ISSN 1562 - 2703

INTERNATIONAL JOURNAL AUTOMATION AUSTRIA

HEFT 1 Jahrgang 18 (2010)

INHALT *Seite*

SCOPE

"International Journal Automation Austria" (IJAA) publishes top quality, peer reviewed papers in all areas of automatic control concerning continuous and discrete processes and production systems.

Only original papers will be considered. No paper published previously in another journal, transaction or book will be accepted. Material published in workshops or symposia proceedings will be considered. In such a case the author is responsible for obtaining the necessary copyright releases. In all cases, the author must obtain the necessary copyright releases before preparing a submission. Papers are solicited in both theory and applications

Before preparing submissions, please visit our instructions to authors (see back cover) or web page.

International Journal Automation Austria also is the official executive authority for publications of IFAC-Beirat Österreich.

Imprint: Propagation of Automatic Control in Theory and Practice.

Frequency: Aperiodically, usually twice a year.
Publisher: IFAC-Beirat Österreich, Peter Kopacek, Alexander Weinmann
Editors in Chief: Alexander Weinmann, Peter Kopacek

Address: 1) Institut für Automatisierungs- und Regelungstechnik (E376), TU Wien,
A-1040 Wien, Gußhausstrasse 27-29, Austria
Phone: +43 1 58801 37677, FAX: +43 1 58801 37699
email: danzinger@acin.tuwien.ac.at
Homepage: http://www.acin.tuwien.ac.at/publikationen/ijaa/

2) Intelligente Handhabungs- und Robotertechnik (E325/A6), TU-Wien,
A-1040 Wien, Favoritenstrasse 9-11, Austria
email: e318@ihrt.tuwien.ac.at

Layout: Rainer Danzinger
Printing: Kopierzentrum der TU Wien

Feedback Linearization of Descriptor Systems – A Classification Approach

F. Gausch, N. Vrhovac
Institute of Electrical Engineering and Information Technology
University of Paderborn, Germany

Abstract: *This article discusses the linearization and decoupling of the input-output behaviour of non-linear time-invariant descriptor systems using a static or a dynamic feedback. The mathematical models are given in a semi-explicit descriptor form (DAE) where their regularity and properness as well as their affinity of the differential equations in the input variables are assumed. The explicit descriptor model is crucial for the analysis of such models, for which reason an algorithm will be introduced initially which guarantees the detection of said model, thus determining the (differential) index.*
Beyond the classification of feedback design procedures this paper is focused on the investigation of a possible increase in the index of the total system, with the increase probably being caused by algebraic loops formed by the algebraic equations of the descriptor model along with a static feedback. Finally we will derive a sufficient condition to detect the occurrence of such algebraic loops as early as in the design phase.

Keywords: *Non-linear descriptor system, input-output decoupling, input-output linearization, differential-geometric methods.*

1 Introduction

The core aspect of this paper is the linearization and decoupling of the input-output behaviour of a certain class of non-linear systems in descriptor form. The main focus of this discussion lies on the classification of the design procedure regarding a static or a dynamic feedback, respectively. Specific attention is directed to the possible existence of algebraic loops which can occur in the closed loop system when a static feedback is used. The investigations are carried out by means of differential-geometric methods.
This range of issues has already been followed and processed with great success for more than two decades for non-linear systems in state space form. Methods going beyond the problem of decoupling were not only developed to investigate characteristics such as reachability, observability or stabilizability and to design tracking control, disturbance attenuation, optimal control, etc. but were also applied increasingly in the non-linear

control. The successes achieved have been documented exhaustingly in a large number of publications. In the meantime a similar situation can be found when investigating systems in descriptor forms; however, the majority of the studies has been made in connection with linear descriptor systems. When compared to the comprehensive discussion of the decoupling problem with non-linear systems in state space form and/or with linear systems in descriptor form, then the investigation of the same problem with non-linear systems in descriptor form is done on a much smaller scale – see for example the contributions (G & Müller, 2004; Müller, 1996; Schlacher & Kugi, 2001; Schlacher, Kugi & Zehetleitner, 2002) and the references therein which are dedicated in many cases not exclusively to the problem of linearization and decoupling.
Initial general approaches directed towards the linearization of descriptor systems – that is other than successes made with regard to specific tasks mainly in connection with motion constraints of manipulators – can probably be ascribed to the work of (Clamroch, 1990; Kawaji & Taha, 1994; Xiaoping & Čelikovský, 1997). The mathematical models in those publications are still very limited formally because the differential equations have an affinity in the descriptor variables, and because the algebraic equations are either dependent only on the differential variables, or they have an affinity in the input variables and the algebraic variables.
Organization of the paper: Chapter 2 firstly summarizes the mathematical tools required for the analysis and the synthesis. These advanced aids are attributed to the former works (G & Müller, 2004; Müller, 2000). Beyond the design of feedback by the novel classification based on the regularity of decoupling matrices chapter 3 then investigates the possibility of increasing the index within the model of the overall system as a result of feedback over a static system, and describes a sufficient condition required for detecting such problems in the design phase. The examples given in chapter 4 are then used to illustrate the problems addressed.

2 Non-linear semi-explicit AI Descriptor Systems

2.1 Formulations and Preliminaries

Consider a non-linear MIMO system given by the semi-explicit AI descriptor model (1):

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{a}(\boldsymbol{x}, \boldsymbol{z}) + \boldsymbol{B}(\boldsymbol{x}, \boldsymbol{z})\, \boldsymbol{u} \\ \boldsymbol{0} &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) \\ \boldsymbol{y} &= \boldsymbol{c}(\boldsymbol{x}, \boldsymbol{z}) \end{aligned} \tag{1}$$

The vectors $\boldsymbol{x} = [x_1 \cdots x_n]^T$ and $\boldsymbol{z} = [z_1 \cdots z_p]^T$ are denoted by *differential* variable $\boldsymbol{x}$ and by *algebraic* variable $\boldsymbol{z}$, respectively. The sufficiently smooth vector-valued function $\boldsymbol{g}$ with dimension p permits dependencies among the differential variables from x_1 to x_n so that the dynamic order $\tilde{n}$ of the system can be smaller than the number n of differential variables. The output $\boldsymbol{y} = [y_1 \cdots y_m]^T$ is assumed not to be affected directly by the input $\boldsymbol{u} = [u_1 \cdots u_m]^T$. The vector-valued functions $\boldsymbol{a}$ and $\boldsymbol{c}$ with the dimensions n and m, respectively, as well as the matrix-valued function $\boldsymbol{B}$ with the dimension $(n \times m)$ are also assumed to be sufficiently smooth.

Under certain assumptions (which will be discussed below) the semi-explicit descriptor model (1) can be transformed into an explicit descriptor model (2)

$$\begin{aligned} \dot{\boldsymbol{w}} &= \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{u}, \dot{\boldsymbol{u}}, \ldots, \overset{(d)}{\boldsymbol{u}}) \\ \boldsymbol{y} &= \boldsymbol{c}(\boldsymbol{w}) \end{aligned} \tag{2}$$

with the descriptor variables $\boldsymbol{w} = [\boldsymbol{x}^T,\, \boldsymbol{z}^T]^T$. This explicit model is required to define *regularity* and *properness* of descriptor model (1) as stated below. The key for the construction of the vector-valued function $\boldsymbol{f}$ (essentially that is determining $\dot{\boldsymbol{z}}$) is the successive generation of the time derivatives of every algebraic equation in $\mathbf{0} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = [g_1 \cdots g_i \cdots g_p]^T$:

$$\begin{aligned} 0 &= \overset{(\nu)}{g_i} = g_{i,\nu}(\boldsymbol{x}, \boldsymbol{u}, \ldots, \overset{(\nu)}{\boldsymbol{u}}) && \nu = 0, \ldots, k_i - 2 \\ & && \text{only if } k_i > 1 \\ 0 &= \overset{(k_i-1)}{g_i} = g_{i,k_i-1}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \ldots, \overset{(k_i-1)}{\boldsymbol{u}}) && k_i \geq 1 \\ 0 &= \overset{(k_i)}{g_i} = g_{i,k_i}(\boldsymbol{x}, \boldsymbol{z}, \dot{\boldsymbol{z}}, \boldsymbol{u}, \ldots, \overset{(k_i)}{\boldsymbol{u}}) \end{aligned} \tag{3}$$

In this equations k_i is the *differential index* of the i-th algebraic equation g_i and also denotes the time derivative of g_i where the time derivative $\dot{z}_j$ of at least one algebraic variable z_j appears for the first time.

Summarizing all the equations from the penultimate steps in a new vector-valued algebraic equation

$$\mathbf{0} = \begin{bmatrix} g_{1,k_1-1} \\ \vdots \\ g_{p,k_p-1} \end{bmatrix} =: \boldsymbol{g}_{k-1} \tag{4}$$

yields a compact formulation of the last differentiation steps

$$\begin{aligned} \mathbf{0} = \dot{\boldsymbol{g}}_{k-1} = \frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}} \dot{\boldsymbol{z}} + \frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} + \frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}} + \\ + \ldots + \frac{\partial \boldsymbol{g}_{k-1}}{\partial \overset{(d-1)}{\boldsymbol{u}}} \overset{(d)}{\boldsymbol{u}} \end{aligned} \tag{5}$$

where

$$d \leq \max_{i=1,\ldots,p} \{k_i\}$$

Now if the Jacobian $\partial \boldsymbol{g}_{k-1} / \partial \boldsymbol{z}$ of the new algebraic equation (4) is a regular matrix (which is not necessarily the case but will be guaranteed below by presupposing regularity of descriptor model (1)), expression (5) can be solved with respect to $\dot{\boldsymbol{z}}$ then constituting the function $\boldsymbol{f}$

$$\boldsymbol{f} = \begin{bmatrix} \boldsymbol{a} + \boldsymbol{B}\,\boldsymbol{u} \\ -\left[\dfrac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}}\right]^{-1}\left(\dfrac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{x}}(\boldsymbol{a}+\boldsymbol{B}\boldsymbol{u}) + \dfrac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{u}}\dot{\boldsymbol{u}} + \right. \\ \left. + \ldots + \dfrac{\partial \boldsymbol{g}_{k-1}}{\partial \overset{(d-1)}{\boldsymbol{u}}}\overset{(d)}{\boldsymbol{u}}\right) \end{bmatrix} \tag{6}$$

in the explicit descriptor model (2).

2.2 Regularity and Properness

All following sections are restricted to *regular* and *proper* descriptor models:

Definition 1 *A semi-explicit descriptor model (1) exhibiting an explicit descriptor model (2) is said to be* regular*.*

Definition 2 *An explicit descriptor model (2) with* $d \leq 1$ *is said to be* proper*.*

In the practically relevant sense regularity and properness do not mean any restriction because, in short, only the system of equations (1) is required to have a unique solution which does not depend on time-derivatives of the input.
Strictly speaking, it follows from the regularity condition that on certain smoothness requirements the differential equation in the model (2) has a unique solution (Vidyasagar, 1978) which consequently applies to the underlying differential-algebraic equations in the model (1), provided the initial values are consistent (Reich, 1990). Consistent initial values are governed by the original algebraic equations along with equations created by the differentiation process (3):

$$\begin{aligned} 0 &= \overset{(\nu)}{g_1} = g_{1,\nu} \qquad \nu = 0, \ldots, k_1 - 1 \\ &\vdots \\ 0 &= \overset{(\nu)}{g_p} = g_{p,\nu} \qquad \nu = 0, \ldots, k_p - 1 \end{aligned}$$

These are k_T restrictions; $\tilde{n} = n + p - k_T$ initial values can be given arbitrarily. k_T is called *total index*:

$$k_T := \sum_{i=1}^{p} k_i \tag{7}$$

Finally, it follows from the properness condition that in the explicit differential equation of the model (2), the first derivative of the input appears at most ($d \leq 1$ holds). Because of this, the solution $\boldsymbol{w}$ does not depend on the derivatives $\dot{\boldsymbol{u}}, \ddot{\boldsymbol{u}}, \ldots$ so that the differential-algebraic system (1) is proper.

2.3 Modified Shuffle Algorithm

This algorithm is used to calculate the algebraic equations (4) for regular and proper semi-explicit descriptor models. The shuffle algorithm was stated for the first time in (Luenberger, 1978) for linear systems and was then extended in (Gear, 1988) for non-linear differential algebraic equations with regard to their numerical solution. Finally, a modification of this algorithm was developed in (Müller, 2000), which plays an important role for the linearization and decoupling of descriptor systems given in the semi-explicit form.

Based on the regularity and properness of the system (1), the calculation scheme (3) for the successive generation of the time derivatives of every algebraic equation now reads:

$$\begin{aligned} 0 &= g_{i,\nu}(\boldsymbol{x}) \qquad \nu = 0,\dots,k_i-2 \;\text{ only if } k_i > 1 \\ 0 &= g_{i,k_i-1}(\boldsymbol{x},\boldsymbol{z},\boldsymbol{u}) \qquad\qquad\qquad k_i \geq 1 \\ 0 &= \frac{\partial g_{i,k_i-1}}{\partial \boldsymbol{x}}\,\dot{\boldsymbol{x}} + \frac{\partial g_{i,k_i-1}}{\partial \boldsymbol{z}}\,\dot{\boldsymbol{z}} + \frac{\partial g_{i,k_i-1}}{\partial \boldsymbol{u}}\,\dot{\boldsymbol{u}} \end{aligned} \tag{8}$$

The last equation in the scheme is considered a differential equation in one or several algebraic variables. It is important to check whether the differential equation produced this way does not functionally depend on other differential equations formed by the shuffle algorithm and therefore has to be eliminated; this check is expressed in the following definition of the equation index k_i

$$k_i := \min_{j=1,2,\dots} \left\{ \begin{array}{c} \dfrac{\partial g_{i,j-1}}{\partial \boldsymbol{z}} \neq \boldsymbol{0}^T \quad \text{and} \\ h_{i,j}^{(0)} \dfrac{\partial g_{i,j-1}}{\partial \boldsymbol{z}} = \sum\limits_{\substack{l=1 \\ \exists\, k_l}}^{p} h_{i,j}^{(l)} \dfrac{\partial g_{l,k_l-1}}{\partial \boldsymbol{z}} \;\overset{\text{only}}{\Longrightarrow} \\ \qquad\qquad h_{i,j}^{(l)}(\boldsymbol{x},\boldsymbol{z},\boldsymbol{u}) = 0 \\ \qquad\qquad {\scriptstyle l=0,1,\dots,p} \end{array} \right\} \tag{9}$$

by means of the auxiliary functions $h_{i,j}^{(l)}$. Obviously, if the second part of check (9) holds only for vanishing auxiliary functions, then an appearance of $\boldsymbol{z}$ detected in $g_{i,j-1}$ by the first part of check (9) cannot be compensated by expressions previously generated by the algorithm (see example 4.1 for demonstration). For that reason, in the definition above $\exists\, k_l$ means that the summation only takes into account the equations for which the index has already been calculated. A possibly required elimination influences the algebraic equation $g_i = 0$ for the next differentiation step; Proposition 1 summarizes that fact.

Proposition 1 *If check (9) detects a functional dependence in the expression $g_{i,j-1}$ this expression g_i reads (with the abbreviation $\tilde{h}_{i,j}^{(l)} := h_{i,j}^{(l)}/h_{i,j}^{(0)}$) in the next differentiation step j:*

$$g_{i,j} = \frac{\partial g_{i,j-1}}{\partial \boldsymbol{x}}\,\dot{\boldsymbol{x}} - \sum_{\substack{l=1 \\ \exists\, k_l}}^{p} \tilde{h}_{i,j}^{(l)} \frac{\partial g_{l,k_l-1}}{\partial \boldsymbol{x}}\,\dot{\boldsymbol{x}} \tag{10}$$

Proof: From $g_{i,j-1}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u})$, the calculation of $g_{i,j} = \dot{g}_{i,j-1}$ yields

$$g_{i,j} = \frac{\partial g_{i,j-1}}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} + \frac{\partial g_{i,j-1}}{\partial \boldsymbol{z}} \dot{\boldsymbol{z}} + \frac{\partial g_{i,j-1}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}}$$

which can be rewritten firstly according to the functional dependence (9)

$$g_{i,j} = \frac{\partial g_{i,j-1}}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} + \sum_{\substack{l=1\\ \exists\, k_l}}^{p} \tilde{h}_{i,j}^{(l)} \frac{\partial g_{l,k_l-1}}{\partial \boldsymbol{z}} \dot{\boldsymbol{z}} + \frac{\partial g_{i,j-1}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}}$$

and secondly taking account of all the equations with fixed index from the last equations (8):

$$g_{i,j} = \frac{\partial g_{i,j-1}}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} - \sum_{\substack{l=1\\ \exists\, k_l}}^{p} \tilde{h}_{i,j}^{(l)} \left[\frac{\partial g_{l,k_l-1}}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} + \frac{\partial g_{l,k_l-1}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}} \right] + \\ + \frac{\partial g_{i,j-1}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}}$$

Lastly, since $g_{i,j}$ has to be differentiated at least one more time ($k_i \geq j+1$) but the descriptor model (1) is presupposed to be proper, $g_{i,j}$ must not depend on $\dot{\boldsymbol{u}}$.
□

The procedure represented above yields the accompanying equation index k_i for every algebraic equation $g_i = 0$; the largest equation index results in the *differential* index k of the descriptor system (1):

$$k := \max_{i=1,\dots,p} \{k_i\} \tag{11}$$

Using the abbreviation (4) once again

$$\boldsymbol{g}_{k-1} := \begin{bmatrix} g_{1,k_1-1}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) \\ \vdots \\ g_{p,k_p-1}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) \end{bmatrix}$$

the modified shuffle algorithm (8) – (11) finally yields the new algebraic equation

$$\boldsymbol{0} = \boldsymbol{g}_{k-1}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) \tag{12}$$

instead of the original equation $\boldsymbol{0} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u})$. This new algebraic equation plays a decisive role in the calculation of the feedback because its Jacobian regarding $\boldsymbol{z}$ is a regular matrix now and the derivative of the algebraic variable can, therefore, be expressed explicitly by (cf. 6):

$$\dot{\boldsymbol{z}} = - \left[\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}} \right]^{-1} \left(\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} + \frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}} \right) \tag{13}$$

Proposition 2 *The Jacobian of the algebraic equation (12) with respect to $\boldsymbol{z}$ is a regular matrix.*

Proof: Because the descriptor model (1) is presupposed to be regular, the existence of such a matrix follows from Definition 1 along with the structure (6) and the modified shuffle algorithm forms that matrix by construction (9)(10).
□

3 Problem Formulation and Analysis

Considering the closed loop system in Fig. 1 with a *dynamic* system specified by the descriptor model (1), completed with either a *static* feedback

$$\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{v}) \tag{14}$$

or a *dynamic* feedback

$$\dot{\boldsymbol{u}} = \dot{\boldsymbol{u}}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{v}) \tag{15}$$

and driven by an external input $\boldsymbol{v} = [v_1, \ldots, v_m]^T$.

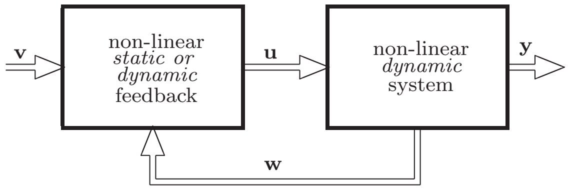


Figure 1: Non-linear system (1) with feedback, $\mathbf{w} := [\boldsymbol{x}^T, \boldsymbol{z}^T]^T$.

Based on that, this section deals with the design of a feedback to achieve a desired behaviour of the overall system – which may for example be a purely integrating and decoupled input-output behaviour of appropriate order δ_i $(i = 1, \ldots, m)$ as follows:

$$\begin{aligned} \overset{(\delta_1)}{y_1} &= v_1 \\ &\vdots \\ \overset{(\delta_m)}{y_m} &= v_m \end{aligned} \tag{16}$$

3.1 Analysis of the Input Output Behaviour of the Given System

In order to design a feedback to achieve an overall dynamic (16), derivatives of the outputs have to be calculated. In doing so, it is helpful to define the following recursive operator $N^\nu c$

$$N^\nu c := (N^{\nu-1}c)' \cdot \dot{\boldsymbol{x}} \quad \text{with} \quad N^0 c = c \tag{17}$$

where its derivative $(N^\nu c)'$ (as n-dimensional row vector) is given by:

$$(N^\nu c)' := \frac{\partial}{\partial \boldsymbol{x}}(N^\nu c) - \frac{\partial}{\partial \boldsymbol{z}}(N^\nu c)\left[\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}}\right]^{-1} \frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{x}} \tag{18}$$

In addition, the m-dimensional row vector $(M^\nu c_i)'$

$$(M^\nu c)' := \frac{\partial}{\partial \boldsymbol{u}}(N^\nu c) - \frac{\partial}{\partial \boldsymbol{z}}(N^\nu c)\left[\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}}\right]^{-1} \frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{u}} \tag{19}$$

is introduced (note that it is not build up at its own recursion).

Proposition 3 *Using the prerequisites (17) – (19), the time-derivatives of the outputs* $y_1, \ldots, y_m$ *in the descriptor model (1) can be written as:*

$$\overset{(\nu)}{y_i} = N^{\nu} c_i \qquad \nu = 1, \ldots, \gamma_i - 1 \ \text{ only if } \gamma_i > 1 \tag{20}$$

$$\begin{aligned} \overset{(\gamma_i)}{y_i} &= N^{\gamma_i} c_i + (M^{\gamma_i - 1} c_i)' \, \dot{\boldsymbol{u}} = \\ &= (N^{\gamma_i - 1} c_i)' \, (\boldsymbol{a} + \boldsymbol{B}\, \boldsymbol{u}) + (M^{\gamma_i - 1} c_i)' \, \dot{\boldsymbol{u}} \end{aligned} \tag{21}$$

with

$$\gamma_i := \min_{j=1,\ldots,n+1} \left\{ (N^{j-1} c_i)' \, \boldsymbol{B} \neq \boldsymbol{0}^T \vee (M^{j-1} c_i)' \ \neq \boldsymbol{0}^T \right\} \tag{22}$$

Proof: The result is obtained by a straightforward calculation considering the presupposed regularity and properness of the descriptor model (1). Starting from any output $y = c(\boldsymbol{x}, \boldsymbol{z})$ (the index i is suppressed in the interest of readability) the first differentiation step $\nu = 1$ reads

$$\begin{aligned} \dot{y} &= \frac{\partial c}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} + \frac{\partial c}{\partial \boldsymbol{z}} \dot{\boldsymbol{z}} = \cdots \text{cf. } (13) \cdots = \\ &= \left(\frac{\partial c}{\partial \boldsymbol{x}} - \frac{\partial c}{\partial \boldsymbol{z}} \left[\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}} \right]^{-1} \frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{x}} \right) \dot{\boldsymbol{x}} - \\ &- \left(\frac{\partial c}{\partial \boldsymbol{z}} \left[\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}} \right]^{-1} \frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{u}} \right) \dot{\boldsymbol{u}} = \\ &= \cdots \text{cf. (17)-(19) and note } \partial c / \partial \boldsymbol{u} = \boldsymbol{0}^T \cdots = \\ &= (N^0 c)' \dot{\boldsymbol{x}} + (M^0 c)' \dot{\boldsymbol{u}} = \\ &= (N^0 c)' \boldsymbol{a} + (N^0 c)' \boldsymbol{B} \boldsymbol{u} + (M^0 c)' \dot{\boldsymbol{u}} \end{aligned} \tag{23}$$

If $(N^0 c)' \boldsymbol{B} \neq \boldsymbol{0}^T$ and/or $(M^0 c)' \neq \boldsymbol{0}^T$ then condition (22) results in $\gamma = 1$ and the differentiation process stops. Assume $\gamma > 1$, then $(N^0 c)' \boldsymbol{B} = (M^0 c)' = \boldsymbol{0}^T$ results in

$$\dot{y} = N^1 c \tag{24}$$

according to (17) and for any $1 < \nu < (\gamma - 1)$ a successive differentiation step is as follows:

$$\overset{(\nu)}{y} = N^{\nu} c \tag{25}$$

$$\begin{aligned} \overset{(\nu+1)}{y} &= \frac{\partial}{\partial \boldsymbol{x}} (N^{\nu} c) \dot{\boldsymbol{x}} + \frac{\partial}{\partial \boldsymbol{z}} (N^{\nu} c) \dot{\boldsymbol{z}} + \frac{\partial}{\partial \boldsymbol{u}} (N^{\nu} c) \dot{\boldsymbol{u}} = \\ &= \cdots \text{cf. } (13) \cdots = \\ &= \left(\frac{\partial}{\partial \boldsymbol{x}} (N^{\nu} c) - \frac{\partial}{\partial \boldsymbol{z}} (N^{\nu} c) \left[\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}} \right]^{-1} \frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{x}} \right) \dot{\boldsymbol{x}} + \end{aligned}$$

$$\begin{aligned}
&+\left(\frac{\partial}{\partial \boldsymbol{u}}(N^\nu c)-\frac{\partial}{\partial \boldsymbol{z}}(N^\nu c)\left[\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}}\right]^{-1}\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{u}}\right)\dot{\boldsymbol{u}}=\\
&=\cdots \text{cf. (18),(19) and (22),(17)} \cdots =\\
&=(N^\nu c)'\dot{\boldsymbol{x}}+(M^\nu c)'\dot{\boldsymbol{u}}=\\
&=(N^{\nu+1}c) \qquad (26)
\end{aligned}$$

Steps (24),(25),(26) complete the induction and the differentiation stops in the last step γ according to (22) with:

$$\overset{(\gamma)}{y} = N^\gamma c + (M^{\gamma-1}c)'\dot{\boldsymbol{u}} \qquad (27)$$

□

Remark 1 *The so-called derivative degree γ_i (22) – it is presupposed to be constant – indicates this derivative of the output variable y_i which shows a dependence of the input $\boldsymbol{u}$ and/or its derivative $\dot{\boldsymbol{u}}$ affected by an algebraic variable $\boldsymbol{z}$ which, in general, is not explicitly known. It must be differentiated from the well-known relative degree r_i which is considered to be solely linked to a dependence on $\boldsymbol{u}$. Obviously, if $(M^{\gamma_i-1}c_i)' \neq \mathbf{0}^T$ in output (21) then $r_i = \gamma_i - 1$ holds. But attention is to be paid to the case $(M^{\gamma_i-1}c_i)' = \mathbf{0}^T$, because the algebraic equations of the descriptor system may compensate the feedthrough of the input $\boldsymbol{u}$ to the γ_i-th derivative of y_i. Generally, such a possible compensation cannot be examined directly because the algebraic variable is not given explicitly as $\boldsymbol{z} = \boldsymbol{z}(\boldsymbol{x}, \boldsymbol{u})$. However, the explicitly known derivative (13) can be used for that examination: The input $\boldsymbol{u}$ cannot have been compensated if a further time derivative of y_i depends on $\dot{\boldsymbol{u}}$.*
According to these considerations the relative degree r_i can be defined for descriptor systems:

$$r_i := \min_{j \le \gamma_i - 1}\left\{(M^j c_i)' \neq \mathbf{0}^T\right\} \qquad (28)$$

The output $\boldsymbol{y}$ is presupposed to be chosen properly thus a constant relative degree

$$\boldsymbol{r} := [r_1, \dots, r_m]^T \text{ exists.} \qquad (29)$$

3.2 Specification of the Feedback by Classification

By means of the abbreviations

$$\overset{(\boldsymbol{\gamma})}{\boldsymbol{y}} := \begin{bmatrix} \overset{(\gamma_1)}{y_1} \\ \vdots \\ \overset{(\gamma_m)}{y_m} \end{bmatrix}, \quad \hat{\boldsymbol{c}} := \begin{bmatrix} (N^{\gamma_1-1}c_1)'\,\boldsymbol{a} \\ \vdots \\ (N^{\gamma_m-1}c_m)'\,\boldsymbol{a} \end{bmatrix},$$

$$\widehat{\boldsymbol{D}} := \begin{bmatrix} (N^{\gamma_1-1}c_1)'\,\boldsymbol{B} \\ \vdots \\ (N^{\gamma_m-1}c_m)'\,\boldsymbol{B} \end{bmatrix} \quad \text{and} \quad \widetilde{\boldsymbol{D}} := \begin{bmatrix} (M^{\gamma_1-1}c_1)' \\ \vdots \\ (M^{\gamma_m-1}c_m)' \end{bmatrix}$$

the time derivative (21) of all outputs of the given system are summarized as:

$$\overset{(\boldsymbol{\gamma})}{\boldsymbol{y}} = \widehat{\boldsymbol{c}} + \widehat{\boldsymbol{D}}\,\boldsymbol{u} + \widetilde{\boldsymbol{D}}\,\dot{\boldsymbol{u}} \tag{30}$$

Now, a regular (case 1), singular (case 2) or zero (case 3) matrix $\widetilde{\boldsymbol{D}}$ is considered as basis for the following design of a linearizing and decoupling feedback.

Case 1: If the (dynamic decoupling) matrix $\widetilde{\boldsymbol{D}}$ has full rank $r = m$, the dynamic feedback (15) can be specified by:

$$\dot{\boldsymbol{u}}(\boldsymbol{x},\boldsymbol{z},\boldsymbol{u},\boldsymbol{v}) = \widetilde{\boldsymbol{D}}^{-1}\left(\boldsymbol{v} - \widehat{\boldsymbol{c}} - \widehat{\boldsymbol{D}}\,\boldsymbol{u}\right) \tag{31}$$

Obviously, the closed-loop system thus obtained has the form (16) with $\delta_i = \gamma_i$ ($i = 1,\ldots,m$) and $[v_1,\ldots,v_m]^T = \boldsymbol{v}$.

Case 2: The matrix $\widetilde{\boldsymbol{D}} \neq \boldsymbol{0}$ and has reduced rank $r < m$ as a result of some zero rows. If so, the corresponding output variables y_i are differentiated with respect to time until they are dependent on $\dot{\boldsymbol{u}}$. As a consequence of assumption (29) this is achieved by the (r_i+1)-th derivative. The dynamic decoupling matrix $\widetilde{\boldsymbol{D}}$ thus obtained has a structure

$$\widetilde{\boldsymbol{D}} := \begin{bmatrix} (M^{r_1}c_1)' \\ \vdots \\ (M^{r_m}c_m)' \end{bmatrix}$$

suitable for the step in case 1 with $\gamma_i = r_i + 1$ ($i = 1,\ldots,m$).

Case 3a: The matrix $\widetilde{\boldsymbol{D}} = \boldsymbol{0}$ and the (static decoupling) matrix $\widehat{\boldsymbol{D}}$ has full rank $r = m$. In addition, the expressions $(N^{\gamma_i-1}c_i)'$ may be assumed to be independent of $\boldsymbol{u}$ – i.e.

$$\frac{\partial}{\partial \boldsymbol{u}}(N^{\gamma_i-1}c_i)' = \boldsymbol{0}, \tag{32}$$

hence the output (30) is affine in $\boldsymbol{u}$ – because otherwise the feedback design can be transferred into the dynamic case 1 by a further differentiation of all output variables. Assuming (32), one can try to achieve an overall dynamics of kind (16) by solving

$$\overset{(\boldsymbol{\gamma})}{\boldsymbol{y}} = \widehat{\boldsymbol{c}}(\boldsymbol{x},\boldsymbol{z}) + \widehat{\boldsymbol{D}}(\boldsymbol{x},\boldsymbol{z})\,\boldsymbol{u} \overset{!}{=} \boldsymbol{v} \tag{33}$$

with respect to a static feedback $\boldsymbol{u}$:

$$\boldsymbol{u}(\boldsymbol{x},\boldsymbol{z},\boldsymbol{v}) = \widehat{\boldsymbol{D}}^{-1}(\boldsymbol{v} - \widehat{\boldsymbol{c}}) \tag{34}$$

For non-linear systems given in AI *state space* form a regular decoupling matrix $\widehat{\boldsymbol{D}}$ is a necessary and sufficient condition for the existence of a static state feedback to create a noninteracting linear overall system (Isidori & Grizzle, 1988).
However, this question is a crucial one if the static feedback (34) is integrated into a system given by a descriptor model (1). This is due to the possibility that the static feedback in combination with the algebraic equation of the dynamic system may perform an algebraic loop which is not resolvable with respect to the control input $\boldsymbol{u}$ (see the loop depicted in Fig. 2).

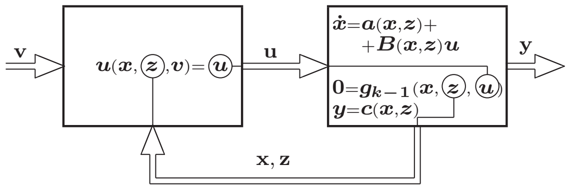


Figure 2: Algebraic loop possibly not resolvable with respect to $\boldsymbol{u}$.

Considering $\boldsymbol{z} = \boldsymbol{z}(\mathbf{x}, \boldsymbol{u})$ as the solution of the algebraic equation of the descriptor system, the input-output description (33) becomes an implicit equation

$$\widehat{\boldsymbol{c}}(\boldsymbol{x}, \boldsymbol{z}(\boldsymbol{x}, \boldsymbol{u})) + \widehat{\boldsymbol{D}}(\boldsymbol{x}, \boldsymbol{z}(\boldsymbol{x}, \boldsymbol{u}))\, \boldsymbol{u} = \boldsymbol{v} \tag{35}$$

to be solved for $\boldsymbol{u}$ – does eq. (35) exhibit such a solution?

- If so, the static feedback (34) obviously results in the closed-loop system of the form (16) with $\delta_i = \gamma_i$ $(i = 1, \ldots, m)$ and $[v_1, \ldots, v_m]^T = \boldsymbol{v}$.
- If not, the static feedback (34) fails. A sufficient condition to detect this problem will be stated below (Corollary 1).

In order to prepare the discussions, all row vectors $(N^{\gamma_i} c_i)'$ (cf. eq. (18)) and $(M^{\gamma_i} c_i)'$ (cf. eq. (19)) are combined

$$(N^{\boldsymbol{\gamma}}\boldsymbol{c})' := \begin{bmatrix} (N^{\gamma_1} c_1)' \\ \vdots \\ (N^{\gamma_m} c_m)' \end{bmatrix}, \; (M^{\boldsymbol{\gamma}}\boldsymbol{c})' := \begin{bmatrix} (M^{\gamma_1} c_1)' \\ \vdots \\ (M^{\gamma_m} c_m)' \end{bmatrix} \tag{36}$$

in the matrices $(N^{\boldsymbol{\gamma}}\boldsymbol{c})'$ and $(M^{\boldsymbol{\gamma}}\boldsymbol{c})'$, respectively.

Theorem 1 *The design equation (35) possesses no unique solution* $\boldsymbol{u}$ *if the matrix* $(M^{\boldsymbol{\gamma}}\boldsymbol{c})'$ *is singular.*

Proof: The algebraic variable $\boldsymbol{z}$ is only given implicitly by the algebraic equations whereas its derivative (13) is given explicitly. Therefore, the influence of $\boldsymbol{z}$ on $\boldsymbol{u}$ in eq.

(35) can be made explicit by differentiating this equation one more time. Starting from there with $\boldsymbol{v} = \mathbf{0}$ arbitrarily, relation (27) yields

$$\overset{(\gamma)}{\boldsymbol{y}} = \widehat{\boldsymbol{c}}(\boldsymbol{x},\boldsymbol{z}) + \widehat{\boldsymbol{D}}(\boldsymbol{x},\boldsymbol{z})\,\boldsymbol{u} = N^{\gamma}\boldsymbol{c} = \mathbf{0}$$

as a function of $\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}$. Another differentiation with respect to time results in:

$$\begin{aligned}
\overset{(\gamma+1)}{\boldsymbol{y}} &= \frac{\partial}{\partial \boldsymbol{x}}(N^{\gamma}\boldsymbol{c})\cdot\dot{\boldsymbol{x}} + \frac{\partial}{\partial \boldsymbol{z}}(N^{\gamma}\boldsymbol{c})\cdot\dot{\boldsymbol{z}} + \frac{\partial}{\partial \boldsymbol{u}}(N^{\gamma}\boldsymbol{c})\cdot\dot{\boldsymbol{u}} = \\
&= \cdots \text{cf. } (13) \cdots = \\
&= \left(\frac{\partial(N^{\gamma}\boldsymbol{c})}{\partial \boldsymbol{x}} - \frac{\partial(N^{\gamma}\boldsymbol{c})}{\partial \boldsymbol{z}}\left[\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}}\right]^{-1}\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{x}}\right)\dot{\boldsymbol{x}} + \\
&+ \left(\frac{\partial(N^{\gamma}\boldsymbol{c})}{\partial \boldsymbol{u}} - \frac{\partial(N^{\gamma}\boldsymbol{c})}{\partial \boldsymbol{z}}\left[\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{z}}\right]^{-1}\frac{\partial \boldsymbol{g}_{k-1}}{\partial \boldsymbol{u}}\right)\dot{\boldsymbol{u}} = \\
&= \cdots \text{cf. } (18), (19) \cdots = \\
&= (N^{\gamma}\boldsymbol{c})'\dot{\boldsymbol{x}} + (M^{\gamma}\boldsymbol{c})'\dot{\boldsymbol{u}} = \mathbf{0}
\end{aligned} \tag{37}$$

From here, Theorem 1 can be concluded from the fact (Krantz, 2002) that if the implicitly given function $\boldsymbol{u}(\boldsymbol{x})$ exists uniquely in eq. (35), it has a derivative

$$\boldsymbol{u}' = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}},$$

too. Note that if $\dot{\boldsymbol{u}}$ in (37) does not exist, than $\boldsymbol{u}'$ does not exist since $\dot{\boldsymbol{u}} = \boldsymbol{u}'\,\dot{\boldsymbol{x}}$ with $\dot{\boldsymbol{x}}$ existing. To proof Theorem 1 the following conclusion chain is taken ($(M^{\gamma}\boldsymbol{c})'$ is abbreviated by $\boldsymbol{M}$ for simplicity):

$$\neg\exists(\boldsymbol{M}^{-1}) \iff \neg\exists\dot{\boldsymbol{u}} \iff \neg\exists\boldsymbol{u}' \implies \neg\exists(\boldsymbol{u}\ \text{uniq.})$$

□

The following Corollary arises from Theorem 1:

Corollary 1 *In case 3a a static feedback does not exist if the matrix $(M^{\gamma}\boldsymbol{c})'$ (36) is singular.*

Remark 2 *If a singular matrix $(M^{\gamma}\boldsymbol{c})'$ causes a static feedback to be unfeasible the matrix $\widetilde{\boldsymbol{D}} = (M^{\gamma}\boldsymbol{c})'$ is suitable for a step in the dynamic case 2.*

Case 3b: The dynamic decoupling matrix $\widetilde{\boldsymbol{D}} = \mathbf{0}$ and the static decoupling matrix $\widehat{\boldsymbol{D}}$ has reduced rank $r < m$. If so, the matrix $\widetilde{\boldsymbol{D}} = (M^{\gamma}\boldsymbol{c})'$ is suitable for a step in the dynamic case 2.

4 Examples

Three examples will be discussed in the following to illustrate the presented design procedure. The first example is dedicated to the modified shuffle algorithm in handling functional dependencies. The second example deals with the design of a static feedback for a non-linear descriptor system resulting in a closed-loop system without algebraic loop problems. Finally, the third example picks up the increase in total index possibly caused by a static feedback.

4.1 Example: Shuffle Algorithm

The simple linear differential-algebraic equations

$$\begin{aligned}
\dot{x_1} &= x_2 + z_1 + z_2 \\
\dot{x_2} &= x_1 + x_2 + z_2 \\
\dot{x_3} &= x_2 \\
0 &= g_1 = x_1 - x_2 + x_3 \\
0 &= g_2 = x_1 - z_1
\end{aligned}$$

are constructed to show how the modified shuffle algorithm (8) - (12) handles a functional dependence while computing the equation indices k_1 and k_2 related to the algebraic equations g_1 and g_2.

In each recursion step (step indicator $j = 0, 1, \ldots$) the algorithm processes all the algebraic equations (equation indicator $i = 1, 2$) for which the indices have not been calculated yet. The algorithm starts with the initializing step $j = 0$ (the equations are marked by $g_{i,j}$):

$$\begin{aligned}
j = 0 : i = 1 &: g_{1,0} = g_1 = x_1 - x_2 + x_3 \\
i = 2 &: g_{2,0} = g_2 = x_1 - z_1 \\
j = 1 : i = 1 &: \frac{\partial g_{1,0}}{\partial \mathbf{z}} = \mathbf{0}^T \quad \overset{(9)}{\Longrightarrow} \quad k_1 > 1 \\
& \quad g_{1,1} \overset{(10)}{:=} \frac{\partial g_{1,0}}{\partial \mathbf{x}} \dot{\boldsymbol{x}} = [1 \quad -1 \quad 1] \dot{\boldsymbol{x}} = \\
& \quad = -x_1 + x_2 + z_1 \\
i = 2 &: \frac{\partial g_{2,0}}{\partial \mathbf{z}} = [-1 \quad 0] \neq \mathbf{0}^T \quad \text{and} \\
& \quad h_{2,1}^{(0)} \frac{\partial g_{2,0}}{\partial \mathbf{z}} = \mathbf{0}^T \overset{\text{only}}{\Longrightarrow} h_{2,1}^{(0)} = 0 \\
& \quad \overset{(9)}{\Longrightarrow} \quad \underline{\underline{k_2 = 1}}
\end{aligned}$$

$$\begin{aligned}
j = 2 : i = 1 : \; & \frac{\partial g_{1,1}}{\partial \mathbf{z}} = [1 \quad 0] \neq \mathbf{0}^T \quad \text{and} \\
& h_{1,2}^{(0)} \frac{\partial g_{1,1}}{\partial \mathbf{z}} - h_{1,2}^{(2)} \frac{\partial g_{2,0}}{\partial \mathbf{z}} = \\
& = h_{1,2}^{(0)} [1 \quad 0] - h_{1,2}^{(2)} [-1 \quad 0] = \mathbf{0}^T \overset{\text{also}}{\Longrightarrow} \\
& h_{1,2}^{(0)} = -h_{1,2}^{(2)} = 1 \neq 0 \quad \Longrightarrow \quad k_1 > 2
\end{aligned}$$

here a functional dependence is detected and therefore $j = 2$ is not the minimal j to determine k_1

$$\begin{aligned}
& g_{1,2} := \frac{\partial g_{1,1}}{\partial \mathbf{x}} \dot{\boldsymbol{x}} - \tilde{h}_{1,2}^{(2)} \frac{\partial g_{2,0}}{\partial \mathbf{x}} \dot{\mathbf{x}} = \\
& = [-1 \quad 1 \quad 0] \dot{\boldsymbol{x}} + [1 \quad 0 \quad 0] \dot{\boldsymbol{x}} = \dot{x}_2 \\
j = 3 : i = 1 : \; & \frac{\partial g_{1,2}}{\partial \mathbf{z}} = [0 \quad 1] \neq \mathbf{0}^T \quad \text{and} \\
& h_{1,3}^{(0)} \frac{\partial g_{1,2}}{\partial \mathbf{z}} - h_{1,3}^{(2)} \frac{\partial g_{2,0}}{\partial \mathbf{z}} = \\
& h_{1,3}^{(0)} [0 \quad 1] - h_{1,3}^{(2)} [-1 \quad 0] = \mathbf{0}^T \overset{\text{only}}{\Longrightarrow} \\
& h_{1,3}^{(0)} = h_{1,3}^{(2)} = 0 \quad \Longrightarrow \quad \underline{\underline{k_1 = 3}}
\end{aligned}$$

According to the definitions (11), (12) the algorithm yields the modified algebraic equations

$$\mathbf{0} = \boldsymbol{g}_{k-1} = \boldsymbol{g}_2 = \begin{bmatrix} g_{1,2} \\ g_{2,0} \end{bmatrix} = \begin{bmatrix} x_1 + x_2 + z_2 \\ x_1 - z_1 \end{bmatrix}$$

with a regular Jacobian matrix $\partial \boldsymbol{g}_2 / \partial \boldsymbol{z}$.

4.2 Example: Static Feedback

A pipe system to mix dry and humid gas is considered here (Fig. 3). Gas carrying a humidity h_1 as well as gas carrying a humidity h_2 are supplied at a pressure p_N. These two gases are mixed in order to produce a specified gas flow Q carrying a specified humidity h at a pressure p_O. The quantities Q and h are referred to as the system outputs, i.e. $y_1 := h$, $y_2 := Q$. The mixing procedure is effected by the opening rates α_1 and α_2 of two motor-driven valves V_1 and V_2, respectively. The opening rates are considered as the systems states, i.e. $x_1 := \alpha_1$, $x_2 := \alpha_2$. Finally, the motor commands u_1 and u_2 are quoted as the system inputs. The *state equations* result from a suitable drive model.
As a characteristic feature of the plant, the overall drop in pressure $\Delta p := p_N - p_O$ is very small (approx. hPa). Thus, the pressure drops along the pipes as well as the pressure drops caused by the valves are to be taken into account in order to achieve a sufficiently accurate mathematical description of the plant behaviour. Taking into account these pressure drops leads to the *algebraic* equations.

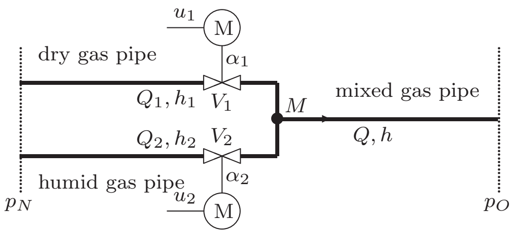


Figure 3: Gas mixing pipe system.

The *output* equations are obtained from the conservation of mass at the mixing point M regarding the gas and the water components, respectively.
Based on appropriate models for the pressure drops, the system description finally reads as follows:

$$\dot{\mathbf{x}} = \frac{1}{T}\mathbf{u}$$

$$\mathbf{0} = \mathbf{g}(\mathbf{x}, \mathbf{z}) = \begin{bmatrix} c_{P1}z_1^2 + c_V\psi(x_1)z_1^q + c_P(z_1+z_2)^2 - \Delta p \\ c_{P2}z_2^2 + c_V\psi(x_2)z_2^q + c_P(z_1+z_2)^2 - \Delta p \end{bmatrix}$$

$$\mathbf{y} = \mathbf{c}(\mathbf{z}) = \begin{bmatrix} \frac{h_1z_1+h_2z_2}{z_1+z_2} \\ z_1+z_2 \end{bmatrix}$$

Here the algebraic variables z_1 and z_2 represent the gas flow Q_1 and Q_2, respectively, the function $\psi(.)$, the power q and the coefficient c_V describing the valve characteristics and the constants c_P, c_{P1} and c_{P2} regarded as pipe pressure drop coefficients were found experimentally. T is a time constant concerning the motor drives.
Calculation of the linearizing and decoupling feedback:
1. *Equation indices k_1, k_2 and system index k:* The modified shuffle algorithm determines $k_1 = k_2 = 1$ relating to both algebraic equations. From relation (11) this results in the system index $k = 1$ so that equation (12) reads:

$$\mathbf{0} = \boldsymbol{g}_{k-1}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = \boldsymbol{g}_0(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{z})$$

2. *Derivative degrees γ_1, γ_2:* Using a computer algebra system it can easily be verified that subject to definitions (18)(19)

$$(N^0c_i)'\,\boldsymbol{B} \neq \mathbf{0}^T \quad \text{and} \quad (M^0c_i)' = \mathbf{0}^T \quad i = 1, 2$$

holds if $h_1 \neq h_2$ is assumed. Thus, $\gamma_1 = \gamma_2 = 1$ is evident from definition (22) and the output derivatives (21) read now:

$$\dot{y}_i = (N^0c_i)'\frac{1}{T}\boldsymbol{u}, \quad i = 1, 2$$

3. *Calculation of the static feedback:* Based on relation (34), straight forward calculations result in the static feedback:

$$\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{v}) = \widehat{\boldsymbol{D}}^{-1}(\boldsymbol{x}, \boldsymbol{z})\boldsymbol{v} = \begin{bmatrix} \rho_1 & -\varrho_1 \\ -\rho_2 & -\varrho_2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$\text{with} \quad \rho_i = T\frac{(z_1+z_2)(c_V q z_i^q \psi(x_i) + 2c_{Pi} z_i^2)}{c_V(h_2-h_1) z_i^{q+1}\psi'(x_i)}$$
$$\text{and} \quad \varrho_i = T\frac{2c_P(z_1+z_2)^2 + 2c_{Pi} z_i^2 + c_V q z_i^q \psi(x_i)}{c_V(z_1+z_2) z_i^q \psi'(x_i)}$$

There is no worry that the implementation of this feedback causes algebraic loops because (cf. Corollary 1)

$$\operatorname{rank}\left\{(M^{\boldsymbol{\gamma}}\boldsymbol{c})'\right\} = \operatorname{rank}\left\{\begin{bmatrix}(M^1 c_1)' \\ (M^1 c_2)'\end{bmatrix}\right\} = 2$$

holds if $h_1 \neq h_2$ and $Q_1, Q_2 > 0$ is assumed.
The closed-loop system possesses a linear decoupled input-output structure over the entire operating range ($0 < x_i \leq 1$); the channel dynamics are purely integrating described by the transfer functions $G_1 = G_2 = 1/s$.

4.3 Example: Algebraic Loop Caused by Static Feedback

It can easily be verified that the descriptor system

$$\begin{aligned} \dot{x}_1 &= f_1(x_1,x_2) + \alpha z + u \\ \dot{x}_2 &= f_2(x_1,x_2) + bu \\ 0 = g &= g_1(x_1,x_2) + z + \beta u \\ y &= x_1 \end{aligned}$$

has an index $k = 1$ (this holds for the equation index, the system index and the total index, too). Note that this example is to show the effect of an algebraic loop in the closed-loop system caused by static feedback.
In doing so, the first derivative of the output

$$\dot{y} = f_1(x_1,x_2) + \alpha z + u$$

yields the static feedback (34)

$$u = v - \alpha z - f_1$$

but the resulting overall system

$$\begin{aligned} \dot{x}_1 &= v \\ \dot{x}_2 &= f_2 + b(v - \alpha z - f_1) \\ 0 &= g_1 + (1-\alpha\beta)z + \beta(v - f_1) \\ y &= x_1 \end{aligned}$$

suffers an increase in index $k > 1$ if $\alpha\beta = 1$ applies to the system parameters; this leads to a non-proper algebraic variable $z = z(x_1, x_2, v, \dot{v})$. Note that the design equation (35) degenerates to

$$(1-\alpha\beta)u = v - f_1 - \alpha g$$

if the algebraic variable z is eliminated by use of the given algebraic equation of the system.
These problems are revealed by the formal procedure including definitions (18), (19), (22) and Corollary 1 (with $\boldsymbol{b} = [1 \;\; b]^T$):

$$\begin{array}{cc} (N^0c)'\boldsymbol{b} = 1, & (M^0c)' = 0 \;\rightarrow\; \gamma = 1 \\ \vdots & (M^1c)' = (M^\gamma c)' = 1 - \alpha\beta \end{array}$$

According to Theorem 1, the design equation (35) is not solvable if $\alpha\beta = 1$. If so, the proposed continuation of the design procedure (Remark 2) results in a dynamic feedback provided that $g_1(x_1, x_2) \neq f_1(x_1, x_2)$.

5 Conclusion

This article covers the linearization and decoupling of the input-output behaviour of time-invariant descriptor systems, the mathematical models of which are given in semi-explicit form. Firstly, we have defined the characteristics of regularity and properness, then we have explained why the requirements for these system characteristics cannot be generally regarded as a restriction. The definitions are based on an explicit descriptor model containing the basic differential equations (in the algebraic variables) required for the subsequent synthesis. The explicit descriptor model will be determined using the modified shuffle algorithm which was specified for the first time in (Müller, 2000).
By defining a recursive operator - taking the role of the Lie derivative for AI state space models - it is then possible to represent the design procedure in a compact form, and divide it into the design of a dynamic and a design of a static feedback. Specific attention has been directed to the possible existence of algebraic loops which can occur in the closed loop system when a static feedback is used. A sufficient condition allows the existence of such loops to be detected already in the design phase, and refers to the approach of a suitable design correction.

In addition to the methodic design of a feedback, its implementation has a practical meaning. The main aspects to be mentioned in this context are the robustness problem and the measuring problem. The literature does not yet seem to provide satisfactory approaches providing solutions for the first problem in connection with descriptor systems whereas possible solutions have already been developed for the latter problem: Either the control problem can be redefined as a tracking problem, even without exact linearization and decoupling where only measurable output variables can be fed back (Huang & Zhang, 1998), or non-measurable system variables are estimated by an observer (Aslund & Frisk, 2006).

References

J. Aslund and E. Frisk, "An observer for non-linear differential-algebraic systems," *Automatica*, vol. 42, pp. 959–965, 2006.

N. H. Clamroch, "Feedback stabilization of control systems described by a class of nonlinear differential-algebraic equations," *Systems & Control Letters*, vol. 15, pp. 53–60, 1990.

F. Gausch and P. Müller, "Statische und dynamische Rückführung in nichtlinearen Deskriptorsystemen (Static and dynamic feedback in nonlinear descriptor systems)," *Automatisierungstechnik*, vol. 52, pp. 569–576, 2004.

C. W. Gear, "Differential-algebraic equation index transformations," *SIAM J. Sci. Stat. Comp.*, vol. 9, pp. 39–47, 1988.

J. Huang and J-F. Zhang, "Impulse-free output regulation of singular nonlinear systems," *Int. J. Control*, vol. 71, no. 5, pp. 789–806, 1998.

A. Isidori and J. W. Grizzle, "Fixed modes and nonlinear noninteracting control with stability," *IEEE Trans. on AC*, vol. 33, no. 10, pp. 907–914, 1988.

S. Kawaji and E. Z. Taha, "Feedback linearization of a class of nonlinear descriptor systems," in *Proc. 33th Conference on Decision & Control*, Lake Buena Vista, FL, 1994, pp. 4035–4037.

S. G. Krantz and H. R. Parks, *The implicit function theorem - History, theory and applications*, Birkhäuser Boston, 2002.

D. G. Luenberger, "Time-invariant descriptor systems," *Automatica*, vol. 14, pp. 473–480, 1978.

P. C. Müller, "Stability and optimal control of nonlinear descriptor systems," *Proc. 3rd Int. Symp. on Methods and Models in Automation and Robotics*, Univ. of Szcecin, 1996, vol. 1, pp. 17–26.

P. Müller, *Linearisierung und Entkopplung von Deskriptorsystemen*, Aachen, Germany: Shaker, 2000.

S. Reich, "On a geometrical interpretation of differential-algebraic equations," *Circuits Systems Signal Process*, vol. 9, no. 4, pp. 367–382, 1990.

K. Schlacher and A. Kugi, "Control of nonlinear descriptor systems, a computer algebra based approach," in *Nonlinear Control in the Year 2000*, A. Isidori and W. Respondek, Eds., *Lecture Notes in Control and Information Sciences:*, vol. 259, pp. 379–395, 2001.

K. Schlacher, A. Kugi and K. Zehetleitner, "A Lie-group approach for nonlinear dynamic systems described by implicit ordinary differential equations," in *Proc. 15th Int. Symp. on Mathematical Theory of Networks and Systems (MTNS'02)*, Univ. of Notre Dame, 2001, 24255.pdf.

M. Vidyasagar, *Nonlinear Systems Analysis*, Englewood Cliffs, N.J.: Prentice-Hall, 1978.

L. Xiaoping and S. Čelikovský, "Feedback control of affine nonlinear singular control systems," *Int. J. Control*, vol. 68, no. 4, pp. 753–774.

A Laboratory Setup for Robust Control Experiments

Martin Horn, Stefan Doczy*, Nicolaos Dourdoumas†
Institute for Smart System Technologies
Control and Measurement Systems
Alpen-Adria University Klagenfurt
martin.horn@uni-klu.ac.at

Abstract: *This paper outlines the concept of an electromechanical laboratory experiment with time-dependent parameters. It consists of a rotating body on which a load with variable moment of inertia is attached via a flexible joint. As the system can be modeled by a set of linear, time-variant ordinary differential equations, it is well suited for many popular control system design methods. As an example the application of a design methodology based on linear matrix inequalities is demonstrated.*

1. Introduction

A number of laboratory experiments were designed to study the behaviour of parameter varying dynamic systems and to explore the power of robust control design techniques. Among them is the so-called rotary flexible joint experiment, which originally was developed by Quanser Consulting (*Quanser, 1995*). The purpose of the experiment is to control the angle of a swivelling body on which an arm is attached via a flexible joint. The load inertia can be changed by attaching an additional extra load to the main arm. A major drawback of the system is that only *time-invariant* parameter changes can be considered as the system must be turned off in order to modify the load inertia.

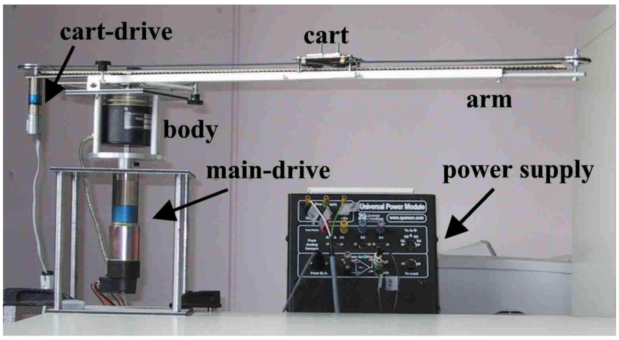


Figure 1.1: parameter variable flexible joint at Graz University of Technology

*Magna Steyr Fahrzeugtechnik, Graz
†Institute for Automation and Control, Graz University of Technology

A modification which immediately suggests itself is to replace the original arm by an arm with variable load inertia (*Doczy, S., 2000, 2001*). This is done by moving a small cart on the arm (see Figure 1.1). The cart is driven by a DC-motor so that the moment of inertia can be varied during operation and the dynamic behaviour of the system subject to *time-variant* parameter changes can be investigated.

Exemplarily linear state controllers achieving robust stabilization and tracking are designed.on the basis of semidifinite programming techniques. It is shown that the plant can be represented by an affine parameter dependent model so that the computation of a stabilizing state controller can be achieved e.g. by employing linear matrix inequalities (*Boyd, S., 1994*). In the present case the basic idea of the design is to find a common quadratic Lyapunov-function for "extremal" plant configurations. This yields nonlinear matrix inequalities which can easily be translated into linear matrix inequalities. The tracking performance is optimized by minimizing the $\mathcal{H}_\infty$-norm of the closed-loop tracking error transfer function which can be formulated as linear matrix inequalities as well.

The paper is organized as follows: Section 2 outlines the mathematical modeling of the plant. Sections 3 and 4 are dedicated to the design of controllers for robust stabilization and tracking, section 5 concludes the work.

2. Modeling

Considering the simplified model in Figure 2.1 the equations describing the motion of the system can easily be derived (*Horn M., 2004*). Let φ be the angle of the body and α the angle between body and the arm. The body can be rotated with the help of a DC-drive on which axis it is attached.

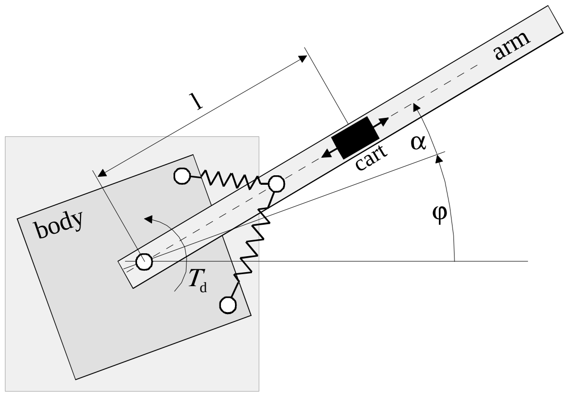


Figure 2.1: rotational flexible joint

Neglecting the reaction moment of the cart drive, the differential equations of motion are:

$$\begin{aligned} J_l\left(\ddot{\alpha}+\ddot{\varphi}\right)+\dot{J}_l\left(\dot{\alpha}+\dot{\varphi}\right) &= -k_s\alpha \\ J_l\left(\ddot{\alpha}+\ddot{\varphi}\right)+\dot{J}_l\left(\dot{\alpha}+\dot{\varphi}\right)+J_b\ddot{\varphi} &= T_d \end{aligned} \tag{1}$$

where J_b denotes inertia of the body, J_l is the load inertia and T_d is the torque of the electrical drive. The right side of the first differential equation represents the linearized restoring moment of the flexible joint, i.e. k_s denotes the joint stiffness. The time dependent load inertia J_l is composed of the constant arm inertia J_a and the inertia J_c of the movable cart with mass m_c, i.e.

$$J_l = J_a + J_c = J_a + m_c l^2 \;\Rightarrow\; \dot{J}_l = 2m_c l\dot{l}. \tag{2}$$

The distance between the cart and the axis of rotation is denoted by l. The torque T_d of the electrical drive is given by

$$T_d = \frac{k_d}{R_m} u - \frac{k_d^2}{R_m} \dot{\varphi}, \tag{3}$$

where k_d and R_m are the motor constant and the armature resistance respectively. The drive input voltage is denoted by u, the motor inductance was neglected in (3). The dynamics of the electrical drive which is used to move the cart can be neglected as well. The differential equations (1) together with (2) and (3) define a fourth order system. Introducing the state vector

$$\mathbf{x} := \left[\begin{array}{cccc} \varphi & \alpha & \dot{\varphi} & \dot{\alpha} \end{array} \right]^T \tag{4}$$

the system can be represented in the general form

$$\dot{\mathbf{x}} = \mathbf{A}(\boldsymbol{\delta})\mathbf{x} + \mathbf{b}\, u. \tag{5}$$

The matrix $\mathbf{A}$ is a function of the real valued parameter vector $\boldsymbol{\delta} = \left[\begin{array}{ccc} \delta_1 & \dots & \delta_k \end{array} \right]^T$ which represents the perturbations in the system parameters. In the present case the vector $\boldsymbol{\delta}$ has the dimension 2 and is made up of the time dependent inertia J_l and its time derivative $\dot{J}_l$. Table 1.1 shows the numerical values for the above introduced constants, $l_{\max}$ is the length of the arm while $\dot{l}_{\max}$ is the maximum speed of the cart.

$k_d = 0.9\,\mathrm{Nm/A}$	$J_b = 85799 \cdot 10^{-6}\,\mathrm{kg\,m^2}$
$k_s = 2.0\,\mathrm{Nm/rad}$	$J_a = 13618 \cdot 10^{-6}\,\mathrm{kg\,m^2}$
$l_{\max} = 0.5\,\mathrm{m}$	$m_c = 0.15\,\mathrm{kg}$
$\dot{l}_{\max} = 0.2\,\mathrm{m/s}$	$R_m = 1.34\ \Omega$

Table 1.1: numerical data for flexible joint

As the cart moves along the arm, the moment of inertia J_l varies within the bounds given by

$$13618 \cdot 10^{-6} \ \leq J_l \leq 51118 \cdot 10^{-6}\,\mathrm{kg\,m^2}\,, \tag{6}$$

the upper and lower bounds for $\dot{J}_l$ can be computed as

$$-30000 \cdot 10^{-6} \ \leq \dot{J}_l \leq 30000 \cdot 10^{-6}\,\mathrm{kg\,m^2/s}. \tag{7}$$

The state matrix $\mathbf{A}(\boldsymbol{\delta})$ can be modeled as an *affine parameter dependent matrix*, i.e.

$$\mathbf{A}(\boldsymbol{\delta}) = \mathbf{A}_0 + \mathbf{A}_1\delta_1 + \mathbf{A}_2\delta_2 \tag{8}$$

where δ_1 and δ_2 are defined as

$$\delta_1 := \frac{k_s}{J_l},\ \delta_2 := \frac{\dot{J}_l}{J_l} \tag{9}$$

and vary within the prescribed *conservative* ranges[1]

$$\begin{aligned} \delta_1 &\in [\delta_{1,\min}, \delta_{1,\max}] = [39.12, 146.86]\,, \\ \delta_2 &\in [\delta_{2,\min}, \delta_{2,\max}] = [-2.20, 2.20]\,. \end{aligned} \tag{10}$$

The constant matrices $\mathbf{A}_0$, $\mathbf{A}_1$ and $\mathbf{A}_2$ are given by

$$\begin{aligned} \mathbf{A}_0 &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 23.3100 & -7.0450 & 0 \\ 0 & -23.3100 & 7.0450 & 0 \end{bmatrix}, \\ \mathbf{A}_1 &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}, \; \mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \end{aligned} \tag{11}$$

Note that due to the quite conservative bounds for the parameter $\boldsymbol{\delta}$ eigenvalues of $\mathbf{A}(\boldsymbol{\delta})$ can move across the imaginary axis in theory. In Figure 2.2 the time responses of the angles $\alpha(t)$ and $\varphi(t)$ of the uncontrolled system for $\mathbf{x}_0 = \begin{bmatrix} 0 & 0.5 & 0 & 0 \end{bmatrix}^T$ are depicted. The given inertia $J_c(t)$ results from moving the cart with maximum speed, starting at the outer end of the arm.

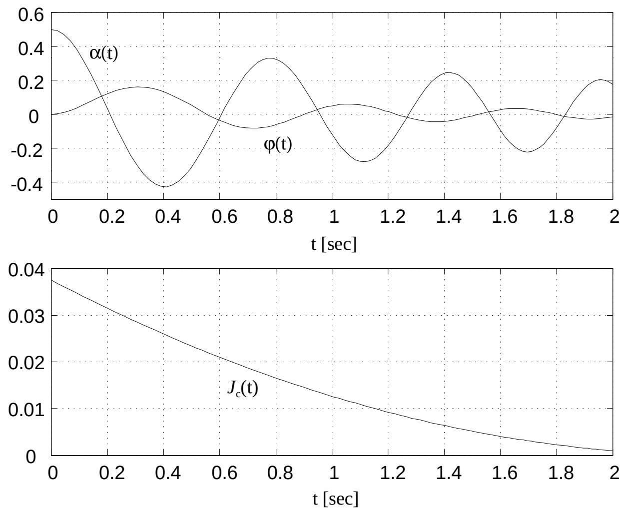


Figure 2.2: $\alpha(t)$, $\varphi(t)$ and cart inertia $J_c(t)$

Given the parameter dependent model (8) for the plant, a number of techniques dedicated to the design of robust control systems can be applied.

3. Stabilization

In the first step the stabilization of the system under investigation is presented. Assuming a linear state controller, i.e.

$$u = \mathbf{k}^T \mathbf{x} \tag{12}$$

[1] This conservative estimation is based on the assumption that J_l and $\dot{J}_l$ can be chosen independently, which is not true in practice.

the controlled system (5) becomes:

$$\dot{\mathbf{x}} = \mathbf{A}(\boldsymbol{\delta})\mathbf{x} + \mathbf{b}\mathbf{k}^T\mathbf{x} = \mathbf{A}_{cl}(\boldsymbol{\delta})\mathbf{x}, \tag{13}$$

where the state matrix of the closed loop system is given by:

$$\mathbf{A}_{cl}(\boldsymbol{\delta}) = \mathbf{A}_0 + \mathbf{b}\mathbf{k}^T + \delta_1\mathbf{A}_1 + \delta_2\mathbf{A}_2. \tag{14}$$

Obviously (13) is an *affine parameter dependent model* as well. Our goal is to determine the controller gain $\mathbf{k}$ such that the closed loop system has an asymptotically stable equlibrium $\mathbf{x} = \mathbf{0}$ for all possible perturbations $\boldsymbol{\delta}$. The solution is based on the following theorem (*Scherer, C., 1999*).

The system (13) is asymptotically stable if there exists a symmetric positive definite matrix $\mathbf{Q} = \mathbf{Q}^T \succ 0$ such that

$$\mathbf{Q}\mathbf{A}_{cl}^T(\boldsymbol{\delta}) + \mathbf{A}_{cl}(\boldsymbol{\delta})\mathbf{Q} \prec 0 \tag{15}$$

for

$$\boldsymbol{\delta} \in \left\{ \begin{bmatrix} \delta_{1,\min} \\ \delta_{2,\min} \end{bmatrix}, \begin{bmatrix} \delta_{1,\min} \\ \delta_{2,\max} \end{bmatrix}, \begin{bmatrix} \delta_{1,\max} \\ \delta_{2,\min} \end{bmatrix}, \begin{bmatrix} \delta_{1,\max} \\ \delta_{2,\max} \end{bmatrix} \right\},$$

where the symbol " $\prec 0$" denotes negative definiteness. Thus in order to guarantee stability for *all possible* parameter values $\boldsymbol{\delta}$ is suffices to satisfy condition (15) for only *four* special parameter combinations. Note that no restrictions on the rate of change of the perturbations are imposed by the above conditions, i.e. the parameters δ_1 and δ_2 may change arbitrarily fast! By introducing the abbreviations for the four "corner" matrices

$$\begin{aligned}
\mathbf{A}_{c,1} &:= \mathbf{A}_0 + \delta_{1,\min}\mathbf{A}_1 + \delta_{2,\min}\mathbf{A}_2, \\
\mathbf{A}_{c.2} &:= \mathbf{A}_0 + \delta_{1,\min}\mathbf{A}_1 + \delta_{2,\max}\mathbf{A}_2, \\
\mathbf{A}_{c.3} &:= \mathbf{A}_0 + \delta_{1,\max}\mathbf{A}_1 + \delta_{2,\min}\mathbf{A}_2, \\
\mathbf{A}_{c.4} &:= \mathbf{A}_0 + \delta_{1,\max}\mathbf{A}_1 + \delta_{2,\max}\mathbf{A}_2,
\end{aligned}$$

the stability condition (15) for the affine parameter dependent model can be rewritten as:

$$\begin{gathered}
\mathbf{Q} \succ 0, \\
\mathbf{Q}(\mathbf{A}_{c,i} + \mathbf{b}\mathbf{k}^T)^T + (\mathbf{A}_{c,i} + \mathbf{b}\mathbf{k}^T)\mathbf{Q} \prec 0, \\
i = 1, 2, 3, 4.
\end{gathered} \tag{16}$$

A suitable matrix $\mathbf{Q}$ and a vector $\mathbf{k}$ which satisfy condition (16) can be regarded as a feasible solution to a system of *nonlinear matrix inequalities*.

In addition to the mere stability of the system we wish to force the closed loop eigenvalues to lie inside a circle located in the complex left halfplane (see Figure 3.1).

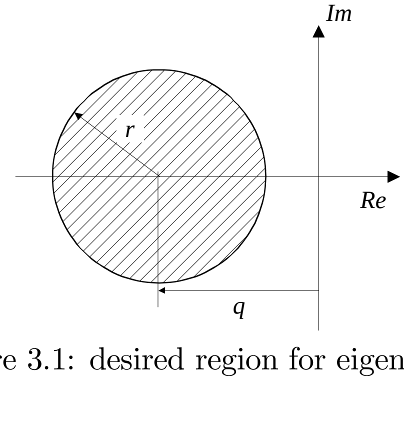


Figure 3.1: desired region for eigenvalues

As it is shown in (*Boyd, S., 1994*) this eigenvalue placement problem can also be treated within the same theoretical framework. For this purpose (16) must be replaced by the following condition:

$$\begin{gathered} \mathbf{Q} \succ 0 \\ \begin{bmatrix} -r\mathbf{Q} & (q\mathbf{I} + \mathbf{A}_{c,i} + \mathbf{b}\mathbf{k}^T)\mathbf{Q} \\ \mathbf{Q}(q\mathbf{I} + \mathbf{A}_{c,i}^T + \mathbf{k}\mathbf{b}^T) & -r\mathbf{Q} \end{bmatrix} \prec 0, \\ i = 1, 2, 3, 4. \end{gathered} \tag{17}$$

where $r, q > 0$. As this problem is nonlinear in the variables $\mathbf{k}$ and $\mathbf{Q}$ as well it is useful to define the auxiliary vector $\mathbf{m}$ as

$$\mathbf{m} := \mathbf{Q}\mathbf{k}, \tag{18}$$

so that the final stabilization and eigenvalue placement problem can be formulated as follows:

Find a matrix $\mathbf{Q}$ and a vector $\mathbf{m}$ such that the conditions

$$\begin{gathered} \mathbf{Q} \succ 0 \\ \begin{bmatrix} -r\mathbf{Q} & q\mathbf{Q} + \mathbf{A}_{c,i}\mathbf{Q} + \mathbf{b}\mathbf{m}^T \\ q\mathbf{Q} + \mathbf{Q}\mathbf{A}_{c,i}^T + \mathbf{m}\mathbf{b}^T & -r\mathbf{Q} \end{bmatrix} \prec 0, \\ i = 1, 2, 3, 4. \end{gathered} \tag{19}$$

hold.

Now (19) represents a system of linear matrix inequalities (LMI) in the variables $\mathbf{Q}$ and $\mathbf{m}$. Problems of this type can be solved by powerful algorithms which have been developed during the last decade (*El Ghaoui, L., 2000*). Having found a feasible solution the sought vector $\mathbf{k}$ can easily be computed as

$$\mathbf{k} = \mathbf{Q}^{-1}\mathbf{m}.$$

In our application the values

$$q = 20,\ \ r = 19$$

were chosen for the eigenvalue region in Figure 3.1. The feasibility problem (19) was solved with the Matlab LMI-toolbox (*Gahinet, P., 1995*) yielding the solution

$$\mathbf{k}^T = \begin{bmatrix} -12.2925 & 162.1516 & -15.1540 & -7.6227 \end{bmatrix}.$$

Figure 3.2 demonstrates the improvement of the system behavior using state feedback. It shows the same experiment as Figure 3, i.e. the cart is moved towards the axis of rotation with maximum speed and the initial condition is chosen $\mathbf{x}_0 = \begin{bmatrix} 0 & 0.5 & 0 & 0 \end{bmatrix}^T$.

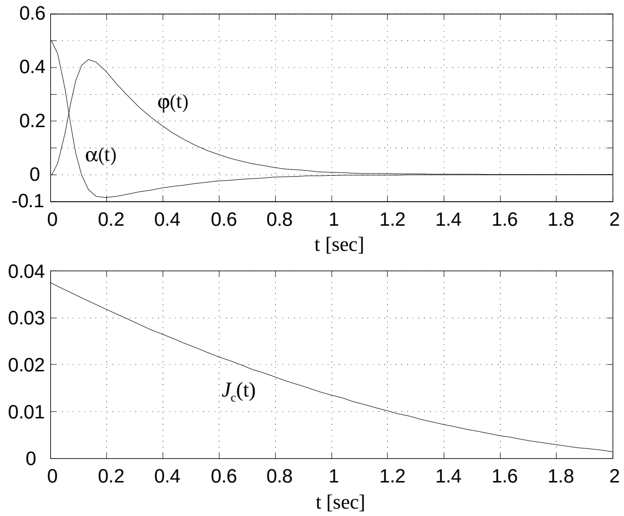


Figure 3.2: $\alpha(t)$, $\varphi(t)$ and cart inertia $J_c(t)$

4. Design for Tracking

Now a second design procedure is outlined. In addition to the stabilization of the plant it addresses the tracking performance of the feedback system. The purpose of the tracking experiment is to make the tip angle of the load

$$y(t) := \alpha(t) + \varphi(t) = \mathbf{c}^T\mathbf{x} \tag{20}$$

follow a reference signal $r(t)$ with vanishing steady state error. This goal can be achieved by adding an integrator to the plant input (*Horn M., 2004*), see Figure 4.1

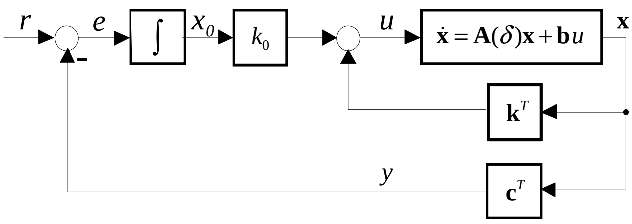


Figure 4.1: augmented feedback system

The new state variable is denoted by x_0. As shown in Figure 6 the controller to be determined is again a linear state controller, i.e.

$$u = \mathbf{k}^T\mathbf{x} + k_0 x_0. \tag{21}$$

By introducing the new state vector

$$\mathbf{z} := \left[\begin{array}{cc} \mathbf{x}^T & x_0 \end{array}\right]^T \tag{22}$$

the overall system can be rewritten as

$$\begin{aligned}\dot{\mathbf{z}} &= \underbrace{\begin{bmatrix} \mathbf{A}(\boldsymbol{\delta}) & \mathbf{0} \\ -\mathbf{c}^T & 0 \end{bmatrix}}_{=:\tilde{\mathbf{A}}(\delta)} \mathbf{z} + \underbrace{\begin{bmatrix} \mathbf{b} \\ 0 \end{bmatrix}}_{=:\tilde{\mathbf{b}}_u} u + \underbrace{\begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}}_{=:\tilde{\mathbf{b}}_r} r, \qquad (23)\\ u &= \underbrace{\begin{bmatrix} \mathbf{k}^T & k_0 \end{bmatrix}^T}_{=:\tilde{\mathbf{k}}^T} \mathbf{z}.\end{aligned}$$

Using (8) the state matrix $\tilde{\mathbf{A}}(\boldsymbol{\delta})$ can be represented as an affine parameter dependent matrix:

$$\tilde{\mathbf{A}}(\boldsymbol{\delta}) = \underbrace{\begin{bmatrix} \mathbf{A}_0 & \mathbf{0} \\ -\mathbf{c}^T & 0 \end{bmatrix}}_{=:\tilde{\mathbf{A}}_0} + \delta_1 \underbrace{\begin{bmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}}_{=:\tilde{\mathbf{A}}_1} + \delta_2 \underbrace{\begin{bmatrix} \mathbf{A}_2 & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}}_{=:\tilde{\mathbf{A}}_2}. \qquad (24)$$

Clearly there are many different ways of specifying criteria for the performance of a tracking control system. A reasonable approach is to minimize the $\mathcal{H}_\infty$-norm of the transfer function from the reference input r to the tracking error e (see Figure 4.1). This means that we want to determine the state controller such that the energy of the error signal is minimized for all reference input signals with energy bounded by 1. If we denote this transfer function by $W(s)$ its $\mathcal{H}_\infty$-norm can be computed by the relation

$$||W||_\infty = \sup_\omega |W(j\omega)|. \qquad (25)$$

In the present case this design goal should be met for a *nominal* plant, which is defined for "average" $\boldsymbol{\delta}$-values in the intervals specified by (10), i.e.

$$\tilde{\mathbf{A}}_n := \tilde{\mathbf{A}}|_{\substack{\delta_1=92.99 \\ \delta_2=0}}. \qquad (26)$$

Considering Figure 6 the tracking error e is given by

$$e = r - y = \underbrace{\begin{bmatrix} -\mathbf{c}^T & 0 \end{bmatrix}}_{=:\tilde{\mathbf{c}}^T} \mathbf{z} + r, \qquad (27)$$

and the nominal error transfer function can be computed as

$$W(s) = \tilde{\mathbf{c}}^T \left(s\mathbf{I} - \tilde{\mathbf{A}}_n - \tilde{\mathbf{b}}_u \tilde{\mathbf{k}}^T \right) \tilde{\mathbf{b}}_r + 1. \qquad (28)$$

The solution of the tracking design is essentially based on the the so-called *bounded real lemma* (*Boyd, S., 1991*) which in the present case can be formulated as follows:

The closed loop system (with the nominal plant model determined by $\tilde{\mathbf{A}}_n$) is asymptotically stable and the inequality

$$||W||_\infty \le \gamma. \qquad (29)$$

holds, iff there exist $\mathbf{P}$, $\tilde{\mathbf{k}}$ and $\gamma > 0$ such that the matrix inequalities

$$\mathbf{P} \succ 0 \qquad (30)$$

$$\begin{bmatrix} \mathbf{P}\tilde{\mathbf{A}}_n^T + \tilde{\mathbf{A}}_n \mathbf{P} + \mathbf{P}\tilde{\mathbf{k}}\tilde{\mathbf{b}}_u^T + \tilde{\mathbf{b}}_u \tilde{\mathbf{k}}^T \mathbf{P} & \tilde{\mathbf{b}}_r & \mathbf{P}\tilde{\mathbf{c}} \\ \tilde{\mathbf{b}}_r^T & -\gamma^2 & 1 \\ \tilde{\mathbf{c}}^T \mathbf{P} & 1 & -1 \end{bmatrix} \prec 0$$

are satisfied.

Thus our problem is to minimize γ^2 subject to (30). In addition to the *nominal performance* the *robust stability* of the parameter dependent system is desired, i.e. the eigenvalues of the closed loop system should be placed in the circle shown in Figure 4. Defining again the four "corner" matrices

$$\begin{aligned}
\tilde{\mathbf{A}}_{c,1} &:= \tilde{\mathbf{A}}_0 + \delta_{1,\min}\tilde{\mathbf{A}}_1 + \delta_{2,\min}\tilde{\mathbf{A}}_2, \\
\tilde{\mathbf{A}}_{c.2} &:= \tilde{\mathbf{A}}_0 + \delta_{1,\min}\tilde{\mathbf{A}}_1 + \delta_{2,\max}\tilde{\mathbf{A}}_2, \\
\tilde{\mathbf{A}}_{c,3} &:= \tilde{\mathbf{A}}_0 + \delta_{1,\max}\tilde{\mathbf{A}}_1 + \delta_{2,\min}\tilde{\mathbf{A}}_2, \\
\tilde{\mathbf{A}}_{c,4} &:= \tilde{\mathbf{A}}_0 + \delta_{1,\max}\tilde{\mathbf{A}}_1 + \delta_{2,\max}\tilde{\mathbf{A}}_2,
\end{aligned}$$

the robust stability condition can be formulated analogously to (17) as

$$\begin{gathered}
\mathbf{Q} \succ 0 \\
\begin{bmatrix} -r\,\mathbf{Q} & (q\mathbf{I} + \tilde{\mathbf{A}}_{c,i} + \tilde{\mathbf{b}}_u\tilde{\mathbf{k}}^T)\mathbf{Q} \\ \mathbf{Q}(q\mathbf{I} + \tilde{\mathbf{A}}_{c,i}^T + \tilde{\mathbf{k}}\mathbf{b}_u^T) & -r\,\mathbf{Q} \end{bmatrix} \prec 0. \\
i = 1,2,3,4.
\end{gathered} \tag{31}$$

The complete design procedure consists of minimizing γ^2 subject to the constraints (30) and (31). In order to obtain *linear matrix inequalities* we have to introduce the artificial restriction

$$\mathbf{Q} = \mathbf{P},$$

which again permits the substitution

$$\tilde{\mathbf{m}} := \mathbf{P}\tilde{\mathbf{k}}. \tag{32}$$

Finally the resulting design problem is a *semidefinite program*, i.e. an optimization problem with a linear objective function and linear matrix inequalities as constraints:

$$\begin{gathered}
\text{minimize } \gamma^2 \\
\text{subject to} \\
\mathbf{P} \succ 0 \\
\begin{bmatrix} \mathbf{P}\tilde{\mathbf{A}}_n^T + \tilde{\mathbf{A}}_n\mathbf{P} + \tilde{\mathbf{m}}\tilde{\mathbf{b}}_u^T + \tilde{\mathbf{b}}_u\tilde{\mathbf{m}}^T & \tilde{\mathbf{b}}_r & \mathbf{P}\tilde{\mathbf{c}} \\ \tilde{\mathbf{b}}_r^T & -\gamma^2 & 1 \\ \tilde{\mathbf{c}}^T\mathbf{P} & 1 & -1 \end{bmatrix} \prec 0 \\
\begin{bmatrix} -r\,\mathbf{P} & q\mathbf{P} + \tilde{\mathbf{A}}_{c,i}\mathbf{P} + \tilde{\mathbf{b}}_u\tilde{\mathbf{m}}^T \\ q\mathbf{P} + \mathbf{P}\tilde{\mathbf{A}}_{c,i}^T + \tilde{\mathbf{m}}\mathbf{b}_u^T & -r\,\mathbf{P} \end{bmatrix} \prec 0 \\
i = 1,2,3,4.
\end{gathered} \tag{33}$$

The unknowns of the design problem are the matrix $\mathbf{P}$, the vector $\tilde{\mathbf{m}}$ and the real number γ^2, the vector $\tilde{\mathbf{k}}$ is given by

$$\tilde{\mathbf{k}} = \mathbf{P}^{-1}\tilde{\mathbf{m}}.$$

For an eigenvalue region defined by

$$r = 67.5,\ q = 70$$

the optimal solution to problem (33) yields a value of

$$\gamma = 1.6218$$

and the sought state controller (21) is found as

$$\begin{aligned}\mathbf{k}^T &= \begin{bmatrix} -4.468 & -2.335 & -0.666 & -0.639 \end{bmatrix} \cdot 10^3, \\ k_0 &= 8.8114 \cdot 10^3.\end{aligned}$$

In Figure 4.2 the tip angle $y(t)$ is shown for a square wave reference signal $r(t)$, the initial condition is $\mathbf{z}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$ and the cart is moving in and out on the load arm with maximum speed.

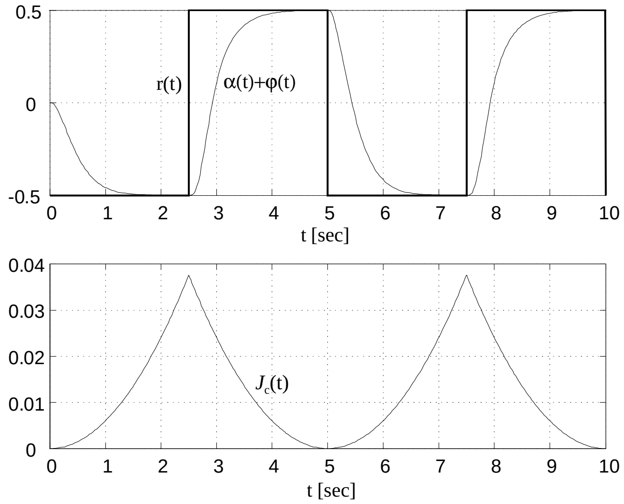


Figure 4.2: $\alpha(t)$, $\varphi(t)$ and cart inertia $J_c(t)$

5. Conclusion

The presented laboratory experiment is a valuable tool for implementing and testing robust control system design techniques. Especially the fact that parameters can be changed during operation makes it superior to many other laboratory plants. A number of improvements to the prototype will be necessary in the future. In order to make the plant more attractive it is planned to increase the mass and the speed of the cart. Besides robust and adaptive control the application of game-theoric approaches (*Basar, T., 1991*) are fields of interest, as the controller and the cart position can be interpreted as two competing players.

References

Basar T. and Bernhard P., 1991: $\mathcal{H}_\infty$ - Optimal Control and Related Minimax Design Problems - A Dynamic Game Approach, Birkhäuser Boston

Boyd S.P. and Barrat C.H., 1991: Linear Controller Design, Prentice Hall

Boyd S., El Ghaoui L., Feron E. and Balakrishnan V., 1994: Linear Matrix Inequalities in System and Control Theory, SIAM Studies in Applied Mathematics, vol.15, 1994

Doczy S., 2000: Control of parameter varying systems, PhD-thesis, Graz University of Technology

Doczy S., 2001: Robuste Regelung eines parameterabhängigen Übertragungssystems, 12. Steirisches Seminar über Regelungstechnik und Prozessautomatisierung, pp. 155 - 180

El Ghaoui L. and Niculescu S. (Eds.), 2000: Advances in Linear Matrix Inequality Methods in Control, SIAM Advances in Design and Control

Gahinet P., Nemirovski A., Laub A.J. and Chilali M., 1995: LMI Control Toolbox, The MathWorks Inc., Natick MA

Horn M., Dourdoumas N., 2004: Regelungstechnik, Pearson

Scherer C. and Weiland S., 1999: Lecture Notes DISC Course on Linear Matrix Inequalities, Version 2.0

Quanser Consulting, 1995: A Comprehensive and Modular Laboratory for Control Systems Design and Implementation

Desensitizing Flat Linear Systems

Alexander Weinmann, OVE, Senior Member IEEE
Vienna University of Technology, Institute of Automation and Control
Gusshausstrasse 27-29/376, A-1040 Vienna / Austria
Phone: +43 1 58801 37611, Fax: +43 1 58801 37699
email: weinmann@acin.tuwien.ac.at

Manuscript received January 17, 2010

Abstract

The paper is addressed to the possibilities of desensitizing the parameters of the flat output of linear single-input single-output systems and some other functions related to flatness. For uncertain system parameters, the feasibility is presupposed that some other certain parameters of the plant or the actuating device can be modified in the construction phase. For feedforward control based on the flat output, the desensitized system forces its output closer to the desired nominal target.

Keywords: Flatness properties, feedforward control, uncertainties, single-input single-output linear systems, traditional flatness

1 Introduction

In industrial applications, the differentially flat output is very useful for generating smooth transients which optionally comply with the actuating variable bounds. For single-input single-output linear systems, a concise definition of the flat output is available which is based on the controllability matrix (*Rothfuß, R., et al., 1997; Zeitz, M., 2009*).

However, the flat output changes if uncertainties arise. Hence, for obvious reasons, it is desirable to keep the output y free of uncertainty influence as much as possible, and/or the flat output parameters as unchanged as possible by actions considering the plant or the actuating system design. Although the flat output is not an exclusive design option, keeping the flat output parameters constant is an advantageous issue.

We consider a single-input single-output linear system $\mathbf{x} \in \mathcal{R}^n$; $\mathbf{A} \in \mathcal{R}^{n\times n}$; $\mathbf{b} \in \mathcal{R}^n$; $\mathbf{c} \in \mathcal{R}^n$; $y \in \mathcal{R}^1$; $u \in \mathcal{R}^1$. The influence via the input vector $\mathbf{b}$ and a state controller cannot be utilized because its action is located in the null space. However, several opportunities can be exhausted to set some design options in the construction phase to indiviually change parameters in an optimal way.

In more detail, this paper addresses the problem

- of carrying out the sensitivity of flatness functions with respect to system parameters,
- of detecting the uncertainty locations which are of largest influence, and
- of achieving information which parameter of $\mathbf{A}$ or $\mathbf{b}$ should be changed in order to find the minimum uncertainty influence on the flat output or some other performance item.

In *Sira-Ramirez, H., and Agrawal, S.K., 2004,* perturbations are carried out by assuming additive-type perturbations signals. In this paper, low system sensitivities with respect to uncertainties are considered as a replacement for robustness activities.

We suppose a dynamic system with a state space representation the parameters of which correspond to real-world data (mass, friction, capacity, inductance). We do not consider given input output data and an arbitrary state space representation.

In Section 2, 3 and 4 gradients of the controllability matrix, the flat output and the modal flatness matrix are considered, respectively. Second-order sensitivity matrices are given in Section 5. An incremental representation is presented in Section 6. The output sensitivity, as far as mainly the feedforward control is concerned, is introduced in Section 7. In Section 8, a short representation of input vector related sensitivity is given. Section 9 is dedicated to a comprehensive example. In the Appendix, basic facts about flatness including an example are recalled. Eventually, some appropriate basic matrix algebra and analysis is presented.

2 Controllability Matrix Gradients

For subsequent use, the controllability matrix (*Franklin, G.F., et al., 2002*) and corresponding products

$$\mathbf{L}_c = [\mathbf{b} \vdots \mathbf{A}\mathbf{b} \vdots \ \ldots \ \vdots \mathbf{A}^{i-1}\mathbf{b}] = \sum_{i=1}^{i=n} \mathbf{e}_i^T \otimes (\mathbf{A}^{i-1}\mathbf{b}) \quad \in \mathcal{R}^{n\times n} \tag{1}$$

$$\mathbf{L}_c^T\mathbf{L}_c = \sum_{i=1}^{i=n}\sum_{k=1}^{k=n}(\mathbf{e}_i\mathbf{e}_k^T)\otimes(\mathbf{b}^T\mathbf{A}^{T,i-1}\mathbf{A}^{k-1}\mathbf{b}) \tag{2}$$

$$\mathbf{L}_c\mathbf{L}_c^T = \sum_{i=1}^{i=n}\mathbf{A}^{i-1}\mathbf{b}\mathbf{b}^T\mathbf{A}^{T,i-1} \tag{3}$$

are considered as far as their derivatives are concerned

$$\frac{\partial\mathbf{L}_c}{\partial\mathbf{k}_r} \overset{(1)}{=} \sum_{k=1}^{n}\frac{\partial}{\partial\mathbf{k}_r}[\mathbf{e}_k^T\otimes(\mathbf{A}^{k-1}\mathbf{b})] \tag{4}$$

$$\overset{(136)}{=} \sum_{k=1}^{n}(\mathbf{I}_n\otimes\mathbf{U}_{1,k})(\frac{\partial\mathbf{A}^{k-1}\mathbf{b}}{\partial\mathbf{k}_r}\otimes\mathbf{e}_k^T)\mathbf{U}_{1,n} \tag{5}$$

$$\overset{(110)}{=} \sum_{k=1}^{n}\mathbf{I}_{n^2}[(\frac{\partial\mathbf{A}^{k-1}}{\partial\mathbf{k}_r}\{\mathbf{I}_s\otimes\mathbf{b}\})\otimes\mathbf{e}_k^T]_{k\geq 2} \quad \text{and since } k=1 \text{ yields zero:} \tag{6}$$

$$\overset{(129)}{=} \sum_{k=2}^{n}\sum_{i=1}^{k-1}[\{\mathbf{I}_n\otimes\mathbf{A}^{i-1}\}\frac{\partial\mathbf{A}}{\partial\mathbf{k}_r}\mathbf{A}^{k-1-i}\mathbf{b}]\otimes\mathbf{e}_k^T \quad \text{and for } \mathbf{A}=\mathbf{A}_o+\mathbf{b}\mathbf{k}_r^T \; : \tag{7}$$

$$\overset{(133)}{=} \sum_{k=2}^{n}\sum_{i=1}^{k-1}[\{\mathbf{I}_n\otimes\mathbf{A}^{i-1}\mathbf{b}\}\mathbf{A}^{k-1-i}\mathbf{b}]\otimes\mathbf{e}_k^T \;, \tag{8}$$

$$\frac{\partial\mathbf{L}_c^T}{\partial\mathbf{k}_r} = \sum_{k=1}^{n}\frac{\partial}{\partial\mathbf{k}_r}[\mathbf{e}_k\otimes(\mathbf{b}^T\mathbf{A}^{T,k-1})] \tag{9}$$

$$= \sum_{k=2}^{n}\sum_{i=1}^{k-1}[(\mathbf{I}_n\otimes\{\mathbf{b}^T\mathbf{A}^{T,i-1}\})\bar{\mathbf{U}}_{n,1}\mathbf{b}^T\mathbf{A}^{T,k-1-i}]\otimes\mathbf{e}_k \tag{10}$$

Derivative of the controllability matrix with respect to an element of the coefficient matrix $A_{\nu\mu}$:

$$\frac{\partial\mathbf{L}_c}{\partial A_{\nu\mu}} = \sum_{k=1}^{n}\frac{\partial}{\partial A_{\nu\mu}}[\mathbf{e}_k^T\otimes(\mathbf{A}^{k-1}\mathbf{b})] \tag{11}$$

$$\overset{(136)}{=} \sum_{k=1}^{n}(\mathbf{I}_1\otimes\mathbf{U}_{1,n})(\frac{\partial\mathbf{A}^{k-1}\mathbf{b}}{\partial A_{\nu\mu}}\otimes\mathbf{e}_k^T)\cdot 1 \tag{12}$$

$$\overset{(110)}{=} \sum_{k=1}^{n}1\cdot[\frac{\partial\mathbf{A}^{k-1}}{\partial A_{\nu\mu}}(\mathbf{I}_1\otimes\mathbf{b})]\otimes\mathbf{e}_k^T) \tag{13}$$

$$\overset{(129)}{=} \sum_{k=1}^{n}\sum_{i=1}^{k-1}(\mathbf{A}^{i-1}\frac{\partial\mathbf{A}}{\partial A_{\nu\mu}}\mathbf{A}^{k-i-1}\mathbf{b})\otimes\mathbf{e}_k^T \tag{14}$$

$$\frac{\partial\mathbf{L}_c}{\partial A_{\nu\mu}} = \sum_{k=2}^{n}\sum_{i=1}^{k-1}(\mathbf{A}^{i-1}\mathbf{E}_{\nu,\mu}\mathbf{A}^{k-i-1}\mathbf{b})\otimes\mathbf{e}_k^T \tag{15}$$

$$\frac{\partial\mathbf{L}_c}{\partial\mathbf{A}} = \sum_{\nu=1}^{n}\sum_{\mu=1}^{n}\mathbf{E}_{\nu,\mu}\otimes\left[\sum_{k=2}^{n}\sum_{i=1}^{k-1}\mathbf{A}^{i-1}\mathbf{E}_{\nu,\mu}\mathbf{A}^{k-i-1}\mathbf{b})\otimes\mathbf{e}_k^T\right] \;, \tag{16}$$

$$\frac{\partial \mathbf{L}_c^T}{\partial A_{\nu\mu}} = \frac{\partial}{\partial A_{\nu\mu}} \sum_{i=1}^{n} \mathbf{e}_i \otimes (\mathbf{A}^{i-1}\mathbf{b})^T \overset{(138)}{=} \sum_{i=1}^{n} \frac{\partial \mathbf{b}^T \mathbf{A}^{T,i-1}}{\partial A_{\nu\mu}} \otimes \mathbf{e}_i \tag{17}$$

$$\frac{\partial \mathbf{L}_c^T}{\partial A_{\nu\mu}} \overset{(131)}{=} \sum_{i=2}^{n} (\sum_{v=1}^{v=i-1} \mathbf{b}^T \mathbf{A}^{T,v-1} \mathbf{E}_{\nu,\mu} \mathbf{A}^{T,i-v-1}) \otimes \mathbf{e}_i \ . \tag{18}$$

$$\frac{\partial \mathbf{L}_c \mathbf{L}_c^T}{\partial A_{\nu\mu}} \overset{(3)}{=} \sum_{k=1}^{n} \frac{\partial \mathbf{A}^{k-1}}{\partial A_{\nu\mu}} \mathbf{b}\mathbf{b}^T \mathbf{A}^{T,k-1}) + \mathbf{A}^{k-1}\mathbf{b}\mathbf{b}^T \frac{\partial \mathbf{A}^{T,k-1}}{\partial A_{\nu\mu}} \tag{19}$$

$$= \sum_{k=2}^{n} \left[\sum_{i=1}^{k-1} [\mathbf{A}^{i-1}\mathbf{E}_{\nu,\mu}\mathbf{A}^{k-i-1}]\mathbf{b}\mathbf{b}^T\mathbf{A}^{T,k-1} + \mathbf{A}^{k-1}\mathbf{b}\mathbf{b}^T \sum_{i=1}^{k-1} \mathbf{A}^{T,i-1}\mathbf{E}_{\nu,\mu}^T \mathbf{A}^{T,k-i-1} \right] . \tag{20}$$

Derivatives with respect to a vector $\mathbf{k}_r \in \mathcal{R}^r$ with $s = 1$ are

$$\frac{\partial \mathbf{L}_c \mathbf{L}_c^T}{\partial \mathbf{k}_r} \overset{(3)}{=} \sum_{k=1}^{n} \frac{\partial \mathbf{A}^{k-1}}{\partial \mathbf{k}_r} [\mathbf{I}_s \otimes (\mathbf{b}\mathbf{b}^T\mathbf{A}^{T,k-1})] + [\mathbf{I}_r \otimes (\mathbf{A}^{k-1}\mathbf{b}\mathbf{b}^T)] \frac{\partial \mathbf{A}^{T,k-1}}{\partial \mathbf{k}_r} \tag{21}$$

$$\overset{(129)}{=} \sum_{k=2}^{n} [\sum_{j=1}^{k-1} (\mathbf{I}_r \otimes \mathbf{A}^{j-1}) \frac{\partial \mathbf{A}}{\partial \mathbf{k}_r} \mathbf{A}^{k-j-1} [\mathbf{I}_s \otimes (\mathbf{b}\mathbf{b}^T\mathbf{A}^{T,k-1})]$$

$$+ [\mathbf{I}_r \otimes (\mathbf{A}^{k-1}\mathbf{b}\mathbf{b}^T)] \sum_{j=1}^{k-1} (\mathbf{I}_r \otimes \mathbf{A}^{T,j-1}) \frac{\partial \mathbf{A}^T}{\partial \mathbf{k}_r} \mathbf{A}^{T,k-j-1}, \tag{22}$$

and for the corresponding trace one has

$$\frac{\partial \mathrm{tr}[\mathbf{L}_c \mathbf{L}_c^T]}{\partial \mathbf{k}_r} = \mathrm{trh}[\frac{\partial (\mathbf{L}_c \mathbf{L}_c^T)}{\partial \mathbf{k}_r}] \ , \quad \text{see Eq.141} \ . \tag{23}$$

3 Flat Output Gradients

The flat output vector gradient is

$$\frac{\partial \mathbf{c}_f}{\partial A_{\nu\mu}} \overset{(79)}{=} \frac{\partial \mathbf{L}_c^{-T}\mathbf{e}_n}{\partial A_{\nu\mu}} \overset{(119)}{=} -\mathbf{L}_c^{-T} \frac{\partial \mathbf{L}_c^T}{\partial A_{\nu\mu}} \mathbf{L}_c^{-T}\mathbf{e}_n \tag{24}$$

$$\overset{(18)}{=} -\mathbf{L}_c^{-T} [\sum_{i=2}^{n} (\sum_{v=1}^{v=i-1} \mathbf{b}^T \mathbf{A}^{T,v-1} \mathbf{E}_{\nu,\mu} \mathbf{A}^{T,i-v-1}) \otimes \mathbf{e}_i] \mathbf{L}_c^{-T}\mathbf{e}_n \ . \tag{25}$$

The squared norm of the flat output turns out as

$$\mathbf{c}_f^T \mathbf{c}_f = \mathbf{e}_n^T \mathbf{L}_c^{-1} \mathbf{L}_c^{-T} \mathbf{e}_n = \mathrm{tr}[\mathbf{L}_c^{-T}\mathbf{e}_n\mathbf{e}_n^T\mathbf{L}_c^{-1}] = [\mathbf{L}_c^{-1}\mathbf{L}_c^{-T}]_{n,n} \tag{26}$$

$$\|\mathbf{c}_f\|_F^2 = \mathrm{tr}[\mathbf{c}_f\mathbf{c}_f^T] = \mathrm{tr}[\mathbf{c}_f^T\mathbf{c}_f] = \mathbf{e}_n^T \mathbf{L}_c^{-1} \mathbf{L}_c^{-T} \mathbf{e}_n \tag{27}$$

$$= [(\mathbf{L}_c^T\mathbf{L}_c)^{-1}]_{n,n} \overset{(2)}{=} \sum_{i=1}^{i=n} \sum_{k=1}^{k=n} \mathbf{E}_{i,k} \mathbf{b}^T \mathbf{A}^{T,i-1} \mathbf{A}^{k-1} \mathbf{b} \ . \tag{28}$$

Even though the following Eqs.(30), (31) and (34) yield zero for a state controller $\mathbf{k} = \mathbf{k}_r \in \mathcal{R}^n$ (due to Eq.(83)), the differential quotients are given for the sake of completeness and algebraic check. The sensitivity of the Frobenius norm of the flat output is

$$\begin{aligned}
\|\mathbf{c}_f\|_F^2 &= \mathrm{tr}[\mathbf{c}_f\mathbf{c}_f^T] = \mathrm{tr}[\mathbf{L}_c^{-T}\mathbf{e}_n\mathbf{e}_n^T\mathbf{L}_c^{-1}] && (29)\\
\frac{\partial\|\mathbf{c}_f\|_F^2}{\partial\mathbf{k}} &\overset{(141)}{=} \mathrm{trh}[\frac{\partial\mathbf{L}_c^{-T}}{\partial\mathbf{k}}\mathbf{e}_n\mathbf{e}_n^T\mathbf{L}_c^{-1} + [\mathbf{I}_n\otimes(\mathbf{L}_c^{-T}\mathbf{e}_n\mathbf{e}_n^T)]\frac{\partial\mathbf{L}_c^{-1}}{\partial\mathbf{k}}] && (30)\\
\frac{\partial\|\mathbf{c}_f\|_F^2}{\partial\mathbf{k}} &= -\mathrm{trh}\left[(\mathbf{I}_n\otimes\mathbf{L}_c^{-T})\{\frac{\partial\mathbf{L}_c^T}{\partial\mathbf{k}}\mathbf{L}_c^{-T}\mathbf{E}_{n,n} + [\mathbf{I}_n\otimes(\mathbf{E}_{n,n}\mathbf{L}_c^{-1})]\frac{\partial\mathbf{L}_c}{\partial\mathbf{k}}\}\mathbf{L}_c^{-1}\right]\ . && (31)
\end{aligned}$$

Alternatively,

$$\begin{aligned}
\|\mathbf{c}_f\|_F^2 &= \mathrm{tr}[\mathbf{c}_f^T\mathbf{c}_f] = \mathbf{e}_n^T\mathbf{L}_c^{-1}\mathbf{L}_c^{-T}\mathbf{e}_n && (32)\\
\frac{\partial\|\mathbf{c}_f\|_F^2}{\partial\mathbf{k}_r} &\overset{(112)}{=} (\mathbf{I}_n\otimes\mathbf{e}_n^T)\frac{\partial\mathbf{L}_c^{-1}\mathbf{L}_c^{-T}}{\partial\mathbf{k}}\mathbf{e}_n && (33)\\
&\overset{(140)}{=} -[\mathbf{I}_n\otimes(\mathbf{e}_n^T\mathbf{L}_c^{-1})]\left[\frac{\partial\mathbf{L}_c}{\partial\mathbf{k}}\mathbf{L}_c^{-1} + \mathbf{L}_c^{-T}\frac{\partial\mathbf{L}_c^T}{\partial\mathbf{k}}\right]\mathbf{L}_c^{-T}\mathbf{e}_n\ . && (34)
\end{aligned}$$

The derivative of the norm of the flat output coefficients with respect to an element of the coefficient matrix is

$$\begin{aligned}
g(\nu,\mu) \triangleq \frac{\partial\|\mathbf{c}_f\|_F^2}{\partial A_{\nu\mu}} &\overset{(31)}{=} -\mathrm{tr}[\mathbf{L}_c^{-T}\{\frac{\partial\mathbf{L}_c^T}{\partial A_{\nu\mu}}\mathbf{L}_c^{-T}\mathbf{E}_{n,n} + \mathbf{E}_{n,n}\mathbf{L}_c^{-1}\frac{\partial\mathbf{L}_c}{\partial A_{\nu\mu}}\}\mathbf{L}_c^{-1}] && (35)\\
&= -2\mathrm{tr}[\mathbf{L}_c^{-T}\mathbf{E}_{n,n}\mathbf{L}_c^{-1}\frac{\partial\mathbf{L}_c}{\partial A_{\nu\mu}}\mathbf{L}_c^{-1}] = -2[\mathbf{L}_c^{-1}\frac{\partial\mathbf{L}_c}{\partial A_{\nu\mu}}\mathbf{L}_c^{-1}\mathbf{L}_c^{-T}]_{nn}\ . && (36)
\end{aligned}$$

Finally,

$$\mathbf{G}(\nu,\mu) \triangleq g(\nu,\mu)\mathbf{E}_{\nu\mu} \overset{(36)}{=} \frac{\partial\|\mathbf{c}_f\|_F^2}{\partial A_{\nu\mu}}\mathbf{E}_{\nu\mu}. \qquad (37)$$

4 Modal Flatness Matrix

The modal matrix for flatness coordinates and its derivative are

$$\begin{aligned}
\mathbf{T}_C &\overset{(84)}{=} \sum_{i=1}^{n}\mathbf{e}_i\otimes(\mathbf{c}_f\mathbf{A}^{i-1}) \overset{(79)}{=} \sum_{i=1}^{n}\mathbf{e}_i\otimes\{\mathbf{e}_n^T\mathbf{L}_c^{-1}\mathbf{A}^{i-1}\} && (38)\\
&\overset{(1)}{=} \sum_{i=1}^{n}\mathbf{e}_i\otimes\{\mathbf{e}_n^T[\sum_{k=1}^{k=n}\mathbf{e}_k^T\otimes(\mathbf{A}^{k-1}\mathbf{b})]^{-1}\mathbf{A}^{k-1}\}\ , && (39)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial\mathbf{T}_C}{\partial A_{\nu\mu}} &\overset{(138)}{=} \sum_{i=1}^{n}(\frac{\partial}{\partial A_{\nu\mu}}\{\mathbf{e}_n^T\mathbf{L}_c^{-1}\mathbf{A}^{i-1}\})\otimes\mathbf{e}_i \qquad (\mathbf{U}_{n,1}=\mathbf{I}_n,\ \mathbf{U}_{1,n}=\mathbf{I}_n) && (40)\\
&\overset{(109)}{=} \sum_{i=1}^{n}\left\{\mathbf{e}_n^T\left(\frac{\partial}{\partial A_{\nu\mu}}\left[\sum_{k=1}^{n}\mathbf{e}_k^T\otimes(\mathbf{A}^{k-1}\mathbf{b})\right]^{-1}\right)\mathbf{A}^{i-1} + \mathbf{e}_n^T\mathbf{L}_c^{-1}\frac{\partial\mathbf{A}^{i-1}}{\partial A_{\nu\mu}}\right\}\otimes\mathbf{e}_i && (41)
\end{aligned}$$

$$\overset{(138),(119)}{=} \sum_{i=1}^{n} \left\{ \mathbf{e}_n^T(-\mathbf{L}_c^{-1})[\sum_{k=1}^{n} \frac{\partial \mathbf{A}^{k-1}\mathbf{b}}{\partial A_{\nu\mu}} \otimes \mathbf{e}_k^T]\mathbf{L}_c^{-1}\mathbf{A}^{i-1} + \mathbf{e}_n^T\mathbf{L}_c^{-1}\frac{\partial \mathbf{A}^{i-1}}{\partial A_{\nu\mu}} \right\} \otimes \mathbf{e}_i \quad (42)$$

$$= \sum_{i=1}^{n} \left\{ \mathbf{e}_n^T(-\mathbf{L}_c^{-1})[\sum_{k=1}^{n} (\frac{\partial \mathbf{A}^{k-1}}{\partial A_{\nu\mu}}\mathbf{b}) \otimes \mathbf{e}_k^T]\mathbf{L}_c^{-1}\mathbf{A}^{i-1} + \mathbf{e}_n^T\mathbf{L}_c^{-1}\frac{\partial \mathbf{A}^{i-1}}{\partial A_{\nu\mu}} \right\} \otimes \mathbf{e}_i \quad (43)$$

$$\overset{(119)}{=} \sum_{i=1}^{n} \Big\{ \mathbf{e}_n^T\mathbf{L}_c^{-1}[-\sum_{k=2}^{n}\sum_{v=1}^{k-1}(\mathbf{A}^{v-1}\mathbf{E}_{\nu,\mu}\mathbf{A}^{k-v-1}\mathbf{b}) \otimes \mathbf{e}_k^T]\mathbf{L}_c^{-1}\mathbf{A}^{i-1}$$
$$+ \left(\sum_{v=1}^{i-1}\mathbf{A}^{v-1}\mathbf{E}_{\nu,\mu}\mathbf{A}^{i-v-1}\right)_{exept\ i=1} \Big\} \otimes \mathbf{e}_i\ , \quad (44)$$

where $\mathbf{L}_c$ could have been inserted from Eq.(1).

5 Second-Order Sensitivity Matrices

From Eq.(15), by additional differentiation with respect to p, q

$$\frac{\partial^2 \mathbf{L}_c}{\partial A_{pq}\partial A_{\nu\mu}} \overset{(139)}{=} \sum_{k=2}^{k=n}\sum_{i=1}^{i=k-1}\left(\frac{\partial \mathbf{A}^{i-1}\mathbf{E}_{\nu,\mu}\mathbf{A}^{k-i-1}\mathbf{b}}{\partial A_{pq}}\right) \otimes \mathbf{e}_k^T \quad (45)$$

$$\overset{(109)}{=} \sum_{k=2}^{k=n}\sum_{i=1}^{i=k-1}\Big[\frac{\partial \mathbf{A}^{i-1}}{\partial A_{pq}}\mathbf{E}_{\nu,\mu}\mathbf{A}^{k-i-1}\mathbf{b} + \mathbf{A}^{i-1}\mathbf{E}_{\nu,\mu}\frac{\partial \mathbf{A}^{k-i-1}}{\partial A_{pq}}\mathbf{b}\Big] \otimes \mathbf{e}_k^T \quad (46)$$

$$\overset{(131)}{=} \sum_{k=2}^{k=n}\sum_{i=1}^{i=k-1}\Big[\left(\sum_{v=1}^{v=i-1}\mathbf{A}^{v-1}\mathbf{E}_{p,q}\mathbf{A}^{i-v-1}\right)_{i\geq 2}\mathbf{E}_{\nu,\mu}\mathbf{A}^{k-i-1}\mathbf{b}$$
$$+\mathbf{A}^{i-1}\mathbf{E}_{\nu,\mu}\left(\sum_{v=1}^{v=k-i-1}\mathbf{A}^{v-1}\mathbf{E}_{p,q}\mathbf{A}^{k-i-v-1}\right)_{k-i\geq 2}\Big] \otimes \mathbf{e}_k^T\ . \quad (47)$$

Suppose that some of of the entries of $\mathbf{A}$ can be altered by plant design activities. Referring to Eq.(36), the sensitivity of the squared norm of the flat output with respect to A_{pq} is

$$S_{pq} \triangleq \frac{\partial^2 \|\mathbf{c}_f\|_F^2}{\partial A_{pq}\partial A_{\nu\mu}} \overset{(36)}{=} \frac{\partial}{\partial A_{pq}}[-2\mathbf{e}_n^T\mathbf{L}_c^{-1}\frac{\partial \mathbf{L}_c}{\partial A_{\nu\mu}}\mathbf{L}_c^{-1}\mathbf{L}_c^{-T}\mathbf{e}_n] \quad (48)$$

$$\overset{(109)}{=} -2\mathbf{e}_n^T\Big(\frac{\partial \mathbf{L}_c^{-1}}{\partial A_{pq}}\frac{\partial \mathbf{L}_c}{\partial A_{\nu\mu}}\mathbf{L}_c^{-1}\mathbf{L}_c^{-T} + \mathbf{L}_c^{-1}\frac{\partial^2 \mathbf{L}_c}{\partial A_{pq}\partial A_{\nu\mu}}\mathbf{L}_c^{-1}\mathbf{L}_c^{-T}$$
$$+ \mathbf{L}_c^{-1}\frac{\partial \mathbf{L}_c}{\partial A_{\nu\mu}}\frac{\partial \mathbf{L}_c^{-1}}{\partial A_{pq}}\mathbf{L}_c^{-T} + \mathbf{L}_c^{-1}\frac{\partial \mathbf{L}_c}{\partial A_{\nu\mu}}\mathbf{L}_c^{-1}\frac{\partial \mathbf{L}_c^{-T}}{\partial A_{pq}}\Big)\mathbf{e}_n \quad (49)$$

$$\overset{(119)}{=} -2\mathbf{e}_n^T\mathbf{L}_c^{-1}\Big(-\frac{\partial \mathbf{L}_c}{\partial A_{pq}}\mathbf{L}_c^{-1}\frac{\partial \mathbf{L}_c}{\partial A_{\nu\mu}}\mathbf{L}_c^{-1} + \frac{\partial^2 \mathbf{L}_c}{\partial A_{pq}\partial A_{\nu\mu}}\mathbf{L}_c^{-1}$$
$$- \frac{\partial \mathbf{L}_c}{\partial A_{\nu\mu}}\mathbf{L}_c^{-1}\frac{\partial \mathbf{L}_c}{\partial A_{pq}}\mathbf{L}_c^{-1} - \frac{\partial \mathbf{L}_c}{\partial A_{\nu\mu}}\mathbf{L}_c^{-1}\mathbf{L}_c^{-T}\frac{\partial \mathbf{L}_c^T}{\partial A_{pq}}\Big)\mathbf{L}_c^{-T}\mathbf{e}_n\ . \quad (50)$$

For a preselected (ν, μ), the matrix $\mathbf{S}(p, q)$ provides the combined sensitivity

$$\mathbf{S}(p,q) \triangleq \frac{\partial}{\partial A_{pq}}\frac{\partial \|\mathbf{c}_f\|_F^2}{\partial A_{\nu\mu}}\Big|_{\nu\mu}\mathbf{E}_{pq} \quad (51)$$

in order to indicate which influence the (p,q) element of $\mathbf{A}$ has on $\frac{\partial\|\mathbf{c}_f\|_F^2}{\partial A_{\nu\mu}}$. This is a measure how some of the residual elements of $\mathbf{A}$ can change the sensitivity of the $\mathbf{c}_f$ norm with respect to the uncertainty. Lowering the sensitivity corresponds to a robustification of $\mathbf{c}_f$. Considering a single uncertainty at $A_{\nu\mu}$, it is desired to keep its influence on the flat output low, i.e., $\frac{\partial\|\mathbf{c}_f\|_F^2}{\partial A_{\nu\mu}}$ low but the corresponding $\mathbf{S}$ should be high. Influencing via $\mathbf{b}\mathbf{k}_r^T$ is impossible due to the orthogonal properties of $\mathbf{c}_f$.

6 Incremental Representation

For an increment $\Delta\mathbf{A}$, we get the following increment of $\Delta\mathbf{L}_c$

$$\Delta\mathbf{L}_c \overset{(1)}{=} \sum_{i=2}^{n} \mathbf{e}_i^T \otimes \left[(\mathbf{A}+\Delta\mathbf{A})^{i-1}\mathbf{b} - \mathbf{A}^{i-1}\mathbf{b}\right] \tag{52}$$

$$\overset{\cdot(123)}{=} \sum_{i=2}^{n} \mathbf{e}_i^T \otimes \sum_{k=1}^{k=i-1} \mathbf{A}^{i-k-1}\Delta\mathbf{A}\cdot\mathbf{A}^{k-1}\mathbf{b}\ . \tag{53}$$

Referring to Eq.(121), the increment of the flat output parameter vector reads as

$$\Delta\mathbf{c}_f = \left(\sum_{i=1}^{n} \mathbf{e}_i^T \otimes \left[(\mathbf{A}+\Delta\mathbf{A})^{i-1}\mathbf{b}\right]\right)^{-T} \mathbf{e}_n - \left(\sum_{i=1}^{n} \mathbf{e}_i^T \otimes \left[\mathbf{A}^{i-1}\mathbf{b}\right]\right)^{-T} \mathbf{e}_n \tag{54}$$

$$\doteq -\mathbf{L}_c^{-T}\Big(\sum_{i=1}^{n} \mathbf{e}_i \otimes \sum_{k=1}^{k=i-1} \mathbf{b}^T\mathbf{A}^{T,i-k-1}\Delta\mathbf{A}^T\mathbf{A}^{T,k-1}\Big)\mathbf{L}_c^{-T}\mathbf{e}_n\ , \tag{55}$$

$$\|\Delta\mathbf{c}_f\|_F^2 = \mathbf{e}_n^T\mathbf{L}_c^{-1}\Delta\mathbf{L}_c\cdot\mathbf{L}_c^{-1}\mathbf{L}_c^{-T}\Delta\mathbf{L}_c^T\cdot\mathbf{L}_c^{-T}\mathbf{e}_n\ . \tag{56}$$

7 Output Sensitivity. Feedforward Control

Interest is focussed on the output feedforward control of the system based on the differential quotient properties of the flat output. Since $\Delta A_{\nu\mu}$ is an uncertain quantity, one can only try to reduce the gradient of the final $y(T)$ versus $\Delta A_{\nu\mu}$ as one of many opportunities. Smaller gradient guarantees smaller changes in $\Delta y(T)$ irrespective of the sign and magnitude of $\Delta A_{\nu\mu}$.

Another proposal qualifying for robustification is to put interest on similar derivatives with respect to the parameters of the (ν,μ) element and the operating (p,q) element. For a given uncertainty position (ν,μ) in the matrix $\mathbf{A}$, find a position (p,q) the parameter of which is permissible for a change in the design phase and, in addition, the influence on the sensitivity is worth while mentioning. That is,

$$d_s \overset{\triangle}{=} \min_{\beta}\left[\|\frac{\partial\mathbf{c}_f}{\partial A_{\nu\mu}} - \beta\frac{\partial\mathbf{c}_f}{\partial A_{pq}}\Big|_{\mathbf{A}+\Delta\mathbf{A}_{pq}}\|_s\right] \quad\rightarrow\quad \min_{pq}\ . \tag{57}$$

The basis for this optimization are the dependencies in $\mathbf{c}_f(\mathbf{A})$. (Needless to say that the best oportunity was $(p,q) \equiv (\nu,\mu)$ but this is considered not realizable.)

The reduction of the norm of the flat output does not imply better feedforward control target results in general. It is only an indicator. A direct process for matching the targets optimally is as follows: Check via simulation which admissible change ΔA_{pq} provides a remarkable rejection of the uncertainty influences at position (ν,μ). Since some plant redesign is required, based on the optimal A_{pq}, only slight improvement can be expected.

$$\Delta_{\nu\mu}\{\texttt{lsim}(\mathbf{A}+\mathbf{E}_{pq}\Delta A_{pq}+\mathbf{E}_{\nu\mu}\Delta A_{\nu\mu},\ \mathbf{b},\ u_{des},\ T\} \rightarrow \min_{pq}\ . \tag{58}$$

The actuating variable u_{des} is the nominal one.

A direct functional dependence of the output on the flat coordinates leads to another approach. Referring to Eq.(91),

$$\frac{\partial y(T)}{\partial A_{\nu\mu}} = -\mathbf{c}^T\mathbf{T}_C^{-1}\frac{\partial \mathbf{T}_C}{\partial A_{\nu\mu}}\mathbf{T}_C^{-1}\mathbf{e}_1 z_{end}\ . \tag{59}$$

We utilize Eq.(84) and Eq.(44) for $\mathbf{T}_C$ and its sensitivity, respectively.

If the sensitivity of $\mathbf{a}_L$ in Eq.(94) is required, one has

$$\frac{\partial \mathbf{a}_L}{\partial A_{\nu\mu}} = \frac{\partial}{\partial A_{\nu\mu}}\mathbf{c}_f^T\mathbf{A}^n\mathbf{T}_C^{-1}. \tag{60}$$

By using the product rule and Eq.(25), Eq.(132) and Eq.(44) the result can easily be achieved.

8 Input Vector Related Sensitivity

Referring to flat *input* vector considerations, which are oriented at some flat input vector $\mathbf{b}_f$ with data related to the actuating device *Zeitz, M., 2009*, we suppose that incremental changes in $\mathbf{b}$ are feasible. The target is to change some b_k such that plant parameter perturbations are reduced on performance items $(\mathbf{c}_f,\ y)$, e.g.,

$$\left.\frac{\partial y(T)}{\partial A_{\nu\mu}}\right|_{\mathbf{b}} - \left.\frac{\partial y(T)}{\partial A_{\nu\mu}}\right|_{\mathbf{b}+\Delta b_k\mathbf{e}_k} \quad\rightarrow\quad \min_k \tag{61}$$

$$\frac{\partial^2 y(T)}{\partial b_k \partial A_{\nu\mu}} \quad\rightarrow\quad \min_k\ , \tag{62}$$

where all the complementary consequences are included.

The derivation of the flat output parameters versus b_k leads to

$$\frac{\partial \mathbf{c}_f}{\partial b_k} \quad=\quad \frac{\partial \mathbf{L}_c^{-T}\mathbf{e}_n}{\partial b_k} = \frac{\partial(\sum_{i=1}^n \mathbf{e}_i^T \otimes \mathbf{A}^{i-1}\mathbf{b})^{-T}\mathbf{e}_n}{\partial b_k} \tag{63}$$

$$\overset{(109)}{=} \frac{\partial(\sum_{i=1}^{n} \mathbf{e}_i^T \otimes \mathbf{A}^{i-1}\mathbf{b})^{-T}}{\partial b_k} \mathbf{e}_n \overset{(119),(138)}{=} -\mathbf{L}_c^{-T} \left[\sum_{i=1}^{n} \frac{\partial(\mathbf{A}^{i-1}\mathbf{b})^T}{\partial b_k} \otimes \mathbf{e}_i \right] \mathbf{L}_c^{-T} \mathbf{e}_n (64)$$

$$\overset{(109)}{=} -\mathbf{L}_c^{-T} [\sum_{i=1}^{n} (\mathbf{e}_k^T \mathbf{A}^{T,i-1}) \otimes \mathbf{e}_i] \mathbf{L}_c^{-T} \mathbf{e}_n \ . \tag{65}$$

9 Example

Consider the third-order system given by Eq.(98) in the Appendix Example.

Flat output norm sensitivity: The matrix $\mathbf{G}$ indicates the sensitivity of the flat output versus $(\nu,\ \mu)$. Having selected, e.g., $\nu = 2,\ \mu = 3$, for the location of the uncertainty, one has

$$\mathbf{G} = \begin{pmatrix} 140.6968 & -67.7118 & 108.9629 \\ -10.4492 & 96.5397 & -28.9106 \\ -542.0639 & -541.6339 & -237.2366 \end{pmatrix}, \quad \mathbf{S} = \begin{pmatrix} -44.5114 & 167.6481 & -53.3699 \\ -1.4906 & -82.6922 & 12.9054 \\ 213.5512 & -29.9149 & 127.2036 \end{pmatrix}. \tag{66}$$

For the preselected $\nu = 2,\ \mu = 3$, the influence for $p = 2,\ q = 1$ is low, for $p = 3,\ q = 1$ remarkable, see Figs. 1 and 2, respectively. The incremental effect Δc_f of the uncertainty for $\Delta A_{21} = 0.4$ and $\Delta A_{31} = 0.4$ turns out as

$$\begin{pmatrix} 0.0666 \\ -0.0115 \\ -0.1989 \end{pmatrix}, \quad \begin{pmatrix} 0.0748 \\ -0.0283 \\ -0.0890 \end{pmatrix}, \quad \text{where } \mathbf{c}_f = \begin{pmatrix} -4.4791 \\ 1.5324 \\ 6.7358 \end{pmatrix}, \tag{67}$$

corresponding to the norm difference 0.2101 0.1197 . Due to the strong influence of ΔA_{31} the influence of the uncertainty ΔA_{23} in only one half.

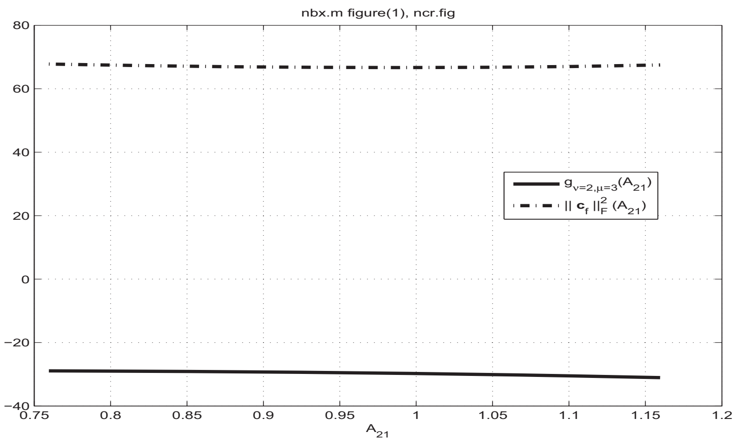


Figure 1: Low influence on the norm of the flat output, $p = 2,\ q = 1$

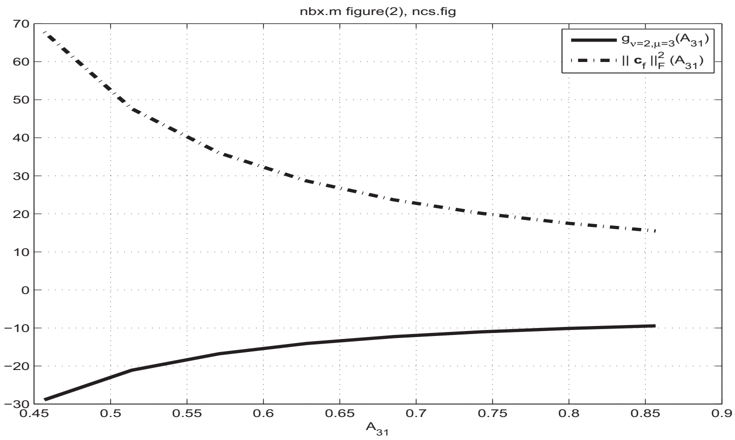


Figure 2: Influence for $p = 3,\ q = 1$

Two uncertainty walks: In addition, an uncertainty walk around the nominal parameters is performed. These results refer to the minimal norm of $\mathbf{c}_f$. (This should not be mixed up with the result of d_s in Eq.(70).)

The uncertainty $(\gamma,\ \delta)$ is given by Fig. 3 and is distributed versus $\mathbf{A}$ corresponding to

$$\Delta\mathbf{A} = [0\ 0\ 0;\ 0\ 0\ 0;\ 0\ \gamma(i)\ \delta(i)] \quad \text{(in Case 1)} \tag{68}$$

$$\Delta\mathbf{A} = [0\ 0\ 0;\ 0\ 0\ \delta(i);\ 0\ \gamma(i)\ 0] \quad \text{(in Case 2).} \tag{69}$$

The influence on the flat output is depicted in Figs. 4 and 5 in Case 1 and 2, correspondingly. The reduction in the flat output is achieved presupposing that one can afford the change in the parameter A_{31} in the plant under control. The step response difference referring to the change in A_{31} is given in Fig. 6.

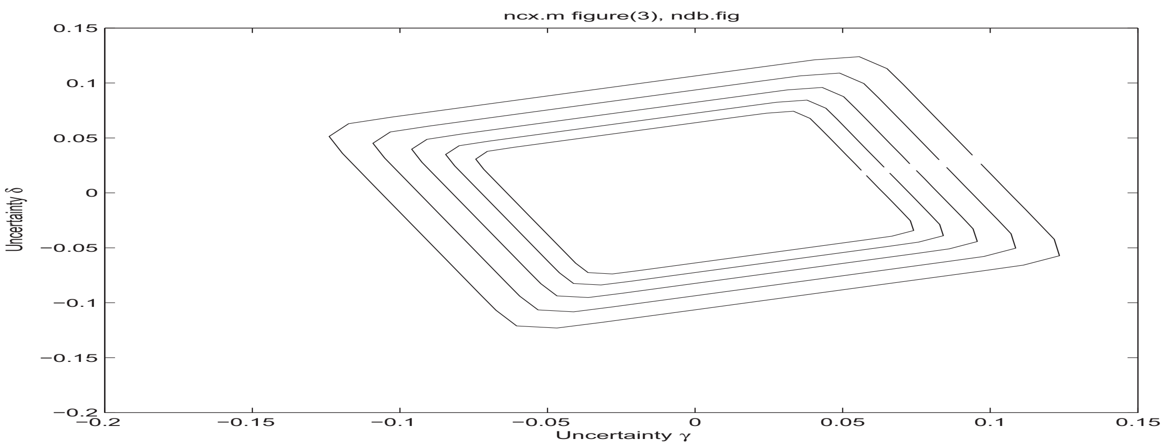


Figure 3: Uncertainty walk corresponding to Case 1 in Eq.(68)

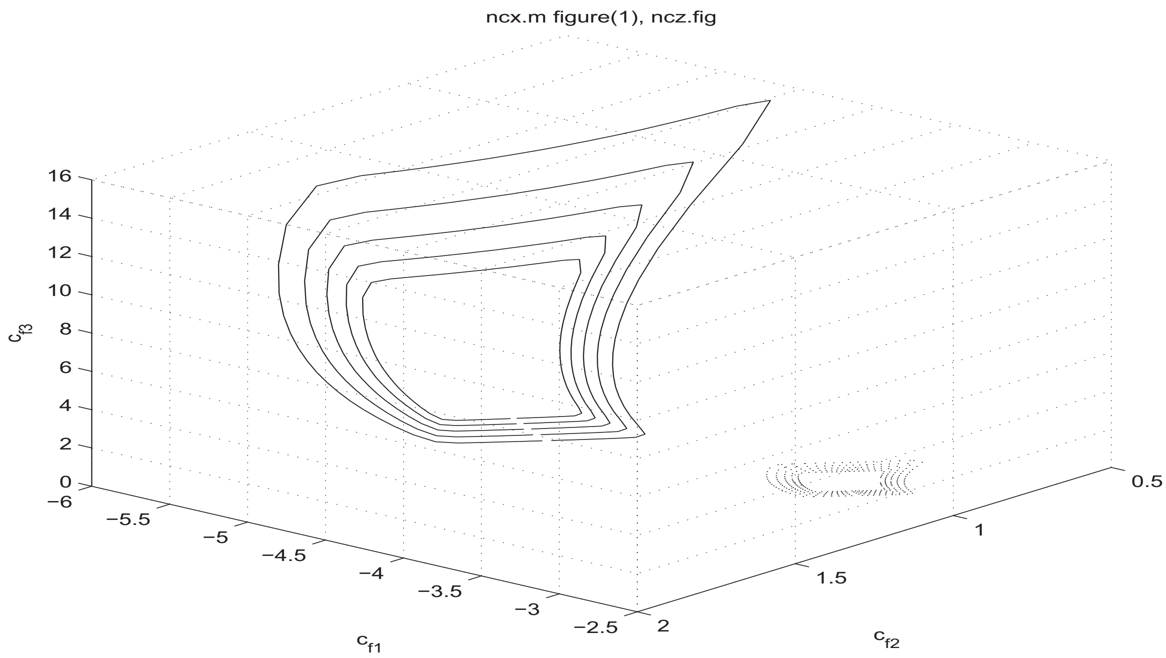


Figure 4: Flat output paramters in Case 1

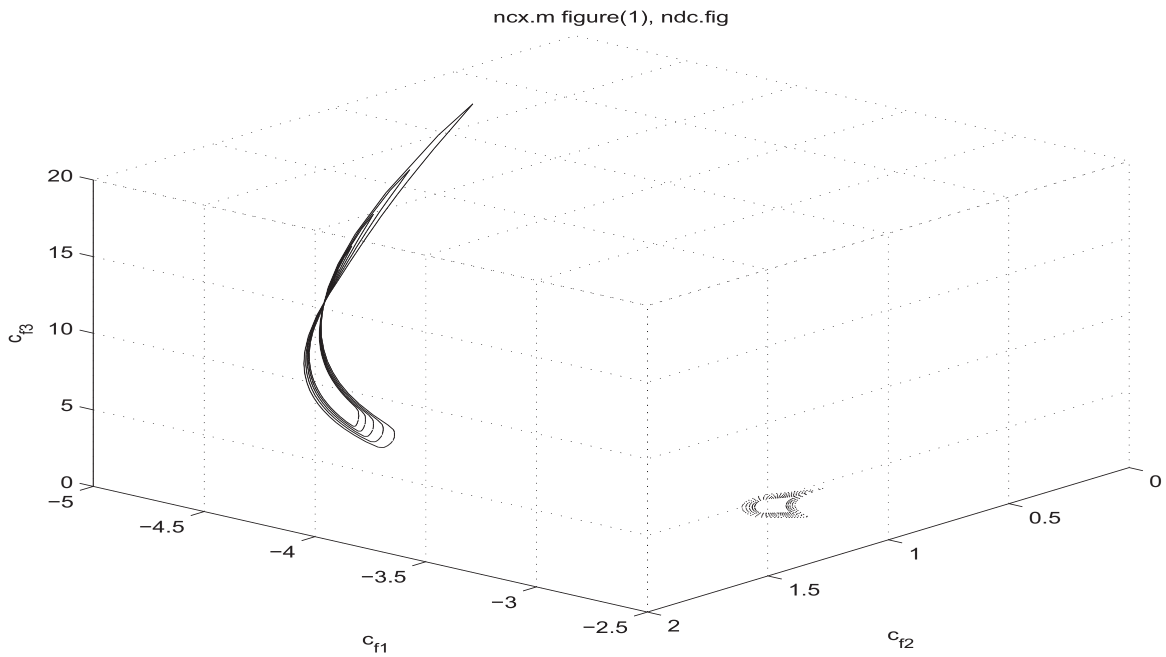


Figure 5: Flat output paramters in Case 2

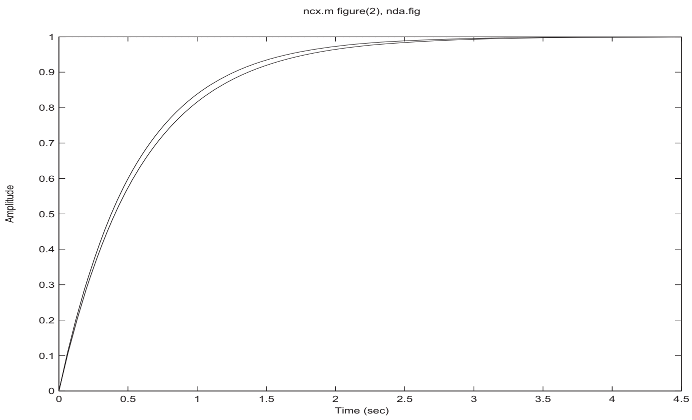


Figure 6: Step response increment referring to the increment in A_{31}

Flat output sensitivity I: Based on Eq.(58), the difference in the output feedforward control $y(T)$ in the original and in the improved version is given in Fig. 7; for an assumed uncertainty walk (see subplot) with the uncertainties at position (γ, δ) (2,3) and (3,3) in Case 1. The assumption for the improved version is the admissible 30% change in the parameters A_{22} and an entire redesign of the system with the change in A_{22} from -2.6642 to -3.4634. If different positions for the uncertainty and reduction parameters (p, q) were selected, the result might be poor.

Note the following facts: In spite of large changes in Fig. 2 referring to ΔA_{31} , the output $y(T)$ is only influenced slightly which demonstates the limited statement of $\mathbf{c}_f$ and its derivative. Fig. 7 shows the remarkable influence of ΔA_{22}. Alternatives of the influence on $y(T)$ at the destination time T are given in the following paragraphs.

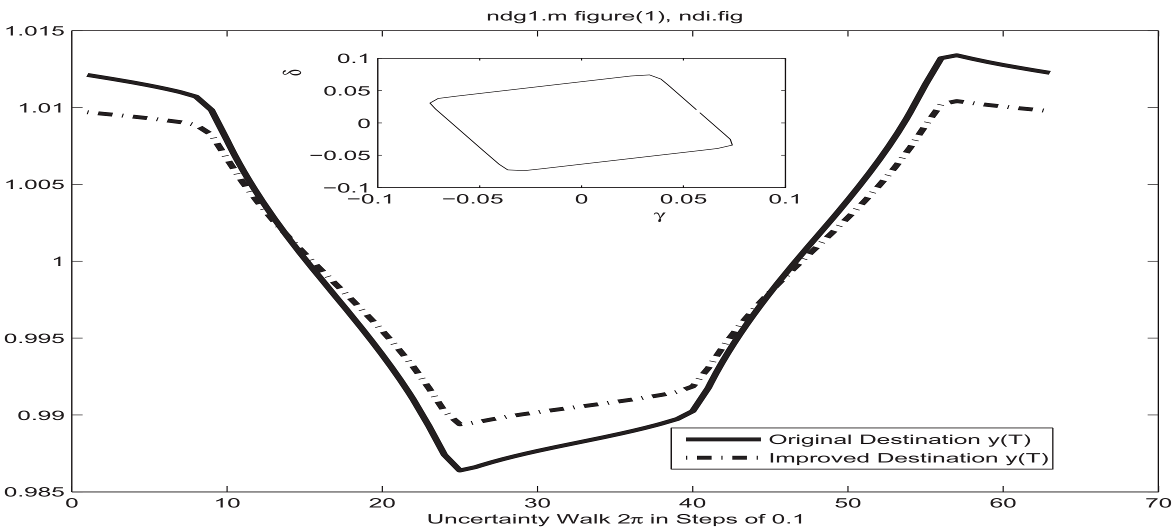


Figure 7: Feedforward control result comparison for a given uncertainty walk

Flat output sensitivity II: The difference amount of Eq.(57) for each position (p, q) is given in what follows, where $\mathbf{D}_s$ is the matrix of d_s , i.e., the tabular of differences referring to Eq.(57) when an additive scalar ΔA_{pq} is superposed to $\mathbf{A}$

$$\mathbf{D}_s \triangleq \text{matrix}[d_s] = \begin{pmatrix} 10.2540 & 2.7664 & 5.9850 \\ 1.1139 & 4.8539 & 0 \\ 1.6002 & 1.9836 & 2.8303 \end{pmatrix} . \tag{70}$$

The sensitivity $\frac{\partial \mathbf{c}_f^T}{\partial A_{\nu\mu}}$ reduces from $(0.5956 \quad -0.1092 \quad -1.7251)$ to the final quantity $(0.5752 \quad -0.1060 \quad -1.6610)$ and the corresponding norm from 1.8283 to 1.7610. The respective original and improved $\mathbf{c}_f$ and $\mathbf{A}$ is

$$\begin{pmatrix} -4.4791 \\ 1.5324 \\ 6.7358 \end{pmatrix} , \quad \begin{pmatrix} -4.2066 \\ 1.3304 \\ 7.2796 \end{pmatrix} \tag{71}$$

$$\begin{pmatrix} -2.8219 & 0.5219 & 0.2089 \\ \mathbf{0.3596} & -2.6642 & 0.9052 \\ 0.0567 & 0.1757 & -2.3246 \end{pmatrix} , \quad \begin{pmatrix} -2.8219 & 0.5219 & 0.2089 \\ \mathbf{0.0656} & -2.6642 & 0.9052 \\ 0.0567 & 0.1757 & -2.3246 \end{pmatrix} . \tag{72}$$

That is, the difference of $\frac{\partial \mathbf{c}_f}{\partial A_{ij}}\Big|_{(\nu,\mu)}$ and $\frac{\partial \mathbf{c}_f}{\partial A_{ij}}\Big|_{(p,q)}$ is considered in its minimal norm. The optimal position is (2,1) in Eq.(70). (Only 0.6 of ΔA_{ij} is used because of limits in changing the parameter A_{pq}.)

Selected output sensitivity: From Eq.(59) we get a reduction of the output sensitivity $\frac{\partial y}{\partial A_{\nu\mu}}$ from 0.0172 to 0.0134, where the optimal position (p, q) is (3,1) and $\Delta \mathbf{A}_{31} = 0.1$. The original and the improved $\mathbf{c}_f$ is

$$\begin{pmatrix} -4.4791 \\ 1.5324 \\ 6.7358 \end{pmatrix} , \quad \begin{pmatrix} -3.7195 \\ 1.3975 \\ 4.4972 \end{pmatrix} , \tag{73}$$

respectively, and the corresponding coefficient matrices $\mathbf{A}$

$$\begin{pmatrix} -2.8219 & 0.5219 & 0.2089 \\ 0.3596 & -2.6642 & 0.9052 \\ \mathbf{0.0567} & 0.1757 & -2.3246 \end{pmatrix} , \quad \begin{pmatrix} -2.8219 & 0.5219 & 0.2089 \\ 0.3596 & -2.6642 & 0.9052 \\ \mathbf{0.1567} & 0.1757 & -2.3246 \end{pmatrix} . \tag{74}$$

This result of the optimal position (3,1) corresponds to the result referring to Fig. 2.

Change versus input vector b: In Eq.(44) a second derivative with respect to $\mathbf{b}$ can be augmented which presents the results if some change in $\mathbf{b}$ is used for the rejection of the uncertainties. Differentiation with respect to all elements b_k simply replaces $\mathbf{b}$ by the appropriate $\mathbf{e}_k$. The result of an uncertainty walk with an entire redesign for a 20% change in the elements b_k is given in Fig. 8.

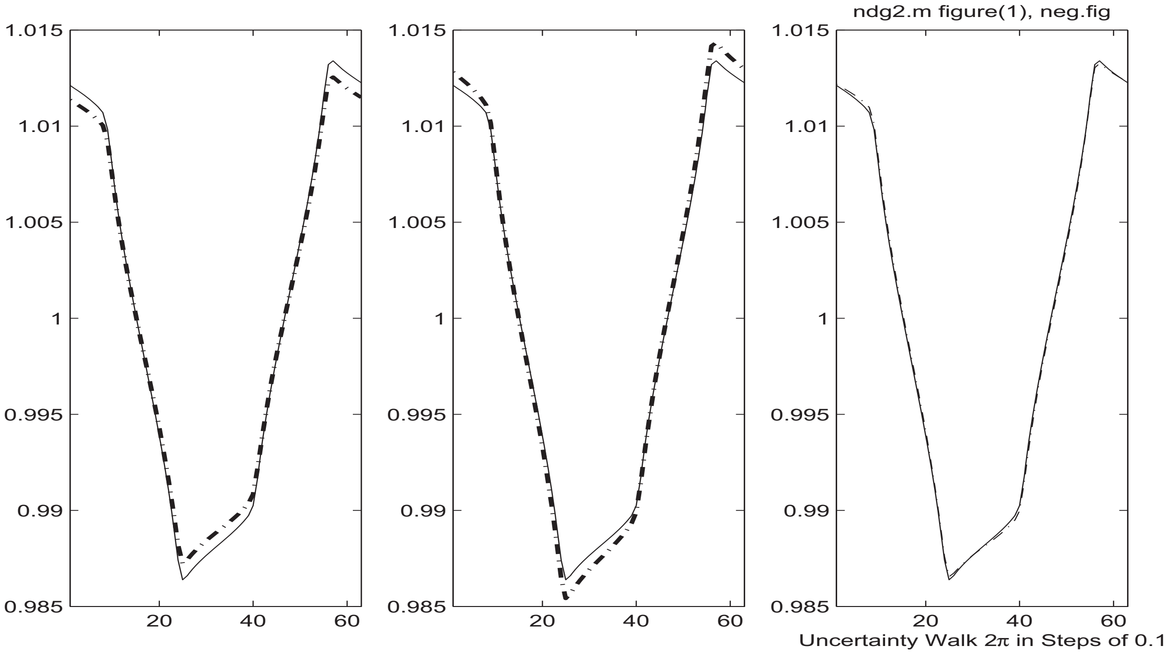


Figure 8: Output sensitivity for incremental change in the three components of $\mathbf{b}$

10 Conclusion

Several sensitivity functions were derived which are related to the topic of traditional flatness in linear single-input single-output systems. Mainly, they are differential quotients of flatness-related functions with respect to coefficient matrix elements. Among the variety of interesting problems, derivations were focussed on $\mathbf{c}_f$, $\mathbf{T}_C$ and $y(T)$. Since the controllability matrix itself contains Kronecker products, differential quotients with respect to scalars were preferred in order to avoid additional Kronecker products as far as possible.

To reduce the influence of an unknown uncertainty, the reduction of the sensitivity versus the perturbed parameter is utilized.

Based on the individual matrix functions comprising the coefficient matrix or input matrix, the influence of some selected matrix elements, which are admissible for some change in the design phase, on perturbed matrix elements were investigated. Reducing the local sensitivity, the system becomes robust to some extent. Since matrix element changes are restricted in many cases, the result of sensitivity decrease and robustness increase is limited. Nevertheless, using a combined matrix for determining the elements of best influence, the method becomes quite efficient.

The apparatus of formulas can be reused for discrete-time systems because the mathematical description is practically the same. Even for the problem of flat input, the presented formulas can be reused when $\mathbf{A}$ and $\mathbf{b}$ are replaced by $\mathbf{A}^T$ and $\mathbf{c}$.

References

Brewer, J.W., 1978, Kronecker products and matrix calculus in system theory, *IEEE-Trans.* **CAS-25**, pp. 772-781

Brewer, J.W., 1978a, Matrix calculus and the sensitivity analysis of linear dynamic systems, *IEEE-Trans.* **AC-23**, pp. 748-751

Franklin, G.F., Powell, J.D., and Emami-Naeini, A., 2002, Feedback Control of Dynamic Systems (Prentice Hall, Upper Saddle River, New Jersey)

Rothfuß, R., Rudolph, J., and Zeitz, M., 1997, Flachheit: Ein neuer Zugang zur Steuerung und Regelung nichtlinearer Systeme, *Automatisierungstechnik* **45**, pp. 517-525

Sira-Ramirez, H., and Agrawal, S.K., 2004, Differentially flat systems, New York, Basel: M. Dekker

Vetter, W.J., 1970, Derivative operations on matrices, *IEEE-Trans.* **AC-15**, pp.241-244 and **AC-16**, p.113

Weinmann, A., 2008, Performance gradients in automatic control analysis, a catalogue of correspondences. *Int. J. Automation Austria* **16**, pp. 1-59. See http://www.acin.tuwien.ac.at/Publikationen/IJAA

Zeitz, M., 2009, Flachheit – Eine nützliche Methodik auch für lineare SISO-Systeme, *16. Steirisches Seminar über Regelungstechnik und Prozessautomatisierung*, pp. 3-15

Appendix

A Basics in Flatness

The step response of a dynamic system of nth order without zeros is flat because it is continual up to the $(n-1)$ derivative. The input can be regained from the output by multiple differentiations, only. The input is differentially parameterizable from the output. Based on this

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t), \qquad \mathbf{x} \in \mathcal{R}^n \tag{75}$$

$$y(t) = \mathbf{C}\mathbf{x}(t) = \mathbf{c}^T\mathbf{x} \ , \tag{76}$$

where $u(t)$ is the step function in the origin. Then, one has $u(0^+) = 1, \quad \dot{u}(0^+) = 0, \quad \ddot{u}(0^+) = 0$ etc. Additionally, for $t = 0^+$ continually $\dot{\mathbf{x}} = \mathbf{b}$, $\ddot{\mathbf{x}} = \mathbf{A}\dot{\mathbf{x}} + \mathbf{b}\dot{u} = \mathbf{A}\mathbf{b}$ through $\mathbf{x}^{(n)} = \mathbf{A}^{n-1}\mathbf{b}$.

The flat output $z \triangleq y_f$ is defined such that — under some specific $\mathbf{c}_f$ — the result $z = \mathbf{c}_f^T\mathbf{x}$ has the relative degree n; thus, one has $\mathbf{c}_f\dot{\mathbf{x}} \triangleq 0, \quad \mathbf{c}_f\ddot{\mathbf{x}} \triangleq 0$ etc. up to $\mathbf{c}_f\mathbf{x}^{(n-1)} \triangleq 0$. Finally, $\mathbf{c}_f\mathbf{x}^{(n)} \triangleq 1$

(*Rothfuß, R., et al., 1997; Sira-Ramirez, H., and Agrawal, S.K., 2004; Zeitz, M., 2009*). Concatenating these conditions,

$$\begin{aligned} \mathbf{c}_f^T(\mathbf{b} \vdots \mathbf{A}\mathbf{b} \vdots \mathbf{A}^2\mathbf{b} \vdots \dots \vdots \mathbf{A}^{n-1}\mathbf{b}) &= (0 \;\; 0 \;\; 0 \;\; 1) && (77)\\ \mathbf{c}_f^T\mathbf{L}_c &= \mathbf{e}_n^T && (78)\\ \mathbf{c}_f &= (\mathbf{e}_n^T\mathbf{L}_c^{-1})^T = \mathbf{L}_c^{-1,T}\mathbf{e}_n \ . && (79) \end{aligned}$$

In doing so, the controllability matrix $\mathbf{L}_c$ (subscript lower $_c$) and the unit vector $\mathbf{e}_n$ surfaced. The unit vector $\mathbf{e}_n$ selects $\mathbf{c}_f$ as the last column of $\mathbf{L}_c^{-T}$, i.e., the last row of $\mathbf{L}_c^{-1}$. Hence,

$$(\mathbf{b} \vdots \mathbf{A}\mathbf{b} \vdots \mathbf{A}^2\mathbf{b} \vdots \dots \vdots \mathbf{A}^{n-1}\mathbf{b})^{-1} \triangleq \begin{pmatrix} \mathbf{h}_1^T \\ \mathbf{h}_2^T \\ \vdots \\ \mathbf{c}_f^T \end{pmatrix} \tag{80}$$

$$\mathbf{I}_n = \begin{pmatrix} \mathbf{h}_1^T \\ \mathbf{h}_2^T \\ \vdots \\ \mathbf{c}_f^T \end{pmatrix} (\mathbf{b} \vdots \mathbf{A}\mathbf{b} \vdots \mathbf{A}^2\mathbf{b} \vdots \dots \vdots \mathbf{A}^{n-1}\mathbf{b}) \tag{81}$$

or

$$\mathbf{c}_f^T\mathbf{b} = 0, \quad \mathbf{c}_f^T\mathbf{A}\mathbf{b} = 0, \quad \mathbf{c}_f^T\mathbf{A}^2\mathbf{b} = 0, \quad \dots \mathbf{c}_f^T\mathbf{A}^{n-1}\mathbf{b} = 1, \tag{82}$$

i.e.,

$$\mathbf{c}_f^T \in \{\mathcal{N}(\mathbf{b})\&\mathcal{N}(\mathbf{A}\mathbf{b})\&\mathcal{N}(\mathbf{A}^2\mathbf{b}) \quad \dots \mathcal{N}(\mathbf{A}^{n-2}\mathbf{b}) \ . \tag{83}$$

This corresponds with the requirement that $\mathbf{c}_f$ is orthogonal to each column of the controllability matrix. For a controllable system the controllability matrix has full rank.

Defining a flatness vector $\mathbf{z}$

$$\mathbf{z} \triangleq \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{pmatrix} = \begin{pmatrix} z \\ \dot{z} \\ \vdots \\ z^{(n-1)} \end{pmatrix} = \begin{pmatrix} \mathbf{c}_f^T\mathbf{x} \\ \mathbf{c}_f^T\mathbf{A}\mathbf{x} \\ \vdots \\ \mathbf{c}_f^T\mathbf{A}^{n-1}\mathbf{x} \end{pmatrix} = \begin{pmatrix} \mathbf{c}_f^T \\ \mathbf{c}_f^T\mathbf{A} \\ \vdots \\ \mathbf{c}_f^T\mathbf{A}^{n-1} \end{pmatrix} \mathbf{x} \triangleq \mathbf{T}_C\mathbf{x} \tag{84}$$

$$\dot{\mathbf{z}} = \begin{pmatrix} \mathbf{0}_{n-1,1} & \mathbf{I}_{n-1} \\ & \mathbf{a}_L^T \end{pmatrix} \mathbf{z} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} u \ . \tag{85}$$

The last row in the previous expression is

$$z^{(n)} = \mathbf{a}_L^T\mathbf{z} + u = \mathbf{c}_f^T\mathbf{A}^n\mathbf{x} + u = \mathbf{c}_f^T\mathbf{A}^n\mathbf{T}_C^{-1}\mathbf{z} + u \ . \tag{86}$$

Based on the definition of $\mathbf{z}$ in Eq.(84), the modal system results in canonical form (companion form, regulator form, with subscript capital $_C$, as a Frobenius matrix) when using flatness coordinates

$$\mathbf{A}_C = \mathbf{T}_C\mathbf{A} \ \mathbf{T}_C^{-1} \tag{87}$$

$$\begin{aligned}
\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{b}\,u && (88)\\
\mathbf{T}_C^{-1}\dot{\mathbf{z}} &= \mathbf{A}\mathbf{T}_C^{-1}\mathbf{z} + \mathbf{b}\,u && (89)\\
\dot{\mathbf{z}} &= \mathbf{A}_C\mathbf{z} + \mathbf{T}_C\mathbf{b}u = \mathbf{A}_C\mathbf{z} + \mathbf{b}_C u && (90)\\
y &= \mathbf{c}^T\mathbf{x} = \mathbf{c}^T\mathbf{T}_C^{-1}\mathbf{z} = \mathbf{c}_C^T\mathbf{z}\ . && (91)
\end{aligned}$$

If we had started the derivation with the assumption of the flat output $z\ (= z_1) = \mathbf{c}_f^T\mathbf{x}$, which is the first row of Eq.(84), and had differentiated continually and inserted and reinserted $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u$, the following system of equations would have been obtained

$$\mathbf{z} = \begin{pmatrix} z \\ \dot{z} \\ \vdots \\ z^{(n-1)} \end{pmatrix} = \underbrace{\begin{pmatrix} \mathbf{c}_f^T \\ \mathbf{c}_f^T\mathbf{A} \\ \vdots \\ \mathbf{c}_f^T\mathbf{A}^{n-1} \end{pmatrix}}_{\mathbf{L}_{bo}} \mathbf{x} + \underbrace{\begin{pmatrix} 0 & 0 & \dots & 0 \\ \mathbf{c}_f^T\mathbf{b} & 0 & \dots & 0 \\ \mathbf{c}_f^T\mathbf{A}\mathbf{b} & \mathbf{c}_f^T\mathbf{b} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_f^T\mathbf{A}^{n-2}\mathbf{b} & \mathbf{c}_f^T\mathbf{A}^{n-3}\mathbf{b} & \dots & \mathbf{c}_f^T\mathbf{b} \end{pmatrix}}_{\mathbf{H}} \begin{pmatrix} u \\ \dot{u} \\ \ddot{u} \\ \vdots \\ u^{(n-2)} \end{pmatrix} . \qquad (92)$$

In order to retrieve $\mathbf{x}$ in terms of $\mathbf{z}$ and to be independent of u and all its derivatives, the matrix $\mathbf{H}$ must vanish. This corresponds to the orthogonality condition mentioned previously. In addition, the observability matrix $\mathbf{L}_{bo} \equiv \mathbf{T}_C$ must be invertible, i.e., the flat output must be an observable output.

Recalculating the original output $y(t)$, the input $u(t)$ and the original state vector $\mathbf{x}$ from the flatness coordinates $\mathbf{z}$ in respective order,

$$\begin{aligned}
y &= \mathbf{c}^T\mathbf{T}_C^{-1}\mathbf{z} && (93)\\
u &\overset{(86)}{=} -\mathbf{a}_L^T\mathbf{z} + z^{(n)} = -\mathbf{c}_f^T\mathbf{A}^n\mathbf{T}_C^{-1}\mathbf{z} + z^{(n)} && (94)\\
\mathbf{x} &= \mathbf{T}_C^{-1}\mathbf{z}\ . && (95)
\end{aligned}$$

In application-oriented control engineering and (open-loop) feedforward control, main interest is focussed on Eq.(94). A transient from y_{ini} to y_{fin} can be easily transformed to a transient in $\mathbf{x}$ and $\mathbf{z}$. The motion runs from $z_{ini} = z_{1ini}$ to $z_{1fin} = z_{end}$. For example, from y_{ini}, $\mathbf{z}_{ini}$ to $y_{fin} = 1$, $\mathbf{z}_{fin} = [z_{end}\ \ 0\ \ 0]^T = z_{end}\mathbf{e}_1$. Hence,

$$(\mathbf{T}_C^{-T}\mathbf{c})^T\mathbf{z}_{fin} = 1 \quad \rightsquigarrow z_{end} = (\mathbf{c}^T\mathbf{T}_C^{-1}\mathbf{e}_1)^{-1} = 1/(\mathbf{c}_C^T\mathbf{e}_1). \qquad (96)$$

(In general, $\mathbf{x}_{fin} = \mathbf{T}_C^{-1}[z_{end}\ \ 0\ \ 0]^T$ is not proportional to $\mathbf{e}_1$.) Then, a smooth $z_{des}(t)$ is set, which is differentiable with respect to t, i.e. for $n = 3$,

$$z_{des}(t) = p_1 t^4/T^4 + \ldots + p_4 t^7/T^7. \qquad (97)$$

Referring to Eq.(94), a smooth $u_{des}(t)$ results.

Appendix Example: Consider the third-order dynamic system with left-hand side eigenvalues $\lambda_i[\mathbf{A}] = (-1.9235\ \ -3.2197\ \ -2.6674)$

$$\mathbf{A} = \begin{pmatrix} -2.8219 & 0.5219 & 0.2089 \\ 0.3596 & -2.6642 & 0.9052 \\ 0.0567 & 0.1757 & -2.3246 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 0.4685 \\ 0.9121 \\ 0.1040 \end{pmatrix}, \quad \mathbf{c} = \begin{pmatrix} 1 \\ 0.5 \\ 0.2 \end{pmatrix}. \qquad (98)$$

Then,

$$\mathbf{L}_c = \begin{pmatrix} 0.4685 & -0.8242 & 1.1832 \\ 0.9121 & -2.1674 & 5.4282 \\ 0.1040 & -0.0550 & -0.2997 \end{pmatrix}, \quad \mathbf{T}_C = \begin{pmatrix} -4.4791 & 1.5324 & 6.7358 \\ 13.5725 & -5.2367 & -15.2070 \\ -41.0458 & 18.3634 & 33.4462 \end{pmatrix}, \qquad (99)$$

$z_{end} = 0.1303$. Suppose $T \stackrel{\triangle}{=} 4$, then further results are $\mathbf{a}_L = (-16.5197 \quad -19.9123 \quad -7.8106)^T$, $\mathbf{c}_f = (-4.4791 \;\; 1.5324 \;\; 6.7358)^T$, $\mathbf{c}_C = (7.6731 \;\; 5.4648 \;\; 0.9453)^T$, $\mathbf{p} = (p_1 \;\; p_2 \;\; \ldots p_4)^T = \mathbf{M}^{-1}(1 \;\; 0 \;\; 0 \;\; 0)^T$

```
M=[1 1 1 1;  h(1,:);  h(2,:);   h(3,:)]
where    h(1,:)= [ 4    5    6     7]/T
         h(2,:)= [12   20   30    42]/T^2
         h(3,:)= [24   60  120   210]/T^3

    1.0000    1.0000    1.0000    1.0000
M = 1.0000    1.2500    1.5000    1.7500
    0.7500    1.2500    1.8750    2.6250
    0.3750    0.9375    1.8750    3.2813
```

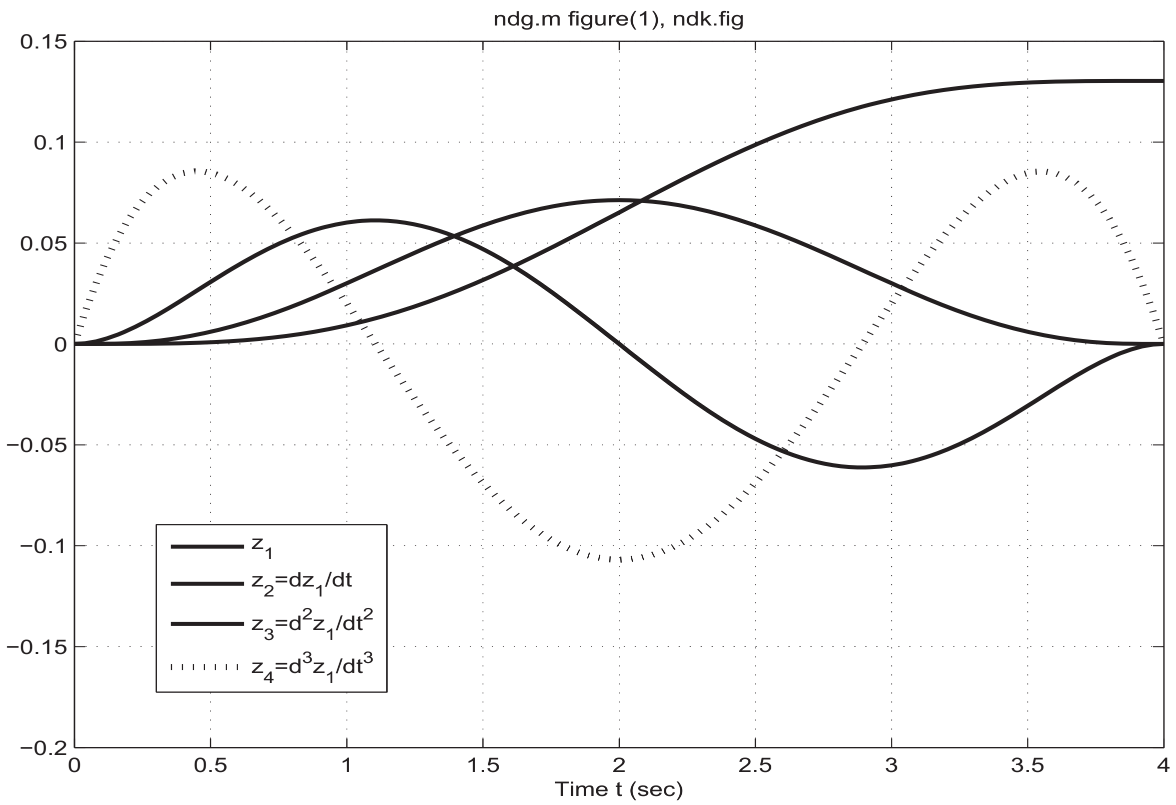


Figure 9: Desired flat coordinates

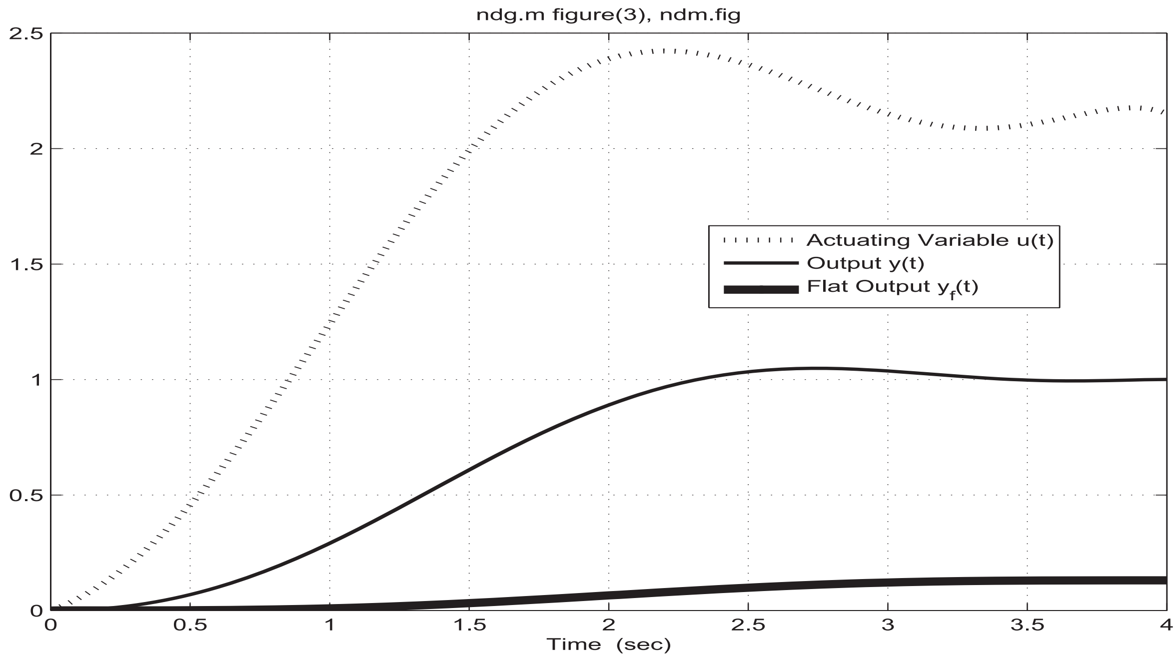


Figure 10: Actuating variable, original and flat output

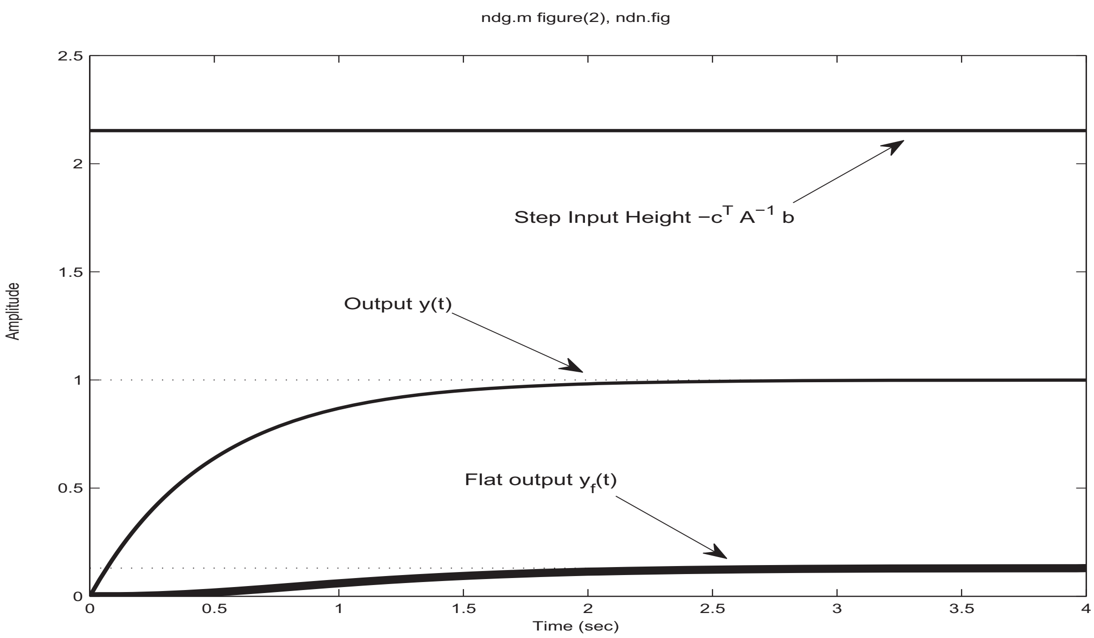


Figure 11: Actuating variable $u(t)$ which is assumed a step. Step responses: output $y(t)$ and flat output $y_f \equiv z_1(t)$, for comparison

B Basic Matrix Algebra and Differential Quotients

$$\text{Unit vector } \mathbf{e}_k = (0,0,\dots 1,\dots,0,0)^T = \mathbf{e}_k^{(m\times 1)} \quad (1 \text{ at position } k \text{ only}) \tag{100}$$

$$\text{Kronecker matrix } \mathbf{E}_{ik}^{(n\times m)} = \mathbf{e}_i \mathbf{e}_k^T \in \mathcal{R}^{n\times m} \quad (1 \text{ at position } (n,m) \text{ only}). \tag{101}$$

$$\text{Change of order permitted:} \quad \mathrm{tr}[\mathbf{AB}] = \mathrm{tr}[\mathbf{BA}]\ . \tag{102}$$

$$\text{Permutation matrix } \mathbf{U}_{k,l} \triangleq \mathbf{U}_{k,l}^{(kl\times kl)} \triangleq \sum_{i=1}^{k}\sum_{j=1}^{l} \mathbf{E}_{ij}^{(k\times l)} \otimes \mathbf{E}_{ji}^{(l\times k)} = \sum_{i}^{k}\sum_{j}^{l} \mathbf{E}_{ij}^{(k\times l)} \otimes \left(\mathbf{E}_{ij}^{(k\times l)}\right)^T. \tag{103}$$

$$\text{Self-derivative matrix } \frac{\partial \mathbf{M}^{(k\times l)}}{\partial \mathbf{M}} = \bar{\mathbf{U}}_{k,l} = \bar{\mathbf{U}}_{k,l}^{(k^2\times l^2)} \triangleq \sum_{i=1}^{k}\sum_{j=1}^{l} \mathbf{E}_{ij}^{(k\times l)} \otimes \mathbf{E}_{ij}^{(k\times l)}\ . \tag{104}$$

Mixed product rule (only applicable if $\mathbf{A},\mathbf{D}$ and $\mathbf{B},\mathbf{G}$ are conformable)

$$(\mathbf{A}\otimes\mathbf{B})\ (\mathbf{D}\otimes\mathbf{G}) = (\mathbf{AD})\otimes(\mathbf{BG})\ . \tag{105}$$

$$(\mathbf{I}_r \otimes \mathbf{A})^v \equiv (\mathbf{I}_r \otimes \mathbf{A}^v). \tag{106}$$

General definition (*Brewer, J.W., 1978; 1978a; Vetter, W.J., 1970; Weinmann, A., 2008*)

$$\frac{\partial \mathbf{G}}{\partial \mathbf{M}} \triangleq \sum_{ij} \mathbf{E}_{ij}^{(r\times s)} \otimes \frac{\partial \mathbf{G}}{\partial M_{ij}} = \begin{pmatrix} \frac{\partial \mathbf{G}}{\partial M_{11}} & \frac{\partial \mathbf{G}}{\partial M_{12}} & \cdots & \frac{\partial \mathbf{G}}{\partial M_{1s}} \\ \frac{\partial \mathbf{G}}{\partial M_{21}} & \frac{\partial \mathbf{G}}{\partial M_{22}} & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \mathbf{G}}{\partial M_{r1}} & \frac{\partial \mathbf{G}}{\partial M_{r2}} & \cdots & \frac{\partial \mathbf{G}}{\partial M_{rs}} \end{pmatrix} \tag{107}$$

$$\mathbf{G}\in\mathcal{R}^{n\times m},\ \mathbf{M}\in\mathcal{R}^{r\times s},\ \frac{\partial \mathbf{G}}{\partial \mathbf{M}} \in \mathcal{R}^{nr\times ms}\ . \tag{108}$$

Differentiation with respect to a scalar p

$$\frac{\partial}{\partial p}\mathbf{A}(p)\mathbf{B}(p)\mathbf{C}(p) = \frac{\partial \mathbf{A}}{\partial p}\mathbf{BC} + \mathbf{A}\frac{\partial \mathbf{B}}{\partial p}\mathbf{C} + \mathbf{AB}\frac{\partial \mathbf{C}}{\partial p}\ . \tag{109}$$

The matrix product rules with $\mathbf{A}\in\mathcal{R}^{n\times m}, \mathbf{B}\in\mathcal{R}^{m\times q}, \mathbf{M}\in\mathcal{R}^{r\times s}$, are

$$\frac{\partial}{\partial \mathbf{M}}\mathbf{A}_M\mathbf{B}_M = \frac{\partial \mathbf{A}_M}{\partial \mathbf{M}}(\mathbf{I}_s\otimes\mathbf{B}_M) + (\mathbf{I}_r\otimes\mathbf{A}_M)\frac{\partial \mathbf{B}_M}{\partial \mathbf{M}}\ , \qquad \frac{\partial(\mathbf{A}_M\mathbf{B}_M)}{\partial \mathbf{M}} \in \mathcal{R}^{nr\times qs}\ , \tag{110}$$

$$\frac{\partial \mathbf{A}_M\mathbf{B}\mathbf{C}_M}{\partial \mathbf{M}} = \frac{\partial \mathbf{A}_M}{\partial \mathbf{M}}(\mathbf{I}_s\otimes\mathbf{B}\mathbf{C}_M) + (\mathbf{I}_r\otimes\mathbf{A}_M\mathbf{B})\frac{\partial \mathbf{C}_M}{\partial \mathbf{M}}\ , \qquad \mathbf{B} = \text{constant}, \tag{111}$$

$$\frac{\partial \mathbf{A}\mathbf{B}_M\mathbf{C}}{\partial \mathbf{M}} = (\mathbf{I}_r\otimes\mathbf{A})\frac{\partial \mathbf{B}_M}{\partial \mathbf{M}}(\mathbf{I}_s\otimes\mathbf{C})\ , \qquad \mathbf{A},\ \mathbf{C} = \text{constant}\ . \tag{112}$$

For $\mathbf{E}_1,\ \mathbf{E}_2,\ \mathbf{E}_3$ independent of $\mathbf{M}$:

$$\frac{\partial}{\partial \mathbf{M}}\mathbf{E}_1\ \mathbf{A}_M\ \mathbf{E}_2\ \mathbf{B}_M\ \mathbf{E}_3 = $$
$$(\mathbf{I}_r\otimes\mathbf{E}_1)\frac{\partial \mathbf{A}_M}{\partial \mathbf{M}}[\mathbf{I}_s\otimes(\mathbf{E}_2\ \mathbf{B}_M\ \mathbf{E}_3)] + [\mathbf{I}_r\otimes(\mathbf{E}_1\ \mathbf{A}_M\ \mathbf{E}_2)]\frac{\partial \mathbf{B}_M}{\partial \mathbf{M}}(\mathbf{I}_s\otimes\mathbf{E}_3)\ . \tag{113}$$

Derivative of the inverse matrix (*Vetter, W.J., 1970*)

$$\frac{\partial \mathbf{A}_M^{-1}}{\partial \mathbf{M}} = -(\mathbf{I}_r\otimes\mathbf{A}_M^{-1})\frac{\partial \mathbf{A}_M}{\partial \mathbf{M}}(\mathbf{I}_s\otimes\mathbf{A}_M^{-1}) \qquad \mathbf{A}_M\in\mathcal{R}^{n\times m}\ ,\mathbf{M}\in\mathcal{R}^{r\times s}, \tag{114}$$

since $(\mathbf{N} \otimes \mathbf{M})^{-1} = \mathbf{N}^{-1} \otimes \mathbf{M}^{-1}$. Note the unchanged order of $\mathbf{N},\ \mathbf{M}$.

$$\text{For } \mathbf{X} \in \mathcal{R}^{n\times n} \quad \frac{\partial(\mathbf{X}\mathbf{X}^{-1})}{\partial \mathbf{X}} = \frac{\partial \mathbf{I}_n}{\partial \mathbf{X}} = \mathbf{0} \quad (\mathbf{M} \text{ replaced by } \mathbf{X}) \tag{115}$$

$$\frac{\partial \mathbf{X}}{\partial \mathbf{X}}(\mathbf{I}_n \otimes \mathbf{X}^{-1}) + (\mathbf{I}_n \otimes \mathbf{X})\frac{\partial \mathbf{X}^{-1}}{\partial \mathbf{X}} \overset{(110)}{=} \mathbf{0} \tag{116}$$

$$\frac{\partial \mathbf{X}^{-1}}{\partial \mathbf{X}} \overset{(104)}{=} -(\mathbf{I}_n \otimes \mathbf{X}^{-1})\bar{\mathbf{U}}_{n,n}(\mathbf{I}_n \otimes \mathbf{X}^{-1}) \tag{117}$$

$$\frac{\partial \mathbf{X}^{-1}}{\partial \mathbf{k}_r} \overset{(117)}{=} -(\mathbf{I}_n \otimes \mathbf{X}^{-1})\frac{\partial \mathbf{X}}{\partial \mathbf{k}_r}\mathbf{X}^{-1} \tag{118}$$

$$\frac{\partial \mathbf{X}^{-1}}{\partial p} = -\mathbf{X}^{-1}\frac{\partial \mathbf{X}}{\partial p}\mathbf{X}^{-1} \ . \tag{119}$$

Power series expansion:

$$(\mathbf{X} + \Delta\mathbf{X})^{-1} \doteq \mathbf{X}^{-1} - \mathbf{X}^{-1}\Delta\mathbf{X}\cdot\mathbf{X}^{-1} + \mathbf{X}^{-1}\Delta\mathbf{X}\cdot\mathbf{X}^{-1}\Delta\mathbf{X}\cdot\mathbf{X}^{-1} - + \ldots \tag{120}$$

$$\Delta(\mathbf{X}^{-1}) \doteq -\mathbf{X}^{-1}(\Delta\mathbf{X})\mathbf{X}^{-1} \ . \tag{121}$$

$$\begin{aligned}(\mathbf{A} + \Delta\mathbf{A})^3 \quad = \quad & \mathbf{A}^3 + \mathbf{A}^2\Delta\mathbf{A} + \mathbf{A}\Delta\mathbf{A}\cdot\mathbf{A} + \Delta\mathbf{A}\cdot\mathbf{A}^2 \\ & + \mathbf{A}\Delta\mathbf{A}^2 + \Delta\mathbf{A}\cdot\mathbf{A}\cdot\Delta\mathbf{A} + \Delta\mathbf{A}\cdot\mathbf{A}\cdot\Delta\mathbf{A} + \Delta\mathbf{A}^2\mathbf{A} + \Delta\mathbf{A}^3\end{aligned} \tag{122}$$

$$(\mathbf{A} + \Delta\mathbf{A})^v \doteq \mathbf{A}^v + \sum_{i=1}^{i=v}\mathbf{A}^{v-i}\cdot\Delta\mathbf{A}\cdot\mathbf{A}^{i-1}, \quad v \geq 1 \tag{123}$$

$v = 2$:

$$\frac{\partial \mathbf{A}^2}{\partial \mathbf{M}} = \frac{\partial \mathbf{A}}{\partial \mathbf{M}}(\mathbf{I}_s \otimes \mathbf{A}) + (\mathbf{I}_r \otimes \mathbf{A})\frac{\partial \mathbf{A}}{\partial \mathbf{M}} \quad \mathbf{M} \in \mathcal{R}^{r\times s} \tag{124}$$

$v \geq 3$:

$$\frac{\partial}{\partial \mathbf{M}}\mathbf{A}^v \overset{(110)}{=} \frac{\partial \mathbf{A}}{\partial \mathbf{M}}(\mathbf{I}_s \otimes \mathbf{A}^{v-1}) + (\mathbf{I}_r \otimes \mathbf{A})\frac{\partial \mathbf{A}^{v-1}}{\partial \mathbf{M}} \quad \in \mathcal{R}^{nr\times qs} \tag{125}$$

$$= \frac{\partial \mathbf{A}}{\partial \mathbf{M}}(\mathbf{I}_s \otimes \mathbf{A}^{v-1}) + \sum_{i=1}^{v-2}(\mathbf{I}_r \otimes \mathbf{A}^i)\frac{\partial \mathbf{A}}{\partial \mathbf{M}}(\mathbf{I}_s \otimes \mathbf{A}^{v-i-1}) + (\mathbf{I}_r \otimes \mathbf{A}^{v-1})\frac{\partial \mathbf{A}}{\partial \mathbf{M}} \tag{126}$$

$$= \sum_{i=1}^{v}(\mathbf{I}_r \otimes \mathbf{A}^{i-1})\frac{\partial \mathbf{A}}{\partial \mathbf{M}}(\mathbf{I}_s \otimes \mathbf{A}^{v-i}) \ . \tag{127}$$

and for direct transfer purposes, only, $\mathbf{k}_r \in \mathcal{R}^r$

$$\frac{\partial \mathbf{A}^{v+1}}{\partial \mathbf{k}_r} = \sum_{i=1}^{v+1}(\mathbf{I}_r \otimes \mathbf{A}^{i-1})\frac{\partial \mathbf{A}}{\partial \mathbf{k}_r}\mathbf{A}^{v-i+1}) \ , \quad v \geq 0 \ . \tag{128}$$

$$\frac{\partial \mathbf{A}^{k-1}}{\partial \mathbf{k}_r} = \sum_{i=1}^{k-1}(\mathbf{I}_r \otimes \mathbf{A}^{i-1})\frac{\partial \mathbf{A}}{\partial \mathbf{k}_r}\mathbf{A}^{k-i-1} \ , \quad k \geq 2 \ . \tag{129}$$

$$\frac{\partial \mathbf{A}^{k-1}}{\partial \mathbf{k}_r} = \sum_{j=1}^{k-1}(\mathbf{I}_r \otimes \mathbf{A}^{j-1})\frac{\partial \mathbf{A}}{\partial \mathbf{k}_r}\mathbf{A}^{k-j-1} \ , \quad k \geq 2 \ . \tag{130}$$

For differentiation with respect to $A_{i,j}$

$$\frac{\partial \mathbf{A}^{k-1}}{\partial A_{i,j}} = \sum_{v=1}^{v=k-1} \mathbf{A}^{v-1}\mathbf{E}_{i,j}\mathbf{A}^{k-v-1} \ , \qquad k \geq 2 \ . \tag{131}$$

$$\frac{\partial \mathbf{A}^{N}}{\partial A_{i,j}} = \sum_{v=1}^{v=N} \mathbf{A}^{v-1}\mathbf{E}_{i,j}\mathbf{A}^{N-v} \ , \qquad n \geq 1 \ . \tag{132}$$

$$\frac{\partial \mathbf{b}\mathbf{k}_r^T}{\partial \mathbf{k}_r} = (\mathbf{I}_n \otimes \mathbf{b})\mathbf{U}_{n,1} = (\mathbf{I}_n \otimes \mathbf{b}) \ , \qquad \mathbf{b} = \text{constant}, \mathbf{k}_r \in \mathcal{R}^n \tag{133}$$

$$\frac{\partial \mathbf{k}_r\mathbf{b}^T}{\partial \mathbf{k}_r} = \bar{\mathbf{U}}_{n,1}\mathbf{b}^T \ , \qquad \mathbf{b} = \text{constant}, \mathbf{k}_r \in \mathcal{R}^n \ . \tag{134}$$

For a rectangular matrix $\mathbf{H} \triangleq \text{matrix}_{i,k}[H_{i,k}] \in \mathcal{R}^{n\times m}$, using $\mathbf{H} \equiv \mathbf{H} \otimes 1$

$$(\sum_{i=1}^{n} \mathbf{e}_i^T \otimes \mathbf{h}_i)\mathbf{H}(\sum_{k=1}^{m} \mathbf{e}_k^T \otimes \mathbf{g}_k) = \sum_{i=1}^{n}\sum_{k=1}^{m}(\mathbf{e}_i^T\mathbf{H}\mathbf{e}_k) \otimes (\mathbf{h}_i\mathbf{g}_k^T) = \sum_{i=1}^{n}\sum_{k=1}^{m} H_{ik}\mathbf{h}_i\mathbf{g}_k^T \ . \tag{135}$$

The derivative of the Kronecker product of $\mathbf{A}_M \in \mathcal{R}^{n\times m}$ and $\mathbf{B}_M \in \mathcal{R}^{k\times l}$ with respect to a matrix $\mathbf{M} \in \mathcal{R}^{r\times s}$ is

$$\frac{\partial(\mathbf{A}_M \otimes \mathbf{B}_M)}{\partial \mathbf{M}} = \frac{\partial \mathbf{A}_M}{\partial \mathbf{M}} \otimes \mathbf{B}_M + (\mathbf{I}_r \otimes \mathbf{U}_{nk})\Big(\frac{\partial \mathbf{B}_M}{\partial \mathbf{M}} \otimes \mathbf{A}_M\Big)(\mathbf{I}_s \otimes \mathbf{U}_{lm}) \ . \tag{136}$$

Note that $\mathbf{I}_r, \mathbf{U}_{nk}$ etc. are not conformable for matrix multiplication.

The derivative with respect to a scalar p

$$\frac{\partial(\mathbf{A}_M \otimes \mathbf{B}_M)}{\partial p} \overset{(136)}{=} \frac{\partial \mathbf{A}_M}{\partial p} \otimes \mathbf{B}_M + \mathbf{U}_{nk}\Big(\frac{\partial \mathbf{B}_M}{\partial p} \otimes \mathbf{A}_M\Big)\mathbf{U}_{lm} \ . \tag{137}$$

$$\frac{\partial \mathbf{e} \otimes \mathbf{B}}{\partial p} \overset{(137)}{=} \frac{\partial \mathbf{B}}{\partial p} \otimes \mathbf{e} \ , \quad \mathbf{e} = \text{constant. Note the change of order. } (\mathbf{U}_{n,1} = \mathbf{I}_n, \ \mathbf{U}_{1,n} = \mathbf{I}_n) \tag{138}$$

$$\frac{\partial \mathbf{A} \otimes \mathbf{e}}{\partial p} \overset{(137)}{=} \frac{\partial \mathbf{A}}{\partial p} \otimes \mathbf{e} \ , \quad \mathbf{e} = \text{constant} \ . \tag{139}$$

Derivative of the trace

$$\frac{\partial \text{tr}[\mathbf{U}]}{\partial \mathbf{k}} = \sum_{i=1}^{n} \frac{\partial U_{ii}}{\partial \mathbf{k}} = \sum_{i=1}^{n} \frac{\partial \mathbf{e}_i^T\mathbf{U}\mathbf{e}_i}{\partial \mathbf{k}} \overset{(112)}{=} = \sum_{i=1}^{n}(\mathbf{I}_n \otimes e_i^T)\frac{\partial \mathbf{U}}{\partial \mathbf{k}}\mathbf{e}_i \ . \tag{140}$$

Select a vector of subtraces from a rectangular matrix $\mathbf{Z} \in \mathcal{R}^{n^2\times n}$:

$$\text{trh}[\mathbf{Z}] \triangleq \sum_{i=1}^{n}\{\mathbf{e}_i \cdot \text{tr}[(\mathbf{e}_i^T \otimes \mathbf{I}_n)\mathbf{Z}]\} \ . \tag{141}$$

An Integrated Safety Monitoring System Design for Human Robot Interactive Tasks

O. Ogorodnikova, Member IEEE
Department of Mechatronics, Optics and Information Engineering,
Budapest University of Technology
Muegyetem rkp. 3-9, H-111, Budapest, Hungary

Received February 11, 2010

Abstract
We propose a theoretical approach based on the safety modes evaluation, defined as permissive states of the system, in which a specific set of monitoring rules is applied. In this research an integrated architecture to monitor the human-robot interactive workspace is presented. The monitoring system is modeled as a separated unit that operates with the information received from the expert system's protocols with the results on the risk analysis, robot controller and the external sensory devices which sense an ambient environment, including human in vicinity. This information after being fused and processed enters a monitoring system decision making unit, where with compliance to monitoring rules, safety modes have been evaluated and activated.

Keyword: Robotics, HRI, Safety Monitoring

1 Introduction

The question of safety in robotics is becoming more and more crucial since the direct hazard, which is the robot itself, cannot be fully isolated in view of the new tendencies in robotics. Thus, safety issue in the last decade is one of the most significant challenges facing engineers in the deployment of advanced robots.

Most of the safety standards are dealing with the safe installation, programming, and operation of robot systems (ANSI/RIA R15.06, 1999; ANSI B11.TR3, 2000; ANSI/RIA/ISO 10218, 2007; ANSI/RIA/ISO 10218-2, 2009). These standards typically contain specifications for operator training, special safeguards for teach-mode operation, and barrier interlock requirements for operational modes. Despite the existence of these safety guidelines, and their general incorporation into the safety practices of industry, there are still a number of serious accidents involving industrial robots occur (http://www.osha-slc.gov). Furthermore, the existing safety manuals are not adequate enough for the rapidly expanding field of industrial and service robots, where the manipulators are expected to work in close conjunction with humans. There are a number of researches devoted to this aspect in diverse fields. An explicit survey of the recently developed safety related tools and approaches in the human robot cooperation field was presented by Santis, A. and Siciliano, B., 2008.

In this paper we are also giving a consideration to that issue by presenting safety monitoring system concept that is based on the levels of interaction definition and safety modes evaluation. Similar approach was introduced by researchers from LAAS laboratory (Guiochet, J. et al., 2008), where the idea of safety modes evaluation and control was elaborated. The present work is aimed to further develop this concept and integrate it into the complex safety monitoring system considering own recourses.

2 Safety Monitoring System Architecture

A theoretical approach of the safety monitoring system which is based on the safety modes evaluation and the safety criteria integration is proposed.

Practically, safety monitoring system (SMS) receives protocols (set of rules and requirements) for each identified task from the Safety Expert System (SES) (Ogorodnikova, O., 2008a) and integrate them into a monitoring algorithm for the decision making process. In real-time operations, the system assesses the robot state, controller's inputs, motor commands, and decides whether it is safe to perform the ongoing robot operation or not. Through continuous monitoring of the robot operation and sensory system characteristics, SMS matches this information with the required for the particular task and associates this data with a corresponding safety mode. The safety monitor activates the corresponding safety mode, and checks a specific set of monitoring rules. In the next step, SMS anticipates dangerous situations by overriding these permissible characteristics and, depending on the rate of the violation, sends the corresponding signals to the robot controller and to the human interface. Personnel awareness about a hazard, that robot's abnormal state might cause, is enhanced by means of an awareness system activation that generates corresponding vibrotactile and visual signals, indicating the overall system state. (Ogorodnikova, O., 2008b) The robot's reaction strategies can be also modified, becoming more flexible than the emergency stop, that can improve the productivity and efficiency of the task performance.

3 Interaction Levels Introduction

Safety monitoring system is operating with the information mainly derived from the SES, robot controller and the sensory system. On the basis of this information and applied rules for the decision making unit, the output (response) of the monitoring system is changing. For instance, an interval of the monitoring area for each safeguarding mode is varied depending on the robot's operational characteristics and configurations. A task itself also brings its own modifications as well as a human factor's diverse characteristics. Each task has been correlated to the distance that supposed to be kept during the interaction (see Tab. 1).

Table 1: Interaction levels and associated tasks differentiation

Interaction Distance	Description	Human Task
L1	Inside restricted Robot work space	Collaboration Teaching
L2	Outside the operational zone, within immediate space in the restricted zone (in close vicinity)	Operation Programming
L3	In safeguard space, within the arm maximal reach	Verification, Monitoring
L4	Outside robot maximal reach	Observing

The first level (L1) corresponds to tasks involving overlapping of the workspaces of the human (operator) and the robot during the task performance, where even physical contact is allowed. In the next level (L2), agents are invisibly separated whether by the task distribution or by the defined control strategy. The human, due to the specificity of the task, can carry out his/her task in a very close proximity to the robot. Within this level human is allowed to enter the restricted workspace (monitored by the safeguarding system), but not the operating space (task required zone) of the robot. The third level (L3) is located further away from the second level, but an operator may still be located within the robot arm's reach and can therefore be exposed to a certain degree of danger or risk of injury. Finally, the fourth interaction level (L4) is defined as the level outside the robot working envelope, but this area is not protected from thrown objects or released energy (radiation).

Boarders between levels are defined based on human psychological, physiological characteristics, on the robot's systems initial and current state, task specific/requirements and the associated risk category. Psychologically evaluated distances implies personnel "well being" (feel safe) during coexistence with robot. These experimentally defined distances may also indicate limiting conditions. For instance, if the calculated value is smaller than the psychologically acceptable one, then the real safety distance value should be measured according to the largest one.

Each interaction level is correlated to the area where the likelihood and probability of injury relies on a certain extent. Therefore, the most dangerous levels should meet the most restrictive requirements. In compliance with the injury severity scale, the first interaction level L1 is correlated with the most restrictive criteria (pain tolerance), the second and the third levels were defined as more permissive levels and were associated with the criteria where 0 and 1% of the severe injury is only acceptable. The last interaction level L4 is the least dangerous since the likelihood of the contact between human and robot is very low, therefore, the criteria with 10% - 50% of the severe injury probability is used for a last interaction level numerical identification (Ogorodnikova, O., 2009).

4 Safety Modes Definition

Safety modes for the monitoring system were defined according to the guidelines in robotic safety (Kuka Roboter GmbH, 2002). These modes are: Manual 1 (the robot moves only as long as one of the enabling switches is held down, movements are executed with reduced velocity), Manual 2 (movements are executed at the programmed velocity), Automatic 1 (M3) or external (operating mode in which a host computer or PLC assumes control of the robot system), and Automatic 2 (M4), where movements are executed at the programmed velocity. During the safety modes formalization additional features were added to common, standardized robot operation modes. Thus, monitored system has been designed with a set of functional modes, linked with a many-to-one relation to the safety modes. The objective of safety modes introduction is to describe the dynamics of the safety monitor, identifying different sets of safety rules activated according to the current tasks.

For each safety mode and each transition between them, a set of rules and characteristics that should be monitored is different. The set of permissive or restrictive rules defines the robot's functional capabilities allowed for the current safety mode and the system's hazardous states, either related to the environmental conditions, or to the critical robot characteristics.

Safety mode's permissiveness may vary with the robot's type, its operating characteristics, task specific, etc. For instance, for the small, light robot with the low effective, boundaries on the speed will be increased due to decreasing danger that this robot may cause to a human. In the Tab. 2 the intervals of permissiveness over the safety-related factors for each safety mode are shown.

Table 2: Safety Modes Parameters Identification

Safety-related characteristics	**Safeguarding / Operational Mode permissiveness**			
	Mode 1	*Mode 2*	*Mode 3*	*Mode 4*
Robot related	$\mathbf{D_{R1}}$	$\mathbf{D_{R2}}$	$\mathbf{D_{R3}}$	$\mathbf{D_{R4}}$
R1:Robot arm speed	$r_1[0,v_1]$	$r_1[0,v_2]$	$r_1[0,v_3]$	$r_1[0,v_4]$)
R2: Robot Base speed (for mobile platforms)	$r_2[0,v_1]$	$r_2[0,v_2]$	$r_2[0,v_3]$	$r_2[0,v_4]$)
R3: Robot acceleration	$r_3[a_0,a_1]$	$r_3[a_1,a_2]$	$r_3[a_2,a_3]$	$r_3[a_3,a_4]$)
…….	…..	…….	…….	………
Rn-1: Drive torque	$r_{n-1}[\tau_0,\tau_1]$	$r_{n-1}[\tau_1,\tau_2]$	$r_{n-1}[\tau_2,\tau_3]$	$r_{n-1}[\tau_3,\tau_4]$)
Rn: Exerted Arm Force	$r_n[f_0,f_1]$	$r_n[f_1,f_2]$	$r_n[f_2,f_3]$	$r_n[f_3,f_4]$)
Distance related	**L1**	**L2**	**L3**	**L4**
L: Distance from hazard	$[0,l_1]$	$[l_1,l_2]$	$[l_2,l_3]$	$[l_3,\infty]$
Safeguard related	$\mathbf{D_{SS1}}$	$\mathbf{D_{SS2}}$	$\mathbf{D_{SS3}}$	$\mathbf{D_{SS4}}$
S1: Control	$s_1(x_1,..,x_m)$	$s_1(x_1,..,x_m)$	$s_1(x_1,..,x_m)$	$s_1(x_1,..,x_m)$
S2: Sensing System	$s_2(y_1,..,y_k)$	$s_2(y_1,..,y_k)$	$s_2(y_1,..,y_k)$	$s_2(y_1,..,y_k)$
S3: Physical Fixed Safeguard	$s_3(z_1,..,z_l)$	$s_3(z_1,..,z_l)$	$s_3(z_1,..,z_l)$	$s_3(z_1,..,z_l)$
S4: Awareness System	$s_4(w_1,..,w_p)$	$s_4(w_1,..,w_p)$	$s_4(w_1,..,w_p)$	$s_4(w_1,..,w_p)$
S5: Personnel Safeguarding	$s_5(w_1,..,w_p)$	$s_5(w_1,..,w_p)$	$s_5(w_1,..,w_p)$	$s_5(w_1,..,w_p)$

Safety modes permissive scope indicates when the system is entering a hazardous state. By means of the applied rules we can monitor conditions that must be maintained in a safety mode and conditions that must be fulfilled to allow transitions between safety modes.

The modes permissive intervals were defined on the basis of the safety-relevant characteristics, which are characterized by a set of permissive frames with dynamically changed variables according to every currently ongoing task. It is assumed that each safety mode contains robot related, distance and safeguarding related characteristics. For all safety-related characteristics a set of admissible domains or groups of characteristics can be defined (Guiochet, J., et al., 2008). In (1) parameter D_R denotes a set of domains for all robot related variables, where $\vec{R}$ is a generalized vector for functional elements r_n. Domain D^i_R defines the permissiveness of the robot relative variables associated with the mode M_i. The safety mode M_i is less permissive than the mode M_j i.e. conditions for the mode M_i operation are more restricted then for the mode M_j.

$$\vec{R}_n = (D^i{}_{r_n},...,D^m{}_{r_n}),\, D_{\vec{R}} = (D^i_{\vec{R}},..., D^m_{\vec{R}}),\; D^i{}_{\vec{R}} \subseteq (D^i_{r_1},...,D^i_{r_n}),\; D^j_{\vec{R}} \supseteq D^i_{\vec{R}} \tag{1}$$

Distance related permissive domain in (2) is defined similarly considering the previously derived values for the interaction levels.

$$\vec{L} = (D^i{}_l,...,D^m{}_l),\ D_{\bar{L}} = (D^i_{\bar{L}},...,D^m_{\bar{L}}),\ D^i{}_L \subseteq l_i,\ D^i_L \subseteq D^j_L \tag{2}$$

A situation with the safeguarding related domains evaluation and their relation is rather different. The permissive interval for each safety mode cannot be precisely identified because elements of the set related to the particular safety mode might be transferred to other modes as well (see Fig. 2). Therefore, for the domain transition rules representation the theory of the intersectional sets was applied. The relationship between categories/domains is described as:

$$\vec{S}_k = (D^i{}_{s_k},...,D^m{}_{s_k}),\ D_S = (D^i_{\bar{S}},...,D^m_{\bar{S}}),\ D_{\bar{S}} \subseteq (D^i_{s_1},...,D^i_{s_5}),\ D^i_{\bar{S}} \cap D^j_{\bar{S}} \tag{3}$$

4.1 Robot Related and Distance Related Categories

Boundary values for the robot related domain (D_R) and its sets (R), as was mentioned above, are estimated on the basis of the danger index approach, where robot parameters as acceleration, force, velocity and effective mass (inertia) shouldn't exceed the certain critical characteristics. For instance, with respect to the force based danger criteria safety modes and consequently, interaction levels 1 and 2 were associated with "No pain" and "No injury" areas, where maximum allowed forces are 130N and 660N respectively (for a head injury estimations). Restrictions for manipulator velocity intervals for each interaction level are identified according to the robot's operational characteristics such as interface stiffness and effective masses (inertia) in compliance with the required safety level. For the more permissive levels (3, 4) manipulator operating ranges will be wider correspondingly. (Ogorodnikova, O., 2009)

The calculations for the intervals in the distance related domain (D_L) mainly depend on the obtained manipulator permissive parameters, where robot operating velocities and working zones are the determinative parameters, however, under certain conditions, the physiological/psychological factor becomes determinant. A minimal safe distance from the hazard (L_s) is computed as:

$$L_s = v_i(T_i + T_c + T_s) + v_h T_r \tag{4}$$

Where v_i is a robot operating velocity, T_i, T_c, T_s - robot stopping, control system and sensory response time correspondingly, v_h and T_r –personnel moving speed to hazard and the time needed to stop the motion respectively.

For the safety mode control algorithm, safe distance was estimated according to the interaction levels and with some human factor characteristics in consideration.

The following distance scale was yielded:

$$\begin{aligned} &L1 = D1 \\ &L2 = [D1 + v_{2\max} T_{st};\ D1 + 0{,}3;\ \infty] \\ &L3 = [v_{3\max} T_{st};\ D2 + H;\ \infty],\ \text{where}\ T_{st} = T_i + T_c + T_s \\ &L4 = [D2 + H;\ D1 + 1{,}2;\ \infty] \end{aligned} \tag{5}$$

Where T_{st} is a time required to "notice" hazard and to cease robot movements. Parameters v_{2max} and v_{3max} are the maximum velocities of the robot allowed on the levels 2 and 3; D1 is a size of the robot operating work space (max), D2 is a maximum reach of the manipulator arm. Constant values 0,3m and 1,2m indicate the optimal distances of interaction for the corresponding levels based on the psychological and ergonomic evaluations (Nagamachi, M., 1983; Nomura, T., Kanda, T., 2003; Bethel, C., *et al.*, 2007; Ogorodnikova, O., 2007a). However, these characteristics are rather relative and may vary from each robot type, task specific and personnel individual characteristics.

For each interaction level the fuzzy membership functions as the dynamically changing characteristics were defined (Zadeh, L., 1983). These parameters vary according to a current value of the variables presented in (5) and other related characteristics.

With the thin lines in the Fig. 1 we indicated dynamically changing conditions, with the thick lines - safety modes bounding conditions and the areas of the levels "strength" with a reference to the Fuzzy logic theory.

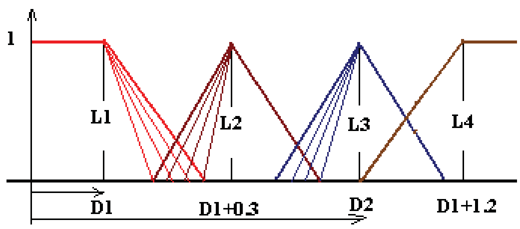


Figure 1: Distances scaling (fuzzy sets and membership functions)

In our research most of the evaluations were related to the industrial articulated arm type robots, but similar examinations can be provided for different robot types and application fields.

4.2 Safeguard system related category

To switch from one authorized mode to another in both directions we need to validate mode related safeguarding category.

Table 3: Transition rules for Safeguarding Category

$SS1 \leftrightarrow SS2 \leftrightarrow SS3 \leftrightarrow SS4: \; SS1 \cap SS2 \cap SS3 \cap SS4$
$SS1 \leftrightarrow SS3, SS2 \leftrightarrow SS4: \; SS1 \cap SS3 \vee SS2 \cap SS4$
$SS1 \cap (SS2, SS3) = \{s_n \mid s_n \in SS1 \wedge s_n \in SS2 \wedge s_n \in SS3\}$
$s_n \in (SS2 \cap SS3) > s_n \in (SS2 \cap SS4)$
$SS2 \cap (SS3, SS4) = \{s_n \mid s_n \in SS2 \wedge s_n \in SS3 \wedge s_n \in SS4\}$
$s_n \in (SS1 \cap SS2) > s_n \in (SS1 \cap SS3)$
$SS1 \leftrightarrow SS4 = 0$

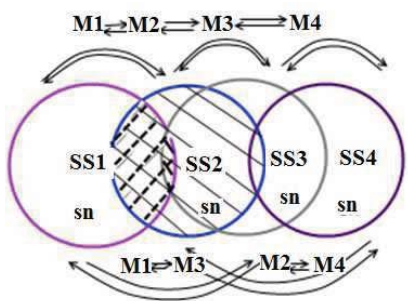


Figure 2: Safeguard Modes transition modeling

As it's seen in Fig. 2, safeguarding elements from one category can be partially found in the following two with a different extent.
For instance, providing a step by step transition from one mode to another we obtain 4 intersected sets, i.e. safety categories. Each of them contains its own set of elements which become common in the intersected regions. From the Fig. 2 and Tab. 3 we can easily define that the direct switch from SS1 to SS4 (and vice-versa) is not possible as elements of the domain SS1 don't enter the SS4 area that means absolutely different protective elements for these two categories. It is also seen that the automatic transitions SS1->SS3, SS2->SS4 are less desirable since the number of the common elements is much smaller than in the step-by-step move. Therefore, in the monitoring system for safety modes transition it is more preferable to establish continuous (gradual) modes changing, if it's not applicable (in the case of the mode absent authorization) we should provide an operational stop before the switch.
A list of safety categories with their elements is presented in the Tab. 4. Here a set of safeguarding elements related to the category *Control* (S1, Tab. 3) is shown, that incorporates the robot and the safety system control elements. These elements were chosen considering the safety standards guidelines, recent developments, and author's own experience in the field. Thus, for each safeguarding level a set of required solutions was determined. With *X* in Tab. 4 we depicted the most important protective means, that should be applied for this particular level of interaction. A compliance with this rule is a necessary requirement for the safety modes transition. With *R* we give an optional, recommended set of elements, which installation is under designer's own consideration.
To assess whether requirements were met or not, we referred to the Boolean algebra method. To each safeguard from the related category we set 0 or 1 value according to its appearance in the system. Default (minimum required elements) set is used as a pattern that we compare with the user's input data. A procedure of comparison is performed with the Boolean operator „meet" or „and". Thereby, after the parameters set has been tested (validated), monitoring system receives a corresponding signal.
Column "User" in the Tab. 4 shows user's (expert) entered information that is compared with the predefined pattern. An example in Fig. 3 shows the case of the successful (a) and not successful (b) assessments for the safeguarding categories SS1 and SS2 respectively.

Table 4: Safeguarding category assessment

Control *S1*	**SS1**	**SS2**	**SS3**	**SS4**	**User**
Robot Control					
• Force, Torque,	X	X	R	R	V
• 0-gravity, Low impedance	R	R			
• Position	X	X	X	X	V
• Compliance, Adaptive	X	R			
Safety Control					
• ESafety circuit	R	R			V
• BusSystem,	R	R	R		V
• Safety controller	X	X	R	R	V
• PLC, PSS, Relay	X	X	X	X	V

SS1 (a)								
P	1	0	1	0	1	0	1	1
$\wedge$	1	0	0	0	1	1	1	1
=	1	0	0	0	1	0	1	1

SS2 (b)								
P	1	0	1	0	1	0	1	1
$\wedge$	1	0	1	0	1	1	1	1
=	1	0	1	0	1	0	1	1

Figure 3: Safeguarding elements compatibility test

5 Conditions for the safety modes transition

Schematically relation between safety modes and their elements is presented in the Fig. 4. To switch from one safety mode to another we need to comply with a range of permissive rules. For a transaction allowance from the mode M_i to M_j and vice-versa a set of requirements has to be met (see Tab. 5).

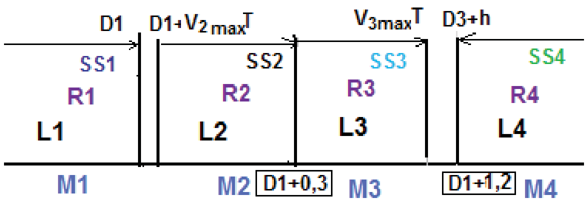


Figure 4: Interaction levels correlated Safety Modes distribution with associated transition parameters

In the direct transition a monitored system increases its functional abilities, if the distance related and safeguarding related conditions are fulfilled (e.g., absence of humans, appropriate safeguarding). If the rules are not fulfilled, the safety monitor should keep the system in the same mode, and reject the request for changing the mode. This implies that the monitored system does not switch to the desired functional mode.

Table 5: Modes transition conditions

Category	$M_i \rightarrow M_j$	$M_j \rightarrow M_i$
Robot related	$value_k \le \max(r_k^j)$ 1≤k≤n	$value_k \ge \min(r_k^i)$, 1≤k≤n
Distance related	$dist \ni [l_{j-1}, l_j]$	$dist \ni [l_{i-1}, l_i]$
Safeguarding related	$set_k \equiv s_k^j$, 1≤k≤4	$set_k \equiv s_k^i$, 1≤k≤4

In the reverse direction functional abilities decrease the permissiveness. Switching to a less permissive mode is guarded by restriction of the functional variables (for instance, speed has to be reduced). If those conditions are not verified, this means that the monitored system is unable to reach such an intending mode.

If the condition is not fulfilled when the mode change is requested, the safety monitor can engage the back-up mode for handling the violation of the mode condition or to trigger the stopping signal directly. In each safety mode, the monitor should be able to detect if there is a violation in the mode corresponding conditions. When it is not possible, the system cannot remain in the same safety mode, so the safety monitor must force a transition towards a safe state, back-up mode, in which actions are undertaken to reach conditions of a less permissive safety mode (for example, a back-up mode might correspond to activation of emergency braking). The function of the back-up mode is to possibly avoid emergency or not desirable stops of the system. Its commands force the robot to change its status by sending "slowing down" or "move faster" requests to the robot controller if there are not severe violations appear. Decision making unit decides whether operational or emergency stops should be activated. Stopping method depends on the danger of the operation and the level of violation. Modes 1, 2 are often accompanied by emergency stop.

Not consecutive modes transitions in most cases are conducted with the operational stop (standby) initiation. The overall transition chart is represented in the Fig. 5.

Below some transition rules for the gradual modes change conditions are given:

$M_i \rightarrow M_j$:

If M_j is authorized, D^j_{SS} is "true", D_{Li} is „true" then $D^i_R = D^j_R$, M_j „active";

If M_j is authorized, D^j_{SS} is "false" and (or) D_{Li} is „false" then $D^i_R \neq D^j_R$, M_j „not active".

If M_j is "active", D_{Lj} is "false" and D^j_R is "true" /"false" and D^i_{SS} is "false"/ "true" then OS;

$M_j \rightarrow M_i$:

If M_i is authorized, D^i_{SS} is "true", $D^j_R = D^i_R$ then $D_{Li} = D_{Lj}$, M_j „active";

If M_i is authorized, D^i_{SS} is "true", $D^j_R \neq D^i_R$ then M_i „not active";

If M_j is "active", D_{Li} is "false" and $D^i_R = D^j_R$ then ES;

If M_i is "active", D_{Li} is "false" and D^i_R is "true"/ "false" and D^i_{SS} is "false"/ "true" then OS/ES;

D_L is "true" if there is no human in the zone.

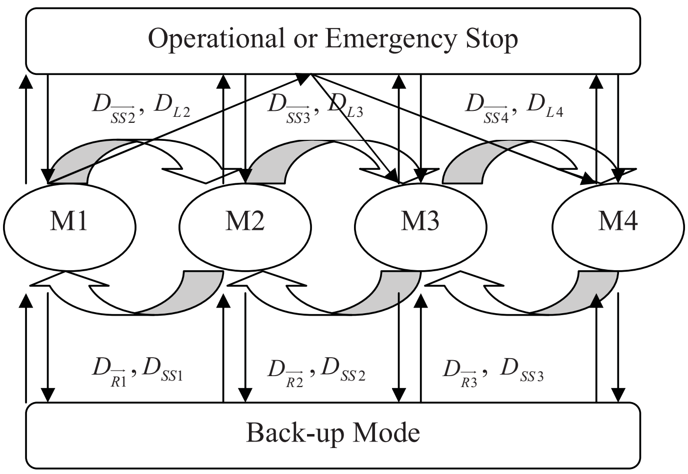


Figure 5: Safety modes transition model

In general, the monitoring unit should be able to detect any violations and inconsistency immediately after their occurrence. When it is not possible, the system cannot remain in the same safety mode, so the monitor must force a transition towards a safe state, in which actions are undertaken to reach conditions of a less permissive safety mode.

6 Method Application

The case study was provided for the human scanning system, project "Ruharobot" (see Fig. 6) where human is scanned means of the vertically moving frame with 4 digital cameras.

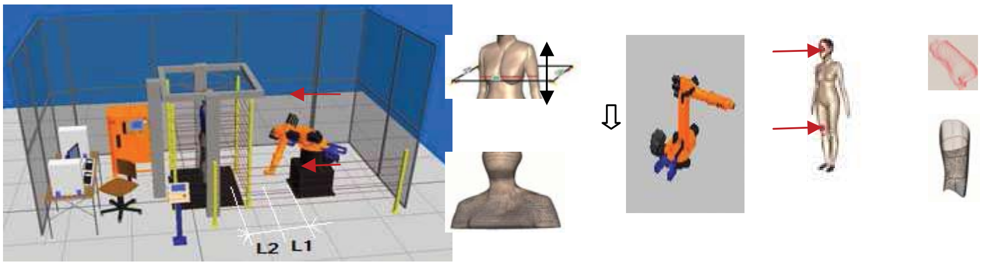


Figure 6: Human-Robot scanning system

KUKA Robot KR6 performs additional measurements with laser range finder mounted on the manipulator's end-effector (Tamas, P. Somlo, J., 2007; Tamas, P. et. al., 2007; Ogorodnikova, O., Olchanskij, D., 2007).
Three interaction levels with associated tasks were identified: robot teaching, program verifying and scanning, that is conducted in the controlled automatic regime (M3) with the human within the distance of the robot possible reach. A risk assessment was provided with the aid of the Safety Expert system assuming that 2 personnel are engaged in the tasks performance. Human factor analysis (authorization for the tasks) was carried out for an operator, who plays an active role in the whole operational process and for the personnel who plays a passive role in the performance (being scanned). After a preliminary task analysis and the risk assessment 3 main risk reduction categories for each interaction level were identified (R1, R3, R5). Risk reduction was achieved by safeguarding means installation and application according to the safety standards guidelines and in view of the information obtained from the robot related accident surveys, crush tests and existing available solutions, etc. (Piggin, R., 2005; KUKA, 2005; Haddadin, S., et al., 2007).
All safety related sensing devices are interconnected via safety modular PLC (programmable logic controller). Robot system comprises its own embedded safe technology that monitors velocity and acceleration of the axes and enables a safe operational stop of the robot. The working space is limited by adjustable software limit switches for all axes backed up by mechanical limit stops in case these limit switches are overrun. Moreover, robot controller is connected to an ESC (electric safety circuit) that provides a failsafe monitoring and evaluation of safety elements. Human is sensed by the ultrasonic sensor mounted on the robot wrist and located near robot base scanner that observes all authorized zones verifying that there is no undesirable event occurs. Perimeter safeguarding system consists of safeguarding fences, light curtains and present sensing safety mats (Ogorodnikova, 2007b).

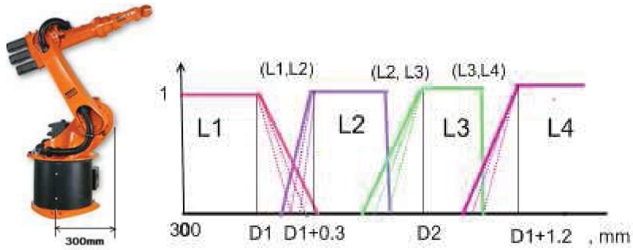


Figure 7: Interactional levels evaluation for KR6 robot

Levels of interaction were identified for the KR6 robot. For the more accurate and precise safety levels and control modes evaluation distances (fuzzy sets) were associated with the trapezoidal and gradual membership functions (see Fig. 7, where D1=0,9m, D2=1,5m)
Thin lines depict every level transient state. These boundary values were considered for the safety mode change. For instance, to switch from the mode M1 to M2 we monitor the maximum boundary values of the level L1 while in the opposite direction we are interested in the minimum value of the L2. Similarly we examine boundary values of the robot related characteristics. In the Tab. 6 we identify input parameters for the safety monitoring system. Mode 4 is not authorized for the considering tasks. For the simplicity, system monitors only manipulator's speed and force as the robot related characteristics. Monitoring distances (boundary values) are dynamically changing depending on the robot speed according to the robot speed. Provided safeguarding assessment proved sufficiency of the existing safety means for the first 3 monitored levels.

Table 6: Safety modes identification for the case study

Safety-related characteristics	**Safeguarding / Operational Mode permissiveness**			
	Mode 1	***Mode 2***	***Mode 3***	***Mode 4***
Robot related	R1	R2	R3	R4
Robot arm speed, mm/s	[0,140]	[%,250]	[%,650]	[%, max)
Arm Force, N	[f_0,10]	[%,50]	[%,150]	[%, max]
Distance related	L1	L2	L3	L4
Distance from hazard, cm	[30,110]	[±110,150]	[±150,200]	[±200,240, ∞]
Safeguarding related	SS1 (V)	SS2 (V)	SS3 (V)	SS4 (X)

To enhance human awareness about ongoing task execution and possible hazards, monitoring system is interconnected with the wearable vibro-tactile interface (Ogorodnikova, O., 2008b), that by means of the tactile and visual signalization provides personnel with the information about abnormal system state. A vibration frequency depends on the case danger and based on the analysis of the human's body (hand) sensitivity level under various conditions. In our research the sensitivity is considered as a function of the destructive noise, produced by robot's motors (under investigation). The highest applicable frequency is equal to the most severe danger level. A control algorithm for this device is also designed to indicate the

transition of the safety modes by low frequency output signal applied on personnel' hand (here) along with the visual signalization (green flash).
Figure 8 demonstrates an algorithm of the M1 and M2 safety modes transition according to the provided case study analysis.

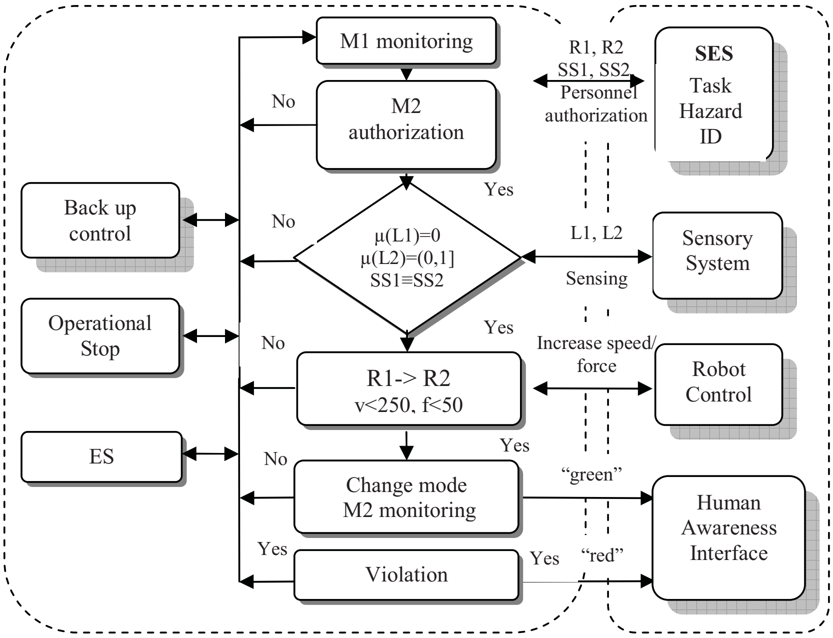


Figure 8: M1-M2 safety modes transition

7 Discussions and Conclusion

The case study was provided for 3 safety modes with the direct modes transmission. Currently, the switch from one mode to another relies on the sensory information and human vigilance. Therefore, to provide system dependability and reliability complexity and ambiguity in the signal fusion and interpretation should be avoided.
In the course of this research we faced with a problem of the data and necessary equipment insufficiency for the all examinations that were planned and have to be done for the theory validation. In spite of the fact that for an effective analysis an explicit knowledge about all elements as well as their functional characteristics is required, preliminary studies already have showed that this theory can be quite applicable for the related issues.
The objective of this study was to develop a concept that would provide enhanced human safety in robotized environment. We believe that introduced method can be considered as one of the possible response to the problem.
This study demonstrates that by means of the safety modes application most of the hazardous situations can be handled reliably quickly and safely. However, for the successful performance the monitoring system should be able to observe and to act independently to provide dependability in the decision making process and be possibly failure free to not cause an additional hazard.

Currently the method of the safety modes evaluation and transition rules formalization was simplified i.e. not all conditions and potential system behaviors were considered. In the future we are planning to overcome these issues by enlarging the system with additional concepts and definitions. Represented in this paper method hasn't found its practical application yet, however, we are convinced that this approach would significantly enhance human safety as well as the reliability in the human robot interaction domain. The work presented here was done with industrial robot application, but this approach is also can be applicable to mobile platforms and service robots as well, where human robot coexistence goes beyond a structured factory environment.

8 References

ANSI/RIA R15.06.1999, Risk Assessment and Risk Reduction: A Guide to Estimate, Evaluate and Reduce Risks Associated with Machine Tools, American National Standards Institute.

ANSI/RIA/ISO 10218-2.2009, Robots for industrial Environment-Safety Requirements, American National Standards Institute.

ANSI/RIA/ISO 10218-1.2007, Robots for industrial Environment-Safety Requirements, American National Standards Institute.

ANSI B11.TR3.2000, Risk Assessment and Risk Reduction - A Guide to Estimate, Evaluate and Reduce Risks Associated with Machine Tools, American national standards institute.

Bethel, C., Salomon, K., Murphy, R. and Burke, J., 2007, Survey of Psychophysiology Measurements Applied to Human-Robot Interaction. In: Proc. 16th IEEE Inter. Conf. on Robot & Human Interactive Communication, Jeju, Korea, Aug. 26 -29.

Guiochet, J., Powell, D., Baudin, E. and Blanquart, J. P., 2008, Online Safety Monitoring Using Safety Modes, In Proc.: work. on Technical Challenges for Dependable Robots in Human Environments, PASADENA, USA.

Haddadin, S., Albu-Schaffer, A., Hirzinger, G., 2007, Dummy crash-tests for the evaluation of rigid human-robot impacts. In: Proc. IARP-IEEE/RAS-EURON Work. on Technical Challenges for Dependable Robots in Human Environments, Roma, Italy.

KUKA Roboter GmbH, 2002, KR C1 software. Operator control, Documentation for KUKA KR6 robot, BH Release 3.2 05.00.02, p. 21.

KUKA Robot Group, 2005, Safety in robot system, Specification, www.kuka.com.

Nagamachi, M., 1983, Human Engineering Oriented Research on Industrial Robots, *Human Engineering*, Vol. 19-5, pp. 259-264.

Nomura, T. and Kanda, T., 2003, On Proposing the Concept of Robot Anxiety and Considering Measurement of It. In: Proc. IEEE Inter. Work. on Robot and Human Interactive Communication, California.

Ogorodnikova, O., 2007a, Human Weaknesses and Strengths in Collaboration with Robots, *Periodica Polytechnica*, Ser. Mech. Eng., Vol.52, No.1, pp. 1-9.

Ogorodnikova, O., 2007b, The role of risk assessment in human centered robotic work cell. In: Proceed. of the 16th International Workshop, RAAD 2007, Ljubljana, pp. 351-357.

Ogorodnikova, O. and Olchanskij, D., 2007, On a 3D scanning system design problem, *Periodica Polytechnica* Ser. Mech. Eng., Vol. 51 No. 1, pp. 1-6.

Ogorodnikova, O., 2008a, Development of an expert system for the risk assessment and safeguarding selection in a robot-human work environment, In: Proc. 7th GEPESZET conference on Mechanical Engineering, Budapest, May 29.

Ogorodnikova, O. 2008b, Creating an active awareness system for humans in robotic workcell, *Acta Polytechnica Hungarica*, Vol. 5, No. 2, pp. 11-20.

Ogorodnikova, O., 2009, A Fuzzy Theory in the Risk Assessment and Reduction Algorithms for a Human Centered Robotics, In: Proceed. of the 18th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN ’09, Toyama, Japan.

OSHA, 2002, see: http://www.osha-slc.gov/dts/osta/otm/ otm_iv/ otm_iv_4.html.

OSHA Technical Manual, Industrial robots and robot safety, see: http://www.osha.gov.

Piggin, R. 2005, Developments in Industrial Robotic Safety, *Industrial Robot: An International Journal*, Vol.32 Issue 4, pp. 303 – 311.

Santis, A., Siciliano, B., 2008, Safety issues for human-robot cooperation in manufacturing systems, Tools and perspectives in virtual manufacturing, VRTest, 2008, Napoli.

Tamas, P, Somlo J., 2007, Robotized Planning in Textile Industry, XV. OGÉT, International Meeting of Manufacturing Problems, Kolozsvár, pp. 374-378.

Tamás, P., Halasz M., Somlo J., 2007, 3d measuring of the Human Body by Robots, Int Conf. Of Innovation and Modelling of Clothing Eng. Proc. IMCEP 2007, Univ.of Maribor, pp. 109-115.

Zadeh, L., 1983, The Role of Fuzzy Logic in the Management of Uncertainty in Expert Systems, *Fuzzy Sets Systems*, 11, p. 199.

Vision Control for Hyperredundant Robots

D. Cojocaru, M. Ivanescu, R.T. Tanasie, S. Dumitru, F. Manta
Faculty of Automation, Computers and Electronics, University of Craiova, Romania

Abstract
A tentacle manipulator is a hyper-redundant or hyper-degree-of-freedom manipulator. Hyperredundant robots produce changes of configuration using a continuous backbone made of sections which bend. The lack of no discrete joints is a serious and difficult issue in the determination of the robot's shape. A solution for this problem is the vision based control of the robot, kinematics and dynamics. A tentacle arm prototype was designed and the practical realization is now running. The control system is an image – based visual servo control where the error control signal is defined directly in terms of image feature parameters. Camera calibration is the essential procedure for all such applications: positioning and orienting the cameras in order to support the accuracy of the image features extraction. The calibration for a pan/tilt/zoom camera shape is achieved by means of an engineered environment and a graphic simulation module.

1 Introduction

An ideal tentacle manipulator is a non-conventional robotic arm with an infinite mobility. It has the capability of takeing sophisticated shapes and of achieving any position and orientation in a 3D space.

Behavior similar to biological trunks, tentacles, or snakes may be exhibited by continuum or hyper-redundant robot manipulators. Hence these manipulators are extremely dexterous, compliant, and are capable of dynamic adaptive manipulation in unstructured environments, continuum robot manipulators do not have rigid joints unlike traditional rigid-link robot manipulators. The movement of the continuum robot mechanisms is generated by bending continuously along their length to produce a sequence of smooth curves (Walker, I.D. and Carlos Carreras, 2006), (Walker, I.D., Darren M. Dawsona, 2005). This contrasts with discrete robot devices, which generate movement at independent joints separated by supporting links.

The snake-arm robots and elephant's trunk robots are also described as continuum robots, although these descriptions are restrictive in their definitions and cannot be applied to all snake-arm robots. A continuum robot is a continuously curving manipulator, much like the arm of an octopus (Davies, J.B.C., 1998). An elephant's trunk robot is a good descriptor of a continuum robot. The elephant's trunk robot has been generally associated with an arm manipulation – an entire arm used to grasp and manipulate objects, the same way that an elephant would pick up a ball. As the best term for this class of robots has not been agreed upon, this is still an emerging issue. Snake-arm robots are often used in association with another device meant to introduce the snake-arm into the confined space.

However, the development of high-performance control algorithms for these manipulators is quite a challenge, due to their unique design and the high degree of uncertainty in their dynamic models. The great number of parameters, theoretically an infinite one, makes very difficult the use of classical control methods and the conventional transducers for position and orientation.

2 A Tentacle Manipulator: Project Description

The research group from the Faculty of Automation, Computers and Electronics, University of Craiova, Romania, started working in research field of hyperredundant robots over 20 years ago. The experiments started on a family of TEROB robots which used cables and DC motors. The kinematics and dynamics models, as well as the different control methods developed by the research group were tested on these robots.

2.1 Robotic arm

Starting with 2008, the research group designed a new experimental platform for hyperredundant robots. This new robot is actuated by stepper motors. The rotation of these motors rotates the cables which by correlated screwing and unscrewing of their ends determine their shortening or prolonging, and by consequence, the tentacle curvature. In the actual stage the manipulator is formed of three segments. All segments are cylindrical. The backbone of the tentacle is an elastic cable made out of steel, which sustains the entire structure and allows the bending. Depending on which cable shortens or prolongs, the tentacle bends in different planes, each one making different angles (rotations) respective to the initial coordinate frame attached to the manipulator segment – i.e. allowing the movement in 3D. Due to the mechanical design it can be assumed that the individual cable torsion, respectively entire manipulator torsion can be neglected. Even if these phenomena would appear, the structure control is not based on the stepper motors angles, but on the information given by the robotic vision system which is able to offer the real spatial positions and orientations of the tentacle segments.

The tentacle arm is designed to be actuated by 3-phase stepper motors. The interfaces are pulse direction based without rotation monitoring. Set-point position of the stepper motor is preset as a pulse signal by a controller via signal interface. A pulse corresponds to one step of the motor (0.5 pps-2.4M pps, Max. Acceleration Rate: 737M pps2, Speed resolution: 16-bit). An electronic relay contact reports operating readiness. The nominal torque MN is 2 Nm. Steps per revolution are selectable from 200 to 10000. The step angle α is selectable from 1.8° to 0.036°. Tree stepper motors are used for each segment of the tentacle. 4-Axis Stepper Motion Controller boards are used. It is a pulse train motion controller which provides T/S curve control, on-the-fly speed change, non-symmetric acceleration and deceleration profile control, and simultaneous start/stop functions. This controller also offers card index settings for multiple cards in one IPC system. The boards offer powerful speed change functions that can be executed while the axis is moving. After motion begins, the target speed can be changed as needed according to the application. By using either a software function or external input signal, the controller can perform simultaneously starts and stops on multiple axes in a one-card configuration, or multiple axes in a multiple-card application (our case).

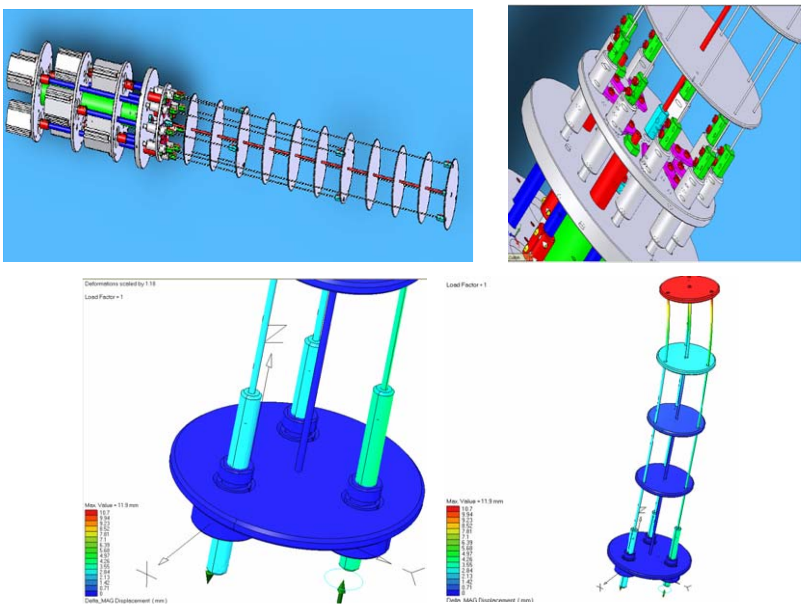


Figure 1a. CAD images of the tentacle design

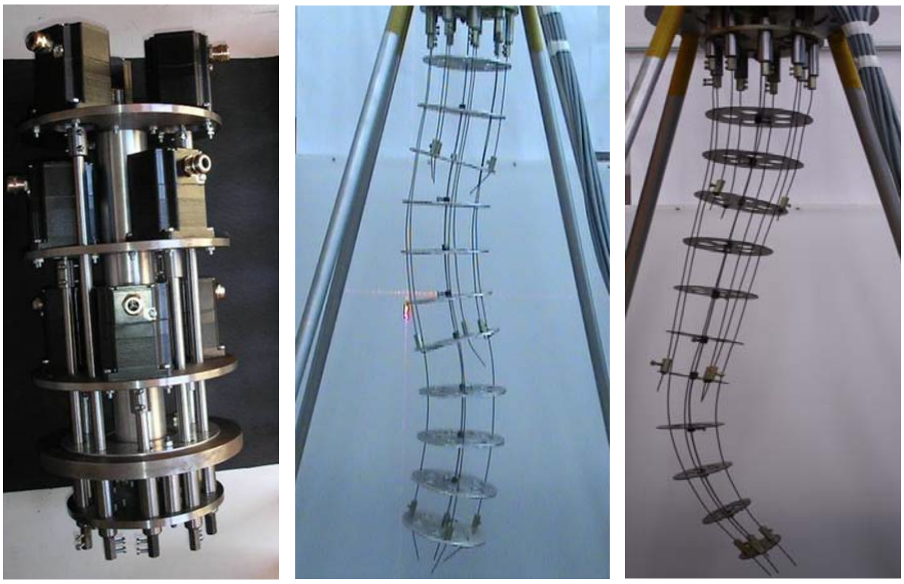


Figure 1b. Implementation of new model

A tentacle arm prototype was designed and the practical realization is now running. It is a cable based mechanism having, in the first implementation, three segments (CAD images during the simulation in Fig. 1a and new model implemntation in Fig. 1b).

2.2 Robot control

A tentacle manipulator is a hyper-redundant or hyper-degree-of-freedom manipulator and there has been a rapidly expanding interest in their study and construction lately. The control of these systems is very complex and a great number of researchers have tried to offer solutions for this difficult problem (Ivanescu, M., 1984). The inverse kinematics problem is reduced to determining the time varying backbone curve behavior. New methods for determining “optimal” hyper-redundant manipulator configurations based on a continuous formulation of kinematics are developed. The difficulty of the dynamic control is determined by integral-partial-differential models with high non-linearities that characterize the dynamic of these systems (Ivanescu, M., 2002).

First, the dynamic model of the system was inferred. The method of artificial potential was developed for these infinite dimensional systems. In order to avoid the difficulties associated with the dynamic model, the control law was based only on the gravitational potential and a new artificial potential (Grosso, E., 1996),(Hutchinson, S., 1996),(Kelly, R., 1996). Servoing was based on binocular vision, a continuous measure of the arm parameters derived from the real-time computation of the binocular optical flow over the two images, and is compared with the desired position of the arm. The control error function was built in 3D Cartesian space by the visual information obtained by two cameras in two image planes. The two 2D errors obtained in the two image planes were determined by the two differences between the actual and desired continuous angle values that define the projections of the arm shape. The plane errors can be considered as the errors of the arm shape. These errors were used to calculate the spatial error and a control law was synthesized.

The general control method is an image based visual servoing one instead of position based (Hannan, M.W. and I. D. Walker, 2005), (Ivanescu, M. and Cojocaru, D., 2007). By consequence, camera calibration based on intrinsic parameters (classic sense, not the one used in this paper) is not necessary.

Two video cameras provide two images of the whole robot workspace. The two images planes are parallel with XOY and ZOY planes from robot coordinate frame, respectively (Fig. 2). The cameras provide the images of the scene that stored in the frame grabber's video memory. Related to the image planes are defined two dimensional coordinate frames, called screen coordinate frames or image coordinate systems. Denote X_{S_1}, Y_{S_1} and Z_{S_2}, Y_{S_2}, respectively, the axes of the two screen coordinate frames provided by the two cameras. The spatial centers for each camera are located at the distances D1 and D2, with respect to the XOY and ZOY planes, respectively. The orientation of the cameras around the optical axes with respect to the robot coordinate frame, are noted by ψ and ϕ, respectively.

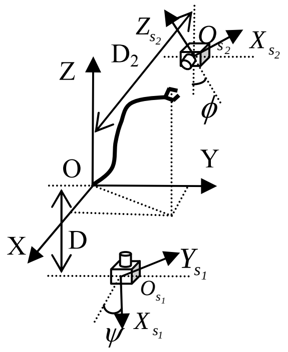


Figure 2. Cameras frames

The control system is an image – based visual servo control (Pressigout, M., 2004) where the error control signal is defined directly in terms of image feature parameters. The desired position of the arm in the robot space is defined by the curve Cd, or, in the two image coordinate frames $Z_{S_1}O_{S_1}Y_{S_1}$ and $Z_{S_2}O_{S_2}Y_{S_2}$, by the projection of the curve C (Fig. 3). The

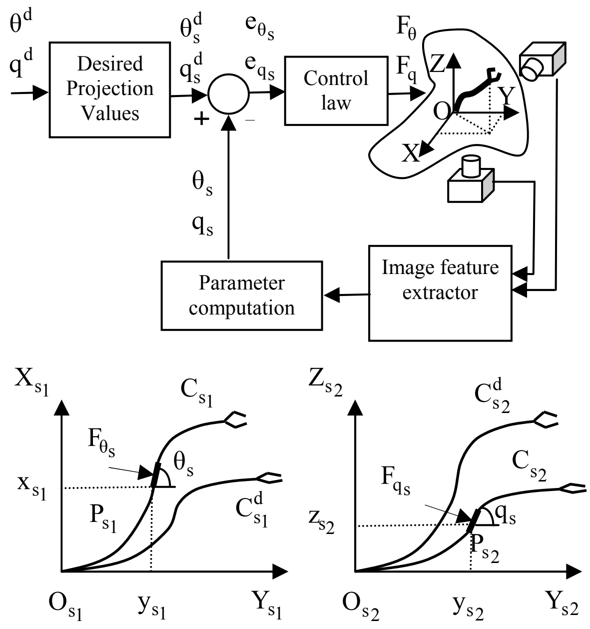


Figure 3. The global control system

control problem of this system is a direct visual servo-control, but the classical concept of the position control, in which the error between the robot end-effector and target is minimized, is not used. In this application the control of the shape of the curve in each point of the mechanical structure is used (Fig. 3). The method is based on the particular structure of the system defined as a "backbone with two continuous angles $\theta(s)$ and q(s)" (the tentacle bends in different planes, each one making different angles (rotations) respective to the initial coordinate frame attached to the manipulator segment, $\theta(s)$ and q(s) representing the bending angle and the plane rotation angle).

The control of the system is based on the control of the two angles $\theta(s)$ and q(s). These angles are measured directly or indirectly. The angle $\theta(s)$ is measured dircetly by the projection on the image plane $Z_{S_1}O_{S_1}Y_{S_1}$ and q(s) is computed from the projection on the image plane $Z_{S_2}O_{S_2}Y_{S_2}$.

The vision system is composed by two pan / tilt / zoom cameras and a frame grabber. In order to implement the visual-servoing system, a benchmark was organized (see Fig. 4) based on two color camera with 0.05lux low light sensitivity and the DT3162 frame grabber from Data Translation. The cameras have motorized Pant/Tilt/Zoom (10x optical zoom) and are mounted in perpendicular planes offering the input for the frame grabber. The Pant/Tilt/Zoom precision is sufficient for this step of the application development (Pan: range +135, 10 50/sec; Tilt: range +90 45, 7 25 /sec; Zoom: 1x~10x optical zoom). Two white screens are placed in front of the cameras in order to increase the image's contrast. The tentacle arm is placed between each camera and its screen.

The image processing tasks are performed using Global LAB Image2 from Data Translation. The robot control algorithms are implemented in a C++ program running on a Pentium IV PC.

Figure 4. Visual Control benchmark

3 Cameras Calibration

The term “camera calibration” in the context of this paper refers to positioning and orienting the two cameras at imposed values (Fig. 5). This calibration is performed only at the beginning, after that the cameras remain still. First, a zoom that maximizes the image resolution of the working space used by the manipulator is performed. Second, positioning of the two cameras brings the manipulator in the middle of the two images. Third, a pan / tilt orientation is performed (as descried later in the paper). At this step the manipulator is moved in a test position that allows free of (or minimum) errors calibration. The test images are compared to the images generated by the graphic simulator (ideal images) which represent references for the calibration operation.

In order, to ease the fulfil of the cameras calibration, a graphic simulator based on a 2D direct kinematics model was designed, implemented and used. By consequence, during the calibration procedure, the robot was commanded to bend in planes perpendicular to the cameras axes. Thus only the arching angle needs to be computed and a 2D model is sufficient to solve the problem. The next version of the software application introduces also the possibility to calibrate in 3D, the test positions corresponding to unrestricted planes orientation. A very important task in developing this application is to control the camera position and orientation. From this point of view, the calibration operation assures that the two cameras’ axes are orthogonal. In the beginning, the tentacle manipulator receives the needed commands in order to stand in a test pose (imposed position and orientation). The same commands are sent to the Graphic Simulator. Two different sets of images are obtained:

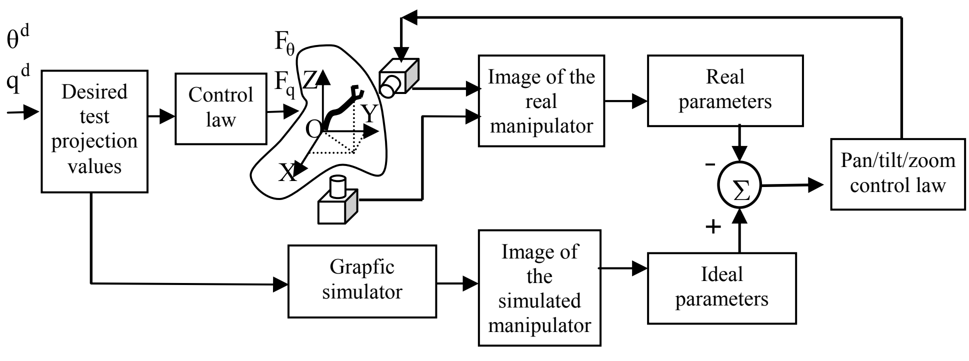


Figure 5. Camera calibration system

real images acquired by the real cameras and simulated images offered by the Graphic Simulator. From these two sets of images, two sets of parameters are computed: real parameters are computed from real images and, respectively, ideal parameters are computed from synthetic images. Comparing the two sets of parameters and knowing the image/parameters behavior for the camera orientation, the cameras are orientated (pan/tilt/zoom) in order to minimize the error.

3.1 Graphic simulator

A graphical simulator was designed and implemented (Cojocaru, D., 2008) in order to test the robot behavior under certain circumstances. The simulator approximates the curved segments of the hyperredundant robot and considers constant the length of the median arc of each segment. To ease the presentation, the term segment will be used in all that follows referring to the median segment (arched or un-arched). For the arched segment, its median arc remains constant. In this paper the term O-X angle will be used to denote the angle that the chord made by an arched element of the robot makes with the O-X axis of a selected reference system.

The inputs for the simulator are: robot configuration; robot initial position; rontrol laws for each of the segments of the hyperredundant robot. The robot configuration consists of the number of segments the hyperredundant robot has, the length of each segment and the angles that the cords make with the O-X axis. The arching angles are computed from these angles. An arching angle is defined as the angle made by the cord (determined by the ends of the arched segment) and the original un-arched segment. For the direct kinematics problem, the control of the robot simulation is accomplished by giving the O-X angles for each of the segments in their final position and the output of the simulation is the hyperredundant robot's end-effector final position in the operation space. In order to compute the final position of the end-effector and the hyperredundant robot's behavior during its motion, a few elements must be computed: the relation between the arching angle and the angle at center determined by the arched segment (this angle determines the length of the arc); the cord length; the relation between an O-X angle and an arching angle; the final arching angles – recurrent set. The computation of the relation between the arching angle and the angle at center determined by the arched segment is determined by the following axiom:

For camera calibration a direct kinematics model was used, thus the rotation angles for each segment are given. For a robot that has only rotation joints, the O-X angle increases (or decreases, depending on the selected positive direction) for each segment with the sum of rotation angles of each of the previous segments (including the current segment). This is true because the orthogonal system attached to the ith segment is obtained from its initial position and applying all the anterior transformations. For a hyperredundant robot the things are different. The arching angle is double the sum of each previous arching angle plus the current arching angle, because the un-arched segment is a prolongation of the previous segment.

The simulator was implemented in Microsoft Visual C++ .NET 2005 (Jones, W., 2004.), (Möller, T., 2002). It used the Microsoft DirectX SDK library for graphical purposes (Luna, F.D., 2003). In order to simulate the circular arched segments a series of intermediate points (that are connected by lines) between the segment origins must be determined. The Catmull-Rom interpolation algorithm (Watt, A., 2000) was used for this simulator because it was need an interpolation algorithm that passes through the control points. Catmull-Rom splines are a family of cubic interpolating splines formulated such that the tangent at each point p_i is calculated using the previous and next point on the splines, $\tau(p_{i+1} - p_{i-1})$.

Designing and implementing algorithms to also solve inverse kinematics problems, designing and implementing algorithms that consider the dynamic compenents in solving direct and inverse kinematics problems are other features of the Graphic Simulator.

3.2 Camera Calibration Algorithms

Camera calibration is the essential procedure for all such applications: positioning and orienting the cameras in order to support the accuracy of the image features extraction. Calibration for a pan/tilt/zoom camera shape is achieved by means of an engineered environment and a graphic simulation module.

Term “camera calibration” in the context of this paper refers to positioning and orienting the two cameras at imposed values. This calibration is performed only at the beginning, after that the cameras remain still. The general control method is an image based visual servoing one instead of position based. Camera calibration based on intrinsic parameters (classic sense, not the one used in this paper) is not necessary. Calibration operation assures that the two cameras’ axes are orthogonal.

Taking into account the presented structure of the tentacle - vision system, in order to apply the tested visual servoing algorithm, the two cameras must be positioned and oriented as:
- both focus on the robot.
- their axes are orthogonal.
- both have the same zoom factor.

Two different algorithms were implemented:
- one uses a cylindrical etalon.
- other uses the graphical simulator.

For the first algorithm, special starting conditions were imposed in order to support the image processing tasks: white background, dark grey cylinder, red vertical equidistant (90 degrees) axes, friendly initial camera's positions and orientations, zoom x1 (Fig. 6).

Three succesive and dependent calibrations are performed:
- Horizontal (pan): position and orientation are obtained in two successive, but dependent steps.
- Vertical (tilt): position and orientation are obtained in two successive, but dependent steps.

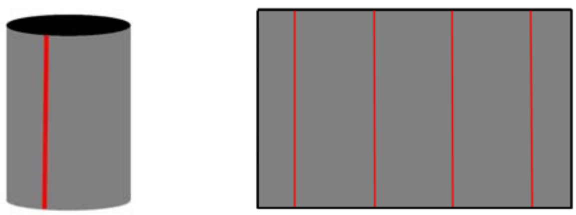


Figure 6 The cylindrical etalon

- Zoom: tuning the two cameras as both look to the cylinder from virtual equal distances.

Both offsets must be under the accepted thresholds. Else, the positioning destroyed the orientation and the procedure must be repeated. A similar algorithm is developed for the vertical orientation and positioning.

The second algorithm works together with the graphic simulator. It was proven that the two camera axes are orthogonal if, when both cameras are looking at the tentacle successively bended as circle's arcs in two orthogonal planes, are seeing also two circle's arcs (Fig. 7). The previous condition is fulfilled if each camera looks at the center of the circle containing the arc and the view line is orthogonal on the plane's circle.
Three calibration steps must be performed:
- Horizontal calibration - positioning and orienting the camera horizontally (pan),

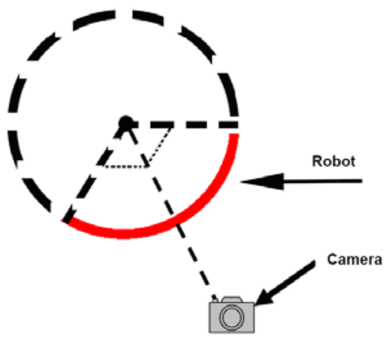


Figure 7 Camera looks to the center of the circle

- Vertical calibration - positioning and orienting the camera vertically (tilt),
- Zoom calibration - tuning the two cameras as both look at the robot from virtual equal distances.

How to “move” the camera in the steps of these algorithms? The image behavior in accordance with camera’s movements was studied. The effect of pan and tilt rotations on two points placed in a quadratic position on a circle was geometrically described. Matrix for coordinate transformations corresponding to rotations with pan and tilt angles, respectively for perspective transformation were used. The variation of the distance between the two

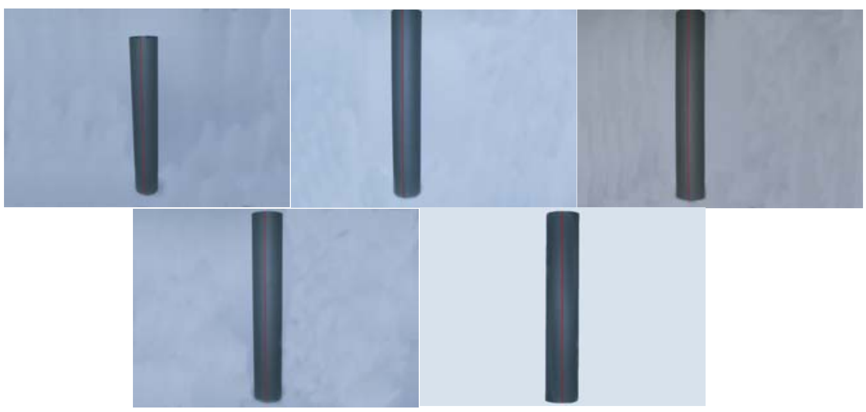


Figure 8 Calibration results

points, placed in a quadratic position on the circle, and the centre of the circle, depending of the tilt angle X, are plotted bellow in Fig. 9.

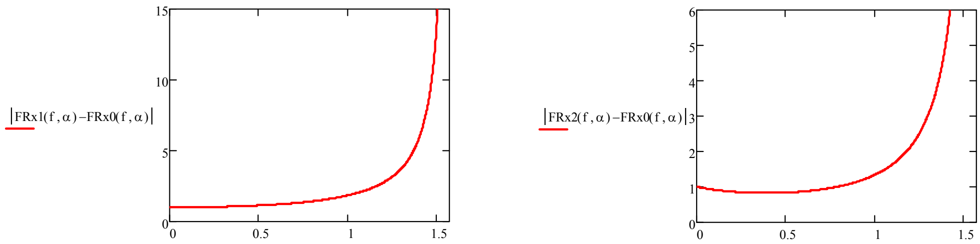


Figure 9 Distance variation for quadratic positions

The variation of the ratio of the two distances is plotted bellow in Fig. 10. The following plot

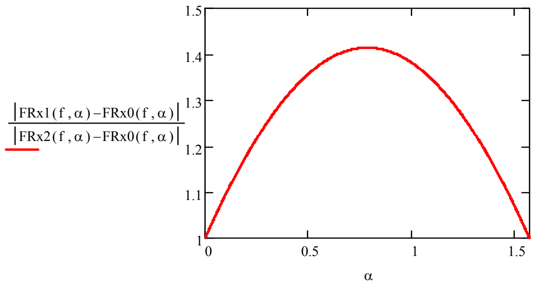


Figure 10 Ratio distances variation

in Fig. 11 shows how is transformed a rectangle (inscribed in the circle and having the edges parallel with the axes OX and OY) when a tilt rotation is performed. Theoretically, by zooming, the distance between the two points varies in a linear way, as it is shown upper right. The image's segmentation is basically a threshold procedure applied to the image's histogram. All the procedures included in the calibration algorithms were mathematically proven. If the calibration algorithm was successfully applied then the system is ready to perform the visualservoing tasks.

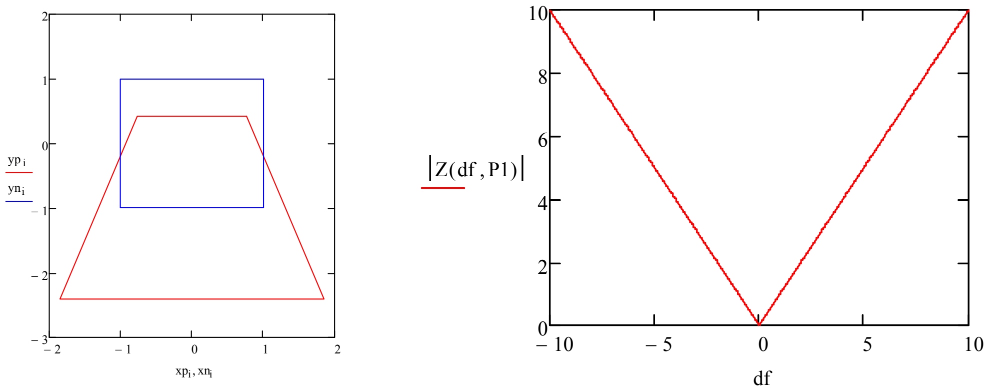


Figure 11 Rectangle transformation and distance variation under zoom influence

The image's segmentation is basically a threshold procedure applied to the image's histogram. All the procedures included in the calibration algorithms were mathematically proven. If the calibration algorithm was successfully applied then the system is ready to perform the visualservoing tasks.

5. Experimental results

In this experiment the robot will be controlled from the initial position (x_1, y_2, z_1) to the final position (x_2, y_2, z_3), passing through the intermediate position (x_3, y_3, z_3). All these positions are represented in image space. The coordinates are 3D because the position of each point was correlated from the images acquired by the two orthogonal cameras.

In order to ease the image processing, the tentacle was „dressed” in a white material, and the four reference discs were marked with black tape: robot base, the three discs that are directly controlled in order to arch and rotate the robot. The initial, intermediate and final position acquired images by both cameras, as well as the results obtained from segmenting these images and determining the relevant clusters are exemplified in the Fig. 12.

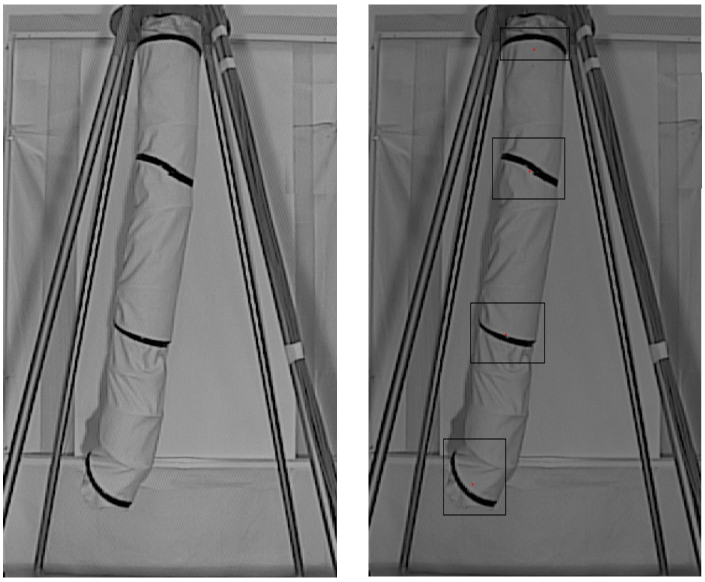


Figure 12 Images from cameras

Images obtained from the simulation environment are exemplified in Fig. 13. The three images illustrate the tentacular robot in the initial position (Fig. 13.a), the intermediate position (Fig. 13.b) and in the final position (Fig. 13.c).

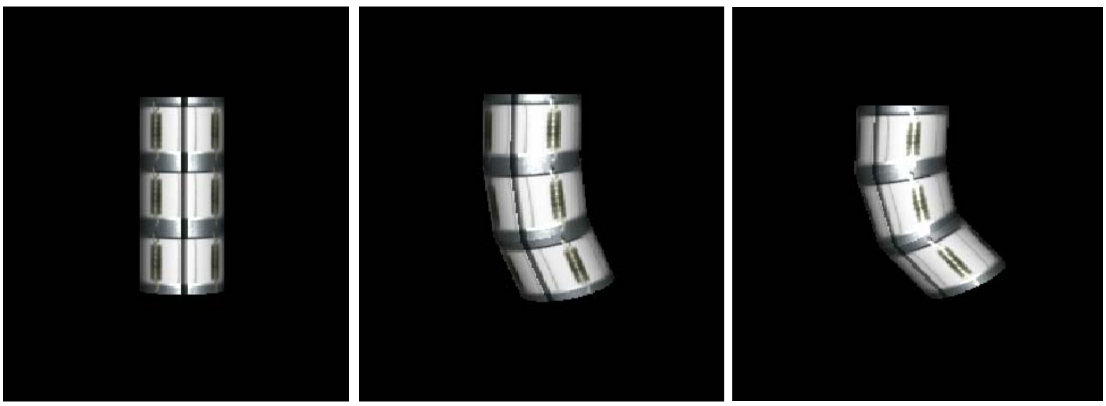


Figure 13 Images obtained from the simulation environment

The coordinates in image coordinate system for the four reference discs for each of the three time moments are presented in Table 1, while Table 2 shows the same coordinates in virtual control space. Distance to the target evolution in time for each reference disc is presented in Table 3.

Table 1. Reference discs centers coordinates computed in image coordinate system.

	Robot base (disc 1)			Disc 2			Disc 3			Disc 4		
	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
Initial position	212	212	41	212	212	164	212	212	334	212	212	502
Intermediate position	212	212	41	210	213	166	190	217	330	186	233	491
Final position	212	212	41	207	226	170	177	243	331	134	268	478

Table 2. Reference discs centers coordinates computed in virtual control space (compatible with graphic environment coordinates system) and transformed using the distance to the cameras.

	Robot base (disc 1)			Disc 2			Disc 3			Disc 4		
	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
Initial position	0	0	0	0	0	100	0	0	200	0	0	300
Intermediate position	0	0	0	-1.63	0.81	101.63	-15.02	3.41	197.27	-16.92	13.67	292.84
Final position	0	0	0	-4.07	11.38	104.88	-23.89	21.16	197.95	-50.76	36.44	284.38

Table 3. Distance to the target evolution in time for each reference disc.

	Disc 2	Disc 3	Disc 4
Initial position	12,98	31,58	64,60
Intermediate position	11,32	19,54	41,86
Final position	0,40	0,42	0,27

Position error evolutions for each disc center obtained from controlling the robot are presented in Fig. 14a, while position error evolutions for each disc center obtained from the simulator are shown in Fig. 14b. It can be observed that the convergence time to target is similar in the two cases. The differences in the two graphs (the control application graph has a more oscillator form) are caused by the mechanical imperfections of the control system. Fig. 15. presents angular speed variations for each segment obtained from controlling the robot - Fig. 15. a) illustrates the arching speeds, while Fig1 15. b) illustrates rotating speeds. Fig. 17. shows the same speeds obtained using data from the graphic simulator.

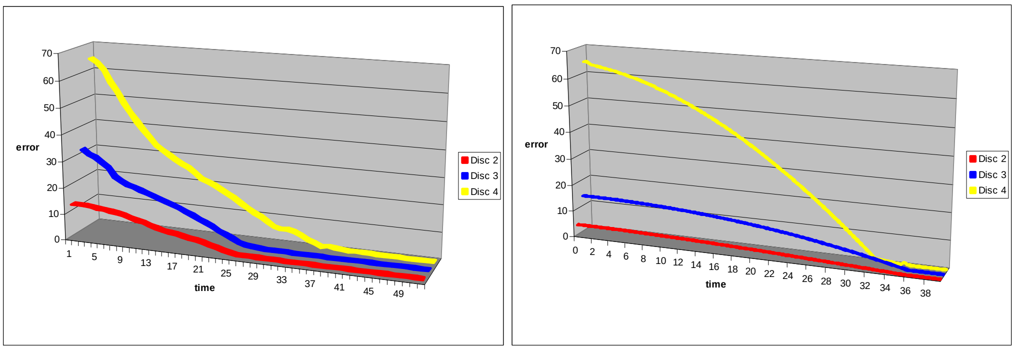
It can be observed that the speeds are relatively similar if the maximum/minimum values are considered, their time variation being different. This fact is caused by the operating mode selected for the real robot controllers, which is a linear speed variation. Also, like in the distance case, oscillations can be observed in the angular speed graphs. These oscillations are caused by the mechanical imperfections of the control system, but also by the vibrations that appear in the cables.

The experiments prove that a correct calibration was accomplished and accurate results were obtained. It order to eliminate the problems that appeared and also to obtain better results several improvements were considered, such as:

- Modify the mechanical structure of the robot by creating cylindrical segments that have smaller and smaller radius in order to ease the terminal segments maneuverability, thus diminishing the load on the motors that operate these terminal segments.

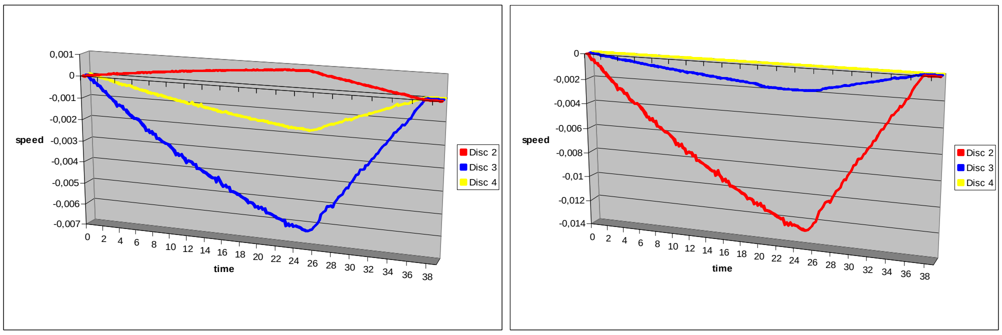
- Another alternative is to use segments shaped as conical frustums, a model that is more similar to the real elephant trunk;
- Enhance the stepper motors control algorithms in order to diminish vibrations and to increase movement precision. For this purpose, the images acquired by a high speed camera can be used. This will create the premises to enhance the control laws, and, as a consequence, to reduce the positioning errors, and finally, to eliminate the possibility to “jump” the target position.

The experimental results proved the validity and efficiency of the designed and implemented algorithms. The comparison between the results obtained from the graphic simulator and real robot control system validated both the built control system and the application implemented in the graphic simulation environment.



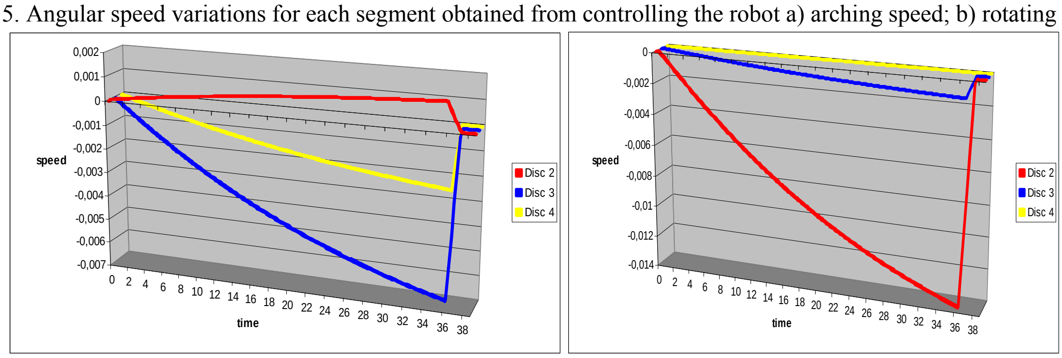
a) b)

Fig. 14. a) Position error evolutions for each disc center obtained from controlling the robot. b) Position error evolutions for each disc center obtained from the simulator.



a) b)

Fig. 15. Angular speed variations for each segment obtained from controlling the robot a) arching speed; b) rotating speed.



a) b)

Fig. 12. Angular speed variation for each segment obtained from the simulator a) arching speed; b) rotating speed.

6 Conclusions

This paper deals with a project for building and controling a hyperredundant robot which is currently developing. Different control methods and algorithms were proposed by the team members. A new tentacle manipulator using cables and stepper motors was designed and is under construction and a visual control using two video cameras is benchmarked. A graphic simulator was created in order to support the system implementation. In this paper, camera calibration using this graphic simulator was presented. The term “camera calibration” in the context of this paper refers to positioning and orienting the two cameras at imposed values.

Acknowledgement: The research presented in this paper was supported by the Romanian National University Research Council CNCSIS through the IDEI Research Grant ID93.

7 References

1. Cojocaru, D., Tanasie, T.R., Ivanescu, M., 2008, Graphic Simulation for Camera Calibration in Visual-Servoing Application, in NewAspects of Signal Processing, Computational Geometry and Artificial Vision, Proceedings of the International Conference on Signal Processing, Computational Geometry and Artificial Vision (ISCGAV'08), p.150-156, ISBN: 978-960-6766-95-4, ISSN: 1790-5109, August 20-22, Rhodes (Rodos) Island, Greece
2. Davies, J.B.C., Lane, D.M., a.o. 1998, Subsea applications of continuum robots, Underwater Technology, Proceedings of the 1998 International Symp. on Volume, Issue 15-17 Apr 1998, Page(s):363 – 369.
3. Grosso, E., Metta, G., a.o. 1996, Robust Visual Servoing in 3D Reaching Tasks, IEEE Transactions on Robotics and Automation, vol. 12, no. 15, pp. 732-742.
4. Hannan, M.W. and I. D. Walker, 2005, Real-time shape estimation for continuum robots using vision, Robotica, volume 23, pp. 645–651. © 2005 Cambridge University Press.
5. Hutchinson, S., Hager, G. D. and Corke, P. F. 1996, A Tutorial on Visual Servor Control, IEEE Transactions on Robotics and Automation, vol. 12, no. 15, pp. 651-670.
6. Ivanescu, M., 1984, Dynamic control for a tentacle manipulator, Proceedings International Conference on Robotics and Factory of the Future, Charlotte, pp. 315-327.
7. Ivanescu, M., 2002, Position Dynamic Control for a Tentacle Manipulator, Proceedings IEEE International Conference on Robotics and Automation, Washington, A1-15, 1531-1539.
8. Ivanescu, M., Cojocaru, D., a.o. 2007, Visual Servoing By Artificial Potential Methods Of A Hyperredundant Robot, Proceedings of 16th International Workshop on Robotics in Alpe-Adria-Danube Region - RAAD 2007, June 7-9, Ljubljana, Slovenia.
9. Jones, W., 2004, An Introduction to 3D Computer Graphics, Course Technology PTR.
10. Kelly, R. 1996, Robust Asymptotically State Visual Servoing, Proceedings IEEE Inernational Conference on Robotics and Automation, vol. 22, no. 15, pp. 759-765.
11. Luna, F.D., 2003, Introduction to 3D Game Programming with DirectX 9.0, Wordware Publishing, Inc., Plano – Texas.
12. Pressigout, M., E. Marchand, 2004, Model-free Augmented Reality by Virtual Visual Servoing, Proceedings of the 17th International Conference on Pattern Recognition (ICPR ‘04), pp. 1051-4651/04.
13. Möller, T. and E. Haines, 2002, Real-TimeRendering. 2nd ed. Natick, Mass.: A K Peters, Ltd.
14. Walker, I.D. and Carlos Carreras, 2006, Extension versus Bending for Continuum Robots, International Journal of Advanced Robotic Sys., Vol. 3, No.2 (2006), ISSN 1729-8806, pp. 171-178.
15. Walker, I.D., Darren M. Dawsona, a.o., 2005, Continuum Robot Arms Inspired by Cephalopods, DARPA Contr. N66001-C-8043, http://www.ces.clemson.edu/~ianw/spie05.pdf
16. Watt, A., 2000, 3D Computer Graphics, Addison Wesley.

Cooperation of human operator and small industrial robot

Borut Povše, Darko Koritnik
R&D department for automation, robotics and electronic instrumentation,
Dax Electronic systems Company, Vreskovo 68, Trbovlje, Slovenia
email: borut.dax@siol.net

Roman Kamnik, *Member*, *IEEE,* Tadej Bajd, *Fellow, IEEE,* Marko Munih, *Member, IEEE*
Laboratory of Robotics and Biomedical Engineering, Faculty of Electrical Engineering,
University of Ljubljana, Tržaška 25, Ljubljana, Slovenia

Abstract
Cooperation between a small industrial robot and human operator is studied in this paper. To ensure safe human-robot interaction several safety features should be introduced into the industrial cell. Despite all the precautions undertaken the collision between robot and man can occur. In present study impact assessments of point robot end-effector with passive mechanical arm were carried out. The impact energy density was calculated and used to evaluate possible injury levels caused by collisions and to determine a safe range of future investigations with human volunteers.

Keywords. Human- Robot cooperation, industrial robot, impact, collision between robot and man

1. Introduction

Future development of industrial production performance and new technologies require coexistence of humans and robotic systems. Future robots will not work behind safety guards with locked doors or light barriers. Instead they will be working in close cooperation with humans which leads to fundamental concern of how to ensure safe physical human robot interaction.

Different approaches have been proposed to study human robot interaction safety (Ikuta, K., 2003), (Heinzmann, J., 2003), (Lim, H., 2000). However, human-robot impacts via crash testing and resulting injuries were to our knowledge mainly investigated by Institute of Robotics and Mechatronics, DLR – German Aerospace Center. Their studies included use of different industrial robots such as Kuka KR3-SI (weight 54 kg), Kuka KR6 (weight 235 kg), Kuka KR500 (weight 2350 kg) and a LWRIII (weight 14kg) light weight robot. The experiments were focused on the chest and head impacts that can cause serious injury or even death. Estimation of injury was made using head injury criteria and compression or viscous criteria for the chest. Injury level was expressed using the abbreviated injury scale, classifying injury severity from 0 (none) to 6 (fatal). The results of the dummy crash-tests indicated that no robot, what ever mass it has, could be life-threatening at end-effector velocity 2 m/s prior to the impact when automobile industry criteria are used and clamping is excluded. Nevertheless, other less dangerous injuries such as fractures of facial and cranial bones can occur already at typical high robot velocities (Haddadin, S., 2007a), (Haddadin, S., 2008a).

When taking clamping of human body in consideration, both head and chest can be severely injured (Haddadin, S., 2008b).

Our research is focused on cooperation of a small industrial robot manipulator and a human worker. Complex assembly is an example of an industrial cell where robot and human can physically interact in order to make the assembly process more efficient and economical. Demanding operations (e.g. insertion of flexible parts) are performed by human worker, while precise assembly of rigid parts is accomplished by a robot.

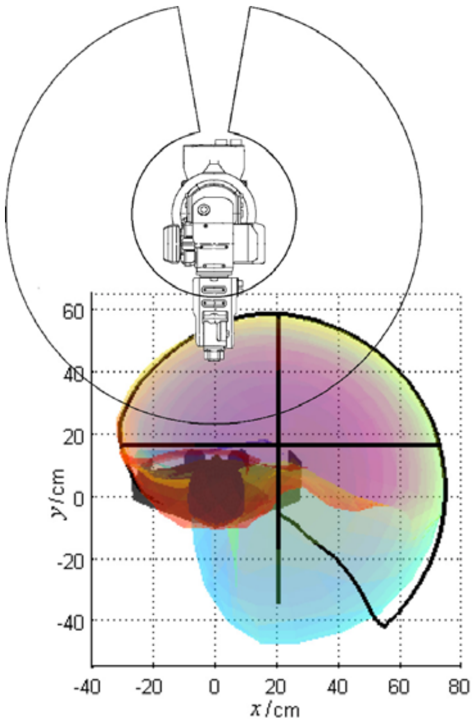


Fig. 1. Common human arm and robot workspace (side view)

We envisage an industrial cell with common human-robot workspace as shown in Fig. 1 (Klopčar, N., 2007). Collision is expected only between robot end-effector and lower arm of human operator. No life-threatening situations can occur; fractures of the lower arm bones are possible in the worst case scenario. The goal is to answer the question whether safe physical human robot interaction is possible when using a small standard industrial robot without human being injured if collision occurs. To ensure safe cooperation, secure end-effector trajectory planning, sensory system (mounted on robot and in the cell) and safety foam rubber clothing (on robot end-effector and human arm) will be introduced into the industrial cell. Nevertheless, the collision between man and robot can occur despite all the precautions undertaken. To study the effect of the impact between robot and a man, a passive mechanical lower arm (PMLA) was developed and equipped with inertial sensors. In the present study impact experiments were carried out with point shaped robot end-effector.

2. Methodology

2.1. Passive mechanical lower arm

In order to get preliminary test results, before starting an investigation with human subjects, a passive mechanical lower arm (PMLA) was built emulating relevant human arm characteristics. The device consists of a vertical base aluminum pillar to which the arm structure is attached (Fig. 2). The connection between the arm and the base is represented by a passively adjustable shoulder joint. Two smaller aluminum profiles, pneumatic cylinder, and pneumatic rotary unit represent the arm structure. The torque produced by the rotary unit compensates for the gravity, similarly to human biceps muscle and holds the lower arm in horizontal position. The viscoelastic human elbow properties are emulated using a pneumatic cylinder attached to the aluminum profiles representing lower and upper arm. The elbow joint characteristic properties (B– viscous damping, k– elasticity) were determined by adjusting the airflow valves connected to the cylinder.

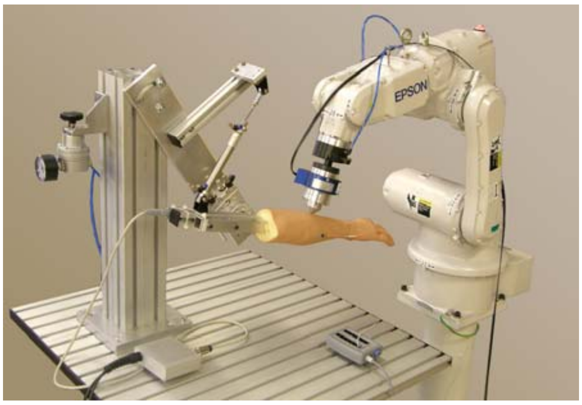


Fig. 2. PMLA and six axis robot with point end-effector

The viscoelastic elbow joint properties are not of significant importance in our current experiments and will be more precisely determined in future studies after completing the experiments with human subjects. The lower arm aluminum structure supports a foam rubber mock-up providing similar elasticity as relaxed muscle tissue. The foam rubber mock-up is covered with silicon esthetic glove resembling human skin. The mechanical lower arm is about the same weight as human lower arm.

2.2. Robot end- effector

Industrial robots are equipped with different grippers and end-effectors according to the task they are performing. In our experiments point shaped robot end-effector was used (Fig. 3) as it appears to be most dangerous in human-robot interactions. With this end-effector we were able to emulate human arm being hit by a conical robot tool.

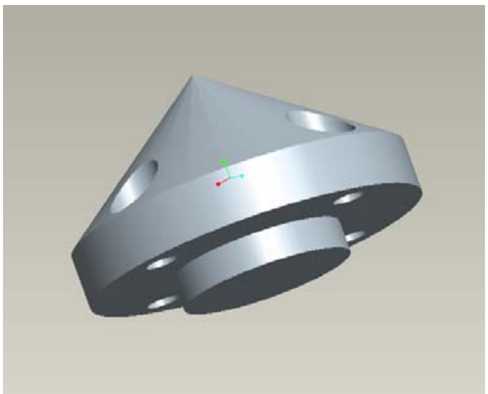


Fig. 3. Robot end-effector for point impact

2.3. Measuring system

The measuring system used in the investigation comprises the inertial sensors incorporating a set of two three-axis accelerometers ADXL203 and three gyroscopes ADXRS150 (Analog Devices, Inc.), three axis force sensor (JR3, Inc.), and the optical kinematic measurement system Optotrak Certus (Northern Digital, Inc.). The inertial sensors were mounted both to the robot end-effector and to the PMLA. The velocities and accelerations were measured at the PMLA supporting aluminum structure and recalculated to the PMLA impact point. The force sensor was installed between the robot's sixth joint and end-effector. The assessed accelerations, velocities, and forces were logged during human-robot impact by a real-time xPC target computer. In addition the robot end-effector and PMLA were equipped with infrared markers. The motion of the robot end-effector and PMLA during the impact was assessed by Optotrak system measuring the motion of infrared measurement markers attached to the objects.

2.4. Impact experiments

In our experiments the robot end-effector collided with the PMLA perpendicularly at constant deceleration. The point of impact was positioned eleven centimeters from the wrist on the dorsal aspect of the lower arm (Fig. 2). The robot end-effector was displaced toward the point of impact along a straight line. Several tests were carried out at different robot decelerations, maximal velocities, and different depths of stop points with regard to the arm surface. The robot end-effector deceleration was changed incrementally from 1000 mm/s^2 to 5000 mm/s^2. The end-effector stop point was located inside the PMLA. The depth from the lower arm surface was changed from 5 mm to 30 mm. After each robot impact, the PMLA was placed into the predefined starting position.

3. Results

3.1. Impact force, PMLA speed, and acceleration

In experiments the robot end-effector was displaced at maximum speed while robot deceleration (acceleration) and the end-effector stop point depth were changed respectively. For example, at robot deceleration set to 1000 mm/s^2 six experiments were carried out with robot end-effector stop point placed from 5 mm to 30 mm (by 5mm steps) inside the PMLA. For each robot deceleration increment of 1000 mm/s^2, all six experiments were repeated. Altogether thirty different experiments were performed. The impact force, PMLA speed, and PMLA acceleration were logged at 8 kHz by the real-time xPC target computer. The measuring results sampled at high frequency give us very good insight into robot end-effector and PMLA impact. The impact forces at maximum robot speed, constant deceleration, and various end-effector stop point depths are shown in Fig. 4. The highest impact force is induced about 30 ms after the start of impact and is highly dependent upon robot end-effector stop point depth. The PMLA speed of impact point is measured using the three-axis gyroscope. The maximal speed is reached about 40 ms after the start of impact and amounts to 0.33 m/s at 30 mm stop point depth (Fig. 5). The corresponding accelerations of the PMLA impact point are shown in Fig. 6.

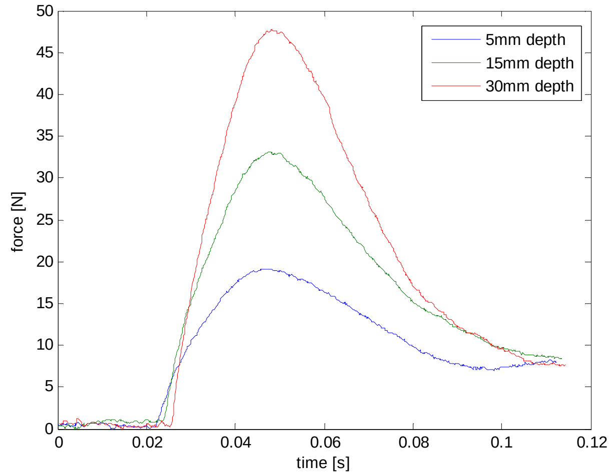


Fig. 4. The contact force during impact at different depths of the stop point

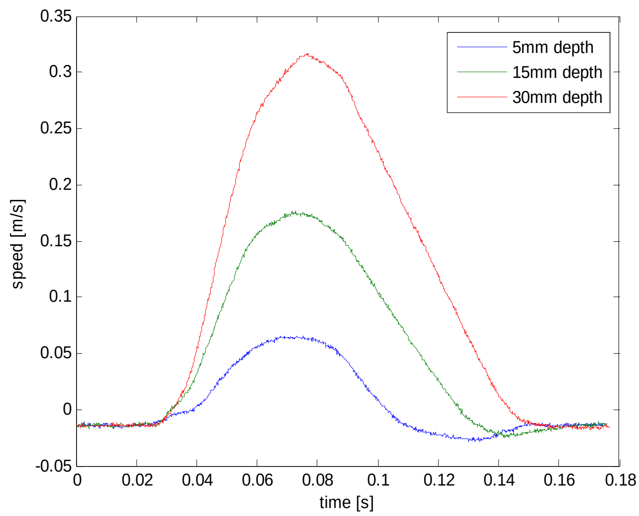


Fig. 5. PMLA speed of impact point

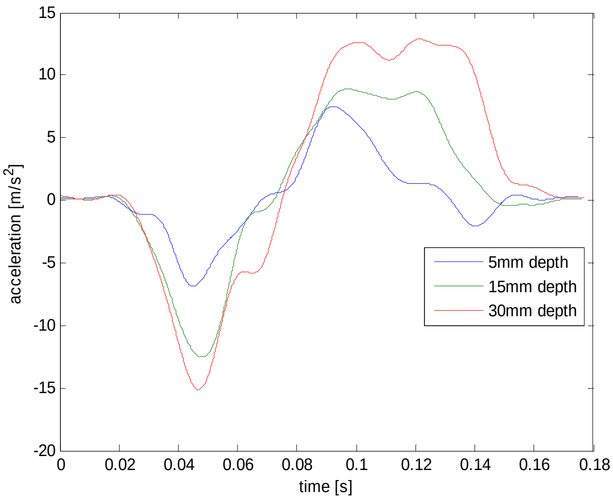


Fig. 6. PMLA acceleration of impact point

3.2. Impact energy density

Although the impact force, PMLA speed, and PMLA deceleration represent useful data for impact studies, they do not provide information regarding the degree of possible injury one would suffer from the impact. The injury intensities for the point impact can be evaluated by calculating the impact energy density. Higher impact energy density causes higher injury level (Haddadin, S., 2007b) and is calculated as

$$e_A = \frac{\int_{s_{impact\_start}}^{s_{impact\_stop}} F \cdot ds}{A_{end-effector}} . \tag{1}$$

In Eq. 1 F is the impact force applied to PMLA; s_{impact_start} and s_{impact_stop} are the distances between the robot end-effector and the center of the lower arm (supporting aluminum rod of the PMLA) at the start and at the end of impact, while $A_{end\text{-}effector}$ is the contact surface area between the robot end-effector and the PMLA. The effective contact surface area used in the calculations was measured using a stamp method. The end-effector was dipped into ink and pressed onto the PMLA. Afterwards, the imprint surface was measured. During impact the robot end-effector kinetic energy is transferred to the lower arm tissue (foam for the PMLA) only from the point where the end-effector touches the lower arm (s_{impact_start}) to the point where the distance between the end-effector and the lower arm center is the smallest (s_{impact_stop}). After that point the robot end-effecor kinetic energy is transferred to the PMLA kinetic energy since the robot end-effector starts to push the lower arm. The energy received by the tissue divided by the contact surface area is the energy density, Eq. 1. However, the energy density can only be used to evaluate contusions expressed by bruises and crushes (Haddadin, S., 2008c). Abrasions, laceration, and stab wounds have to be investigated using different criteria and are not likely to occur during impact with the end-effectors used in our investigation. The point end-effector has 0.04 cm^2 contact surface area and is not considered to be a sharp robot tool that can penetrate human skin at the forces and velocities evaluated in our experiments. In most cases robot tool sharp edges and corners can be avoided or covered with round shaped material when constructing the end-effector.

Tab. 1. The energy density [J/cm^2] during point impact

Stop point depth [mm] \ Robot acceleration [mm/s^2]	1000	2000	3000	4000	5000
5	0.18 J/cm^2	0.18 J/cm^2	0.27 J/cm^2	0.25 J/cm^2	0.47 J/cm^2
10	0.27 J/cm^2	0.42 J/cm^2	0.49 J/cm^2	0.72 J/cm^2	1.25 J/cm^2
15	0.57 J/cm^2	0.66 J/cm^2	1.42 J/cm^2	1.35 J/cm^2	1.63 J/cm^2
20	0.41 J/cm^2	0.93 J/cm^2	1.32 J/cm^2	1.84 J/cm^2	2.59 J/cm^2
25	0.56 J/cm^2	1.37 J/cm^2	1.69 J/cm^2	2.04 J/cm^2	3.05 J/cm^2
30	0.67 J/cm^2	1.60 J/cm^2	2.00 J/cm^2	3.25 J/cm^2	3.80 J/cm^2

Point robot end-effectors impact energy densities were calculated and are presented in Tab. 1. Maximal energy densities were reached at highest deceleration (5000 mm/s^2) and at the stop point depth of 30 mm.
In literature the tolerance values were published regarding energy density of the impact and corresponding injury. Tissue injuries occur at the impact energy density higher than 2.52 J/cm^2, while hematoma or suffusion already occur below this value (Haddadin S., 2007b). Point impact energy densities that surpass the safe energy density limit are painted red in Tab. 1. The experimental results with our PMLA reveal that point impact can beside suffusion and hematoma cause serious tissue injury.

4. Conclusion

We have presented a human-robot impact emulation system and the preliminary tests conducted with the PMLA. Experiments were performed using point robot end-effector tool. In the experiments the robot end-effector collided with the PMLA perpendicularly at constant decelerations. The injury probability was evaluated by calculating the impact energy density.

In future, investigation with human volunteers will be carried out applying only safe robot impacts in order to verify the properties of our PMLA model.

5. References

Haddadin S., Albu-Schäffer A., Hirzinger G. 2007, Safety Evaluation of Physical Human-Robot Interaction via Crash-Testing, *Robotics: Science and Systems Conference (RSS2007)*, Atlanta, USA.

Haddadin S., Albu-Schäffer A., Hirzinger G. 2007, Safe Physical Human-Robot Interaction: Measurements, Analysis & New Insights, in *International Symposium on Robotics Research (ISRR2007), Hiroshima, Japan*.

Haddadin S., Albu-Schäffer A., Hirzinger G. 2008, The Role of the Robot Mass and Velocity in Physical Human-Robot Interaction – Part I: Unconstrained Blunt Impacts, *IEEE International Conference on Robotics and Automation (ICRA2008), pp. 1331-1338,* Pasadena, USA.

Haddadin S., Albu-Schäffer A., Hirzinger G. 2008, The Role of the Robot Mass and Velocity in Physical Human-Robot Interaction - Part II: Constrained Blunt Impacts, *IEEE International Conference on Robotics and Automation (ICRA2008), pp. 1339-1345,* Pasadena, USA.

Haddadin S., Albu-Schäffer A., Hirzinger G. 2008, Evaluation of Collision Detection and Reaction for a Human-Friendly Robot on Biological Tissues, *IARP International Workshop on Technical challenges and for dependable robots in Human environments*, Pasadena, USA.

Heinzmann J. and Zelinsky A. 2003, Quantitative Safety Guarantees for Physical Human-Robot Interaction, *Int. J. of Robotics Research*,vol. 22, no. 7-8, pp. 479–504.

Ikuta K., Ishii H., and Nokata M. 2003, Safety Evaluation Method of Design and Control for Human-Care Robots, *Int. J. of Robotics Research*, vol. 22, no. 5, pp. 281–298.

Klopčar N., Tomšič M., Lenarčič J. 2007, A Kinematic Model of Shoulder Complex to Evaluate the Arm-Reachable Workspace, *Journal of Biomechanics*, vol. 40, pp. 86-91.

Lim H.-O. and Tanie K. 2000, Human Safety Mechanisms of Human-Friendly Robots: Passive Viscoelastic Trunk and Passively Movable Base, *Int. J. of Robotics Research*, vol. 19, no. 4, pp. 307–335.

Acknowledgements This work was supported in part by the European Union. "Operation part financed by the European Union, European Social Fund."

1. Österreichisch-Kubanischer Automatisierungstag „Automation and Mechatronics“

Centro Universidad Jose Antonio Echeverria - CUJAE,
Havanna, Kuba
16. November 2009

Der erste österreisch-kubanische Automatisierungstag wurde von der Abteilung Intelligente Handhabungs- und Robotertechnik (IHRT) der TU Wien in Zusammenarbeit mit der TU La Habana „Centro Universidad Jose Antonio Echeverria – CUJAE”, im Rahmen des Kooperationsabkommens beider Universitäten, organisiert. Zweck dieses ersten bilateralen Automatisierungstages war es, österreichische Leistungen auf dem Gebiet der Automatisierungstechnik im weitesten Sinne einem kubanischen Fachpublikum zu präsentieren und umgekehrt.

Die Eröffnung vor zirka 100 fachkundigen Teilnehmern wurde von österreichischer Seite von P. Kopacek und vom Vizerektor für Internationale Beziehungen der CUJAE Llanos Orestes Santiago vorgenommen.

In seinem Eröffnungsreferat „ Mechatronics and Robotics in Austria- State of the Art and further Development Trends“ ging P. Kopacek zunächst auf die Mechatronik in Österreich ein und illustrierte an Hand von ausgeführten industriellen Anwendungsbeispielen diesen Überblick. Danach versuchte er den derzeitigen Stand und Entwicklungstendenzen sowohl der Prozess- als auch der Fertigungsautomatisierung in Österreich zu skizzieren.
Basierend auf einer vom BMVIT in Auftrag gegebenen Studie „ Robotics in Austria“ erläuterte er sodann den derzeitigen Stand der Robotertechnik in Österreich. Ende 2006 waren in Österreich 4382 Industrieroboter im Einsatz, davon 2981 für Handhabungsaufgaben und Maschinenbeschickung, 630 für Schweißen und Löten, 225 in Produktionsprozessen, 128 für Montage und Demontage sowie 418 für verschiedene Anwendungen.

Von 18 Universitäts- und Fachhochschulinstituten arbeiten 13 an mobilen Robotern, 11 in Robotersoftware und 4 entwickeln menschenähnliche (humanoide) Roboter. 7 Forschungsinstitute sind auf den Gebieten Industrieroboter, mobile Roboter und Robotersoftware tätig. 57 Firmen – überwiegend Systemintegratoren - arbeiten mit Industrierobotern oder deren Komponenten, einige auch an mobilen Robotern.

Roboterfußball begann Mitte der 90er Jahre als Spiel, um junge Menschen für Robotik und Informatik zu begeistern. Seither hat es sich zu einem eigenen Wissenschaftsgebiet entwickelt. In seinem Referat „ Robotsoccer – from Nonsense to Sense“ skizzierte N. Jesse die wissenschaftlichen Herausforderungen von Roboterfußball. Er verwies hier etwa die ultraschnelle Bilderkennung und –auswertung, die Bahnplanung und -verfolgung und die Notwendigkeit eines kooperativen Handelns der Roboter. Der Referent erläuterte zunächst die verschiedenen Ligen mit ihren speziellen Regeln und Settings. Am Beispiel der MiroSot Liga veranschaulichte er sodann zwei zentrale Aspekte: Es wurde deutlich, dass in der Bildverarbeitung außergewöhnlich anspruchsvolle Probleme zu lösen sind. Im Hinblick auf die spielerische „Intelligenz“ betonte N. Jesse, dass hier noch viele Fragen ungelöst sind. Experimente mit Neuronalen Netzen, Fuzzy Logik und Evolutionären Algorithmen lieferten interessante, gleichwohl nur erste rudimentäre Resultate. Im zweiten Teil seines Vortrages widmete sich der Referent der Frage des wissenschaftlichen und praktischen Nutzens von Roboterfußball. Er ging hier ein auf drei Aspekte: den Nutzen in der universitären Ausbildung, die Bedeutung für die Entwicklung von Dienstleistungsrobotern und die Relevanz für das wissenschaftliche Arbeitsgebiet „autonome Agenten“. Der Referent gelangte zu dem Resultat, dass Roboterfußball durchaus ein interessantes Testfeld für die Entwicklung von autonomen Dienstleistungsrobotern (Agenten) darstellt, den zentralen Nutzen sieht er jedoch in dem Nutzen für die Ausbildung von Studierenden: Roboterfußball ist interdisziplinär, aufgrund der internationalen Wettbewerbe spannend, motivierend und unterhaltsam.

G. Kronreif (PROFACTOR GmbH) ging, als Ergänzung zum Einleitungsreferat, in seinem Beitrag „ Service Robot Applications in Austria“ speziell auf dieses Teilgebiet der Robotertechnik ein. Einleitend wurden die verschiedenen Anwendungsgebiete vorgestellt, in welchen Robotertechnik oder Teilbereiche daraus zur Unterstützung und Prozessverbesserung eingesetzt werden können. Beispiele dafür sind Roboteranwendungen im privaten Umfeld – wie für Reinigungsaufgaben, Freizeit und Hobby u.a. – oder für öffentliche Aufgaben – wie in der Gastronomie, im Bereich Verteidigung und Sicherheit, für kommunale Aufgaben u.v.m.

Im Hauptteil des Vortrags wurden dann Beispiele für die erfolgreiche Positionierung österreichischer Firmen und Forschungseinrichtungen in ausgewählten Anwendungen präsentiert. Der Bogen vorgestellter Systeme reichte dabei vom multi-modalen Containerumschlag über Spiel und Freizeit, bis hin zu Anwendungen in der Raumfahrt. Ein Schwerpunkt dabei war in den Bereichen Medizin, Rehabilitation und Pflege gegeben. Vorgestellt wurde hier zum Beispiel die österreichische Entwicklung „IntraROB" – eine modulare Roboterplattform für eine Vielzahl minimal-invasiver Eingriffe, wie zum Beispiel für die transrektale Ultraschall-gestützte Brachytherapie für Prostatakarzinome, die perkutane Thermoablation, die Biopsieprobenentnahme im Abdominalbereich oder für endoskopische Eingriffe in der Nasennebenhöhle. Weiters vorgestellt wurden österreichische Entwicklungen zur Unterstützung der Bewegungstherapie bei Schlaganfall-Patienten sowie Robotersysteme für Spiel-Unterstützung für behinderte Kinder. Abschließend wurde festgehalten, dass die österreichischen Arbeiten im Bereich Service-Robotertechnik auch im internationalen Umfeld durchaus erfolgreich und angesehen sind, und dass die Arbeit in diesem Anwendungsgebiet der Robotertechnik sehr stark von der Zusammenarbeit in inter-disziplinären Teams abhängig ist.

Über „Cost Oriented – COA – Robots" berichtete anschließend P. Kopacek. Dieses auf „Low Cost Automation – LCA" zurückgehende Thema gewinnt nun zunehmend auch für die Robotertechnik an Bedeutung. Insbesonders Klein- und Mittelbetriebe – KMU`s - haben im Allgemeinen nur beschränkte Investitionsmöglichkeiten. Es sind daher derzeit auf internationaler Ebene Forschungsprojekte zu diesem Thema in Arbeit. An Hand von am IHRT der TU Wien realisierten Beispielen zeigte der Vortragende preiswerte – „cost oriented"- Lösungen auf. Es wurde ein sechs rädriger mobiler Roboter, welcher unter anderem zur Landminenortung eingesetzt werden kann, vorgestellt. Das Hauptaugenmerk lag dabei auf Bewegungsmöglichkeiten in extrem unebenen Terrain. Im Anschluss daran wurden ein mobiler Miniroboter für Weltraumanwendungen sowie ein „ Teen –Sized" humanoider Roboter präsentiert. Weiters wurden zwei Anwendungsbeispiele in KMU`s dargestellt.

Zum Abschluss stellte M.O.Torres Pineiro (Automatic and Computer Department, CUJAE) einen an der Abteilung entwickelten Roboter vor. Dieser extreme COA Roboter ist überwiegend aus Bauteilen und Baugruppen von PC`s beziehungsweise PS-Zubehör zusammengebaut und mit einer fortschrittlichen Software ausgestattet. So wurden beispielsweise die Roboter-Antriebe aus Drucker-Komponenten gestaltet. Er kann als erster Prototyp für zukünftige, einfache, industrielle Anwendungen angesehen werden.

In der Zusammenfassung dieses ersten „Kubanisch-Österreichischen Automatisierungstages" wurden ausführlich weitere Kooperationsmöglichkeiten diskutiert und übereinstimmend beschlossen diese Aktivitäten zu intensivieren.

N.Jesse, P.Kopacek, G.Kronreif

1. Österreichisch-Kubanischer Informationstechniktag „IT in Automation“

Universidad de las Ciencias Informàticas, UCI

Havanna, Kuba

17. November 2009

Der erste österreichisch-kubanische Informationstechnik–Tag wurde von der Abteilung Intelligente Handhabungs- und Robotertechnik (IHRT) der TU Wien in Zusammenarbeit mit der „Universidad de las Ciencias Informàticas, UCI”, organisiert. Die UCI ist ein universitäres Studienzentrum im Municipio von Boyeros und hat zwei wesentliche Ziele: das Land mit Computertechnik auszustatten sowie Software zu exportieren. Die Universität wurde 2002 mit dem ersten Studienjahrgang und 2008 Studenten sowie 300 Hochschullehrern eröffnet - im Jahr 2007 verließ der erste Abschlussjahrgang mit insgesamt 1334 Absolventen als Ingenieure der Informatik die Universität. Derzeit zählt die UCI rund 10.000 Studenten. Sie sollen als primäres Ziel die Computerisierung des Landes vorantreiben. Mehr als 60% der Studenten sind in Produktiv- und Forschungsprojekte für Software für die kubanische Gesellschaft und andere Staaten, wie zum Beispiel für Bildung, Gesundheit, Sport, freie Software, Websites und -portale, Multimedia usw. tätig.

Zweck dieses ersten bilateralen IT-Tages war es, österreichische Leistungen auf dem Gebiet der Informationstechnik im weitesten Sinne einem kubanischen Fachpublikum zu präsentieren und umgekehrt. Die Eröffnung vor zirka 140 fachkundigen Teilnehmern wurde von österreichischer Seite vom Präsidenten der Österreichisch–Kubanischen Freundschaftsgesellschaft H. Mikosch sowie von P. Kopacek (IHRT) vorgenommen. Seitens der UCI wurden die Teilnehmer vom Vizerektor für Internationale Beziehungen T. Lopez Jimenez begrüßt.

In seinem Einleitungsreferat „Directions of Research and Technology Development for Automation in Cuba“ berichtete Manuel Lazo. Koordinator des “Regional Program of Automatic of MIC (Ministry Of ICT in Cuba)”.über Schwerpunkte der kubanischen

Regierung auf diesem Gebiet. Diese sind im Wesentlichen von den Bedürfnissen der kubanischen Industrie bestimmt und konzentrieren sich auf Telekommunikation und Prozessautomatisierung.

P. Kopacek versuchte sodann in seinem Vortrag „IT for Automation in Austria – State of the Art and Future Development Trends“ einen Überblick über die österreichische Szene zu geben. Er konnte in der kurzen dafür zur Verfügung stehenden Zeit nur auf einige Anwendungsbeispiele von Fuzzy, Neuro und Neuro-Fuzzy Methoden in der Prozessautomatisierung eingehen. Auf dem Gebiet der Fertigungsautomatisierung lag der Schwerpunkt seiner Präsentation auf dem Gebiet der Produktionsautomatisierung sowie der Robotik. Anwendungsbeispiele waren die Automatisierung einer Rohrwalzstraße für geschlossene Blechprofile, Montagezellen für Schweißtransformatoren sowie eine Demontagezelle für Minidisketten. Weiters wurden die Entwicklung von mobilen Minirobotern und ihren Anwendungen im Rahmen des ESA Projektes „Roby Space“ – ein Miniroboter für Weltraum Anwendungen – und der humanoide Roboter „Archie“ vorgestellt.

In seinem zweiten Vortrag „Robotics in Austria and the Strategic Research Agenda of Europe“ präsentierte P. Kopacek zunächst die Ergebnisse der gleichnamigen vom BMVIT in Auftrag gegebenen Studie und ging sodann auf die Neufassung der „Strategic Research Agenda – SRA“ ein welche im Juli 2009 offiziell in Brüssel vorgestellt wurde. Demnach sollten ab dem Jahre 2020 autonome oder mit dem Menschen kooperierende Arbeitsroboter, mobile, intelligente Roboter für Logistikaufgaben (z.B. Personentransport), Roboter zur Überwachung und Intervention sowie Roboter zur Inspektion und für die Freizeit verfügbar sein.

Bildverarbeitung spielt in der Automatisierungstechnik eine essentielle Rolle. In seinem Vortrag zeigte N. Jesse als Vize-Präsident der „Federation of International Robot Soccer Associations“ (FIRA) die vielfältigen Herausforderungen der Bildverarbeitung im Roboterfußball auf. Sein Vortrag hatte den Titel „Image Processing for MiroSot-Robots – Harvesting from Experience“. Im Anschluss an einige einführende Bemerkungen zu den verschiedenen Roboterfußball-Wettbewerben wurden am Beispiel der MiroSot-Challenge dargestellt, dass eine flexible, robuste und effiziente Bildverarbeitung in diesem überaus schnellen Spiel innovative Konzeptionen und Algorithmen voraussetzt. Ein weiteres

Augemerk galt auch Ergebnissen einer Studie, in der Roboterfußballer mit einer lokalen Bildverarbeitung ausgestattet wurden. Den Abschluss des Vortrags bildete eine kurze Diskussion der Frage, inwieweit sich diese low-cost-Erfahrungen auf industrielle Fragestellungen übertragen lassen.

Mit seinem Beitrag "IT and Robotics for disabled Kids" stellte G. Kronreif (PROFACTOR GmbH) ein neues Anwendungsgebiet der IT und Robotertechnik vor. Die grundlegende Fragestellung für dieses Forschungsgebiet ist es, Möglichkeiten für ein eigenständiges Spielen auch Kindern mit Behinderungen zur Verfügung zu stellen. Damit sollen dieser Anwendergruppe dieselben Möglichkeiten zum „spielerischen Lernen" zugänglich gemacht werden, wie es für nicht-behinderte Kinder der Fall - und alltäglich - ist. Das erwartete Ziel solcher Arbeiten ist es, durch diesen Zugang zum Spiel bekannte Probleme wie die soziale Isolation behinderter Kinder oder die teilweise extrem hohe Abhängigkeit zu reduzieren und letztlich eine spielerische Unterstützung im Lernprozess darzustellen. Nach einer kurzen Vorstellung internationaler Aktivitäten in diesem Themenkreis – vor allem zur Unterstützung autistischer Kinder – wurden zwei Projekte im Detail vorgestellt. Das Robotersystem PlayROB ist ein Assistenzsystem, dass schwer körperbehinderten Kindern das autonome Spielen mit LEGO™-Bausteinen ermöglicht. Durch klassische Eingabegeräte – wie Kopftaster, 5-fach Tastatur, Joystick oder Saug-Blas-Eingabegerät gesteuert, positioniert der Roboter interaktiv die Bausteine am gewünschten Platz und bietet den Kindern damit ein völlig neues Spielgefühl. Es wurden in den letzten 4 Jahren sechs solcher Robotersysteme an österreichischen Schulen installiert – diese erfreuen sich immer noch sehr großer Beliebtheit bei den Kindern. Das zweite vorgestellte Projekt – IROMEC – ist ein durch das IST-Programm der EU geförderte Forschungsprojekt zur Entwicklung eines interaktiven, mobilen Robotersystems für Kinder mit physischer oder mentaler Behinderung sowie für Kinder mit Autismus. Auch dieses Robotersystem soll Spielabläufe unterstützen und die Kinder zu Kommunikation und Kooperation anregen. Sieben Prototypen des IROMEC-Robotersystem befinden sich derzeit in verschiedenen Institutionen in ganz Europa in einer Intensiv-Testphase; erste Ergebnisse sind sehr vielversprechend.

Von kubanischer Seite wurden folgende Beiträge präsentiert:

SCADA Guardián del ALBA.

Espinosa A., Chávez A., Pérez M., Universidad de las Ciencias Informáticas-UCI

Multicell WiMax Data Interchange System Design based on OFDM Modality.

Carrera E., Universidad de las Ciencias Informáticas-UCI.

Generic Interface for IT Systems Device Interactions.

García L. E., Universidad de las Ciencias Informáticas-UCI, Instituto Minero Metalúrgico de Moa-ISMMM.

Computational simulation for industrial processes. Model and algorithms of natural gas compression for oil extraction.

González C., Universidad de las Ciencias Informáticas-UCI, Instituto de de Ciencias e Investigación en Matemática y Física-ICIMAF.

Logic Programmable Controller based on Reconfigurable Hardware.

Ramírez M., Universidad de las Ciencias Informáticas-UCI, Instituto Superior Politécnico José Antonio Echevarría-ISPJAE

Die präsentierten Projekte zeigten einen wesentlichen Arbeitsbereich der UCI, nämlich die Entwicklung von Soft- und Hardware-Komponenten zur Prozesssteuerung und Überwachung. Ein sehr bedeutendes Anwendungsgebiet für die entwickelten Komponenten ist in der Rohstoff-Industrie, wie zum Beispiel in der Erdöl/Erdgas- und Nickel-Förderung und –Verarbeitung.

Den Abschluss bildete eine Podiumsdiskussion welche in Form von Präsentationen von Firmen und Forschungseinrichtungen abgehalten wurde.

Von kubanischer Seite waren dies:

Mulet, C.: “State of IT works in der “Company of Service, Communication and Electronics of Nickel Industry - SERCONI“

Reyes, G.: “State of IT works in Company of Complete Automation - CEDAI”

Gonzales, C.: “State of IT works in Institute of Cybernetics, Mathematics and Physics - ICIMAF”

Von österreichischer Seite präsentierten:

Kronreif, G.: “Research and Implementation of Automation Projects at Profactor-GmbH”

Jesse, N.: “Software for Automation at Quinscape”

Kopacek, P.: Research in Automation at the “Austrian Society for Systems Engineering and Automation – SAT”

Rommens, E. and P.Kopacek: “The International Federation of Automatic Control – IFAC as a nucleus for Advanced Automation”

G. Kronreif berichtete über ausgewählte Projekte der PROFACTOR GmbH im Bereich Bildverarbeitung und Automatisierungstechnik. Präsentiert wurden ein Sensorsystem zur 360°-Oberflächeninspektion, ein speziell entwickelter Sensor zur Qualitätskontrolle in der CFK-Fertigung, Arbeiten im Themenfeld Thermographie zur „on-line“ Qualitätskontrolle, sowie ein Sensorsystem zur Schweißnahtkontrolle. Als Beispiel für Automatisierungs-Lösungen im Themenbereich „Losgröße 1“ wurde eine entwickelte Anlage zur automatisierten Zylinderschloß-Montage vorgestellt. Den Abschluss des Vortrags bildeten zwei Projekte im Bereich Bildverarbeitung und Robotertechnik – das Softwaresystem „iRobFeeder“ als industrie-taugliche Lösung des Problems „Griff in die Kiste“ sowie „CompleteInspect“ als einfache aber leistungsstarke Lösung zur optischen Qualitätskontrolle.

In seinem Vortrag skizzierte N. Jesse die hohe Dynamik der Automatisierung in administrativen Kontexten. Er erläuterte die verschienen Geschäftsfelder der von ihm mitbegründeten QuinScape GmbH. Anhand von Beispielen betonte er die Bedeutung von sogenannten. Portalen, in denen die verschiedenen Datenbanken von Unternehmen flexibel miteinander „verbunden“ und Workflows abgebildet werden können. In diesem Zusammenhang berichtete er auch über ein von ihm geleitetes Projekt zu Einführung von elektronischen Workflows in drei mazedonischen Universitäten. Weiter führte er unter Bezug auf die von QuinScape entwickelte Software DocScape aus, dass es künftig auch möglich ist, die heute noch unzureichend IT-unterstützen Prozesse zu automatisieren. In seinem Schlusswort gelangte er zu dem Ergebnis, dass zwar die Randbedingungen und Algorithmen in der Produktionstechnologie sich erheblich unterscheiden, letztendlich aber auch in den Verwaltungsprozessen von Institutionen ein massiver Trend hin zu einer Vollautomatisierung besteht.

P. Kopacek berichtete in Vertretung von B. Kopacek (Österreichische Gesellschaft für System und Automatisierungstechnik – SAT) über neuere Entwicklungen auf dem Gebiet der „Semiautomatisierten Demontage von Elektro- und Elektronikgeräten“. Im zweiten Teil des Vortrags berichtete er über das Thema „End of Life (EoL) Management of Automation Devices“ welches durch die von der EU 2005 verabschiedete WEEE (Waste from Electric and Electronic Equipment) Direktive zunehmend an Bedeutung gewinnt.
Im letzten österreichischen Vortrag wurde ein Überblick über die Aktivitäten der „International Federation of Automatic Control – IFAC“ und insbesonders des „IFAC Beirates Österreich - Austrian Supervisory Board of IFAC” gegeben.

In ihren Zusammenfassungen stellten die Organisatoren P. Kopacek und Juan Antonio Fung. (Head of the Automatic Center at School 5 of UCI) übereinstimmend fest, dass im kubanischen Markt ein großer Bedarf an speziellen IT Lösungen für Automatisierungsaufgaben besteht, welcher durch österreichische Firmen und Forschungseinrichtungen, in Kooperation mit kubanischen Partnern, abgedeckt werden könnte.
Dieser erste österreichisch-kubanische IT-Tag wurde von der „Universidad de las Ciencias Informàticas, UCI” in hervorragendster Weise organisiert. Auf Grund des großen Erfolges wurde beschlossen diese Veranstaltung im Jahr 2011 zu wiederholen.

N.Jesse, P. Kopacek, G. Kronreif

Instruction to authors – presented as a pattern paper (18 pt)

A. Maier, F. Huber (12 pt)
Department ....., Vienna, Austria

Received April 8, 1999

Abstract
This paper shows …… (italics, 12 pt)

1 General (14 pt)

Authors should prepare their manuscript camera ready, format A 4, 12 typeface and must present their manuscript in good quality. At the left/right edge 2.5 cm, at the to/bottom edge 3 cm. Authors are invited to use papers of this journal as a sample. Please do not use an eraser or erasing fluid. Footnotes should be avoided if possible. Authors are expected to submit their paper to one of the publishers (O.Univ.Prof.Dr. Peter Kopacek, Intelligent Handling Devices and Robotics, Vienna University of Technology, Favoritenstrasse 9-11, A-1040 Vienna, Austria, Fax: +43 1 58801-31899 or O.Univ.Prof.Dr. Alexander Weinmann, Institute of Automation and Control, Vienna University of Technology, Gusshausstr. 27-29, A-1040 Vienna, Austria, Fax: +43 1 58801 37699).

Email address for submitting pdf-manuscripts: weinmann@acin.tuwien.ac.at

2 References (14 pt)

Within the paper references should appear in the following form:
(Mayer, H., 1990) or (*Mayer, H., 1990)* (12 pt);
Mayer, H., 1990, discovered that....

3 Figures and Tables (14 pt)

Figures and Tables should be integrated in the paper and have to be referred to as Fig. 4.1 or Tab. 5.2.

4 References

References are to be listed alphabetically according to first author. (11 pt)

5 Word Processing System/Editor

Microsoft Word for Windows 1.1 or higher; Microsoft Word 5.0; TeX or LaTeX.

Wenn unzustellbar, retour an:

IFAC-Beirat Österreich (E318 / E376)
Favoritenstraße 9-11, A-1040 Wien /
Gußhausstraße 27-29, A-1040 Wien