psi)$ and we find two functions $h_{b1}(\psi)$ and $h_{b2}(\psi)$ one corresponding to a '+' sign and one corresponding to a '−' sign:

$$h_{b1}(\psi) = (1 + ra)(1 + sb) = 1 + sb + (1 + sb)ra =$$

$$= 1 + s\,e^{j\psi} + \frac{r^2}{2}(e^{2j\psi} - 1) +$$

$$+ r\,e^{j\psi}\sqrt{-r^2\sin^2\psi + 2s\cos\psi + 1 + s^2}$$

likewise

$$h_{b2}(\psi) = 1 + s\,e^{j\psi} + \frac{r^2}{2}(e^{2j\psi} - 1) -$$

$$r\,e^{j\psi}\sqrt{-r^2\sin^2\psi + 2s\cos\psi + 1 + s^2}$$

Now we substitute $\psi = 0$ and $\psi = \pi$,resulting in:

$$h_{b1}(0) \ = \ 1 \ + \ s \ + \ \frac{r^2}{2}(1 \ - \ 1) \ + \ r\sqrt{1 \ + \ 2s \ + \ s^2} \ =$$

$$= \ 1 \ + \ s \ + \ r(1 \ + \ s) \ = \ 1 \ + \ s \ + \ r \ + rs$$

$$h_{b1}(\pi) \ = \ 1 \ - \ s \ + \ \frac{r^2}{2}(1 \ - \ 1) \ - \ r\sqrt{1 \ - \ 2s \ + \ s^2} \ =$$

$$= \ 1 \ - \ s \ - \ r(1 \ - \ s) \ = \ 1 \ - \ s \ - \ r \ + rs$$

$$h_{b2}(0) \ = \ 1 \ + \ s \ + \ \frac{r^2}{2}(1 \ - \ 1) \ - \ r\sqrt{1 \ + \ 2s \ + \ s^2} \ =$$

$$= \ 1 \ + \ s \ - \ r(1 \ + \ s) \ = \ 1 \ + \ s \ - \ r \ - rs$$

$$h_{b2}(\pi) \ = \ 1 \ - \ s \ + \ \frac{r^2}{2}(1 \ - \ 1) \ + \ r\sqrt{1 \ - \ 2s \ + \ s^2} \ =$$

$$= \ 1 \ - \ s \ + \ r(1 \ - \ s) \ = \ 1 \ - \ s \ + \ r \ - rs$$

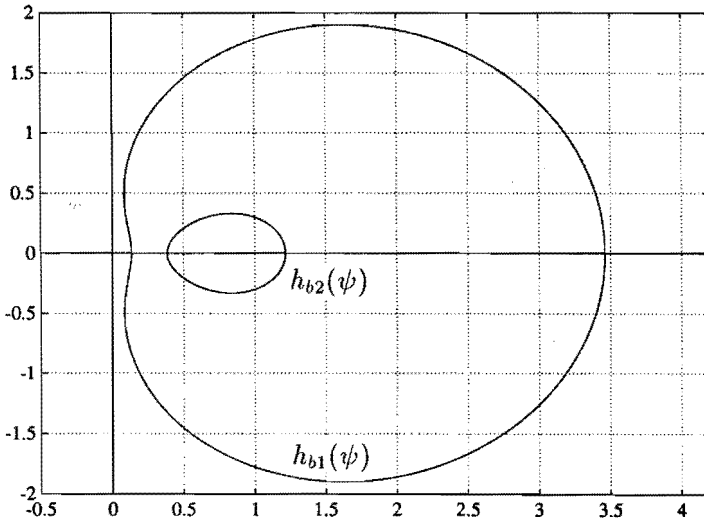Fig.A2: The two boundary functions $h_{b1}(\psi)$ and $h_{b2}(\psi)$.

So the function $h_{b1}(\psi)$ is the desired bound of the region $h(\phi, \psi)$. The function $h_{b2}(\psi)$ is an inner boundary that is caused by fixing the values $\alpha_u = s$ and $\alpha_y = r$. If we consider again $0 \leq \alpha_u \leq s$ and $0 \leq \alpha_y \leq r$, then all inner points will be allowed, and the inner boundary vanishes.

□

Appendix A3

Proof of theorem 3.3:

The function

$$h(\phi, \psi) = (1 + r\,e^{j\phi})(1 + s\,e^{j\psi})$$

where $\phi = f(\psi)$, gives the boundary $h(f(\psi), \psi) = h_b(\psi)$ (see Appendix A2) of the uncertainty region $\bar{p}(\alpha_1, \alpha_2, \phi, \psi)$ with α_1, α_2, ϕ and ψ are in the allowed regions. We seek for a disk with a minimal radius, that encloses the region $\bar{p}(\alpha_1, \alpha_2, \phi, \psi)$.

We will consider the distance ρ between the point $c_0 > 1$ and the function $h_b(\psi)$, and determine the maxima of this function $|\,h_b(\psi)\, - \,c_0\,|$. Then we prove that maxima are reached in the points $h(\phi_0, \psi_0) = h_b(\psi_0) = c_0 + j\,\rho_0$ and $h(-\phi_0, -\psi_0) = h_b(-\psi_0) = c_0 - j\,\rho_0$. This means that the circle with center point c_0 and radius ρ_0 encloses the region $h(\phi, \psi)$. Because the points $c_0 + j\,\rho_0$ and $c_0 - j\,\rho_0$ are both in the region, the mentioned circle will be the smallest enclosing circle, see Fig.A3.

First we will take a closer look at the centre-point c_0 and radius ρ_0:

$$\rho_0 = \mathrm{Im}\left(h(\phi_0, \psi_0)\right) = r\,sin\phi_0 \,+\, s\,sin\psi_0 \,+\, rs\,sin(\phi_0 + \psi_0)$$

it is clear that $\rho_0 \,>\, r \,+\, s \,=\, \mathrm{Im}\left(h(\frac{\pi}{2}, \frac{\pi}{2})\right)$ and thus it must hold:

$$sin\phi_0 \,>\, 0 \ , \quad sin\psi_0 \,>\, 0 \ \text{and} \ sin(\phi_0 + \psi_0) \,>\, 0$$

Further we know from the conditions for ϕ_0 and ψ_0 that:

$$r\,cos\phi_0 \,=\, s\,cos\psi_0 \,=\, -rs\,cos(\phi_0 + \psi_0) \stackrel{def}{=} y_0$$

Therefore we find: $0 < \phi_0 < \frac{\pi}{2}$, $0 < \psi_0 < \frac{\pi}{2}$ and $\frac{\pi}{2} < (\phi_0 + \psi_0) < \pi$.
We can derive: $cos\phi_0 \,=\, y_0/r$, $cos\psi_0 \,=\, y_0/s$, $cos(\phi_0 + \psi_0) \,=\, -y_0/rs$, $sin\phi_0 \,=\, 1/r\sqrt{r^2 - y_0^2}$, $sin\psi_0 \,=\, 1/s\sqrt{s^2 - y_0^2}$ and so:

$$y_0 \,=\, -rs\,cos(\phi_0 + \psi_0) \,=\, rs\,sin\phi_0\,sin\psi_0 \,-\, rs\,cos\phi_0\,cos\psi_0 \,=$$

$$=\, \sqrt{r^2 - y_0^2}\,\sqrt{s^2 - y_0^2} \,-\, y_0^2$$

this leads to

$$y_0^2 + y_0 \,=\, \sqrt{r^2 - y_0^2}\,\sqrt{s^2 - y_0^2}$$

$$(y_0^2 + y_0)^2 = (r^2 - y_0^2)(s^2 - y_0^2)$$

$$y_0^4 + 2y_0^3 + y_0^2 - y_0^4 + r^2 y_0^2 + s^2 y_0^2 - r^2 s^2 = 0$$

$$2y_0^3 + y_0^2 (1 + r^2 + s^2) - r^2 s^2 = 0 \qquad (12.2)$$

where we choose the solution $0 < y_0 < rs < 1$, which is unique for $0 < r < 1$ and $0 < s < 1$. From this y_0 we can easily determine $\phi_0 = arccos(y_0/r)$ and $\psi_0 = arccos(y_0/s)$.

We can now compute:

$$c_0 = 1 + r\cos\phi_0 + s\cos\psi_0 + rs\cos(\phi_0 + \psi_0) =$$

$$= 1 + y_0 + y_0 - y_0 = 1 + y_0 > 1$$

$$\rho_0 = r\sin\phi_0 + s\sin\psi_0 + rs\sin(\phi_0 + \psi_0) =$$

$$= \sqrt{r^2 - y_0^2} + \sqrt{s^2 - y_0^2} + \sqrt{r^2 s^2 - y_0^2}$$

So finally we can give two important results:

$$2y_0^3 + y_0^2 (1 + r^2 + s^2) - r^2 s^2 = 0$$

and

$$1 < c_0 = 1 + y_0 < 1 + rs$$

We will use these reults later.
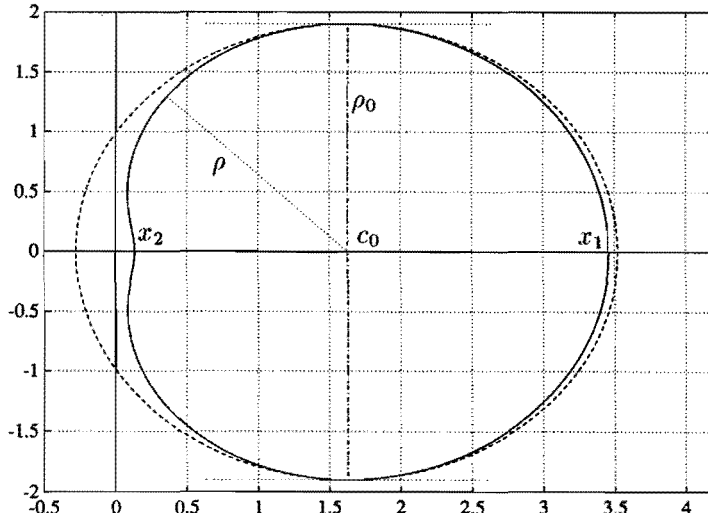
Fig.A3: The boundary function $h_b(\psi)$ and distance ρ to c_0.

Now we are interested in the maximum distance

$$\rho_{max} = \max_{\phi,\psi} \rho(\phi,\psi) = \max_{\phi,\psi} |h(\phi,\psi) - c_0| =$$

$$= \max_{\phi,\psi} |(1 - c_0 + r e^{j\phi}) + s(1 + r e^{j\phi}) e^{j\psi}| \qquad (12.3)$$

For every choice of ϕ we can choose a ψ such that:

$$\rho_{max} = \max_{\phi,\psi} \rho(\phi,\psi) =$$

$$= \max_{\phi,\psi} |(1 - c_0 + r e^{j\phi}) + s(1 + r e^{j\phi}) e^{j\psi}| =$$

$$= \max_{\phi} \{|r e^{j\phi} - (c_0 - 1)| + |s(1 + r e^{j\phi})|\} =$$

$$= \max_{\phi} \left(\sqrt{r^2 + (c_0 - 1)^2 - 2r(c_0 - 1)\cos\phi} + s\sqrt{r^2 + 1 + 2r\cos\phi} \right)$$

Note that the maximization problem over two variables ϕ and ψ has reduced to a maximization problem over only one variable ϕ.

To compute the maximum we put the derivative of $\rho(\phi)$ equal to zero:

$$\frac{d\rho(\phi)}{d\phi} = \frac{r(c_0 - 1)\sin\phi}{\sqrt{r^2 + (c_0 - 1)^2 - 2r(c_0 - 1)\cos\phi}} +$$

$$+ \frac{-rs\sin\phi}{\sqrt{r^2 + 1 + 2r\cos\phi}} =$$

$$= r\sin\phi \left(\frac{(c_0 - 1)}{\sqrt{r^2 + (c_0 - 1)^2 - 2r(c_0 - 1)\cos\phi}} + \right.$$

$$\left. + \frac{-s}{\sqrt{r^2 + 1 + 2r\cos\phi}} \right) = 0$$

so either

$$\sin\phi = 0 \qquad (12.4)$$

or

$$(c_0 - 1)\sqrt{r^2 + 1 + 2r\cos\phi} -$$

$$+ s\sqrt{r^2 + (c_0 - 1)^2 - 2r(c_0 - 1)\cos\phi} = 0 \qquad (12.5)$$

From equation (3) we get the solutions

$$\phi_1 = 0 \quad \text{and} \quad \phi_2 = \pi$$

(We restrict ourselves to $0 \leq \phi < 2\pi$).
$\phi_1 = 0$ and $\phi_2 = \pi$ correspond to the points x_1 and x_2 in Fig.A3.
We know that $c_0 \geq 1$ and from equation (4) we derive

$$(c_0 - 1)^2 (r^2 + 1 + 2r \cos\phi) -$$

$$+ s^2 (r^2 + (c_0 - 1)^2 - 2r(c_0 - 1)\cos\phi) = 0 \qquad (12.6)$$

and so

$$\cos\phi = \frac{(c_0 - 1)^2(r^2 + s^2 + 1) + r^2 s^2}{2r(c_0 - 1)(s^2 + 1 - c_0)} \overset{def}{=} b_0$$

If $|b_0| \leq 1$ we obtain solutions

$$\phi_3 = arccos(b_0) \quad \text{and} \quad \phi_4 = -arccos(b_0) = -\phi_3$$

To see which solutions are maxima we calculate the second derivative of the function $\rho(\phi)$:

$$\frac{d^2 \rho(\phi)}{d\phi^2} = r\cos\phi \left(\frac{(c_0 - 1)}{\sqrt{r^2 + (c_0 - 1)^2 - 2r(c_0 - 1)\cos\phi}} + \right.$$

$$\left. + \frac{-s}{\sqrt{r^2 + 1 + 2r\cos\phi}} \right) +$$

$$+ r\sin\phi \left(\frac{-r(c_0 - 1)^2 \sin\phi}{\sqrt{r^2 + (c_0 - 1)^2 - 2r(c_0 - 1)\cos\phi}} + \right.$$

$$\left. + \frac{-rs\sin\phi}{\sqrt{r^2 + 1 + 2r\cos\phi}} \right)$$

If we substitute the solutions ϕ_3 or ϕ_4 we find the first term to be zero because

$$\frac{(c_0 - 1)}{\sqrt{r^2 + (c_0 - 1)^2 - 2r(c_0 - 1)\cos\phi}} + \frac{-s}{\sqrt{r^2 + 1 + 2r\cos\phi}} = 0$$

and the second term is negative because $(c_0 - 1) > 0$, $r > 0$ and $s > 0$. This means that

$$\frac{d^2 \rho(\phi)}{d\phi^2} < 0$$

for ϕ_3 and ϕ_4 and so $\rho(\phi_3)$ and $\rho(\phi_4)$ are maxima. Consequently $\rho(\phi_1)$ and $\rho(\phi_2)$ must be minima.

We substitute the solutions ϕ_3 in equation (2) and we find the corresponding solution ψ_3 by making:

$$\psi_3 = \arg \frac{r\,e^{j\phi_3} + 1 - c_0}{r\,e^{j\phi_3} + 1}$$

In the same way we can find for $\phi_4 = -\phi_3$ the corresponding $\psi_4 = -\psi_3$. These solutions (ϕ_3, ψ_3) and (ϕ_4, ψ_4) will lead to to the points $h(\phi_3, \psi_3)$ and $h(\phi_4, \psi_4)$ which are a complex conjugated pair, symmetric with respect to the real axis. This points are the maxima of the function $|h(\phi, \psi) - c_0|$ for $c_0 > 1$.

Now we will prove that the points

$$h(\phi_3, \psi_3) = c_0 + j\,\rho_0$$

and

$$h(\phi_4, \psi_4) = c_0 - j\,\rho_0$$

We can prove this by showing that $\phi_3 = \phi_0$: If we choose $\cos\phi = \cos\phi_0 = y_0/r$ and we fill in $c_0 - 1 = y_0$ in equation (5) we get:

$$(c_0 - 1)^2 (r^2 + 1 + 2r\cos\phi) -$$

$$s^2(r^2 + (c_0 - 1)^2 - 2r(c_0 - 1)\cos\phi) =$$

$$= y_0^2(r^2 + 1 + 2y_0) - s^2(r^2 + y_0^2 - 2y_0 y_0) =$$

$$= 2y_0^3 + y_0^2(1 + r^2 + s^2) - r^2 s^2$$

From equation (1) we know that this is zero, so we know that $\phi_3 = \phi_0$ and thus $\psi_3 = \psi_0$.

We proved that the points $c_0 \pm j\,\rho_0$ are maxima of the function $|h(\phi, \psi) - c_0|$ and so the circle $c_0 + \rho_0\,e^{j\alpha}$ is the smallest enclosing circle of $h_b(\psi)$.

$\square$

Appendix A4

Proof of theorem 7.1:

We defined $Y = S(E, X) = F_l(E, X)$.
Now define

$$E_- = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} E^{-1} \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$$

Then there holds: $X = S(E_-, Y) = F_l(E_-, Y)$.

From Packard et al.([5]) we know that for any matrix M with proper dimensions there holds:

$$\max_{\|X\| \le 1} \|S(M, X)\| \le \sigma_{max}(M)$$

We will use this property in the following way:

Let $\|X\| \le 1$, then

$$\|Y\| \le \max_{\|X\| \le 1} \|S(E, X)\| \le \sigma_{max}(E) \le 1$$

and so $\|X\| \le 1 \implies \|Y\| \le 1$

Let $\|Y\| \le 1$, then

$$\|X\| \le \max_{\|Y\| \le 1} \|S(E_-, Y)\| \le \sigma_{max}(E_-) \le 1$$

and so $\|Y\| \le 1 \implies \|X\| \le 1$

Thus:

$$\|Y\| \le 1 \iff \|X\| \le 1$$

$\square$

Appendix A5

Proof of theorem 8.1:

We defined the matrix

$$R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} = S(K_0, \Sigma, \Gamma_0)$$

and

$$\begin{aligned}
R_{12}^l &= (R_{12}^T R_{12})^{-1} R_{12}^T \\
C_0 &= (R_{22} R_{12}^l R_{11} - R_{21})^T \\
A &= C_0^T (C_0^T C_0) R_{22} R_{12}^l \\
T &= S(J_-(A), R)
\end{aligned}$$

Now we define $B_0 = C_0^T (C_0^T C_0)^{-1}$ so that $A = B_0 R_{22} R_{12}^l$
Finally we define the matrices

$$X = S(K_0, \Sigma, \Gamma_0, \Delta_t) = S(R, \Delta_t)$$

and

$$Y = S(J_-(A), X)$$

For this definition of X, Y and orthogonal matrix J_-, Lemma 7.3 claims:

$$\| X \| \leq 1 \quad \Longleftrightarrow \quad \| Y \| \leq 1$$

This results in:

$$\| X \|_\infty \leq 1 \quad \Longleftrightarrow \quad \| Y \|_\infty \leq 1 \tag{12.7}$$

We compute

$$Y = S(J_-(A), X) = S(J_-(A), R, \Delta_t) = S(T, \Delta_t)$$

because

$$T = S(J_-(A), R) = \begin{bmatrix} T_0 & T_1 \\ T_2 & T_3 \end{bmatrix}$$

It is easily varified by the definition of $\left(J_-(A)\right)_{22} = A$ that $T_3 = 0$:

$$T_3 = R_{22} + R_{21} A (I - R_{11} A)^{-1} R_{12} =$$

$$= \left\{ R_{22} R_{12}^l (I - R_{11} A) + R_{21} A \right\} (I - R_{11} A)^{-1} R_{12} =$$

$$= \left\{ R_{22} R^l_{12} - R_{22} R^l_{12} R_{11} B_0 R_{22} R^l_{12} + \right.$$

$$\left. + R_{21} B_0 R_{22} R^l_{12} \right\} (I - R_{11} A)^{-1} R_{12} =$$

$$= \left\{ (I - R_{22} R^l_{12} R_{11} B_0 + R_{21} B_0) R_{22} R^l_{12} \right\} (I - R_{11} A)^{-1} R_{12} =$$

$$= \left\{ (I - (R_{22} R^l_{12} R_{11} - R_{21}) B_0) R_{22} R^l_{12} \right\} (I - R_{11} A)^{-1} R_{12} =$$

$$= \left\{ (I - I) R_{22} R^l_{12} \right\} (I - R_{11} A)^{-1} R_{12} = 0$$

The implication (12.7) turns into

$$\| S(K_0 , \Sigma , \Gamma_0 , \Delta_t) \|_\infty \leq 1 \quad \Longleftrightarrow \quad \| T_0 + T_1 \Delta_t T_2 \|_\infty \leq 1$$

$$\square$$

Appendix A6

Proof of theorem 9.1:
The first part of the proof is to show that it hold:

$$S(G_W(\theta), Q) = L_0(Q) + \sum_{i=1}^{n} \theta_i L_i(Q) \tag{12.8}$$

Define

$$\hat{G} = S(\hat{H}, T, V) = \begin{bmatrix} \hat{G}_{11} & \hat{G}_{12} & \hat{G}_{13} & \hat{G}_{14} \\ \hat{G}_{21} & \hat{G}_{22} & \hat{G}_{23} & \hat{G}_{24} \\ \hat{G}_{31} & \hat{G}_{32} & \hat{G}_{33} & \hat{G}_{34} \\ \hat{G}_{41} & \hat{G}_{42} & \hat{G}_{43} & \hat{G}_{44} \end{bmatrix}$$

Then

$$\begin{bmatrix} \hat{G}_{11} & \hat{G}_{12} \\ \hat{G}_{21} & \hat{G}_{22} \end{bmatrix} = \begin{bmatrix} \hat{H}_{11} & \hat{H}_{12} \\ \hat{H}_{21} & \hat{H}_{22} \end{bmatrix} +$$

$$+ \begin{bmatrix} \hat{H}_{14} \\ \hat{H}_{24} \end{bmatrix} T_{11}(I - \hat{H}_{44}T_{11})^{-1}[\hat{H}_{41} \ \hat{H}_{42}] = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

If $[\hat{H}_{41} \ \hat{H}_{42}] = [0 \ 0]$ it follows

$$[\hat{G}_{41} \ \hat{G}_{42}] = V_2 T_{21}(I - \hat{H}_{44}T_{11})^{-1}[\hat{H}_{41} \ \hat{H}_{42}] = [0 \ 0]$$

and so

$$S(\begin{bmatrix} N & 0 \\ 0 & M \end{bmatrix}, \hat{G}, Q) = \hat{G}_{33} + [\hat{G}_{31} \ \hat{G}_{32}] \begin{bmatrix} N & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} \hat{G}_{13} \\ \hat{G}_{23} \end{bmatrix} +$$

$$+ (\hat{G}_{34} + [\hat{G}_{31} \ \hat{G}_{32}] \begin{bmatrix} N & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} \hat{G}_{14} \\ \hat{G}_{24} \end{bmatrix}) Q(I - \hat{G}_{44}Q)^{-1}\hat{G}_{43}$$

resulting in $L_0(Q)$ and $L_i(Q)$:

$$L_0 = W_{\Delta 1} \left(\hat{G}_{33} + [\hat{G}_{31} \ \hat{G}_{32}] \begin{bmatrix} N_0 & 0 \\ 0 & M_0 \end{bmatrix} \begin{bmatrix} \hat{G}_{13} \\ \hat{G}_{23} \end{bmatrix} \right) W_{\Delta 2} +$$

$$+ W_{\Delta 1} \left(\hat{G}_{34} + [\hat{G}_{31} \ \hat{G}_{32}] \begin{bmatrix} N_0 & 0 \\ 0 & M_0 \end{bmatrix} \begin{bmatrix} \hat{G}_{14} \\ \hat{G}_{24} \end{bmatrix} \right) \cdot$$

$$Q(I - \hat{G}_{44}Q)^{-1}\hat{G}_{43}W_{\Delta 2}$$

and

$$L_i = W_{\Delta 1}[\hat{G}_{31} \ \hat{G}_{32}] \cdot$$

$$\begin{bmatrix} N_i & 0 \\ 0 & M_i \end{bmatrix} \left(\begin{bmatrix} \hat{G}_{13} \\ \hat{G}_{23} \end{bmatrix} + \begin{bmatrix} \hat{G}_{14} \\ \hat{G}_{24} \end{bmatrix} Q(I - \hat{G}_{44}Q)^{-1}\hat{G}_{43} \right) W_{\Delta 2}$$

If $\begin{bmatrix} \hat{H}_{14} \\ \hat{H}_{24} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ it follows:

$$\begin{bmatrix} \hat{G}_{14} \\ \hat{G}_{24} \end{bmatrix} = \begin{bmatrix} \hat{H}_{14} \\ \hat{H}_{24} \end{bmatrix} (I - \hat{H}_{44}T_{11})^{-1}T_{21}V_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

and so

$$S(\begin{bmatrix} N & 0 \\ 0 & M \end{bmatrix}, \hat{G}, Q) = \hat{G}_{33} + [\hat{G}_{31} \ \hat{G}_{32}] \begin{bmatrix} N & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} \hat{G}_{13} \\ \hat{G}_{23} \end{bmatrix} +$$

$$+ \hat{G}_{34}Q(I - \hat{G}_{44}Q)^{-1} \left(\hat{G}_{43} + [\hat{G}_{41} \ \hat{G}_{42}] \begin{bmatrix} N & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} \hat{G}_{13} \\ \hat{G}_{23} \end{bmatrix} \right)$$

resulting in $L_0(Q)$ and $L_i(Q)$:

$$L_0 = W_{\Delta 1} \left(\hat{G}_{33} + [\hat{G}_{31} \ \hat{G}_{32}] \begin{bmatrix} N_0 & 0 \\ 0 & M_0 \end{bmatrix} \begin{bmatrix} \hat{G}_{13} \\ \hat{G}_{23} \end{bmatrix} + \right) W_{\Delta 2} +$$

$$W_{\Delta 1} \hat{G}_{34} Q (I - \hat{G}_{44} Q)^{-1} \left(\hat{G}_{43} + [\hat{G}_{41} \ \hat{G}_{42}] \begin{bmatrix} N_0 & 0 \\ 0 & M_0 \end{bmatrix} \begin{bmatrix} \hat{G}_{13} \\ \hat{G}_{23} \end{bmatrix} \right) W_{\Delta 2}$$

$$L_i = W_{\Delta 1} \left([\hat{G}_{31} \ \hat{G}_{32}] + \hat{G}_{34} Q (I - \hat{G}_{44} Q)^{-1} [\hat{G}_{41} \ \hat{G}_{42}] \right) \cdot$$

$$\begin{bmatrix} N_i & 0 \\ 0 & M_i \end{bmatrix} \begin{bmatrix} \hat{G}_{13} \\ \hat{G}_{23} \end{bmatrix} W_{\Delta 2}$$

The second part of the proof is to show that the function

$$J\left(\gamma(\theta, z) \right) = J(\max_{Q \in \mathbf{Q}} \| S(G_W(\theta), Q) \|) =$$

$$= J\left(\max_{Q \in \mathbf{Q}} \| L_0(Q) + \sum_{i=1}^{n} \theta_i L_i(Q) \| \right) \tag{12.9}$$

is convex over θ:

$$J\left(\gamma(\lambda \theta_a + (1 - \lambda)\theta_b, z) \right) =$$

$$= J\left(\max_{Q \in \mathbf{Q}} \| L_0(Q) + \sum_{i=1}^{n} (\lambda \theta_{ai} + (1 - \lambda)\theta_{bi}) L_i(Q) \| \right) \leq$$

$$\leq J\left(\max_{Q \in \mathbf{Q}} \left(\| \lambda L_0(Q) + \sum_{i=1}^{n} \lambda \theta_{ai} L_i(Q) \| + \right. \right.$$

$$+ \left\|(1-\lambda)L_0(Q) + \sum_{i=1}^{n}(1-\lambda)\theta_{bi}L_i(Q)\right\|\Bigg)\Bigg) \le$$

$$\le J\left(\max_{Q\in\mathbf{Q}}\left\|\lambda L_0(Q) + \sum_{i=1}^{n}\lambda\theta_{ai}L_i(Q)\right\| + \right.$$

$$\left. + \max_{Q\in\mathbf{Q}}\left\|(1-\lambda)L_0(Q) + \sum_{i=1}^{n}(1-\lambda)\theta_{bi}L_i(Q)\right\|\right) \le$$

$$\le J\left(\max_{Q\in\mathbf{Q}}\left\|\lambda L_0(Q) + \sum_{i=1}^{n}\lambda\theta_{ai}L_i(Q)\right\|\right) +$$

$$+ J\left(\max_{Q\in\mathbf{Q}}\left\|(1-\lambda)L_0(Q)) + \sum_{i=1}^{n}(1-\lambda)\theta_{bi}L_i(Q)\right\|\right) =$$

$$= \lambda J\Big(\gamma(\theta_a, z)\Big) + (1-\lambda)J\Big(\gamma(\theta_b, z)\Big)$$

$\square$

Appendix A7

Derivations of sub-gradients:

We have the definition:

$$p(z) \overset{def}{=} \text{Re} \begin{bmatrix} u_0(z)^* L_1(Q_0, z) v_0(z) \\ u_0(z)^* L_2(Q_0, z) v_0(z) \\ \vdots \\ u_0(z)^* L_n(Q_0, z) v_0(z) \end{bmatrix}$$

Now also define:

$$\hat{c} \overset{def}{=} \begin{bmatrix} \gamma(\hat{\theta}, z_1) \\ \gamma(\hat{\theta}, z_2) \\ \vdots \\ \gamma(\hat{\theta}, z_N) \end{bmatrix} \quad \text{and} \quad c \overset{def}{=} \begin{bmatrix} \gamma(\theta, z_1) \\ \gamma(\theta, z_2) \\ \vdots \\ \gamma(\theta, z_N) \end{bmatrix}$$

$$\hat{c}_{ss} \overset{def}{=} \begin{bmatrix} \alpha \hat{c} \\ \beta \gamma(\hat{\theta}, z_0) \end{bmatrix} \quad \text{and} \quad c_{ss} \overset{def}{=} \begin{bmatrix} \alpha \hat{c} \\ \beta \gamma(\theta, z_0) \end{bmatrix}$$

$$q \overset{def}{=} \text{Re} \begin{bmatrix} p(z_1) & p(z_2) & \cdots & p(z_N) \end{bmatrix} \quad \text{and} \quad q_{ss} \overset{def}{=} \begin{bmatrix} q & \text{Re}\, p(z_0) \end{bmatrix}$$

We start with a look on the function $\gamma(\theta, z)$. For each $z \in \Omega$ we can derive:

$$\gamma(\theta, z) = \max_{Q \in \mathbf{Q}} \left\| L_0(Q, z) + \sum_{i=1}^{n} \theta_i L_i(Q, z) \right\| \geq$$

$$\geq \left\| L_0(Q_0, z) + \sum_{i=1}^{n} \theta_i L_i(Q_0, z) \right\| \geq$$

$$\geq \text{Re} \left\{ u_0^*(z) \left(L_0(Q_0, z) + \sum_{i=1}^{n} \theta_i L_i(Q_0, z) \right) v_0(z) \right\} =$$

$$= \text{Re} \left\{ u_0^*(z) \left(L_0(Q_0, z) + \sum_{i=1}^{n} \hat{\theta}_i L_i(Q_0, z) + \right. \right.$$

$$\left. \left. + \sum_{i=1}^{n} (\theta_i - \hat{\theta}_i) L_i(Q_0, z) \right) v_0(z) \right\} =$$

$$= \max_{Q \in \mathbf{Q}} \left\| L_0(Q_0, z) + \sum_{i=1}^{n} \hat{\theta}_i L_i(Q_0, z) \right\| +$$

$$+ \sum_{i=1}^{n}(\hat{\theta}_i - \theta_i)\mathrm{Re}\left\{u_0^*(z)L_i(Q_0, z)v_0(z)\right\} =$$

$$= \gamma(\hat{\theta}, z) + \sum_{i=1}^{n}(\hat{\theta}_i - \theta_i)\mathrm{Re}\left\{u_0^*(z)L_i(Q_0, z)v_0(z)\right\} =$$

$$= \gamma(\hat{\theta}, z) + (\hat{\theta}^T - \theta^T)\, p(z)$$

so we can derive the subgradient g_∞ :

$$\|\gamma(\theta, z)\|_\infty \geq \|\gamma(\theta, z_0)\| \geq$$

$$\geq \left\|\gamma(\hat{\theta}, z_0) + \sum_{i=1}^{n}(\hat{\theta}_i - \theta_i)\mathrm{Re}\left\{u_0^*(z_0)L_i(Q_0, z_0)v_0(z_0)\right\}\right\| \geq$$

$$\geq \|\gamma(\hat{\theta}, z_0)\| + \sum_{i=1}^{n}(\hat{\theta}_i - \theta_i)\mathrm{Re}\left\{u_0^*(z_0)L_i(Q_0, z_0)v_0(z_0)\right\} =$$

$$= \|\gamma(\hat{\theta}, z)\|_\infty + (\hat{\theta}^T - \theta)^T\, p(z_0)$$

This gives the sub-gradient

$$g_\infty = p(z_0)$$

Now we will concentrate on a sub-gradient for g_2. First we derive:

$$c = \begin{bmatrix} \gamma(\theta, z_1) \\ \gamma(\theta, z_2) \\ \vdots \\ \gamma(\theta, z_N) \end{bmatrix} \geq \begin{bmatrix} \gamma(\hat{\theta}, z_1) \\ \gamma(\hat{\theta}, z_2) \\ \vdots \\ \gamma(\hat{\theta}, z_N) \end{bmatrix} + \mathrm{Re}\begin{bmatrix} p(z_1)^T \\ p(z_2)^T \\ \vdots \\ p(z_N)^T \end{bmatrix}(\hat{\theta} - \theta) =$$

$$= \hat{c} + q^T(\hat{\theta} - \theta)$$

Next we can derive:

$$\{J_2(\theta, z)\}^2 = \sum_z \gamma^2(\theta, z) = c^T c \geq$$

$$\geq \left(\hat{c} + q^T(\hat{\theta} - \theta)\right)^T\left(\hat{c} + q^T(\hat{\theta} - \theta)\right) =$$

$$= \hat{c}^T\hat{c} + 2(\hat{\theta} - \theta)^T p(z)\hat{c} + (\hat{\theta} - \theta)^T q\, q^T(\hat{\theta} - \theta) \geq$$

$$\geq \hat{c}^T \hat{c} + 2(\hat{\theta} - \theta)^T q \hat{c} + (\hat{c}^T \hat{c})^{-1} (\hat{\theta} - \theta)^T q \hat{c} \hat{c}^T q^T (\hat{\theta} - \theta) =$$

$$= \left((\hat{c}^T \hat{c})^{1/2} + (\hat{\theta} - \theta)^T q \hat{c} (\hat{c}^T \hat{c})^{-1/2} \right)^2$$

and so

$$J_2(\theta, z) \geq = (\hat{c}^T \hat{c})^{1/2} + (\hat{\theta} - \theta)^T q \hat{c} (\hat{c}^T \hat{c})^{-1/2} =$$

$$= J_2(\hat{\theta}, z) + (\hat{\theta} - \theta)^T \left(\sum_{z \in \Omega} p(z) \gamma(\hat{\theta}, z) J_2\big(\gamma(\hat{\theta}, z)\big) \right)$$

and so

$$g_2 = \left(\sum_{z \in \Omega} p(z) \gamma(\hat{\theta}, z) \right) / J_2\big(\gamma(\hat{\theta}, z)\big)$$

For the sub-gradient g_s for J_s we can derive:

$$J_s\big(\gamma(\theta, z)\big) = \alpha J_2\big(\gamma(\theta, z)\big) + \beta J_\infty\big(\gamma(\theta, z)\big) \geq$$

$$\alpha J_2\big(\gamma(\theta, z)\big) + \alpha g_2^T (\hat{\theta} - \theta) + \beta J_\infty\big(\gamma(\theta, z)\big) + \beta g_\infty^T (\hat{\theta} - \theta) =$$

$$= J_s\big(\gamma(\theta, z)\big) + \alpha g_2^T (\hat{\theta} - \theta) + \beta g_\infty^T (\hat{\theta} - \theta)$$

and so

$$g_s = \alpha g_2 + \beta g_\infty$$

The subgradient g_{ss} for J_{ss} can be derived in the same way as the sub-gradient for J_2:

$$\{ J_{ss}(\theta, z) \}^2 = \alpha^2 \sum_z \gamma^2(\theta, z) + \beta^2 \gamma^2(\theta, z_0) = c_{ss}^T c_{ss} \geq$$

$$\geq \left(\hat{c}_{ss} + q_{ss}^T (\hat{\theta} - \theta) \right)^T \left(\hat{c}_{ss} + q_{ss}^T (\hat{\theta} - \theta) \right) =$$

$$= \hat{c}_{ss}^T \hat{c}_{ss} + 2(\hat{\theta} - \theta)^T p_{ss}(z) \hat{c}_{ss} + (\hat{\theta} - \theta)^T q_{ss} q_{ss}^T (\hat{\theta} - \theta) \geq$$

$$\geq \hat{c}_{ss}^T \hat{c}_{ss} + 2(\hat{\theta} - \theta)^T q_{ss} \hat{c}_{ss} + (\hat{c}_{ss}^T \hat{c}_{ss})^{-1} (\hat{\theta} - \theta)^T q \hat{c}_{ss} \hat{c}_{ss}^T q_{ss}^T (\hat{\theta} - \theta) =$$

$$= \left((\hat{c}_{ss}^T \hat{c}_{ss})^{1/2} + (\hat{\theta} - \theta)^T q_{ss} \hat{c}_{ss} (\hat{c}_{ss}^T \hat{c}_{ss})^{-1/2} \right)^2$$

and so

$$J_{ss}(\theta, z) \geq = (\hat{c}_{ss}^T \hat{c}_{ss})^{1/2} + (\hat{\theta} - \theta)^T q_{ss} \hat{c}_{ss} (\hat{c}_{ss}^T \hat{c}_{ss})^{-1/2} =$$

$$J_{ss}(\hat{\theta}, z) + (\hat{\theta} - \theta)^T \left(\alpha^2 \sum_{z \in \Omega} p(z) \gamma(\hat{\theta}, z) \beta^2 p(z_0) \gamma(\hat{\theta}, z_0) \right) / J_{ss}\left(\gamma(\hat{\theta}, z) \right)$$

and so

$$g_{ss} = \left(\alpha^2 \sum_{z \in \Omega} p(z) \gamma(\hat{\theta}, z) \beta^2 p(z_0) \gamma(\hat{\theta}, z_0) \right) / J_{ss}\left(\gamma(\hat{\theta}, z) \right)$$

For the sub-gradient g_{max} for J_{max} we can distinguish two cases:

(1). Suppose $\alpha J_2 \geq \beta J_\infty$:

$$J_{max}\left(\gamma(\theta, z) \right) = \max \left(\alpha J_2\left(\gamma(\hat{\theta}, z) \right), \beta J_\infty\left(\gamma(\hat{\theta}, z) \right) \right) \geq$$

$$\geq \alpha J_2\left(\gamma(\hat{\theta}, z) \right) \geq$$

$$\geq J_2\left(\gamma(\theta, z) \right) + \alpha g_2^T(\hat{\theta} - \theta) =$$

$$= J_{max}\left(\gamma(\theta, z) \right) + \alpha g_2^T(\hat{\theta} - \theta) =$$

(2). Suppose $\alpha J_2 < \beta J_\infty$:

$$J_{max}\left(\gamma(\theta, z) \right) = \max \left(\alpha J_2\left(\gamma(\hat{\theta}, z) \right), \beta J_\infty\left(\gamma(\hat{\theta}, z) \right) \right) \geq$$

$$\geq \beta J_\infty\left(\gamma(\hat{\theta}, z) \right) \geq$$

$$\geq J_\infty\left(\gamma(\theta, z) \right) + \beta g_\infty^T(\hat{\theta} - \theta) =$$

$$= J_{max}\left(\gamma(\theta, z) \right) + \beta g_\infty^T(\hat{\theta} - \theta) =$$

and so

$$g_{max} = \begin{cases} \alpha g_2 & \text{for } \alpha J_2 \geq \beta J_\infty \\ \beta g_\infty & \text{for } \alpha J_2 < \beta J_\infty \end{cases}$$

□

Appendix B: Assumptions

In this appendix we summerize the assumptions that are made on the true process, the model, the model error, the experimental conditions and the disturbances.

True process:
(1). $P_t(z) \in R\!P$.
(2). $P_t(z)$ has p inputs and q outputs.
(3). $P_t(z)$ is stabilized by a controller $K_{ana}(z) \in R\!P$.

The stabilizing controller has a normalized lcf $K_{ana}(z) = M_C(z)^{-1} N_C(z)$. For $P_t(z) \in R\!H_\infty$ we may choose $K_{ana}(z) = 0$.

Model:
(1). $P(\theta, z) \in R\!P$ has p inputs and q outputs and is in a model set **P**.
(2). $P(\theta, z)$ is parametrized by the parameter vector $\theta \in \Theta$.
(3). There is a bijective mapping from Θ to **P**.
(4). $P(\theta, z)$ is also stabilized by the controller $K_{ana}(z) \in R\!P$.

For each model in **P** there is a unique left coprime factorization $P(\theta, z) = M(\theta, z)^{-1} N(\theta, z)$ defined by the bijective mapping π :

$$[\, N(\theta, z) \quad M(\theta, z)\,] \; = \; \pi\left(P(\theta, z)\right)$$

Model error:
We introduce the model error as in the coprime factor configuration of Fig.5.3.
Let $P_t(z)$, $N(z)$, $M(z)$, $N_C(z)$ and $M_C(z)$ be defined as above.
Then there will exist a function $\Delta_t(z) \in R I\!H_\infty$ such that

$$P_t(z) \; = \; (\, M(z) \, - \, N_C(z)\, \Delta_t(z) \,)^{-1} (\, N(z) \, + \, M_C(z)\, \Delta_t(z)\,)$$

Let $h_\Delta(k)$ be the Markov parameters of $\Delta_t(z)$.
We assume:

$$\sum_{k=l_\Delta+1}^{\infty} \| \, h_\Delta(k) \, \| \; \leq \; \epsilon_\Delta$$

where $l_\Delta > 0$, $\epsilon_\Delta > 0$.

Experiments
Consider the true process in closed loop with a stabilizing controller $K_{exp}(z)$ as in the configuration of Fig.2.4. The controller $K_{exp}(z)$ is not necessarilly the same controller as the stabilizing controller $K_{ana}(z)$.
The system is excited by the reference signal $r(k)$ and measured in the input signal $u(k)$ and output signal $y(k)$, The true noise signals $d_t(k)$ and $e_t(k)$ are unknown.
Let l_1 be the impulse response length of the function $(\, I + P_t\, K_{exp}\,)\, P_t$ and let l_2 be the impulse response length of the function $(\, I + P_t\, K_{exp}\,)$.
We do p experiments (so equal to the number of inputs) with an observation interval $k = 0, .., 2N - 1$ where:

$$2N \gg l_\Delta \;\; , \quad 2N \gg l_1 \;\; \text{and} \;\; 2N \gg l_2$$

For every experiment we get the data set $\{\, r_i(k)\, , \, u_i(k)\, , \, y_i(k)\,\}$ and so $r_i(k)$, $u_i(k)$, $y_i(k)$ are in $I\!L_2[0, 2N-1]$. The true noise signals $d_{ti}(k) \in I\!L_2[0, 2N-1]$ and $e_{ti}(k) \in I\!L_2[0, 2N - 1]$ are unknown.
We compute the z-transformation of all signals and we obtain for the i-th experiment: $r_i(z)\, , \, u_i(z)\, , \, y_i(z)\, , \, d_{ti}(k)\, , \, e_{ti}(k)$
We define

$$
\begin{aligned}
U(z) &= [\, u_1(z), u_2(z), .., u_p(z)\,] \\
Y(z) &= [\, y_1(z), y_2(z), .., y_p(z)\,] \\
R(z) &= [\, r_1(z), r_2(z), .., r_p(z)\,] \\
D_t(z) &= [\, d_{t1}(z), d_{t2}(z), .., d_{tp}(z)\,] \\
E_t(z) &= [\, e_{t1}(z), e_{t2}(z), .., e_{tp}(z)\,]
\end{aligned}
$$

The vectors $r_1(z)$, $r_2(z),\ldots,r_p(z)$ are assumed to be linearly independent for all $z \in C$ so that the matrix $R(z)$ is invertible for all $z \in C$.

Bounds for the disturbances :
Assume that for the every experiment the noise on the k-th input is bounded by a known filter $W_{dk}(z)$, and that the noise on the l-th output is bounded by a known filter $W_{el}(z)$ with

$$
|d_{tk}(z)| \leq |W_{dk}(z)| \quad \text{and} \quad |e_{tl}(z)| \leq |W_{el}(z)| \quad \text{for } z \in C
$$

Construct the diagonal matrix

$$
W(z) = \text{diag}(\, W_{d1}(z), W_{d2}(z), \cdots, W_{dp}(z),\ W_{e1}(z), W_{e2}(z), \cdots, W_{eq}(z)\,)
$$

and the matrices

$$
V_1 = \begin{bmatrix} (I_p \otimes 1_p) & 0 \\ 0 & (I_q \otimes 1_p) \end{bmatrix} \quad \text{and} \quad V_2 = (I_p \otimes [\, 1_p\ 1_q\,]^T)
$$

structured description :
The true disturbance signal can be written as

$$
F_t(z) = \begin{bmatrix} D_t(z) \\ E_t(z) \end{bmatrix} = W(z)\, V_1\, \tilde{Q}_{st}(z)\, V_2
$$

where $\tilde{Q}_{st}(z)$ is diagonal $\|\, \tilde{Q}_{st}(z)\,\|_\infty \leq 1\,$ }

unstructured description :
The true disturbance signal can be written as

$$
F_t(z) = \begin{bmatrix} D_t(z) \\ E_t(z) \end{bmatrix} = W(z)\, \tilde{Q}_{ut}(z)
$$

where $\|\, \tilde{Q}_{ut}(z)\,\|_\infty \leq \sqrt{(p+q)p}\,$ }

Appendix C: List
of symbols

$\mathbb{R}$	set of real numbers
$\mathcal{C}$	unit circle in the complex plane
$\mathbb{L}_2(-\infty, \infty)$	time domain Lebesgue space
$R\mathbb{P}$	set of all finite dimensional real rational transfer functions
$R\mathbb{L}_\infty$	set of $R\mathbb{P}$-functions that have full rank on $\mathcal{C}$
$R\mathbb{H}_\infty$	set of all stable $R\mathbb{P}$-functions
Θ	parameter set
$\mathbf{P}$	model set
$\tilde{\mathbf{P}}$	set of all systems that do not falsify the data
Ω	set of observation frequencies
$\bar{x}(k)$	discrete time domain signal
$x(z)$	z-transform of $\bar{x}(k)$
τ	backward shift operator
z	elements on the unit circle $\mathcal{C}$

211

$\| \cdot \|$	$= \sigma_{max}(\,\cdot\,)$, matrix-norm
$\| \cdot \|_\infty$	infinity norm
$\| \cdot \|_2$	two norm
$\| \cdot \|_{s\infty}$	infinity norm for the sampled frequency domain
$\| \cdot \|_{s2}$	two norm for the sampled frequency domain
$H^*(s)$	$= H^T(-s)$ for $z \in C$, complex conjugate
θ	model parameters
$P_t(z)$	true process ($P_t \in R\!P$)
$P(z)$	model ($P \in \mathbf{P}$)
$\Delta_t(z)$	model error ($\Delta_t \in R\!H_\infty$)
N_t, M_t	coprime factors of true process $P_t(z)$
N, M	coprime factors of model $P(z)$
N_C, M_C	coprime factors of controller $K(z)$
u, y	measured input and output signal
u_t, y_t	true input and output signal
d_t, e_t	true input and output distrubances
U, Y	matrices with measured input and output signals for multiple experiments
U_t, Y_t	matrices with true input and output signals for multiple experiments
D_t, E_t	matrices with true input and output disturbances for multiple experiments
Q_t	matrix with scaled true disturbance signals
π_t	bijective mapping form $R\!P$ to $R\!H_\infty \times R\!H_\infty$
π	bijective mapping form $\mathbf{P}$ to $R\!H_\infty \times R\!H_\infty$
l_1	maximum length of the impulse response of $(I + P_t K_{exp})^{-1} P_t$
l_2	maximum length of the impulse response of $(I + P_t K_{exp})^{-1}$
l_Δ	maximum length of the impulse response of $\Delta_t(z)$
$2N$	length of the onservation interval
p	number of input signals of the process
q	number of output signals of the process
n	number of model parameters

Bibliography

[1] B.D.O. Anderson. Algebraic properties of minimal degree spectral factors. *Automatica*, 9:pp491–500, 1973.

[2] L.J.J.M. Ariaans. Inequality constrained optimization in fortran and matlab in aid of robust identification. Int. Report, Group Measurement and Control, Faculty of Electircal Engineering, Eindhoven University of Technology, The Netherlands (In Dutch), 1991.

[3] A.C.P.M. Backx. Identification of an industrial process: A Markov parameter approach. PhD-thesis, Eindhoven University of Technology, The Netherlands, 1987.

[4] A.C.P.M. Backx and A.A.H. Damen. Identification of industrial MIMO processes for fixed controllers. *Journal A*, 30:pp3–12, 1989.

[5] G.J. Balas, J.C. Doyle, K. Glover, A. Packard, and R. Smith. *μ-analysis and synthesis toolbox (μ-tools)*. Manual software-package, MUSYN Inc. and The Math. Works Inc., 1991.

[6] S.P. Boyd and C.H. Barratt. *Linear Controller Design, limits of performance*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1991.

[7] J.B. Cruz jr, J.S. Freudenberg, and D.P. Looze. A relationship between sensitivity and stability of multivariable feedback systems. *IEEE Trans. Automatic Control*, AC-26:pp66–74, 1981.

[8] D.K. De Vries and P.M.J. Van den Hof. Quantification of model uncertainty from data: input design, interpolation, and connection with robust control design specifications. In *Proceedings of the American Control Conference, Chicago, Illinois, USA*, pages pp3170–3175, 1992.

[9] V.F. Demyanov and L.V. Vasilev. *Nondifferentiable Optimization*. Optimization Software (Springer-Verlag), 1985.

[10] J.C. Doyle. Analysis of feedback systems with structured uncertainties. *IEE Proceedings*, vol. 129, Part D(no.6):pp242–250, 1982.

[11] J.C. Doyle. Lecture notes. ONR/Honeywell Workshop on Advances in Multivariable Control, 1984.

[12] J.C. Doyle, B.A. Francis, and A.R. Tannenbaum. *Feedback Control Systems*. MacMillan Publishing Company, New York, USA, 1992.

[13] J.C. Doyle, K. Glover, P.P. Khargonekar, and B.A. Francis. State-space solutions to standard H_2 and H_∞ control problems. *IEEE Trans. Automatic Control*, AC-34:pp831–847, 1989.

[14] J.C. Doyle, A. Packard, and K. Zhou. Revieuw on LFT's, LMI's and μ. In *Proceedings of the 30th Conference on Decision and Control, Brighton, UK*, pages pp1227–1232, 1991.

[15] J.C. Doyle and G. Stein. Multivariable feedback design: concepts for a classical-modern synthesis. *IEEE Trans. Automatic Control*, AC-26:pp4–16, 1981.

[16] P. Eykhoff. *System Identification: Parameter and State Estimation*. Wiley and Sons, London, 1974.

[17] F.F.W.M. Fokkelman. An angle feedback loop for the ball balancing process. Int. Report, Group Measurement and Control, Faculty of Electircal Engineering, Eindhoven University of Technology, The Netherlands (In Dutch), 1991.

[18] B.A. Francis. *A course in H_∞-control theory*. Lecture notes in Control and Information sciences, vol.88, 1987.

[19] J.S. Freudenberg, D.P. Looze, and J.B. Cruz jr. Robustness analysis using singular value sensitivities. *International Journal of Control*, 35(1):pp95–116, 1982.

[20] M. Gevers. Connecting identification and robust control: a new challenge. In *Proceedings 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation, Budapest, Hungary*, pages 1–10, 1991.

[21] K. Glover. All optimal hankel-norm approximations of linear multivariable systems and their L_∞ error bounds. *International Journal of Control*, 39(6):pp1115–1193, 1984.

[22] K. Glover and D. McFarlane. Robust stabilization of normalized coprime factor plant descriptions with H_∞-bounded uncertainty. *IEEE Trans. Automatic Control*, AC-34:pp821–830, 1989.

[23] G.C. Goodwin, B. Ninness, and M.E. Salgado. Quantification of uncertainty in estimation. In *Proceedings of Conference on Decision and Control, Tampa, Florida, USA*, 1989.

[24] G.C. Goodwin and M.E. Salgado. A stochastic embedding approach for quantifying uncertainty in the estimation of restricted complexity models. *International Journal of Adaptive Control and Signal Processing*, (3):pp333–356, 1990.

[25] M. Groetschel, L. Lovasz, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization, Volume 2 of Algorithms and Combinatorics*. Springer-Verlag, 1988.

[26] G. Gu and P.P Khargonekar. A class of algorithms for identification in H_∞. *Automatica*, 28(2):pp229–312, 1992.

[27] F. Hansen and G. Franklin. On a fractional representation approach to closed-loop experiment design. In *Proc. 7th ACC, Stanford, CA*, pages pp1319–1320, 1988.

[28] F. Hansen, G. Franklin, and R. Kosut. Closed-loop identification via the fractional representation: experiment design. In *Proc. 8th ACC, Pittsburgh, PA*, pages pp1422–1427, 1989.

[29] A.J. Helmicki, C.A. Jacobsen, and C.N. Nett. H_∞ identification of stable LSI systems: a scheme with direct application to controller design. In *Proc. 8th ACC, Pittsburgh, PA*, pages pp1625–1632, 1989.

[30] A.J. Helmicki, C.A. Jacobsen, and C.N. Nett. Control oriented system identification: A worst case/deterministic approach in H_∞. *IEEE Trans. Automatic Control*, AC-36(10):pp1163–1176, October 1991.

[31] P.S.C. Heuberger. On approximate system identification with system based orthogonal functions. PhD-thesis, Delft University of Technology, The Netherlands, 1991.

[32] P. Janssen. On model parametrization and model structure selection for identification of MIMO-systems. PhD-thesis, Eindhoven University of Technology, The Netherlands, 1988.

[33] H.W. Joosten. A user-program for the ball balancing process. Int. Report, Group Measurement and Control, Faculty of Electircal Engineering, Eindhoven University of Technology, The Netherlands (In Dutch), 1990.

[34] T. Kailath. *Linear Systems*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1980.

[35] R.O. LaMaire, L. Valavani, M. Athans, and G. Stein. A frequency-domain estimator for use in adaptive control systems. *Automatica*, 27:pp23–38, 1991.

[36] W.R.H.M. Liebregts. Modelling of a roller-pump. Int. Report, Group Measurement and Control, Faculty of Electircal Engineering, Eindhoven University of Technology, The Netherlands, 1991.

[37] W.R.H.M. Liebregts. A new identification technique for H_∞-robust control design: Identification of the water-vessel process. MSc-thesis, Faculty of Electircal Engineering, Eindhoven University of Technology, The Netherlands, 1991.

[38] M. Lilja. Controller design by frequency domain approximation. PhD-thesis, Lund institute of Technology, Sweden, 1988.

[39] L. Ljung. *System Identification - Theory for the User*. Prentice-Hall, Englewood Cliffs, New Jersey, 1987.

[40] J. Lunze. *Robust Multivariable Feedback Control*. Prentice Hall, Englewood Cliffs, New Yersey, USA, 1988.

[41] J.M. Maciejowski. *Multivariable Feedback Control Design*. Addison-Wesley Publishers, Wokingham, England, 1989.

[42] M.C. Maciejowski. Balanced realizations in system identification. In *Proceedings 7th IFAC Symposium on Identification and System Parameter Estimation, York, UK*, 1985.

[43] D. McFarlane. Robust controller design using normalized coprime factor plant perturbations. PhD-thesis, Queens' College, University of Cambridge, UK, 1988.

[44] B.C. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Trans. Automatic Control*, AC-26:pp17–31, 1981.

[45] R. Ober. Balanced realizations: Canonical form, parametrization, model reduction. *International Journal of Control*, (46(2)):pp643–670, 1987.

[46] R. Ober and D. McFarlane. Balanced canonical forms for minimal systems: A normalized coprime factor approach. *Linear Algebra and its Applications*, (122/123/124):pp23–64, October 1989.

[47] A. Packard, G.J. Balas, J.C. Doyle, K. Glover, and R. Smith. Control design with μ-tools. in preparation, March 1991.

[48] I. Postlethwaite and Y.K. Foo. Representations of uncertainty and robustness tests for multivariable feedback systems. In *Tzafestas S.G.(Ed).,Multivariable Control, Riedel Publ.Co.*, pages pp151–160, 1984.

[49] I. Postlethwaite and Y.K. Foo. Robustness with simultaneous pole and zero movement across the $j\omega$-axis. *Automatica*, 21(4):pp433–443, 1985.

[50] R.M. Redheffer. Inequalities for a matrix Riccati equation. *Journal Mathematics and Mechanics*, vol.8(no.3), 1959.

[51] L.E. Scales. *Introduction to Non-Linear Optimization*. MacMillan Publishers LTD, London, UK, 1985.

[52] R.J.P. Schrama. Application of the fractional representation approach in identification: the noiseless case. *Selected topics in Identification, Modelling and Control, Delft University Press*, 1990.

[53] R.J.P. Schrama. A framework for control-oriented approximate identification. In *Proceedings MTNS'91, Kobe, Japan*, 1991.

[54] R.J.P. Schrama. Approximate identification and control design. PhD-thesis, Delft University of Technology, The Netherlands, 1992.

[55] T. Söderström and P. Stoica. *System Identification.* Prentice Hall, Hemel Hempstead, UK, 1989.

[56] T.J.J. Van den Boom. System identification of MIMO-systems for H_∞ robust control. *Technical Report, Cambridge University, UK,* (CUED/F-INFENG/TR.88), 1991.

[57] T.J.J. Van den Boom. System identification for coprime factor plant description with bounded error. In *Proceedings of the American Control Conference, Chicago, Illinois, USA,* pages pp1248–1252, 1992.

[58] T.J.J. Van den Boom. MIMO-systems identification by minimum error bounds. Submitted to the 12th IFAC World Congress, Sydney, Australia, 1993.

[59] T.J.J. Van den Boom, A.A.H. Damen, and M.H. Klompstra. Identification for robust control using an H_∞-norm. *EUT-Report 92-E-261, Department of Electrical Engineering, Eindhoven University of Technology, The Netherlands,* (July), 1992.

[60] T.J.J. Van den Boom, M.H. Klompstra, and A.A.H. Damen. System identification for H_∞ robust control design. In *Proceedings 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation, Budapest, Hungary,* pages 1431–1436, 1991.

[61] P.M.J. Van den Hof. On residual-based parametrization and identification of multivaribale systems. PhD-thesis, Eindhoven University of Technology, The Netherlands, 1989.

[62] M. Vidyasagar. *Control System Synthesis: A Factorization Approach.* The MIT-press, Cambridge, Massachusetts, USA, 1985.

[63] M. Vidyasagar and H. Kimura. Robust controllers for uncertain linear multivariable systems. *Automatica,* 22(1):pp83–94, 1986.

[64] M. Vidyasagar, H. Schneider, and B Francis. Algebraic and topological aspects of feedback stabilization. *IEEE Trans. Automatic Control,* AC-27:pp880–894, 1982.

[65] B. Wahlberg. System identification using Laguerre models. *IEEE Trans. on Automatic Control,* may 1991.

[66] B. Wahlberg and L. Ljung. On estimation of transfer function error bounds. In *Proceedings of the European Control Conference, Grenoble France,* 1991.

[67] Y.-C. Zhu. Black-box identification of MIMO-transfer functions: asymptotic properties of prediction error models. *International Journal of Adaptive Control*, 3:pp357–373, 1989.

[68] Y.-C. Zhu. Estimation of transfer functions: asymptotic theory and a bound of model uncertainty. *International Journal of Control*, 49:pp2241–2258, 1989.

[69] Y.C. Zhu. Identification and control of MIMO industrial processes: An integration approach. PhD-thesis, Eindhoven University of Technology, The Netherlands, 1990.

Samenvatting

Een systeem-identificatie procedure in het frequentie-domein wordt gepresenteerd waarbij een model wordt afgeleid met een begrensde ongestructureerde modelfout.

De verstoringen aan de ingang en de uitgang van het proces worden verondersteld begrensd te zijn in het frequentie-domein, hetgeen impliceert dat er begrenzingsfuncties bestaan voor de absolute waarde van de discrete Fourier transformatie van de ruissignalen.

Allereerst beschouwen we de identificatie van een SISO-proces met een additieve of multiplicatieve modelfout-structuur. We leiden onzekerheidsgebieden af voor de procesdynamica en we schatten een optimaal model door H_∞-fitting.

Vervolgens breiden we de identificatiemethode uit naar MIMO-systemen en beschouwen we verscheidene modelfout-structuren, zoals additieve modelfouten, ingang en uitgang multiplicatieve modelfouten, reversie-type modelfouten en coprieme modelfouten. Al deze modelfout-structuren passen in een basisschema met coprieme factoren.

We laten zien dat voor een vast gekozen model de werkelijke modelfout kan worden geschreven als een functie van een bekende matrix $G(z)$ en een onbekende diagonale matrix $Q_t(z)$. De matrix $G(z)$ is opgebouwd uit bekende informatie, zoals het model, de modelfout-structuur, de gemeten data en de ruis-begrenzingsfuncties. De diagonale matrix $Q_t(z)$ bevat de werkelijke geschaalde ruissignalen. Deze onbekende ruissignalen worden binnen de eenheidsbol verondersteld.

Een bovengrens wordt afgeleid voor de grootste singuliere waarde van de modelfout. We maken gebruik van de theorie van gestructureerde singuliere waarden (μ-analyse) en van de zogenaamde Redheffer sterprodukten. De bovengrens voor de modelfout wordt geminimaliseerd met betrekking tot een of andere norm (H_∞, H_2 of een combinatie).

De keuze van een lineaire parametrisatie zal leiden tot een convex optimaliseringsprobleem en de algorithmen zullen robuust convergent zijn.

Tenslotte wordt een praktijkvoorbeeld gepresenteerd waarbij een laboratoriumproces, bestaande uit een stelsel van vier watervaten met 2 ingangen en 2 uitgangen, wordt geïdentificeerd en een model met een bovengrens voor de modelfout wordt afgeleid.

Curriculum vitae

- 29 augustus 1961 : geboren te Bergen op Zoom.

- juli 1980 : eindexamen havo aan het Moller lyceum te Bergen op Zoom.

- september 1980 - juli 1981 : studie Electrotechniek aan de HTS te Breda.

- september 1981 - augustus 1988 : studie Electrotechniek aan de Technische Universiteit te Eindhoven.

- november 1988 - mei 1990 : tewerkgesteld als erkende gewetensbezwaarde militaire dienst bij de vakgroep Meten en Regelen, Faculteit Electrotechniek aan de Technische Universiteit te Eindhoven.

- juni 1990 - augustus 1992 : werkzaam als toegevoegd onderzoeker bij de vakgroep Meten en Regelen, Faculteit Electrotechniek aan de Technische Universiteit te Eindhoven.

- vanaf september 1992 : werkzaam als universitair docent bij de vakgroep Regeltechniek, Faculteit Electrotechniek aan de Technische Universiteit te Delft.

Stellingen behorende bij het proefschrift:

MIMO SYSTEM IDENTIFICATION FOR H_∞ ROBUST CONTROL

A frequency domain approach with minimum error bounds

van
Ton van den Boom

Eindhoven, 5 januari 1993

1. Een goede manier om van gemeten data van een industrieel proces te komen tot een optimale regelaar voor dat proces, is door gebruik te maken van een iteratieve procedure. Hierbij vindt tijdens elke iteratie achtereenvolgens plaats een H_∞-identificatie, een ontwerp van een robuuste regelaar en een bepaling van de modelfoutverzameling.
   (Dit Proefschrift, pagina's 138-139)

2. De in dit proefschrift behandelde H_∞-identificatiemethode is gebaseerd op een 'slechtste geval' criterium (worst case criterion). Dit impliceert niet dat het gemiddelde gedrag van het model slecht is.
   (Dit Proefschrift, pagina's 54-62 en W.R.H.M. Liebregts: 'A new identification technique for H_∞ robust control design,' Afstudeerverslag, ER, TUE, 1991)

3. Een dataverzameling met begrenzingen op verstoringen leidt tot een verzameling $\tilde{\mathbf{P}}$ met mogelijke overdrachten. Een gepostuleerde modelverzameling $\mathbf{P}$ beperkt de keuze van het nominale model $P \in \mathbf{P}$. Een nominaal model kan als optimaal gekwalificeerd worden indien het de afstand tot de verst gelegen niet gefalsificeerde overdracht in $\tilde{\mathbf{P}}$ minimaliseert. Als 'afstand' kan hierbij de oneindig norm van de verschillen van de coprieme factoren worden gehanteerd.
   (Dit Proefschrift, hoofdtuk 5)

4. Een systeem is Lyapunov-stabiel in een equilibriumtoestand als voor iedere begintoestand, dicht genoeg bij de equilibriumtoestand, de toestand dicht bij deze equilibriumtoestand zal blijven voor alle tijd. Omdat evaluatie over een oneindige tijd principieel niet mogelijk is, kan volgens de definitie voor geen enkel praktijk systeem stabiliteit bewezen worden.
   (M.Gopal: 'Modern Control System Theory', Wiley Eastern Limited, 1989)

5. Voor een lage orde SISO-proces kunnen met een goed ingestelde PID-regelaar vrijwel dezelfde prestaties worden behaald als met een optimale H_∞-regelaar. Een PID-regelaar kan echter sneller ontworpen worden en kan tijdens het gebruik eenvoudiger aangepast worden. Voor lage orde SISO-processen verdient een PID-regelaar daarom ook de voorkeur boven een H_∞-regelaar.
   (Ton van den Boom, Martin Klompstra en Ad Damen: 'A comparison of PDD, LQG, H_∞ and H_2-controllers for a laboratory process', 9th IASTED International Congres on Modelling, Identification and Control', Innsbruck, Austria, 1990)

6. In de literatuur over het ontwerpen van robuuste regelaars wordt meer aandacht besteed aan het oplossen van het H_∞-standaard probleem dan aan het converteren van een praktijkprobleem naar dit H_∞-standaard probleem. Dit impliceert echter niet dat het converteren naar een H_∞-standaard probleem eenvoudiger is dan het oplossen ervan.

7. De vrije-markteconomie vindt vanaf het begin van deze eeuw haar begrenzing in het van overheidswege gevoerde sociaal-economisch beleid. Een verdere evolutie zal plaatsvinden naar een sociaal-economisch-ecologisch beleid. De grenzen aan de sociale zekerheid en de economische vooruitgang zullen dan worden aangegeven door milieubelangen.

8. Het universitair onderwijs in Nederland dient er voor te waken niet in dezelfde fout te vervallen als de communistische planningseconomie, waarin het behalen van kwantiteit voorop stond en kwaliteit nog nauwelijks telde.
Overigens dient ook voor het omgekeerde te worden gewaakt: kwaliteit gedijt niet zonder kwantiteit.

9. Een goede belichter van toneelvoorstellingen schept sferen en leidt de toeschouwer rond op het toneel zonder dat deze zich daar van bewust is. Dit heeft tot gevolg dat een goede belichter zelden een compliment zal krijgen vanuit het publiek.

10. Genealogie kan een wezenlijke bijdrage leveren tot het verkrijgen van betere inzichten in emigratie- en immigratiebewegingen in de 16e tot en met de 20e eeuw.
(J.B.van Loon: 'Migratie, naamkunde en genealogie,' De Brabantse Leeuw, jrg. 36, no. 4, 1987)

11. Genealogen hanteren het begrip 'familie' in een bredere zin dan niet-genealogen. Hieruit mag niet worden geconcludeerd dat genealogen meer tijd kwijt zullen zijn aan familiebezoeken.