Experimental Control Design by TP Model Transformation

Fetah Kolonic
Faculty of Electrical Engineering and Computing
University of Zagreb
Unska 3 HR-10000 Zagreb Croatia
fetah.kolonic@fer.hr

Alen Poljugan
Faculty of Electrical Engineering and Computing
University of Zagreb
Unska 3 HR-10000 Zagreb Croatia
alen.poljugan@fer.hr

***Abstract*— The Tensor Product (TP) model transformation is a recently proposed technique for transforming given Linear Parameter Varying (LPV) models into affine model form, namely, to parameter varying convex combination of Linear Time Invariant (LTI) models. The main advantage of the TP model transformation is that the Linear Matrix Inequality (LMI) based control design frameworks can immediately be applied to the resulting affine models to yield controllers with tractable and guaranteed performance. The main objective of this paper is to study how the TP model transformation performs in a real world, experimental environment. The study is conducted trough the example of a translational electromechanical system, the Single Pendulum Gantry (SPG).**

I. Introduction

The affine model form is a dynamic model representation whereupon LMI based control design techniques can immediately be executed. It describes given LPV models by a parameter varying convex combination of LTI models. The TP model form is a kind of affine decomposition, where the convex combination is defined by one variable weighting functions of each parameter separately. Convex optimization or linear matrix inequality based control design techniques can immediately be applied to affine, hence to TP models [1]–[3]. An important advantage of the TP model representation is that the convex hull defined by the LTI models can readily be modified and analyzed via the one variable weighting functions. Furthermore, the feasibility of the LMI's can be considerably relaxed by modifying the type of the resulting convex hull.

TP model transformation based control design was proposed in recent papers [4], [5]. It is a numerical method capable of transforming given Linear Parameter Varying (LPV) state-space models (where the parameter vector may contain elements of the state vector as well) into a parameter varying convex combination of linear time invariant (LTI) models, namely to *polytopic* models. The convex combination is determined in such a way that a set of linear matrix inequality (LMI) control design theorems can immediately be applied to the resulting polytopic model. Especially those LMIs which have been developed under the Parallel Distributed Compensation framework (PDC) [6]. Based on this TP model transformation and PDC design concept the control solutions of complex and benchmark problems were presented in papers [7]–[10] and at the special session of IEEE Int. Conf. Fuzzy Systems 2004 [11] and at the IEEE Int. Conf. Intelligent Engineering Systems [12]–[14]. The 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 main contribution of this paper is that it investigates the performance of the nonlinear TP model transformation based control design in an 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.

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

II. Tensor Product Model Transformation based Control Design

Consider the following parameter-varying state-space model:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(\mathbf{p}(t))\mathbf{x}(t) + \mathbf{B}(\mathbf{p}(t))\mathbf{u}(t), \qquad (1)$$
$$\mathbf{y}(t) = \mathbf{C}(\mathbf{p}(t))\mathbf{x}(t) + \mathbf{D}(\mathbf{p}(t))\mathbf{u}(t),$$

with input $\mathbf{u}(t)$, output $\mathbf{y}(t)$ and state vector $\mathbf{x}(t)$. The system matrix

$$\mathbf{S}(\mathbf{p}(t)) = \begin{pmatrix} \mathbf{A}(\mathbf{p}(t)) & \mathbf{B}(\mathbf{p}(t)) \\ \mathbf{C}(\mathbf{p}(t)) & \mathbf{D}(\mathbf{p}(t)) \end{pmatrix} \in \mathbb{R}^{O \times I} \qquad (2)$$

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 $\Omega = [p1_{min}, p1_{max}] \times [p2_{min}, p2_{max}] \times \cdots \times [pN_{min}, pN_{max}] \subset \mathbb{R}^N$. The parameter $\mathbf{p}(t)$ can also include some elements of $\mathbf{x}(t)$.

1-4244-9713-4/06/$20.00 ©2006 IEEE

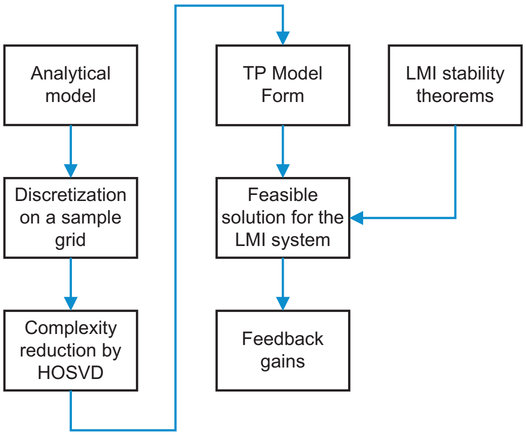

Fig. 1. Main steps of Tensor Product Model Transformation based Control Design

Main steps of Tensor Product Model Transformation based control design is shown in Fig.1. First, the nonlinear analytical model (1) is sampled on $\mathbf{\Omega}$. In other words, certain linear models are selected for certain operational points. The model (1) is estimated by linear interpolation between the sampled points. If all intervals of $\mathbf{\Omega}$ are sampled N_s times, there are N_s^N sampled systems S_s, which is usually cannot be handled numerically. That is why the next step is the reduction of the number of the sampled systems. This reduction is based on high order singular value decomposition (HOSVD). The price which should be paid for the reduction is that nonlinear interpolation is necessary between the well selected linear models. In other words, model (1) is estimated by nonlinear combination of some well selected linear systems. The final step is the controller design based on linear matrix inequality (LMI) for the all systems simultaneously. All

The TP model transformation starts with the given LPV model (1) and results in the TP model representation

$$\dot{x} \approx \mathcal{S} \underset{n=1}{\overset{N}{\otimes}} \mathbf{w}_n(p_n) \begin{pmatrix} x \\ u \end{pmatrix} \tag{3}$$

that can always be transformed to the typical form:

$$\begin{pmatrix} \dot{\mathbf{x}}(t) \\ \mathbf{y}(t) \end{pmatrix} \approx \sum_{r=1}^{R} w_r(\mathbf{p}(t))\mathbf{S}_r \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \end{pmatrix} \tag{4}$$

where

$$\left\| \mathbf{S}(\mathbf{p}(t)) - \sum_{r=1}^{R} w_r(\mathbf{p}(t))\mathbf{S}_r \right\| \leq \varepsilon.$$

Here, ε symbolizes the approximation error, $w_r(\mathbf{p}(t) \in [0,1]$ are the coefficient functions. Let the size of $\mathbf{S}_r$ is $O \times I$ in the followings. For further details about TP model transformation, please, refer to [4], [5], [16]. The convex combination of the LTI vertex systems is ensured by the conditions:

Definition 1 *The model (4) is convex if:*

$$\forall r \in [1,R], \mathbf{p}(t) \quad : \quad w_r(\mathbf{p}(t)) \in [0,1] \tag{5}$$

$$\forall \mathbf{p}(t) \quad : \quad \sum_{r=1}^{R} w_r(\mathbf{p}(t)) = 1 \tag{6}$$

This simply means that $\mathbf{S}(\mathbf{p}(t))$ is within the convex hull of the LTI vertex systems $\mathbf{S}_r$ for any $\mathbf{p}(t) \in \mathbf{\Omega}$.

The matrix $\mathbf{S}(\mathbf{p}(t))$ has a finite element TP model representation in many cases ($\varepsilon = 0$ in (4)). In this case, the TP model is said to be exact.

The TP model transformation finds the minimal number of LTI systems in exact case. If finite element TP model does not exist then the TP model transformation helps the trade-off between the number of LTI vertex systems and the ε [5]. The TP model transformation offers options to generate different types of the weighting functions $w(\cdot)$. Different weighting functions determine different types of convex hulls of the given LPV model. For instance in the case of generating tight convex the TP model transformation results in LTI systems, where as many of the LTI systems as possible are equal to the $\mathbf{S}(\mathbf{p}(t))$ over some $\mathbf{p}(t) \in \mathbf{\Omega}$ and the rest of the LTI's are close to $\mathbf{S}(\mathbf{p}(t))$ (in the sense of L_2 norm).

As an alternative way of LMI based control design the PDC framework was introduced by Tanaka and Wang [6]. The PDC design framework determines one LTI feedback gain to each LTI vertex systems of a given TP model. The framework starts with the LTI vertex systems $\mathbf{S}_r$, and results in the vertex LTI gains $\mathbf{F}_r$ of the controller. These gains $\mathbf{F}_r$ are computed by the LMI based stability theorems. After having the $\mathbf{F}_r$, the control value $\mathbf{u}(t)$ is determined by the help of the same TP model structure as used in (4):

$$\mathbf{u}(t) = -\left(\sum_{r=1}^{R} w_r(\mathbf{p}(t))\mathbf{F}_r \right) \mathbf{x}(t). \tag{7}$$

The LMI theorems, to be solved under the PDC framework, are selected according to the stability criteria and the desired control performance. For instance, the speed of response, constraints on the state vector or on the control value can be considered via properly selected LMI based stability theorems.

III. Case Study of the Single Pendulum Gantry

The Single Pendulum Gantry system is used for educational purposes at University of Zagreb, Croatia. It is an experimental testbed, and the goal is to design, compare and evaluate several controller approaches. For more details about the testbed, please, refer to [17], [18].

Let us consider the stabilization problem as shown in Figure 2. Only a brief discussion is presented here, for detailed description, please, refer to [17], [18]. Letting $\mathbf{x} = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \end{pmatrix}^T = \begin{pmatrix} x_c & \dot{x}_c & \alpha & \dot{\alpha} \end{pmatrix}^T$, the equations of motion in linear parameter-varying state-space form is:

$$\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x})\mathbf{x} + \mathbf{B}(\mathbf{x})u, \tag{8}$$

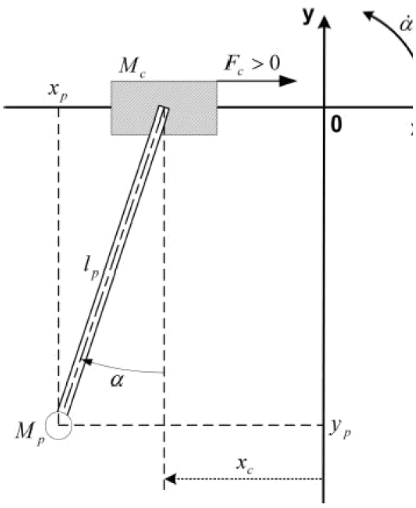

Fig. 2. Schematic of the Single Pendulum Gantry model

where

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

$$\begin{aligned}
a_1 &= -(I_p + M_p l_p^2)\left(\frac{\eta_g K_g^2 \eta_m K_t K_m}{R_m r_{mp}^2} + B_{eq}\right) \\
a_2 &= \frac{M_p^2 l_p^2 g \cos(x_3) sin(x_3)}{x_3} \\
a_3 &= (M_p^2 l_p^3 + l_p M_p l_p)\sin(x_3) x_4 + M_p l_p B_p \cos(x_3) \\
a_4 &= M_p l_p \cos(x_3)\left(B_{eq} - \frac{\eta_g K_g^2 \eta_m K_t K_m}{R_m r_{mp}^2}\right) \\
a_5 &= \frac{-(M_c + M_p) M_p l_p \sin(x_3)}{x_3} \\
a_6 &= -(M_c + M_p) B_p - M_p^2 l_p^2 \cos(x_3)\sin(x_3) x_4 \\
a_x &= (M_c + M_p) I_p + M_c M_p l_p^2 + M_p^2 l_p^2 \sin^2(x_3) \\
b_1 &= -(I_p M_p l_p)^2 \frac{\eta_g K_g \eta_m K_t}{R_m r_{mp}} \\
b_2 &= -M_p l_p \cos(x_3) \frac{\eta_g K_g \eta_m K_t}{R_m r_m p}
\end{aligned}$$

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

A. 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 $\pm 25\,\text{deg}$ for safety reasons, and the angular acceleration for the motor is maximum $0.7\,\frac{\text{rad}}{\text{s}}$. For the TP model transformation we define the transformation space as $\Omega = [\frac{-27}{180}\pi, \frac{27}{180}\pi] \times [-0.8, 0.8]$ (note that these intervals can be arbitrarily defined). Let the density of the sampling grid be 137×137. The sampling results in $\mathbf{A}_{i,j}^s$ and $\mathbf{B}_{i,j}^s$, where $i, j = 1 \dots 137$. Then we construct the matrix $\mathbf{S}_{i,j}^s = \begin{pmatrix} \mathbf{A}_{i,j}^s & \mathbf{B}_{i,j}^s \end{pmatrix}$, and after that the tensor $\mathcal{S}^s \in \mathbb{R}^{137 \times 137 \times 4 \times 5}$ from $\mathbf{S}_{i,j}^s$. The TP transformation is made by the beta version of a Matlab toolbox. If we execute HOSVD on the first two dimensions of $\mathcal{S}^s$ then we find that the rank of $\mathcal{S}^s$ on the first two dimensions are 7 and 2 respectively.

The singular values are as follows in the dimension x_3: $\sigma_{1,1} = 1609.4$, $\sigma_{1,2} = 206.72$, $\sigma_{1,3} = 12.604$, $\sigma_{1,4} = 10.719$, $\sigma_{1,5} = 2.3109$, $\sigma_{1,6} = 0.14075$, $\sigma_{1,7} = 0.001854$, and in the dimension x_4: $\sigma_{2,1} = 1622.7$, $\sigma_{2,2} = 10.965$. 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:

$$\mathbf{S}(p) = \sum_{r=1}^{14} w_r(x_3, x_4)\left(\mathbf{A}_r x + \mathbf{B}_r u\right). \tag{9}$$

As in most cases it is to expensive in computational sense to work with 14 affine models, and in real world situations the actuators accuracy is much worth than the modelling 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 affine models. The theoretical maximum L_2 approximation error is the sum of the discarded singular values the means $\sigma_{1,5} + \sigma_{1,6} + \sigma_{1,7} = 2.4535$, 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 3.

The LTI system matrices of the affine model are:

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

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

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

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

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

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

TABLE I

PARAMETERS OF THE SPG SYSTEM

Description	Parameter	Value	Units
Equivalent viscous damping coefficient	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_{mp}	0.00635	m

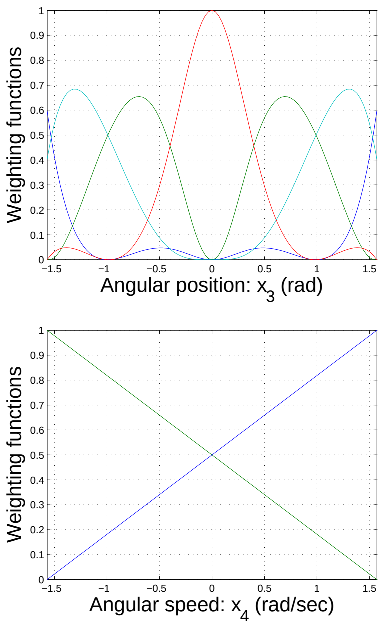

Fig. 3. Weighting functions of the TP modelon dimensions $x_3(t)$ and $x_4(t)$

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

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

A linearized model is selected for the conventional state feedback control design

$$\mathbf{A}_{lin} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -11.651 & 1.521 & 0.0049 \\ 0 & 0 & 0 & 1 \\ 0 & 26.845 & -26.109 & -0.0841 \end{pmatrix} \quad \mathbf{B}_{lin} = \begin{pmatrix} 0 \\ 1.530 \\ 0 \\ -3.526 \end{pmatrix} \tag{10}$$

IV. CONTROLLER DESIGN

We compare the control performances to various different alternative solutions.

A. Conventional controller based on pole placement

CONTROLLER 1: The poles of the closed loop linearized system (10) with state feedback are selected in the following way

$$\textbf{Poles} = \begin{pmatrix} -1.8182 + 1.9067i \\ -20 \\ -1.8182 - 1.9067i \\ -40 \end{pmatrix} \tag{11}$$

The state feedback control is

$$u = -\mathbf{F}x, \quad \mathbf{F} = \begin{pmatrix} 160 & 88 & -210 & 23 \end{pmatrix} \tag{12}$$

B. Derivation of TP based controllers

In the present case the controller (7) has the following form:

$$u = -\left(\sum_{r=1}^{8} w_r(x_3, x_4) \mathbf{F}_r \right) x, \tag{13}$$

Two methods are presented to define the feedback gains $\mathbf{F}_r$ for the eight systems.

CONTROLLER 2: The feedback gains $\mathbf{F}_r$ are selected separately for the all systems to place closed loop system poles to (11).

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) *Affine model (4) with control value (7) is asymptotically stable if there exist* $\mathbf{X} > 0$ *and* $\mathbf{M}_r$ *satisfying equations*

$$-\mathbf{X}\mathbf{A}_r^T - \mathbf{A}_r\mathbf{X} + \mathbf{M}_r^T\mathbf{B}_r^T + \mathbf{B}_r\mathbf{M}_r > \mathbf{0} \tag{14}$$

for all r and

$$-\mathbf{X}\mathbf{A}_r^T - \mathbf{A}_r\mathbf{X} - \mathbf{X}\mathbf{A}_s^T - \mathbf{A}_s\mathbf{X} + \tag{15}$$

$$+\mathbf{M}_s^T\mathbf{B}_r^T+\mathbf{B}_r\mathbf{M}_s+\mathbf{M}_r^T\mathbf{B}_s^T+\mathbf{B}_s\mathbf{M}_r\geq\mathbf{0}.$$

for $r<s\leq R$, *except the pairs* (r,s) *such that* $w_r(\mathbf{p}(t))w_s(\mathbf{p}(t))=0,\forall\mathbf{p}(t)$, *and where the feedback gains are determined form the solutions* $\mathbf{X}$ *and* $\mathbf{M}_r$ *as*

$$\mathbf{F}_r=\mathbf{M}_r\mathbf{X}^{-1}. \tag{16}$$

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* $\|\mathbf{x}(0)\|\leq\phi$, *where* $\mathbf{x}(0)$ *is unknown, but the upper bound* ϕ *is known. The constraint* $\|\mathbf{u}(t)\|_2\leq\mu$ *is enforced at all times* $t\geq0$ *if the LMIs*

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

hold.

Theorem 3 (Constraint on the output) *Assume that* $\|\mathbf{x}(0)\|\leq\phi$, *where* $\mathbf{x}(0)$ *is unknown, but the upper bound* ϕ *is known. The constraint* $\|\mathbf{y}(t)\|_2\leq\lambda$ *is enforced at all times* $t\geq0$ *if the LMIs*

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

hold.

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:

$$\begin{aligned}
\mathbf{F}_1 &= \begin{pmatrix}118.3947 & 51.3126 & -45.6237 & 16.4703\end{pmatrix}\\
\mathbf{F}_2 &= \begin{pmatrix}118.0638 & 51.3291 & -46.1783 & 16.4069\end{pmatrix}\\
\mathbf{F}_3 &= \begin{pmatrix}117.5669 & 52.2320 & -50.2620 & 15.9900\end{pmatrix}\\
\mathbf{F}_4 &= \begin{pmatrix}141.1224 & 63.2115 & -56.1795 & 19.1934\end{pmatrix}\\
\mathbf{F}_5 &= \begin{pmatrix}118.2570 & 51.2608 & -45.6394 & 16.4747\end{pmatrix}\\
\mathbf{F}_6 &= \begin{pmatrix}118.2075 & 51.3926 & -46.1995 & 16.3999\end{pmatrix}\\
\mathbf{F}_7 &= \begin{pmatrix}117.5665 & 52.2318 & -50.2620 & 15.9900\end{pmatrix}\\
\mathbf{F}_8 &= \begin{pmatrix}141.1182 & 63.2101 & -56.1805 & 19.1918\end{pmatrix}
\end{aligned}$$

V. Experimental results

The experimental results with the three controllers are presented in Fig. 4-6, The reference was a pulse train. In the first set of plots (Fig. 4) the time functions of the reference and the load position is shown. In the second set of plots (Fig. 5), 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 difference among the three responses. The main difference appears in the control activity. According to Fig. 6, the **CONTROLLER 3** has the most smooth time functions.

VI. Conclusion

This paper presented a method by which a controller can be automatically designed for a non linear system using commercial Matlab functions. The experimental results proved that this method can produce a controller which can work in a real situation.

ACKNOWLEDGMENTS

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, and 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, and M. Chilali, *LMI Control Toolbox*. The MathWorks, Inc., 1995.

[3] C. W. Scherer and 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, and 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 and H. O. Wang, *Fuzzy Control Systems Design and Analysis — A Linear Matrix Inequality Approach*. John Wiley and Sons, Inc., 2001.

[7] P. Baranyi, P. Korondi, R.J.Patton, and H. Hashimoto, “Global asymptotic stabilisation of the prototypical aeroelastic wing section via TP model transfromation,” *Asian Journal of Control*, vol. 7, no. 2, pp. 99–111, 2004.

[8] P. Baranyi, Z. Petres, P. Varlaki, and P. Michelberger, “Observer and control law design to the TORA system via TPDC framewrok,” *WSEAS Transactions on Systems*, vol. 1, pp. 156–163, 2005.

[9] G. Hancke and A. Szeghegyi, “Application study of the TP model transformation in the control of an inverted pendulum,” in *International Conference on Computational Cybernetics (ICCC)*, Sifok, Hungary, 2003.

[10] J. Bokor, P. Baranyi, P. Michelbereger, and P. Varlaki, “Tp model transformation in non-linear system control,” in *3rd IEEE International Conference on Computational Cybernetics (ICCC)*, Mauritius, Greece, 13-16 April 2005, pp. 111–119, (plenary lecture) ISBN: 0-7803-9474-7.

[11] Z. Petres, B. Resko, and P. Baranyi, “Reference signal control design of the TORA system: a TP model transformation approach,” in *IEEE Int. Conf. Fuzzy Systems*, Budapest, Hungary, 2004, p. Proc. on CD.

[12] P. Várkonyi, D. Tikk, P. Korondi, and P. Baranyi, “A new algorithm for RNO-INO type tensor product model representation,” in *IEEE 9th International Conference on Intelligent Engineering Systems*, Athens, Greece, 16-19 September 2005, pp. 263–266.

[13] Z. Petres, P. L. Vrkonyi, P. Baranyi, and P. Korondi, “Different affine decomposition of the TORA system by TP model transformation,” in *IEEE 9th International Conference on Intelligent Engineering Systems*, Athens, Greece, 16-19 September 2005, pp. 93–98.

[14] P. Baranyi, P. L. Várkonyi, and 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, 16-19 September 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.

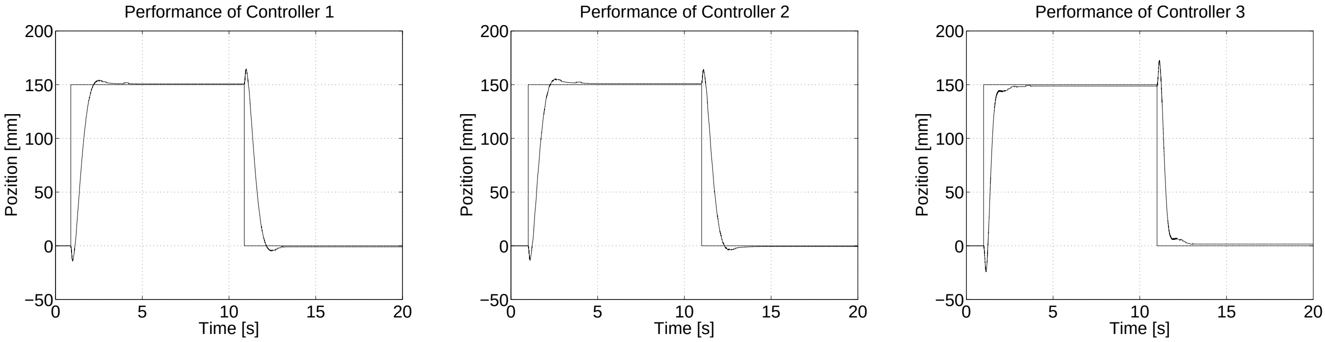

Fig. 4. The position of the load M_p, comparison of the performances of three controllers

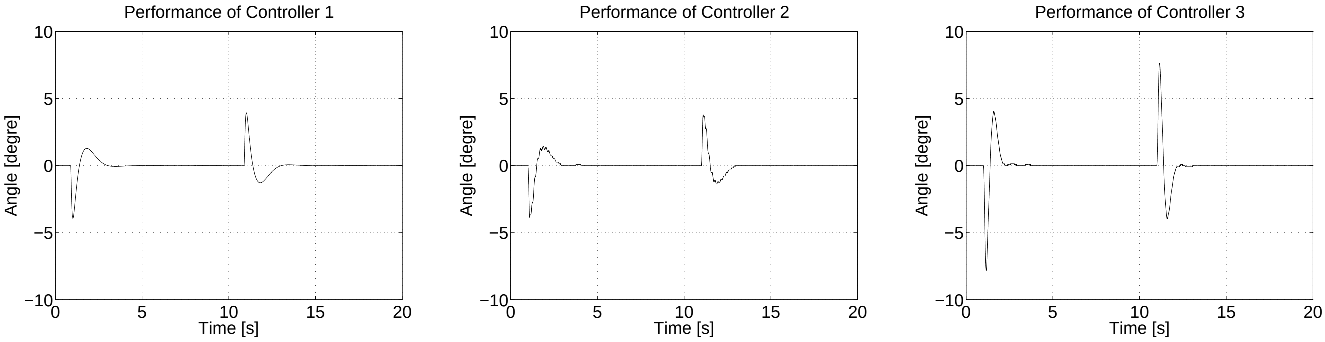

Fig. 5. The the angle of the load (M_p), comparison of the performances of three controllers

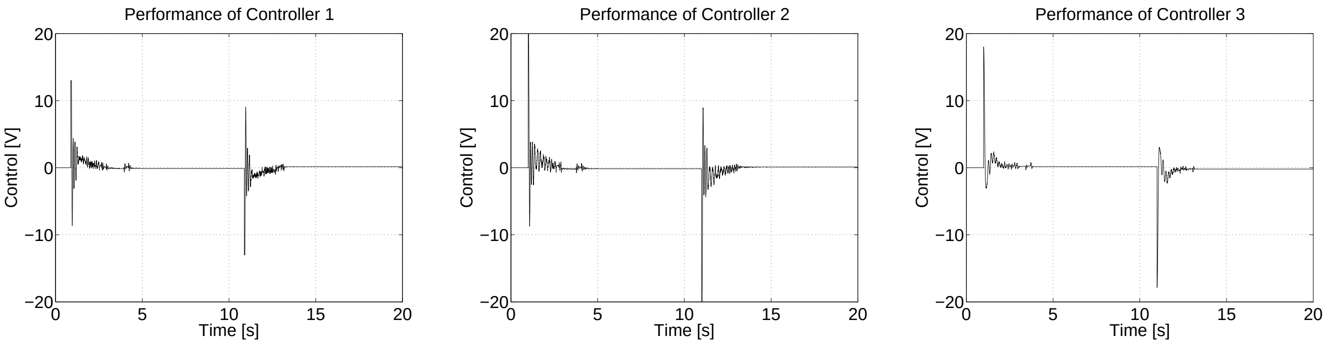

Fig. 6. The control signal, comparison of the performances of three controllers

[16] P. Baranyi, "Tensor product model based control of 2-D aeroelastic system," *Journal of Guidance, Control, and Dynamics (in Press).*

[17] F. Kolonic, A. Poljugan, and Željko Jakopovic, "Laboratory-based and industrial-oriented course in mehatronics," in *Proceedings of 13th International Conference on Electrical Drives and Power Electronics (EDPE'05)*, Dubrovnik, 2005, pp. 1–8.

[18] F. Kolonic, A. Poljugan, and A. Slutej, "Modern laboratory concept for mechatronic education," in *XXVII International Convention MIPRO 2004*, Opatija, Croatia, May 2004, pp. 143–146.