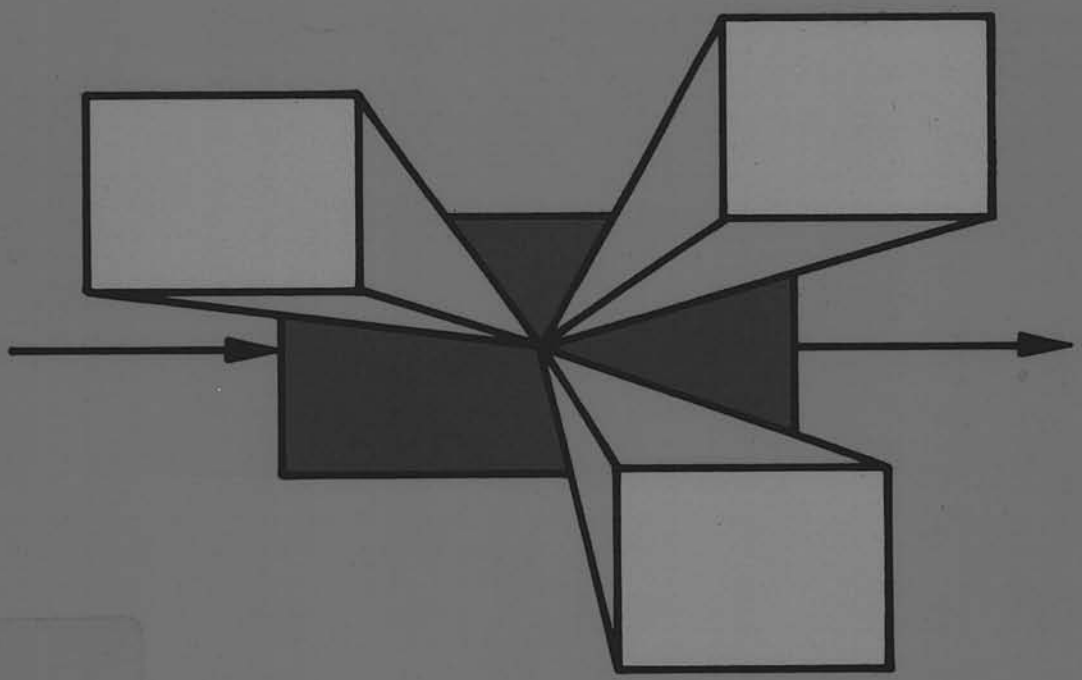
SELECTED TOPICS IN

Identification Modelling and Control

Volume 11, 1998

Edited by O.H. Bosgra, P.M.J. Van den Hof and C.W. Scherer



Delft University Press

716007

IDENTIFICATION, MODELLING AND CONTROL

SELECTED TOPICS IN IDENTIFICATION, MODELLING AND CONTROL

Progress Report on Research Activities in the Mechanical Engineering Systems and Control Group

Edited by O.H. Bosgra, P.M.J. Van den Hof and C.W. Scherer

Volume 11, December 1998

Mechanical Engineering Systems and Control Group
Delft University of Technology

Delft University Press/1998

Published and Distributed by

Delft University Press
Postbus 98
2600 MG Delft
The Netherlands
Tel.: (0)15 - 2783254
Telefax: (0)15 - 2781661

By order of

Mechanical Engineering Systems and Control Group
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
Tel.: +31-15-2785572; Telefax: +31-15-2784717
email: mesc@wbmt.tudelft.nl
www: http://www-sr.wbmt.tudelft.nl/sr

CIP-GEGEVENS KONINKLIJKE BIBLIOTHEEK, DEN HAAG

Selected

Selected topics in identification, modelling and control:
progress report on research activities in the mechanical engineering
systems and control group. - Delft:Mechanical Engineering Systems and Control Group,
Delft University of Technology, Vol. 11-ed. by O.H. Bosgra,
P.M.J. Van den Hof and C.W. Scherer.-ill. Met lit.opg.
ISBN 90-407-1817-2
SISO 656 UDC 531.7 + 681.5 NUGI 841

Cover design by Ruud Schrama

Contents

Volume 11, December 1998

Editorial

This eleventh issue of our selective collection of reports on the scientific progress in our group is in line with the tradition of trying to cover a broad range of topics, touching both theoretical aspects of identification and control and their application to real-world problems.

Let us specifically mention the external authors that appear in this issue of our collection. Zoltán Szabó and József Bokor both from the Hungarian Academy of Sciences contributed to a paper on realization theory for system expansions in terms of generalized orthonormal basis functions that continues work in the identification area. The integration of identification and control for the design of a micro actuator, the outcome of a post-doctoral project of our former Ph.D. student Raymond de Callafon at the University of California at San Diego, is co-authored by D.H.F Harper, F.E. Talke and Robert Skelton (UCSD). A further step towards the extension of validation techniques for model and uncertainty structures as they appear in robust control theory has been taken in a project by our former M.Sc. student Michiel Krüger that has been completed at the University of California at Berkely together with Kameshwar Poolla (UCB). Our continuing activities in process control are represented by a paper on the sequential optimization of large-scale industrial processes that is co-authored by former M.Sc. student Joris van der Schot and Ton Backx (Aspentech). We would particularly like to thank these "guest" authors for their contribution to this issue.

The remaining contributions by our Ph.D. students and staff members center around activities most of which have been previously addressed in this series: robustness tests for parameter dependent systems and their application in the controller design for a CD-player, the multi-level control of our hydraulically driven flight-simulator, the modeling of a flexible wind-turbine, and the input parametrization problem in model predictive control.

Additional information on the activities of our group, as well as reprint versions of the papers in this and previous volumes of our progress report, can be found on our WWW-site:

`www-sr.wbmt.tudelft.nl/sr`

Finally we would like to wish all our colleagues, friends and contacts a happy and properous 1999.

Okko Bosgra o.h.bosgra@wbmt.tudelft.nl
Paul Van den Hof p.m.j.vandenhof@wbmt.tudelft.nl
Carsten Scherer c.w.scherer@wbmt.tudelft.nl
Editors

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Validation of uncertainty models in the presence of stochastic disturbance signals

Michiel V.P. Krüger[§‡] and Kameshwar Poolla[♯]

[§] *Mechanical Engineering Systems and Control Group*
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.
E-mail: m.kruger@iname.com
[♯] *Department of Mechanical Engineering, University of California, Berkeley, CA, USA.*

Abstract. In this paper a new procedure is developed for the validation of uncertainty models. Previous work has been in a deterministic, a probabilistic and even a probabilistic - worst-case framework. However, the assumptions made in these approaches limit the applicability of the developed methods. Therefore a new, more general validation method for uncertainty models, also in a mixed probabilistic - worst-case framework, is proposed. This method leads to more realistic uncertainty bounds. Numerical results of this method are encouraging.

Keywords. Model validation, uncertainty model, stochastic disturbance signal, mixed probabilistic - worst-case approach.

1 Introduction

During the last decade higher performance demands for closed-loop systems resulted in new control design techniques such as H_∞ and robust control (Balas *et al.*, 1991; Packard and Doyle, 1993; Zhou *et al.*, 1996). H_∞-theory aims at minimization of the maximum energy gain between input and output signals. Robust control theory extends this concept to uncertainty models which consist of a nominal model which approximates the relevant dynamics of the real system, and a description of the signal and model uncertainty. Robust control theory guarantees stability and performance for all systems within the uncertainty description. Successful implementation of the designed robust controller on the real system then depends on the ability of the uncertainty model to capture the dynamics of the real system. An accurate uncertainty model, suited for robust control design, is therefore neccessary.
Knowledge of the system can be fruitfully used to model a system. However, in many situations accurate knowledge of the system is not available or the model becomes too complex to be used for control design. Therefore, identification techniques have been developed which estimate a (low-order) nominal model based on I/O-data (Ljung, 1987).
Once a model has been identified, one question remains: is the model good enough for the purpose it was designed for? Validation methods, which confront the estimated model with (new) measurements from the real system can answer this question. These methods test whether the postulated model, the prior information and the assumptions are consistent with the (new) observed data. A model and the priors are not invalidated if there is nothing in the data which is in conflict with the model and the priors. This, however, does not mean that the model and the priors are validated as future measurements may invalidate the model and the priors. Strictly speaking it is therefore only possible to invalidate a model based on observed data.
As the nominal model is always an approximation of the real system, a residual signal is needed to account for the observed data. In the classical identification framework (Ljung, 1987) it is assumed that

[‡]M.V.P. Krüger was with the Mechanical Engineering Systems and Control Group, Delft University of Technology, and is now with the Department of Mechanical Engineering, University of California, Berkeley, CA, USA.

the residual can be modeled as filtered white-noise which is uncorrelated with the input signal. Validation methods in the classical framework, therefore, test whether the noise signal, needed to account for the observed data, has the properties of an uncorrelated white noise signal. However, these validation methods can only be applied to one model at a time instead of model sets and they are therefore not suited for uncertainty models which descibe a family of models around a nominal model. This problem resulted in validation methods for uncertainty models which can deal with model sets (Poolla *et al.*, 1994; Smith *et al.*, 1997). These methods, which assume the disturbance signal and uncertainty to be *deterministic* and *hard-bounded*, compute the minimum uncertainty and disturbance signal needed to account for the observed data. If the size of the disturbance signal and uncertainty are smaller than a pre-specified value, the model is not invalidated. However, these methods have some significant disadvantages as mentioned by Hjalmarsson (1993) and Goodwin *et al.*(1992). First of all, the disturbance signal can be worst-case, i.e. highly correlated with the input signal, without stating the probability that this will occur. Secondly, a hard bound on the disturbance signal implies that no outliers occur. However, in order to account for occasional but unlikely large values of the disturbance signal, the error bound must be unnecessary large. The resulting uncertainty model is therefore often very conservative.

In order to decrease the conservatism in the model it was proposed in De Vries (1994) and De Vries and Van den Hof (1995) to use a hard-bounded uncertainty description to account for undermodeling and a stochastic disturbance signal, which is not worst-case at all time instants, to account for noise disturbances. This approach results in a mixed probabilistic - worst-case model for the real system. However, no general validation method, without limiting assumptions such as periodicity of the input signal (De Vries, 1994) or LTI uncertainties (Hakvoort, 1994), has yet been developed to deal with these mixed models. In this paper we will present a method to solve this problem.

The remainder of this paper is organized as follows. In section 2 the problem is defined in mathematical terms. Section 3 will then discuss an optimal model inversion method based on the Kalman Filter which will be used to solve the validation problem. The main results will then be discussed in Section 4. Section 5 will discuss a numerical example. Finally, the conclusions will be presented in Section 6.

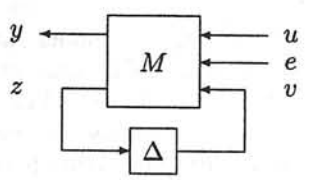


Fig. 1: Model of the data-generating system

2 Problem formulation

Suppose that an uncertainty model is postulated for the real system and suppose that the real system is affected by a stochastic disturbance signal which can be modeled as a filtered unit-variance white noise signal (this is a realistic assumption as every measurement is corrupted by noise). By incorporating the noise model in the uncertainty model, we can assume, without loss of generality, that the disturbance signal is a unit-variance white noise signal. In the sequel of this paper we will denote this stochastic disturbance signal with e.

The uncertainty model then consists of a LTI nominal model M including weighting functions for the uncertainty and disturbance signal, an uncertainty block Δ, I/O-data u and y, and an unknown unit-variance white noise signal e, see Fig. 1.

The uncertainty Δ is assumed to be norm-bounded with a block-diagonal structure,

$$\Delta \in \mathbf{\Delta}(\gamma) := \Big\{\Delta = \mathrm{diag}(\Delta_1, \cdots, \Delta_N) : \Delta_i \text{ is } n_i \times n_i, \text{ and } \|\Delta\| \leq \gamma\Big\}. \quad (1)$$

Note that no assumptions on the class of the uncertainty are made *a priori*, i.e. the uncertainty is not assumed to be, for example, LTI. The unit ball of $\mathbf{\Delta}$ is denoted by

$$\mathbf{B\Delta} = \{\Delta \in \mathbf{\Delta}(1)\}. \quad (2)$$

The validation problem for uncertainty models in the presence of noise can then be stated as follows:

Problem 2.1
Given a perturbation model M, and an I/O-datum (u,y) with length N, see Fig. 1, does there exist a disturbance signal e and an uncertainty Δ, where e is a unit-variance white noise signal and $\Delta \in \mathbf{B\Delta}$ (see (2)), such that,

$$y = F_l(M, \Delta)\begin{bmatrix} u \\ e \end{bmatrix}$$

where

$$F_l(M,\Delta) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} + \begin{bmatrix} M_{13} \\ M_{23} \end{bmatrix} \Delta (I - M_{33}\Delta)^{-1} \begin{bmatrix} M_{31} & M_{32} \end{bmatrix} \quad (3)$$

is the lower LFT of M and Δ.

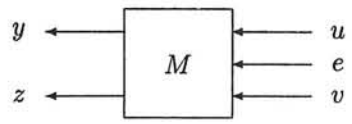


Fig. 2: The postulated nominal model.

The model is not invalidated if a unit-variance white noise signal e and an uncertainty $\|\Delta\| < 1$ can be found such that the model can account for the observed data. This problem will be solved in two steps which will eleborated in next sections. The basic idea is as follows:

1. Omit the uncertainty Δ which connects z and v, see Fig. 2, and estimate signals $\hat{e}$ and $\hat{v}$ such that
   (a) $y = M \begin{bmatrix} u \\ \hat{e} \\ \hat{v} \end{bmatrix}$, i.e. the $(\hat{e}, \hat{v})$-pair is feasible
   (b) $\|\hat{e}\|^2 + \|\hat{v}\|^2$ is minimized
   (c) $\hat{e}$ is a unit-variance white-noise signal
2. Compute $\hat{z}$ and determine the minimum value of $\|\Delta\|$ such that $\hat{v} = \Delta\hat{z}$.

If $\|\Delta\| > 1$ the observed data can not be generated by the proposed model without violating the assumptions and the model is then invalidated. The question how to change the model if it is invalidated will not be addressed in this paper.
In Section 3 we will show how the Kalman Filter (KF) can be used to estimate a feasible $(\hat{e}, \hat{v})$-pair satisfying 1(a) ans 1(b). However, in general the estimated noise signal $\hat{e}$ will not be unit-variance and white (1(c)). Therefore a different feasible $(\hat{e}, \hat{v})$-pair must be estimated. This can be done by using a weighting in the KF estimation process. An iterative procedure will then result in which the weighting is updated for each new KF estimation step according to the difference between the obtained and desired properties of $\hat{e}$ until $\hat{e}$ becomes unit-variance and white. In section 4 we will extensively discuss this iterative procedure (step 1 of the validation procedure). In this section we will also briefly discuss step 2. Standard validation techniques, available in literature, will be used in this step. A schematic picture of the validation process is given in Fig. 3.

3 Optimal model inversion

Consider the following problem (see also Fig. 2):

Problem 3.1 *Given an uncertainty model M and an I/O-datum (u,y) with length N, estimate a disturbance signal $\hat{e}$ and an uncertainty signal $\hat{v}$ such*

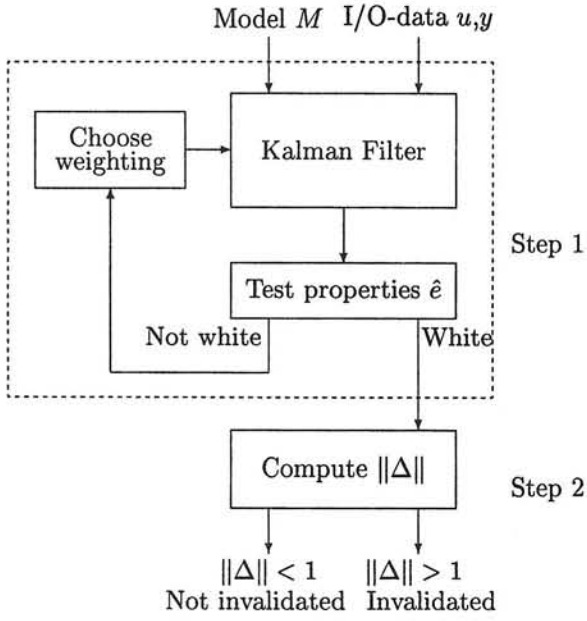


Fig. 3: Scheme of the validation procedure.

that the $(\hat{e}, \hat{v})$-pair is feasible and $\|\hat{e}\|^2 + \|\hat{v}\|^2$ is minimized.

Minimality with respect to $\|\cdot\|_2$ allows occasional outliers in the data without introducing too much conservatism. Note that we do not yet impose any constraints on the estimated noise signal $\hat{e}$.
Suppose that the model M is LTI with the following state-space description:

$$x_{k+1} = Ax_k + B_u u_k + B_w w_k \quad (4)$$
$$y_k = Cx_k + D_u u_k + D_w w_k \quad (5)$$

with $w_k = \begin{bmatrix} e_k \\ v_k \end{bmatrix}$ the output of a stationary white noise process. The Kalman Filter (KF) will now be used to estimate this unknown signal. The KF acts like an observer:

$$\hat{x}_{k+1} = (A - KC)\hat{x}_k + (B_u - KD_u)u_k + Ky_k \quad (6)$$

where $\hat{x}$ is the estimated state and K is the optimal steady-state Kalman Gain. The estimated output is then defined by

$$\hat{y}_k = C\hat{x}_k + D_u u_k. \quad (7)$$

Now define $\tilde{y}$ as the error between $\hat{y}$ and y, i.e.,

$$\begin{aligned} \tilde{y}_k &= \hat{y}_k - y_k \\ &= C\hat{x}_k + D_u u_k - y_k \end{aligned} \quad (8)$$

The signal $\tilde{y}$ is then the output of a system with input signals u and y and internal state $\hat{x}$. Define $\gamma_k = \mathbf{E}[\tilde{y}_k \tilde{y}_k^*]$, where $\mathbf{E}$ denotes mathematical expectation.

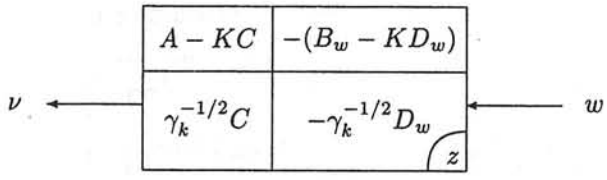


Fig. 4: This state-space system maps w into ν.

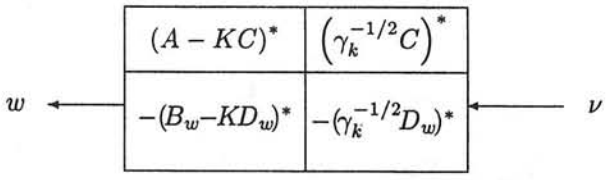


Fig. 5: This adjoint filter computes w from ν.

Fact 3.2
Suppose that the disturbance signal w is a unit-variance, white and zero mean signal. The signal $\tilde{y}$ is then also white zero mean. So, $\gamma_k^{-1/2}\tilde{y}_k$ is then unit-variance, white, zero mean.

Upon use of (4), (5) and (6) the state error $z = \hat{x} - x$ can be determined,

$$z_{k+1} = (A - KC)z_k - (B_w - KD_w)w_k. \quad (9)$$

Now define ν as the weighted prediction error,

$$\nu_k = \gamma_k^{-1/2}\tilde{y}_k = \gamma_k^{-1/2}Cz_k - \gamma_k^{-1/2}D_w w_k. \quad (10)$$

The signal ν is thus the output of a system with w as input and z as internal state, see Fig. 4. Now define $G(q)$ as the transfer function between the disturbance w and the signal ν, i.e.,

$$\nu(t) = G(q)w(t) = G(q)\begin{bmatrix} e(t) \\ v(t) \end{bmatrix} \quad (11)$$

with $G(q) = \begin{bmatrix} \sum_{k=0}^{\infty} g_e(k)q^{-k} & \sum_{k=0}^{\infty} g_v(k)q^{-k} \end{bmatrix}$.
The relation between the disturbance signal w and the signal ν can then be written as,

$$\begin{bmatrix} \nu(0) \\ \nu(1) \\ \vdots \\ \nu(N-1) \end{bmatrix} = T_{\nu w} \begin{bmatrix} w(0) \\ w(1) \\ \vdots \\ w(N-1) \end{bmatrix} \quad (12)$$

where $w(\cdot) = \begin{bmatrix} e(\cdot) \\ v(\cdot) \end{bmatrix}$, N is the length of the data record and $T_{\nu w} \in \mathbf{R}^{N \times 2N}$ is a lower triangular Toeplitz matrix.

Fact 3.3
As assumed in Fact 3.2, the disturbance signal w is unit-variance white, and therefore $\mathbf{E}[ww^*] = I$. The signal ν (equation (10)) is by definition white, zero mean with $\mathbf{E}[\nu\nu^*] = T_{\nu w}\mathbf{E}[ww^*]T_{\nu w}^* = I$. Therefore

$$T_{\nu w}T_{\nu w}^* = I. \quad (13)$$

The signals e and v are thus related by (12) to the signal ν which can be computed. This linear system, however, is *underdetermined* and it has therefore an infinite number of solutions. In order to (in)validate the system we are interested in the minimal solution to the linear system of equation (12). We therefore have to solve the following problem:

$$\text{minimize} \left\| \begin{matrix} e \\ v \end{matrix} \right\| \text{ such that } \nu = T_{\nu w}\hat{w} = T_{\nu w}\begin{bmatrix} \hat{e} \\ \hat{v} \end{bmatrix}.$$

Upon use of the pseudo-inverse of $T_{\nu w}$ and (13) we get $\hat{w} = T_{\nu w}^*(T_{\nu w}T_{\nu w}^*)^{-1}\nu = T_{\nu w}^*\nu$. Using the adjoint filter in Fig. 5 we can compute w from the signal ν.

Remark 3.4 *In the validation problem w consists of a stochastic signal e and an undermodeling signal v, which is in general not stochastic. A nonlinear filter could, therefore, be better, but the derivation of such a filter for general noise statistics is considerably more difficult. Considerations of computation and implementation thus compel us to seek a linear state estimator. We assume that violation of the assumption that w is stochastic hardly affects the optimality of the estimated signal w.*

4 Iterative validation procedure

In this section our main results will be presented. We will consider again the system in Fig. 2. In Section 3 we explained how a feasible $(\hat{e}, \hat{v})$-pair with minimum variance could be constructed. However, the estimated noise signal $\hat{e}$ will in general not be unit-variance white according to our assumption. First of all, we will explain how the variance can be forced to become 1 with the use of a constant weighting. The developed method can then easily be extended to a frequency dependent weighting in order to guarantee whiteness of $\hat{e}$.
The residual r needed to account for the observed data is the result of the signals e and v,

$$r = y - M_{11}u = M_{12}e + M_{13}v. \quad (14)$$

Therefore, there is an infinite set of consistent (e, v)-pairs. Only one single consistent pair, sufficiently small in size, is needed in the validation procedure. For each $\gamma \in [0, \|M_{12}^{-1}r\|^2]$ with $\gamma = \|e\|^2$ there exists a signal $v(\gamma)$ such that $\|v(\gamma)\|^2$ is minimized and such that the $(e, v(\gamma))$-pair is feasible. A trade-off curve between $\|e\|^2$ and $\|v\|^2$ can then be made,

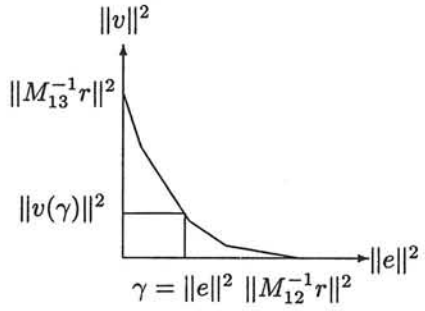


Fig. 6: Tradeoff curve between uncertainty and noise.

see Fig. 6. This figure implies that no (e^*, v^*)-pair can be found which is feasible if the point defined by $(\|e^*\|^2, \|v^*\|^2)$ lies below the curve.
This plot is a direct extension of the uncertainty tradeoff curve as discussed by Kosut (1995).
The Kalman Filtering procedure will compute a feasible $(\hat{e}, \hat{v})$-pair such that the cost function $J = \|\hat{v}\|^2 + \|\hat{e}\|^2$ is minimized. The point $(\|\hat{e}\|^2, \|\hat{v}\|^2)$ will then lie on the tradeoff-curve. However, in general, $\text{cov}(\hat{e}) = \frac{\|\hat{e}\|^2}{N} \neq 1$, where N is the number of datapoints. Suppose $\|\hat{e}_1\|^2 < N$. A new feasible $(\hat{e}_2, \hat{v}_2)$-pair, with $\|\hat{e}_2\|^2 > \|\hat{e}_1\|^2$ can be estimated if the effect of $\|\hat{e}_2\|^2$ on J is decreased, i.e. $J(\alpha) = \|\hat{v}_2\|^2 + \alpha^2\|\hat{e}_2\|^2$, with $\alpha < 1$.
Suppose that a weighting α has been defined (how this can be done will be explained later). A new system $M(\alpha)$ which consists of the nominal model M and the inverse of the weighting α can then be created, see Fig. 7. The KF estimation procedure can then be used to estimate a feasible $(\hat{e}(\alpha), \hat{v})$-pair for $M(\alpha)$. The final step will be the multiplication of the estimated noise signal $\hat{e}(\alpha)$ with the inverse of the used weighting α to get the estimated noise signal $\hat{e}$ for the original system M, i.e. $\hat{e} = \alpha^{-1}\hat{e}(\alpha)$. If the weighting function α is chosen properly, $\|\hat{e}\|^2$ will get closer to N. Several updates are often required to get $\|\hat{e}^2\| = N$. This implies an iterative procedure to get the desired properties for $\hat{e}$.
Convergence of the iterative procedure is very important in order to guarantee that $\hat{e}$ becomes unit-variance.
In the sequel of this section we will use matrices instead of dynamic systems to describe the relation

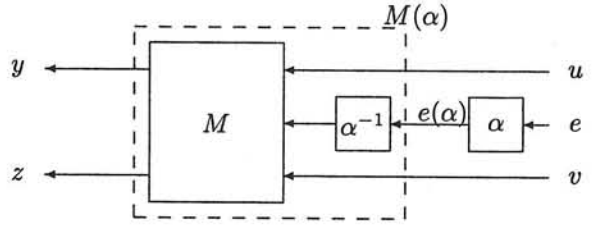


Fig. 7: New system $M(\alpha)$.

between two signals. All the signals will therefore be represented in vectors, i.e., $e \in \mathbf{R}^{N\times 1}$, $v \in \mathbf{R}^{N\times 1}$ where N is the number of data points.
The Kalman Filter, applied to the model $M(\alpha)$ (see Fig. 7), solves the following problem

$$\min_{e,v} \left\| \begin{matrix} \alpha e \\ v \end{matrix} \right\|^2 \text{ such that } r = M^* \begin{bmatrix} e \\ v \end{bmatrix} \tag{15}$$

where $r \in \mathbf{R}^{N\times 1}$ is the residual, i.e. $r = y - M_{11}u$ (see Fig. 2), and $M^* = [\ M_{12}\ \ M_{13}\ ]$ with $M_{11} \in \mathbf{R}^{N\times N}$, $M_{12} \in \mathbf{R}^{N\times N}$ and $M_{13} \in \mathbf{R}^{N\times N}$ all lower triangular Toeplitz matrices.
The vector $\begin{bmatrix} e \\ v \end{bmatrix}$ in (15) can then be written as:

$$\begin{bmatrix} e \\ v \end{bmatrix} = \begin{bmatrix} e_0 \\ v_0 \end{bmatrix} - \begin{bmatrix} B1 \\ B2 \end{bmatrix} f \tag{16}$$

where $\begin{bmatrix} e_0 \\ v_0 \end{bmatrix} \in \mathbf{R}^{2N\times 1}$ is a feasible solution to $r = M^* \begin{bmatrix} e \\ v \end{bmatrix}$ and $\begin{bmatrix} B1 \\ B2 \end{bmatrix} f$ is in the null-space of M^*, with $f \in \mathbf{R}^{N\times 1}$, $B_1 \in \mathbf{R}^{N\times N}$ and $B_2 \in \mathbf{R}^{N\times N}$. Without loss of generality we will assume:

1.
$$[\ B1^*\ \ B2^*\ ] \begin{bmatrix} B1 \\ B2 \end{bmatrix} = I. \tag{17}$$

2.
$$[\ B1^*\ \ B2^*\ ] \begin{bmatrix} e_0 \\ v_0 \end{bmatrix} = 0. \tag{18}$$

The problem in (15) then results in:

$$\min_f \left\| \begin{bmatrix} \alpha e_o \\ v_0 \end{bmatrix} - \begin{bmatrix} \alpha B1 \\ B2 \end{bmatrix} f \right\|^2.$$

The optimal solution f^{opt} to this problem is:

$$\begin{aligned} f^{opt} &= \left[\alpha^2 B_1^* B_1 + B_2^* B_2\right]^{-1} [\ \alpha B_1^*\ \ B_2^*\ ] \begin{bmatrix} \alpha e_o \\ v_0 \end{bmatrix} \\ &= \left[(\alpha^2 - 1)B_1^* B_1 + I\right]^{-1} [\ (\alpha^2 - 1)B_1^* e_0\ ]. \end{aligned} \tag{19}$$

Define the following cost function:

$$\begin{aligned} &J(\alpha) = \text{cov}(e^{opt}) \\ &= \frac{1}{N}\left\| e_0 - B_1 f^{opt} \right\|^2 \\ &= \frac{1}{N}\left\| \left(I - B_1[(\alpha^2-1)B_1^* B_1 + I]^{-1}(\alpha^2-1)B_1^*\right) e_0 \right\|^2 \end{aligned} \tag{20}$$

where N is the number of data points. This expression can then be rewritten with use of the Matrix Inverse Lemma,

$$\begin{aligned} J(\alpha) &= \frac{1}{N}\left\| \left[I + (\alpha^2 - 1)B_1 B_1^*\right]^{-1} e_0 \right\|^2 \\ &= \frac{1}{N} e_0^* \left[I + (\alpha^2 - 1)B_1 B_1^*\right]^{-2} e_0. \end{aligned} \quad (21)$$

The following iterative procedure is used to update the cost weighting α:

$$\begin{aligned} \alpha_0^2 &= 1 \\ \alpha_{k+1}^2 &= \alpha_k^2 \sqrt{J(\alpha_k)}. \end{aligned} \quad (22)$$

Now we want to find $\bar{\alpha}$ such that $J(\bar{\alpha}) = 1$, where $J(\alpha)$ is given by (21).

Theorem 4.1
The iterative procedure given in (22) *will converge to* $J(\bar{\alpha}) = 1$.

Proof:
Introduce $\beta = \alpha^2$ and $K(\beta) = [I + (\beta - 1)W]$, where $W = B_1 B_1^*$ and $I - W > 0$.

$$\begin{aligned} |\alpha_{k+1}^4 - \bar{\alpha}^4| &= |\beta_{k+1}^2 - \bar{\beta}^2| \\ &= \frac{1}{N} |e_0^* K(\beta_k)^{-2} Q K(\bar{\beta})^{-2} e_0| \end{aligned}$$

where

$$Q = (\beta_k^2 - \bar{\beta}^2)\left[(I - W)^2 + \left(\frac{2\beta_k\bar{\beta}}{\beta_k + \bar{\beta}}\right)(W - W^2)\right].$$

This leads then to the following expression,

$$|\beta_{k+1}^2 - \bar{\beta}^2| = \frac{1}{N} |e_0^* X K(\bar{\beta})^{-2} e_0| \, |\beta_k^2 - \bar{\beta}^2| \quad (23)$$

where

$$\begin{aligned} X &= K(\beta_k)^{-2} Y \\ Y &= \left[(I - 2W + W^2) + \left(\frac{2\beta_k\bar{\beta}}{\beta_k + \bar{\beta}}\right)(W - W^2)\right]. \end{aligned}$$

It can be shown that $|X| < I$. Substitution of this strict inequality in (23) leads to

$$|\beta_{k+1}^2 - \bar{\beta}^2| < \frac{1}{N} |e_0^* K(\bar{\beta})^{-2} e_0| \, |\beta_k^2 - \bar{\beta}^2| = |\beta_k^2 - \bar{\beta}^2|.$$

Therefore it can be concluded that the algorithm converges. □

For faster convergence the following relation is proposed to update α in the case that $\alpha \approx \bar{\alpha}$

$$\alpha_{k+1}^2 = \alpha_k^2 J(\alpha_k). \quad (24)$$

Again we want to find $\bar{\alpha}$ such that $J(\bar{\alpha}) = 1$, where $J(\alpha)$ is given by (21).

Theorem 4.2
The iterative procedure given by (24) *will converge to* $J(\bar{\alpha}) = 1$ *if* $|\alpha_k - \bar{\alpha}| < \varepsilon$ *and* ε *is small.*

Proof:
This proof is similar to the proof of Theorem 4.1. □

The first method (22) will converge for any α_0 while the second method (24) will only converge for α_0 close to $\bar{\alpha}$. The rate of convergence, however, may be slower for the first method than for the second method. Therefore, we propose the following procedure:

1. Use (22) in the first iteration steps to bring any initial condition α_0 close to $\bar{\alpha}$.
2. Use (24) in the following iteration steps to estimate $\bar{\alpha}$ in order to increase the rate of convergence.

The iterative procedure with a constant weighting α will only guarantee $\mathrm{cov}(\hat{e}) = 1$; however, the spectral properties of the estimated noise signal $\hat{e}$ may still be not satisfying. Therefore, instead of a constant weighting α, a frequency-dependent weighting $\alpha(\omega)$ should be used in the iterative procedure. This frequency-dependent weighting should be chosen such that differences between the spectral density of the estimated noise signal $\hat{e}$ and the desired spectral density for this signal are eliminated.
The spectral density $\Phi_x(\omega)$ of a signal x is defined by (Ljung, 1987)

$$\Phi_x(\omega) = \sum_{\tau=-\infty}^{\infty} R_x(\tau) e^{-i\omega\tau} \quad (25)$$

where $R_x(\tau)$ is the autocorrelation function of the signal x,

$$R_x(\tau) = \mathbf{E}x(t)x(t-\tau). \quad (26)$$

In the case that the signal x is a unit-variance white noise sequence $\Phi_x(\omega) = 1$.
The frequency dependent weighting $\alpha(\omega)$ should therefore be updated according to the following iterative procedure:

$$\begin{aligned} \alpha_0(\omega) &= 1 \\ \alpha_{k+1}^2(\omega) &= \alpha_k^2(\omega) F(e^{i\omega}) \end{aligned} \quad (27)$$

where

$$|F(e^{i\omega})|^2 = \Phi_{\hat{e}}(\omega). \quad (28)$$

For faster convergence the following iterative procedure can be used:

$$\begin{aligned} \alpha_0(\omega) &= 1 \\ \alpha_{k+1}^2(\omega) &= \alpha_k^2(\omega) |F(e^{i\omega})|^2. \end{aligned} \quad (29)$$

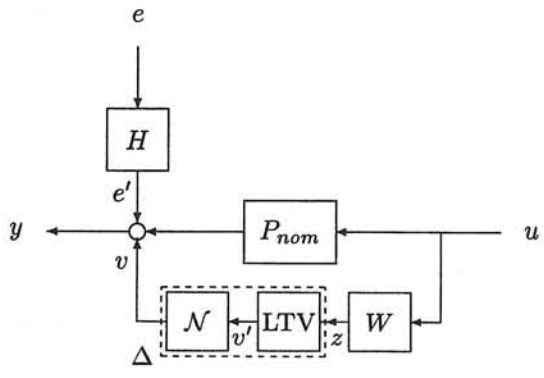


Fig. 8: Data-generating system

Although convergence has not (yet) been proved for this frequency dependent iterative procedure, it has been successfully used in several experiments.
In this new approach with a frequency dependent weighting the spectrum of $\hat{e}$ is forced to become flat. As a result, chances are much higher that $\hat{e}$ is uncorrelated with $\hat{v}$ and u than in the deterministic approach.
Once a feasible $(\hat{e}, \hat{v})$-pair with $\hat{e}$ unit-variance and white is available (step 1 of the validation procedure) we can continue with step 2 which involves the computation of the input to the uncertainty Δ (M is completely known), and $\|\Delta\|$, i.e. the size of the uncertainty which connects z and v. Knowledge and insight in the system can be fruitfully used to choose the class of uncertainty (LTI, LTV, static nonlinearity, etc.). Several validation techniques (Poolla *et al.*, 1994; Smith *et al.*, 1997), depending on the class of uncertainty which is assumed, can then be used to compute $\|\Delta\|$.

5 Numerical example

A simple numerical example will be used to show the ability of the iterative procedure to recover the used e and v signals. The model in this example consists of a second order low-pass filter P_{nom} with a lightly damped resonance peak at $z = 0.9893 \pm 0.0396i$,

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 0.9797 & -0.0407 \\ 0.0407 & 0.9988 \end{bmatrix} x_k + \begin{bmatrix} -0.4469 \\ 0.0445 \end{bmatrix} u_k \\ y_k &= \begin{bmatrix} -0.4469 & -0.0445 \end{bmatrix} x_k + 0.0995 u_k \end{aligned}$$

disturbed by a filtered white noise sequence and a weighted additive uncertainty, which consists of an LTV system and a static nonlinearity $\mathcal{N}$, see Fig. 8. The weighting for the uncertainty is given by

$$W(z) = \frac{0.4786 - 0.4738 z^{-1}}{1 - 0.9048 z^{-1}}$$

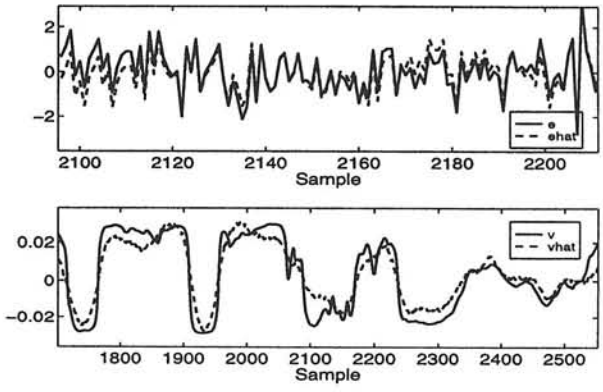


Fig. 9: Estimated signals $\hat{e}$ and $\hat{v}$ and real signals e and v.

and the white noise disturbance signal e is filtered by

$$H(z) = \frac{0.0153 - 0.0141 z^{-1}}{1 - 0.9608 z^{-1}}.$$

The static nonlinearity is defined as

$$\mathcal{N} : v_k = \frac{\text{atan}(100 v'_k)}{50}. \qquad (30)$$

The LTV system consisted of a slowly varying gain between -1 and 1. A low frequency input signal u and a unit variance white noise signal e were used to generate data for the validation procedure; 4096 uniformly spaced samples were used. The signals v and $e' = H(z)e$ were chosen such that $\|e'\|^2 \approx \|v\|^2$. Aim of the first step in the validation procedure is to estimate the e and v. The computed size of the uncertainty in the second step will be more realistic if the real signals e and v are well approximated.
The initial weighting was constant for all frequencies, i.e. $\alpha_0(\omega) = 1$. For each estimated noise signal $\hat{e}$ the spectral density was computed at 128 uniformly spaced frequency points between 0 and π. A fourth order, stable, minimum-phase (to avoid unstable behavior of $M(\alpha)$) system $F(e^{i\omega})$ was then estimated such that $|F(e^{i\omega})|^2 \approx \Phi_{\hat{e}}(\omega)$. In the first 5 iteration steps $\alpha(\omega)$ was updated according to (27). In the next 10 iteration steps $\alpha(\omega)$ was updated according to (29). In this way, the spectrum of $\hat{e}$ was forced to become (almost) flat. Fig. 9 shows the estimated noise and uncertainty signals, resp. $\hat{e}$ and $\hat{v}$ together with the real signals e and v. Both $\hat{e}$ and $\hat{v}$ are good approximations of the real signals e and v. Besides, the cross-correlation between $\hat{e}$ and u is within the 99% confidence bounds, see Fig. 10.
Hence, we can conclude that the iterative procedure has successfully approximated the real signals e and v such that the $(\hat{e}, \hat{v})$-pair is feasible without violating the assumptions on the estimated noise signal $\hat{e}$.

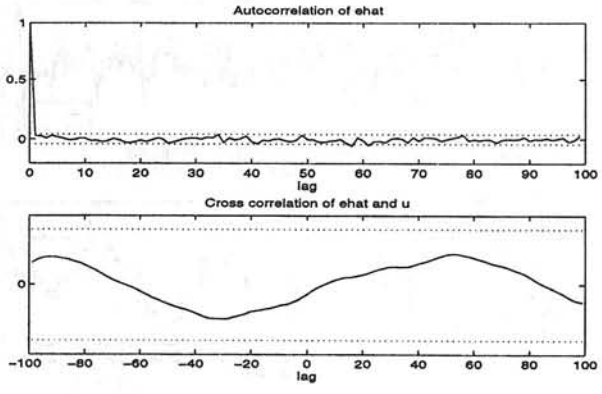


Fig. 10: Auto- and cross-correlation of resp. $\hat{e}$ and $\hat{e}$,u together with confidence bounds.

6 Conclusions

In this paper we have made an attempt to solve the uncertainty model validation problem in a mixed probabilistic - worst-case framework without limiting assumptions on the uncertainty or the input signal. For this purpose a two-step method was proposed. The residual signal, needed to reproduce the observed data, is separated in a stochastic signal e, which accounts for disturbances on the system, and a deterministic signal v which accounts for undermodeling. In the first step a feasible $(\hat{e}, \hat{v})$-pair is computed such that $\hat{e}$ is unit-variance and white. An iterative procedure has been developed to achieve this. Convergence of this iterative procedure is only proved in the case that a constant weighting is used; examples, however, show that this iterative procedure converges also if a frequency dependent weighting, based on the spectrum of $\hat{e}$, is used. In some cases uncorrelatedness between $\hat{e}$ and u cannot be achieved with this frequency dependent weighting function. Therefore it is still an open problem how to *guarantee* that $\hat{e}$ and u are uncorrelated.
In the second step standard model validation techniques can be used to estimate $\|\Delta\|$. The model is invalidated if the size of the uncertainty is bigger than a pre-specified value.
Solutions to the deterministic validation problem are based on convex optimization problems constructed directly from I/O-data. As a result these methods are limited to 500 to 1000 data samples with the current state of the art in optimization and computed technology (Smith *et al.*, 1997). The validation method for uncertainty models, developed in this paper, is computationally more attractive and is not limited to short data sequences.

Acknowledgements

The authors would like to thank Bill Helton at the University of California, San Diego, and Paul Van den Hof at Delft University of Technology for fruitful discussions and contributions to this paper.

References

Balas, G.J., J.C. Doyle, K. Glover, A. Packard and R. Smith (1991). *The μ-Analysis and Synthesis Toolbox.* The MathWorks Inc.

De Vries, D.K. (1994). *Identification of Model Uncertainty for Control Design.* Ph.D. thesis, Delft University of Technology, Delft, the Netherlands.

De Vries, D.K. and P.M.J. Van den Hof (1995). Quantification of uncertainty in transfer function estimation: a mixed probabilistic - worst-case approach. *Automatica,*31, 543–557.

Goodwin, G.C., M. Gevers and B. Ninness (1992). Quantifying the error in estimated transfer functions with application to model order selection. *IEEE Trans. Autom. Control,* AC–37, 913–928.

Hakvoort, R.G. (1994). *System Identification for Robust Process Control: Nominal Models and Error Bounds.* Ph.D. thesis, Mechanical Engineering Systems and Control Group, Delft University of Technology, Delft, the Netherlands.

Hjalmarsson, H. (1993). *Aspects on Incomplete Modeling in System Identification.* Ph.D. thesis, Electrical Engineering, Linköping University, Linköping, Sweden.

Kosut, R.L. (1995). Uncertainty model unfalsification: a system identification paradigm compatible with robust control design. *Proc. 34th IEEE Conf. Decision and Control,* New Orleans, pp. 3492–3497.

Ljung, L. (1987). *System Identification - Theory for the User.* Prentice-Hall, Englewood Cliffs, NJ.

Packard, A. and J.C. Doyle (1993). The complex structured singular value. *Automatica,* 29, 71–109.

Poolla, K., P. Khargonekar, A. Tikku, J. Krause and K. Nagpal (1994). A time-domain approach to model validation. *IEEE Trans. Autom. Control,* AC–39, 951–959.

Smith, R., G. Dullerud, S. Rangan and K. Poolla (1997). Model validation for dynamically uncertain systems. *Mathematical Modelling of Systems,* 3, 43–58.

Zhou, K., J.C. Doyle and K. Glover (1996). *Robust and Optimal Control.* Prentice Hall, Englewood Cliffs, NJ.

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Gain scheduling control for a CD player mechanism using LPV techniques[‡]

M. Dettori[§] and C. W. Scherer

Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.
E-mail: m.dettori@wbmt.tudelft.nl

Abstract. This paper investigates the application of a recently developed gain scheduling technique to the control of the radial positioning servosystem in a compact disc (CD) player mechanism. In this application the gain of the plant and the disturbances affecting it depend on the operating point of the system. We model this situation with a linear parametrically-varying (LPV) generalized plant where the performance specifications vary according to the operating conditions. Our purpose is to show in how far stability and performance can be guaranteed through the design of an LPV controller.

Keywords. Gain scheduling, Linear Parametic Varying (LPV) control, Linear Matrix Inequalities (LMIs), CD player.

1 Introduction

Gain scheduling is a technique that allows to synthesize controllers for nonlinear systems using linear design tools. In many applications, it is known how the behavior of a system varies with the operating points and it is possible to parametrize these variations in terms of some variables that can be measured on-line. In the classical approach to gain scheduling, one linearizes the system around a certain number of operating points and designes a linear controller at each point. In a subsequent step, all these local linear controllers are "glued" together by means of some interpolation algorithms to get a global one, valid over the whole operative range. As a main drawback of this procedure there are no general recipes to perform the interpolation step in such a way that stability and performance can be guaranteed. Furthermore it is not clear what properties of the linear controllers enforce a desired specification for the global one. A more recent approach to gain scheduling is based on LPV techniques (Apkarian and Gahinet, 1995; Helmersson, 1995; Packard, 1994; Scorletti and El Ghaoui, 1995; Wu, 1995). In this framework it is possible to systematically design a family of linear controllers scheduled on the operating point without the need for interpolation procedures. It has been shown that the synthesis of an LPV controller results in a convex optimization problem that can be solved using Linear Matrix Inequalities (LMIs). In this paper we apply the full block scalings LPV techniques presented in Scherer (1997) and Nijo *et al.* (1996) to the control of a CD player mechanism. The model of this system exhibits a gain that depends nonlinearly on the position of the track that is being read on the disc. Furthermore the disturbances acting on the system are described as periodic signals with harmonic components at the (time-varying) rotational frequency and its multiples. The need for a scheduling is therefore twofold: to account for gain variations and to get adaptive disturbance suppression.

[‡]This paper was presented at the *6th IEEE Mediterranean Conference on Control and Systems*, June 9-11, 1998, Alghero, Italy.

[§]The research of Marco Dettori is sponsored by Philips Research Laboratories, Eindhoven, The Netherlands

2 Theory overview

In this section we recapitulate briefly the theoretical background. For a more complete discussion we refer to Scherer (1997).

2.1 From nonlinear to LPV systems

There are no general recipes to transform a nonlinear system into an LPV. In this subsection we want to sketch one of the possibilities, without being too rigorous from a mathematical viewpoint. Consider a nonlinear system of the form

$$\begin{aligned} \dot{x} &= a(x,q_1) + b_1(x,q_1)w + b_2(x,q_1)u \\ z &= c_1(x,q_1) + d_1(x,q_1)w + d_2(x,q_1)u \\ y &= c(x,q_1) + d(x,q_1)w \end{aligned} \quad (1)$$

where x is the state, w an exogenous input (disturbance to be rejected and/or reference to be tracked), u the control input, z the performance output, y the measured output and $q_1 \in \Pi_1$ is an unknown parameter that affects the system description. If $x = 0$ is an equilibrium of the system, that is $a(0,q_1) = 0$, $c_1(0,q_1) = 0$ and $c(0,q_1) = 0$ for all q_1, it is possible to rewrite (1) as:

$$\begin{aligned} \dot{x} &= A(x,q_1)x + B_1(x,q_1)w + B_2(x,q_1)u \\ z &= C_1(x,q_1)x + D_1(x,q_1)w + D_2(x,q_1)u \\ y &= C(x,q_1)x + D(x,q_1)w \end{aligned}$$

where $A(.)$, $B_1(.)$, $B_2(.)$, $C_1(.)$, $D_1(.)$, $D_2(.)$, $C(.)$ and $D(.)$ are smooth matrix valued mappings. To transform the nonlinear system into an LPV system, we replace the occurrence of the state $x(t)$ in the system matrices with a time-varying parameter $q_2(t)$. With $p = (q_1, q_2)'$ we obtain

$$\begin{aligned} \dot{x} &= A(p(t))x + B_1(p(t))w + B_2(p(t))u \\ z &= C_1(p(t))x + D_1(p(t))w + D_2(p(t))u \\ y &= C(p(t))x + D(p(t))w. \end{aligned} \quad (2)$$

It is obvious that this replacement introduces conservatism: disregarding the coupling between x and p, (2) admits a larger set of trajectories than (1). On the other hand, we get two advantages. First, we can make use of the existing systematic design techniques for (2). Second, a controller that guarantees stability and performances for (2) does it as well for (1). In some cases it is known that the state trajectories of (1) satisfy a bound like $x(t) \in \Pi_2$ for all $t \geq 0$. Then the design specifications for (2) should be achieved only for $p(t) \in \Pi = \Pi_1 \times \Pi_2$. The starting point in LPV control design is the characterization of performance for (2) in terms of a Lyapunov function. In this paper we choose the L_2 induced gain from w to z as performance measure. It is known that (2) is exponentially stable and has L_2 gain smaller than γ if there exists a Lyapunov matrix $X > 0$ that satisfies the strict bounded real lemma

$$\begin{bmatrix} A(p)'X + XA(p) & XB_1(p) & C_1(p)' \\ B_1(p)'X & -\gamma I & D_1(p)' \\ C_1(p) & D_1(p) & -\gamma I \end{bmatrix} < 0, \ \forall p \in \Pi.$$

In order to get rid of the parameter dependence, as an alternative to gridding techniques, a set of extra variables is introduced that are called scalings. These variables code the information about the parameters through quadratic inequalities. The passage from the analysis inequalities in terms of X and the scalings to the synthesis inequalities is then performed as described in Scherer *et al.* (1997).

2.2 LPV synthesis: problem formulation

The dependence of the system (2) on the parameter $p \in \Pi$ is supposed to be such that we can represent it as linear fractional transformation (LFT):

$$\begin{bmatrix} \dot{x} \\ z_1 \\ z_2 \\ y \end{bmatrix} = \begin{bmatrix} A & G_1 & G_2 & B \\ H_1 & F_1 & F_{12} & E_1 \\ H_2 & F_{21} & F_2 & E_2 \\ C & D_1 & D_2 & 0 \end{bmatrix} \begin{bmatrix} x \\ w_1 \\ w_2 \\ u \end{bmatrix}, \ w_2 = \Delta(p) z_2 \quad (3)$$

where $w_1 \rightarrow z_1$ is the performance channel and $\Delta : \Pi \rightarrow \mathbf{R}^{k \times l}$ is a continuous, possibly nonlinear function. The value set of $\Delta(.)$ is assumed to be contained in a convex set with finitely many extreme points: $\Delta(\Pi) \subset \text{conv}\{\Delta_1, ..., \Delta_q\}$. It is well known that in case of rational dependence on p of the system matrices in (2), one can obtain the LFT representation (3) with the linear function

$$\Delta(p) = \text{diag}(p_1 I, ..., p_m I) \quad (4)$$

and shift and rescale the parameter set to

$$\Pi = \{(p_1, ..., p_m) : -1 \leq p_j \leq 1 \ \forall j = 1, ..., m\}.$$

In robust control design, the parameter p is considered unknown and the goal is to find an LTI controller that stabilizes and achieves performances for the system (3) for every $p \in \Pi$. In LPV design, the parameter p is still considered a priori unknown, but on-line measurable. This extra information is then used in the control law. In this framework, the structure of the controller is assumed to be the same of the plant (3): an LTI part K

$$\begin{bmatrix} \dot{x}_c \\ u \\ z_c \end{bmatrix} = \begin{bmatrix} A & B_{c1} & B_{c2} \\ C_{c1} & D_{c1} & D_{c12} \\ C_{c2} & D_{c21} & D_{c2} \end{bmatrix} \begin{bmatrix} x_c \\ y \\ w_c \end{bmatrix} \quad (5)$$

scheduled with

$$w_c = \Delta_c(p) z_c \tag{6}$$

where Δ_c is a $k_c \times l_c$ real matrix-valued function. We can recast this problem in the standard robust control set-up, just representing the controlled system as an augmented LTI part

$$\left[\begin{array}{c} \dot{x} \\ \hline z_1 \\ z_2 \\ z_c \\ \hline y \\ w_c \end{array}\right] = \left[\begin{array}{c|c|cc|cc} A & G_1 & G_2 & 0 & B & 0 \\ \hline H_1 & F_1 & F_{12} & 0 & E_1 & 0 \\ H_2 & F_{21} & F_2 & 0 & E_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & I \\ \hline C & D_1 & D_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 \end{array}\right] \left[\begin{array}{c} x \\ \hline w_1 \\ w_2 \\ w_c \\ \hline u \\ z_c \end{array}\right] \tag{7}$$

scheduled as

$$\left[\begin{array}{c} w_2 \\ w_c \end{array}\right] = \left[\begin{array}{cc} \Delta(p) & 0 \\ 0 & \Delta_c(p) \end{array}\right] \left[\begin{array}{c} z_2 \\ z_c \end{array}\right] \tag{8}$$

and interconnected with the LTI controller (5). As the only difference, also $\Delta_c(.)$ is a design variable. The synthesis problem for the chosen performance specification amounts to finding an LTI controller (5) and a scheduling function (6) such that, for all parameter curves $p : [0, \infty) \to \Pi$, the closed loop system is exponentially stable and has minimal L_2 induced gain from w_1 to z_1. Notice that in this problem formulation the rate of change of the parameters is not taken into account. Hence a possible a priori knowledge of the boundedness of this rate cannot be brought into the design what is another potential source of conservatism.

2.3 LPV synthesis: solution

The discussion of the theory behind the LPV synthesis problem is beyond the scope of this paper and we refer again to Scherer (1997). In what follows we will only present algorithmically the sequence of steps that have to be performed to design an LPV controller. As mentioned before, the first step is the introduction of scalings to code the information about the parameters. In Scherer (1997) it is assumed that the set of admissible parameters is described through a set of symmetric matrices

$$P = \left[\begin{array}{cc} Q & S \\ S' & R \end{array}\right] \text{ with } Q < 0 \tag{9}$$

that satisfy the constraint

$$\left[\begin{array}{c} \Delta(p) \\ I \end{array}\right]' P \left[\begin{array}{c} \Delta(p) \\ I \end{array}\right] > 0 \text{ for all } p \in \Pi. \tag{10}$$

Notice that, due to the assumption $\Delta(\Pi) \subset \text{conv}\{\Delta_1, ..., \Delta_q\}$, the constraint (10) is implied by

$$\left[\begin{array}{c} \Delta_j \\ I \end{array}\right]' P \left[\begin{array}{c} \Delta_j \\ I \end{array}\right] > 0 \text{ for all } j = 1, ..., q \tag{11}$$

that allows to parametrize the scalings with a finite number of linear matrix inequalities. In the literature other choices for the set of scalings can be found. In Scorletti and El Ghaoui (1995) the scalings are restricted to have block diagonal structure $Q = \text{diag}(Q_1, ..., Q_m) < 0$, $R = -Q$ and $S = \text{diag}(S_1, ..., S_m)$ with $S + S' = 0$ or even $S = 0$ like in Apkarian and Gahinet (1995). The choice of the full-block scalings (9),(10) comprises the other ones what can reduce the conservatism in the solution of the synthesis problem. On the other hand, this choice increases the number of variables of the problem and can create problems in numerical implementations. Once a set of scalings has been chosen, the algorithm proceeds as follows:

- Partition the scalings as

$$\begin{aligned} Q &= \left[\begin{array}{cc} Q_1 & Q_{12} \\ Q_{12}' & Q_2 \end{array}\right] \\ S &= \left[\begin{array}{cc} S_1 & S_{12} \\ S_{21} & S_2 \end{array}\right] \\ R &= \left[\begin{array}{cc} R_1 & R_{12} \\ R_{12}' & R_2 \end{array}\right] \end{aligned} \tag{12}$$

  according to that in (8).

- Find, if feasible, $X, Y, Q_1, R_1, S_1, \tilde{Q}_1, \tilde{R}_1, \tilde{S}_1$ that minimize γ subject to the LMI constraints:

$$K_n' \left[\begin{array}{ccc|c|c} \text{sy}(XA) & XG_1 & H_1' & XG_2 + H_2'S_1' & H_2'R_1 \\ * & -\gamma I & F_1' & F_{21}'S_1' & F_{21}'R_1 \\ * & * & -\gamma I & F_{12} & 0 \\ \hline * & * & * & Q_1 + \text{sy}(S_1F_2) & F_2'R_1 \\ \hline * & * & * & * & -R_1 \end{array}\right] K_n < 0$$

$$K_m' \left[\begin{array}{ccc|c|c} \text{sy}(AY) & G_1 & YH_1' & YH_2' + G_2\tilde{S}_1 & G_2\tilde{Q}_1 \\ * & -\gamma I & F_1' & F_{21}' & 0 \\ * & * & -\gamma I & F_{12}\tilde{S}_1 & F_{12}\tilde{Q}_1 \\ \hline * & * & * & \tilde{R}_1 + \text{sy}(F_2\tilde{S}_1) & F_2'\tilde{Q}_1 \\ \hline * & * & * & * & -\tilde{Q}_1 \end{array}\right] K_m < 0$$

$$\left[\begin{array}{cc} X & I \\ I & Y \end{array}\right] > 0$$

$$Q_1 < 0, \left[\begin{array}{c} \Delta_j \\ I \end{array}\right]' \left[\begin{array}{cc} Q_1 & S_1 \\ S_1' & R_1 \end{array}\right] \left[\begin{array}{c} \Delta_j \\ I \end{array}\right] > 0$$

$$\tilde{R}_1 < 0, \left[\begin{array}{c} I \\ \Delta_j' \end{array}\right]' \left[\begin{array}{cc} \tilde{Q}_1 & \tilde{S}_1 \\ \tilde{S}_1' & \tilde{R}_1 \end{array}\right] \left[\begin{array}{c} I \\ \Delta_j' \end{array}\right] > 0$$

  for all $j = 1, ..., m$, where K_n and K_m are basis matrices of the kernels of

$$[\, B' \;\; 0 \;\; E_1' \;\; 0 \;\; E_2' \,], \; [\, C \;\; D_1 \;\; 0 \;\; D_2 \;\; 0 \,]$$

and $\mathrm{sy}(A) := A' + A$. This is a standard LMI problem that can be solved with available solvers, like LMILAB (Gahinet *et al.*, 1995).

- Define

$$M := \begin{bmatrix} -Q_1 & S_1 \\ S_1' & -R_1 \end{bmatrix}, \tilde{M} := \begin{bmatrix} \tilde{Q}_1 & \tilde{S}_1 \\ \tilde{S}_1' & \tilde{R}_1 \end{bmatrix}.$$

Suppose that $M - \tilde{M}^{-1}$ is nonsingular (what can always be achieved by a small perturbation since the synthesis inequalities are strict). Find a decomposition $(M-\tilde{M}^{-1})^{-1} = J\Lambda J'$ where J is orthogonal and $\Lambda = \mathrm{diag}(\Lambda_1, -\Lambda_2)$ with $\Lambda_1 > 0$ and $\Lambda_2 > 0$. Denote by k_c and l_c respectively the number of positive and negative eigenvalues of Λ.

- Define

$$Z := \begin{bmatrix} 0_{k\times l} \\ I_l \end{bmatrix}, \quad \tilde{Z} := \begin{bmatrix} I_k \\ 0_{l\times k} \end{bmatrix} \tag{13}$$

and calculates matrices T_1 and T_2 of respectively k_c and l_c columns such that

$$\begin{aligned} T_1'(\Lambda - J'\tilde{Z}'(\tilde{Z}'M\tilde{Z})^{-1}\tilde{Z}'J)T_1 &> 0 \\ T_1'(\Lambda - J'Z(Z'MZ)^{-1}Z'J)T_2 &< 0 \end{aligned}$$

- Set $T := [T_1\ T_2]$, $N := JT$, $L := J'\Lambda J$ and calculate the missing blocks of the full scalings (13) from

$$N = \begin{bmatrix} -Q_{12} & S_{12} \\ S_{21}' & -R_{12} \end{bmatrix}, L = \begin{bmatrix} -Q_2 & S_2 \\ S_2' & -R_2 \end{bmatrix}$$

- Calculate matrices $\tilde{U}$, $\tilde{W}$ and $\tilde{V}$ from

$$\begin{bmatrix} \tilde{U} & \tilde{W} \\ \tilde{W}' & \tilde{V} \end{bmatrix} := \begin{bmatrix} Q - SR^{-1}S' & SR^{-1} \\ R^{-1}S' & -R^{-1} \end{bmatrix}^{-1}$$

and partition them as in (13).

- The required scheduling function Δ_c is then given as

$$\Delta_c(p) := -\tilde{W}_2 + \begin{bmatrix} \tilde{U}_{12}' & \tilde{W}_{21} \end{bmatrix} \begin{bmatrix} \tilde{U}_1 & \tilde{W}_1 + \Delta(p) \\ (\tilde{W}_1 + \Delta(p))' & \tilde{V}_1 \end{bmatrix} \begin{bmatrix} \tilde{W}_{12} \\ \tilde{V}_{12} \end{bmatrix}.$$

- It only remains to design the LTI part (5) of the controller. This can be done as a nominal design for the extended plant (7) with the quadratic performance specification

$$\int_0^\infty \left[\begin{array}{c} w_2 \\ w_c \\ \hline z_2 \\ z_c \end{array}\right]' \left[\begin{array}{cc|cc} -\gamma I & 0 & 0 & 0 \\ 0 & Q & 0 & S \\ \hline 0 & 0 & \frac{1}{\gamma}I & 0 \\ 0 & S' & 0 & R \end{array}\right] \left[\begin{array}{c} w_2 \\ w_c \\ \hline z_2 \\ z_c \end{array}\right] dt \le 0$$

along the lines of Scherer *et al.* (1997).

3 Application to the CD Player

A Compact Disc (CD) player is an optical device that decodes and reproduces binary coded information. The information signal is stored in a spiral shaped track on a reflective disc which rotates by effect of a turn-table DC-motor. The rotational frequency varies according to the position on the disc of the track that is being read, approximatively from 8 Hz (innermost position) to 4 Hz (outermost position). Track following is performed by a radial arm at the end of which an optical element is mounted. A diode in this element generates a laser beam which is focused in a spot on the information layer of the disc. A system of four photodiodes provides position error information, which is the only signal available for control. A controller is needed for accurate radial and vertical positioning of the laser spot. In this paper we will consider only the radial loop. From frequency domain identification experiments and curve fitting techniques we estimated a model of the transfer function $H(s)$ from the current which drives the arm to the position error. Due to the geometry of the arm, the gain of H varies in a nonlinear way according to the position on the disc of the track that is being read. Through identification experiments at several track positions, the gain profile of fig. 1 has been estimated as function of the radial displacement r from the center of the disc (Dötsch, 1998). In this way we get a model of the system that is in the LPV form (2) with r as parameter. The main control specification is to keep the time domain amplitude of the position error signal bounded in the presence of disturbances. To avoid loosing track, the maximum allowable error should be $0.1\mu m$. The major source of disturbances is given by the eccentricity of the track that, by standardization of Compact Discs, cannot exceed $100\mu m$. Hence the controlled system should achieve a time domain attenuation of the disturbances of a factor 1000. The information about the physical origin of the disturbance signal leads us to model its spectrum as a series of pulses centered around the rotational frequency and its higher harmonics, with an amplitude that is decaying at a rate -40 dB/dec. Due to the variation of the rotational frequency during operational conditions, the location of this pulses will vary in time. In the current controller implementation for audio players, disturbance suppression is achieved by pushing down the amplitude of the sensitivity function in the whole frequency band where they can occur. As a rule of thumb, the objective is achieved when the sensitivity has amplitude below -60 dB up to the rotational frequency. The field of use of the CD mechanism, however, is being gradually extended to new appli-

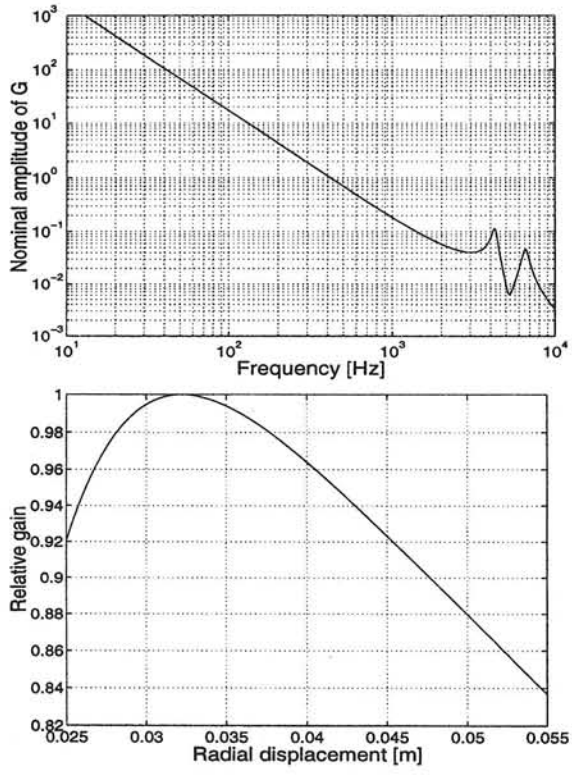


Fig. 1: Nominal amplitude of $H(j\omega)$ together with its normalized gain as function of the radial displacement

cations, like CD-ROM or DVD-ROM, that require a faster data readout and a shorter access time, together with a higher density of the data on the disc. These improvements are achieved by an increase of the rotational frequency of the disc to 30 Hz. In this case, a uniform attenuation of the sensitivity function below -60 dB up to 30 Hz would result in an untolerable increase of the bandwidth. The presence of high-frequent resonant modes (due to flexibility of the mechanical structure) and unmodeled dynamics puts un upper bound on the achievable bandwidth. Therefore we suggest to selectively suppress the disturbances only at the frequencies where they are present, and to schedule this suppression with the on-line-measured rotational frequency.

3.1 LPV model

As it emerges from the preceding discussion, in the CD player there are two features that depend on the operating conditions: the gain and the location of the disturbance spectrum. Both these variations can be parametrized in terms of one parameter, namely the rotational frequency. The variation of the gain in fig. 1 can, in fact, be related to the rotational frequency f_{rot} through the expression $2\pi f_{rot} = v_s/r$, where v_s is the constant linear velocity at the scanning point. The variation of the disturbance location is not a phenomenon that is intrinsic to the plant, but it can be brought into the design via scheduling the weighting functions. This can be seen as an extension of the concept of gain scheduling: it is not only a way to adapt to parametric variations of the plant, but also a tool to impose a different specification for every operating point. As an advantage of the generalized plant set-up, scheduling of the system and of the performance filters can be done exactly in the same fashion, just by "pulling out the deltas" from the combined state space realizations and writing it as an LFT representation like in (3). In our design we chose as performance filter the series connection of two notches

$$W_1(s) = \frac{s^2 + 2\zeta_z\omega_0 s + \omega_0^2}{s^2 + 2\zeta_p\omega_0 s + \omega_0^2} \frac{s^2 + 4\zeta_z\omega_0 s + 4\omega_0^2}{s^2 + 4\zeta_p\omega_0 s + 4\omega_0^2}$$

where $\omega_0 = 2\pi f_{rot}$ is the scheduling parameter. The values ζ_p and ζ_z are chosen in order to obtain a good trade-off between disturbance suppression (related to the notch height) and robustness against inaccurate positioning (related to the notch width). It is interesting to mention that in our experience the numerical solvability of the synthesis LMI is strongly related to the shape of the notches: when we choose for very sharp notches ($\zeta_p = 0.1$ and $\zeta_r = 10$) the solver is not able to assess feasibility. This can be due to numerical problems generated by poles too close to the imaginary axis. Another possible explanation can be in the present design method which does not account for the rate of variation of the parameter. Asking for the instantaneous positioning of a very selective notch can be a too severe requirement. The analysis of this phenomenon will be a subject of further research. We would like to mention an alternative approach that consists in using the internal model principle in place of the performance weightings. The relation between the two approaches will be analyzed in forthcoming work. A second weighting function $W_2(s)$ has been used to enforce high-frequent roll-off of the controller for robustness issues. An LFT representation (3) for the generalized plant has been determined where $\Delta(p) = pI$ is of size 8×8 and the performance channel $w_1 \to z_1$ of size 2×2 is defined through the standard weighted PS/KS problem. The performance specification of minimizing the L_2 gain of this channel admits the intuitive interpretation of H_∞ loop-shaping, according to the chosen weights, of the closed-loop transfer functions for "frozen" value of the parameters. Obviously, this frequency-domain interpretation is meaningful only in the case of "slowly" time-varying parameters.

3.2 Results

We designed controllers for several values of the damping coefficients ζ_p and ζ_z in order to find the best trade-off between the amount of disturbance suppression and the admissible interval of variation of the rotational frequency. For the choice $\zeta_p = 0.2$ and $\zeta_z = 2$ we got a controller that achieves exponential stability and an L_2 gain of 2.18 for $f_{rot} \in [25Hz, 35Hz]$. Five different plots of the sensitivity function and the corresponding controller for the five "frozen" values $f_{rot} = 25, 27.5, 30, 32.5, 35Hz$ are plotted respectively in figures 2 and 3, from which we can clearly see that the controller does effective adaptive notch placement. However,

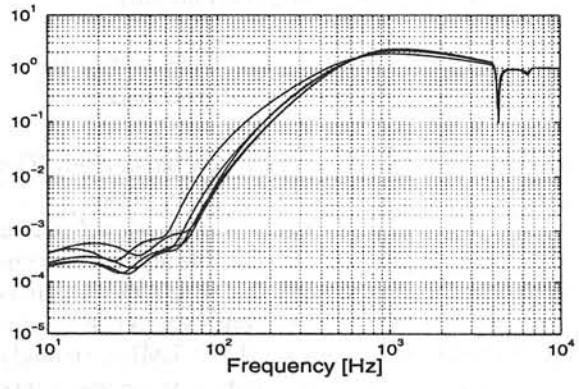


Fig. 2: Amplitude of the sensitivity function for five "frozen" values of f_{rot}

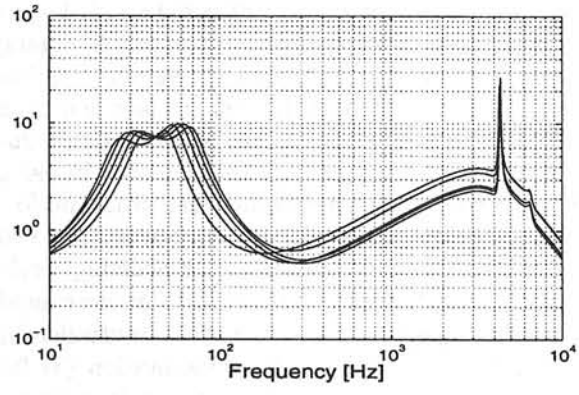


Fig. 3: Amplitude of the designed controller for five "frozen" values of f_{rot}

the significative test of the design should be done through time-domain simulations (or even better real implementation) that include also the transient effects. In fig. 4 the position error signal corresponding to a track disturbance $w(t) = \sin(2\pi f_{rot}(t)t + \phi_1) + 0.25 * \sin(4\pi f_{rot}(t)t + \phi_2)$ is shown, where ϕ_1

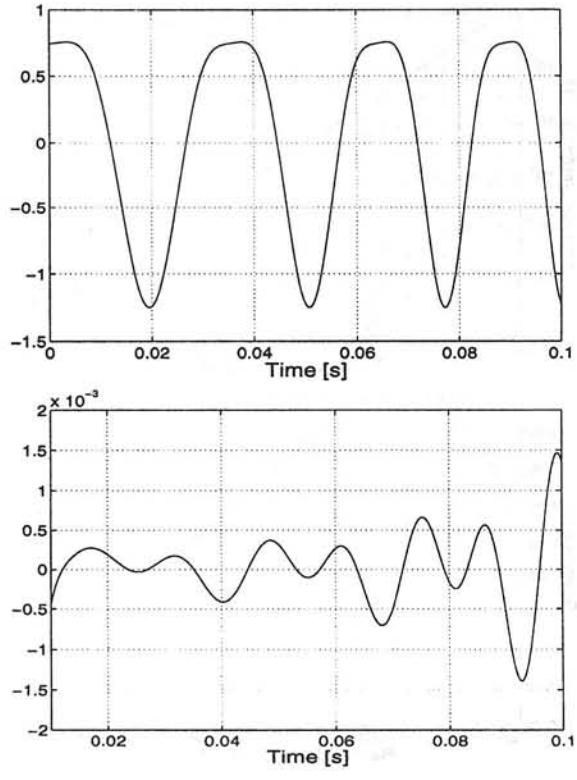


Fig. 4: Disturbance (above) and corresponding error signal (below) for linear variation of f_{rot}

and ϕ_2 are random initial phases.
In this simulation f_{rot} varies linearly from 25 Hz at $t = 0$ s to 35 Hz at $t = 0.1$ s; the linear variation of the rotational frequency is the characteristic of the sequential reading mode of the CD, although here it is assumed to be much faster for simulation purposes. We can see that the error signal remains in the prescribed amplitude bound. Finally, we have plotted in fig. 5 the position error signal corresponding to the same disturbance $w(t)$ when f_{rot} undergoes a step variation from 25 Hz to 35 Hz at $t = 0.005$ s. This behavior corresponds to the command of track jumping in the CD. After transient effects, the controller brings the error back into the desired range in about 7 milliseconds.

4 Conclusions

In this paper we have shown that LPV-based gain scheduling techniques can be used to successfully design a controller that improves the performance of the CD player mechanism, accomplishing the requirements of the new high-level applications in a broad interval of the (increased) rotational frequency. To this end we used the concept of gain scheduling not only as adaptation to plant variations, but also as a tool to impose performance spec-

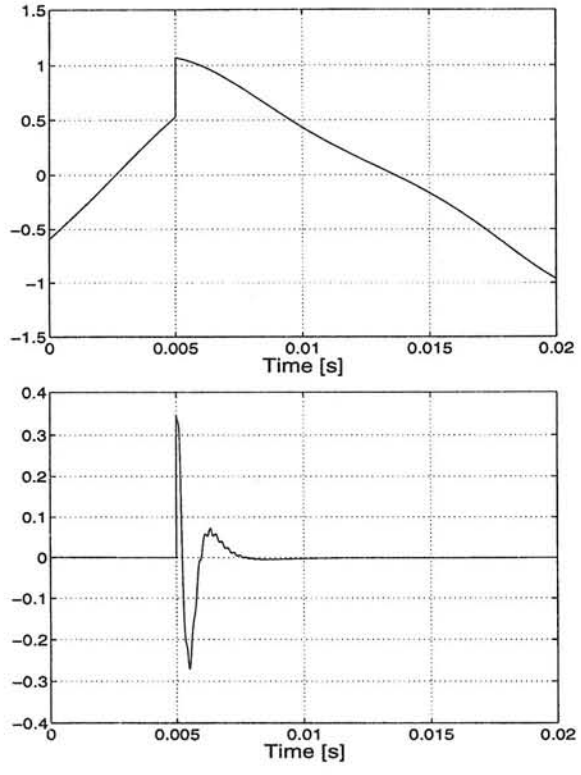


Fig. 5: Disturbance (above) and corresponding error signal (below) for step variation of f_{rot}

ifications that depend on the operating conditions.

Acknowledgments

The authors want to acknowledge Maarten Steinbuch and Meindert Norg from Philips Research Laboratories for many fruitful discussions.

References

Apkarian, P., and P. Gahinet (1995). A convex characterization of gain-scheduled H_∞ -controllers. *IEEE Trans. Autom. Control, AC-40*, 853-864.

Dötsch, H. (1998). *Identification for Control Design with Application to a Compact Disk Mechanism.* Ph.D. Thesis, Mech. Eng. Sys. and Contr. Group, Delft Univ. of Techn., The Netherlands.

Helmersson, A. (1995). *Methods for Robust Gain-Scheduling.* Ph.D. Thesis, Linköping University, Sweden.

Packard, A. (1994). Gain-scheduling via linear fractional transformations. *System and Control Letters, 22*, 79-92.

Gahinet, P., A. Nemirovski, A.J. Laub, and M. Chilali (1995). *LMI Control Toolbox.* The Mathworks Inc.

Nijo, R.G.E., C.W. Scherer, and S. Bennani (1996). Parametrically varying flight control system design with full block scalings. In: O.H. Bosgra et al. (Eds.), *Selected Topics in Identif., Modelling and Control*, Vol. 9, Delft University Press, pp. 113-120.

Scherer, C.W., P. Gahinet and M. Chilali (1997). Multi-Objective Output-Feedback Control via LMI Optimization. *IEEE Trans. Autom. Control, AC-42*, 896-911.

Scherer, C.W. (1997). Robust mixed control and LPV control with full block scalings, preprint.

Scorletti, G., and L. El Ghaoui (1995). Improved linear matrix inequality conditions for gain-scheduling. *Proc. 34th IEEE Conf. Decision and Control*, pp. 3626-3631.

Wu, F. (1995). *Control of Linear Parameter Varying Systems.* Ph.D. Thesis, University of California at Berkeley, Berkeley, USA.

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Robust stability analysis for parameter dependent systems using full block S-procedure

M. Dettori [‡] **and C.W. Scherer**

Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.
E-mail: m.dettori@wbmt.tudelft.nl

Abstract. In this paper we present an approach to analyze robust stability and robust performance of systems that depend rationally on uncertain parameters on the basis of the so-called full block S-procedure. The use of this tool allows us to derive in a straightforward manner the analysis inequalities for robust stability and robust performance in the presence of time-invariant and/or time-varying parameters. Furthermore, in the resulting tests we have the flexibility to search for the existence of an affine parameter-dependent or a rational parameter-dependent Lyapunov function. In the first case, the analysis problem is convex and, through the use of a general class of scalings, we get sufficient conditions which are shown to be less conservative than existing alternatives. In the second case, the analysis problem is not convex, but the use of a D-K type iteration can lead to improvements of the results compared to the affine case.

Keywords. Robust stability, robust performance, LMI, parametric uncertainty, parameter-dependent Lyapunov functions.

1 Introduction

A fundamental issue in control design is to assess to what extent stability and performance properties of a (controlled) system can be guaranteed in face of uncertainty or variations of the system parameters. A lot of research effort has been devoted to the investigation of this problem. One of the main approaches is based on the application of Lyapunov's stability theory and it has led to the notions of quadratic stability and affine quadratic stability for systems whose stability can be proven via a quadratic Lyapunov function that is, respectively, constant or affinely dependent on the parameters. With the recent developments in the field of semidefinite programming, this approach has become very attractive also from a computational viewpoint. In Boyd *et al.* (1994) the search for a Lyapunov function that assesses robust stability or performance of systems affinely dependent on parameters has been cast in an LMI framework. This line has been further developed in Fu and Barabanov (1997), Gahinet *et al.* (1996), Feron *et al.* (1995). In the latter paper the authors represent the parametric uncertainty through quadratic inequalities and make use of the standard S-procedure (Boyd *et al.*, 1994) to arrive to stability tests expressed in terms of a certain class of scalings.

This paper aims to show that using a generalization of the S-procedure we arrive at stability (and performance) conditions which are expressed in terms of a more general class of scalings, leading to less conservative results. Furthermore, our criteria do not apply only to system that are affinely dependent on the uncertain parameters, as usually considered, but can also handle rational dependence. As an extra novelty, we address the search for a quadratic Lyapunov function which depends rationally on the parameters. This search is based on an algorithm that is reminiscent of the D-K iteration (see e.g. Zhou *et al.* (1996)). Finally, we illustrate our results by numerical examples.

[‡]The research of Marco Dettori is sponsored by Philips Research Laboratories, Eindhoven, The Netherlands.

2 Notation

Throughout the paper we use bold face letters to indicate sets; for instance $\boldsymbol{\delta}$ is a set and δ its generic element. With $\boldsymbol{\delta} = conv(\boldsymbol{\delta_0})$ we denote that $\boldsymbol{\delta}$ is the convex hull of $\boldsymbol{\delta_0}$. With I we indicate the identity matrix of unspecified size, while I_n is an identity matrix of size $n \times n$. Given the matrices M_1,...,M_k, we indicate with $diag(M_1, ..., M_k)$ the matrix

$$\begin{pmatrix} M_1 & 0 & ... & 0 \\ 0 & M_2 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & .. & M_k \end{pmatrix}.$$

We use the following notation for an LFT representation:

$$\mathcal{F}_l\left(\left(\begin{array}{c|c} F_A & F_B \\ \hline F_C & F_D \end{array}\right), M\right) := F_A + F_B M (I - F_D M)^{-1} F_C$$

and we say that the LFT is well-posed when $I - F_D M$ is nonsingular. Sometimes, to avoid writing huge formulas we denote an inequality like $M'TM < 0$ with $(*)'TM < 0$. Finally we use the term scalings either referring to the matrix $P = \left(\begin{array}{c|c} Q & S \\ \hline S' & R \end{array}\right)$ or to its blocks Q, S and R.

3 The set-up

Consider the uncertain system:

$$\dot{x} = A(\delta)x, \quad x(0) = x_0 \tag{1}$$

where the parameter vector δ belongs to the set $\boldsymbol{\delta} = \{\delta = (\delta_1, ..., \delta_k) : \delta_j \in [\underline{\delta}_j, \bar{\delta}_j],\ j = 1, ..., k\}$. Without loss of generality we assume that the parameter values are shifted in such a way that the nominal value corresponds to $\delta = 0$, hence $0 \in \boldsymbol{\delta}$. Notice that $\boldsymbol{\delta}$ is the convex hull of the finite set $\boldsymbol{\delta^0} = \{\delta = (\delta_1, ..., \delta_k) : \delta_j \in \{\underline{\delta}_j, \bar{\delta}_j\},\ j = 1, ..., k\}$.
In this paper we focus on rational parameter-dependence that allows to represent the describing matrix as a linear fractional transformation (LFT):

$$A(\delta) = \mathcal{F}_l\left(\left(\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right), \Delta_A(\delta)\right),$$

where $\Delta_A(.)$ is a linear mapping $\mathbf{R}^k \to \mathbf{R}^{l \times q}$. It is important to notice that the image of $\boldsymbol{\delta}$ under $\Delta_A(.)$ is nothing but the convex hull of the set $\boldsymbol{\Delta_A^0} = \{\Delta_A(\delta) : \delta \in \boldsymbol{\delta^0}\}$.

4 Main result

By standard Lyapunov theory, the uncertain system (1) is asymptotically stable if and only if there exists a symmetric matrix function $X(\delta)$ on $\boldsymbol{\delta}$ such that for all $\delta \in \boldsymbol{\delta}$:

$$X(\delta) > 0 \tag{2}$$

$$A(\delta)'X(\delta) + X(\delta)A(\delta) < 0 \tag{3}$$

The latter inequality can be equivalently rewritten in the form:

$$\begin{pmatrix} A(\delta) \\ X(\delta) \end{pmatrix}' \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix} \begin{pmatrix} A(\delta) \\ X(\delta) \end{pmatrix} < 0 \text{ for all } \delta \in \boldsymbol{\delta} \tag{4}$$

that is more suitable to derive our results.
We cannot solve the problem at this level of generality, because standard algorithms do not allow to solve functional inequalities directly. We should therefore impose a structure on $X(\delta)$ in order to arrive at inequalities on matrix unknowns. Furthermore, condition (4) imposes an infinite number of constraints on the unknowns.
In this paper we try to search for a rational $X(\delta)$. This amounts to searching for the coefficients in the LFT representation:

$$X(\delta) = \mathcal{F}_l\left(\left(\begin{array}{c|c} X_A & X_B \\ \hline X_C & X_D \end{array}\right), \Delta_X(\delta)\right) \tag{5}$$

where $\Delta_X(\delta) = diag(\delta_1 I, ..., \delta_k I)$ is block diagonal and, hence, $\Delta_X(\delta) \in conv(\boldsymbol{\Delta_X^0})$, with $\boldsymbol{\Delta_X^0} = \{diag(\delta_1 I, ..., \delta_k I) : \delta_j \in \{\underline{\delta}_j, \bar{\delta}_j\},\ j = 1, ..., k\}$. This assumption is needed to enforce the symmetry constraint on $X(\delta)$ what will be subject of discussion in the next sections. Notice that the often considered affine dependence on the parameter is just one special case where X_D vanishes.
Generalizing the result of Feron *et al.* (1995) we can show that the positivity condition (2) is automatically implied by the inequality (3), provided that the LFT representation (5) is well-posed:

Lemma 4.1 *If the system (1) is nominally stable (i.e. stable for $\delta = 0$) and $I - X_D \Delta_X(\delta)$ is nonsingular for all $\delta \in \boldsymbol{\delta}$, (3) implies $X(\delta) > 0$ for all $\delta \in \boldsymbol{\delta}$.*

Proof: Condition (3) for $\delta = 0$ and nominal stability imply $X(0) = X_A > 0$. Suppose that, by contradiction, there exists a $\tilde{\delta} \in \boldsymbol{\delta}$ such that $X(\tilde{\delta})$ is not positive definite. Since $\boldsymbol{\delta}$ is pathwise connected, there exists a continuous curve $\delta(t)$, $t \in [0, 1]$, with $\delta(t) \in \boldsymbol{\delta}$ for all $t \in [0, 1]$, $\delta(0) = 0$ and $\delta(1) = \tilde{\delta}$. The hypothesis of well-posedness implies continuity of $X(\delta(t))$. Consider now the set $T = \{t \in [0, 1] : X(\delta(t))$ is not positive definite$\}$. By hypothesis, this set is non-empty and bounded; hence $t_0 = \inf T$ is finite. We claim that $X(\delta(t_0))$ is positive semidefinite but <u>not</u> positive definite. Suppose that, by contradiction, there exists an x_0 such

that the function $f(t) = x_0' X(\delta(t)) x_0$ is negative in t_0: by continuity there exists a non-empty interval $(t_0 - \epsilon, t_0]$ where $f(t)$ is negative, contradicting t_0 being a lower bound of T. Therefore $X(\delta(t_0)) \geq 0$. Suppose now that $X(\delta(t_0)) > 0$: by continuity there exists a non-empty interval $[t_0, t_0 + \epsilon)$ where $X(\delta(t)) > 0$, contradicting t_0 being the largest lower bound of T. Having proved the claim, we conclude the existence of an $\tilde{x}$ such that $X(\delta(t_0))\tilde{x} = 0$ which contradicts (3). Hence $X(\delta) > 0$ for all $\delta \in \boldsymbol{\delta}$.

The key-point of our result consists in the application of the full block S-procedure (Scherer, 1997) to the inequality (4). Roughly speaking, the full block S-procedure is a tool which allows to eliminate parameter dependence in a certain class of matrix inequalities through the introduction of scalings. Let us recall this result in our setting:

Lemma 4.2 (Full Block S-Procedure) .
Suppose $F(\delta) = \mathcal{F}_l\left(\left(\begin{array}{c|c} F_A & F_B \\ \hline F_C & F_D \end{array}\right), \Delta(\delta)\right)$ *and* T *is a symmetrix matrix. Then for all* $\delta \in \boldsymbol{\delta}$

$F(\delta)'TF(\delta) < 0$ *and* $I - F_D\Delta(\delta)$ *is nonsingular*

if and only if there exists a symmetric scaling $\begin{pmatrix} Q & S \\ S' & R \end{pmatrix}$ *such that*

$$\left(\begin{array}{cc} F_A & F_B \\ \hline 0 & I \\ F_C & F_D \end{array}\right)' \left(\begin{array}{c|cc} T & 0 & 0 \\ \hline 0 & Q & S \\ 0 & S' & R \end{array}\right) \left(\begin{array}{cc} F_A & F_B \\ \hline 0 & I \\ F_C & F_D \end{array}\right) < 0 \tag{6}$$

and

$$\begin{pmatrix} \Delta(\delta) \\ I \end{pmatrix}' \begin{pmatrix} Q & S \\ S' & R \end{pmatrix} \begin{pmatrix} \Delta(\delta) \\ I \end{pmatrix} > 0 \quad \forall \delta \in \boldsymbol{\delta}. \tag{7}$$

Notice that (6) and (7) ensure the well-posedness of the LFT representation of $F(\delta)$. The inequality (6) is independent from the parameters which only constrain the scalings through (7). Ideally one has to determine the set of all scalings that satisfy (7) and check whether there is one element in this set for which (6) holds. Unfortunately, the exact description of this set is in general hard, if not impossible. This is the reason to work with subsets that admit tractable descriptions. Obviously, the smaller the subset, the smaller the freedom to satisfy (6). Among the various possibilities for the choice of this subset, let us point out two important cases:

- If $\Delta(\delta)$ has the specific structure $\Delta(\delta) = diag(\delta_1 I, ..., \delta_k I)$ (what can be always supposed without loss of generality) one can choose the subset of the block-diagonal scalings described by:

$$\begin{aligned} Q &= diag(Q_1, ..., Q_k) < 0, \quad R = -Q \\ S &= diag(S_1, ..., S_k), \quad S_j + S_j' = 0 \end{aligned}$$

  where the partition is the same as that of $\Delta(\delta)$. The structure of this class of scalings is tailored to make the constraint (7) automatically satisfied. On the other hand, the number of scalar variables is kept relatively low what leads to fast computations.
  This class of scalings is the one considered in μ theory (the D and G scalings) and has been also used in Feron *et al.* (1995) to search for a parameter-dependent Lyapunov function.

- A larger subset of all the scalings that satisfy (7) can be implicitly parametrized by a finite number of inequalities:

$$\begin{pmatrix} \Delta(\delta) \\ I \end{pmatrix}' \begin{pmatrix} Q & S \\ S' & R \end{pmatrix} \begin{pmatrix} \Delta(\delta) \\ I \end{pmatrix} > 0 \quad \forall \delta \in \boldsymbol{\delta}^0 \tag{8}$$

  In order to let (8) imply (7), an extra concavity constraint should be imposed on Q what introduces conservatism. The simplest constraint to think of is $Q < 0$. A less conservative condition is based on partial concavity arguments and can be formulated as follows: represent $\Delta(\delta)$ as

$$\Delta(\delta) = \sum_{j=1}^{k} \delta_j R_j L_j$$

  with matrices R_j, L_j of full column, row rank respectively. Then it suffices to restrict Q as:

$$R_j' Q R_j < 0, \quad \text{for} \quad j = 1, ..., k \tag{9}$$

  Clearly this second class of scalings, full block, comprises the block-diagonal class as a special case. As a price to pay for reducing conservatism, the number of scalar variables is largely increased, slowing down numerical computations.

In implementing the tests to be presented, one has the flexibility to choose the class of scalings which is more suitable, based on conservatism against computational efficiency considerations. We stress the fact that even in the case of block-diagonal scalings, our results are new in that they allow to treat also nonlinear (rational) parameter-dependence of $A(\delta)$. To apply the full block S-procedure to the inequality (4), we have just to replace $F(\delta)$ with

$\left(A(\delta)' \;\; X(\delta)' \right)'$. It is easily seen that this function admits the LFT representation

$$\begin{pmatrix} A(\delta) \\ X(\delta) \end{pmatrix} = \mathcal{F}_l \left(\left(\begin{array}{c|cc} A & B & 0 \\ X_A & 0 & X_B \\ \hline C & D & 0 \\ X_C & 0 & X_D \end{array} \right), \begin{pmatrix} \Delta_A(\delta) & 0 \\ 0 & \Delta_X(\delta) \end{pmatrix} \right)$$

We arrive at the following result

Lemma 4.3 *There exists a function* $X(\delta) = X_A + X_B\Delta_X(\delta)(I - X_D\Delta_X(\delta))^{-1}X_C$ *which satisfies (4) if and only if there exist four matrices* X_A, X_B, X_C, X_D *and a symmetric scaling*

$$P = \left(\begin{array}{cc|cc} Q_1 & Q_{12} & S_1 & S_{12} \\ Q_{12}' & Q_2 & S_{21} & S_2 \\ \hline S_1' & S_{21}' & R_1 & R_{12} \\ S_{12}' & S_2' & R_{12}' & R_2 \end{array} \right) \tag{10}$$

such that for all $\delta \in \boldsymbol{\delta}$

$$\begin{pmatrix} * \\ * \\ * \\ * \end{pmatrix}' P \begin{pmatrix} \Delta_A(\delta) & 0 \\ 0 & \Delta_X(\delta) \\ I & 0 \\ 0 & I \end{pmatrix} > 0 \tag{11}$$

and

$$\left(\begin{array}{c} * \\ * \\ \hline * \\ * \\ * \\ * \end{array} \right)' \left(\begin{array}{cc|cccc} 0 & I & 0 & 0 & 0 & 0 \\ I & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & Q_1 & Q_{12} & S_1 & S_{12} \\ 0 & 0 & Q_{12}' & Q_2 & S_{21} & S_2 \\ 0 & 0 & S_1' & S_{21}' & R_1 & R_{12} \\ 0 & 0 & S_{12}' & S_2' & R_{12}' & R_2 \end{array} \right) \left(\begin{array}{ccc} A & B & 0 \\ X_A & 0 & X_B \\ \hline 0 & I & 0 \\ 0 & 0 & I \\ C & D & 0 \\ X_C & 0 & X_D \end{array} \right) < 0. \tag{12}$$

As we can see, (12) is not convex in X_A, X_B, X_C, X_D and the scalings together. Furthermore we should impose the constraint that $X(\delta)$ is symmetric. We will do this in the next two section, separating the case in which we seek for an affine function $X(\delta)$, from the case in which we seek for a genuine LFT function. In the first case we will obtain an LMI test, while in the second we will propose D-K-like iteration.

5 Search for affine $X(\delta)$

If we examine (12), we observe that the left hand side is affine in the unknowns if keeping X_C and X_D fixed. This motivates the choice for a Lyapunov matrix of the structure $X(\delta) = X_0 + \delta_1 X_1 + ... + \delta_k X_k$, where the X_j, $j = 0, 1, ..., k$ are $n \times n$ symmetric matrices. The LFT representation then reads as:

$$X(\delta) = \mathcal{F}_l \left(\left(\begin{array}{c|c} X_A & X_B \\ \hline V & 0 \end{array} \right), \Delta_X(\delta) \right) \tag{13}$$

where $X_A = X_0$, $X_B = \left(X_1 | ... | X_k \right)$ and $V = \left(I_n | ... | I_n \right)'$.
We observe that this parametrization of the affine $X(\delta)$ can be inefficient from a computational viewpoint, requiring each diagonal block of $\Delta_X(\delta)$ to be $n \times n$, irrespective of its 'original' size in $\Delta_A(\delta)$. Nevertheless, in this way we ensure symmetry of the Lyapunov matrix while keeping $X_C = V$ fixed.
Substituting the expression (13) in (12), we arrive at the following result:

Theorem 5.1 *The uncertain system (1) is asymptotically stable for all* $\delta \in \boldsymbol{\delta}$ *if there exist two matrices* X_A *and* X_B, *and a symmetric scaling* P *of the structure (10) that satisfies (11) and*

$$\left(\begin{array}{c} * \\ * \\ \hline * \\ * \\ * \\ * \end{array} \right)' \left(\begin{array}{cc|cccc} 0 & I & 0 & 0 & 0 & 0 \\ I & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & Q_1 & Q_{12} & S_1 & S_{12} \\ 0 & 0 & Q_{12}' & Q_2 & S_{21} & S_2 \\ 0 & 0 & S_1' & S_{21}' & R_1 & R_{12} \\ 0 & 0 & S_{12}' & S_2' & R_{12}' & R_2 \end{array} \right) \left(\begin{array}{ccc} A & B & 0 \\ X_A & 0 & X_B \\ \hline 0 & I & 0 \\ 0 & 0 & I \\ C & D & 0 \\ V & 0 & 0 \end{array} \right) < 0. \tag{14}$$

6 Search for rational $X(\delta)$

A first problem is to impose the symmetry constraint on $X(\delta) = X_A + X_B\Delta_X(\delta)(I - X_D\Delta_X(\delta))^{-1}X_C$. Due to the block diagonal structure of $\Delta_X(\delta)$, the constraints $X_A = X_A'$, $X_B = X_C'$ and $X_D = X_D'$ would ensure symmetry of $X(\delta)$. These requirements are, however, too strong. This can be seen, for instance, by considering the case of affine dependence $X(\delta) = X_A + X_B\Delta_X(\delta)X_B'$: the coefficients of the scalar δ_j's are forced to be positive semidefinite. A possibility to overcome this problem is to choose a Lyapunov matrix of the structure $Z(\delta) = X(\delta) + X(\delta)'$. In this way the symmetry is automatically ensured, at the price of increasing the size of the Δ block by a factor two. In fact, the LFT representation of $Z(\delta)$ is:

$$Z(\delta) = \mathcal{F}_l \left(\left(\begin{array}{c|cc} X_A + X_A' & X_B & X_C' \\ \hline X_C & X_D & 0 \\ X_B' & 0 & X_D' \end{array} \right), \begin{pmatrix} \Delta_X(\delta) & 0 \\ 0 & \Delta_X(\delta) \end{pmatrix} \right). \tag{15}$$

Substituting this expression in (12) we can formulate the following theorem

Theorem 6.1 *The uncertain system (1) is asymptotically stable for all* $\delta \in \boldsymbol{\delta}$ *if there exist four ma-*

trices X_A, X_B, X_C, X_D and a symmetric scaling

$$P = \left(\begin{array}{ccc|ccc} Q_1 & Q_{12} & Q_{13} & S_1 & S_{12} & S_{13} \\ Q_{12}' & Q_2 & Q_{23} & S_{21} & S_2 & S_{23} \\ Q_{13}' & Q_{23}' & Q_3 & S_{31} & S_{32} & S_3 \\ \hline S_1' & S_{21}' & S_{31}' & R_1 & R_{12} & R_{13} \\ S_{12}' & S_2' & S_{32}' & R_{12}' & R_2 & R_{23} \\ S_{13}' & S_{23}' & S_3' & R_{13}' & R_{23}' & R_3 \end{array}\right)$$

such that for all $\delta \in \boldsymbol{\delta}$

$$\left(\begin{array}{c} * \\ * \\ \hline * \\ * \\ * \\ * \end{array}\right)' P \left(\begin{array}{ccc} \Delta_A(\delta) & 0 & 0 \\ 0 & \Delta_X(\delta) & 0 \\ 0 & 0 & \Delta_X(\delta) \\ \hline I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{array}\right) > 0 \qquad (16)$$

and

$$\left(\begin{array}{c} * \\ * \\ \hline * \\ * \\ * \\ * \\ * \\ * \end{array}\right)' \left(\begin{array}{cc|cccccc} 0 & I & 0 & 0 & 0 & 0 & 0 & 0 \\ I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & Q_1 & Q_{12} & Q_{13} & S_1 & S_{12} & S_{13} \\ 0 & 0 & Q_{12}' & Q_2 & Q_{23} & S_{21} & S_2 & S_{23} \\ 0 & 0 & Q_{13}' & Q_{23}' & Q_3 & S_{31} & S_{32} & S_3 \\ 0 & 0 & S_1' & S_{21}' & S_{31}' & R_1 & R_{12} & R_{13} \\ 0 & 0 & S_{12}' & S_2' & S_{32}' & R_{12}' & R_2 & R_{23} \\ 0 & 0 & S_{13}' & S_{23}' & S_3' & R_{13}' & R_{23}' & R_3 \end{array}\right) \cdot$$

$$\cdot \left(\begin{array}{cccc} A & B & 0 & 0 \\ X_A + X_A' & 0 & X_B & X_C' \\ \hline 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \\ C & D & 0 & 0 \\ X_C & 0 & X_D & 0 \\ X_B' & 0 & 0 & X_D' \end{array}\right) < 0. \qquad (17)$$

In the inequality (17) cross-products between the scalings S and R and the variables X_B, X_C and X_D appear, rendering the problem nonconvex. A possible solution is to perform an iterative two-step procedure, reminiscent of the D-K iteration, where in each step a convex problem is solved. To this end we need to introduce a small modification in the inequality (16) by multiplying the blocks $\Delta_A(\delta)$ and $\Delta_X(\delta)$ with a factor r. This parameter represents the stability radius and the idea is to build an iteration that successively maximizes r, starting from the value $r = 0$ (corresponding to nominal stability analysis). If the value $r = 1$ is reached, then the given system is robustly stable against all the admissible parameter variations. The proposed algorithm is as follows (note than in the inequalities (11) and (16) $\Delta_A(\delta)$ and $\Delta_X(\delta)$ should be replaced by $r\Delta_A(\delta)$ and $r\Delta_X(\delta)$):

Inizialization: Maximize r over (11) and (14). This amounts to find the biggest stability radius that can be ensured with an affine parameter-dependent Lyapunov function according to Theorem 3. Let r^{aff}, X_A^{aff} and X_B^{aff} the value of r, X_A and X_B obtained in this way. Then solve (16) and (17) on the scalings, for $r = r^{\text{aff}}$, $X_A = 0.5X_A^{\text{aff}}$, $X_B = 0.5X_B^{\text{aff}}$, $X_C = V$ and $X_D = 0$. This second phase of the inizialization process is required to get the initial values of the scalings to use in the iteration. Note in fact that the scalings obtained at the end of the "affine optimization" cannot be directly used since they are smaller in size.

After this inizialization the iteration starts. The step $j-1$ leads to a Lyapunov matrix and scaling P such that (16) and (17) hold for the parameter $r = r_{j-1}$. The j-th step then proceeds in two substeps:

First substep: Fix S and R and maximize r by varying X_A, X_B, X_C, X_D and Q over (16) and (17). The obtained $\tilde{r}_j$ satisfies $r_{j-1} \leq \tilde{r}_j$.

Second substep: Fix X_B, X_C and X_D and maximize r by varying X_A, Q, S and R over (16) and (17). The obtained r_j satisfies $\tilde{r}_j \leq r_j$.

The iteration will define a nondecreasing sequence $r_1 \leq r_2 \leq \ldots$. If there is one index for which $r_j \geq 1$ then robust stability is assessed. Otherwise, this algorithm cannot guarantee robust stability.

We want to make the following remarks on the iteration:

- In the numerical implementation, the conditions (11) and (16) should be substituted by conditions on a finite number of points. As already noticed, this is done through the choice of certain subset of scalings. If we choose the block diagonal scalings, then (11) and (16) are automatically satisfied. If we choose the full block scalings, then we enforce (11) and (16) only for $\delta \in \boldsymbol{\delta}_0$ and we add the concavity constraints (9).

- The parameter r multiplies the scalings in (11) and (16). Therefore, the minimization has to be performed by bisection.

It is important to stress the fact that this iterative approach is only one of the possible ways to deal with Bilinear Matrix Inequality (BMIs are linear in two subsets of the unknowns separately) like (17). In particular, there are no general criteria that give

a systematic way to initialize the algorithm (and a wrong starting point can let the whole procedure fail) or to divide the unknowns in the two subsets. One can then make different choices and get different algorithms with different chances of success.

7 Time-varying parameters

In this case the differential equation (1) that define the system reads as:

$$\dot{x}(t) = A(\delta(t))x(t), \quad t \geq 0 \tag{18}$$

where the parameter vector δ is a continuously differentiable time-varying function $\delta : [0, \infty) \to \mathbf{R}^k$. We assume that $\delta(t) \in \boldsymbol{\delta}$ for all $t \geq 0$ and also the rate of variation is bounded: $\dot{\delta}(t) \in \dot{\boldsymbol{\delta}} = conv(\boldsymbol{\delta_0})$, $\boldsymbol{\delta_0} = \{(\dot{\delta}_1, ..., \dot{\delta}_k) \in \mathbf{R}^k : \dot{\delta}_j \in \{\underline{\beta}_j, \bar{\beta}_j\},\ j = 1, ..., k\}$ for all $t \geq 0$. As we did for $\boldsymbol{\delta}$, we assume without loss of generality that $0 \in \boldsymbol{\delta_0}$.
The Lyapunov criterion in this case reads as (Scherer, 1995):

Theorem 7.1 *If there exists a continuously differentiable and symmetric function $X : \boldsymbol{\delta} \to \mathbf{R}^{n \times n}$ such that for all $\delta \in \boldsymbol{\delta}$ and all $\dot{\delta} \in \dot{\boldsymbol{\delta}}$,*

$$\sum_{j=1}^{k} \frac{\partial X}{\partial \delta_j}(\delta)\dot{\delta}_j + A(\delta)'X(\delta) + X(\delta)A(\delta) < 0, \tag{19}$$

then the uncertain system (18) is exponentially stable.

Analogously to the time-invariant case it can be shown that the positivity condition $X(\delta) > 0$ is automatically implied by (19), provided that $(I - X_D\Delta_X(\delta))$ is nonsingular for all $\delta \in \boldsymbol{\delta}$.
Let us assign to $X(\delta)$ the function $\dot{X}(\delta, \dot{\delta})$ given by

$$\dot{X}(\delta, \dot{\delta}) = \sum_{j=1}^{k} \frac{\partial X}{\partial \delta_j}(\delta)\dot{\delta}_j. \tag{20}$$

The condition (19) can then be rewritten in the form:

$$\begin{pmatrix} I \\ A(\delta) \\ X(\delta) \\ \dot{X}(\delta, \dot{\delta}) \end{pmatrix}' \begin{pmatrix} 0 & 0 & 0 & \frac{1}{2}I \\ 0 & 0 & I & 0 \\ 0 & I & 0 & 0 \\ \frac{1}{2}I & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} I \\ A(\delta) \\ X(\delta) \\ \dot{X}(\delta, \dot{\delta}) \end{pmatrix} < 0 \tag{21}$$

for all $\delta \in \boldsymbol{\delta}$, $\dot{\delta} \in \dot{\boldsymbol{\delta}}$.
The application of the full block S-procedure is straightforward, once an LFT representation for $\left(A(\delta)'\ X(\delta)'\ \dot{X}(\delta, \dot{\delta})'\right)'$ has been determined. This is made explicit in the next two subsection for the affine and for the rational case.

7.1 $X(\delta)$ affine

In this case $X(\delta) = X_A + X_B\Delta_X(\delta)V$, and hence $\dot{X}(\delta, \dot{\delta}) = X_B\Delta_X(\dot{\delta})V$. The required LFT representation is therefore:

$$\begin{pmatrix} A(\delta) \\ X(\delta) \\ \dot{X}(\delta, \dot{\delta}) \end{pmatrix} = \tag{22}$$

$$\mathcal{F}_l\left(\left(\begin{array}{c|ccc} A & B & 0 & 0 \\ X_A & 0 & X_B & 0 \\ 0 & 0 & 0 & X_B \\ \hline C & D & 0 & 0 \\ X_C & 0 & 0 & 0 \\ X_C & 0 & 0 & 0 \end{array}\right), \begin{pmatrix} \Delta_A(\delta) & 0 & 0 \\ 0 & \Delta_X(\delta) & 0 \\ 0 & 0 & \Delta_X(\dot{\delta}) \end{pmatrix}\right).$$

The application of the full block S-procedure leads to the following result

Theorem 7.2 *The uncertain system (18) is exponentially stable for all $\delta \in \boldsymbol{\delta}$ and all $\dot{\delta} \in \dot{\boldsymbol{\delta}}$ if there exist two matrices X_A and X_B, and a symmetric scaling*

$$P = \left(\begin{array}{ccc|ccc} Q_1 & Q_{12} & Q_{13} & S_1 & S_{12} & S_{13} \\ Q_{12}' & Q_2 & Q_{23} & S_{21} & S_2 & S_{23} \\ Q_{13}' & Q_{23}' & Q_3 & S_{31} & S_{32} & S_3 \\ \hline S_1' & S_{21}' & S31' & R_1 & R_{12} & R_{13} \\ S_{12}' & S_2' & S_{32}' & R_{12}' & R_2 & R_{23} \\ S_{13}' & S_{23}' & S_3' & R_{13}' & R_{23}' & R_3 \end{array}\right)$$

such that for all $(\delta, \dot{\delta}) \in \boldsymbol{\delta} \times \dot{\boldsymbol{\delta}}$

$$\begin{pmatrix} * \\ * \\ * \\ \hline * \\ * \\ * \end{pmatrix}' P \left(\begin{array}{ccc} \Delta_A(\delta) & 0 & 0 \\ 0 & \Delta_X(\delta) & 0 \\ 0 & 0 & \Delta_X(\dot{\delta}) \\ \hline I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{array}\right) > 0 \tag{23}$$

and

$$\begin{pmatrix} * \\ * \\ * \\ * \\ \hline * \\ * \\ * \\ * \\ * \end{pmatrix}' \left(\begin{array}{cccc|c} 0 & 0 & 0 & \frac{1}{2}I & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ \frac{1}{2}I & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & P \end{array}\right) \left(\begin{array}{cccc} I & 0 & 0 & 0 \\ A & B & 0 & 0 \\ X_A & 0 & X_B & 0 \\ 0 & 0 & 0 & X_B \\ \hline 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \\ C & D & 0 & 0 \\ V & 0 & 0 & 0 \\ V & 0 & 0 & 0 \end{array}\right) < 0. \tag{24}$$

Like in the time-invariant case, this last condition is an LMI whose feasibility can be directly checked,

once the set of scalings has been chosen.
Notice that we have applied the full block S-procedure introducing scalings for both parameters δ and $\dot{\delta}$. As an alternative, one could introduce scalings only for δ. This would lead to an inequality similar to (24) that depends on $\dot{\delta}$. The key-point is that the parameter $\dot{\delta}$ enters affinely, as can be seen in (19), so that the resulting problem will be again an LMI problem. In this second case we have a scaling P of a smaller size, at the expense of solving an inequality like (24) for each extreme point $\dot{\delta} \in \dot{\boldsymbol{\delta}}$.

7.2 $X(\delta)$ rational

In the case where the Lyapunov matrix is described as (15), we can give the following LFT representation of $\dot{Z}(\delta, \dot{\delta})$:

$$\dot{Z}(\delta,\dot{\delta}) = \mathcal{F}_l\left(\left(\begin{array}{c|ccccccc} 0 & 0 & 0 & X_B & X_C' & X_B & X_C' \\ \hline X_C & X_D & 0 & 0 & 0 & 0 & 0 \\ X_B' & 0 & X_D' & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & X_D & 0 & X_D & 0 \\ 0 & 0 & 0 & 0 & X_D' & 0 & X_D' \end{array}\right), \right.$$
$$\left.\begin{pmatrix} \Delta_X(\delta) & 0 & 0 & 0 \\ 0 & \Delta_X(\delta) & 0 & 0 \\ \Delta_X(\dot{\delta}) & 0 & 0 & 0 \\ 0 & \Delta_X(\dot{\delta}) & 0 & 0 \\ 0 & 0 & \Delta_X(\delta) & 0 \\ 0 & 0 & 0 & \Delta_X(\delta) \end{pmatrix}\right).$$

Applying the full block S-procedure we arrive at the following result:

Theorem 7.3 *The uncertain system (18) is exponentially stable for all $\delta \in \boldsymbol{\delta}$ if there exist four matrices X_A, X_B, X_C, X_D and a symmetric scaling*

$$P = \left(\begin{array}{c|c} Q & S \\ \hline S & R \end{array}\right)$$

such that (25) and (26) are satisfied for all $(\delta, \dot{\delta}) \in \boldsymbol{\delta} \times \dot{\boldsymbol{\delta}}$.

Despite of the bigger size of this inequalities (due to the choice made to enforce the symmetry constraint on $Z(\delta)$) the structure is the same as that of the time-invariant case. A similar D,K-like iteration procedure can be applied to numerically solve the problem.

8 Robust performance

Our results for robust stability extend in a straightforward manner to robust performance analysis. Let us consider the uncertain system

$$\begin{aligned} \dot{x}(t) &= A(\delta(t))x(t) + B(\delta(t))w_p(t), \quad x(0) = 0 \\ z(t) &= C(\delta(t))x(t) + D(\delta(t))w_p(t), \end{aligned}$$

where the parameter vector δ is a continuously differentiable time-varying function $\delta : [0, \infty) \to \mathbf{R}^k$. As before $\delta(t) \in \boldsymbol{\delta}$ and $\dot{\delta}(t) \in \dot{\boldsymbol{\delta}}$ for all $t \geq 0$. Suppose the system can be given the following LFT representation:

$$\begin{pmatrix} A(\delta) & B(\delta) \\ C(\delta) & D(\delta) \end{pmatrix} = \mathcal{F}_l\left(\left(\begin{array}{cc|c} A & B_1 & B_2 \\ C_1 & D_1 & D_{12} \\ \hline C_2 & D_{21} & D_2 \end{array}\right), \Delta_\Sigma(\delta)\right). \tag{27}$$

We consider the so-called Quadratic Performance specification: the performance is defined by the matrix

$$P_p = \left(\begin{array}{c|c} Q_p & S_p \\ \hline S_p' & R_p \end{array}\right), \quad R_p \geq 0 \tag{28}$$

and we say that the system (27) has robust quadratic performance if it is robustly stable and there exists an $\epsilon > 0$ such that for every $w_p \in \mathcal{L}_2$:

$$\int_0^\infty \begin{pmatrix} w_p(t) \\ z_p(t) \end{pmatrix}' P_p \begin{pmatrix} w_p(t) \\ z_p(t) \end{pmatrix} \leq -\epsilon \|w_p\|_2^2. \tag{29}$$

Notice that the quadratic performance spec admits as special cases the $\mathcal{H}_\infty$ spec for $Q_p = -\gamma^2 I$, $S_p = 0$ and $R_p = I$ and the positive real spec for $Q_p = 0$, $S_p = -\frac{1}{2}I$ and $R_p = 0$. Furthermore the $\mathcal{H}_2$ and the generalized $\mathcal{H}_2$ cases can be obtained with little modifications. The following analysis result holds (Scherer, 1995):

Theorem 8.1 *The uncertain system (27) has robust quadratic performance if there exists a symmetric matrix function $X(\delta) > 0$ such that for all $\delta \in \boldsymbol{\delta}$ and all $\dot{\delta} \in \dot{\boldsymbol{\delta}}$*

$$\begin{pmatrix} * \\ * \\ * \\ * \\ \hline * \\ * \end{pmatrix}' \left(\begin{array}{cccc|cc} 0 & 0 & 0 & \frac{1}{2}I & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 & 0 \\ \frac{1}{2}I & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & Q_p & S_p \\ 0 & 0 & 0 & 0 & S_p' & R_p \end{array}\right) \left(\begin{array}{cc} I & 0 \\ A(\delta) & B(\delta) \\ X(\delta) & 0 \\ \dot{X}(\delta,\dot{\delta}) & 0 \\ \hline 0 & I \\ C(\delta) & D(\delta) \end{array}\right). \tag{30}$$

To apply the full block S-procedure we only need to have the LFT representation of the outer factor. The discussion for the robust stability case can be repeated with minor modifications for the robust performance case, so we can easily derive the tests for the existence of an affine $X(\delta)$ or for a rational one.

9 Numerical examples

In this section we present some numerical examples of the application of the proposed stability tests. All the computations have been performed using the *LMI Lab* of the *LMI Control Toolbox* for Matlab

$$\begin{pmatrix} * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ \hline * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \end{pmatrix}' \begin{pmatrix} \Delta_A(\delta) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \Delta_X(\delta) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \Delta_X(\delta) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Delta_X(\delta) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \Delta_X(\delta) & 0 & 0 \\ 0 & 0 & 0 & \Delta_X(\dot{\delta}) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \Delta_X(\dot{\delta}) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \Delta_X(\delta) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \Delta_X(\delta) \\ \hline I & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I \end{pmatrix} P \begin{pmatrix} * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ \hline * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \end{pmatrix} > 0 \tag{25}$$

$$\begin{pmatrix} * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \end{pmatrix}' \left(\begin{array}{ccccc|c} 0 & 0 & 0 & \frac{1}{2}I & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ \frac{1}{2}I & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & P \end{array}\right) \left(\begin{array}{c|cccccccccc} I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A & B & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ X_A + X_A' & 0 & X_B & X_C' & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & X_B & X_C' & X_B & X_C' \\ \hline C & D & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ X_C & 0 & X_D & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ X_B' & 0 & 0 & X_D' & 0 & 0 & 0 & 0 & 0 & 0 \\ X_C & 0 & 0 & 0 & X_D & 0 & 0 & 0 & 0 & 0 \\ X_B' & 0 & 0 & 0 & 0 & X_D' & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & X_D & 0 & X_D & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & X_D' & 0 & X_D' \end{array}\right) < 0 \tag{26}$$

(Gahinet *et al.*, 1995). We consider the uncertain system defined by

$$A(\delta) = \begin{pmatrix} -1 & \delta_1 & 0 & \delta_2 \\ 0.5\delta_1 & -2 & 0.5\delta_2 & 0 \\ 2a\delta_1 & 0 & -3 + a\delta_2 & 0 \\ 0 & -2a\delta_1 & 0 & -4 - a\delta_2 \end{pmatrix}$$

for several values of the parameter a. We define the stability radius ρ as the largest positive number for which the algorithm can assess the stability of the uncertain system for all parameters in the set $\{(\delta_1, \delta_2) : |\delta_j| \leq \rho,\ j = 1, 2\}$. In the following figure we compare the stability radia we have obtained for five different values of a using the algorithm proposed in Theorem 3 with full block scalings and the algorithm proposed in Feron *et al.* (1995).

From the figure we see that the proposed algorithm performs always better and the performance improvement increases with the value of a: for $a = 0.1$ is about 10% and for $a = 1$ is about 34%.

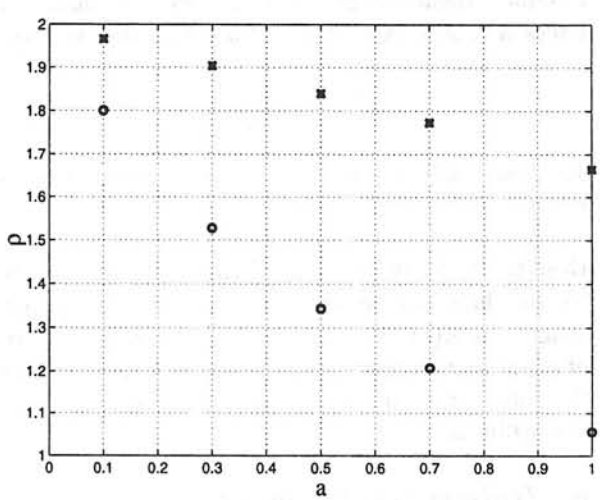


Fig. 1: Values of ρ obtained with the algorithm in Theorem 3 (*) and with the algorithm in Feron *et al.* (1995) (o)

10 Conclusions

In this paper we have presented a general result based on the full block S-procedure from which is possible to derive in a straightforward manner several robust stability and robust performance criteria. As novelties, these criteria allow to deal with rational parameter dependence of the uncertain system and to address the search of a rational parameter-dependent Lyapunov function. Finally, we have shown with numerical examples that, even in the standard case of affine parameter-dependent systems with affine parameter-dependent Lyapunov functions, the proposed criterion performs better than existing alternatives.

References

Boyd, S., L. El Ghaoui, E. Feron, and V. Balakrishnan (1994). *Linear Matrix Inequalities in System and Control Theory.* SIAM Studies in Applied Mathematics, Philadelphia.

Feron, E., P. Apkarian, and P. Gahinet (1995). S-procedure for the analysis of control systems with parametric uncertainties via parameter-dependent Lyapunov functions. *Proc. European Control Conf.*, Rome, Italy, pp. 141-146.

Fu, M., and N. E. Barabanov (1997). Improved upper bounds for the mixed structured singular value. *IEEE Trans. Autom. Control, AC-42*, 1447-1452.

Gahinet, P., P. Apkarian, and M. Chilali (1996). Affine parameter-dependent Lyapunov functions and real parametric uncertainty. *IEEE Trans. Autom. Control, AC-41*, 436-442.

Gahinet, P., A. Nemirovski, A.J. Laub, and M. Chilali (1995). *LMI Control Toolbox.* The Mathworks Inc.

Scherer, C.W. (1995). Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control. In: A. Isidori (Ed.), *Trends in Control, A European Perspective.* Springer Verlag, Berlin, pp. 173-216.

Scherer, C.W. (1997). A full block S-procedure with applications. *Proc. 36th IEEE Conf. Decision and Control*, San Diego, CA, pp. 2602-2607.

Zhou, K., J.C. Doyle, and K. Glover (1996). *Robust and Optimal Control.* Prentice Hall.

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Robust performance analysis for parameter dependent systems using tensor product splines[‡]

Carsten W. Scherer[§]

Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD, Delft, The Netherlands
E-mail: c.w.scherer@wbmt.tudelft.nl

Abstract. We show that tensor product splines form a suitable tool to numerically construct parameter dependent Lyapunov functions to test robust performance for systems that depend on time-varying parameters. Based on the excellent approximation power of this class of multivariate splines, we show how it is possible to avoid conservatism in the corresponding robust performance analysis test.

Keywords. Robust performance, linear parameter varying systems, parameter dependent Lyapunov functions, linear matrix inequalities, tensor product splines.

1 Introduction

In this paper we reveal how to arrive at non-conservative tests for the stability and performance analysis of linear systems that depend possibly non-linearly on a time-varying parameter. The approach is based on finding a parameter dependent Lyapunov matrix that defines a Lyapunov function which is quadratic in the system's state. It is well-known that such a stability test can be reduced to solving a partial differential linear matrix inequality (PDLMI) for the parameter dependent Lyapunov matrix.

In the literature, several approaches have been proposed to tackle such problems. It is known how to reduce the search for a solution of the PDLMI by using basis functions and a gridding of the parameter space in order to arrive at a finite number of conventional LMIs that are amenable to standard algorithms. It remains unclear which basis functions to choose, and how to perform the gridding in order to guarantee the validity of the PDLMI throughout the parameter set.

The purpose of the present paper is to show that tensor product splines form a systematic choice of basis functions that is guaranteed to be successful irrespective of the particular dependence of the system on the parameter. Secondly, we reveal that scaling techniques allow to guarantee the validity of the PDLMIs even if choosing for a coarse subdivision of the parameter set.

The proposed approach is seen as a generalization of those based on constructing continuous and piecewise affine solutions to the PDLMI (Johansson and Rantzer, 1997; Lim and How, 1997). Our technique will allow to construct piecewise polynomial Lyapunov matrices with an arbitrary order of smoothness over the whole parameter set without the need to worry about interpolation conditions on the boundary of the sub-boxes in a partition, and without requiring the exclusion of certain parameter hypersurfaces. In addition, we will prove that a sufficiently fine parameter set partition will guarantee the success of the algorithm. Finally, we will show that this technique can be employed even if the system matrices are not rational in the parameter, by simply approximating such non-linearities with multivariate splines.

[‡]This paper was presented at the 37th IEEE Conference on Decision and Control, 16-18 December 1998, Tampa, FL, USA.

[§]The author would like to thank Marco Dettori and Jan van Hulzen (Delft University of Technology) for many helpful discussions.

2 Tensor product splines

Let us recall some bits of the theory of tensor product splines as exposed in the excellent monograph by Schumaker (1981). Suppose the parameter set is given as

$$\delta = [a_1, b_1] \times \cdots \times [a_d, b_d].$$

Let each interval be partitioned as

$$a_i = \delta_{i,0} < \delta_{i,1} < \cdots < \delta_{i,k_i} < \delta_{i,k_i+1} = b_i.$$

On $[a_i, b_i]$, we consider the space of all functions whose restriction to $[\delta_{i,l}, \delta_{i,l+1}]$, $l = 0, \ldots, k_i$, is a polynomial of degree at most $m_i - 1$, and that is $m_i - 2$ times continuously differentiable. (Since we will typically choose $m_i = 3, 2, 1$ to obtain polynomial pieces of degree $2, 1, 0$, we work with functions that are continuously differentiable ($m_i = 3$), continuous ($m_i = 2$) or piecewise constant ($m_i = 1$). Everything what will be said holds true if we allow for variations in smoothness over breakpoints.) This space of splines of order m_i is known to have dimension $m_i + k_i$. It admits a basis that consists of B-splines $B_{i,j}(.)$, $j = 1, \ldots, m_i + k_i$. These functions have minimal support and they form a partition of unity. Using suitable recursion formulas, one can easily calculate the values and the derivatives of these functions in a numerically stable fashion. We will require the following knowledge about their support: On the interval $(\delta_{i,l}, \delta_{i,l+1})$, $l = 0, \ldots, k_i$, the functions $B_{i,j}(.)$ are strictly positive for $j = l+1, \ldots, l+m_i$ and all the others vanish on the closure $[\delta_{i,l}, \delta_{i,l+1}]$.
The corresponding space of tensor-product splines $\mathcal{S}$ is defined as the span of the functions

$$B_j(\delta) := \prod_{i=1}^{d} B_{i,j_i}(\delta_i)$$

parametrized by $j := (j_1, ..., j_d)$, with $1 \le j_i \le m_i + k_i$. Since all these functions are linearly independent, the space $\mathcal{S}$ has dimension $\prod_{i=1}^{d}(m_i + k_i)$. More importantly, on the box

$$\delta_l = \delta_{(l_1,\ldots,l_d)} := [\delta_{1,l_1}, \delta_{1,l_1+1}] \times \cdots \times [\delta_{d,l_d}, \delta_{d,l_d+1}]$$

parametrized by $l := (l_1, ..., l_d)$ only the functions B_j with $l_i + 1 \le j_i \le l_i + m_i$ do not vanish such that any $s \in \mathcal{S}$ has the representation

$$s(\delta_1, \ldots, \delta_d) = \sum_{j_1=l_1+1}^{l_1+m_1} \cdots \sum_{j_d=l_d+1}^{l_d+m_d} \alpha_{(j_1,\ldots,j_d)} \prod_{i=1}^{d} B_{i,j_i}(\delta_i).$$

Hence, only $\prod_{i=1}^{d} m_i$ of all possible $\prod_{i=1}^{d}(m_i + k_i)$ coefficients affect the linear combination due to the small support of the B-splines. This is particularly important since m_i can be kept small and fixed, whereas k_i might have to be taken large in order to reduce the approximation error in quasi-interpolation schemes. Finally, we clearly have

$$B_j(\delta) = \sum_{\nu_1=0}^{m_1-1} \cdots \sum_{\nu_d=0}^{m_d-1} \beta_{j,l,(\nu_1,\ldots,\nu_d)} \prod_{i=1}^{d} (\delta_i - \delta_{i,l_i})^{\nu_i}$$

on the box δ_l with coefficients that can be efficiently calculated; any non-vanishing B_j on the box δ_l is a multivariable polynomial with degree at most $\prod_{i=1}^{d}(m_i - 1)$.
This discussion leads us to an essential insight to arrive at our computational algorithms.

Lemma 1 *There exist matrices $A_{j,l}$, B, $C_{j,l}$, and D of dimension not larger than $\prod_{i=1}^{d} m_i$ such that*

$$B_j(\delta) = \mathcal{F}_l\left(\left(\begin{array}{c|c} A_{j,l} & B \\ \hline C_{j,l} & D \end{array}\right), \Delta(\delta - \delta_l)\right) \quad on \quad \delta_l$$

where $\Delta(\delta) = diag\,(\delta_1 I, ..., \delta_d I)$.

Remark. Note that one could as well let the degree of the polynomials vary with the parameter subboxes such that B and D would vary in size; this would allow to control the number of variables in the LMIs to be derived.
As a fundamental benefit in relying on spline-spaces, we can resort to excellent non-trivial results about the approximation power of tensor product splines. These results are based on quasi-interpolation operators that lead to computable approximation schemes and that realize, at the same time, the best theoretical approximation order. Therefore, these quasi-interpolation operators are practically useful to approximate general multivariable nonlinearities. Let us denote by $B(\delta)$ the Banach space of all bounded functions on δ equipped with the norm $\|f\|_\infty := \sup_{\delta \in \delta} |f(\delta)|$.

Theorem 2 *Consider an arbitrary sequence of partitions of the parameter box δ for which the measure*

$$\max_{i=1}^{d} \max_{j=0}^{k_i} |\delta_{i,j+1} - \delta_{i,j}| \qquad (1)$$

of the mesh size converges to zero. For each partition there exists a bounded linear (quasi-interpolation) projection mapping $\mathcal{Q} : B(\delta) \to \mathcal{S}$ such that the resulting sequence of operators has the following properties: If $f \in B(\delta)$, the uniform approximation error

$$\|f - \mathcal{Q}f\|_\infty$$

converges to zero. If f is continuously differentiable and if $m_i > 2$, then even

$$\|f - \mathcal{Q}f\|_\infty + \sum_{i=1}^{d} \|\frac{\partial}{\partial \delta_i}(f - \mathcal{Q}f)\|_\infty$$

converges to zero.

By letting the coarseness of the partition converge to zero, we can hence let $\mathcal{Q}f \in \mathcal{S}$ uniformly converge to f. If f and $\mathcal{Q}f$ are continuously differentiable, the uniform approximation error for the first order partial derivatives converges to zero as well. This result differs somewhat from that given in Schumaker (1981; Theorem 12.7) but it can be proved along the same lines.

3 Model description

Let us assume that the system to be analyzed is described as

$$\begin{aligned} \dot{x} &= A(\delta(t))x + B(\delta(t))w \\ z &= C(\delta(t))x + D(\delta(t))w \end{aligned} \qquad (2)$$

where $\delta(.)$ is any (continuously differentiable) parameter curve that satisfies $\delta(t) \in \boldsymbol{\delta}$, $\dot{\delta}(t) \in \dot{\boldsymbol{\delta}}$. The value of the parameter curve is contained in the parameter box $\boldsymbol{\delta} = \{\delta = (\delta_1, \ldots, \delta_d) \in \mathbb{R}^d \mid \delta_i \in [a_i, b_i]\}$ whereas its derivative is assumed to be contained in the rate box $\dot{\boldsymbol{\delta}} = \{\dot{\delta} = (\dot{\delta}_1, \ldots, \dot{\delta}_d) \in \mathbb{R}^d \mid \dot{\delta}_i \in [\dot{a}_i, \dot{b}_i]\}$. The system itself is described with possibly non-linear functions $A(.)$, $B(.)$, $C(.)$, $D(.)$.

Assumptions. $A(.)$, $B(.)$, $C(.)$, $D(.)$ are continuous on the parameter box $\boldsymbol{\delta}$. There exists a $\delta_0 \in \boldsymbol{\delta}$ for which $A(\delta_0)$ has all its eigenvalues in the open left-half plane. Finally, the rate box $\dot{\boldsymbol{\delta}}$ contains 0. (Note that we allow for $\dot{a}_i = \dot{b}_i = 0$ to capture the situation that $\delta_i(.)$ is known to be constant.)

4 Problem formulation

Our goal is to characterize whether the system (2) is robustly performing as characterized in the following easily proved result.

Theorem 3 *With*

$$P_0 = \left(\begin{array}{ccc|cc} 0 & \frac{1}{2}I & 0 & 0 & 0 \\ \frac{1}{2}I & 0 & I & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & -\gamma I & 0 \\ 0 & 0 & 0 & 0 & I \end{array}\right),$$

suppose there exists a continuously differentiable symmetric valued function $X(.)$ on $\boldsymbol{\delta}$ such that

$$\begin{pmatrix} * \\ * \\ * \\ \hline * \\ * \end{pmatrix}^T P_0 \left(\begin{array}{cc} \sum_{i=1}^{d} \dot{\delta}_i \frac{\partial X}{\partial \delta_i}(\delta) & 0 \\ I & 0 \\ X(\delta)A(\delta) & X(\delta)B(\delta) \\ \hline 0 & I \\ C(\delta) & D(\delta) \end{array}\right) < 0 \quad (3)$$

holds for all $\delta \in \boldsymbol{\delta}$ and $\dot{\delta} \in \dot{\boldsymbol{\delta}}$. Then, for any admissible parameter curve, system (2) is exponentially stable and the square of the L_2-gain of $w \to z$ is smaller than γ.

Remark 1. The function $X(.)$ is guaranteed to be positive definite: If we plug in $\delta = \delta_0$, $\dot{\delta} = 0$, we infer from the stability of $A(\delta_0)$ that $X(\delta_0) > 0$. Since $\boldsymbol{\delta}$ is connected and the inequality (3) prevents $X(.)$ from becoming non-singular, the claim follows.

Remark 2. The solvability of (3) is sufficient for guaranteeing exponential stability and the L_2-gain bound. In case of $\dot{\boldsymbol{\delta}} = \{0\}$, it can be shown that the converse holds as well and the characterization in this result is *exact*. It is well-known that, alternatively, one could characterize robust performance by solving a (frequency parametrized) family of real SSV-problems. Therefore, deciding the solvability of (3) corresponds to *exactly solving this family of pure real SSV-problems.* The benefit of working with (3) is to avoid frequency gridding what is particularly crucial in view of the possible discontinuity of the real SSV in the frequency.

Our main goal in this paper is to provide general tools that allow to find the infimal attenuation level γ for which (3) still has a solution.

5 Model approximation

In a concrete problem, the parameter δ often enters $A(.)$, $B(.)$, $C(.)$, $D(.)$ in a non-linear fashion. Our technique for robust performance analysis will rely on a Linear Fractional Transformation (LFT) representation of these functions. In order to arrive at such a representation, it is customary to approximate the non-linearities by rational functions of the parameters. Typically, one would resort to polynomial interpolation or approximation in order to arrive at an LFT representation that is valid over the whole parameter range. However, it is well-known that polynomial approximation shows a pretty poor global behavior. It has been argued that approximation by splines is an excellent alternative to overcome the shortcomings of polynomial approximation techniques. In addition, simple and practically useful quasi-interpolation schemes allow to systematically approximate non-linearities (Schumaker, 1981). In practice, the non-linearities enter

(2) sparsely, and they often depend on one parameter only. Hence one can resort to univariate spline approximation such that the resulting overall approximant is most easily constructed as a tensor product spline. This is one of the motivations to prefer tensor product splines over the more general box splines (de Boor et al., 1993).

In summary, by Theorem 2, for any $\epsilon > 0$ we can find a partition of the parameter box $\boldsymbol{\delta}$ into subboxes $\boldsymbol{\delta}_l$ such that

$$\left\| \begin{pmatrix} A(.) & B(.) \\ C(.) & D(.) \end{pmatrix} - \begin{pmatrix} \mathcal{Q}A(.) & \mathcal{Q}B(.) \\ \mathcal{Q}C(.) & \mathcal{Q}D(.) \end{pmatrix} \right\|_\infty < \epsilon$$

where $\mathcal{Q}A(.)$, $\mathcal{Q}B(.)$, $\mathcal{Q}C(.)$, $\mathcal{Q}D(.)$ are tensor product splines as described in Section 2. By Lemma 1, these approximants hence admit the LFT representation

$$\mathcal{F}_l \left(\left(\begin{array}{cc|c} A_l & B_l^1 & B_l^2 \\ C_l^1 & D_l^{11} & D_l^{12} \\ \hline C_l^2 & D_l^{21} & D_l^{22} \end{array} \right), \Delta(. - \delta_l) \right) \qquad (4)$$

on the box $\boldsymbol{\delta}_l$.

Remark. If the parameters do not vary in time, it suffices to use $m_i = 1$ such that $\mathcal{Q}A(.)$, $\mathcal{Q}B(.)$, $\mathcal{Q}C(.)$, $\mathcal{Q}D(.)$ will be piecewise constant functions. In general, one would rather choose $m_i = 2$ to arrive at continuous approximants, but smoothness of higer order is not required for the discussion to follow.

Instead of solving (3), we now rather consider the inequality obtained by replacing $A(.)$, $B(.)$, $C(.)$, $D(.)$ with their approximants $\mathcal{Q}A(.)$, $\mathcal{Q}B(.)$, $\mathcal{Q}C(.)$, $\mathcal{Q}D(.)$:

$$\left(\begin{array}{c} * \\ * \\ * \\ \hline * \\ * \end{array} \right)^T P_0 \left(\begin{array}{cc} \sum_{i=1}^d \dot{\delta}_i \frac{\partial X}{\partial \delta_i}(\delta) & 0 \\ I & 0 \\ \hline X(\delta)\mathcal{Q}A(\delta) & X(\delta)\mathcal{Q}B(\delta) \\ 0 & I \\ \mathcal{Q}C(\delta) & \mathcal{Q}D(\delta) \end{array} \right) < 0. \qquad (5)$$

6 Construction of solutions

We search for solutions of (5) of the form

$$X(\delta) = \sum_j X_j B_j(\delta) \qquad (6)$$

with symmetric matrix coefficients X_j. Here the basis functions for the underlying spline space is constructed with respect to the same partition used in the system representation; of course, one could also first refine this partition if necessary.

It is obvious that (5) turns into an infinite family of LMIs in X_j that is parametrized with δ and $\dot{\delta}$. Note that the right-hand side is a rational function in $(\delta, \dot{\delta})$. In general, it will not suffice to solve the inequality at the extreme points of the parameter box $\boldsymbol{\delta}_l$ in order to guarantee its validity throughout the whole box. Fortunately, there is a systematic technique how we can still reduce the infinite family of LMIs to a finite family.

For that purpose we recall that $B_j(.)$, $\frac{\partial B_j}{\partial \delta_i}(.)$ and the system approximants admit LFT representations on $\boldsymbol{\delta}_l$. By standard LFT calculus,

$$\left(\begin{array}{cc} \sum_j X_j \sum_{i=1}^d \dot{\delta}_i \frac{\partial B_j}{\partial \delta_i}(\delta) & 0 \\ \sum_j X_j B_j(\delta)\mathcal{Q}A(\delta) & \sum_j X_j B_j(\delta)\mathcal{Q}B(\delta) \\ \hline \mathcal{Q}C(\delta) & \mathcal{Q}D(\delta) \end{array} \right)$$

admits, on the box $\boldsymbol{\delta}_l$, a lower LFT representation with coefficient matrix

$$\left(\begin{array}{cc|c} \sum_j X_j A_{j,l}^1 & 0 & \sum_j X_j B_{j,l}^{12} \\ \sum_j X_j A_{j,l}^2 & \sum_j X_j B_{j,l}^{21} & \sum_j X_j B_{j,l}^{22} \\ \hline C_l^1 & D_l^{11} & D_l^{12} \\ \hline C_l^2 & D_l^{21} & D_l^{22} \end{array} \right)$$

and with $\Delta_l(\delta, \dot{\delta}) = \mathrm{diag}(\delta_1 I, ..., \delta_d I, \dot{\delta}_1 I, ..., \dot{\delta}_d I)$. This allows to apply the full-block S-procedure (Scherer, 1997): Solving (5) is equivalent to finding symmetric X_j and scalings Q_l, S_l, R_l such that, with

$$P_e = \left(\begin{array}{c|cc} P_0 & 0 & 0 \\ \hline 0 & Q_l & S_l \\ 0 & S_l^T & R_l \end{array} \right),$$

the following inequalities hold:

$$(*)^T P_e \left(\begin{array}{cc|c} \sum_j X_j A_{j,l}^1 & 0 & \sum_j X_j B_{j,l}^{12} \\ I & 0 & 0 \\ \hline \sum_j X_j A_{j,l}^1 & \sum_j X_j B_{j,l}^{21} & \sum_j X_j B_{j,l}^{22} \\ 0 & I & 0 \\ C_l^1 & D_l^{11} & D_l^{12} \\ \hline 0 & 0 & I \\ C_l^2 & D_l^{21} & D_l^{22} \end{array} \right) < 0 \qquad (7)$$

$$\forall \delta \in \boldsymbol{\delta}_l, \dot{\delta} \in \dot{\boldsymbol{\delta}} : \left(\begin{array}{c} * \\ * \end{array} \right) \left(\begin{array}{cc} Q_l & S_l \\ S_l^T & R_l \end{array} \right) \left(\begin{array}{c} \Delta(\delta, \dot{\delta}) \\ I \end{array} \right) > 0. \qquad (8)$$

This is, still, an infinite family of LMIs. By imposing extra constraints on P_l, one can reduce the infinite to a finite family of LMIs and, still, guarantee their validity over the whole parameter box. Let us recall three possibilities (Iwasaki and Hara, 1998; Dettori and Scherer, 1998; Scherer, 1996):

- Full block scalings that are indirectly described by LMIs.
- Block-diagonal scalings that can be explicitly parametrized (as appearing in standard SSV-theory) to satisfy (8).
- Fixed scalings as in standard small-gain arguments.

In concrete situation one has to take the following trade-off into account: Choosing a large class of scalings leads to an LMI system with many variables; then one expects to be able to work with a coarse grid of the parameter space what reduces the number of LMIs. Taking a smaller class of scalings reduces the number of variables; however, through the possible need for a finer parameter grid one might be forced to increase the number of LMIs.
Independent of which of the relaxation schemes is employed, we can always guarantee the success of the technique if the subdivision of the parameter set is sufficiently refined.

Theorem 4 *Suppose that the symmetric X_j and P_l solve (7)-(8). Then $X(.)$ as defined in (6) is a solution of (5).*
Consider any sequence of partitions for which (1) converges to zero. Suppose that for any such partitions the LMIs (7)-(8) have a solution. Then $X(.)$ as defined in (6) solves (3) if the mesh size (1) is sufficiently small.
Conversely, if (3) does have a solution, the LMIs (7)-(8) do have a solution if the mesh size (1) is sufficiently small.

Remarks 1. It is straightforward to formulate the corresponding result for minimizing the disturbance attenuation level γ.
Remarks 2. The first statement extends results that have been obtained for piecewise affine Lyapunov matrices and piecewise affine system descriptions: the feasibility of the LMI system leads to a solution of the analysis PDLMI (5) for LFT system descriptions. In the second part, we need to make sure that the LFT approximation of the system matrices is good enough to arrive at a solution of (3).
Remarks 3. As a novel aspect, the third statement reveals that our scheme is only successful if the original PDLMI (3) does have a solution. Hence, we have found a sufficient and *necessary* test for the solvability of the LMI (3) that is amenable to numerically reliable algorithms.
Remarks 4. The numerical effort decreases drastically if it is known that the parameters are constant. Then it suffices to work with piecewise constant Lyapunov matrices. Since the inequalities (7)-(8) are decoupled, they can be solved for each parameter box independently. Still, however, it is possible to exactly solve the corresponding robust performance problem that admits an alternative formulation as a real SSV-problem.

7 Example

We have implemented the proposed scheme (with block-diagonal scalings as in SSV-theory) in Matlab Version 5 using the LMI-Toolbox and the Spline-toolbox.
Let us consider the system defined with

$$\mathcal{F}_l\left(\left(\begin{array}{cc|ccc} -1 & 0 & 1 & 1 & 0 \\ 1 & -1 & 0 & 0 & -1 \\ \hline 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 \end{array}\right), \begin{pmatrix} \delta & 0 \\ 0 & \delta \end{pmatrix}\right)$$

for $\delta \in [-0.9, 0.9]$, $\dot{\delta} \in [-r, r]$, $r \in [0, 0.7]$. This system is not quadratically stable. We applied the proposed technique with splines of order three ($m_i = 2$) for the break-points $\{-0.9, -0.5, 0, 0.5, 0.9\}$.

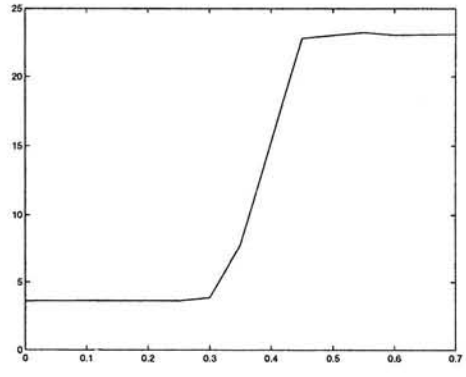


Fig. 1: Optimal γ for $r \in [0, 0.7]$.

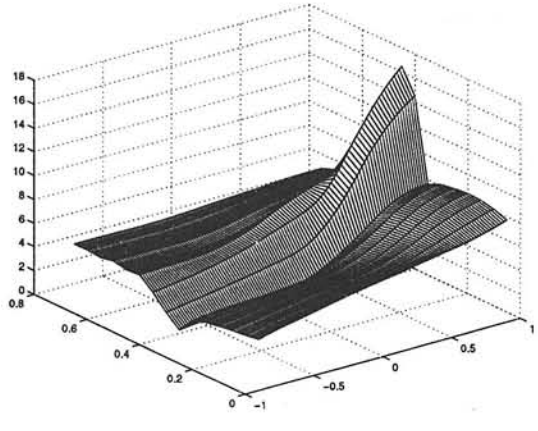


Fig. 2: Largest eigenvalue of $X(\delta)$ for $\delta \in [-0.9, 0.9]$ and $r \in [0, 0.7]$.

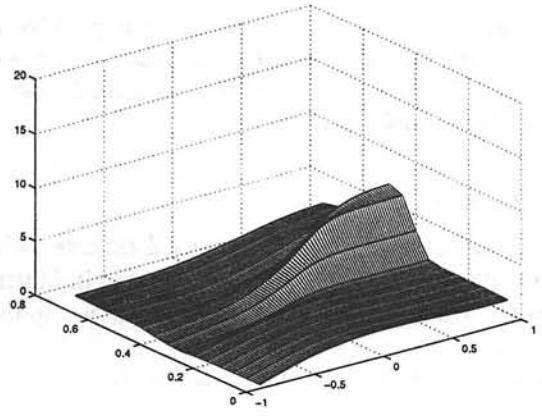


Fig. 3: Smallest eigenvalue of $X(\delta)$ for $\delta \in [-0.9, 0.9]$ and $r \in [0, 0.7]$.

In Figure 1, we plot the optimal attenuation level γ over the rate of variation. It is nice to observe how the attenuation level increases with an increasing bound on the rate. For $r = 0$, one can confirm the achievable attenuation level by applying the corresponding SSV test. Finally, Figures 2 and 3 show the largest and the smallest eigenvalue of the Lyapunov matrix what indicates how it needs to be adjusted to account for changes on the admissible rate of parameter variations.

8 Conclusions

We have proposed a numerical scheme that allows to test robust performance of systems that depend on time-varying rate-bounded parameters without conservatism. The scheme admits an immediate specialization to time-invariant parameters and, hence, provides an alternative to real SSV-techniques without the need for frequency gridding.

Among the many non-addressed questions, we intend to address in our further research the possibilities for an automatic refinement of the mesh and the closely related question of how to decide when to stop the algorithm.

References

Apkarian, P., P. Gahinet, and G. Becker (1994). Self-scheduled $\mathcal{H}_\infty$ control of linear parameter-varying systems. *Proc. Amer. Contr. Conf.*, pp. 856-860.

Feron, E., P. Apkarian, and P. Gahinet (1996). Analysis and synthesis of robust control systems via parameter-dependent Lyapunov functions. *IEEE Trans. Automat. Control, 41*, 1041-1046.

Becker, G., A. Packard, D. Philbrick, and G. Balas (1993). Control of parametrically-dependent linear systems: A single quadratic Lyapunov approach. *Proc. Amer. Contr. Conf.*, San Francisco, CA, pp. 2795-2799.

Becker, G. and A. Packard (1994). Robust performance of linear parametrically varying systems using parametrically dependent linear feedback. *Systems Control Lett., 23*, 205-215.

de Boor, C., K. Höllig, and S. Riemenschneider (1993). *Box Splines.* Springer-Verlag, New York.

Dettori, M. and C.W. Scherer (1998). Robust stability analysis for parameter dependent systems using full block S-procedure. *Proc. 37th IEEE Conf. Decision Contr.*, Tampa, FL.

Gahinet, P., A. Nemirovskii, A.J. Laub, and M. Chilali (1994). The LMI control toolbox. *Proc. 33rd IEEE Conf. Decision Contr.*, Lake Buena Vista, FL, pp. 2038-2041.

Helmersson, A. (1995). *Methods for Robust Gain-Scheduling.* Ph.D. Thesis, Linköping University, Sweden.

Iwasaki, T. and S. Hara (1998). Well-posedness of feedback systems: insights into exact robustness analysis and approximate computations. *IEEE Trans. Aut. Control, 43*, 619-630.

Johansson, M. and A. Rantzer (1997). On the computation of piecewise quadratic Lyapunov functions. *Proc. 36th IEEE Conf. Decision Contr.*, San Diego, CA, pp. 3515-3520.

Köse, I.E., F. Jabbari, and W.E. Schmittendorf (1996). A direct characterization of L_2-gain controllers for LPV systems. *Proc. 35th IEEE Conf. Decision Contr.*, Kobe, Japan, pp. 3990-3995.

Lim, S. and J.P. How (1997). Analysis of LPV systems using a piecewise affine parameter-dependent Lyapunov function. *Proc. 36th IEEE Conf. Decision Contr.*, San Diego, CA, pp. 978-983.

Packard, A. (1994). Gain scheduling via linear fractional transformations. *System and Control Letters, 22*, 79-92.

Scherer, C.W. (1996). Robust generalized H_2 control for uncertain and LPV systems with general scalings. *Proc. 35th IEEE Conf. Decision Contr.*, Kobe, Japan, pp. 3970-3975.

Scherer, C.W. (1995). Mixed H_2/H_∞ control. In: A. Isidori (Ed.), *Trends in Control, A European Perspective*, Springer-Verlag, Berlin, pp. 173-216.

Scherer, C.W. (1997). A full block S-procedure with applications. *Proc. 36th IEEE Conf. Decision Contr.*, San Diego, CA, pp. 2602-2607.

Schumaker, L.L. (1981). *Splines Functions: Basic Theory.* John Wiley & Sons, New York.

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Large scale dynamic-economic optimization of industrial processes: a modified sequential approach

R.L. Tousain[‡], J.J. van der Schot[‡], A.C.P.M. Backx[§] and O.H. Bosgra[‡]

[‡] *Mechanical Engineering Systems and Control Group*
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.
E-mail: R.L.Tousain@wbmt.tudelft.nl
[§] *Aspentech Europe b.v., De Waal 32, 5684 PH Best, The Netherlands.*

Abstract. This paper addresses economic operating optimality of chemical processes during transients: e.g. batch operation or grade-changes in continuous settings. A general, dynamic-economic problem formulation is set up, with the process dynamics described by a DAE model and the economic objective defined as the maximization of *added value*. The so-called *sequential approach*, which is often considered most convenient and robust for large-scale dynamic optimization, uses control parametrization to discretize the problem and solves successive quadratic approximations to converge to a local optimum. In many problems where realistic economic objectives are used, the nonlinearity of the objective function is expected to be more severe than the nonlinearity of the plant, and - which is worse - far from quadratic. In such cases, many iterations will be needed to converge, because the quadratic fit at each iteration varies greatly over the search space. Here, we present the early results of a new approach, *successive sequential quadratic programming*, which solves a nonlinear program every iteration, based on a linear approximation of the process dynamics and the real, nonlinear cost function. For the class of problems under consideration, such an approximation will be more accurate than a quadratic one, yielding faster progress towards the solution. The method is applied on a realistic grade change problem for a HDPE polymerization simulation model.

Keywords. Large scale dynamic optimization, economic optimization, control parametrization method, SSQP method, SQP method, HDPE polymerization.

1 Introduction

Economic optimization of the operation of industrial processes is believed to lead to significant increase in plant profitability. There has been a tradition in process industries to regard economic process optimization (determining where to drive the plant) seperately from process control (taking the actions required to reach that optimum operating range, while compensating for disturbances), e.g. (Figueroa et al., 1994; Amini et al., 1992; Bailey et al., 1993; Bandoni et al., 1994). This is reflected in the *layered* structure of the state-of-the art combined optimization and control system for (petro-) chemical plants, as indicated in Figure 1. Both layers have different goals and operate on different domains : the optimizer calculates steady state optimal operating conditions based on the plant economics, whilst the advanced control layer provides setpoint tracking and disturbance rejections, thus regarding plant dynamics. The underlying idea of steady-state optimization is that process dynamics are either unimportant, or just a nuisance impeding reaching the optimum operating conditions instantaneously. To reject the effect of slow disturbances and plant-model mismatch, the optimal setpoints are calculated and implemented recursively. The idea might come up that, if we are able to increase the optimization frequency (by using fast numerical optimizers and by reducing the

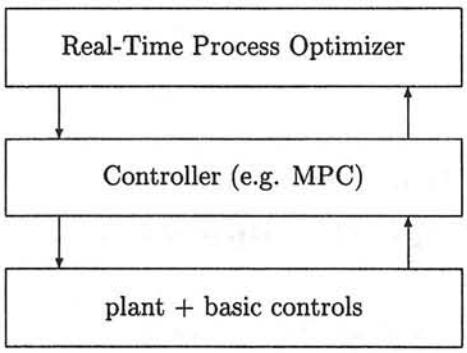


Fig. 1: Control/optimization hierarchy of state-of-the-art (petro-)chemical plants.

natural time-to-steady-state of the process through increased controller aggressiveness), then we can also optimally deal with transitions in the process behavior. Next to practical problems, however, this line of thought is fundamentally unsuited since the actual optimum operating conditions comprise both out- and inputs, which are linked by dynamic laws. A conceptually better viewpoint on optimal operation would be to take these process dynamics as relevant aspects of the overall plant economics. This means that we let go of the idea of "optimal operating point" and change it for "optimal operating trajectory". Instead of optimizing steady-state plant economics and merely *controlling* plant dynamics, we seek to economically optimize plant dynamics. Note that the applicability of such dynamic-economic operating strategies is not limited to those processes that are moved deliberately from one operating point to another: every process where money is lost due to the inability to optimally adjust the operating conditions according to disturbances is also a candidate. Shifting attention to dynamic optimality of the process not only has consequences for our own image of the process, but also for the technologies and strategies needed to come close to dynamic-economic optimal operation. The INCOOP[1] project aims at developing an integrated approach towards dynamic-economic optimal operation of chemical processes. Main ingredients in such an approach are techniques for large-scale, on-line dynamic optimization, state estimation and multivariable control. In this paper, we limit our attention to dynamic optimization in an off-line setting.

In literature, dynamic optimization is extensively addressed, as it is an important ingredient in many control strategies. In Nonlinear Model Predictive Control for example, a nonlinear optimization problems is solved every sample to calculate the controls (Biegler and Rawlings, 1992; Ali and Elnashaie, 1997; Sistu et al., 1993). For traditional reasons, the objective functions that are defined for such control-related dynamic optimization problems often involve quadratic penalties on tracking errors and input moves. We feel that for real-life problems linear or quadratic objective functions are often oversimplifications.

Two basic solution methods for large-scale dynamic optimization can be distinguished: the *sequential* approach and the *simultaneous* approach. The sequential approach utilizes parametrization of the controls to discretize the problem (Jang, 1987; Vassiliadis, 1993). An integration tool is used to evaluate the model equations and hence the objective function, the constraints and the gradients. Successive search directions towards the local optimum are determined by an outer loop optimizer, which is generally a Sequential Quadratic Programming tool (SQP) (Edgar and Himmelblau, 1988). In the simultaneous approach, both the controls and the states are parametrized (Li and Biegler, 1989) to transform, mostly via collocation on finite elements, the dynamic optimization problem into a Nonlinear Program which can be solved using a Nonlinear Programming tool (e.g. SQP or Generalized Reduced Gradient (GRG) (Edgar and Himmelblau, 1988)). The big advantage of the simultaneous approach is that the objective function and the process model equations converge simultaneously (infeasible path method), while the process model equations are necessarily satisfied in every iteration in the sequential approach. However, the sequential approach is simpler in implementation and especially for stiff systems, it may actually be an advantage instead of a disadvantage that the model equations are satisfied every iteration. All problems related to numerical integration are dealt with by the integration tool, whereas for the simultaneous approach these numerical problems interfere with the choice of parametrization and collocation intervals. For these reasons, in our research we focussed on the sequential approach.

An important aspect in dynamic optimization is the solution time. Minimizing solution time is especially relevant in on-line applications of optimization. Because the model integrations are most time-consuming, the solution time will be strongly related to the number of iterations needed to converge. This number can actually be quite low if the problem is approximately quadratic (a quadratic approximate program is solved every iteration to yield a search direction). For problems that are strongly ("non-quadratically") nonlinear, many iterations will be

[1] INCOOP is an abbreviation for "INtegration of COntrol and OPtimization".

needed and the solution time will be long. In this paper we present the early results of a new optimization method that deals specifically with economic objective functions (van der Schot, 1998). This method increases solution speed for the large class of problems where the nonlinearity of the cost function is more severe than that of the process model. The outline of the paper is as follows: in Section 2 the dynamic-economic optimization problem is defined. The conventional sequential approach to solving this problem is described in Section 3. The particular structure of a class of dynamic-economic problems is investigated in Section 4. This investigation leads to the introduction of our new approach (Successive SQP) in Section 5. The application of SSQP on a polymerization grade change case is described and evaluated in Section 6. Some discussion on the proposed algorithm is contained by Section 7. Finally, conclusions are given in Section 8.

2 Problem formulation

Off-line determination of optimal trajectories can be useful in batch processes and in the control of transients that are known well in advance. The translation of the real-life problem to a mathematical problem formulation comprises the definition of a *model* and an *objective function* (Edgar and Himmelblau, 1988). The construction of an adequate model[2] is a challenge in itself and an academic problem when it comes to the definition of general model adequacy requirements for dynamic optimization. The construction of a suitable objective function is often a case-specific task which requires knowledge of not only the process behavior but also of feedstock and consumer market situations (see e.g. (McAuley and MacGregor, 1992)). For reasons of generality and consistency we decided to have the objective indicated by a unified and unambiguous performance indicator: money. Assuming the prices of all material and energy flows to be known, we can construct a so-called "money conservation law" (e.g. (Westerterp et al., 1984) mentioned the "money balance" in a modeling framework) for the process:

$$\dot{\$}(t) = \$_a(t) + \$_{in}(t) - \$_{out}(t), \tag{1}$$

where \$ is the money holdup, $\$_a$ is the added value, and $\$_{in}$ and $\$_{out}$ are respectively the costs related to the physical input to the process and the revenues related to the physical output of the process. What we want to maximize is the *added value* over a certain time interval $[t_0, t_f]$:

$$\max \int_{t_0}^{t_f} \$_a dt = \$(t_f) - \$(t_0) - \int_{t_0}^{t_f} (\$_{in}(t) - \$_{out}(t))dt, \tag{2}$$

subject to a so-called "continuity constraint"

$$\$_L < \$(t_f) < \$_U, \tag{3}$$

where $\$_L$ is the minimum holdup that is required for continuity of operation and $\$_U$ is the maximum holdup that is tolerable. Constraint (3) may be redundant if other constraints with respect to production volume are already defined. The prices that are used to calculate the revenues will often depend on product quality, which is related to the operating conditions of the process. More generally, we introduce functions Φ and Ξ and a vector of process variables z such that:

$$\$_{out}(t) - \$_{in}(t) = \Phi(z(t)), \tag{4}$$
$$\$(t_f) = \Xi(z(t_f)), \tag{5}$$

z, which contains all variables that appear in the cost function or in the later to be mentioned physical constraints, will generally be a combination of some of the inputs u, some of the state variables x and some of the algebraic variables y, i.e.

$$z = C_x x + C_y y + Du, \tag{6}$$

with respectively state, algebraic variable and input selector matrices C_x, C_y and D. The input, state and algebraic variables are related through the process dynamics, which we assume to be described by a DAE:

$$\dot{x}(t) = f(x(t), u(t), y(t)), \tag{7}$$
$$0 = g(x(t), u(t), y(t)). \tag{8}$$

This model is included in the overall optimization as a set of constraints. We so obtain the general form of the (infinite dimensional) dynamic optimization problem:

$$\begin{aligned} \max_{u(t)} \; & V = \textstyle\int_{t_0}^{t_f} \Phi(z(t))dt + \Xi(z(t_f)), \\ \text{s.t.} \; & \dot{x}(t) = f(x(t), u(t), y(t)), \\ & 0 = g(x(t), u(t), y(t)), \\ & z(t) = C_x x(t) + C_y y(t) + Du(t), \\ & 0 \leq c(z(t)). \end{aligned} \tag{9}$$

c is a general constraint function that represents the physical constraints and also includes the continuity constraint (3). The choice of c involves an investigation of the *back-off*, i.e. control freedom, that needs to be taken into account. This issue has been dealt with in literature for the steady state case (Bandoni et al., 1994); for the dynamic case the concept of back-off is rather new and a nice research topic.

[2] Note that model *adequacy* in the framework of optimization is not necessarily related in a one-to-one fashion to model *accuracy*, (Forbes et al., 1994).

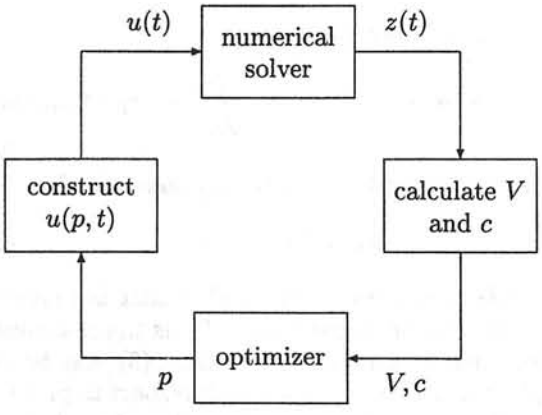


Fig. 2: Sequential solution strategy for large scale dynamic optimization.

3 Standard sequential approach, using SQP

The conventional sequential solution approach is based on the discretization of the controls: $u = u(p,t)$, introducing parameters p, to yield a finite dimensional problem. Common parametrization methods make use of splines, polynomials or wavelets. The solution strategy based on this parametrized problem, which is of sequential nature, is indicated in Figure 2 (Støren and Hertzberg, 1995).

An initial parameter vector is chosen. With the according input trajectories, the model is integrated using a numerical solver to obtain values for the objective function V, the constraints c and the gradients dV/dp and dc/dp. The gradients can be quite efficiently obtained by integrating, along with the model equations, the so-called *sensitivity equations* (Støren and Hertzberg, 1995). These can be derived by taking the derivatives of equations (7) and (8) with respect to the parameters p:

$$\frac{\partial}{\partial p}\left(\frac{\partial x}{\partial t}\right) = \frac{\partial f}{\partial x}\frac{\partial x}{\partial p} + \frac{\partial f}{\partial u}\frac{\partial u}{\partial p} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial p}, \quad (10)$$

$$0 = \frac{\partial g}{\partial x}\frac{\partial x}{\partial p} + \frac{\partial g}{\partial u}\frac{\partial u}{\partial p} + \frac{\partial g}{\partial y}\frac{\partial y}{\partial p}. \quad (11)$$

Substitution of (11) in (10) (under the condition that $\frac{\partial f}{\partial y}$ is nonsingular, i.e. in the case the DAE system is of index 1), and change of the order of differentiation yields:

$$\frac{\partial}{\partial t}\left(\frac{\partial x}{\partial p}\right) = A\frac{\partial x}{\partial p} + B\frac{\partial u}{\partial p}, \quad (12)$$

with

$$A = \left[\frac{\partial f}{\partial x} - \frac{\partial f}{\partial y}\left[\frac{\partial g}{\partial y}\right]^{-1}\frac{\partial g}{\partial x}\right], \quad (13)$$

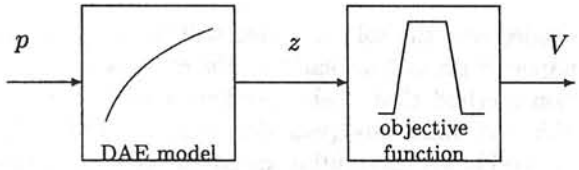


Fig. 3: Nonlinearity characteristic of the process model and the economic objective function.

$$B = \left[\frac{\partial f}{\partial u} - \frac{\partial f}{\partial y}\left[\frac{\partial g}{\partial y}\right]^{-1}\frac{\partial g}{\partial u}\right]. \quad (14)$$

The integration of the sensitivity equations can be done quite efficiently since the Jacobian matrices that are required can be taken from the integration of the model equations. Based on the state sensitivities, dV/dp and dc/dp are calculated in a straightforward manner. With this gradient information and an approximated Hessian, a Quadratic Program (QP) is set up and solved to yield a search direction, followed by a line search in that direction. Alternatively, the trust region (TR) approach to SQP sets up a QP within certain limits, i.e. it determines the step size a priori to be restrained to some surrounding of the current point in which we "trust" the quadratic model (Moré, 1983). The QP solution then yields a direction and step size which is implemented on the nonlinear model. If satisfactory improvement is obtained then the new point is adopted, otherwise the TR is reduced and a new QP solved.

4 Problem structure

The solution time of the SQP is strongly related to the number of integrations that needs to be performed (and thus the number of *iterations*). The number of iterations can actually be quite small if the problem is approximately quadratic. For strongly nonlinear[3] problems however, many iterations (and thus many costly integrations) will be needed. The dynamic-economic problem as we formulated it in Section 2 is in fact expected to be strongly nonlinear in many practical cases. The main reason for this is the relation between product quality and its price, which will be almost discontinuous in many problems. The nonlinearity of the process itself, on the other hand, is expected to be rather smooth and less severe. This is illustrated in Figure 3.

Hence, for the class of problems under consideration, the mapping from p to V comprises a smoothly nonlinear part $z(p)$ that is hard (or in terms of costs:

[3]The adjective "strongly" must be interpreted as "other than quadratic" here...

expensive) to evaluate and a strongly nonlinear part $V(z)$ that is easy and inexpensive to evaluate. It is easy to see that for this class of problems the application of the standard SQP approach is highly inefficient. Take for example the case where the model is Linear Time Varying and the cost function strongly nonlinear. If we would, unaware of the linearity of the model, apply the SQP method, then we would need many iterations (and as many integrations) to converge to the optimum, since the quadratic approximation of $V(p)$ will only be valid in a small region. However, one integration of the model's sensitivity equations provides all the information we need to characterize the process behavior; all other integrations are redundant and a waste of time. Based on this understanding of the structure of many dynamic-economic problems, we now introduce the Successive SQP approach that explicitly takes into account the nonlinearity of the cost function.

5 New approach using successive SQP (SSQP)

To speed up *sequential* solution of strongly nonlinear dynamic-economic optimization problems we need to find a way to increase the progress/iteration. Intuitively we can accomplish this by increasing the validity range of the approximate problems that we solve every iteration (recall that the conventional SQP method utilizes a quadratic approximation). In the SSQP method that we propose here, this is realized by construction of a nonlinear approximation, based on a linear approximation of the model and the fully nonlinear objective function. The approach comprises a 5 step-procedure, as described below (we adopt the trust region approach here for reasons that will be discussed in Section 7):

Step 1: initialization
l is set to 1 (first iteration). An initial trust region TR_p^1, and a parameter vector of length M, $p^1 = p_0$, is chosen to initialize the dynamic optimization.

Step 2: objective and constraint evaluation
The model equations are integrated with input $u^l(t) = u(p^l, t)$. The solutions are denoted $x^l(t)$ and $y^l(t)$. For reasons that will become clear shortly, the objective function is evaluated by calculating the Riemann sum:

$$V_d^l = V_d(\mathbf{z}^l) = \sum_{i=1}^{N} \Phi(z^l(iT))T + \Xi(z^l(t_f)), \quad (15)$$

where $\mathbf{z} = [z(T)^T z(2T)^T \dots z(NT)^T]^T$, $N = t_f/T$, and T is the integration time step that is used. The constraints c can be evaluated using interior points or using an end-point constraint $0 = \int_{t_0}^{t_f} min(c(z(t)), 0)dt$; since the constraint handling can be done exactly the same way as in the standard sequential approach, we do not treat this issue extensively here.

Step 3: gradient evaluation
The sensitivities of z^l with respect to p^l are evaluated along the trajectories $(x^l(t), u^l(t), y^l(t))$ at discrete times $T, 2T, \dots, NT$. The gradients can be stacked in a sensitivity matrix $\mathbf{S}_{\Delta p^l \to \Delta \mathbf{z}^l}$ to yield:

$$\begin{bmatrix} \Delta z^l(T) \\ \Delta z^l(2T) \\ \vdots \\ \Delta z^l(NT) \end{bmatrix} = \mathbf{S}_{\Delta p^l \to \Delta \mathbf{z}^l} \begin{bmatrix} \Delta p_1^l \\ \Delta p_2^l \\ \vdots \\ \Delta p_M^l \end{bmatrix}, \quad (16)$$

or, in short notation: $\Delta \mathbf{z}^l = \mathbf{S}_{\Delta p^l \to \Delta \mathbf{z}^l} \Delta p^l$. A practical and efficient way of constructing $\mathbf{S}_{\Delta p^l \to \Delta \mathbf{z}^l}$ is given in the Appendix. The choice of T contains two aspects: the relevancy of the different time scales for the economic performance of the plant, and the desired accuracy of the sensitivity matrix.

Step 4: determination of search step
A search step is calculated by solving the following NLP:

$$\begin{aligned} \min_{\Delta p^l} \ & \tilde{V}_d = V_d(\mathbf{z}^l + \mathbf{S}_{\Delta p^l \to \Delta \mathbf{z}^l} \Delta p^l), \\ \text{s.t.} \ & \text{linearized constraints} \\ & \Delta p^l \in TR_p^l, \end{aligned} \quad (17)$$

using for example SQP optimization. All Jacobians can be calculated analytically, so the coding of the optimization can be done efficiently. Instead of using linearized constraints, we can also include nonlinear interior point constraints c_d based on the original nonlinear constraints c:
$0 \le c_d(\mathbf{z}^l + \mathbf{S}_{\Delta p^l \to \Delta \mathbf{z}^l} \Delta p^l)$.

Step 5: evaluation of progress; adaptation of trust region
The new solution $p^\star = p^l + \Delta p^l$ is implemented on the nonlinear model as in to **step 2** and the resulting objective is denoted $V_d^\star$. The following rule base is used for the trust region adaptation:

IF $V_d^\star < V_d^l$,
  set $V_d^{l+1} = V_d^\star$, $p^{l+1} = p^\star$, calculate TR_p^{l+1}, and goto **step 3** $(l = l + 1)$,
ELSE
  reduce TR_p^l and goto **step 4** $(l = l)$.

The update of the Trust Region may involve both shape and size updates. Of course, the exact limits

and update rules are tuning parameters to the algorithm (see e.g. (Moré, 1983)).
The SSQP procedure and a comparison with the (trust-region-based) SQP method are summarized in Table 1.

6 Example on HDPE process

The proposed method has been tested on a medium-scale model of a fluidized bed high-density polyethylene (HDPE) reactor functionally similar to the problem studied by McAuley and MacGregor (1992). This sample process has been chosen because polyethylene manufacturers are faced with an increasing need to operate flexibly with respect to different quality grades of polymer, which are mainly characterized by the density and so-called melt-index of the product. This flexibility calls for frequent grade transitions from one product to the other, during which the plant produces off-spec material which can only be sold at a lower price than on-spec product (Figure 4). We have applied the methods developed to define and find the economically optimal grade transition between two given grades (A and B). Finding such trajectories can lead to significantly improved plant profitability.

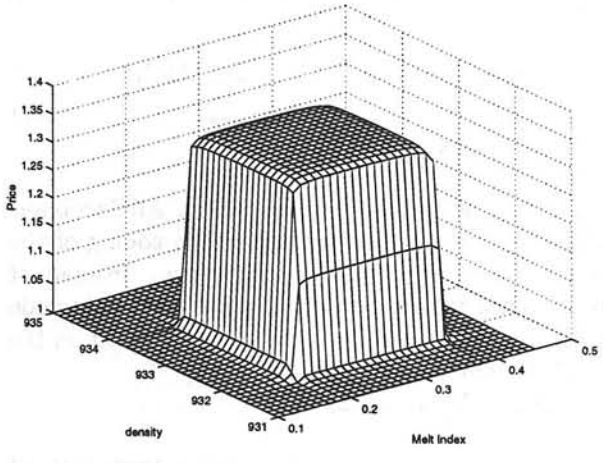


Fig. 4: Relation between quality (indicated by *melt-index* and *density*) and price for a certain grade of HDPE.

Test setup

The process model is a complex, stiff, nonlinear DAE system with 3800 variables (of which 100 states) implemented in SpeedUp. The optimization routine was developed in MATLAB, with a simple communication protocol for the transfer of data and commands. As with all control parameterization methods, there was no need to alter the existing process model, hence numerical integration problems were avoided. All solution times reported were obtained on a multiplatform setup of a 500 MHz Alpha station (simulation code) and a Dual Pentium 200 MHz machine (optimizer code) and exclude communication time. The manipulated variables chosen are those that most influence the product quality, and the inventory control: the feed ratio C_2/C_4[4], the feed ratio H_2/C_2, the purge, the N_2 feed and the bedlevel setpoint. These were parameterized by a total of 50 variables (10 each) using piece-wise linear functions. The inputs were allowed to vary from time 0 to 12 and held constant at the (supposed known) desired steady-state values from $t = 12$ to $t = 25$ in order to force the process to attain steady state within the interval concerned. This is similar to the use of control and prediction horizons in Model Predictive Control algorithms. The initial grade change trajectories were chosen as ramps, yielding a transition following the dashed line in Figure 5. The dotted lines represent the specifications on melt-index and density for both grades.

Solution

Maximizing the added value over a grade transition for this reactor poses a dynamic optimization problem that possesses the structure of Figure 3: a costly but relatively smooth process model and sharp edges in the objective function. Therefore, the SSQP method was used to solve the problem. The optimal trajectories were calculated for several market conditions (prices for grade A, grade B and off-spec product), each yielding a different grade-change policy. The solution time for one setting of the market conditions is summarized below, where the numbers correspond to the steps in Table 1.

(1)	$\mathbf{S}_{\Delta p^l \to \Delta z^l}$	546	CPUs (7 times dz/dp)
(2)	NLP sol.	194	CPUs (585 times $V(z)$ and 215 times dV/dz)
(3)	DAE integr.	667	CPUs (16 times $z(p)$)
	Total	1407	CPUs = 24 CPUmin.

The typical time needed for one optimization run was 25 CPU minutes. Of this time, more than 85% was spent on process model integration and sensitivity calculations, in line with previous research. This shows that integration of the process model is the main bottleneck in optimization and any attempt to reduce the number of model integrations needed is likely to reduce total optimization time. As we can see, the number of process simulations needed

[4] implemented on the model using a ratio-controller.

Table 1: Comparison between SQP and SSQP in the "trust region" setting

SQP		SSQP	
1.	linearize $z(p)$	1.	linearize $z(p)$
	linearize $V(z)$		do not approximate $V(z)$
	update Hessian of $V(z(p))$		
2.	solve QP within TR $V(z(p))$	2.	solve NLP within TR $z(p)$
3.	evaluate V on nonlinear model	3.	evaluate V on nonlinear model
	V better : adopt new & goto 1		V better : adopt new & goto 1
	V worse : reduce TR & goto 2		V worse : reduce TR & goto 2

is surprisingly low for SSQP, only 16 time-expensive simulations! This should be seen in contrast to the number of inner loop iterations of 585, which indicates the nonlinearity of the economic objective.

Results

The results on the HDPE process show improved profitability of plant operation during a grade transition. The optimizer changed the input profiles to such an extent that both the off-spec time and the off-spec volume were reduced, which yielded an increase in added value (profit + net holdup) over the fixed interval of 25 hours. In Figure 5 the optimal trajectories for melt-index, density and the production flow are plotted in solid lines for the situation where off-spec production is evaluated at the feed price and grade A resp. grade B is worth DM 1.35 and DM 1.45. We see that the optimizer succeeds in bringing the production from A to B in a short time, whilst the production flow is somewhat attenuated during the changeover. Apparently, the space within the grade specifications is optimally exploited to perform the grade change at maximum efficiency. More details on the example can be found in (van der Schot, 1998).

7 Discussion

We motivated the development and application of the SSQP method by stating that the nonlinearity of economic cost functions will in many problems be non-quadratic and more severe than that of the process model. We did not give a mathematical proof for this. Nor did we provide a feasible way to check this. At the moment we do not have such a check available in a suitable form. We may however, look at the Taylor expansion of the nonlinear mapping $p \rightarrow V$ to get a hold of some heuristics. The Hessian in the Taylor expansion is given by:

$$\frac{\partial^2 V}{\partial p^2} = \frac{\partial^2 V}{\partial z^2}\left(\frac{\partial z}{\partial p}\right)^2 + \underbrace{\frac{\partial V}{\partial z}\frac{\partial^2 z}{\partial p^2}}_{2^{nd}\text{order dynamics}}. \qquad (18)$$

In the SQP method, this Hessian is approximated from successive gradients, using for example a BFGS update (Edgar and Himmelblau, 1988). In the SSQP method, only the part of the Hessian related to the second order derivative of the cost function is present. Thus, a basic assumption underlying the SSQP approach, which allows us to discard the second order dynamics term of $z(p)$ from the second order term of $V(p)$ is that the contribution of $\frac{\partial V}{\partial z}\frac{\partial^2 z}{\partial p^2}$ is an order of magnitude smaller than that of $\frac{\partial^2 V}{\partial z^2}\left(\frac{\partial z}{\partial p}\right)^2$. If this is not the case, the SSQP method may actually yield worse progress than the SQP method. On the other hand, all higher order derivatives of the cost function also appear in the Taylor expansion, since we do *not* approximate the cost function, which will favor the SSQP approach in those situations where nonlinearity of the process is less severe.

Our choice for the trust region-based determination of a search step instead of the more common line search-approach, needs some motivation. Although we might just as well use a line search to confront the search directions with the real nonlinear process behavior, we feel that the trust region approach is more powerful in the SSQP setting. Note that the solutions resulting from the inner loop NLP will provide proper search directions (and will actually also be good "search steps") if they are located within the region where the inner loop approximation is expected to be accurate (i.e. in the so-called "Trust Region".). If this is not the case, the quality of the search directions/steps may actually be rather poor and slow convergence may result. A second advantage of the Trust Region approach is that not only the search *step* is altered after an unsuccesful implementation of a new solution, but - at the relatively

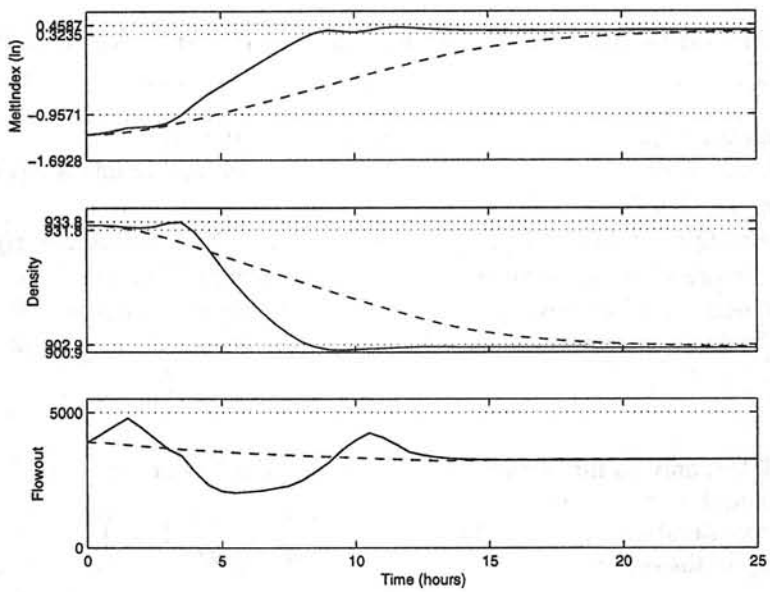


Fig. 5: Grade change under normal market conditions for loose MI specifications (dotted-lines: specifications, dashed-lines: initial trajectories, solid-lines: optimal trajectories).

low cost of 1 extra solution of the inner loop NLP - a new, and thus more suitable, search *direction* is also calculated.

8 Conclusions

The INCOOP project aims at an integration of process control and plant-wide optimization. In this paper we dealt with the problem of dynamic-economic optimization in an off-line setting. We defined a dynamic-economic objective as to maximize the added value of the plant over some time interval. Mathematically, this comes down to solving an optimization problem with algebraic-differential constraints. Based on an investigation of the sequential approach to solving such problems and its inefficiency for strongly nonlinear objective functions, we motivated the introduction of a new approach, successive sequential quadratic programming. The proposed method solves an NLP - constructed from a linearized process model and an exact (non-approximated) objective function - instead of a QP in the inner loop of a dynamic optimization problem. This yields much more accurate steps in the outer loop, hence greatly reducing the number of nonlinear model integrations needed. The extra inner loop iterations are inexpensive, thus total optimization time can be reduced by an order of magnitude for a large class of problems. The optimization approach was tested succesfully on a HDPE polymerization grade change problem, with the process described by a complex, stiff, nonlinear DAE system with 3800 variables implemented in SpeedUp.

References

Amini, Z., J. Lin, and P.D. Roberts (1992). Single iterative algorithm with global feedback for integrated system optimisation and parameter estimation of large scale industrial processes: optimality, convergence and simulation. *IEEE proceedings-D, 139*, 447-459.

Ali, E.E. and S.S.E. Elnashaie (1997). Nonlinear model predictive control of industrial type IV fluid catalytic cracking (FCC) units for maximum gasoline yield. *Ind. Eng. Chem. Res., 36*, 389-398.

Bailey, J.K., A.N. Hrymak, S.S. Treiber and R.B. Hawkins (1993). Nonlinear optimization of a hydrocracker fractionation plant. *Computers Chem. Engin., 17*, 123-138.

Bandoni, J.A., J.A. Romagnoli and G.W. Barton (1994). On optimising control and the effect of disturbances: calculation of the open-loop backoffs. *Computers Chem. Engin., 18*, Suppl. S505-S509.

Biegler, L.T. and J.B. Rawlings (1992). Optimization approaches to nonlinear model predictive control. In Y. Arkun and W.H. Ray (Eds.), *Chemical Process Control - CPC IV, Proc. 4th Engin. Foundations Conf.*, South Padre Island, TX, USA, pp. 543-571.

Edgar, T.F. and D.M. Himmelblau (1988). *Op-*

timization of Chemical Processes. McGraw-Hill Book Company.

Figueroa, J.L., P. Bahri, G.W. Barton and J.A. Romagnoli (1994). Economic assessment of alternatives in optimizing control. *Institution of Chemical Engineers Symposium Series*, n. 133, 207-212.

Forbes, J.F., T.E. Marlin and J.F. MacGregor (1994), Model adequacy requirements for optimizing plant operations. *Computers Chem. Engin., 18*, 497-510.

Jang, S. (1987). Control of constrained multivariable nonlinear process using a two-phase approach. *Ind. Eng. Chem. Res., 26*, 2106-2114.

Lee, J.H. and N.L. Ricker (1994). Extended Kalman filter based nonlinear model predictive control. *Ind. Eng. Chem. Res., 33*, 1530-1541.

Li, W.C. and L.T. Biegler (1989). Multistep, Newton-type control strategies for constrained nonlinear processes. *Chem. Engng. Res. Dev., 67*, 562-577.

Li, W.C., L.T. Biegler, C.G. Economou and M. Morari (1990). A constrained pseudo-Newton control strategy for nonlinear systems. *Computers Chem. Engin., 14*, 451-468.

McAuley, K.B. and J.F. MacGregor (1992). Optimal grade transitions in a gas phase polyethylene reactor. *AIChE Journal, 38*, 1564-1576.

Moré, J.J. (1983). Recent developments in algorithms and software for trust region methods. In A. Bachem, M. Grötschel and B. Korte (Eds.), *Mathematical Programming: the State of the Art.* Springer Verlag, Berlin, pp. 258-287.

Schot, J.J. van der (1998). *Practical Approach to Dynamic-Economic Optimization and Control of Industrial Processes.* M.Sc.-Thesis, Mechan. Engin. Systems & Control Group, Delft University of Technology.

Storen, S. and T. Hertzberg (1994). The sequential linear quadratic programming algorithm for solving dynamic optimization problems - a review. *Computers Chem. Engin., 19*, 495-500.

Sistu, P.B., R.S. Gopinath and B.W. Bequette (1993). Computational issues in nonlinear predictive control. *Computers Chem. Engin., 17*, 361-366.

Vassiliadis, V. (1993). *Computational Solution of Dynamic Optimization Problems with General Differential-Algebraic Constraints.* Ph.D.-Thesis, Imperial College of Science, UK.

Westerterp, K.R., W.P.M. van Swaaij and A.A.C.M. Beenackers (1984). *Chemical Reactor Design and Operation.* Wiley, New York.

Appendix: construction of $\mathbf{S}_{\Delta \mathbf{p} \rightarrow \Delta \mathbf{z}}$

An efficient and convenient way of obtaining the mapping dz/dp is by using a high order fixed step size integration algorithm to solve the sensitivity equations. Even if the Jacobians are not available from the model integration this would require only few evaluations of the Jacobians along the nominal trajectories. Most simulation tools provide this option in the form of a linearization algorithm, which does nothing more than the evaluation of the Jacobians by numerical perturbation. In SpeedUp the linearization option is called "CDI" (Control Design Interface). In Mathworks' Simulink, the linearization tool is called "linmod". The result is generally presented in state space form:

$$\Delta \dot{x}(t) = A_k \Delta x(t) + B_k \Delta u(t), \qquad \text{(A.1)}$$

where A and B are given by respectively (13) and (14). Having at our disposal these Jacobians for $t = T, 2T, \ldots, NT$ we can integrate the sensitivity equations to obtain

$$\Delta x((k+1)T) = \Phi_k \Delta x(kT) + \Gamma_k \Delta u(kT),$$

$k = 0, \ldots, N-1$, where $\Phi_k = e^{A_k T}$ and $\Gamma_k = \int_{kT}^{(k+1)T} e^{A_k \tau} B_k \, d\tau$ (Lee and Ricker, 1994). Note that applying this high order integration comes down to the discretization of the state space system (A.1), zero-order-holding u. For convenience and ease of notation, we assume that the economically relevant variable z are related to x and u only: $\Delta z = C_x \Delta x + D \Delta u$. Following (Lee and Ricker, 1994) we can construct the I/O matrix $\mathbf{T}_u$ by integrating the process model:

$$\begin{bmatrix} \Delta z(T) \\ \Delta z(2T) \\ \vdots \\ \Delta z(NT) \end{bmatrix} =$$

$$\begin{pmatrix} D_0 & 0 & & & \\ C\Gamma_0 & D_1 & 0 & & \\ C\Phi_1\Gamma_0 & C\Gamma_1 & D_2 & 0 & \\ C\Phi_2\Phi_1\Gamma_0 & C\Phi_2\Gamma_1 & C\Gamma_2 & D_3 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix} \begin{bmatrix} \Delta u(T) \\ \Delta u(2T) \\ \vdots \\ \Delta u(NT) \end{bmatrix}.$$

Introducing $\mathbf{U}_p$ as the parametrization-dependent mapping from p to u,

$$\begin{bmatrix} \Delta u(T) \\ \Delta u(2T) \\ \vdots \\ \Delta u(NT) \end{bmatrix} = \mathbf{U}_p \Delta p,$$

we obtain $\mathbf{S}_{\Delta p \rightarrow \Delta \mathbf{z}} = \mathbf{T}_u \mathbf{U}_p$.

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Multiple level control of a hydraulically driven flight simulator motion system[‡]

S.H. Koekebakker, P.C. Teerhuis and A.J.J. van der Weiden

Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.
E-mail: s.h.koekebakker@wbmt.tudelft.nl

Abstract. The Simona institute does research into flight simulator motion techniques. Core of the institute will be the Simona Research Simulator of which the motion system has been constructed. With flight simulator motion systems one wants to provide the pilot in the simulator with appropriate generalized specific forces. This has to be attained by steering six parallel hydraulic servo actuators. To deal with the complex control requirements, a fast multi-processor board was connected to the system. A multi level control structure was developed in which for each level specific tasks were defined in close relation with the other levels. The control structure consists of the following. With fast inner loop pressure control the hydraulic actuators are turned into force generators. Acceleration references can be provided to a feedback linearising controller which accounts for the non-linear mechanics. References consist of both stabilizing corrective accelerations by outer loop feedback of the reconstructed platform pose as desired accelerations provided by a smoothly interpolating reference model based feed forward. In this way an extensive model based controller could be implemented on the motion system. A benchmark test is being developed which has to quantify the level of improved performance.

Keywords. Robotics, multivariable nonlinear systems, mathematical model based control, implementation, aeronautics.

1 Introduction

The Simona flight simulator motion system, see Fig. 1., is a hydraulically driven mechanical system with six degrees of freedom and shows relevant non-linear fast and slow dynamics. It forms a representative test bed to do research into motion control. The Simona institute (Advani et al., 1997) tries to enhance the general standard of various aspects of flight simulator technology and as such the motion controller has to serve both as a research object and as a safe and high performance part of experiments e.g. into human perception.

The eventual task of high performance simulator motion control is improved motion realism. With flight simulator motion systems one wants to provide the pilot in the simulator with appropriate generalized specific forces i.e. both rotational and translational accelerations plus gravity (Martin, 1995). Particularly in case of tight pilot control, e.g. landing, phase lag of realized versus desired accelerations should be minimal. In this way smaller differences between simulator and in-the-air flight conditions will exist.

By construction design for control (low mass, centre of gravity, stiff synergetic structure) an important basis for enhanced performance has been attained. This has to be fully exploited by advanced controller design techniques.

With the hardware, fast multi processor dsp-boards, and software, automatic dsp-code generation from higher level simulation programs, used in this

[‡]The original version of this paper was presented at the 2[nd] *IEEE Systems, Man and Cybernetics IMACS Conference CESA'98*, April 1-4, 1998, Hammammet, Tunisia.

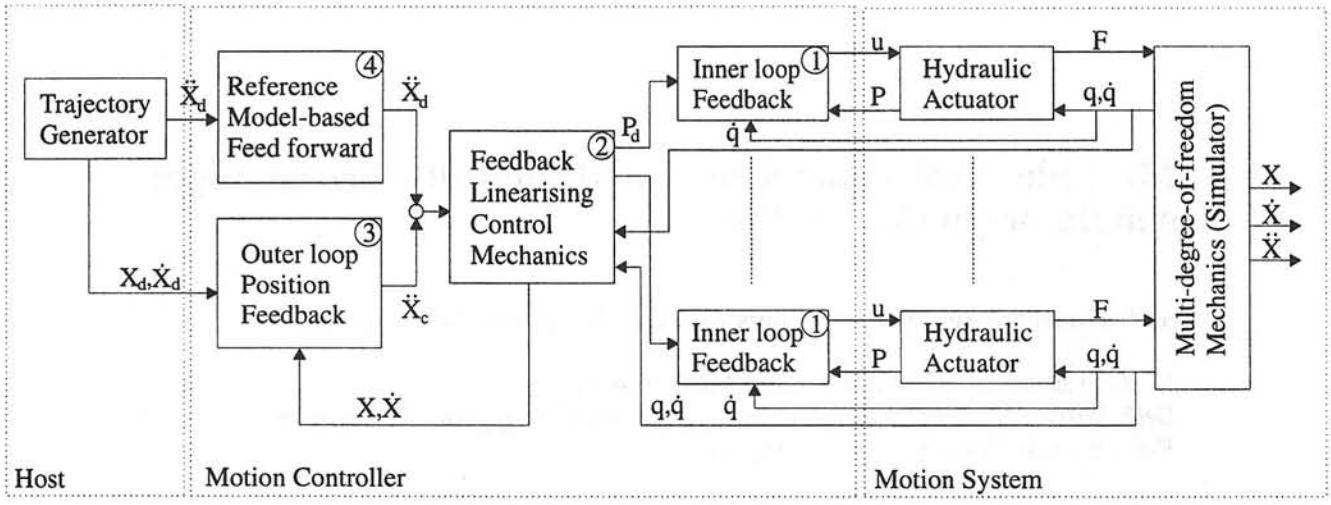


Fig. 2: Multi level control of the Simona Motion System

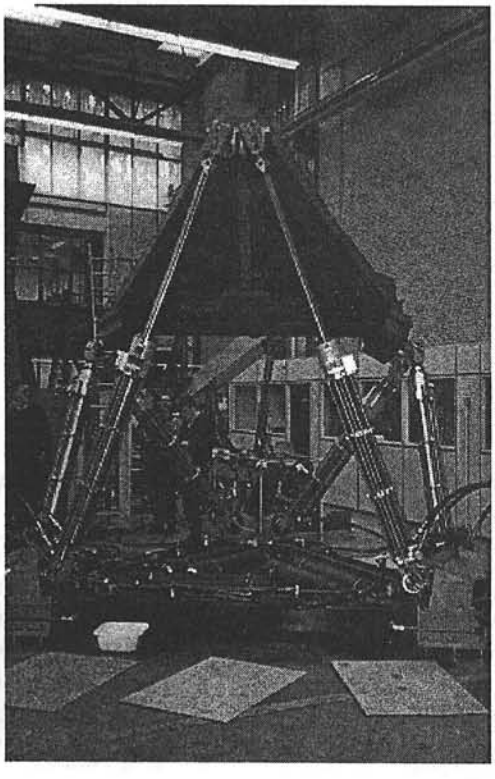


Fig. 1: The motion system of the SRS with an temporary dummy platform load in the central workshop of mechanical engineering.

project, it is possible to iterate fast on a controller design method of setting the specifications, analyzing the system, synthesizing the controller and implementing on the actual experimental set up. Complex model based controllers can be implemented on the motion system.

2 Problem Description

Since the motion controller to be designed has to deal with a complex system, extensive modelling took place with the control objective in mind, even before construction (Koekebakker et al., 1996; Van Schothorst, 1997). In the earlier stages measurements could be done on an experimental set up in which each hydraulic actuator was tested separately. At this point tests can be performed with a dummy platform replacing the eventual simulator on top of the motion system (Koekebakker et al., 1998c).
By analysis of both theoretic models derived from basic physical laws as experimental models based on measurements taken, an inventory of the relevant control problems was put together.

- **Control objectives**
  Flight simulation or fooling a pilots motion awareness basically forms a control problem with mixed objectives. The system should provide for the accelerations being simulated without running out of stroke. This problem is mainly left to a host which has to come up with feasible trajectories but the motion controller still has to both track reference accelerations as to stabilize platform pose.

- **Hydraulics.**
  Control of the long-stroke hydraulic actuators used in flight simulator motion systems is not

easy since the phase lag introduced by the servo valve together with the non-negligible high frequent transmission line resonances form a stability problem (Van Schothorst et al., 1994).

- **Hydraulics/Mechanics**
  Further the bilateral coupling of the hydraulic mechanical system with strong energy exchange via pressure/flow and force/velocity introduces mechanical pose dependent resonances with interaction over the actuators. The dynamics resulting from this interaction forms the most relevant part of the system as shown in Koekebakker et al. (1998c).

- **Mechanics**
  With the actuators mounted in parallel to the simulator, the construction forms a so-called Stewart platform. Due to the resulting kinematic loops, care should be taken to model this system with only explicit differential equations (Koekebakker et al., 1996). This requires modelling in appropriate coordinates i.e. the platform pose. As only the actuators lengths are being measured, a transformation to platform pose is required to be able to apply model based control. This transformation is, dual to the actuator trajectory generation of serial robots, explicitly known from platform pose to actuator length but not injective.

- **Trajectory generation**
  The reference acceleration and pose will be calculated at a low sample rate host computer which incorporates a complex airplane model. References have to be introduced smoothly to the motion controller. But smoothing the signal should not result in responses with too much phase lag since the timing of on-set motion is an important part of simulation quality.

This paper will describe a control strategy which takes into account the afore mentioned problems and resulted in a controller which could be implemented on a real-time control computer connected to the motion system.

3 Control Strategy

By design of a control structure which has four levels, each of which have their own specifications in close relation with the other levels, one can circumvent having to solve a too complex set of problems at once. Further it is shown that this structure leads to an implementable controller.

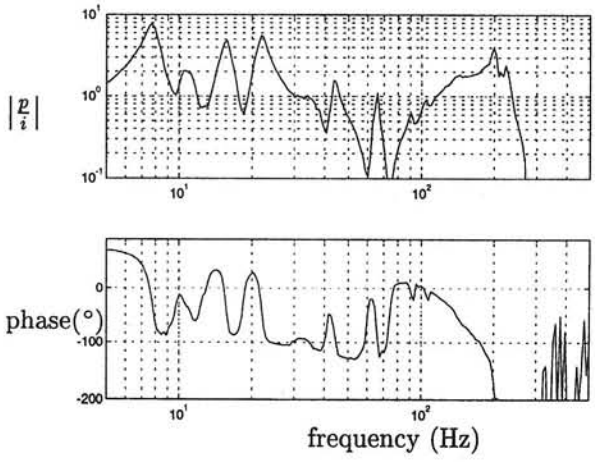


Fig. 3: Typical systems frequency response, bodeplot of (1,1)-element valve steering voltage to actuator pressure.

General idea

Looking at Fig. 2. in which the control structure is schematically depicted we consider the following control levels with reference to the applied control theory.

Level 1. Local hydraulic pressure control loops (Heintze and Van der Weiden, 1995; Van Schothorst, 1997; Sepehri et al., 1990).

Level 2. Multivariable feedback linearisation (Koekebakker, 1998b; Slotine et al., 1991).

Level 3. Outer loop position stabilisation (Qu and Dawson, 1996).

Level 4. Reference model based control (Koekebakker et al., 1998a; Tomizuka, 1993).

In short, the actuators are turned into pressure generators by local controllers. These controllers receive their reference pressure from a feedback linearisation loop in which pressures can be calculated necessary to track desired accelerations. Desired accelerations are partly corrections which are required to stabilise the pose of the simulator and for most the cues generated to provide the pilot with reasonable motion awareness. As these cues have to be smoothed but not delayed, a reference model based controller has to calculate appropriate cues for the feedback linearisation controller.
The next sections will describe the different control levels more closely.

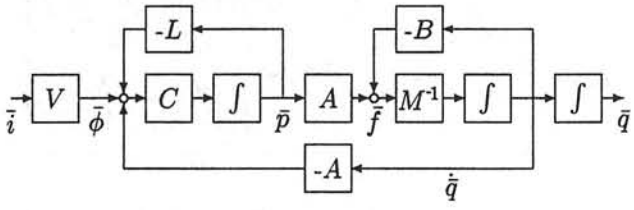


Fig. 4: Basic structure hydraulically driven motion system.

Inner loop pressure control

To turn the hydraulic actuators into nice force generators two of the afore mentioned control problems have to be solved at this level. Feedback of the pressure can result in stability problems since the relatively long transmission lines cause badly damped resonances together with phase lag of the valve. Further the coupling between the mechanics and hydraulics results in the pose and load dependent rigid modes of the system. Looking at the frequency response of the system at its neutral pose in Fig. 3., the rigid modes can be observed in the frequency area between 7 Hz and 25 Hz. At 200 Hz the transmission lines cause peaking (at 75 Hz also a notch results). The valve has a bandwidth of 150 Hz which can clearly be observed by looking at the phase. Flexibility in the mechanics caused some additional parasitic modes between 40 Hz and 80 Hz.

The coupling from which the rigid modes result, can be dealt with using the control method introduced by Sepehri et al. (1990) and successfully applied by Heintze and Van der Weiden (1995). A hydraulically driven motion system basically has the structure given in Fig.4. Through the valves, V, the oil flows, $\bar{\phi}$, can be steered by the inputs, $\bar{i}$. The required oil flows are mainly determined by the speed at which the volumes in the actuators have to be filled. These are equal to the velocities, $\dot{\bar{q}}$, of the actuators times the area of the piston, A. Together with the oil loss due to the leakage L, the net oil flow difference cause the pressures $\bar{p}$ to rise through the hydraulic oil stiffness, C.

$$\dot{\bar{p}} = C(V\bar{i} - L\bar{p} - A\dot{\bar{q}})$$

The acceleration of the actuators is determined by the inverse pose dependent mass matrix, M^{-1}, which causes the interaction, times the forces supplied by the actuators minus the viscous friction along the actuators due to the hydraulic bearings,

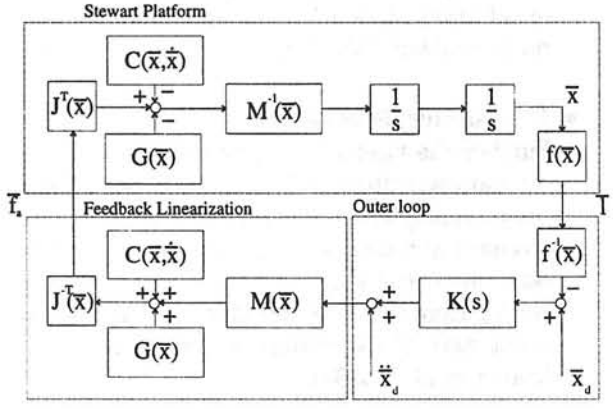


Fig. 5: Modified feedback linearising control structure

B.

$$M\ddot{\bar{q}} = A\bar{p} - B\dot{\bar{q}}$$

Gravity forces and the less relevant coriolis and centripetal forces are assumed to be dealt with at the higher levels.

By compensation of the oil flow due to actuator velocity, the hydraulics can be decoupled from the mechanics. A smooth 50 Hz bandwidth pressure generator can be obtained by filtering the pressure feedback signal properly. As the inner loop controller does not interact with the mechanics it could be designed and tested with the one degree of freedom experimental set up (Van Schothorst, 1997). In this way the hydraulic servo actuators which usually are considered velocity engines, are turned into force generators.

Multivariable feedback linearisation

The force generators can now be used to control the non-linear and multivariable mechanics. With model based calculation of the required forces to accelerate along the desired path, $\ddot{X}_d$, given a measured pose and velocity, the system is provided with both feed forward and decoupled feedback linearised correction paths to be used by the higher level controllers.

The proposed control structure is given in Fig.5. This structure differs from the standard computed torque controller of a mechanical system. In modelling for control the parallel Stewart platform configuration, one has to take care of generating a explicit set of differential equations (Koekebakker et al., 1996). Since this is only possible taking the platform pose as the generalised coordinates, the controller has to incorporate an algorithm which cal-

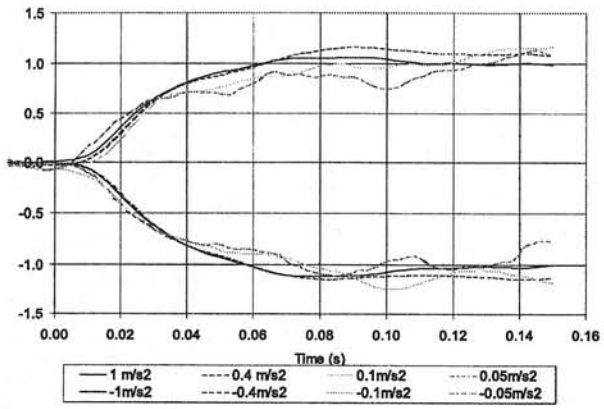


Fig. 6: Acceleration step response of the platform in surge direction at different (normalised) amplitudes controlled by the multiple level controller.

culates these coordinates from the measured actuator lengths, $\bar{l}$, and translates desired platform forces into required actuator pressures. In Koekebakker (1998b) the structure of this part of the controller will be presented in more detail.
The actuator lengths, $\bar{l}$ can be calculated from the platform pose $\bar{x}$.

$$\bar{l} = \bar{f}(\bar{x})$$

Measuring the actuator lengths, the platform pose has to be reconstructed iteratively.

$$\bar{x}_{k+1} = \bar{x}_k + J(\bar{x}_k)^{-1}(\bar{l}_k - \bar{l})$$

With the jacobian, J, defined by

$$J(\bar{x}) = \frac{\partial \bar{l}}{\partial \bar{x}}$$

In Koekebakker (1998a) it is shown that this iteration converges within the accuracy of the length measurements after 2 iterations in every part of the workspace going through the iteration at 1 kHz. This update frequency is attained.
The desired actuator pressures can be constructed by calculation of the platform mass matrix M, coriolis and centripetal forces C and gravity G. These are functions of the reconstructed platform pose.

$$\bar{P}_d = A^{-1}J^{-1}(M(\ddot{\bar{x}}_d + \ddot{\bar{x}}_c) + C + G)$$

Although careful identification of the model parameters should improve this controller, filling in design values for these parameters like masses, inertias and centres-of-gravity already results in reasonable performance by decoupling interaction up till 80%.

Outer loop control

The outer loop controller will have to stabilise the simulator pose to prevent the actuators from running out-of-stroke. As the feedback linearising controller decouples the mechanics into separate double integrators, the outer loop can generate correction accelerations resulting from a filtered PD-structure to stabilize these integrators. The correction accelerations, $\ddot{x}_c$, should not exceed human sensory thresholds (Hosman and Van der Vaart, 1980) i.e. generate no noticeable false cues. Therefore the correction should be sufficiently smooth (filtered) and only requires limited bandwidth (well below 1 Hz).
Although a P(I)D-structure of the outer loop controller robustifies the system (Qu and Dawson, 1996), explicit robust control will have to be used to more accurately deal with the varying system conditions one encounters working within a real-time environment.

Reference model based control

The motion control computer is provided with desired simulator accelerations ($\ddot{X}_d$) and poses (X_d) by a host which controls the over all simulation (including visual, instrumental and acoustic stimuli). These signals are generated by a model of the vehicle to be simulated and a subsystem called wash-out filters which translate vehicle motion into feasible simulator motion. Although the host comes up with new set points at relative low update frequency (ca. 60 Hz) the fact that the system being simulated are known, will enable a reasonable prediction of the next set point.
The reference model based control has the task to deal with the set points and future predictions in a proper way. Using knowledge of the set points supplied (mainly the fact that the signals do not contain information at frequencies higher than 30 Hz) a smooth interpolation filter provides a suitable reference acceleration to the feedback linearisation level together with a smooth jerk (derivative acceleration) signal which can be used as lead signal in the same feed forward channel.
The reference model based control shows considerable improvement w.r.t. phase lag or delay in simulating on-set of abrupt (e.g. landing bump) and fast varying motion (e.g. turbulence) (Koekebakker et al., 1998a).

4 Implementational issues

The multiple level controller has been implemented on a real-time multi-processor dsp motion computer (dSpace, 1996) connected to the motion system

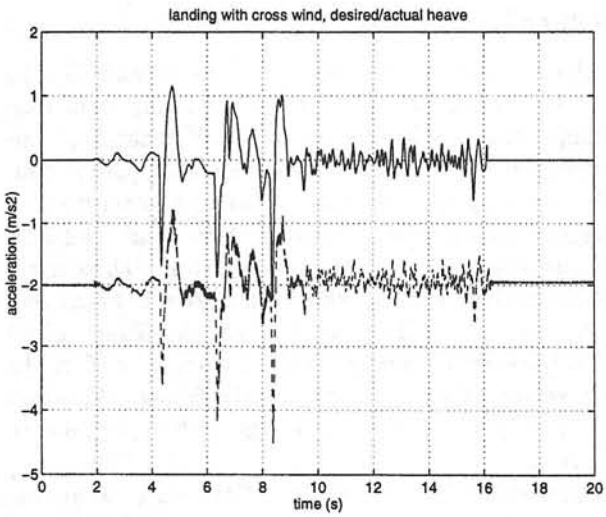


Fig. 7: Example simulator critical manoeuvre: landing with cross wind, desired and measured (biased, dashed) acceleration in heave

with a temporary dummy platform (ca. 2.5 tons). In this set-up one C40-processor has to perform all communication with the outside world (bottleneck w.r.t. sampling frequency) and could be run at 5 kHz. Also the coprocessor which calculates the inner loop control runs at 5 kHz necessary to deal with the relevant fast actuator dynamics.

In this respect the multiple level structure pays-off since the other levels, especially the feedback linearising control run on yet another coprocessor at 1 kHz which is just sufficient to go through all the algorithms involved.

Design of the control structure was performed in the user friendly environment of Matlab/Simulink[1] from which c-code can be generated automatically and connected with user-written code. In this way rapid prototyping of complex controllers as presented in this paper becomes feasible. Going from a Simulink model to a controller running in a real time environment takes about 10 minutes.

In Fig. 6. the platform coordinates are plotted during a acceleration step in the most difficult direction of surge (forward motion). The platform is moved fast and smooth at its desired acceleration at amplitudes ranging from 0.05 m/s^2 to 1 m/s^2. At amplitudes of 0.005 g, some noise can be observed. Plots had to be shifted to let $t = 0$ correspond and scaled to normalised desired value of 1.

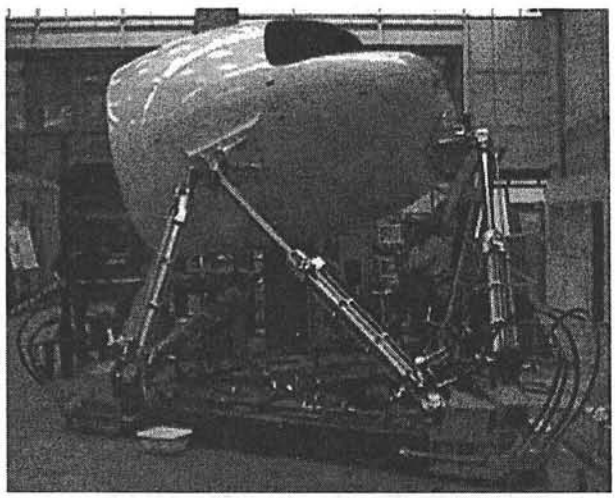


Fig. 8: Testing Simona research simulator motion system with shuttle

5 Performance quantification

Performance of the motion system was defined by the degree of motion realism attained. There are no measures known which exactly quantify this. More research into human perception has to point out how this has to be done. The Simona Research Simulator could play a role in attaining this goal.

At this moment a test is being developed which should enable some quantification. Part of this test will be a set of benchmark manoeuvres which are considered as simulator critical. At the simulator facility of the Royal Dutch Airline, 30 manoeuvres such as hard landing, response to maximum clear air turbulence, brake release during take-off roll, etc., were recorded by the aerospace group of the Simona institute. In Fig. 7 the heave accleration of the motion system carrying a dummy platform with the expected full operational payload of 4000 kg is depicted in performing a landing manoeuvre with cross wind. The three landing bumps and high frequency taxiing rumble after touch down are in close correspondance to the desired response. The results of a full performance calibration test, as it is called in flight simulation motion evaluation, can be found in (Koekebakker et al., 1998b). Also the light weight shuttle has been tested on top of the motion system in the Central Workshop at Mechanical Engineering as depicted in Fig. 8. Of course, final evaluation can only be performed after the simulator has put into operation at the Simona building where a new hydraulic power unit has to be connected.

[1] Matlab and Simulink are registered trademarks of the MathWorks, Inc.

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Modeling the structural dynamics of a flexible wind turbine: the Lagerwey LW-50/750

David-P. Molenaar‡ and Sjoerd Dijkstra

Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.
E-mail: D.P.Molenaar@wbmt.tudelft.nl

Abstract. For successful large scale application of wind energy the price of electricity generated by wind turbines should decrease. Model based control can be important since it has the potential to reduce fatigue loads, while simultaneously maintaining a desired amount of energy production. The controller synthesis, however, requires a mathematical model describing the most important dynamics of the complete wind turbine. In this paper a systematic approach for modeling the structural dynamics of flexible wind turbines wil be presented. The so-called superelement modeling method is used to divide the structure into a number of superelements consisting of four rigid bodies connected by torsional springs, and dampers. The stiffness of the torsional springs is derived directly from physical data. SD/FAST is used to generate the equations of motion. Finally, the power of this modeling approach will be demonstrated through comparing measured rotor blade eigenfrequencies with those of the superelement approximation.

Keywords. Structural modeling, multibody system, superelement, model-based control, wind turbine.

1 Introduction

In the 1970s the concern about the limited fossil fuel resources, and their impact on the environment awakened. Due to this growing concern, interest revived in using renewable energy sources in order to meet the constantly rising world energy demand. In addition, the oil crises of 1973 and 1979 led to the realization that the amount of energy import should be decreased so as to become less dependent of oil exporting countries. One way to use renewable energy sources is to generate electrical energy using wind turbines.

1.1 Wind energy

However, for successful large-scale application of wind energy the price of electricity generated by wind turbines should decrease. To achieve this goal, the main requirement is an increase of the reliability of wind turbines. At present, the reliability is lower than expected due to the underestimation of fatigue loads in the wind turbine design. Furthermore, over the recent years a number of wind turbines are blown to pieces during storms. These operating difficulties have resulted in a relatively high price of electricity generated by wind turbines.

Hence, the most important underlying motivation for wind energy research for large-scale energy production is the aim to reduce the price of the produced electrical energy. A wind turbine will then have low construction costs, long lifetime, low maintenance level, and efficient energy conversion. It is evident that each of these points more or less independently reduces the cost per unit of delivered electrical energy. Following from this ambition, there is a trend towards lighter and more flexible wind turbines. Additionally, advanced control can be im-

‡Author to whom correspondence should be addressed.
The research of David Molenaar is sponsored by Stork Product Engineering and Lagerwey Windturbine b.v.

portant (Bongers, 1994; Molenaar, 1995; Molenaar, 1996).

1.2 The Lagerwey LW-50/750 wind turbine

The joint project of Stork Product Engineering, Lagerwey Windturbine B.V., and Delft University of Technology offers the possibility to investigate the potential of advanced control design on a real wind turbine: the Lagerwey "LW-50/750".
The LW-50/750 is a 750 kW, variable speed wind turbine with an upwind rotor of 50.5-meter diameter. The rotor consists of 3 blades that can be actively and individually pitched over the full span. The pitch control is used for power control at full load, and to stop the turbine if the safe operating limits are exceeded. The turbine is equipped with a gearless synchronous ring-generator, which converts the mechanical power into electrical power at variable frequency. Subsequently, an IGBT based frequency converter is used to convert the electrical energy of varying frequency to the grid with a fixed 50 Hz. frequency. The generator torque will follow an external set point signal, which is calculated by the control computer. The support structure consists of a 46.165-meter conical tower with a circular cross section, and a foundation. The tower is made of tubular steel.
The Lagerwey LW-50/750 belongs to a new generation of wind turbines which are more flexible than the majority of the turbines now on the market. It has soft characteristics realized in all subsystems in order to reduce internal stresses, and thereby to make lighter, and hence less costly components possible. The explanation for this is that in a stiff system concept the transient air loadings go into the structural components as physical strains, and (sooner or later) induce fatigue damage. In a soft system concept, on the other hand, these air loadings are reacted primarily by the subsystem masses with little strain energy involvement. In other words: the subsystems bend under the transient air loading by which internal stresses will be reduced. Hence, there is low tendency for fatigue failure in a dynamically soft system.

1.3 Model based control

Flexible, variable speed wind turbines as the LW-50/750, however, require a careful design analysis. If the system is not well tuned, the potential benefits can be lost or even reversed. The reason for this is that due to the variable speed operation the system travels through a broad spectrum of excitation frequencies which implies that the eigenfrequencies of the individual components should be designed carefully. In addition, flexible systems will be more prone to instabilities (Bierbooms *et al.*, 1987).
Furthermore, since a wind turbine consists of mechanical components for which the fatigue loads are mainly determined by the damping of the mechanical resonance frequencies (Molenaar, 1995), there is the possibility that by active control of the generator shaft reaction torque the damping of these modes is increased, implying an additional reduction of the fatigue loading.
Besides reduction of fatigue loads, optimal energy should be extracted from the wind. These two control objectives are, however, more or less conflicting. Hence, a trade-off has to be made between the amount of acceptable fatigue loads and the desired energy production. To achieve the conflicting control objectives, a mathematical model describing the most important dynamics of the complete wind turbine is required to design such a controller. Hence, the following problem can be formulated, as a first step in modeling the complete system.

1.4 Problem formulation

Examine the possibility to devise a systematic approach for modeling the structural dynamics of flexible wind turbines like the Lagerwey LW-50/750, suited for time-domain simulation, analysis of dynamic loads, and model based control design.

1.5 Motivation of approach

The intended use of the models is thus for both time-domain simulation, analysis of dynamic loads, and control design purposes, which is difficult to achieve in the same model. The LW-50/750 should therefore modeled as a set of interacting modules (*e.g.* rotor, and tower) allowing easy configuration changes and providing maximum physical insight. In the wind energy community there is a wide variety in different design codes used to model a wind turbine's dynamic behaviour (Molenaar, 1998b). In general, these codes have been developed from a modeling, and not from a control design point of view. Furthermore, the current generation design codes has received very little validation against measurements from especially flexible, variable speed wind turbines (Pedersen, 1996). Therefore we have decided to develop a new wind turbine design code incorporating all current know-how regarding modeling, identification, and control of wind turbines. We will use the *general-purpose* simulation program SIMULINK to create *special-purpose* wind turbine dynamic models in a modular setting.
In this article we will limit ourselves to the modeling of the structural dynamics. The structural models should meet the following requirements:

- The structural model should allow easy interaction with in particular the aerodynamic submodule. This interaction should take place via rotor blade/tower movements and aerodynamical forces;
- The model parameters of the structural model should have a physical interpretation and besides that, it should be possible to derive them directly from (known) physical data;
- The model structure should allow experimental model parameter updating in order to reach the desired accuracy;
- The model order should stay limited, *i.e.* must be suited for model based control design.

The outline of this paper is as follows. First, in Section 2 the concepts and software tools for mechanical modeling are reviewed. Subsequently, the superelement modeling method will be discussed, and evaluated using an Euler-Bernoulli beam. Next, in Section 4 the developed systematic modeling approach will be presented and demonstrated on a single rotor blade. Section 5 concludes this paper.

2 Mechanical modeling: concepts and software tools

More and more engineering work relies on mathematical models of the studied object (Ljung and Glad, 1994). Loosely put, a model of a system is a tool we use to answer questions about the system without having to do an experiment. After all, it is sometimes more appropriate to perform simulations than to carry out experiments. The reason might be one of the following: it either is too **expensive**, too **dangerous**, or simply **impossible**: the real system does not (yet) exist. A dynamic model of a system of interest allows the designer to study more alternative concepts in shorter time, reduces the number of tests needed for design validation, and thus reduces the costs and risks of the complete design cycle. In addition, a dynamic model of a real system offers the possibility to design, and evaluate the impact of different control strategies on the system's performance. Hence, we have to make a *model* of the system under consideration.
In this paper we will focus our attention on mechanical models of flexible wind turbines. But what kind of mechanical model should be used or is most adequate for the actual problem? Without exception, all different kinds of mechanical models are based on the classical mechanics formulated by Sir Isaac Newton (1642-1727) in his book *Philosophiæ Naturalis Principia Mathematica* in 1687 (Newton, 1687).

At present, four different kinds of mechanical models are commonly used (Kreuzer, 1994):

- **Multibody Systems**. In the Multibody System (MBS) approach, a real mechanical system is approximated with a finite number of **rigid** bodies, coupled by inelastic joints (*e.g.* slider, pin) to the Newtonian reference frame;
- **Finite Element Systems**. In the FES method, the real flexible structure is regarded as an assembly of a finite, but large number geometrically simple, discrete elements;
- **Continuous Systems**. Continuous System (COS) consist of flexible bodies of which the mass and elasticity can be exactly mathematically represented;
- **Hybrid Multibody Systems**. A Hybrid Multibody System (HMBS) may built up of a combination off all three methods of modeling mentioned above, leading to the most complex model of a mechanical system.

The MBS approach leads to a finite number of differential equations. The COS mechanical modeling method, on the other hand, results in partial differential equations which can be solved exactly only for very simple geometric structures (*e.g.* beams). The FES approach results in complex, high order models mainly suited for layout, design, and thorough system analysis. Since we want a limited order model of the real system under consideration, we have decided to use the MBS approach to model the Lagerwey LW-50/750.

3 Superelement method

In the Multibody System approach a real mechanical system is approximated with a finite number of rigid bodies. However, when the deformation of (a part of) the system has a significant effect on the dynamic behaviour, the elasticity can no longer be neglected. Inclusion of elasticity by way of so-called flexible bodies is essential in order to reach the same level of accuracy as for stiff mechanical systems. The price to be paid is, of course, an increased model order. There are several ways of modeling flexible bodies within the MBS methodology, see for an overview *e.g.* (Shabana, 1985; Shabana, 1997). The simplest way is to equally distribute the mass of the flexible body into lumped masses, interconnected by ideal, massless springs and dampers. This is the so-called "lumped-mass method". A more accurate model is obtained by using the concept of the so-called "superelement" as introduced by Rauh and Schiehlen (Rauh, 1989; Rauh and Schiehlen, 1987).

Following this approach, a (part of a) flexible body is approximated with a number of superelements consisting of a series of rigid bodies connected by elastic force elements as springs, and dampers, *see* Fig. 1. The main question is "What should be the values of the spring constants in order to produce a comprehensive and accurate model of the flexible system?"

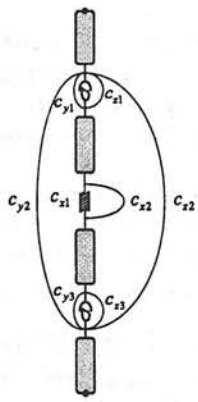


Fig. 1: *Superelement: i.e. multibody approximation of a flexible body consisting of four rigid bodies connected by springs, and dampers.*

3.1 Determination of spring constants

The purpose of this paragraph is to determine the torsional spring constants of each superelement. We will limit ourselves to bending; the interested reader is referred to (Molenaar, 1998a) for the axial and torsional result. Consider thereto an Euler-Bernoulli beam with length L loaded with a constant force F perpendicular to the longitudinal axis. Since this is a case of pure bending, we may use the basic differential equation of the deflection curve of a prismatic beam to determine the total deflection δ at the free end (GereTimoshenko, 1987), *i.e.*

$$\frac{d^2v}{dy^2} = -\frac{M}{EI}$$

with v the transverse displacement, $v'' = \frac{d^2v}{dy^2}$, EI the bending stiffness, and M the bending moment. Substituting the expression for the bending moment, the differential equation becomes

$$EIv'' = -M = FL - Fy$$

The first integration of this equation gives

$$EIv' = FLy - \frac{Fy^2}{2} + C_1$$

The constant of integration C_1 can be found from the condition that the slope of the beam is zero at the support; thus $v'(0) = 0$, which results in $C_1 = 0$. Therefore

$$EIv' = FLy - \frac{Fy^2}{2} \tag{1}$$

Integration of this equation yields

$$EIv = \frac{FLy^2}{2} - \frac{Fy^3}{6} + C_2$$

The boundary condition on the deflection at the support is $v(0) = 0$, which shows that $C_2 = 0$. Thus, the equation of the deflection curve is

$$v = \frac{Fy^2}{6EI}(3L - y) \tag{2}$$

The angle of rotation θ_F and the deflection δ_F at the free end of the beam are readily found by substituting $y = L$ into Eqs. (1) and (2) respectively. The equation of the deflection curve for an Euler-Bernoulli beam loaded by a couple M at the end of the beam can be determined analogously. The results for both cases are summarized in Eq. (3).

$$\begin{bmatrix} \delta \\ \theta \end{bmatrix} = \frac{1}{6EI}\begin{bmatrix} 2L^3 & 3L^2 \\ 3L^2 & 6L \end{bmatrix}\begin{bmatrix} F \\ M \end{bmatrix} \tag{3}$$

Inversion of this equation results in

$$\begin{bmatrix} F \\ M \end{bmatrix} = \frac{EI}{L^3}\begin{bmatrix} 12 & -6L \\ -6L & 4L^2 \end{bmatrix}\begin{bmatrix} \delta \\ \theta \end{bmatrix} \tag{4}$$

From Fig. 2 it can be easily derived that

$$\begin{bmatrix} c_{z1}+c_{z2} & c_{z2} \\ c_{z2} & c_{z2}+c_{z3} \end{bmatrix}\begin{bmatrix} \Delta\gamma_1 \\ \Delta\gamma_2 \end{bmatrix} = \\ = \begin{bmatrix} (1-k)L & 1 \\ Lk & 1 \end{bmatrix}\begin{bmatrix} F \\ M \end{bmatrix} \tag{5}$$

and that the following relation holds

$$\begin{bmatrix} \delta \\ \theta \end{bmatrix} = \begin{bmatrix} L(1-k) & kL \\ 1 & 1 \end{bmatrix}\begin{bmatrix} \Delta\gamma_1 \\ \Delta\gamma_2 \end{bmatrix} \tag{6}$$

with k the partitioning coefficient. Substituting Eq. (5) and (6) in Eq. (4) gives the following spring coefficients

$$c_{x1} = \frac{6EI_x}{L}(1-2k)^2 \tag{7}$$

$$c_{z1} = \frac{6EI_z}{L}(1-2k)^2 \tag{8}$$

$$c_{x3} = \frac{6EI_x}{L}(1-2k)^2 \tag{9}$$

$$c_{z3} = \frac{6EI_z}{L}(1-2k)^2 \tag{10}$$

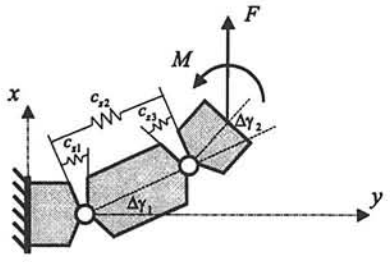


Fig. 2: *Defections and slopes of a superelement with length L. Each (symmetric) superelement consists of three rigid bodies with lengths kL, $(1-2k)L$, and kL.*

$$c_{x2} = \frac{2EI_x}{L}(-1+6k-6k^2) \tag{11}$$

$$c_{z2} = \frac{2EI_z}{L}(-1+6k-6k^2) \tag{12}$$

in which I_x, and I_z represent the area moments of inertia with respect to the centroid C of an Euler-Bernoulli beam of length L. Since the polar area moment of inertia with respect to an axis perpendicular to the plane of the figure at point C is equal to the sum of the area moments of inertia with respect to any two perpendicular axes x and z passing through the same point and lying in the plane of the figure, we get the following expression for I_x and I_z

$$I_x = I_z = \tfrac{1}{2}I_p = \tfrac{1}{4}\pi R^4 \tag{13}$$

The partitioning coefficient k $(0 < k < \frac{1}{2})$ of the superelement exerts influence on the kind of approximation of the eigenfrequencies. In (Rauh, 1989) is has been concluded that choosing a partitioning coefficient from the range $\frac{1}{5} \leq k \leq \frac{1}{4}$ results in models that approximate the exact eigenfrequencies with a limited number of superelements. Furthermore, with $k = \frac{1}{2}(1-\frac{1}{\sqrt{3}}) \approx 0.211$ it follows that $c_{x2} = c_{z2} = 0$, and that Eq. (7) to (10) reduce to

$$c_{x1} = \frac{2EI_x}{L} \tag{14}$$

$$c_{z1} = \frac{2EI_z}{L} \tag{15}$$

$$c_{x3} = \frac{2EI_x}{L} \tag{16}$$

$$c_{z3} = \frac{2EI_z}{L} \tag{17}$$

resulting in a relatively simple model. In this case, some eigenfrequencies are smaller, and some are larger than in reality, while the errors decrease very fast.

3.2 Comparison of the exact eigenfrequencies with Superelement approximation

Now we will compare the exact eigenfrequencies of an Euler-Bernoulli beam with those of the Superelement approximation in order to pass judgment about the appropriateness to use one of the approximations for modeling the structural dynamics of flexible mechanical systems.
The Euler-Bernoulli beam considered has length $L = 50$ m, modulus of elasticity $E = 21 \cdot 10^{10}$ N/m^2, area of beam cross-section $A = \pi$ m^2, mass per unit length $\rho = 7850$ kg/m^3, and area moment of inertia $I = \frac{1}{4}\pi$ m^4.

3.2.1 Exact eigenfrequencies

The exact analytical eigenfrequencies are obtained by solving the following frequency equation (Weaver *et al.*, 1990):

$$\cos(kL)\cosh(kL) = -1$$

The first four roots of this equation are:

$$\begin{aligned} k_1 * L &= 1.875104069 \\ k_2 * L &= 4.694091133 \\ k_3 * L &= 7.854757438 \\ k_4 * L &= 10.99554073 \end{aligned}$$

The exact eigenfrequencies in radians per second are:

$$\omega_i = k_i^2 \cdot \sqrt{\frac{EI}{\rho A}} \tag{18}$$

where E the modulus of elasticity, I area moment of inertia, ρ the mass density of the material, and A the cross-sectional area of the beam. These frequencies are listed in Fig. 3, and will serve as reference solution.

Mode	Exact eigenfrequencies
1	3.637
2	22.79
3	63.82
4	125.1

Fig. 3: *The first four exact eigenfrequencies of the Euler-Bernoulli beam.*

3.2.2 Superelement approximation

Now we will approximate the Euler-Bernoulli beam using a number of superelements. The beam is built

in at the base. The torsional spring constants for each superelement are determined as follows:

$$c_{z1} = c_{z3} = \frac{2EI_z}{L_{se}} \qquad [\text{Nm}]$$

with E the modulus of elasticity, I_z the area moment of inertia, and L_{se} the length of the superelement which is, in turn, defined as

$$L_{se} = \frac{L}{N_{se}}$$

with L the length of the Euler-Bernoulli beam and N_{se} the number of superelements the beam is subdivided in. SD/FAST (Hollars *et al.*, 1994), a *general-purpose* multibody program which automatically derives *special-purpose* simulation code for mechanical systems approximated by rigid bodies, is used to generate the equations of motion.
The first four eigenfrequencies of the Superelement approximation as function of the number of superelements are listed in Fig. 4.

Mode	Number of superelements N_{se}, with $k = \frac{1}{2}(1-\frac{1}{\sqrt{3}})$				
	1	2	3	4	5
1	3.599	3.636	3.636	3.636	3.636
2	36.86	22.12	22.70	22.75	22.75
3	-	73.88	60.90	63.18	63.46
4	-	159.7	140.0	117.8	122.9

Fig. 4: *The first four eigenfrequencies in radians per second of the Superelement approximation as function of the number of superelements.*

The exact analytical solution is used to evaluate the Superelement approximation. In order to do so, the relative frequency errors are computed. This error is defined as:

$$\text{Rel. error} = \frac{\text{Multibody eigenfreq.}}{\text{Exact eigenfreq.}} \cdot 100 - 100 \qquad [\%]$$

The relative errors for the first four eigenfrequencies of the Superelement approximation are plotted in Fig. 5. The pattern is clear: dividing the beam into more superelements produces more eigenfrequencies (of which only the first four are shown), and improves the accuracy. The limiting case being an infinite number of superelements of which the eigenfrequencies equal to those of the exact solution. In addition, the mode shapes become better defined with an increasing number of superelements, since information on more locations along the beam is available. As expected, some eigenfrequencies are smaller, and some are larger than in reality, while the errors reach the indicated 1% error bound rather fast.

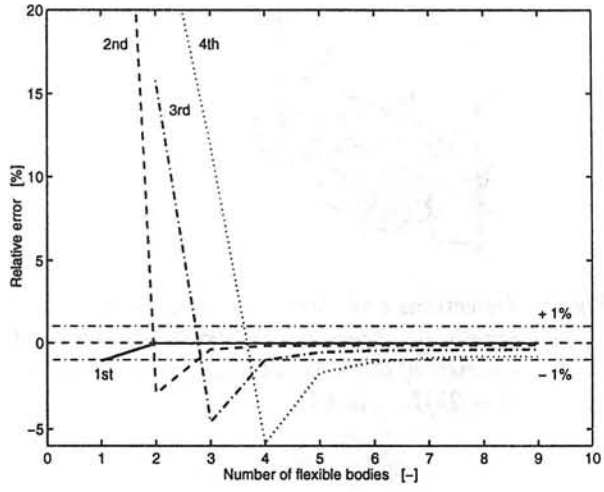


Fig. 5: *The relative errors for the first four eigenfrequencies of the Superelement approximation as function of the number of superelements N_{se} with $k = \frac{1}{2}(1 - \frac{1}{\sqrt{3}})$. Dashed-dotted lines: + 1% and - 1% error bound.*

3.3 Centrifugal stiffening

It is important to stress that the applied superelement modeling approach automatically acounts for centrifugal stiffening effects. After all, as the length of the rigid bodies within each superelement are constant, it follows that deformation of the blade automatically produces axial deformations and thereby automatically produces centrifugal stiffening. It is shown in (Molenaar, 1998a) that for an Euler-Bernoulli beam this modeling approach represents a consistent approximation to the continuum model, in the sense that it represents a discretization of the continuum model with an approximation accuracy that increases with an increasing number of superelements.

4 Multibody model of the LW-50/750

In this section we will present a systematic approach for modeling the structural dynamics of flexible wind turbines like the Lagerwey LW-50/750, suited for time-domain simulation, analysis of dynamic loads, and model based control design. The so-called superelement modeling method will be used to divide the structure into a number of superelements consisting of a series of rigid bodies connected by torsional springs, and dampers. The stiffness of the torsional springs will be derived directly from physical data. SD/FAST is used to generate the equations of motion.
The resulting models of varying complexity can be easily coupled to obtain a model describing the

structural behaviour of the complete Lagerwey LW-50/750 or any other flexible wind turbine. All models are available within the MATLAB/SIMULINK environment. We will demonstrate the modeling approach by comparing the non-rotating rotor blade eigenfrequencies obtained from a full-scale modal test with those from the superelement approximation.

4.1 Example: APX-45 rotor blade

The rotor blades of the Lagerwey LW-50/750 are designed, and manufactured by Aerpac Special Products B.V., Hengelo, The Netherlands (Rodenburg *et al.*, 1996). The blades are mainly made of glass fiber reinforced epoxy (GRE). The applied laminated composite layer structure and the fact that the layers of the composite are composed of orthotropic material implies that it can cost a significant amount of time deciphering the parameters defining the blade. Here it is assumed that the blade definition file is provided by the manufacturer.
The APX-45 rotor blade will be subdivided into a number of superelements. Chosen is to use the rotor blade specific FAROB output file Table.frb as the starting point. FAROB is the structural blade modeling module within the wind turbine design code FOCUS used to design the APX-45 rotor blade. Table.frb contains - among other things - the blade mass, and the flexural rigidity in the two principal bending directions at a number of locations beginning at the blade tip and ending with the blade root. Undefined locations are interpolated in a subsequent step after converting the file to a MATLAB MAT-file. Note that, in principle, any file that contains the mentioned blade data can be used as starting point. Obviously, the resulting model accuracy strongly depends on the quality of the input data.
At this point the user has to select the complexity of the rotor blade model by specifying the number of superelements the blade is to be subdivided in. The required SD/FAST parameters are then automatically generated and stored in a blade specific file with the specified number of superelements appended to the basename "Blade" (*e.g.* Blade2.mat contains the SD/FAST data of a rotor blade divided into 2 superelements). Finally, the SD/FAST data is converted into a pretty format, and at the same time loaded into MATLAB's workspace.
The developed blade modeling procedure consists of three main steps:

- **Step 1:** Run Frb2Mat.m by simply typing

  ```
  >> Frb2Mat
  ```

  in the MATLAB command window to convert a rotor blade specific FAROB output file Table.frb to a MATLAB MAT-file named FileName.mat. "FileName" is a user-selected name (*e.g.* APX45) which has to be entered when the following message emerges in the command window: `Name for generated MAT-file:` This step is illustrated in the upper part of Fig. 6.

- **Step 2:** Run BladePM.m by typing

  ```
  >> BladePM
  ```

  in the MATLAB command window. The user is asked first to enter in the command window the name of the generated MAT-file containing the blade data. Subsequently, the parameters in this file are converted into a blade specific file named BladeX.mat with "X" a user-specified number representing the number of superelements N_{se} in which the blade is subdivided. This step is illustrated in the middle part of Fig. 6.

- **Step 3:** Conversion of BladeX.mat to the required SD/FAST input parameters (Mass, Inertia, Inb2Joint, Body2Joint) and spring/damper constants by running BladeXda.m. This step is also required to load the data into the workspace for each simulation and is illustrated in the lower part of Fig. 6.

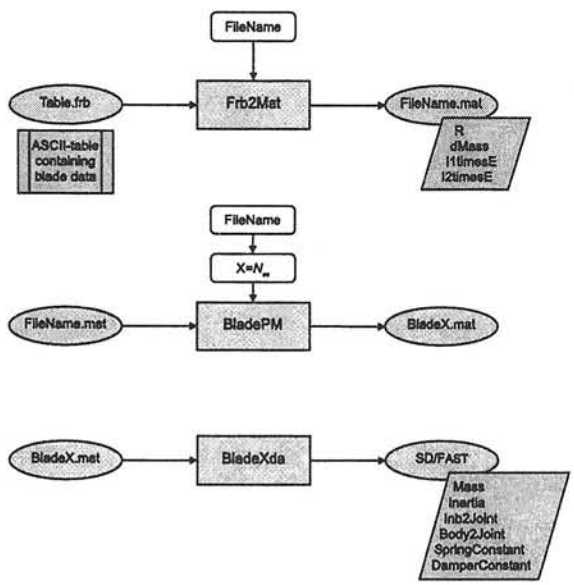


Fig. 6: *Main three steps in the blade modeling procedure.*

The non-rotating rotor blade eigenfrequencies obtained from the full scale modal test performed by the Stevin Laboratory of Delft University of Technology (Van Leeuwen *et al.*, 1997) are used to evaluate the multibody approximation. In order to do

so, the relative frequency errors are computed. This error is defined as:

$$\text{Rel. error} = \frac{\text{Multibody eigenfreq.}}{\text{Measured eigenfreq.}} \cdot 100 - 100 \qquad [\%]$$

The relative errors for the first two eigenfrequencies of the multibody approximation are plotted in Fig. 7. From this figure it is clear that the errors do not converge to zero with an increasing number of superelements. Investigation has shown that they converge to zero when compared to the eigenfrequencies as computed by FAROB. This illustrates the fact that the quality of the models is limited by the input. The main reason that the blade structural properties in reality may differ from predicted values is that the blades are manufactured by hand-lay-up. However, the results show that it is possible to derive limited order multibody models suited for time-domain simulation, analysis of dynamic loads, and model based control design directly from physical data.

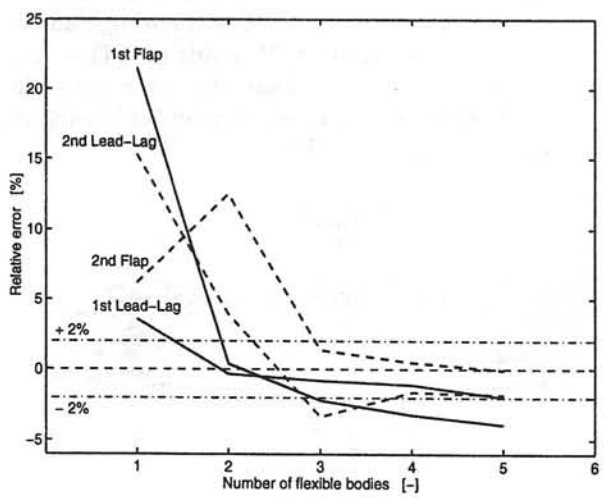


Fig. 7: *The relative errors for the first two flap and lead-lag eigenfrequencies of the APX-45 rotor blade as function of the number of superelements. Dashed-dotted lines: + 2% and - 2% error bound respectively.*

5 Conclusions

In this paper the possibility to devise a systematic approach for modeling the structural dynamics of flexible wind turbines like the Lagerwey LW-50/750, suited for time-domain simulation, analysis of dynamic loads, and model based control design has been investigated. The following conclusions can be drawn:

- The so-called superelement modeling method used to divide the structure into a number of superelements consisting of a series of rigid bodies connected by torsional springs, and dampers represents a consistent approximation to the continuum model. In the sense that it represents a discretization of the continuum model with an approximation accuracy that increases with an increasing number of superelements.

  Comparison of the exact eigenfrequencies of an Euler-Bernoulli beam with those from the superelement method has shown that the latter is particularly useful for approximating the first number of eigenfrequencies with a limited number of superelements. In general, these lowest frequency modes have the largest amplitude and are the most important to be approximated well for time-domain simulation, analysis of dynamic loads, and control system design;

- The power of the presented modeling approach to approximate the dynamic behaviour of flexible mechanical systems with limited order multibody models derived directly from physical data has been shown by comparing the rotor blade non-rotating eigenfrequencies with those obtained from a full scale modal test. Hence, it can be concluded that the presented approach is very suitable for modeling the structural dynamics of flexible wind turbines for time-domain simulation, analysis of dynamic loads, and control design.

6 Future work

Modeling without a making statement about the quality of the obtained models is useless. In this article modal test data is used to validate the obtained rotor blade models. However, in a next step the rotor (consisting of three blades connected to the hub) will be dynamically coupled to the tower. The resulting system modes may not be equal to the sum of the component modes. Besides that there are a number of uncertainties (*e.g.* how rigid or flexible is the tower connected to the concrete foundation?) which have significant impact on the model quality. Therefore, the structural model of the Lagerwey LW-50/750 will be validated using experimental data. This allows us to experimentally determine the damping too.
Furthermore, we have focused our attention in this article on the error between the measured and modeled eigenfrequencies. Thereby assuming that the mode shapes become better defined with an increasing number of superelements. Although this is a reasonable assumption (note that with an increas-

ing number of superelements information on more locations along the beam is available) in a next step attention should be payed to the error between the measured, and modeled eigen vectors (or: mode shape). After all, the motion a superposition of the various mode shapes, each at different amplitude.

References

Bierbooms, W.A.A.M., P.M.M. Bongers and M. Steinbuch (1987). *An Integrated Dynamic Model of a Wind Turbine.* Delft University of Technology, Institute for Wind Energy, IW 87.011 R, 73p.

P.M.M. Bongers (1994). *Modeling and Identification of Flexible Wind Turbines and a Factorizational Approach to Robust Control.* PhD. Thesis, Delft University of Technology.

J.M. Gere and S.P. Timoshenko (1987). *Mechanics of Materials.* Van Nostrand Reinhold (International), Hong Kong.

Hollars, M.G., D.E. Rosenthal and M.A. Sherman (1994). *SD/FAST User's Manual.* Symbolics Dynamics, Inc., Version B.2.

Kreuzer, E. (1994). *Computerized Symbolic Manipulation in Mechanics.* Springer Verlag.

Ljung, L. and T. Glad (1994). *Modeling of Dynamical Systems.* Prentice-Hall, Englewood Cliffs, New Jersey.

Molenaar, D-P. (1995). *Control Design: A Way of Reducing Fatigue Loads on a Wind Turbine? - NedFlex: A Flexible, Variable Rotational Speed Wind Turbine.* Mechan. Engin. Systems and Control Group, Delft Univ. Technology, The Netherlands, TUD-WBMR-S-746.

Molenaar, D-P. (1996). *Modeling and Control of the NedFlex Turbine - NedFlex: a Flexible, Variable Rotational Speed Wind Turbine.* Mechan. Engin. Systems and Control Group, Delft Univ. Technology, The Netherlands, TUD-WBMR-A-746.

Molenaar, D-P. (1998a). *Modeling the Structural Dynamics of the Lagerwey LW-50/750 Turbine.* Report N-522, Mechan. Engin. Systems and Control Group, Delft Univ. Technology.

Molenaar, D-P. (1998b). *State of the Art of Wind Turbine Design Codes.* Report N-523, Mechan. Engin. Systems and Control Group, Delft Univ. Technology.

Newton, I. (1687). *Philosophiæ Naturalis Principia Mathematica.* Jussu Societatis Regiæ ac Typis Josephi Streater, London.

B.M. Pedersen (1996). State of the Art of Aeroelastic Codes for Wind Turbine Calculations - Summary. *Proceedings of the 28^{th} IEA Meeting of Experts "State of the Art of Aerolelastic Codes for Wind Turbine Calculations"*, Technical University of Denmark, Lyngby, pp. 207-208.

Rauh, J. (1989). *Ein Beitrag zur Modellierung elastischer Balkensysteme.* Fortschritt-Berichte VDI, VDI-Verlag GmbH, Düsseldorf, Reihe 18: Mechanik/Bruchmechanik, Nr. 37, 126p.

Rauh, J. and W. Schiehlen (1987). Various Approaches for the Modeling of Flexible Robot Arms. *Proceedings of the Euromech-Colloquium 219 on Refined Dynamical Theories of Beams, Plates, and Shells and their Applications*, Kassel, 1986. Springer-Verlag, Berlin Heidelberg, pp. 420–429.

Rodenburg, R.V., C. Biever and J.K. de Boer (1996). Design of the APX45/750 Rotor Blade with the Integrated Design Tool FOCUS. *Contributions European Union Wind Energy Conference, EUWEC 1996*, Göteborg, Sweden, IW-96106, pp. 37–40.

Shabana, A.A. (1997). Flexible Multibody Dynamics: Review of Past and Recent Developments. *Multibody System Dynamics, 1*, 189–222.

Shabana, A.A. (1985). Substructure synthesis methods for dynamic analysis of multi-body systems. *Computers and Structures, 20*, 737–744.

Van Leeuwen, J.L. and D.R.V. van Delft (1997). *Full Scale Tests on the APX45 Rotor Blade part II; Natural Frequency Tests.* Stevin report 6-96-9, 6p.

Weaver, W., S.P. Timoshenko and D.H. Young (1990). *Vibration Problems in Engineering.* John Wiley & Sons, Inc., New York.

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Realization algorithms for expansions in generalized orthonormal basis functions[‡]

P.S.C. Heuberger[§⋆], Z. Szabó[♯], T.J. de Hoog[§], P.M.J. Van den Hof[§¶] and J. Bokor[♯]

[§] *Mechanical Engineering Systems and Control Group*
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.
[♯] *Computer and Automation Research Institute*
Hungarian Academy of Sciences, Kende u. 13-17, H 1502 Budapest, Hungary

Abstract. In this paper a realization theory and associated algorithms are presented for the construction of minimal realizations on the basis of a sequence of expansion coefficients in a generalized orthonormal basis. Both the exact and the partial realization problem are addressed and solved, leading to extended versions of the classical Ho-Kalman algorithm which is restricted to handling expansion sequences in the standard basis functions z^{-k}. In the construction of the realization algorithms, fruitful use is made of a system analysis in the transform domain, being induced by the choice of basis functions. The resulting algorithms can also be applied in approximate realization and in system approximation.

Keywords. Orthogonal basis functions; discrete-time systems; system approximation; realization; identification.

1 Introduction

In recent years renewed attention has been paid to the development and use of orthonormal basis functions in system theory, and particularly in system identification. Considering linear system descriptions in terms of orthogonal basis functions expansions, linearly parametrized models can result by restricting the models to finite expansions and considering the expansion coefficients as the parameters to be estimated.

In this work the choice of basis functions is known to be rather crucial for determining the length of the expansion that is needed (and thus the number of parameters that has to be estimated) for arriving at accurate system descriptions. The flexibility that is available in the basis construction mechanism can essentially contribute to a fast convergence of the series expansion for a specific system.

For Laguerre functions (Wahlberg, 1991), this flexibility rests in the choice of a single pole location, while two pole locations can be fixed in the two-parameter Kautz functions (Kautz, 1954; Wahlberg, 1994). Generalized versions of these approaches have been developed by Heuberger et al. (1995) using repeated blocks of all-pass sections of user-chosen order, and have been analysed for system identification purposes in Van den Hof et al. (1995). Ninness and Gustafsson (1997) have presented and analysed an alternative structure where the need for repetition of all-pass sections has been removed.

These generalized orthonormal basis functions (GOBF) have been shown to be powerful not only in the classical prediction error identification problem, but also in frequency domain identification (Ninness and Gómez, 1996; Schipp et al., 1996; De Vries and Van den Hof, 1998), model uncertainty estimation (De Vries and Van den Hof, 1995; Hakvoort

[‡]This paper has been accepted for presentation at the 1999 IFAC World Congress, Beijing, China, July 4-9, 1999.

[⋆]Peter Heuberger is on partial leave from the Dutch National Institute of Public Health and the Environment(RIVM).

[¶]Author to whom correspondence should be adressed. E-mail: P.M.J.vandenHof@wbmt.tudelft.nl. The research reported here has been supported by the Dutch Technology Foundation (STW) under contract DWT55.3618.

and Van den Hof, 1997) and system approximation (Wahlberg and Mäkilä, 1996; Oliveira e Silva, 1996). The particular basis functions of Heuberger et al. (1995) give rise to a general theory on signal and system transformations induced by these so-called Hambo basis functions, such that a signal $x(z)$ and a system $P(z)$ admit alternative descriptions, denoted by $\widetilde{x}(\lambda)$ and $\widetilde{P}(\lambda)$ in the transform domain. This is caused by the repetiton of basis poles that is involved in these functions. These transformations have been analysed in Heuberger and Van den Hof (1996).

Here we address the following problem:

Consider a linear time-invariant system in the form of a series expansion

$$P(z) = \sum_{k=1}^{\infty} c_k F_k(z)$$

where $\{F_k(z)\}_{k=1,2,\cdots}$ is a sequence of (orthonormal) basis functions in $\mathcal{H}_2$. The problem is to construct a minimal state space realization (A, B, C, D) of P on the basis of the sequence of coefficients $\{c_k\}_{k=1,\cdots,N}$. For $N = \infty$ and $F_k(z) = z^{-k}$ the problem reduces to a minimal (exact) realization problem, as solved by the celebrated Ho-Kalman algorithm (Ho and Kalman, 1966). Using the same basis functions, and considering finite N, the corresponding minimal *partial* realization algorithm is analysed and solved in Tether (1970), including the formulation of conditions under which the problem has a unique solution. In this latter situation, the required algorithm for obtaining a solution is basically the same Ho-Kalman algorithm as applied in the infinite data case.

Szabó and Bokor (1997) have extended the exact realization theory to the situation of Hambo basis functions, for the case of infinite data ($N = \infty$), see also Szabó et al. (1998). In this paper it will be shown how these results connect to system descriptions in the related transform domain, and additionally the partial realization problem ($N < \infty$) will be addressed and solved, and consequences will be indicated also for the approximative case.

The paper is structured as follows: first in section 2 the theory on generalized orthonormal basis functions is summarized. In sections 3 and 4 some results are formulated that are instrumental in handling the realization problems. Next the exact realization problem is addressed in section 5 and the partial problem in section 6. Some comments on the approximate realization problem conclude the paper.

Unless otherwise stated, all systems in this paper are scalar systems in the discrete time domain. Multivariable extensions are straightforward. In the discrete time domain the space $\mathcal{H}_2$ can either be defined to include only strictly proper systems or to include proper systems as well. In this paper we will use the latter definition, formally defined by stating that $\mathcal{H}_2$ consists of all complex valued functions which are analytic outside the closed unit disk and squared integrable over the unit circle. In this version of the paper all proofs are omitted.

2 Generalized orthonormal basis functions

In this section the main ingredients of the theory on orthonormal basis functions will be briefly reviewed. For details see Heuberger et al. (1995), Van den Hof et al. (1995), Heuberger and Van den Hof (1996), Ninness and Gustafsson (1997). The basis functions are constructed from state trajectories related to balanced realizations of square inner functions (i.e. stable all-pass systems). A transfer function $G_b \in \mathcal{H}_2$ is inner if it satisfies $|G_b(e^{i\omega})| = 1$ for all $\omega \in [-\pi, \pi]$. It was shown in Roberts and Mullis (1987) that (in discrete time) inner functions can be realized by particular state space realizations that are orthogonal, i.e. they satisfy $G_b(z) = D + C(zI - A)^{-1}B$ where

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^* \begin{bmatrix} A & B \\ C & D \end{bmatrix} = I. \tag{1}$$

It is straightforward to show that the elements $\phi_1(z), \cdots, \phi_{n_b}(z)$ of the n_b dimensional vector function

$$V_1(z) = [zI - A]^{-1}B \in \mathcal{H}_2^{n_b} \tag{2}$$

are mutually orthonormal in the standard H_2 inner product sense (with $\mathbb{T}$ denoting the unit circle),

$$< \phi_i, \phi_j >= \frac{1}{2\pi i} \oint_{\mathbb{T}} \phi_i(z)\phi_j^*(1/z^*)\frac{dz}{z} = \delta_{ij}. \tag{3}$$

Furthermore consecutive multiplication of these functions with $G_b(z)$ results in an orthonormal set, given by the components $\{\phi_{(k-1)n_b+1}, \cdots, \phi_{kn_b}\}$ of the n_b dimensional vector functions

$$V_k(z) = [zI - A]^{-1}BG_b^{k-1}(z) \qquad k \in \mathbb{N} \tag{4}$$

and it can be shown that the set of all these functions $\{\phi_i(z)\}_{i=1}^{\infty}$ constitutes a basis for the strictly proper part of H_2.

Directly resulting from the basis for strictly proper stable systems in $\mathcal{H}_2$, a basis for the related Hilbert space of $\ell_2[1, \infty)$-signals follows, by considering the inverse z-transforms of the $\mathcal{H}_2$-signals. Denoting

$$V_k(z) \;=\; \sum_{t=1}^{\infty} v_k(t) z^{-t} \tag{5}$$

$$v_k(t) := \frac{1}{2\pi}\int_{-\pi}^{\pi} V_k(e^{i\omega})e^{it\omega}d\omega. \quad (6)$$

the components of the n_b-dimensional functions $v_k(t)$ will constitute an orthonormal basis for $\ell_2[1,\infty)$. Note that these functions are in fact the impulse responses of the functions $V_k(z)$ with the property that

$$v_1(t) = A^{t-1}B \quad (7)$$
$$v_{k+1}(t) = (G_b(q)\cdot I)v_k(t). \quad (8)$$

When it is neccesary to know the underlying realization these functions will be denoted by

$$v_k(t)^{\{A,B,C,D\}} \quad (9)$$

Considering this general class of basis functions, for any strictly proper system $H(z) \in H_2$ or signal $y(t) \in \ell_2[1,\infty)$ there exist unique series expansions:

$$H(z) = \sum_{k=1}^{\infty} L_k^T V_k(z) \qquad L_k \in \mathbb{R}^{n_b\times 1}, \quad (10)$$

$$y(t) = \sum_{k=1}^{\infty} \mathcal{Y}^T(k)v_k(t) \quad \mathcal{Y}(k) \in \mathbb{R}^{n_b\times 1}. \quad (11)$$

Note that these basis functions can incorporate system dynamics in a very general way. One can construct an inner function G_b from any given set of poles, and thus the resulting basis can incorporate dynamics of any complexity, combining e.g. both fast and slow dynamics in damped and/or resonant modes.
For specific choices of $G_b(z)$ well known classical basis functions can be generated, such as the standard pulse basis $V_k(z) = z^{-k}$, the Laguerre basis and the two-parameter Kautz functions (Kautz, 1954; Wahlberg, 1994), see Heuberger et al. (1995).

Signal and system transformations

The ℓ_2-basis functions generate a signal transformation $\ell_2[1,\infty) \to \ell_2^{n_b}[1,\infty)$ as follows. Let $y(t) \in \ell_2[1,\infty)$ with an expansion as in equation (11), then we denote the *Hambo* transform as the mapping H: $\ell_2[1,\infty) \to H_2^{n_b}$, determined by

$$\mathsf{H}(y) := \widetilde{y}(\lambda) = \sum_{k=1}^{\infty} \mathcal{Y}(k)\lambda^{-k} \quad (12)$$

This Hambo transform of ℓ_2-signals also induces a linear system transformation. This transformed system describes the dynamical relationship between (transformed) input and output signals:

Let $P \in \mathcal{H}_2$ and let $u, y \in \ell_2$ such that $y(t) = P(q)u(t)$. Consider the Hambo transform $\widetilde{u}(\lambda), \widetilde{y}(\lambda)$ of the ℓ_2 signals u, y as defined above, then there exists a $\widetilde{P} \in H_2^{n_b\times n_b}$ satisfying

$$\widetilde{y}(\lambda) = \widetilde{P}(\lambda)\widetilde{u}(\lambda). \quad (13)$$

The mapping T: $H_2 \to H_2^{n_b\times n_b}$ defined by $\mathsf{T}(P) := \widetilde{P}(\lambda)$ is referred to as the *Hambo system-transform.* It turns out that this function $\widetilde{P}(\lambda)$ can be expressed through variable transformations, using the $n_b \times n_b$ inner function

$$N(z) := A + B(z-D)^{-1}C \quad (14)$$

with McMillan degree 1 and balanced realization (D, C, B, A). If we write $P(z) = \sum_{k=1}^{\infty} p_k z^{-k}$, then

$$\widetilde{P}(\lambda) = \sum_{k=1}^{\infty} p_k N^k(\lambda) \quad (15)$$

or differently denoted: $\widetilde{P}(\lambda) = P(z)|_{z^{-1}=N(\lambda)}$.
Important properties of the Hambo system transform are invariance of the McMillan degree and the fact that

$$\widetilde{G}_b^k(\lambda) = \lambda^{-k} I_{n_b}. \quad (16)$$

From a signal processing point of view the representation of systems as operators acting on transformed signals fits into the H-matrix representation framework as discussed by Audley and Rugh (1973).

Given this background material the problem addressed in this paper can be formulated as follows:

Given a system $G(z)$ and an orthonormal basis $\{V_k(z)\}_{k=1}^{\infty}$, such that $G(z) = \sum_{k=1}^{\infty} L_k^T V_k(z)$ construct an algorithm to derive a minimal state space realization of $G(z)$, based on the expansion sequence $\{L_1, \cdots, L_N\}$.

This problem will be approached through the Hambo system transform, by revealing the relation between the expansion coefficients $\{L_k\}$ and the Markov parameters of the transformed system $\widetilde{G}(\lambda)$ and thus creating the Hankel[1] matrix of the transformed system, denoted by $\widetilde{H}$. Furthermore the relation between $\widetilde{H}$ and the Hankel matrix of $G(z)$, denoted by H, will be established. At the same time these relations are given for the shifted system $\overleftarrow{G}(z) := zG(z)$, with Hankel matrix $\overleftarrow{H}$ and its Hambo transform $\widetilde{\overleftarrow{G}}(z)$, with Hankel matrix $\widetilde{\overleftarrow{H}}$. The solution of the problem is based on an extension of the Ho-Kalman algorithm and the solution of the classical minimal partial realization problem, given by Tether (1970).

[1] The Hankel matrix associated with the Hankel *operator.*

3 The Hambo transform of $\phi_k(z)$

The key property of the basis functions, that will be used in this paper, is the fact that the Hambo transform of the elements of $V_1(z)$ can be calculated explicitly, as stated in the following proposition:

Proposition 3.1 *Let the vector valued function* $V_1(z) = \left[\ \phi_1(z)\ \ \phi_2(z)\ \ \cdots\ \ \phi_{n_b}(z)\ \right]^T$ *be given by (2). Then for* $1 \leq k \leq n_b$ *there exist matrices* $P_k, Q_k \in \mathbb{R}^{n_b \times n_b}$ *such that*

1. *the Hambo system transform of* $\phi_k(z)$ *satisfies*

$$\widetilde{\phi_k}(\lambda) = P_k^T + Q_k^T \lambda^{-1} \qquad 1 \leq k \leq n_b.$$

2. *the matrices* $\{P_k, Q_k\}$ *are the solution of the following set of Sylvester equations:*

$$\begin{aligned} AP_kA^T &+ Be_k^TA^T = P_k \\ AQ_kA^T &+ Be_k^TC^TB^T + AP_kC^TB^T = Q_k \end{aligned}$$

where e_k *is the* k^{th} *Euclidean unit vector.* □

This so-called 2-Markov parameter description is possible, because the poles of $\phi_k(z)$ are contained in the poles of $G_b(z)$ (Heuberger and Van den Hof, 1996).
To facilitate the notation in the following sections we introduce a compact form for the set of all matrices $\{P_k, Q_k\}$:

$$\mathcal{P} := \left[\ P_1\ \ P_2\ \ \cdots\ \ P_{n_b}\ \right] \in \mathbb{R}^{n_b \times n_b^2} \quad (17)$$

$$\mathcal{Q} := \left[\ Q_1\ \ Q_2\ \ \cdots\ \ Q_{n_b}\ \right] \in \mathbb{R}^{n_b \times n_b^2} \quad (18)$$

4 From expansion coefficients to Hankel matrices in the transform domain

In this section it will be shown how the material of the previous sections allows the calculation of the Hambo system transform, given an expansion in terms of orthonormal basis functions. The system transform will be expressed in terms of Markov parameters, connecting directly to the Hankel matrix of the system transform. Hence, given an expansion $G(z) = \sum_{k=1}^{\infty} L_k^T V_k(z)$ we will show how this can be translated into an expansion $\widetilde{G}(\lambda) = \sum_{k=0}^{\infty} M_k \lambda^{-k}$. The result is given in terms of the notation (17,18), and $\otimes$ denoting Kronecker product.

Proposition 4.1

$$\begin{aligned} \widetilde{G}(\lambda) &= \sum_{k=0}^{\infty} M_k \lambda^{-k}, \qquad \textit{where} \\ M_0 &= [L_1^T \otimes I]\mathcal{P}^T \quad (19) \\ M_k &= [L_{k+1}^T \otimes I]\mathcal{P}^T + [L_k^T \otimes I]\mathcal{Q}^T \quad (20) \end{aligned}$$

□

In the course of the paper we will need a similar expression for the Hambo transform of the *shifted* function $\overleftarrow{G}(z) := zG(z)$. This expression readily follows from the observation that

$$V_1(z)z = B + AV_1(z) \quad (21)$$

which shows that

$$zG(z) = \sum_{k=1}^{\infty} \Big((L_k^TB) + (L_k^TA)V_1(z) \Big) G_b^{k-1}(z) \quad (22)$$

$$= \sum_{k=1}^{\infty} (L_k^TB)G_b^{k-1}(z) + \sum_{k=1}^{\infty} (L_k^TA)V_k(z). \quad (23)$$

We can now apply equation (16) to the first summation and Proposition 4.1 to the second summation, resulting in the next proposition:

Proposition 4.2

$$\begin{aligned} \widetilde{\overleftarrow{G}}(\lambda) &= \sum_{k=0}^{\infty} \overleftarrow{M}_k \lambda^{-k}, \qquad \textit{where} \\ \overleftarrow{M}_0 &= (L_1^TB)I + [(L_1^TA) \otimes I]\mathcal{P}^T \\ \overleftarrow{M}_k &= (L_{k+1}^TB)I + [(L_{k+1}^TA) \otimes I]\mathcal{P}^T \\ &\quad + [(L_k^TA) \otimes I]\mathcal{Q}^T \end{aligned}$$

□

The latter two Propositions show how the set of expansion coefficients can be efficiently transformed into Markov parameters of the Hambo system transform of the (shifted) transfer function, which immediately results in the Hankel matrices $\widetilde{H}$ and $\widetilde{\overleftarrow{H}}$.
This section will be concluded with establishing the connection between the Hankel matrices in the transform domain $(\widetilde{H}, \widetilde{\overleftarrow{H}})$ and the corresponding Hankel matrices in the standard domain $(H, \overleftarrow{H})$. This connection turns out to be determined by a set of transformation matrices, that are derived from the all-pass function $G_b(z)$.

Proposition 4.3 *Given a system* $G(z)$ *with Hankel matrix* H *and its Hambo transform* $\widetilde{G}(\lambda)$ *with Hankel matrix* $\widetilde{H}$*, then there exist unitary matrices* T_1, T_2 *such that* $H = T_1^{-1}\widetilde{H}T_2 = T_1^T\widetilde{H}T_2$ *where*

$$(T_1)_{ij} = v_i(j)^{\{A,B,C,D\}} \quad (24)$$

$$(T_2)_{ij} = v_i(j)^{\{A^T,C^T,B^T,D^T\}} \quad (25)$$

where $v_i(j)^{\{F,G,H,J\}}$ *is defined by equations (7-9).*

□

This property immediately shows that the rank of the Hankel matrices in both domains are equal and that any full rank decomposition (for instance by singular value decomposition) of $\widetilde{H}$ immediately results in a full rank decomposition of H. This property will be of use in the generalization of the Ho-Kalman realization algorithm, discussed next.

5 Exact realization

In this section the generalization of the Ho-Kalman algorithm will be discussed, generalized in terms of the GOBF basis. It is assumed that the Hankel matrices involved are fully known, i.e. the matrices are of infinite dimension, or equivalently all Markov parameters are known. The classical minimal realization problem can be stated as:

Given a sequence of $p \times m$ constant matrices $\{Y_i\}_{i=1}^{\infty}$, find a triple $\{A, B, C\}$ with A of minimal dimension such that $Y_i = CA^{i-1}B$ for $i \in \mathbb{N}$.

This problem has a solution if and only if there exist integers N', N such that $\text{rank}(H_{N',N}) = \text{rank}(H_{N'+i,N+j}) = n_0$ for all $i, j = 0, 1, 2, \cdots$, where

$$H_{N',N} = \begin{bmatrix} Y_1 & Y_2 & \cdots & Y_N \\ Y_2 & Y_3 & \cdots & Y_{N+1} \\ \vdots & \vdots & & \vdots \\ Y_{N'} & Y_{N'+1} & \cdots & Y_{N+N'-1} \end{bmatrix} \tag{26}$$

and in this case n_0 is the minimal state space dimension. Under the assumption that the unknown system is finite dimensional, a minimal state space realization can be determined using the Ho-Kalman algorithm, also known as Ho's algorithm.

A concise description of the standard Ho-Kalman is given first, for details see Ho and Kalman (1966). The description is in terms of the infinite Hankel matrices rather than in terms of the finite Hankel matrices, to facilitate the derivation of the generalized realization algorithm later in this section.

Algorithm 5.1 *(Ho-Kalman).*
Given Hankel matrices $H, \overleftarrow{H}$, with elements $H_{ij} = g_{i+j-1}$ and $\overleftarrow{H}_{ij} = g_{i+j}$, where $\{g_k\}$ is the set of Markov parameters of a finite dimensional system $G(z)$, the following steps result in a minimal state space realization $\{A_g, B_g, C_g\}$ of $G(z)$:

1. *Let H have a full rank decomposition $H = \Gamma \cdot \Delta$, i.e. rank(Γ)=rank(Δ)=rank(H).*
2. *Then $\overleftarrow{H}$ obeys the relation $\overleftarrow{H} = \Gamma \cdot A_g \cdot \Delta$.*
3. *Hence A_g can be retrieved with $A_g = \Gamma^{+} \cdot \overleftarrow{H} \cdot \Delta^{+}$, with $(\cdot)^{+}$ indicating the Moore-Penrose pseudo-inverse.*
4. *Furthermore B_g and C_g are created from $B_g = \Delta \begin{bmatrix} I_m \\ 0 \end{bmatrix}$ and $C_g = \begin{bmatrix} I_p & 0 \end{bmatrix} \Gamma$.*

□

The extended minimal realization problem addressed here is summarized by:

Given a sequence of $n_b \times 1$ constant matrices $\{L_i\}_{i=0}^{\infty}$, and an orthonormal basis $\{V_k(z)\}_{k=1}^{\infty}$ find a triple $\{A_g, B_g, C_g\}$ with A_g of minimal dimension such that the system $G(z) = C_g[zI - A_g]^{-1}B_g$ obeys an expansion $G(z) = \sum_{k=0}^{\infty} L_k^T V_k(z)$

For the solution of this problem explicit use is made of the transformation property $H = T_1^T \widetilde{H} T_2$ and $\overleftarrow{H} = T_1^T \overleftarrow{\widetilde{H}} T_2$, given by Proposition 4.3. Substituting these relations in Algorithm 5.1 results in a realization algorithm based on the Markov parameters in $\widetilde{H}$ and $\overleftarrow{\widetilde{H}}$. The missing link is then the relation between the expansion coefficients $\{L_k\}$ and these Hankel matrices. That relation is established by Propositions 4.1 and 4.2.

This approach leads to the following algorithm.

Algorithm 5.2 *(Generalized Ho-Kalman).*
Given a series expansion $G(z) = \sum_{k=0}^{\infty} L_k^T V_k(z)$, the following steps result in a minimal state space realization $\{A_{\widetilde{g}}, B_{\widetilde{g}}, C_{\widetilde{g}}\}$ of $G(z)$:

1. *Calculate the Markov parameters $\{M_k\}, \{\overleftarrow{M}_k\}$ of the Hambo system transforms $\widetilde{G}(\lambda), \overleftarrow{\widetilde{G}}(\lambda)$ using Propositions 4.1 and 4.2.*
2. *Create Hankel matrices $\widetilde{H}, \overleftarrow{\widetilde{H}}$, with elements $\widetilde{H}_{ij} = M_{i+j-1}$ and $\overleftarrow{\widetilde{H}}_{ij} = \overleftarrow{M}_{i+j-1}$.*
3. *Let $\widetilde{H}$ have a full rank decomposition $\widetilde{H} = \widetilde{\Gamma} \cdot \widetilde{\Delta}$*
4. *Then $\overleftarrow{\widetilde{H}}$ obeys the relation $\overleftarrow{\widetilde{H}} = \widetilde{\Gamma} \cdot A_{\widetilde{g}} \cdot \widetilde{\Delta}$*
5. *Hence $A_{\widetilde{g}}$ can be retrieved with $A_{\widetilde{g}} = \widetilde{\Gamma}^{+} \cdot \overleftarrow{\widetilde{H}} \cdot \widetilde{\Delta}^{+}$*
6. *Furthermore $B_{\widetilde{g}}$ and $C_{\widetilde{g}}$ are created from $B_{\widetilde{g}} = \widetilde{\Delta} T_2 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $C_{\widetilde{g}} = \begin{bmatrix} 1 & 0 \end{bmatrix} T_1^T \widetilde{\Gamma}$*

□

An important feature here is the fact that only one row c.q. column of the transformation matrices T_1^T, T_2 is required to calculate the state space realization. In general the calculation of $B_{\tilde{g}}$ and $C_{\tilde{g}}$ involves infinite dimensional matrix calculations. In the special case that $\widetilde{G}(\lambda)$ is a finite impulse response model, this will be reduced to finite operations. This occurs when the underlying system $G(z)$ has only a finite number of non-zero expansion coefficients:

$$G(z) = \sum_{k=1}^{N} L_k^T V_k(z) \tag{27}$$

which results with Propositions 4.1 and 4.2 in system transforms

$$\widetilde{G}(\lambda) = \sum_{k=0}^{N} M_k \lambda^{-k} \qquad \widetilde{\overleftarrow{G}}(\lambda) = \sum_{k=0}^{N} \overleftarrow{M}_k \lambda^{-k}. \tag{28}$$

In this case we can decompose $\widetilde{H}$ and $\widetilde{\overleftarrow{H}}$ as

$$\widetilde{H} = \begin{bmatrix} \widetilde{H}_1 & 0 \\ 0 & 0 \end{bmatrix} \qquad \widetilde{\overleftarrow{H}} = \begin{bmatrix} \widetilde{\overleftarrow{H}}_1 & 0 \\ 0 & 0 \end{bmatrix} \tag{29}$$

and the full rank decomposition of $\widetilde{H}$ reduces to:

$$\widetilde{H} = \begin{bmatrix} \widetilde{\Gamma}_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \widetilde{\Delta}_1 & 0 \\ 0 & 0 \end{bmatrix} \tag{30}$$

where $\widetilde{\Gamma}_1, \widetilde{\Delta}_1$ are finite matrices. It follows that the realization $\{A_{\tilde{g}}, B_{\tilde{g}}, C_{\tilde{g}}\}$ can be calculated by

$$A_{\tilde{g}} = \widetilde{\Gamma}_1^{+} \cdot \widetilde{\overleftarrow{H}}_1 \cdot \widetilde{\Delta}_1^{+} \tag{31}$$

$$B_{\tilde{g}} = \widetilde{\Delta}_1 (T_2)_1 \tag{32}$$

$$C_{\tilde{g}} = (T_1^T)_1 \widetilde{\Gamma}_1 \tag{33}$$

where $(T_2)_1$ and $(T_1^T)_1$ are finite submatrices of the transformation matrices T_2 and T_1^T.
This algorithm is basically the same as the algorithm in (Szabó and Bokor, 1997). The main difference is the numerically more efficient calculation of the Markov parameters (step 1) and the transformation matrices T_1, T_2. For numerical examples of this application see (Szabó and Bokor, 1997; Szabó et al., 1998).

6 Partial realization

In this section the generalization of the so-called partial realization problem is described and a solution to this problem is presented. The partial realization problem (Tether, 1970) deals with the case of limited information, i.e. only a finite number of Markov-parameters is known. The problem is the construction of a finite dimensional minimal state space realization that fits these Markov parameters. The *minimal* partial realization problem aims at finding such a realization with a minimal McMillan degree over all possible realizations. The partial realization problem can be solved, using the Ho-Kalman algorithm, without posing any restrictions on the given set of Markov parameters, and it is straightforward to show that also the minimal partial realization problem has at least one solution (Tether, 1970). However, the question when such a solution is unique is more involved. The key property that ensures a unique solution is a rank condition on the finite Hankel matrix, created with the given set of Markov parameters.
The following lemma gives conditions for the existence of a unique minimal partial realization.

Lemma 6.1 *(Tether, 1970)*
Let $\{Y_1, \cdots, Y_{N_0}\}$ be an arbitrary sequence of $p \times m$ matrices and let $H_{i,j}, i+j \leq N_0$, be a corresponding block Hankel matrix. Then a minimal partial realization given by the Ho-Kalman algorithm is unique (modulo similarity transformations) if and only if there exist positive integers N', N such that

1. $N' + N = N_0$
2. $rank(H_{N',N}) = rank(H_{N'+1,N}) = rank(H_{N',N+1})$.

□

Generalizing this property to the GOBF case is more involved than in the exact realization case, because proposition 4.3 is only valid for infinite Hankel matrices and not for finite matrices. The key idea to overcome this problem is to extend these finite matrices to infinite dimensions, using the Ho-Kalman algorithm *in the transform domain.* This will result in (minimal) realizations of the underlying systems $\widetilde{G}(z)$ and $\widetilde{\overleftarrow{G}}(z)$, which are guaranteed to be unique under rank conditions given by lemma 6.1. Since a minimal realization for $\widetilde{G}(z), \widetilde{\overleftarrow{G}}(z)$ automatically gives a full rank decomposition for the associated infinite Hankel matrices in terms of the product of the observability and controllability matrices

$$\widetilde{H} = \widetilde{\Gamma} \cdot \widetilde{\Delta} \qquad \widetilde{\overleftarrow{H}} = \widetilde{\overleftarrow{\Gamma}} \cdot \widetilde{\overleftarrow{\Delta}}$$

we can apply Algorithm 5.1 to obtain

$$\widetilde{\overleftarrow{H}} = \widetilde{\overleftarrow{\Gamma}} \cdot \widetilde{\overleftarrow{\Delta}} = \widetilde{\Gamma} \cdot A_g \cdot \widetilde{\Delta} \rightarrow A_{\tilde{g}} = \widetilde{\Gamma}^{+} \widetilde{\overleftarrow{\Gamma}} \widetilde{\overleftarrow{\Delta}} \widetilde{\Delta}^{+},$$

which expression can be calculated using Sylvester and Lyapunov equations. Along the same line of reasoning the matrices B_g and C_g are derived. This leads to the following algorithm:

Algorithm 6.2 *Generalized Minimal Partial Realization*

Let $\{L_k\}_{k=1}^{N_0}$ be the first N_0 expansion coefficients of a scalar system $G(z)$ and let $\{M_k, \overleftarrow{M}_k\}_{k=0}^{N_0-1}$ be the Markov parameters of the Hambo system transform of $G(z)$ respectively its shift $\overleftarrow{G}(z)$, as defined by Propositions 4.1 and 4.2. Assume that both $\{\overleftarrow{M}_k\}_{k=1}^{N_0-1}$ and $\{M_k\}_{k=1}^{N_0-1}$ satisfy the conditions of Lemma 6.1. Then a unique minimal state space realization $\{A_{\tilde{g}}, B_{\tilde{g}}, C_{\tilde{g}}\}$ of $G(z)$ is obtained as follows:

1. *Use the Ho-Kalman algorithm to create minimal state space realizations $\{\widetilde{A}, \widetilde{B}, \widetilde{C}\}$ and $\{\overleftarrow{\widetilde{A}}, \overleftarrow{\widetilde{B}}, \overleftarrow{\widetilde{C}}\}$, such that for $1 \leq k \leq N_0$*

$$\overleftarrow{M}_k = \overleftarrow{\widetilde{C}}\,\overleftarrow{\widetilde{A}}^{k-1}\,\overleftarrow{\widetilde{B}} \qquad M_k = \widetilde{C}\widetilde{A}^{k-1}\widetilde{B}$$

2. *Observe that the infinite Hankel matrices $\widetilde{H}, \overleftarrow{\widetilde{H}}$, constituted by these realizations have full rank decompositions*

$$\widetilde{H} = \widetilde{\Gamma} \cdot \widetilde{\Delta} := \begin{bmatrix} \widetilde{C} \\ \widetilde{C}\widetilde{A} \\ \vdots \end{bmatrix} \begin{bmatrix} \widetilde{B} & \widetilde{A}\widetilde{B} & \cdots \end{bmatrix}$$

$$\overleftarrow{\widetilde{H}} = \overleftarrow{\widetilde{\Gamma}} \cdot \overleftarrow{\widetilde{\Delta}} := \begin{bmatrix} \overleftarrow{\widetilde{C}} \\ \overleftarrow{\widetilde{C}}\,\overleftarrow{\widetilde{A}} \\ \vdots \end{bmatrix} \begin{bmatrix} \overleftarrow{\widetilde{B}} & \overleftarrow{\widetilde{A}}\,\overleftarrow{\widetilde{B}} & \cdots \end{bmatrix}$$

3. *Apply the generalized Ho-Kalman algorithm 5.2 (step 5 and 6) to derive*

$$A_{\tilde{g}} = \widetilde{\Gamma}^{+}\overleftarrow{\widetilde{\Gamma}}\,\overleftarrow{\widetilde{\Delta}}\,\widetilde{\Delta}^{+} \tag{34}$$

$$= (\widetilde{\Gamma}^T\widetilde{\Gamma})^{-1}(\widetilde{\Gamma}^T\overleftarrow{\widetilde{\Gamma}})(\overleftarrow{\widetilde{\Delta}}\widetilde{\Delta}^T)(\widetilde{\Delta}\widetilde{\Delta}^T)^{-1} \tag{35}$$

$$B_{\tilde{g}} = \widetilde{\Delta} T_2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{36}$$

$$C_{\tilde{g}} = \begin{bmatrix} 1 & 0 \end{bmatrix} T_1^T \widetilde{\Gamma} \tag{37}$$

□

Remark 6.3 *Note that – for the algorithm to work properly – it is not required that the realization $\{\overleftarrow{\widetilde{A}}, \overleftarrow{\widetilde{B}}, \overleftarrow{\widetilde{C}}\}$ is minimal.*

Given this algorithm, the question will arise why and when it would be of use. This will typically be the case in an approximation or identification setting, using an orthonormal basis function approach. One of these situations is where a finite number of expansion coefficients is estimated, i.e. estimation of the parameters $\{L_k\}$ in a model structure

$$y(t) = \sum_{k=1}^{N} L_k^T V_k(q) u(t) + e(t), \tag{38}$$

and the number of significant coefficients L_k is relatively high, such that a direct state space realization would result in a high order model. One approach in this case would be to apply model reduction techniques to obtain a lower order model. However, such a procedure will not make use of the intrinsic information contained in the expansion coefficients or the directly related Markov parameters of the transformed system.

The merit of Algorithm 6.2 is that it makes full use of this information and hence will improve the quality of the resulting approximation. It will furthermore give much more insight in the McMillan degree of the underlying system.

It is important to note that the presented partial realization algorithm intrinsically requires two applications of the standard Ho-Kalman mechanism. Only in the classical case this can be reduced to one application, because there a realization of $\{M_k\}_{k=1}^{N_0}$ immediateley results in a realization of $\{\overleftarrow{M}_k\}_{k=1}^{N_0}$. Also in the generalized case these sets with Markov parameters are obviously closely related and it might be possible to further simplify the algorithm.

7 Approximate realization

The generalized Ho-Kalman algorithms described in the previous sections can be applied in an approximate fashion as is the case with the standard Ho-Kalman algorithm (Zeiger and McEwen, 1974; Kung, 1978). Most commonly the full rank decomposition of $\widetilde{H}$ will be computed by means of a singular value decomposition. One can then simply truncate the SVD by setting the smaller singular values to zero, and proceed as in the non-approximative case. When algorithm 2 is applied in an approximate sense, in the situation where all expansion coefficients are known, the resulting realization will be exactly the same as the one obtained by application of the Ho-Kalman algorithm in the original domain. This is caused by the unitary transformations in Proposition 4.3.

The situation becomes different in the case where only a finite number of Markov parameters is given. Note that in that partial realization case we need only to truncate the SVD of $\widetilde{H}$ and not that of $\overleftarrow{\widetilde{H}}$. The consequences of this method when compared to the standard partial realization algorithm still have to be further explored.

8 Conclusions

The problem has been addressed of constructing a minimal state space realization on the basis of a sequence of expansion coefficients in a generalized orthonormal basis function expansion. The classical Ho-Kalman algorithm, designed for minimal realization on the basis of Markov parameters, has been extended for expansions in a general class of basis functions. Whereas in the classical situation one algorithm essentially solves both the exact (infinite data) and the partial (finite data) problem, in the generalized case different algorithms result. The presented algorithms can also be used for constructing reduced order state space models on the basis of estimated expansion coefficients.

References

Audley, D.R. and W.J. Rugh (1973). On the H-matrix system representation. *IEEE Trans. Autom. Control, 18*, 235-242.

De Vries, D.K. and P.M.J. Van den Hof (1995). Quantification of uncertainty in transfer function estimation: a mixed probabilistic - worst-case approach. *Automatica, 31*, 543-557.

De Vries, D.K. and P.M.J. Van den Hof (1998). Frequency domain identification with generalized orthonormal basis functions. *IEEE Trans. Autom. Control, 43*, 656-669.

Hakvoort, R.G. and P.M.J. Van den Hof (1997). Identification of probabilistic uncertainty regions by explicit evaluation of bias and variance errors. *IEEE Trans. Autom. Control, 42*, 1516-1528.

Heuberger, P.S.C., P.M.J. Van den Hof and O.H. Bosgra (1995). A generalized orthonormal basis for linear dynamical systems. *IEEE Trans. Autom. Control, AC-40*, 451–465.

Heuberger, P.S.C. and P.M.J. Van den Hof (1996). The Hambo transform: a signal and systems transform induced by generalized orthonormal basis functions. In: *Prepr. 13th Triennial IFAC World Congress*, San Francisco, Vol. I, pp. 357-362.

Ho, B.L., and R.E. Kalman (1966). Effective construction of linear state-variable models from input-output functions. *Regelungstechnik, 14*, 545-548.

Kautz, W.H. (1954). Transient synthesis in the time domain. *IRE Trans. Circ. Th., CT-1*, 29–39.

Kung, S. (1978). A new identification and model reduction algorithm via singular value decompositions. *Proc 12th Asilomar Conf. Circuits Syst. and Computers*, Pacific Grove, CA, Nov. 6-8, 1978, 705-714.

Ninness, B.M. and F. Gustafsson (1997). A unifying construction of orthonormal bases for system identification. *IEEE Trans. Autom. Control, AC-42*, 515-521.

Ninness, B. and J.C. Gómez, (1996). Asymptotic analysis of MIMO system estimates by the use of orthonormal basis functions. In: *Prepr. 13th Triennial IFAC World Congress*, San Francisco, Vol. I, pp. 363-368.

Oliveira e Silva, T., (1996). A N-width result for the generalized orthonormal basis function model. In: *Prepr. 13th Triennial IFAC World Congress*, San Francisco, Vol. I, pp. 375-380.

Roberts, R.A. and C.T. Mullis (1987). *Digital Signal Processing*, Addison Wesley Publ. Comp., Reading, Massachusetts.

Schipp, F., Gianone, L., Bokor, J. and Z. Szabó, (1996). Identification in generalized orthogonal basis - a frequency domain approach. In: *Prepr. 13th Triennial IFAC World Congress*, San Francisco, Vol. I, pp. 387-392.

Szabó, Z. and J. Bokor (1997). Minimal state space realization for transfer functions represented by coefficients using generalized orthonormal basis. In: *Proc. 36th IEEE Conf. Decision and Control*, San Diego, CA, 169-174.

Szabó, Z., J. Bokor, P.S.C. Heuberger and P.M.J. Van den Hof (1998). Extended Ho-Kalman algorithm for systems represented in generalized orthonormal bases. Submitted to *Automatica*.

Tether, A.J., (1970). Construction of minimal linear state-variable models from finite input-output data. *IEEE Trans. Autom. Control, AC-15*, 427-436.

Van den Hof, P.M.J., P.S.C. Heuberger and J. Bokor (1995). System identification with generalized orthonormal basis functions. *Automatica, 31*, 1821-1834.

Wahlberg, B. (1991). System identification using Laguerre models. *IEEE Trans. Autom. Control, AC-36*, 551–562.

Wahlberg, B. (1994). System identification using Kautz models. *IEEE Trans. Autom. Control, AC-39*, 1276–1282.

Wahlberg, B. and P.M. Mäkilä (1996). On approximation of stable linear dynamical systems using Laguerre and Kautz functions. *Automatica, 32*, 693-708.

Zeiger, H.P. and J. McEwen (1974). Approximate linear realizations of given dimension via Ho's algorithm. *IEEE Trans. Autom. Control, AC-19*, 153.

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Model predictive control with generalized input parametrization

Edwin T. van Donkelaar[‡], Okko H. Bosgra and Paul M.J. Van den Hof

Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.
E-mail: e.t.vandonkelaar@wbmt.tudelft.nl

Abstract. For certain applications the performance that can be achieved with model predictive control is restricted by the large computational demand of the on-line optimization. For these applications, such as large scale and fast sampled systems, it is important to choose the degrees of freedom in the on-line optimization problem carefully to obtain a satisfactory trade-off between performance and complexity. In this article it is investigated how, alternative to e.g. the standard pulse or blocking mechanisms, other input parametrizations can be used to obtain high performance model predictive control with only a small amount of free variables. An efficient parametrization is obtained using the observation that the class of all solutions to a finite or infinite horizon LQ control problem can be parametrized with a number of free parameters that is equal to the model order, without loss of performance. The inifinite horizon controller with this parametrization is shown to provide a stable closed-loop system, also if constraints are active. The complexity of the parametrization can be systematically reduced using standard LQG-balanced reduction. Constrained stability of the closed-loop system is preserved with this reduction approach. The proposed algorithms are illustrated with a simulation example.

Keywords. Model predictive control, infinite receding horizon control, input parametrization.

1 Introduction

Model predictive control (MPC) or receding horizon control (RHC) is a control strategy where at each time instant the control action is determined by an on-line optimization of a cost function. An optimal input trajectory over a certain horizon in the future is determined of which the first sample is used as actual control input. To obtain a control action for the next time instant this procedure is repeated. This strategy is denoted as receding horizon control strategy. Because at each time instant an optimization problem is solved, restrictions on control inputs and other process variables can be incorporated explicitly by adding a set of constraints to the optimization problem. For an overview of the extensive literature on model predictive control see Garcia and Morari (1989) and Morari and Lee (1997).

The large computational load has restricted the use of MPC to relatively slow processes as often encountered in the process industry. It is still a difficult task to design a high performance model predictive controller for applications where the computation time is a major restriction e.g. in large scale and fast sampled systems.

A design variable with which the optimization complexity can be reduced, is the parametrization of the input trajectory. The location and number of free variables in the input trajectory have a large influence on both the computational load and the performance of the controller. Therefore, a trade-off has to be made between performance and com-

[‡]The research of Edwin van Donkelaar is supported by the Dutch Technology Foundation (STW) under contract DWT55.3618.

plexity. The larger the freedom in the optimization problem, the higher the performance level that can be obtained, but at the cost of a larger computational burden. In this paper it is investigated how the input parametrization can be chosen such that only a small number of parameters is consumed to obtain a high level of performance.

In literature several choices are made for the location of the degrees of freedom in the input trajectory. For finite horizon predictive control the input can be parametrized freely over a prediction horizon (Kwon and Byun, 1989). For this choice the optimization complexity can become very high if a large amount of preview of possible constraint activation is required. A possible way to reduce the number of degrees of freedom is to choose a parametrization of the input in which the first few samples are free and after that are held constant. This approach is followed in many industrial applications of model predictive control such as dynamic matrix control (DMC, Cutler and Ramaker (1980)) and model algorithmic control (MAC, Richalet (1993)). Another way to parametrize the input is by blocking (Ricker *et al.*, 1988). where the input trajectory is built up by a number of blocks over which the input is held constant. In Lee *et al.* (1995) wavelet theory is applied to analyze the concept of blocking. In predictive functional control (PFC, Richalet *et al.* (1987)) the input is parametrized in terms of polynomials and goniometric functions.

For infinite horizon predictive control the input is parametrized freely over the first few samples and thereafter it is assumed to be either fixed to zero (Rawlings and Muske, 1993) or an LQ optimal controller is assumed to be active (Scokaert and Rawlings, 1996). In the latter situation it is shown that infinite horizon LQ optimality can be obtained but possibly at the cost of a considerable computational load. Especially for the class of systems we are focussing on i.e. large scale systems and fast sampled systems, the number of degrees of freedom needed to obtain optimality will typically be high.

With the approaches described above to reduce the number of degrees of freedom and thereby the computational load, a large amount of trial and error is needed to obtain a suitable input parametrization for which a good trade-off between complexity and performance is achieved. Also the performance loss can be high if only a few degrees of freedom are allowed in the optimization problem.

In this article it is investigated how other parametrization of the input trajectory for model predictive controllers can be used to improve the trade-off between performance and complexity. An approach is discussed to determine a parametrization such that only a small number of parameters is consumed to obtain a closed-loop system that is equivalent to finite or infinite horizon LQ optimal control in the unconstrained case. The input trajectory is parametrized in terms of an expansion in basis functions which can be calculated a priori. The number of applied basis functions, and hence the optimization complexity, can be reduced systematically with a standard model reduction tool, namely LQG balanced reduction. This reduction method provides the user with valuable information about how much the number of basis functions can be reduced, such that the decrease in unconstrained performance level is small. This provides a tool to make the trade-off between unconstrained performance and complexity in a more systematic way.

In section 2 the problem is specified in mathematical terms. In section 3 an approach is presented to choose the degrees of freedom in finite and infinite horizon model predictive control. In section 4 this procedure is applied to an infinite horizon LQ criterion with constraints. In section 5 a systematic procedure for the reduction of the number of free variables is described. In section 6 the ideas are illustrated on simulation examples. Section 7 concludes the paper.

2 Model predictive control

Linear model predictive control or receding horizon control provides a solution to the problem of constrained control of systems. Let the system be given by the state-space description

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \quad x(0) = x_0 \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector and $u(t) \in \mathbb{R}^{n_u}$ the input vector at time t. The matrix A has eigenvalues strictly inside the unit disc, $|\lambda\{A\}| < 1$ and the pair (A, B) is controllable and (C, A) is observable. The aim is to control this system while satisfying constraints on the input and state variables

$$\begin{aligned} K_u u(t) &< k_u \text{ for all } t \\ K_x x(t, x_0) &< k_x \text{ for all } t \end{aligned} \tag{2}$$

where $K_u \in \mathbb{R}^{n_u \times n_u}, K_u \in \mathbb{R}^{n \times n}$ are the matrices that specify the input and state constraints. These matrices can have a different row dimension than specified and can be time-varying but are specified in this way for convenience of notation.

The most commonly used cost function in model predictive control is the quadratic cost function given by

$$J(u(\cdot)) = \sum_{t=0}^{P-1} \{x^T(t) Q_1 x(t) + u^T(t) Q_2 u(t)\} + \tag{3}$$

$$+x^T(P)Q_0x(P).$$

with the weighting matrices $Q_1, Q_2 \geq 0$ and the pair $\{Q_1, A\}$ is detectable. The optimization problem that has to be solved on-line in model predictive control is given by the quadratic programming problem

$$\min_U \{U^T HU + 2x_0^T g^T U\} \tag{4}$$

subject to $K_U U \leq k_U, K_X U \leq k_X(x_0)$,
with $U^T = [u^T(0)\, u^T(1) \ldots u^T(P-1)]$ and the Hessian $H = R + G^T QG$ and $g^T = H_x^T QG$ where

$$G = \begin{bmatrix} 0 & & \cdots & 0 \\ B & 0 & \ddots & \\ AB & B & \ddots & \vdots \\ \vdots & \ddots & \ddots & \\ & & & 0 \\ A^{T-1}B & A^{T-2}B & \cdots & AB \;\; B \end{bmatrix}, H_x = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^T \end{bmatrix} \tag{5}$$

where $G \in \mathbb{R}^{Pn \times Pn_u}$ and $H_x \in \mathbb{R}^{Tn \times n}$,

$$Q = diag(Q_1, \ldots, Q_1, Q_0) \in \mathbb{R}^{(P+1)n \times (P+1)n} \text{ and}$$

$$R = diag(Q_2, \ldots, Q_2) \in \mathbb{R}^{Pn_u \times Pn_u} \tag{6}$$

$$\begin{aligned} K_U &= diag(K_u, \ldots, K_u) \in \mathbb{R}^{n_{cu} \times Pn} \text{ and} \\ k_U &= \begin{bmatrix} k_u^T & \ldots & k_u^T \end{bmatrix}^T \mathbb{R}^{n_{cu}} \\ K_X &= diag(K_x, \ldots, K_x)G \in \mathbb{R}^{n_{cx} \times Pn} \text{ and} \\ k_X(x_0) &= \begin{bmatrix} k_x^T & \ldots & k_x^T \end{bmatrix}^T - K_X H_x x_0 \in \mathbb{R}^{n_{cx}} \end{aligned}$$

The cost function that is minimized in (4) is defined as $J(U, x_0)$.
At each time instant the value x_0 in the quadratic program is updated with the measured value of the state (full information case) or a prediction of the current state (partial information case). In this way at each time instant an updated quadratic program is specified where possibly another set of constraints is active.
The input trajectory U can be parametrized in various ways. A general way to describe the parametrization of U is such that the class of obtainable input trajectories is a subspace of $\mathbb{R}^{Pn_u}$ which has a (considerably) lower dimension than the full space. This can be described by

$$U(\theta) = \phi\theta \text{ with } \phi \in \mathbb{R}^{Pn_u \times n_\theta}, \theta \in \mathbb{R}^{n_\theta} \tag{7}$$

where θ is the free parameter and ϕ is a user-chosen matrix of which the columns form a basis for the space of all input trajectories that can be achieved. For each input sample over the horizon this parametrization is described by

$$u(t, \theta) = \varphi^T(t)\theta \text{ with } \varphi^T(t) \in \mathbb{R}^{n_u \times n_\theta}, \theta \in \mathbb{R}^{n_\theta}$$

where n_θ is the number of free variables in the optimization. The input parametrization above includes the conventional input parametrization and parametrization using blocking. These parametrizations boil down to respectively:

$$\phi_{pulse} = \begin{bmatrix} I & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & & I \\ & & \ddots & \vdots \\ 0 & \cdots & 0 & I \end{bmatrix},$$

$$\phi_{block} = \begin{bmatrix} I & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ I & 0 & \cdots & 0 \\ 0 & I & & \vdots \\ \vdots & \vdots & & 0 \\ & I & & 0 \\ & 0 & & I \\ & & & \vdots \\ 0 & 0 & 0 & I \end{bmatrix} \in \mathbb{R}^{Pn_u \times Mn_u}$$

where M is the control horizon. With the input space parametrized by the column span of ϕ, the optimization problem that has to be solved is, similar to conventional MPC: a quadratic programming problem. This is given by $\theta^* = arg\min_\theta J(\theta)$ with

$$J(\theta) = \theta^T \phi^T H \phi\theta + 2x_0^T g^T \phi\theta \tag{8}$$

subject to the constraints

$$K_U\phi\theta \leq k_U, K_X\phi\theta \leq k_X(x_0)$$

which is linear in the free parameter θ.
The matrix ϕ is clearly a design variable that is able to influence the controlled behaviour in a crucial way. The main question is, how this freedom can be used in a structured way to obtain a clear trade-off between closed-loop performance and optimization complexity.

3 Finding a suitable input parametrization

From a conceptual point of the view, the input parametrization problem, as stated in the previous section, is the following. Let the system to

be controlled be given by $\mathcal{G}$. The input of the system is given by $u(t) \in \mathbb{R}^{n_u}$ and the state by $x(t) \in \mathbb{R}^n$. The inputs and states are constrained by $K_u u(t) < k_u$ and $K_x x(t) < k_x$ respectively. At this point the system can be either linear or nonlinear but it is assumed to be discrete time. Let the on-line optimization problem for receding horizon control be given by

$$\inf_{U \in \mathcal{U}} J(\mathcal{G}, U, x_0) \tag{9}$$

where U is the input trajectory which is an element of some set $\mathcal{U}$ which is totally specified by the linear input and state constraints imposed on the system, $J(\cdot)$ is a cost function that reflects the desired performance and x_0 is some variable that is a function of the measurement with which the optimization problem is initialized at each time instant t. The variable x_0 can be either a measured state (full information state feedback) or a predicted state (partial information state feedback) or the measured output (general dynamic output feedback).
The input parametrization problem is to find a suitable subspace $\mathcal{U}_r$ with a lower dimension than the full space $\mathbb{R}^{n_u P}$ such that the solution to the reduced order optimization problem

$$\inf_{U \in \mathcal{U} \cap \mathcal{U}_r} J(\mathcal{G}, U, x_0) \tag{10}$$

provides *good control* for all possible measurements x_0.
A possible way to quantify *good control* is by making the reduced problem deviate as little as possible from the original problem. A possible way to choose a space $\mathcal{U}_r$ of dimension p is then by solving

$$\min_{\mathcal{U}_p, dim(\mathcal{U}_p) \leq p} V(\mathcal{U}_p) \tag{11}$$

where $V(\cdot)$ reflects the deviation of the reduced order problem to the full order problem. Possible cost functions are

- minimize the worst-case deviation in value of the cost function

  $$V_1(\mathcal{U}_p) = \sup_{x_0 \in \mathcal{X}} \left(\min_{U \in \mathcal{U} \cap \mathcal{U}_p} J(U, x_0) - \min_{U \in \mathcal{U}} J(U, x_0) \right) \tag{12}$$

  where the measurement x_0 varies over some set $\mathcal{X}$.

- minimize the worst-case deviation between the optimal input trajectories

  $$V_2(\mathcal{U}_p) = \sup_{x_0 \in \mathcal{X}} \left\| arg \min_{U \in \mathcal{U} \cap \mathcal{U}_p} J(U, x_0) - arg \min_{U \in \mathcal{U}} J(U, x_0) \right\|_\gamma \tag{13}$$

  where this deviation is measured in some norm $\|\cdot\|_\gamma$.

- minimize the worst-case difference between the first samples of the optimal input trajectories

  $$V_3(\mathcal{U}_p) = \sup_{x_0 \in \mathcal{X}} \|[I_{n_u}\ 0 \cdots 0](U_r^*(\mathcal{U}_p, x_0) - U^*(x_0))\|_\gamma \tag{14}$$

  with $U^*(\mathcal{U}_p, x_0) = arg \min_{U \in \mathcal{U} \cap \mathcal{U}_p} J(U, x_0)$ and $U^*(x_0) = arg \min_{U \in \mathcal{U}} J(U, x_0)$. This function seems suitable because only the first sample of the input trajectory is actually applied in a receding horizon strategy.

The optimization problems discussed above are related to n-width problems (Pinkus, 1985) where an optimal n-dimensional subspace is calculated by an optimization of a cost function with a free variable that is a set of a certain dimension. Unfortunately, the problems stated above are as such untractable. This is mainly because the cost functions must be evaluated for all possible sets of active constraints which is combinatorial in size. Therefore, only in specific cases a solution can be found.
The first case in which a solution can be found to the problems stated above is the *unconstrained case*. In this case the optimal control profiles are generated by a time-varying state feedback which can be calculated a priori. The only thing that cannot be computed a priori is the state with which the optimization problem is initialized. This simple observation is reflected in the following lemma for the linear quadratic cost function from (4).

Lemma 3.1 *Consider the quadratic cost function given by* (4) *with finite horizon P and let there be no constraints. Then, the subspace*

$$\mathcal{U}_r = im\{H^{-1} g\}$$

with $H \in \mathbb{R}^{Pn_u \times Pn_u}, g \in \mathbb{R}^{Pn_u \times n}$ given in (4) *is a solution of* (11) *with any of the cost functions* (12), (13) *or* (14) *and $\mathcal{X} = \mathbb{R}^n$.*

Proof: The lemma follows from the well known result that the unconstrained solution to (4) is given by $U(x_0) = H^{-1} g x_0$. □

Apparently if one wants to find the unconstrained optimum for all possible initialisations x_0, a search over an n-dimensional subspace is sufficient instead of a Pn_u-dimensional one.
Theorem 3.1 holds for a finite horizon criterion but a similar simple result also holds for the infinite horizon case.

Lemma 3.2 *Consider the quadratic cost function given by* (4) *with infinite horizon* $P = \infty$ *and let there be no constraints. Then, the subspace*

$$\mathcal{U}_r = im\{ \begin{bmatrix} F \\ F(A-BF) \\ F(A-BF)^2 \\ \vdots \end{bmatrix} \} \subset l_2^n[0,\infty), \qquad (15)$$

with $\{A, B\}$ *state space matrices given in* (1) *and* F *the LQ-optimal state feedback given by* $F = (B^TXB+Q_2)^{-1}B^TXA$ *with* X *the unique nonnegative definite solution of the Algebraic Riccati Equation*

$$X = A^T[X + XB(B^TXB + Q_2)^{-1}B^TX]A + Q_1,$$

is a solution of (11) *with any of the cost functions* (12), (13) *or* (14) *and* $\mathcal{X} = \mathbb{R}^n$.

Proof: The unconstrained solution to the problem (4) with infinite horizons is the LQ optimal control profile given by

$$u(t, x_0) = F(A-BF)^t x_0, \quad t = 0, 1, 2, \ldots$$

which directly shows the result. □

The lemma above indicates that an efficient input parametrization for infinite horizon model predictive control is generated by a dynamical system $\{F, A-BF\}$. With this parametrization the infinite dimensional optimization problem is reduced to a finite dimensional optimization problem with a number of free variables that is equal to the model order.

4 Model predictive control with input parametrization

In this section it is described how the input parametrization for infinite receding horizon control discussed in the previous section can be aplied in an efficient model predictive control algorithm. The properties of this algorithm are investigated where special attention is paid to nominal performance and constrained closed-loop stability.

4.1 Infinite horizon model predictive control

If input and state constraints are incorporated in infinite receding horizon control, a finite parametrization of the input space is needed to obtain a finite dimensional constrained optimization problem. In the introduction several approaches to this problem that can be found in the literature are discussed. In this section the parametrization of lemma 3.1 is applied to obtain an alternative parametrization. The closed-loop behaviour of the resulting controller is equivalent to LQ optimal state feedback control if no contraints are active. This is obtained with only n degrees of freedom in the optimization problem, where n is the model order. This controller and some of its properties is given in the following proposition.

Proposition 4.1 *Let a linear discrete-time system be given by* (1) *subject to input and state constraints* (2). *Let the receding horizon controller cost function be given by* (3) *with* $P = \infty$ *and let* x_0 *be either the measured state vector (full information case) or a prediction thereof (partial information case). Let the input over the infinite horizon be parametrized as* $u(t, \theta) = F(A-BF)^t\theta$ *and let*

$$\tilde{A} = \begin{bmatrix} A-BF & 0 \\ BF & A \end{bmatrix} \text{ and } \tilde{C} = \begin{bmatrix} \sqrt{R}F & 0 \\ 0 & \sqrt{Q}C_z \end{bmatrix}.$$

Then

1. *the optimal control input* $u^*(0)$ *is given by* $u^*(0) = F\theta^*$ *with* θ^* *the solution to the finite dimensional quadratic programming problem*

$$\min_{\theta \in \mathbb{R}^n} \begin{bmatrix} \theta^T & x_0^T \end{bmatrix} Y \begin{bmatrix} \theta \\ x_0 \end{bmatrix} \qquad (16)$$

subject to: $K_u F(A-BF)^t\theta < k_u$ *and*

$$\begin{bmatrix} 0 & K_x \end{bmatrix} \tilde{A}^t \begin{bmatrix} \theta \\ x_0 \end{bmatrix} < k_x, t = 0, 1, \ldots, N_c$$

where N_c *is the constraint horizon that is chosen such that after this time instant no constraints are active and* Y *is the solution of the Lyapunov equation*

$$\tilde{A}^TY\tilde{A} + \tilde{C}^T\tilde{C} = Y$$

2. *if no constraints are active this controller is equivalent to LQ control with state feedback* F.

Proof: The input and state trajectories over the infinite horizon are given by

$$\begin{bmatrix} u(t,\theta) \\ x(t,\theta,x_0) \end{bmatrix} = \tilde{A}^t \begin{bmatrix} \theta \\ x_0 \end{bmatrix} \qquad (17)$$

Substituting this in the cost function yields

$$J(\theta, x_0) = \begin{bmatrix} \theta^T & x_0^T \end{bmatrix} \left(\sum_{t=0}^{\infty} \tilde{A}^{Tt}\tilde{C}^T\tilde{C}\tilde{A}t \right) \begin{bmatrix} \theta \\ x_0 \end{bmatrix}$$

The matrix in this expression can be calculated with the Lyapunov equation in statement 1. Statement

2 can be proven by the fact that the unconstrained optimal solution is given by $\theta = -x_0$ which is the state measurement or prediction. From the description (17) it follows that this yields a state feedback control with state feedback F which is LQ-optimal. □

As described in the proposition the quadratic programming problem is either initialized with a state measurement or a state estimate. The latter is more likely because usually measurement of all the state variables is too costly or even impossible. It is well known in literature that an LQ optimal state feedback combined with a Kalman filter gives an LQG controller (Anderson and Moore (1989) and Kwakernaak and Sivan (1972)). This dynamic output feedback controller is an optimal controller with respect to white noise disturbances on the outputs and states with known covariance matrices. With the approach discussed in this section the optimization problem can be built up by solving one Riccati equation for the solution of the LQ control problem and a Lyapunov equation to specify the cost function. This can be done quickly because good software tools are available for solving Riccati and Lyapunov equations, also for large scale problems. Therefore, the procedure is flexible for on-line changes in the internal model, the parametrization and the controller cost function. This flexibility of the proposed method can be a favourable property for constrained control of nonlinear systems with switching linear predictive controllers such as nonlinear quadratic dynamic matrix control (NLQDMC, Garcia and Morari (1989)).
Another property is that the tuning of the proposed algorithm is simple. A standard LQG control design is needed. The only additional choice that has to be made is the number of samples in the future over which the constraints are evaluated. This is given by the constraint horizon N_c which must be chosen such that possible constraint activation can be detected sufficiently long in advance. The parameter N_c is no tuning variable for nominal unconstrained performance as it has no influence on the closed-loop performance. The contraints horizon can also be chosen automatically as in Rawlings and Muske (1993).

4.2 Nominal stability under constraints

In this section constrained stability properties are analyzed of the controller described in the previous section. In the next proposition it is proven that under mild conditions the controller provides a stable closed loop, also if constraints are active.

Proposition 4.2 *The predictive control strategy given in proposition 4.1 is globally asymptotically stable if and only if the optimization problem* (16) *is feasible*

Proof: First consider the full information case. Global time index is denoted with t and the local time index within the optimization is denoted with k. Let the input trajectory $u_t^*(k) = -F(A-BF)^k\theta_t^*$ be a feasible but possibly not optimal solution at time t. Let the corresponding cost be given by $J(t)$. The first sample of this trajectory is applied as current input $u(t) = F\theta^*$. This yields a state $x(t)$ which is equal to the predicted state if no disturbances are present and the model and plant are equal. Then a feasible trajectory for $t+1$ is given by $u_{t+1}^*(k) = -F(A-BF)^k\theta_{t+1}^*$ with $\theta_{t+1}^* = (A-BF)\theta_t^*$ as this is equivalent with the previous trajectory without the first sample. Denote the corresponding cost function with $J(t+1)$. This performance cost level need not be optimal therefore it holds that

$$J(t+1) \leq J(t) - x^T(t)Q_1x(t) - u^T(t)Q_2u(t)$$

Because $Q_1, Q_2 \geq 0$ the sequence $J(t)$ is non increasing. It is bounded from below by zero and therefore $J(t)$ converges to zero, hence $x(t), u(t)$ also converge to zero. Therefore the nonlinear state feedback is stabilizing. Due to the separation principle this stabilizing state feedback combined with a stable observer yields a stabilizing dynamic output feedback (Zheng and Morari, 1995). □

The proposition implies that also in the presence of constraints the closed loop system remains stable if and only if the optimization problem is feasible. Feasibility can only be lost if hard state (output) constraints are used: only then is it possible that there is no input trajectory in the set of feasible input trajectories that renders the state (output) inside the feasible set of states (outputs). In Scokaert and Rawlings (1996a) it is described how the problem of feasibility can be avoided. Often applied methods are constraint softening (Zheng and Morari, 1995) or discarding constraints that are not crucial until the problem becomes feasible (Froisy, 1994). In both cases the proposed algorithm is also stabilizing in the presence of constraints.

5 Systematic reduction of the complexity

In this section it is described how the complexity of the on-line optimization of the observer-based state feedback described in the previous section can be reduced while keeping track of the performance loss.

This is done by choosing the parametrization of input in terms of a linear combination of profiles that have the largest contribution to the cost function. If the complexity must be reduced, it is possible to base the system-based input parametrization on a reduced order model i.e.

$$u(t,\theta) = F_r(A_r - B_r F_r)^t \theta \tag{18}$$

where $\{A_r, B_r\}$ are state-space matrices for the reduced order system and F_r is the LQ-optimal state feedback for this reduced order model.
The model reduction algorithm that is applied should be such that the reduced order basis functions have the largest contribution to the cost function and the functions that are discarded have the smallest. For LQG control this can be done with LQG-balanced reduction Jonckheere and Silverman (1983). With this reduction technique first a similarity transformation is applied on the state space system that provides a coordinate system in which each state is equally well controllable with an LQ-controller as it can estimated with a Kalman filter. This similarity transformation is obtained by forcing the solution of the control discrete algebraic Riccati equation (CDARE) and the filter discrete algebraic Riccati equation (FDARE) to be equal and diagonal. These equations are given respectively by

$$\begin{aligned} X &= A^T X A - A^T X B(B^T X B + I)^{-1} B^T X A \\ &\quad + C^T C \qquad (19) \\ Y &= A Y A^T - A Y C^T (C X C^T + I)^{-1} C Y A^T \\ &\quad + B^T B. \qquad (20) \end{aligned}$$

In Jonckheere and Silverman (1983) it is proven that there exists such a similarity transformation in the continuous-time case. The same transformation holds for the discrete time case as is discussed in the next proposition.

Proposition 5.1 *(Jonckheere and Silverman, 1983) Let a linear time-invariant discrete-time system be given by* (1) *and let the positive definite solutions to the Riccati equations* (19),(20) *be* $P, Q > 0$. *Let* T *be given by*

$$T^{-1} = R^T U \Sigma^{-\frac{1}{2}}$$

with a Cholesky decomposition $R^*R = Y$ *and the eigenvalue decomposition* $RXR^* = U\Sigma^2 U^*$ *where* $U^*U = I$ *and* $\Sigma = diag\{\sigma_1, \ldots, \sigma_n\}$.
Then the transformed system $\{\tilde{A}, \tilde{B}, \tilde{C}\} = \{TAT^{-1}, TB, CT^{-1}\}$ *satisfies the to the Riccati equations* (19) *and* (20) *with*

$$\tilde{X} = \tilde{Y} = \Sigma$$

Proof: The proof follows from the fact that the solutions to the Riccati equations for the transformed system are given by

$$\begin{aligned} \tilde{X} &= TXT^* = \Sigma^{-\frac{1}{2}} U^* R P R^* U \Sigma^{-\frac{1}{2}} = \Sigma \\ \tilde{Y} &= (T^*)^{-1} Y T^{-1} = \Sigma^{\frac{1}{2}} U^* (R^*)^{-1} (R^* R) R^{-1} U \Sigma^{\frac{1}{2}} = \Sigma \end{aligned}$$

For a full proof see Jonckheere and Silverman (1983). □

The states that have a "small" corresponding value on the diagonal of Σ, are both "easy" to filter and have a low contribution to the controller cost function. These can be discarded if the order of the controller is to be reduced. The states that have a "large" corresponding value on the diagonal of Σ, are both "difficult" to filter and are essential states to control and must certainly be accounted for in a reduced order controller. The diagonal elements are invariants for linear systems and can be used to decide on the reduction order in a similar manner as Hankel singular values are used in balanced reduction. In this way optimization complexity can be traded-off against nominal performance in a more quantitative way.
Let the model in LQG-balanced form be given by $\{A, B, C, D\}$ and the reduction order is given by n_r. Then the resulting reduced order model can simply be obtained by

$$\begin{aligned} A_r &= [I_{n_r}\ 0] A [I_{n_r}\ 0]^T, \quad B_r = [I_{n_r}\ 0] B, \\ C_r &= C [I_{n_r}\ 0]^T, \qquad\qquad D_r = D. \end{aligned}$$

Then the reduced order input parametrization can be constructed with (18). The input parametrization based on the reduced order model is again generated by a stable dynamic system because the LQ-optimal state feedback is guaranteed to be stabilizing. Due to this fact and stability of the system it can be proven that for any reduction order the receding horizon controller is stable also in the constrained case. This is given in the next corollary.

Corollary 5.2 *The predictive control strategy given in proposition 4.1 with input parametrization generated by a stable dynamic system* $\{C_p, A_p\}$ *following*

$$u(t,\theta) = C_p A_p^t \theta$$

is globally asymptotically stable if and only if the optimization problem (16) *is feasible.*

Proof: Along identical lines as the proof of proposition (4.1) only with $\{C_p, A_p\}$ instead of $\{F, A-BF\}$. □

Note that this theorem implies that constrained closed-loop stability with the proposed controller is preserved if the input is parametrized with any stable system. However, the model that is applied for the prediction is still equal to the plant.

6 Simulation example

In this section two simulation examples are given to demonstrate the properties of the proposed approach, denoted with MPC_{ip}, compared to model predictive control with a finite prediction and control horizon, denoted with MPC. The first simulation example is a scalar system where a large number of free variables is needed with a conventional pulse parametrization to obtain satisfactory performance while with the proposed algorithm only a few. The second example is multivariable and is used to show the possibilities of the reduction method of section 5.

Simulation example 1

The first system that is considered is a highly oscillatory nonminimum-phase system given by the transfer function

$$G(z) = \frac{-5.7980z^3 + 19.5128z^2 - 21.6452z + 7.9547}{z^4 - 3.0228z^3 3.8630z^2 - 2.6426z + 0.8084}$$

The open-loop step response is given in figure 1.

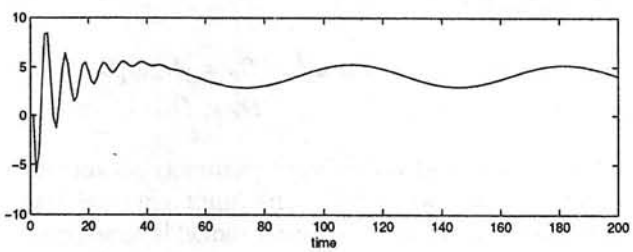


Fig. 1: *Step response of the plant*

If conventional MPC is applied to this system it is necessary to take a prediction horizon that is long enough to incorporate at least one full period, i.e. P=100. Also the choice of the control horizon is critical for this system. A control horizon which is equal to the prediction horizon gives good performance. However, decreasing the degrees of freedom easily gives bad performance as can be seen from figure 6. Reduction of the control horizon upto M=75 is possible without considerable loss of performance, further reduction gives bad performance due to the slow oscillation. Due to the long prediction and control horizon the computational burden is large for MPC. With the approach presented in this article, MPC_{ip}, the number of degrees of freedom is equal to the model order, i.e. n=4. This yields a controlled performance, depicted in figure 6, that is practically identical to the fully parametrized controller only at a much lower computational cost. Also if constraints are active MPC_{ip} performs better than conventional MPC with the same number of degrees of freedom as can be seen from figure 6. It performs slightly less than the fully parametrized MPC but at a much lower computational burden.

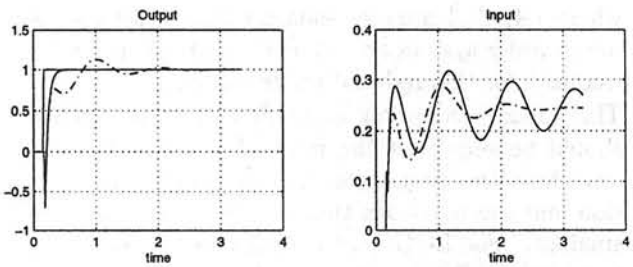


Fig. 2: *Closed-loop step response conventional MPC with M=P=100 (solid) and P=100,M=4 (dash dotted) and MPC_{ip} (dashed) with 4 degrees of freedom which coincides with the solid line.*

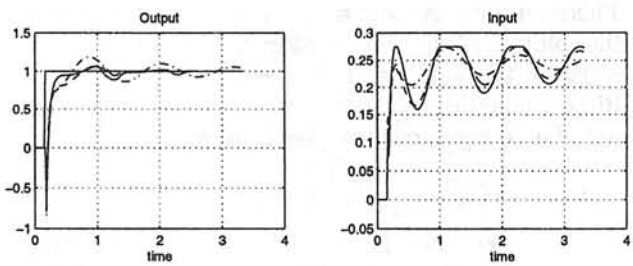


Fig. 3: *Constrained closed-loop step response conventional MPC with M=P=100 (solid) and P=100,M=4 (dash dotted) and MPC_{ip} (dashed) with 4 degrees of freedom.*

To give an indication of the computational load, on a Pentium 233 MHz computer the simulation of 200 time samples cost 511.83 seconds for fully parametrized conventional MPC and only 10.885 seconds for MPC_{ip}. To illustrate the idea of using basis functions to build up the space of allowable input trajectories, the four basis functions are plotted in figure 2.

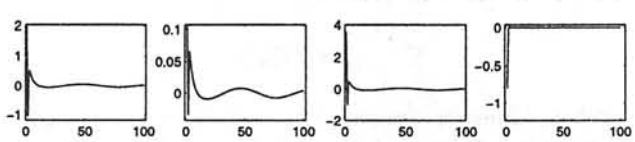


Fig. 2: *Four basis functions used in MPC_{ip}.*

Simulation example 2

The second system that is considered is a 4 input 4 output subsystem of the nonlinear simulation model

of a fluidized bed catalytic cracking unit (FCCU) given in McFarlane *et al.* (1993). A detailed flowsheet of the process can be found in that article. The fluidized catalytic cracking (FCC) process is used to crack a blend of oil products with a high boiling point into lighter and more valuable components such as gasoline. The overall economic performance of an oil refinery largely depends on the economic operation of the FCC unit (Tatrai *et al.*, 1994). Therefore accurate modelling and control of this process is important.

The four inputs are the fresh feed F_3, the slurry recycle F_4, reactor/regenerator differential pressure ΔP and lift air blower setpoint V_{lift}. The four outputs are the regenerator temperature T_{reg}, reactor temperature T_r, the oxygen concentration in the stack gas outlet of the regenerator C_{O_2} and the reactor stand-pipe level l_{sp}. The system has large interaction, a combination of fast and slow dynamical phenomena and it is nonlinear. A linear model of this system is obtained that has order 8 that is accurate around the working condition.

Model predictive control with the proposed input parametrizations from lemma 3.1 and 3.2 are applied. In both cases the number of degrees of freedom is 8. To detect constraint activation well in advance the constraint horizon is chosen $N_c = 100$, which large because of the slow dynamics in the system. For comparison a model predictive controller is designed with a pulse parametrization with the same number of degrees of freedom, i.e. $M = 2$. The prediction horizon is taken to be $P = 100$. Both controllers are tuned with tuning parameters $Q = I, R = I$. In this comparison the complexity of the online optimization for MPC_{ip} and MPC is identical. The unconstrained closed-loop behaviour is tested by a step signal on the reference of the first output. The simulation result is depicted in figure 6.

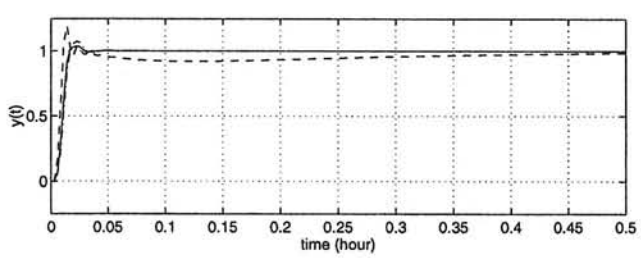


Fig. 4: *Controlled output T_{reg} on a step on the reference signal of 2° C. The applied controllers, with the same number of degrees of freedom, are MPC (dashed), MPC_{ip} with infinite prediction horizon (solid) and with finite horizon (dashdotted).*

For this example the MPC_{ip}-controller is better able to deal with the combination of fast and slow dynamics than MPC with a pulse parametrization

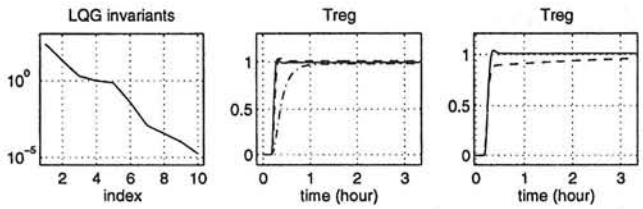


Fig. 5: *Left: LQG invariants of the model. Middle: the controlled output T_{reg} on a step on the reference signal of 2° C; unreduced MPC_{ip} of order 10 (solid), reduced to order 5 (dashed) and reduced to order 4 (dash dotted). Right: MPC with the input parametrization proposed in this paper, reduced to 5 degrees of freedom (solid) and MPC with conventional parametrization with 5 degrees of freedom (dashed).*

with the same number of degrees of freedom. This is because the latter has a higher overshoot as well as undershoot.

The procedure of section 5 is applied to assess whether the complexity can be reduced without significant loss of performance. For this purpose the LQG invariants of the model are determined. These are plotted in figure 6.

From this figure it can be seen that upto order 5 the LQG invariants are larger or equal than one and are significantly smaller for higher orders. Therefore the input parametrization can be reduced to order 5 without considerable loss in closed-loop performance. The closed-loop step response for the full order and reduced order sitution is also given in figure 6. This figure indicates that the loss in performance is indeed small. Further reduction to order 4 shows a large loss in performance which is in accordance with the LQG invariants. Hence, these invariants give a good indication of the smallest number of free variables that is needed to obtain good unconstrained performance. In figure 6 also a comparison is made with MPC_{ip} and MPC with the same number of degrees of freedom. From this is becomes clear that the loss of unconstrained perormance of the former is much smaller than for the latter parametrization.

Constraints on the input and state variables can be accounted for but may have a large influence on the closed-loop performance because the input parametrization is based on unconstrained considerations. It is a topic of current research to choose the basis functions such that more robustness for active constraints can be obtained.

7 Conclusions

In this article it is investigated how, alternative to e.g. the standard pulse or blocking mechanisms, other input parametrizations can be used to obtain high performance model predictive control with only a small amount of free variables. An efficient parametrization is obtained using the observation that the class of all solutions to a finite or infinite horizon LQ control problem can be parametrized with a number of free parameters that is equal to the model order, without loss of unconstrained performance.

The algortihms are efficient and relatively easy to tune because only the horizon over which the constraints are accounted for needs to be chosen and no control horizon. Furthermore, the choice for the prediction horizon has no influence on the unconstrained performance therefore only a standard LQ design procedure is needed. It is also shown that the infinite receding horizon controller with the proposed parametrization provides closed-loop stability, also if constraints are active.

For an infinite horizon LQ cost function a systematic way is discussed to further reduce the number of free variables in such a way that the loss in performance can be assessed a priori. This reduction procedure utilizes an LQG-balanced realization of the model to select the basis functions that have the largest influence on the value of the cost function. The smallest number of basis functions that is needed to obtain approximately the unreduced performance can conveniently be read from the LQG invariants of the model. The closed-loop system with the infinite horizon controller with the reduced order parametrization is also shown to be stable in both the constrained and unconstrained case.

In the present algorithm the choice of the basis is independent of the constraints. Hence, the more the constraints play a role in the control problem, the more performance is lost compared to a free parametrization. It is therefore a topic of current research to choose the basis functions such that more robustness of the parametrization for active constraints is obtained.

References

Anderson, B. and J. Moore (1989). *Optimal Control, Linear Quadratic Methods.* Prentice-Hall Inc., Englewood Cliffs, NJ.

Cutler, C. and B. Ramaker (1980). Dynamic matrix control, a computer algorithm. *Proc. Joint Automatic Control Conf.*, paper WP5-B.

Froisy, J. (1994). Model predictive control: past, present and future. *ISA Transactions, 22*, 235-243.

Garcia, C. and M. Morari (1989). Model predictive control: theory and practice, a survey. *Automatica, 25*, 335-348.

Jonckheere, E. and L. Silverman (1983). A new set of invariants for linear systems, application to reduced order compensator design. *IEEE Trans. Autom. Control, AC-28*, 953-964.

Kwakernaak, H. and R. Sivan (1972). *Linear Optimal Control Systems.* Wiley-Intersc., New York.

Kwon, W. and D. Byun (1989). Receding horizon tracking control as a predictive control and its stability properties. *Int. J. Control, 50*, 1807-1824.

Lee, J., Y. Chikkula and Z. Yu (1995). Improving computational efficiency of model predictive control algorithms using wavelet transformation. *Int. J. Control, 61*, 859-883.

Mc Farlane, R., R. Reineman, J. Bartee and C. Georgakis (1993). Dynamic simulator for a model IV fluid catalytic cracking unit. *Computers Chem. Engng., 17*, 275-300.

Morari, M. and J, Lee (1997). Model predictive control: past, present and future. *Proc. PSE'97 ESCAPE Symposium*, Trondheim, Norway, pp. 1-12.

Pinkus, A. (1985). *n-Widths in Approximation Theory*, Springer Verlag, Berlin.

Rawlings, J. and K. Muske (1993). The stability of constrained receding horizon control. *IEEE Trans. Autom. Control, AC-38*, 1512-1516.

Richalet, J. (1993). Industrial applications of model based predictive control. *Automatica, 29*, 1251-1274.

Richalet, J.. S.E. Ata-Doss, C. Aber, H. Kuntze, A. Jacubasch and W. Schill (1987). Predictive functional control, application to fast and accurate robots. *Proc. 10th IFAC World Congress*, Munich, Germany.

Ricker, N., T. Subrahmanian and T. Sim (1988). Case studies of model predictive control in pulp and paper production. In: T.J. McAvoy *et al.* (Eds.), *Proc. IFAC Workshop on Model Based Process Control*, p. 13, Pergamon Press, Oxford.

Scokaert, P. and J. Rawlings (1996). Infinite horizon linear quadratic control with constraints. *Proc. 13th IFAC World Congress*, San Francisco, CA, USA, pp. 109-114.

Scokaert, P. and J. Rawlings (1996). On infeasibilities in model predictive control. In: *Chemical Process Control V, Assessment and new directions for research*, Tahoe City, CA, USA.

Tatrai, F., P. Lant, P. Lee, I. Cameron and R. Newell (1994). Model reduction for regulatory control: an FCCU case study. *Trans. IChem, 72*, Part A, 402-407.

Zheng, A. and M. Morari (1995). Stability of model predictive control with mixed constraints. *IEEE Trans. Autom. Control, AC-40*, 1818-1823.

Selected Topics in Identification, Modelling and Control
Vol. 11, December 1998

Dynamic modeling and feedback control of a piezo-based milli-actuator

R.A. de Callafon[‡], D.H.F. Harper[§], R.E. Skelton and F.E. Talke[♯]

University of California, San Diego
Dept. of Applied Mechanics and Engineering Sciences
9500 Gilman Drive, La Jolla, CA 92093-0411, U.S.A.
E-mail: callafon@ames.ucsd.edu

Abstract. In magnetic disk drive actuators, the application of piezo electric material can be used to refine and accomplish track following in (extremely) high track density magnetic data storage. In this paper the results on the modeling and control of a piezo-based milli-actuator are presented. The modeling is done on the basis of a least squares curve fitting of an estimated frequency response and taking into account uncertainties in the modeled resonance modes of the actuator. The design and implementation of a robust controller provides a high bandwidth and accurate positioning of the tip of the suspension and illustrates the efficiency of the piezo-based milli-actuator.

Keywords. System identification; piezoelectric; robust control; high track density recording.

1 Introduction

An unavoidable trend in magnetic recording is the aim to reduce the size or surface on which the magnetic media has to be stored. Especially, in magnetic disk drives there is an ungoing need to increase the storage capacity and areal density of the disk (Grochowski *et al.*, 1993; Grochowski and Hoyt, 1996). As a result, the track density needs to be increased significantly and the data has to be recorded and read with extreme precision.

The areal density is a combination of track density, measured in track per inch (TPI) in radial direction of the disk and bit density, measured in bits per inch (BPI) in tangential direction of the disk. For future high track density recording applications with areal densities of 10Gbit/in^2, the track density approaches 25kTPI, yielding a track pitch of 1μm and an allowable servo error of 0.1μm (Miu and Tai, 1995; Grochowski and Hoyt, 1996). These position requirements go beyond the possibilities of existing hard disk drive mechanisms.

In many of the existing hard disk drive mechanisms, a single Voice Coil Motor (VCM) actuator is used to perform the positioning of the read/write head over the disc surface. Novel design concepts such as advances in head and disk design, interface and channel technologies have allowed the improvement of the storage capacity. However, the development of a faster and more accurate servo mechanism that is able to position the read/write head with increased precision is still an active field of research (Cheung *et al.*, 1996; Koganezawa *et al.*, 1996; Horsley *et al.*, 1997).

Most of the research is directed towards the design of so-called micro- and milli-actuators. These actuators are used in a dual-stage concept, where the VCM is used for the gross movements, while a second (milli-) actuator is used for the fine movements of the read/write head located at the tip of the suspension. In this way, an accurate and high bandwidth actuator can be obtained that is believed

[‡]Author to whom all correspondence should be addressed. Raymond de Callafon was with the Mechanical Engineering Systems and Control Group, Delft University of Technology. He is now with Department of AMES, University of California, San Diego.

[§]Now with IBM, San Jose.

[♯]Center of Magnetic Recording Research at UCSD.

to explore the possibilities for high track density recording (Fan *et al.*, 1995).
The aim of this paper is to present the results on the modeling and control of a prototype of a piezo electric milli-actuator. The milli-actuator used in this paper is based on the principle of two piezoelectric stacks that are inserted into the E-block, behind the base of the suspension. A schematic picture of this principle is depicted in Figure 1. As indicated in this figure, the push/pull configuration of the piezo stacks is used to achieve a radial displacement of the tip of the suspension. The advantage of this proposed design is that it does not modify the shape of the suspension itself, thereby eliminating the need for suspension redesign.

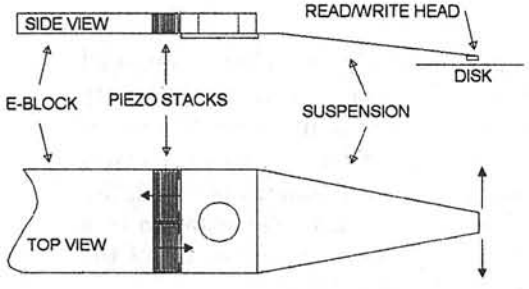


Fig. 1: Principle for piezo electric milli-actuator

The outline of the paper is as follows. First, the modeling of a prototype for the milli-actuator is presented in section 2. The modeling is done by curve fitting a measured frequency response and assuming additional uncertainties in the modeled resonance modes of the suspension and actuator. In section 3, the obtained model with uncertainty is used in an H_∞-norm based optimization to design a robust feedback controller for the prototype design. To illustrate the effectiveness of the feedback controlled milli-actuator, in section 3 also the measured closed-loop step responses are presented where the milli-actuator and read/write head suspension is applied to a rotating disk. Finally, the paper is ended by the conclusions and future research topics mentioned in section 4.

2 Modeling of milli-actuator

2.1 Prototype design

A prototype was built to study the properties of the milli-actuator and the read/write head suspension in interaction with a rotating hard disk. The prototype is used to gather experimental data for modeling purposes and to test feedback controllers being designed. Compared to the configuration depicted in Figure 1, for the prototype a slightly different design is used. A picture of the prototype being used is depicted in Figure 2.

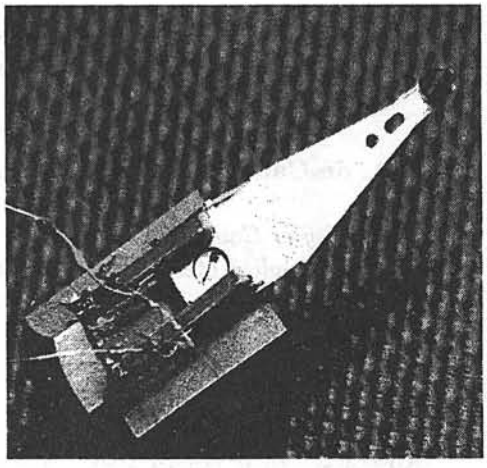


Fig. 2: Bottom view of prototype with special e-block, piezo stacks, wiring and suspension of read/write head

In the prototype design of Figure 2, the connection of the suspension to the e-block is used as a pivoting device. The piezo stacks are attached with cyanoacrylate to the bottom of the suspension and a special e-block. The special e-block is used in order to accommodate the test platform to support the read/write head suspension over a rotating disk, whereas the bottom connection of the stacks provide easy access to the piezo stacks for experimentation purposes.

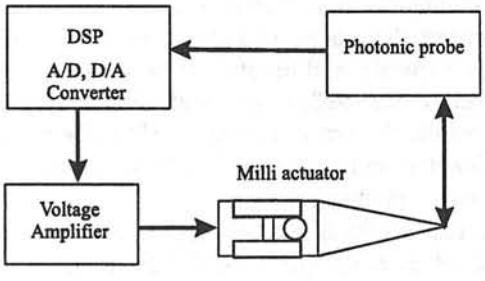


Fig. 3: Experimental set up with DSP, milli-actuator, photonic probe and voltage amplifier

The modeling of the prototype design is done via system identification techniques, where measured time domain data is used to formulate a dynamical model of the the milli-actuator and the read/write head suspension in interaction with a rotating hard disk. As indicted in Figure 3, the experimental data

is obtained by using a photonic probe to measure the relative displacement $y(t)$ of the read/write head located at the suspension tip. For excitation purposes, the piezo stacks are supplied with a random input voltage $u(t)$. The generation of the input voltage u and the measurement of the relative position y is done with a Digital Signal Processor (DSP).

To illustrate the attainable static displacement at the suspension tip with the prototype design, an open-loop measured step response is plotted in Figure 4. From Figure 4 it can be observed that for an 8 Volt input voltage step, an average displacement of approximately 3μm is measured at the suspension tip. However, the open-loop behavior of the milli-actuator exhibits many lightly damped resonance modes that need to be modeled and possibly controlled in order to achieve the required position accuracy of 0.1μm.

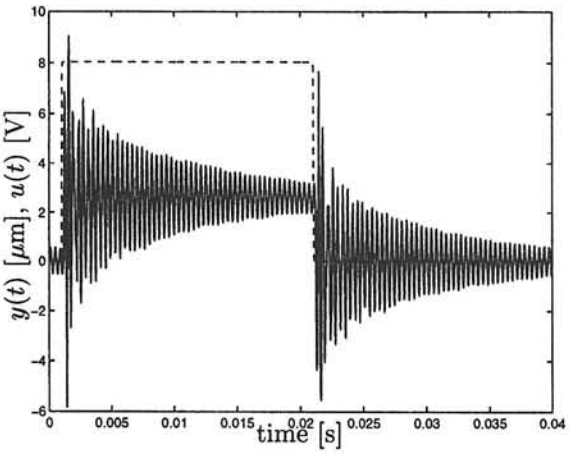


Fig. 4: Measured tip position $y(t)$ in μm (—) to a step input $u(t)$ in voltage (- -)

Since the milli-actuator is a fast responding mechanical system and experiments can be gathered relatively easily, it is advantageous to model the milli-actuator via a frequency domain based system identification technique (Pintelon *et al.*, 1994). In such a system identification technique, a frequency response is measured and used to find a dynamical model by curve fitting the obtained frequency response. For the piezo based milli-actuator depicted in Figure 2, the frequency domain of interest lies between the 100 Hz and the 10 kHz. For this range, a frequency response will be estimated and used to construct a dynamical model for the milli-actuator.

2.2 Frequency response estimation

The experimental setup depicted in Figure 3 is used to measure the frequency response of the milli-actuator. To obtain a frequency response of the milli-actuator, the input signal $u(t)$, being the input to the piezo stacks is designed as a sum of 300 sinusoids. Formally, the input $u(t)$ is given as

$$u(t) = \sum_{k=1}^{N} a_k sin(w_k t + \phi_k), \text{ with } N = 300 \quad (1)$$

where the frequency grid $\Omega := (\omega_1, \omega_2, \ldots \omega_N)$ is chosen such that the frequencies $f_k = \omega_k/(2\pi)$ are distributed approximately logarithmically between 100 Hz and 9 kHz.

The amplitude a_k in (1) for each sinusoid is kept constant and set to 1, while the phase ϕ_k is chosen randomly using a uniform distribution between $-\pi$ and π. In this way, a noisy signal input signal $u(t)$ is obtained with a well defined auto spectrum $S_{uu}(\omega)$ (Ljung, 1987).

The periodic input signal $u(t)$ is applied to the piezo stacks of the milli-actuator and the relative displacement $y(t)$ of the tip of the suspension is measured and stored by the DSP. The signals $u(t)$ and $y(t)$ are sampled at 20 kHz and used to estimate the cross spectrum $S_{yu}(j\omega)$ using spectral analysis (Priestley, 1981).

With the estimated cross spectrum $S_{yu}(j\omega)$ and the auto spectrum $S_{uu}(\omega)$, the frequency response $G(j\omega_k)$ of the milli-actuator can be estimated along the frequency grid Ω via

$$G(j\omega_k) = \frac{S_{yu}(j\omega_k)}{S_{uu}(\omega_k)}$$

An amplitude Bode plot of the estimated frequency response $G(j\omega_k)$ has been depicted in Figure 5.

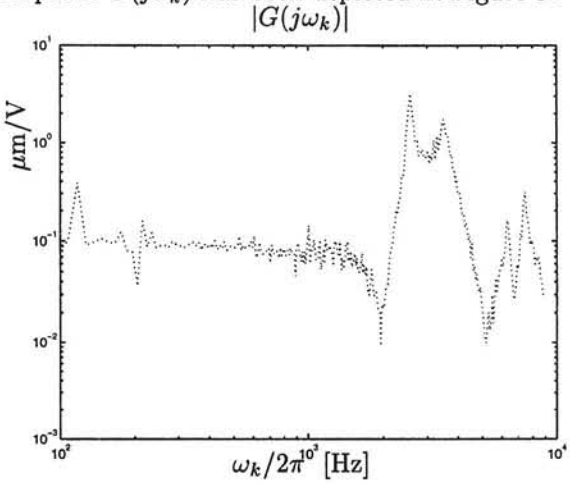


Fig. 5: Amplitude bode plot of measured open-loop frequency response between piezo stack voltage and suspension tip displacement

It can be seen from Figure 5 that the milli-actuator exhibits several lightly damped resonance modes.

It should be noted that the frequency response depicted in Figure 5 is measured while the read/write head at the tip of the suspension was supported by a rotating disk.

2.3 Least squares curve fitting

Given the estimated frequency response $G(j\omega_k)$ along the frequency grid $\Omega = (\omega_1, \omega_2, \ldots \omega_N)$, the aim of the frequency domain identification is to find a (discrete time) linear time invariant model P of limited complexity that approximates the data $G(j\omega_k)$. This model can be used to characterize the natural frequency and damping of the resonance modes of the milli-actuator. Additionally, such a discrete time model can be used to design a digital feedback controller.

To address the limited complexity, the SISO model P to be determined is parametrized in a transfer function representation

$$P(z,\theta) = \frac{b_0 + b_1 z^{-1} + \ldots b_n z^{-n}}{1 + a_1 z^{-1} + \cdots a_n z^{-n}} \qquad (2)$$

where $z = e^{j\omega}$ denote the z-transform variable and

$$\theta := [b_0 \; b_1 \; \cdots b_n a_1 \cdots a_n]$$

denotes a real valued parameter of unknown coefficients in the transfer function representation given in (2). Furthermore, it can be seen from (2) that the order or complexity of the linear model can be specified with the integer value n.

The approximation of the data $G(j\omega_k)$ by the model $P(z,\theta)$ is addressed by considering the following curve fit error

$$E(j\omega_k,\theta) := [G(j\omega_k) - P(e^{j\omega_k},\theta)]W(j\omega_k) \qquad (3)$$

that needs to be minimized. In (3), $W(j\omega_k)$ denotes a scalar weighting function used to influence the curve fitting of the frequency response data.

With the definition of the curve fit error in (3), a parameter $\hat{\theta}$ is estimated by solving the following (non-linear) minimization

$$\hat{\theta} = \arg\min_{\theta} \sum_{k=1}^{N} E(j\omega_k,\theta)E^*(j\omega_k,\theta)$$

where * is used to denote the complex conjugate transposed. A computational procedure to address this minimization is presented in de Callafon *et al.* (1996) and used in this paper to find a model $P(z,\hat{\theta})$ via a LS curve fitting.

2.4 Modeling results

Using the estimated frequency response depicted in Figure 5, the LS curve fitting described in the previous section is used to find a dynamical model of the milli-actuator. The order n of the discrete time model $P(z,\theta)$ is set to 10 in order to capture the various resonance modes present in the estimated frequency response. The weighting $W(j\omega_k)$ in (3) is set to the inverse of the data $G(j\omega_k)$ so as to minimize a relative curve fit error

$$E(j\omega_k,\theta) = \frac{[G(j\omega_k) - P(e^{j\omega_k},\theta)]}{G(j\omega_k)}.$$

The LS curve fitting procedure of de Callafon *et al.* (1996) is used and the results have been depicted in Figure 6.

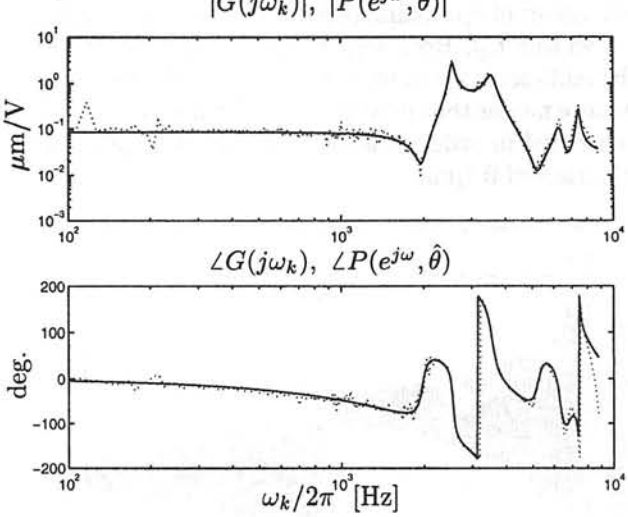


Fig. 6: Bode plot of measured frequency response (dotted) and 10th order linear time invariant model (solid)

It can be observed from Figure 6 that the dominant frequency modes have been modeled accurately by the estimated model $P(z,\hat{\theta})$. The model P can be used for the design of a servo controller for the milli-actuator. However, to design a robust controller, uncertainties and product variability in the milli-actuator have to be taken into account.

2.5 Modeling uncertainties

To design a robust performing servo controller for the milli-actuator, uncertainties in the modeled resonance modes of the actuator have to be taken into account. In this paper, the uncertainties in the milli-actuator are modeled by assuming that the nominal model $\hat{P} = P(z,\hat{\theta})$ is allowed to be perturbed to a model P via a unstructured bounded multiplicative perturbation Δ (Zhou *et al.*, 1996). In this way, a set of models $\mathcal{P}$ is found that is given by

$$\mathcal{P} = \{P \mid P = \hat{P}(1+\Delta),\ \|W\Delta\|_\infty < 1\} \qquad (4)$$

where W is stable and stably invertible frequency dependent weighting function.

In order to pursue the design of a robust controller, a choice for the weighting function W in (4) is made. The choice for W is based on the assumption that the frequency response of the nominal model $\hat{P}$ around the zeros at 1.9 and 5.2 kHz and the two resonance modes at 2.5 and 3.5 kHz are allowed to vary. With a possible choice for the weighting filter W depicted in Figure 7 on the top, the a resulting mplitude Bode plot of the models P within the set $\mathcal{P}$ of (4) is guaranteed to lie between the dashed lines depicted in Figure 7 at the bottom.

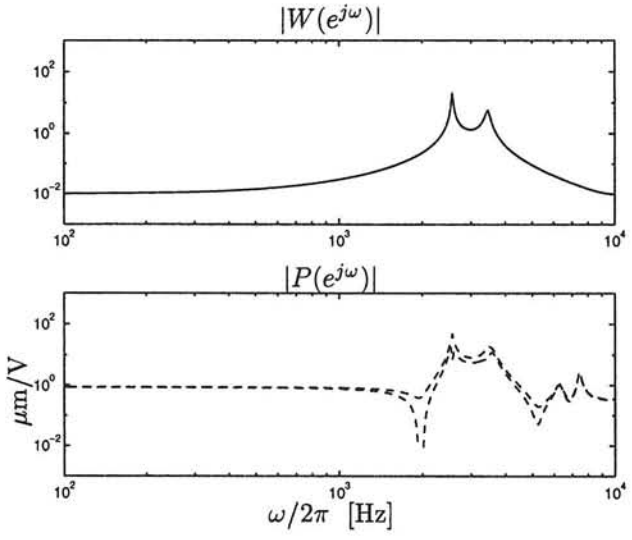


Fig. 7: Weighting function W (4th order) and amplitude Bode plot range of models P in (4)

Given the nominal model $\hat{P}$ and the multiplicative uncertainty with the stable and stable invertible frequency dependent weighting function W in (4), a robust servo controller for the milli-actuator can be designed.

3 Control of milli-actuator

3.1 Robust control design

For the design of a controller, consider the block scheme depicted in Figure 8.

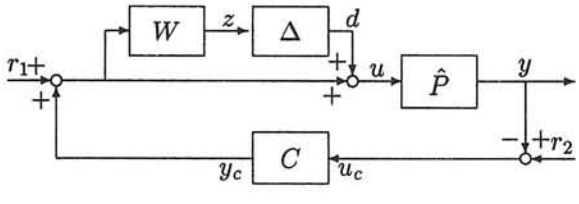


Fig. 8: Feedback controller C design of nominal model $\hat{P}$ with multiplicative uncertainty

In Figure 8, the models P within the set $\mathcal{P}$ of (4) are represented by the nominal model $\hat{P}$ and a weighted W multiplicative uncertainty Δ. In case $\Delta = 0$, Figure 8 simply represents the feedback connection of the nominal model $\hat{P}$ and the controller C. In that case, the map between $col(r_2, r_1)$ and $col(y, u)$ is given by the transfer function matrix $T(\hat{P}, C)$ with

$$T(\hat{P}, C) = \begin{bmatrix} P \\ I \end{bmatrix} (I + C\hat{P})^{-1} \begin{bmatrix} C & I \end{bmatrix} \qquad (5)$$

where r_2 indicates the suspension tip position reference signal and r_1 a piezo voltage feedforward signal. For the design of a feedback controller C that robustly stabilizes the feedback connection depicted in Figure 8 for all $P \in \mathcal{P}$ given in (4), a μ-synthesis is used (Zhou *et al.*, 1996). For that purpose, the block diagram of Figure 8 is rewritten in the standard plant configuration of Figure 9.

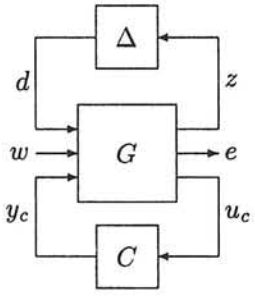


Fig. 9: Standard plant configuration

Most of the signals used in Figure 9 can be found back in Figure 8. Using the signal w and e to indicate respectively $col(r_2, r_1)$ and $col(y, u)$ it can be seen that G in Figure 9 is given by

$$G = \left[\begin{array}{c|cc|c} 0 & 0 & W & W \\ \hline \hat{P} & 0 & \hat{P} & \hat{P} \\ I & 0 & I & I \\ \hline -\hat{P} & I & -\hat{P} & -\hat{P} \end{array}\right] \qquad (6)$$

and consists of the previously determined nominal model $\hat{P}$ and weighting function W. Additionally, the (performance) signals w and e can be weighted to incorporate additional performance specifications.

The standard plant configuration can be used in a μ-synthesis to design a robustly stabilizing or robustly performing feedback controller C. Subsequently, the designed feedback controller has to be reduced in order to be implementable in a DSP environment. The feedback controller C designed with the μ-synthesis toolbox (Balas *et al.*, 1995) has a McMillan degree of 25. This can be reduced to the order of 5 using a closed-loop reduction technique (Wortelboer,

1993) without significant performance deterioration of the designed feedback system. The Bode plot of the final design of a 5th order linear discrete time controller that can be implemented on the milli-actuator is depicted in Figure 10.

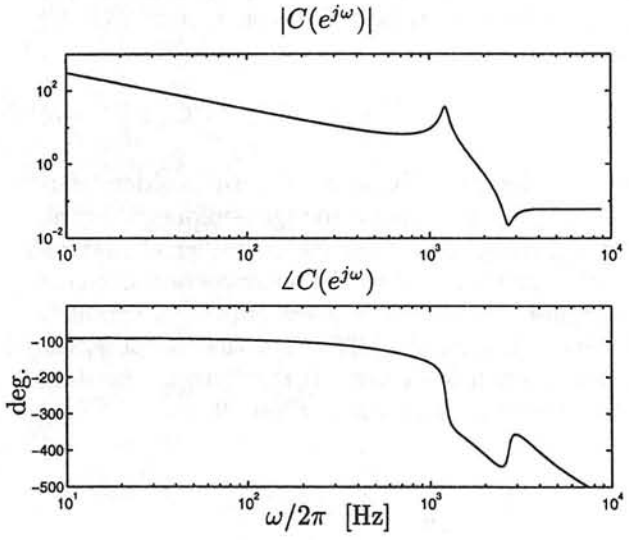


Fig. 10: Bode plot of 5th order robust linear feedback controller

As can be seen from Figure 10, the designed feedback controller C has integral action and has a sharp role-off at the area where the model uncertainty becomes larger.

3.2 Implementation of control

The experimental setup depicted in Figure 3 is used to implement the designed feedback controller on the DSP system using a sample frequency of 20 kHz. To illustrate the effect of the feedback control on the dynamical behavior of the milli-actuator, first an experimentally obtained closed-loop frequency response of the milli-actuator is presented in Figure 11.

The amplitude Bode plot in Figure 11 represent the estimated frequency response of the transfer function $P(I + CP)^{-1}C$ between suspension tip position reference signal r_2 and suspension tip displacement y in Figure 8. Similar as the previously discussed experimental results, the frequency response is estimated while the suspension is applied to a rotating disk. Compared to the open-loop frequency response depicted in Figure 5, it can be seen that a significant reduction of the resonance modes has been obtained. Furthermore, the milli-actuator is able to track signals up to approximately 1.5 kHz.

The improved control of the flexible resonance modes in the suspension with the milli-actuator can also be seen from an experimentally obtained closed-

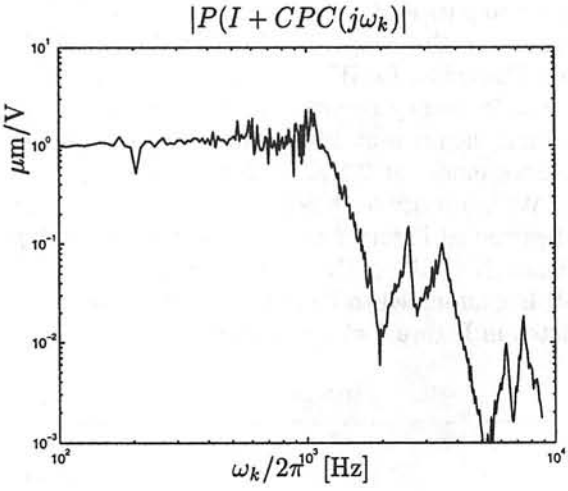


Fig. 11: Amplitude bode plot of measured closed-loop frequency response between suspension tip position reference and suspension tip displacement

loop step response. To illustrate the attainable closed-loop controlled static displacement, a closed-loop step response has been depicted in Figure 12.

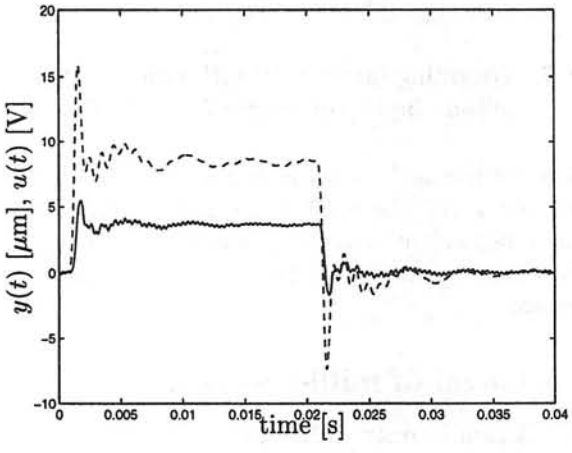


Fig. 12: Measured tip position $y(t)$ in μm (—) and feedback control signal $u(t)$ in voltage (- -) to a step on suspension tip position reference signal r_2

Compared to the open-loop step response depicted in Figure 4, it can be seen from Figure 12 that indeed a significant reduction of the flexible modes in the suspension and the milli-actuator has been achieved. Moreover, the milli-actuator is able to achieve the position accuracy requirement of 0.1μm in the step tests and has a settling time of less than 10ms.

4 Conclusions

In this paper the results on the modeling and control of a piezo-based milli-actuator are presented. The modeling is based on an experimental data and uses frequency domain identification techniques to curve fit an estimated frequency response. Modeling uncertainties and product variability in the milli-actuator are taken into account by assuming a weighted multiplicative uncertainty for the nominal model being estimated. For that purpose, a frequency dependent weighting function is chosen. However, it is worthwhile to develop a systematic procedure to estimate the contributions of modeling uncertainties in more detail on the basis of experimental data.

The model and suggested modeling uncertainty are used in a robust control design. The order of the designed controller is reduced for to address implementation issues on a DSP. The implementation of the feedback controller provides a high bandwidth and accurate positioning of the tip of the suspension and illustrates the efficiency of the piezo-based milli-actuator. The piezo stack are not only able to move the tip of the suspension but are also able to control the flexibilities in the suspension for increased position performance requirements of the read/write head. Although only the control of the milli-actuator is addressed in this paper, extension to dual-stage control that includes the voice coil motor are in line of the work presented in this paper.

References

Balas, G.J., J.C. Doyle, K. Glover, A. Packard and R. Smith (1995). *μ–Analysis and Synthesis Toolbox.* MUSYN Inc. and The Mathworks, Inc.

Cheung, P., R. Horowitz and R.T. Howe (1996). Design, fabrication, position sensing, and control of an electrostatically–driven polysilicon microactuator. *IEEE Trans. Magnetics, 32*, 122–128.

de Callafon, R.A., D. Roover and P.M.J. Van den Hof (1996). Multivariable least squares frequency domain identification using polynomial matrix fraction descriptions. *Proc. 35th IEEE Conf. Decision and Control*, Kobe, Japan, pp. 2030–2035.

Fan, L.S., H.H. Ottesen, T.C. Reiley and R.W. Wood (1995). Magnetic recording head positioning at very high track densities using a microaactuator–based, two–stage servo system. *IEEE Trans. Industr. Electronics, 42*, 222–223.

Grochowski, E. and R.F. Hoyt (1996). Future trends in hard disk drives. *IEEE Trans. Magnetics, 32*, 1850–1854.

Grochowski, E., R.F. Hoyt and J.S. Heath (1993). Magnetic hard disk drive form factor evolution. *IEEE Trans. Magnetics, 29*, no. 6.

Horsley, D.A., A. Singh, A.P. Pisano and R. Horowitz (1997). Angular micropositioner for disk drives. *Proc. IEEE MEMS Workshop*, Nagoya, Japan.

Koganezawa, S., K. Takaishi, Y. Mizoshita, Y. Uematsu, T. Yamada and S. Hasegawa (1996). A flexural piggyback milli–actuator for over 5 Gbit/in^2 density magnetic recording. *IEEE Trans. Magnetics, 32*, 3908–3910.

Ljung, L. (1987). *System Identification: Theory for the User.* Prentice–Hall, Englewood Cliffs, NJ, USA.

Miu, D.K. and Y.C. Tai (1995). Silicon micromachined SCALED technology. *IEEE Trans. Industr. Electronics, 42*, 234–239.

Pintelon, R., P. Guillaume, Y. Rolain, J. Schoukens and H. Van hamme (1994). Parametric identification of transfer functions in the frequency domain – a survey. *IEEE Trans. Autom. Control, AC–39*, 2245–2260.

Priestley, M.B. (1981). *Spectral Analysis and Time Series.* Academic Press, London, England.

Wortelboer, P.M.R. (1993). *Frequency Weighted Balanced Reduction of Closed–Loop Mechanical Servo–Systems: Theory and Tools.* PhD thesis. Delft University of Technology, Delft, the Netherlands.

Zhou, K., J.C. Doyle and K. Glover (1996). *Robust and Optimal Control.* Prentice–Hall, Upper Saddle River, NJ, USA.

Doctoral dissertations published in 1998

Hans Dötsch
Identification for Control Design with Application to a Compact Disc Mechanism
17 March 1998
Advisors: Okko Bosgra and Paul Van den Hof

Current address:
AMP-Holland B.V., P.O. Box 288, 5201 AG 's-Hertogenbosch, The Netherlands.
E-mail: hdotsch@amp.com

Jan do Livramento
Dynamic Modelling for Analysis and Design of Bottle Conveying Systems in High-Speed Bottling
6 April 1998
Advisors: Okko Bosgra and Ton van der Weiden

Current address:
Vitrite, P.O. Box 8002, 4330 EA Middelburg, The Netherlands.

Raymond de Callafon
Feedback Oriented Identification for Enhanced and Robust Control - A fractional approach applied to a wafer stage
13 October 1998
Advisors: Okko Bosgra and Paul Van den Hof

Current address:
Department of Applied Mechanics and Engineering Sciences, University of California, San Diego, 9500 Gilman Drive, La Jolla, CA 92093-0411, USA.
E-mail: callafon@ames.ucsd.edu

More information on these dissertations can be found on WWW-page:
`www-sr.wbmt.tudelft.nl/sr/thesis`

CONTENTS Volume 11, December 1998

TU Delft
Delft University of Technology

ISBN 90-407-1817-2