Tensor Product Model Transformation-based Controller Design for Gantry Crane Control System – An Application Approach

Fetah Kolonic, Alen Poljugan, Ivan Petrovic

Faculty of Electrical Engineering and Computing, University of Zagreb
Unska 3, HR-10000 Zagreb, Croatia
fetah.kolonic@fer.hr, alen.poljugan@fer.hr, ivan.petrovic@fer.hr

Abstract: The Tensor Product (TP) model transformation is a recently proposed technique for transforming given Linear Parameter Varying (LPV) state-space models into polytopic model form, namely, to parameter varying convex combination of Linear Time Invariant (LTI) systems. The main advantage of the TP model transformation is that it is executable in a few minutes and the Linear Matrix Inequality (LMI)-based control design frameworks can immediately be applied to the resulting polytopc models to yield controllers with tractable and guaranteed performance. Various applications of the TP model transformation-based design were studied via academic complex and benchmark problems, but no real experimental environment-based study was published. Thus, the main objective of this paper is to study how the TP model transformation performs in a real world problem and control setup. The laboratory concept for TP model-based controller design, simulation and real time running on an electromechanical system is presented. Development system for TP model-based controller with one hardware/software platform and target system with real-time hardware/ software support are connected in the unique system. Proposed system is based on microprocessor of personal computer (PC) for simulation and software development as well as for real-time control. Control algorithm, designed and simulated in MATLAB/SIMULINK environment, use graphically oriented software interface for real-time code generation. Some specific conflicting industrial tasks in real industrial crane application, such as fast load positioning control and load swing angle minimization, are considered and compared with other controller types.

Keywords: Parallel Distributed Compensation, Linear matrix inequalities, TP model transformation, Single Pendulum Gantry (SPG), position control, swing angle control

1 Introduction

The main contribution of this paper is that it investigates the performance of the TP model transformation-based control design in a real experimental setup, and evaluates and compares the results. The study is conducted through the example of a translational electromechanical system, the Single Pendulum Gantry (SPG), an educational testbed of University of Zagreb. We derive three different controllers. One is based on a classical linearization and pole placement technique. It approximates the given model by one LTI system and derives one feedback gain. The second and the third ones are based on the TP model, that is the nonlinear combination of LTI models. The second design is very similar to the first one, but determines feedback gains by pole-placement to all LTI component of the model. Finally the control value is given by nonlinear combination of the feedback gains. The third one also generates one feedback gains to each LTI system, but the gain are oprimised by linear matrix inequalities instead of pole-placement. The performance of the three controllers are compared on the reail system.

The TP model representation belongs to the class of polytopic models. The TP model represents the Linear Parameter Varying state-space models by the parameter varying combination of Linear Time Invariant (LPV) models. The TP model transformation was proposed as a uniform and automatic way to transform given LPV models to TP model form [4, 5]. The TP model transformation was soon introduced as the Higher Order Singular Value Decomposition (HOSVD) of Linear Parameter Varying (LPV) state-space models, and the result of the TP model transformation was defined as the HOSVD-based canonical form of LPV models [20, 21]. Further, the TP model transformation offers options to satisfy various convexity constrains on the type of the resulting parameter varying combination. For instance, the Linear Matrix Inequality-based control designs [1, 2, 3], under the Parallel Distributed Compensation framework [6], can immediately be executed on the resulting polytopic model if the parameter varying combination defines a convex combination. Furthermore, if it is, for instance, define tight convex hull then the feasibility of the LMI-based design is significantly relaxed. The TP model transformation is capable of generating various types of convex parameter combinations for the resulting polytopic model automatically [10, 13, 14, 16].

The TP model transformation was applied in non-linear complex and benchmark problems for controller and observer design [8, 9, 16, 18]. The approximation properties of the TP model form were examined in papers [11, 12]. Tradeoff property of the TP model transformation was studied in [11]. Further computational improvement of the TP model transfromation is presented in [19, 20, 21, 22].

The practical advantage of the TP model transformation-based control design framework is that it can be uniformly and automatically executed on a regular computer without human interaction. Recently, the TP transformation is applied for sliding surface sector design of a variable structure system to reduce the chattering, which is the main problem of sliding mode control [15].

The paper is organized as follows: Section II discuses the theoretical background of TP model transformation-based control design. Section III introduces the laboratory development system. Section IV describes mathematical model of the experimental set up. Section V explains the basic steps of the controller design. Section VI presents the experimental results and Section VII concludes this paper.

2 Tensor Product Model Transformation-based Control Design Methology

2.1 Definition of the TP Model Form of LPV Models

Consider the following linear parameter-varying state-space model:

$$\begin{pmatrix} \dot{x}(t) \\ y(t) \end{pmatrix} = S(p(t)) \begin{pmatrix} x(t) \\ u(t) \end{pmatrix} \tag{1}$$

with input $\mathbf{u}(t)$, output $\mathbf{y}(t)$ and state vector $\mathbf{x}(t)$. The system matrix $\mathbf{S}(\mathbf{p}(t)) \in \mathrm{R}^{\mathrm{OxI}}$ is a parameter-varying object, where $\mathbf{p}(t) \in \Omega$ is time varying N-dimensional parameter vector, and is an element of the closed hypercube $\Omega = [a_1;b_1]\times[a_2;b_2]\times\ldots\times[a_N;b_N] \subset \mathrm{R}^N$. $\mathbf{p}(t)$ can also include the elements of the state-vector $\mathbf{x}(t)$, therefore (1) is considered in the class of nonlinear dynamic state-space models.

Definition 1 ***Finite element TP model:*** *The* $\mathbf{S}(\mathbf{p}(t))$ *of (1) is given for any parameter* $\mathbf{p}(t)$ *as the convex combination of LTI system matrices* **S** *also called* vertex systems:

$$\begin{pmatrix} \dot{x}(t) \\ y(t) \end{pmatrix} = S \overset{N}{\underset{n=1}{\otimes}} w_n(p_n(t)) \begin{pmatrix} x(t) \\ u(t) \end{pmatrix} \tag{2}$$

where row vector $w_n(p_n) \in \mathrm{R}^{In}$ $n = 1,\ldots,N$ *contains one variable weighting functions* $w_{n,i_n}(p_n),(i_n = 1..I_n)$. *Function* $w_{n,i_n}(p(t)) \in [0,1]$ *is the* i_n*-th weighting function defined on the n-th dimension of* Ω*, and* $p_n(t)$ *is the n-th element of vector* $\mathbf{p}(t)$. $I_n < \infty$ *denotes the number of the weighting functions used in the n-th dimension of* Ω*. Note that the dimensions of* Ω *are respectively assigned to the elements of the parameter vector* $\mathbf{p}(t)$*. The* $(N+2)$*-dimensional coefficient tensor* $S \in R^{I_1 \times I_2 \times \cdots \times I_N \times O \times I}$ *is constructed from LTI vertex systems* $\mathbf{S}_{i_1 i_2 \ldots i_N} \in R^{O \times I}$.

For tensor notation we refer to [23]

Definition 2 ***Convex finite element TP model:*** *Assume that we have the explicit function (2) of a TP model. If the weighting functions satisfy that:*

$$\forall n, p_n : \sum_{i=1}^{I_n} w_{n,i}(p_n) = 1 \tag{3}$$

then (2) becomes a convex combination, namely, the LTI systems $\mathbf{S}_{i_1,\ldots,i_N}$ *form a convex hull of the given LPV model.*

2.2 The Main Steps of the TP Model Transformation

The TP model transformation starts with (1) and results in the convex finite element TP model form (2) with (3). Main steps of the Tensor-Product Model Transformation is shown in Fig. 1. First the transformation space is defined by Ω which the parameter vector $\mathbf{p} \in \Omega$ varies in. Then the parameter varying system matrix is discretised in Ω. This means the computation of system matrix **S**(**g**) over the grid points **g** of a hyper rectangular grid net defined in Ω. The second step extracts the singular value-based orthonorm structure of the system, namely, this step determines the minimal number the LTI systems in orthonorm position according to the ordering of the singular values and defines the orthonorm discretised weighting functions of the searched polytopic model. The second part of this step is capable of modifying the LTI systems and the discretised weighting functions, in order to satisfy further conditions for the weighting functions. For instance, this step can ensure the convexity of the weighting functions (3). The third step determines the continuous weighting functions from the discretised ones. If the given LPV model has no TP structure, then the resulting TP model is an approximation of the given LPV model. The approximation accuracy can be controlled by the TP model transformation.

2.3 TP Model Transformation-based Control

The structure of the control design is shown in Fig. 1. The LMI-based control design theorems under the PDC framework can immediately be executed on the finite element convex TP model. The multi-objective control performance can be expressed in terms of LMIs. For instance, various LMI theorems are proposed in [6] for different conditions. We also can find a number of further and relaxed theorems in the related literature. LMI theorems are also proposed for obsrever design as well.

In conclusion, we can substitute the LTI systems of the resulting TP model into LMIs selected according to the desired control performances. The solution of these LMIs determines one LTI feedback gain **F** to each LTI system. Computing the feedback gains over the same weighting functions by tensor product gives the control value as:

$$\mathbf{u}(t) = -\mathcal{F} \bigotimes_{n=1}^{N} \mathbf{w}_n(p_n(t))\mathbf{x}(t) \tag{4}$$

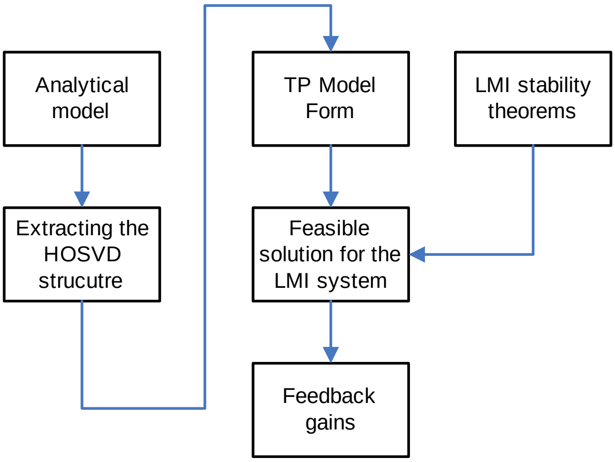

Analytical model
Extracting the HOSVD strucutre
TP Model Form
Feasible solution for the LMI system
Feedback gains
LMI stability theorems

Figure 1
Main steps of Tensor Product Model Transformation-based Control Design

This can always be given in the typical polytopic form where the same feedback gains are applied, but with linear indexing and the tensor product of the one variable weighting functions are given as multi variable weighting functions with the same indexing:

$$u(\mathrm{t}) = -\left(\sum_{r=1}^{R} w_{\mathrm{r}}(p(\mathrm{t}))F_{\mathrm{r}}\right)x(\mathrm{t}) \tag{5}$$

3 Concept of the Development System

The development system is realized in the frame of mechatronics laboratory (Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia). It is based on the different electromechanical modules, which can be controlled, analyzed and optimized from a personal computer, Fig. 2.

Instead of a separate target microcontroller for the controlling task (each electromechanical module), proposed solution is based on the microprocessor of the personal computer and advanced software tools. The testbed consists of specific electromechanical plant (mechanical subsystem) controlled by PC. The user application is modeled, simulated, programmed and run on the PC. The communication with electromechanical plant is provided by a data acquisition card (DAC) mounted in PCI slot of a personal computer and terminal board. The terminal board covers a broad range of input and output signals allowing interfacing to a variety of devices via analogue and digital signals as well as quadrature encoders. Communication between the computer and the electromechanical plant is fast

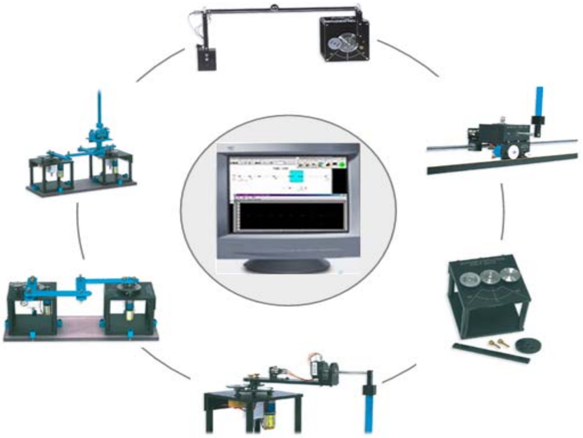


Figure 2
Development system based on different PC controlled electromechanical models

enough to ensure real time controlling of the system. This solution is based on the Windows operating system which is not real-time environment and because of that, specific and optimized software tools have to be used.

Systems 'hardware chain' consists of a personal computer (PC), data acquisition board (DAC), terminal board, and power supply with amplifier unit (UPM) and different electromechanical plants, Fig. 3. There are no strong demands on PC, it should be Pentium class processor or better (the faster the better), 16 MB RAM minimum, with Windows 95/98/Me/NT/2000/XP. Terminal unit is connected to DAC board supplied with 16 differential 14 bit analogue inputs, 4 analogue 12 bit outputs, 6 optical encoder inputs, 48 programmable digital inputs. Universal power module (UPM) with +/-15V, 3A has amplifier for electromechanical plant's actuators (DC motors). Electromechanical plants are modular in construction, each one has module with rotational or translational output, [24, 25]. This is according to the possibility of industrial translational and rotational crane models investigations. Rotational module is equipped with DC motor with planetary gearbox, incremental encoder as a speed feedback, load antibacklash gearbox and additional mass for experiments with variable inertia load. In rotational experiments with pendulum, incremental encoder for pendulum angle measurement is added.

Planar translational module (Single Pendulum Gantry, SPG) consists from a cart moving on the horizontal track and suspended pendulum. There is DC motor on the cart (the same as for rotational module) with planetary gearbox and two incremental encoders for cart position feedback and pendulum angle measurement. These two modules are core of practically all mechatronic experiments, aimed for TP model transformation-based controller developing, as well as for other controller types investigation.

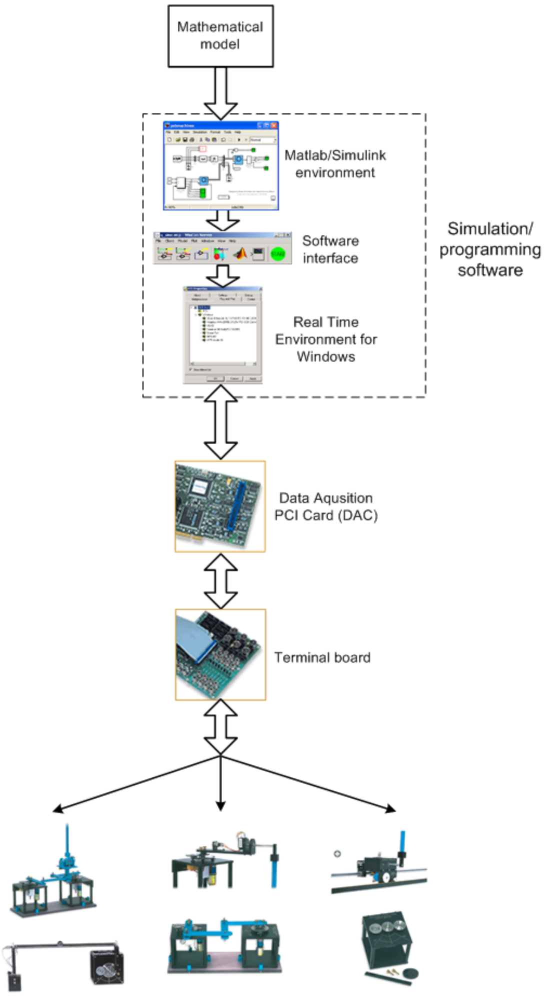
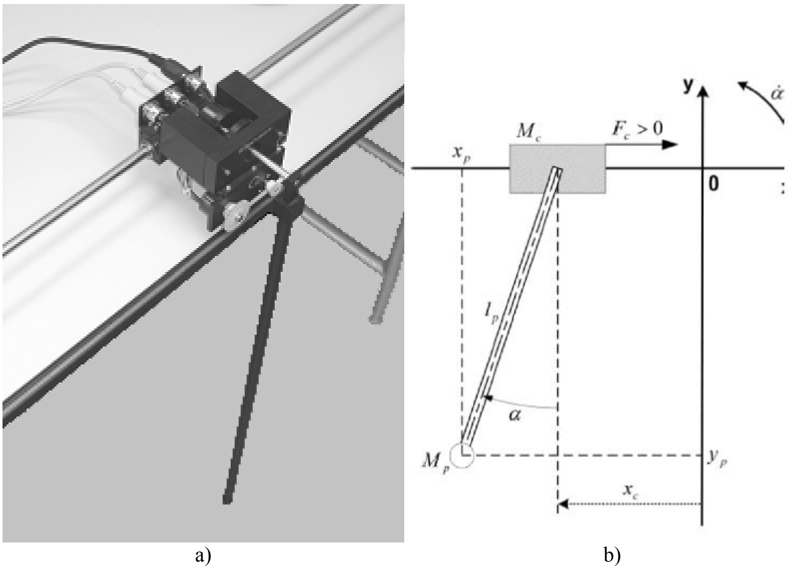

Mathematical model
Matlab/Simulink environment
Software interface
Real Time Environment for Windows
Simulation/ programming software
Data Aqusition PCI Card (DAC)
Terminal board

Figure 3
Structure block diagram of development system with different electromechanical models

Other modules are coupled with basic rotational and translational accessories (pendulums, arms, gears, etc.) forming different type of experiments. It is possible to run close to twenty different experiments with different levels of difficulty. Some of them are:

- Position and speed control with rotational and translational electromechanical plants
- Ball and beam experiment with balancing the ball on the beam
- Antipendulum control in rotational and translational moving (SISO and MIMO experiments)
- MIMO experiments with 2D gantry and 2D robot inverted pendulum
- Self erected inverted pendulum in rotational and translational moving (only SISO experiments)

The system software core is WinCon, real-time Windows 2000/XP application, [26]. It allows running code generated from a Simulink diagram in real-time on the same PC (also known as local PC) or on a remote PC. There is no need to write code by hand. Before a Simulink model may be run in real-time, it is needed first to generate the real-time code in Real-Time Workshop (RTW). Changes are as easy as modifying the Simulink diagram. Data from the real-time running code may be plotted on-line in WinCon scopes and model parameters may be changed on the fly through WinCon control panels as well as Simulink. The automatically



a) b)

Figure 4
SPG photo in mechatronics laboratory a), and schematics of the model b)

generated real-time code constitutes a stand-alone controller (i.e. independent from Simulink) and can be saved in WinCon projects together with its corresponding user-configured scopes and control panels.

4 TP Model-based Controller Design to the Single Pendulum Gantry

The Single Pendulum Gantry system, shortly described in prior section, is used for experiment verification of the TP model transformation-based position and load swing angle controller, Fig. 4a. It is also used for education and research purposes in Laboratory of Mechatronics at University of Zagreb. It is an experimental test-bed, and the goal is to design, compare and evaluate several controller approaches, [17].

4.1 Equation of Motion of the Single Pendulum Gantry

Let us consider the stabilization problem as shown in Figure 4b. Only a brief discussion is presented here, for detailed description, please, refer to [17]. Letting

$\mathbf{x} = \begin{pmatrix} \mathrm{x}_1 & \mathrm{x}_2 & \mathrm{x}_3 & \mathrm{x}_4 \end{pmatrix}^{\mathrm{T}} = \begin{pmatrix} \mathrm{x}_{\mathrm{c}} & \dot{\mathrm{x}}_{\mathrm{c}} & \alpha & \dot{\alpha} \end{pmatrix}^{\mathrm{T}}$, the equations of motion in linear parameter-varying state-space form is:

$$\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x})\mathbf{x} + \mathbf{B}(\mathbf{x})u \qquad (6)$$

where

$$\mathbf{A}(\mathbf{x}) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & a_1/a_x & a_2/a_x & a_3/a_x \\ 0 & 0 & 0 & 1 \\ 0 & a_4/a_x & a_5/a_x & a_6/a_x \end{pmatrix} \qquad \mathbf{B}(\mathbf{x}) = \begin{pmatrix} 0 \\ b_1/a_x \\ 0 \\ a_2/b_x \end{pmatrix}$$

$$a_1 = -\left(I_p + M_p l_p^2\right)\left(\frac{\eta_g K_g^2 \eta_m K_t K_m}{R_m r_{mp}^2} + B_{eq}\right)$$

$$a_2 = \frac{M_p^2 l_p^2 g \cos(x_3)\sin(x_3)}{x_3}$$

$$a_3 = \left(M_p^2 l_p^3 + l_p M_p l_p\right)\sin(x_3)x_4 + M_p l_p B_p \cos(x_3)$$

$$a_4 = M_p l_p \cos(x_3)\left(B_{eq} - \frac{\eta_g K_g^2 \eta_m K_t K_m}{R_m r_{mp}^2}\right)$$

$$a_5 = \frac{-\left(M_c + M_p\right)M_p l_p \sin(x_3)}{x_3}$$

$$a_6 = -\left(M_c + M_p\right)B_p - M_p^2 l_p^2 \cos(x_3)\sin(x_3)x_4$$

$$a_x = \left(M_c + M_p\right)I_p + M_c M_p l_p^2 + M_p^2 l_p^2 \sin^2(x_3)$$

$$b_1 = -\left(I_p M_p l_p\right)^2 \frac{\eta_g K_g \eta_m K_t}{R_m r_{mp}}$$

$$b_2 = -M_p l_p \cos(x_3)\frac{\eta_g K_g \eta_g K_t}{R_m r_{mp}}$$

The parameters of the experimental system are given in Table I.

TABLE I
PARAMETERS OF THE SPG SYSTEM

Description	Parameter	Value	Units
Equivalent viscoust damping coefficien	B_{eq}	5.4	N ms/rad
Viscous damping coefficient	B_p	0.0024	N ms/rad
Planetary gearbox efficiency	η_g	1	—
Motor efficiency	η_m	1	—
Gravitational constant of earth	g	9.81	m/s^2
Pendulum moment of inertia	I_p	0.0078838	kg m^2
Rotor moment of inertia	J_m	3.9001e-007	kg m^2
Planetary gearbox gear ratio	K_g	3.71	—
Back electro-motive force constant	K_m	0.0076776	Vs
Motor torque constant	K_t	0.007683	Nm/A
Pendulum length from pivot to COG	l_p	0.3302	m
Lumped mass of the cart system	M_c	1.0731	kg
Pendulum mass	M_p	0.23	kg
Motor armature resistance	R_m	2.6	Ω
Motor pinion radius	$r_m\, p$	0.00635	m

4.2 TP Model Representations of the Single Pendulum Gantry

Observe that the nonlinearity is caused by state values $x_3(t)$ and $x_4(t)$. The operation range of the pendulum's tip is limited to ±25deg for safety reasons, and the angular acceleration for the motor is maximum 0.7 rad/s. For the TP model transformation we define the transformation space as $\mathrm{W} = \left[\frac{-27}{180}\pi, \frac{27}{180}\pi\right] \times [-0.8, 0.8]$ (note that these intervals can be arbitrarily defined). Let the density of the sampling grid be 137×137. The sampling results in $\mathbf{A}^{s}_{i,j}$ and $\mathbf{B}^{s}_{i,j}$, where *i, j* = 1…137. Then we construct the matrix $\mathbf{S}^{s}_{i,j} = \left(\mathbf{A}^{s}_{i,j} \quad \mathbf{B}^{s}_{i,j}\right)$, and after that the tensor $S^{s} \in R^{137 \times 137 \times 4 \times 5}$ from $\mathbf{S}^{s}_{i,j}$. If we execute HOSVD on the first two dimensions of S^{s} then we find that the rank of S^{s} on the first two dimensions are 7 and 2 respectively. The singular values are as follows in the dimension x_3: $S_{1,1}$ = 1609.4, $s_{1,2}$ = 206.72, $s_{1,3}$ = 12.604, $s_{1,4}$ = 10.719, $s_{1,5}$ = 2.3109, $s_{1,6}$ = 0.14075, $s_{1,7}$ = 0.001854, and in the dimension x_4: $s_{2,1}$ = 1622.7, $s_{2,2}$ = 10.965. This means that the SPG system can be exactly given as convex combination of $7 \times 2 = 14$ linear vertex models (the L_2 numerical error of the TP model transformation for exact model is less than 10^{-12}). The TP model transformation describes SPG system as:

$$\mathbf{S}(p) = \sum_{=1}^{14} w_r(x_3, x_4)(\mathbf{A}_r x + \mathbf{B}_r u) \tag{7}$$

As in most cases it is to expensive in computational sense to work with 14 LTI models, and in real world situations the actuators accuracy is much worth than the modeling accuracy, it is possible to reduce the model. If we only keep the four biggest singular values in dimension x_3 and keep the two singular values in dimension x_4, the system can be reduced to 8 LTI models. The theoretical maximum L_2 approximation error is the sum of the discarded singular values $s_{1,5} + s_{1,6} + s_{1,7} = 2.4535$. However by checking the actual L_2 error for 10000 test points, an average maximal error of 0.080307 is received. Thus, the system can be reduced to a system of half the complexity while it is still accurate enough for real world experiments. The resulting basis functions are depicted in Figure 5.

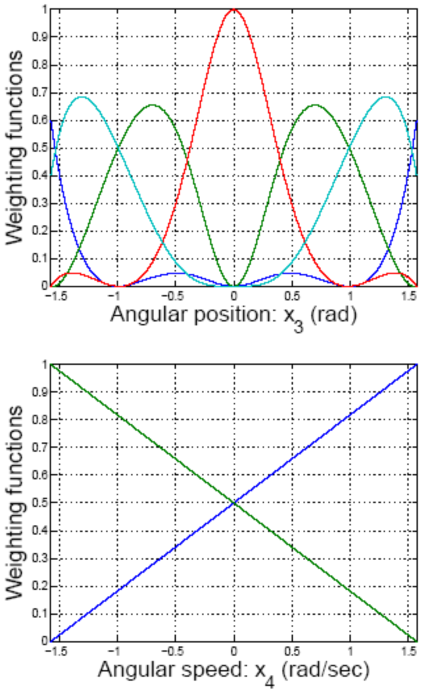


Figure 5

Weighting functions of the TP model

The LTI system matrices of the TP model are:

$$\mathbf{A}_1 = \begin{pmatrix} 0 & 1.0000 & 0 & 0 \\ 0 & -11.2630 & 1.2457 & -0.0192 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 22.8870 & -24.2374 & -0.0311 \end{pmatrix} \qquad \mathbf{B}_1 = \begin{pmatrix} 0 \\ 1.4794 \\ 0 \\ -3.0061 \end{pmatrix}$$

$$\mathbf{A}_2 = \begin{pmatrix} 0 & 1.0000 & 0 & 0 \\ 0 & -11.2906 & 1.2657 & 0.0270 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 23.1794 & -24.3744 & -0.1306 \end{pmatrix} \qquad \mathbf{B}_2 = \begin{pmatrix} 0 \\ 1.4830 \\ 0 \\ -3.0455 \end{pmatrix}$$

$$\mathbf{A}_3 = \begin{pmatrix} 0 & 1.0000 & 0 & 0 \\ 0 & -11.8223 & 1.6427 & 0.0052 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 28.5811 & -26.9299 & -0.0852 \end{pmatrix} \qquad \mathbf{B}_3 = \begin{pmatrix} 0 \\ 1.5528 \\ 0 \\ -3.7540 \end{pmatrix}$$

$$\mathbf{A}_4 = \begin{pmatrix} 0 & 1.0000 & 0 & 0 \\ 0 & -12.4388 & 2.1008 & 0.0066 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 35.3681 & -30.0863 & -0.0901 \end{pmatrix} \qquad \mathbf{B}_4 = \begin{pmatrix} 0 \\ 1.6338 \\ 0 \\ -4.6455 \end{pmatrix}$$

$$\mathbf{A}_5 = \begin{pmatrix} 0 & 1.0000 & 0 & 0 \\ 0 & -11.2630 & 1.2457 & 0.0275 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 22.8870 & -24.2374 & -0.1316 \end{pmatrix} \qquad \mathbf{B}_5 = \begin{pmatrix} 0 \\ 1.4794 \\ 0 \\ -3.0061 \end{pmatrix}$$

$$\mathbf{A}_6 = \begin{pmatrix} 0 & 1.0000 & 0 & 0 \\ 0 & -11.2906 & 1.2657 & -0.0185 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 23.1794 & -24.3744 & -0.0324 \end{pmatrix} \qquad \mathbf{B}_6 = \begin{pmatrix} 0 \\ 1.4830 \\ 0 \\ -3.0455 \end{pmatrix}$$

$$\mathbf{A}_7 = \begin{pmatrix} 0 & 1.0000 & 0 & 0 \\ 0 & -11.8223 & 1.6427 & 0.0053 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 28.5811 & -26.9299 & -0.0855 \end{pmatrix} \qquad \mathbf{B}_7 = \begin{pmatrix} 0 \\ 1.5528 \\ 0 \\ -3.7540 \end{pmatrix}$$

$$\mathbf{A}_8 = \begin{pmatrix} 0 & 1.0000 & 0 & 0 \\ 0 & -12.4388 & 2.1008 & 0.0063 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 35.3681 & -30.0863 & -0.0894 \end{pmatrix} \qquad \mathbf{B}_8 = \begin{pmatrix} 0 \\ 1.6338 \\ 0 \\ -4.6544 \end{pmatrix}$$

5 Controller Design

We compare the control performances to various different alternative solutions.

5.1 Conventional Controller based on Pole Placement

CONTROLLER 1: A linearized model is selected for the conventional state feedback control design as

$$\mathbf{A}_{lin} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -11.651 & 1.521 & 0.0049 \\ 0 & 0 & 0 & 1 \\ 0 & 26.845 & -26.109 & -0.0841 \end{pmatrix} \qquad \mathbf{B}_{lin} = \begin{pmatrix} 0 \\ 1.530 \\ 0 \\ -3.526 \end{pmatrix} \tag{8}$$

The poles of the closed loop linearized system (8) with state feedback are selected in the following way

$$\mathbf{Poles} = \begin{pmatrix} -1.8182+1.9067i \\ -20 \\ -1.8182-1.9067i \\ -40 \end{pmatrix} \tag{9}$$

The state feedback control is

$$u = -\mathbf{F}x, \quad \mathbf{F} = \begin{pmatrix} 160 & 88 & -210 & 23 \end{pmatrix}. \tag{10}$$

5.2 Derivation of TP-based Controllers

In the present case the controller (5) has the following form:

$$u = -\left(\sum_{r=1}^{8} w_r(x_3, x_4)\mathbf{F}_r \right) x \,. \tag{11}$$

Two methods are presented to define the feedback gains **F***r* for the eight systems.

CONTROLLER 2: The feedback gains **F***r* are selected separately for the all systems to place closed loop system poles to (9).

CONTROLLER 3: We design here a controller capable of asymptotically stabilize the SPG and satisfy the given constraints. We apply the following LMIs. The derivations and the proofs of these theorems are fully detailed in [6].

Theorem 1 (Asymptotic stability) *convex finite element TP model (2) with control value (5) is asymptotically stable if there exist* **X** > 0 *and* **M***r satisfying equations*

$$-\mathbf{X}\mathbf{A}_r^T - \mathbf{A}_r\mathbf{X} + \mathbf{M}_r^T\mathbf{B}_r^T + \mathbf{B}_r\mathbf{M}_r > 0 \qquad (12)$$

for all r and

$$-\mathbf{X}\mathbf{A}_r^T - \mathbf{A}_r\mathbf{X} - \mathbf{X}\mathbf{A}_s^T - \mathbf{A}_s\mathbf{X} + \mathbf{M}_s^T\mathbf{B}_r^T + \mathbf{B}_r\mathbf{M}_s + \mathbf{M}_r^T\mathbf{B}_s^T + \mathbf{B}_s\mathbf{M}_r \geq 0 \qquad (13)$$

for r < s · R, except the pairs (*r; s*) *such that wr*(**p**(*t*))*ws*(**p**(*t*)) = 0,*8***p**(*t*)*, and where the feedback gains are determined form the solutions* **X** *and* **M***ᵣ* *as*

$$\mathbf{F}_r = \mathbf{M}_r\mathbf{X}^{-1} \qquad (14)$$

In order to satisfy the constraints defined earlier, the following LMIs are added to the previous ones.

Theorem 2 (Constraint on the control value) *Assume that k***x**(0)*k* · f*, where* **x**(0) *is unknown, but the upper bound* f *is known. The constraint k***u**(*t*)*k*2 · *μ is enforced at all times t* ¸ 0 *if the LMIs*

$$\varphi^2\mathbf{I} \leq \mathbf{X}$$

$$\begin{pmatrix} \mathbf{X} & \mathbf{M}_i^T \\ \mathbf{M}_i & \mu^2\mathbf{I} \end{pmatrix} \geq 0$$

Theorem 3 (Constraint on the output) *Assume that k***x**(0)*k* · f*, where* **x**(0) *is unknown, but the upper bound* f *is known. The constraint k***y**(*t*)*k*2 · l *is enforced at all times t* ¸ 0 *if the LMIs hold.*

$$\varphi^2\mathbf{I} \leq \mathbf{X}$$

$$\begin{pmatrix} \mathbf{X} & \mathbf{X}\mathbf{C}_i^T \\ \mathbf{C}_i\mathbf{X} & \lambda^2\mathbf{I} \end{pmatrix} \geq 0$$

The bounds of the control value and the output is guaranteed by Theorem 2 and 3. Thus we solve these LMIs for the constrains together with the LMIs of Theorem 1 to guarantee asymptotic stability. By using the LMI solver of MATLAB Robust Control Toolbox, the following feasible solution and feedback gains are obtained for the controller:

$$\mathbf{F}_1 = \begin{pmatrix} 118.3947 & 51.3126 & -45.6237 & 16.4703 \end{pmatrix}$$

$$\mathbf{F}_2 = \begin{pmatrix} 118.0638 & 51.3291 & -46.1783 & 16.4069 \end{pmatrix}$$

$$\mathbf{F}_3 = \begin{pmatrix} 117.5669 & 52.2320 & -50.2620 & 15.9900 \end{pmatrix}$$

$$\mathbf{F}_4 = \begin{pmatrix} 141.1224 & 63.2115 & -56.1795 & 19.1934 \end{pmatrix}$$

$$\mathbf{F}_5 = \begin{pmatrix} 118.2570 & 51.2608 & -45.6394 & 16.4747 \end{pmatrix}$$

$$\mathbf{F}_6 = \begin{pmatrix} 118.2075 & 51.3926 & -46.1995 & 16.3999 \end{pmatrix}$$

$$\mathbf{F}_7 = \begin{pmatrix} 117.5665 & 52.2318 & -50.2620 & 15.9900 \end{pmatrix}$$

$$\mathbf{F}_8 = \begin{pmatrix} 141.1182 & 63.2101 & -56.1805 & 19.1918 \end{pmatrix}$$

6 Experimental Results

The basic structure of TP model transformation-based controller used for simulation and for real time running too, is realized in Simulink environment and presented in Fig. 6.

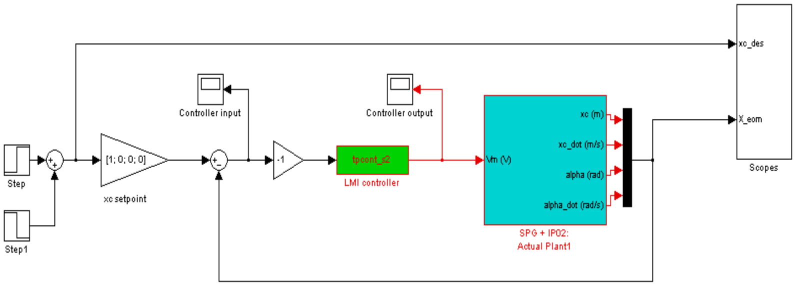


Figure 6

TP model transformation-based position and swing angle controller for SPG electromechanical system realized in Simulink environment

The experimental results with the three controllers, mentioned in chapter V, are presented in Figs. 7-9. The reference was a pulse train. In the first set of plots (Fig. 7), the time functions of the reference and the load position is shown. In the second set of plots (Fig. 8), the time functions of the angle of the load are shown. As it was expected, the performances of **CONTROLLER 1** and **CONTROLLER 2** are quite similar since they are set to have the same poles. The **CONTROLLER 3** seems to be faster but there are no significant differences among the three responses. The main difference appears in the control activity. According to Fig. 9, the **CONTROLLER 3** has the smoothest time functions.

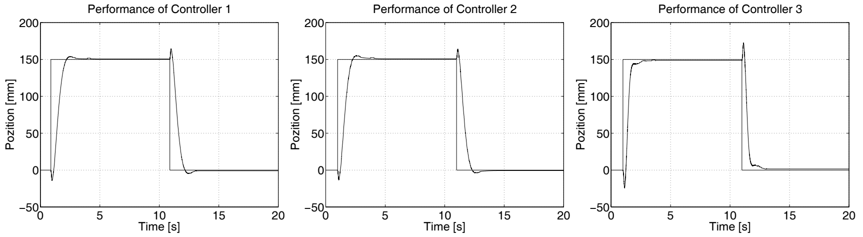


Figure 7

The position of the load (*Mp*), comparison of the performances of three controllers

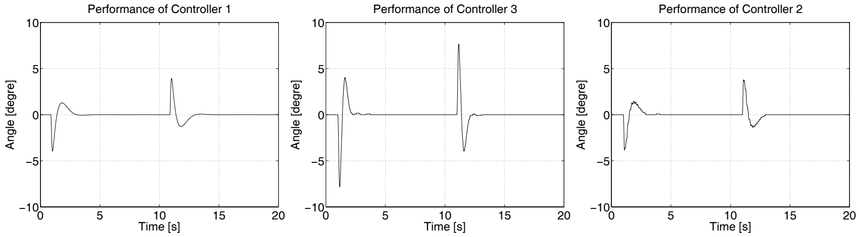


Figure 8
The angle of the load (*Mp*), comparison of the performances of three controllers

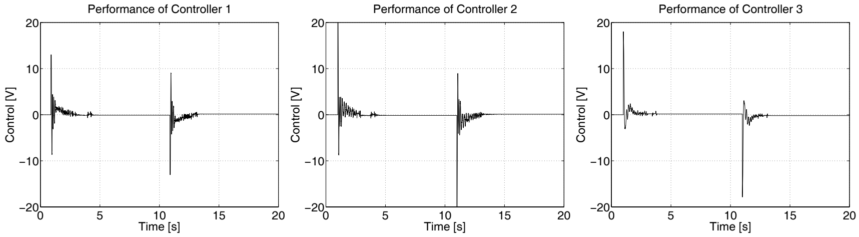


Figure 9
The control signal, comparison of the performances of three controllers

Conclusion

This paper presented a method by which a TP-based controller can be automatically designed for a non linear system using commercial Matlab functions. For investigation of such controller in real-time environment, development system with different electromechanical models with complete development tools is realized. Development system is based on the microprocessor of the personal computer and advanced software, enabling modeling, simulation, programming and real-time running of different electromechanical systems on the PC. For adequate case study, single pendulum gantry electromechanical system has been chosen, in order to mimic the real industrial task-load position and swing angle control in gantry crane load (e.g. container) handling application. Keeping the four biggest singular values for angular position and two for angular speed, 14 LTI models are reduced to 8, resulting for real-world application acceptable trade-off between modeling accuracy and computational time. Owing to this key step, the bridge between theoretical background to practical applications of TP model transformation technique is built. Experimental results realized on SPG electromechanical system, confirmed that TP model transformation-based controller can handle real time application in a good manner.

Acknoledgment

The authors wish to thank for their financial support stemming from the Hungarian-Croatian Intergovernmental Science and Technology Cooperation Program.

References

[1] S. Boyd, L. E. Ghaoui, E. Feron, V. Balakrishnan: Linear Matrix Inequalities in System and Control Theory, *Philadelphia PA:SIAM, ISBN 0-89871-334-X*, 1994

[2] P. Gahinet, A. Nemirovski, A. J. Laub, M. Chilali: *LMI Control Toolbox*. The MathWorks, Inc., 1995

[3] C. W. Scherer, S. Weiland: *Linear Matrix Iequalities in Control*, ser. DISC course lecture notes, DOWNLOAD: http://www.cs.ele.tue.nl/SWeiland/lmid.pdf, 2000.

[4] P. Baranyi, D. Tikk, Y. Yam, R. J. Patton: From Differential Equations to PDC Controller Design via Numerical Transformation, *Computers in Industry, Elsevier Science*, Vol. 51, pp. 281-297, 2003

[5] P. Baranyi: TP Model Transformation as a Way to LMI-based Controller Design, *IEEE Transaction on Industrial Electronics*, Vol. 51, No. 2, pp. 387-400, April 2004

[6] K. Tanaka, H. O. Wang: *Fuzzy Control Systems Design and Analysis — A Linear Matrix Inequality Approach*. John Wiley and Sons, Inc., 2001

[7] P. Baranyi: Tensor-Product Model-Based Control of Two-Dimensional Aeroelastic System, *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 2, March-April, 2006, pp. 391-400

[8] P. Baranyi, Z. Petres, P. Várlaki, P. Michelberger: Observer and Control law Design to the TORA System via TPDC Framework, *WSEAS Transactions on Systems*, Issue 1, Vol. 5, January 2006, pp. 156-163 (ISSN 1109-2777)

[9] G. Hancke, Á. Szeghegyi: Application Study of the TP Model Transformation in the Control of an Inverted Pendulum, in *International Conference on Computational Cybernetics (ICCC)*, Siófok, Hungary, 2003

[10] J. Bokor, P. Baranyi, P. Michelbereger, P. Várlaki: TP Model Transformation in Non-Linear System Control, in *3rd IEEE International Conference on Computational Cybernetics (ICCC)*, Mauritius, April 13-16, 2005, pp. 111-119, (plenary lecture) ISBN: 0-7803-9474-7

[11] P. Baranyi, P. Korondi, R. J. Patton, H. Hashimoto: Trade-off between Approximation Accuracy and Complexity for TS fuzzy Models, *Asian Journal of Control*, Vol. 6, No. 1, Issue 21, March 2004, pp. 21-33

[12] D. Tikk, P. Baranyi, R. J. Patton, J. Tar: Approximation Capability of TP model forms, *Australian Journal of Intelligent Information Processing Systems*, Vol. 8, No. 3, 2004, pp. 155-163

[13] Z. Petres, P. L. Várkonyi, P. Baranyi, P. Korondi: Different Affine Decomposition of the TORA System by TP Model Transformation, in *IEEE 9th*

International Conference on Intelligent Engineering Systems, Athens, Greece, September 16-19, 2005, pp. 93-98

[14] P. Baranyi, P. L. Várkonyi, P. Korondi: Different Affine Decomposition of the Model of the Prototypical Aeroelastic Wing Section by TP Model Transformation, in *IEEE 9th International Conference on Intelligent Engineering Systems*, Athens, Greece, September 16-19, 2005, pp. 105-110

[15] P. Korondi: Sliding Sector Design based on Tensor Product Model Transformation, *Transaction on Automatic Control and Computer Science*, Vol. 51, No. 1, pp. 101-108, 2006

[16] P. Baranyi: Output Feedback Control of 2-D Aeroelastic System, *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 3, May-June 2006, pp. 762-767 (ISSN 0731-5090)

[17] F. Kolonic, A. Poljugan, A. Slutej: Modern Laboratory Concept for Mechatronic Education, in *XXVII International Convention MIPRO 2004*, Opatija, Croatia, May 2004, pp. 143-146

[18] P.Baranyi: Case Study of the TP Model Transformation in the Control Design of a Complex Dynamic Model with Structural Non-Linearity, *IEEE Transaction on Industrial Electronics,* Vol. 53, No. 3, July 2006. pp. 895-904

[19] Z. Petres, P. Baranyi, H. Hashimoto: Decrease of the Computational Load of TP Model Transformation, Proc. of 3rd International Conference on Mechatronics (ICM 2006), Budapest, Hungary, July 3-5, 2006, pp. 655-659

[20] P. Baranyi, P. Várlaki, L. Szeidl: Numerical Reconstruction of the HOSVD-based Canonical Form of Polytopic Dynamic Models, in proc. 10th Int. Conf. on Intelligent Engineering Systems (INES 2006), London, United Kingdom, June 26-28, 2006, pp. 196-201 (ISBN 1-4244-9708-8)

[21] P. Baranyi, L. Szeidl, P. Várlaki, Y. Yam: Definition of the HOSVD-based Canonical Form of Polytopic Dynamic Models, Proc. of 3rd International Conference on Mechatronics (ICM 2006), Budapest, Hungary, July 3-5, 2006, pp. 660-665

[22] P. Baranyi, A. R. Várkonyi-Kóczy, Y. Yam, R. J. Patton: Adaption of TS Fuzzy Models without Complexity Expansion: HOSVD-based approach, *IEEE Transaction on Instrumentation and Measurement*, Vol. 54, No. 1, February, 2005, pp. 52-60

[23] L. D. Lathauwer, B. D. Moor, J. Vandewalle: A Multi Linear Singular Value Decomposition, Journal on Matrix Analysis and Applications, Vol. 21, No. 4, 2000, pp. 1253-1278

[24] SRV02, Instruction Manual, Quanser

[25] IP02, Instruction Manual, Quanser

[26] WinCon, Instruction Manual, Quanser