Production Systems and Information Engineering
Volume 4 (2006), pp. 53–63

TRADE-OFF PROPERTIES OF TENSOR PRODUCT MODEL TRANSFORMATION: A CASE STUDY OF THE TORA SYSTEM

ZOLTÁN PETRES
Computer and Automation Research Institute,
Hungarian Academy of Sciences, Hungary
petres@tmit.bme.hu

PÉTER BARANYI
Computer and Automation Research Institute,
Hungarian Academy of Sciences, Hungary
baranyi@sztaki.hu

[Received November 2005 and Accepted March 2006]

Abstract. The Tensor Product (TP) based models have been applied widely in approximation theory, and approximation techniques. Recently, a controller design framework working on dynamic systems has also been established based on TP model transformation combined with Linear Matrix Inequalities (LMI) within Parallel Distributed Compensation (PDC) framework. The effectiveness of the control design framework strongly depends on two main properties of the TP model used. One of them is the approximation accuracy, and the other one is computational complexity. Therefore, the primary aim of this paper is to investigate the relation of the two contradictory goals, namely, the trade-off between the dynamic TP model's accuracy and complexity. The study is conducted through the example of Translational Oscillations with a Rotational Actuator (TORA) system.

Keywords: Tensor Product model transformation, approximation accuracy, computational complexity, TORA System

1. Introduction

The demand for the decomposition of multivariate functions to univariate ones goes back to the very end of the 19th century. In 1900, in his memorable lecture at the Second International Congress of Mathematicians in Paris, D. Hilbert, the famous German mathematician, listed 23 conjectures, hypotheses concerning unsolved problems which he considered would be the most important ones to solve by the mathematicians of the 20th century [1, 2]. In the 13th, he addressed the problem of multivariate continuous function decomposition to finite superposition of continuous functions of fewer variables. The motivation is straightforward: one dimensional functions are much easier to calculate with, handle and visualize. Hilbert presumed that this problem cannot be solved in general, *i.e.* there exist multivariate continuous function that cannot be decomposed to univariate continuous functions. This was disproved by Kolmogorov in 1957 [3] in his general representation theorem, when he provided a constructive proof.

TP based approximation has reached modeling approaches of non-linear dynamic systems, and furthermore, there are now controller design frameworks based on TP model [4, 5]. Generally, TP model, in broad sense, is an approximation technique where the approximating functions are in tensor product form, whereas a TP model form is a particular approximating function in a TP model. In this paper we consider TP model in a narrower sense, when a TP model is applied to dynamic system control.

A large variety of LMI based control design techniques have been developed in the last decade [6, 7]. Powerful commercialized softwares have also been developed [8] for solving LMIs and related control problems. Recently, a number of LMI based controller designs have been carried out for TP models (also termed as polytopic or TS models in fuzzy theory) under PDC [9]. Further, a TP model transformation has been developed to transfer non-linear dynamic models to TP model whereupon PDC design frameworks can readily be executed [4, 5]. One can find a case study of TP model transformation in the control design of a prototypical aeroelastic wing section [10] that exhibits various control phenomena such as limit cycle oscillation and chaotic vibration.

A crucial point of these control design frameworks is the modeling accuracy. If TP model does not appropriately describe the real system the resulting control may not ensure the required control performance. On the other hand, by increasing the modeling accuracy the model's complexity also drastically increases and makes difficult any further calculation. Therefore, an optimal trade-off has to be chosen between the modeling accuracy and computational complexity for efficient controller design. This paper is devoted to analyze the approximation capabilities and complexity issues of TP model forms when applied to a case study, the Translational Oscillations with a Rotational Actuator (TORA) system.

The paper is organized as follows: Section 2 introduces the fundamentals of TP modeling. Section 3 discuss the TORA system and the properties of the resulting TP forms. Section 4 derives some conclusions.

2. Preliminaries

2.1. Linear Parameter-Varying state-space model

Consider the following parameter-varying state-space model:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(\mathbf{p}(t))\mathbf{x}(t) + \mathbf{B}(\mathbf{p}(t))\mathbf{u}(t), \tag{2.1}$$

$$\mathbf{y}(t) = \mathbf{C}(\mathbf{p}(t))\mathbf{x}(t) + \mathbf{D}(\mathbf{p}(t))\mathbf{u}(t),$$

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

$$\mathbf{S}(\mathbf{p}(t)) = \begin{pmatrix} \mathbf{A}(\mathbf{p}(t)) & \mathbf{B}(\mathbf{p}(t)) \\ \mathbf{C}(\mathbf{p}(t)) & \mathbf{D}(\mathbf{p}(t)) \end{pmatrix} \in \mathbb{R}^{O\times I} \tag{2.2}$$

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 $\Omega = [a_1, b_1] \times [a_2, b_2] \times \quad \times [a_N, b_N] \subset \mathbb{R}^N$ $\mathbf{p}(t)$ can also include some elements of $\mathbf{x}(t)$.

2.2. Convex state-space TP model

$\mathbf{S}(\mathbf{p}(t))$ can be approximated for any parameter $\mathbf{p}(t)$ as the convex combination of LTI system matrices $\mathbf{S}_r$, $r = 1, . \quad , R$. Matrices $\mathbf{S}_r$ are also called *vertex systems*. Therefore, one can define weighting functions $w_r(\mathbf{p}(t)) \in [0,1] \subset \mathbb{R}$ such that matrix $\mathbf{S}(\mathbf{p}(t))$ can be expressed as convex combination of system matrices $\mathbf{S}_r$. The explicit form of the TP model in terms of tensor product becomes:

$$\begin{pmatrix} \dot{\mathbf{x}}(t) \\ \mathbf{y}(t) \end{pmatrix} \approx \mathcal{S} \bigotimes_{n=1}^{N} \mathbf{w}_n(p_n(t)) \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \end{pmatrix} \tag{2.3}$$

that is

$$\left\| \mathbf{S}(\mathbf{p}(t)) - \mathcal{S} \bigotimes_{n=1}^{N} \mathbf{w}_n(p_n(t)) \right\| \leq \varepsilon.$$

Here, ε symbolizes the approximation error, row vector $\mathbf{w}_n(p_n) \in \mathbb{R}^{I_n}$ $n = 1, \ldots, N$ contains the one variable weighting functions $w_{n,i_n}(p_n)$. Function $w_{n,j}(p_n(t)) \in [0,1]$ is the j-th one variable 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 $(n = 1, . \quad , N)$ is the number of the weighting functions used in the n-th dimension of the parameter vector $\mathbf{p}(t)$. The $(N+2)$-dimensional tensor $\mathcal{S} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N\times O\times I}$ is constructed from LTI vertex systems $\mathbf{S}_{i_1 i_2 \ldots i_N} \in \mathbb{R}^{O\times I}$ For further details we refer to [10, 4, 11]. The convex combination of the LTI vertex systems is ensured by the conditions:

Definition 1. *The TP model (2.3) is convex if:*

$$\forall n \in [1, N], i, p_n(t) : w_{n,i}(p_n(t)) \in [0,1]; \tag{2.4}$$

$$\forall n \in [1,N], p_n(t) \quad \sum_{i=1}^{I_n} w_{n,i}(p_n(t)) = 1. \tag{2.5}$$

This simply means that $\mathbf{S}(\mathbf{p}(t))$ is within the convex hull of the LTI vertex systems $\mathbf{S}_{i_1 i_2 \ldots i_N}$ for any $\mathbf{p}(t) \in \Omega$.

$\mathbf{S}(\mathbf{p}(t))$ has a finite element TP model representation in many cases ($\varepsilon = 0$ in (2.3)). However, exact finite element TP model representation does not exist in general ($\varepsilon > 0$ in (2.3)), see Ref. [12]. In this case $\varepsilon \mapsto 0$, when the number of the LTI systems involved in the TP model goes to ∞. However, these models also have a finite element TP model transformation, but it is not exact, there is some approximation error. As a result we have

$$\mathbf{S}(\mathbf{p}(t)) \underset{\gamma}{\approx} \mathcal{S} \bigotimes_{n=1}^{N} \mathbf{w}_n(p_n(t)),$$

where the error γ is bounded as:

$$\gamma = \left(\left\| \mathbf{S}(\mathbf{p}(t)) - \mathcal{S} \bigotimes_{n=1}^{N} \mathbf{w}_n(p_n(t)) \right\|_{L_2} \right)^2 \leq \sum_k \sigma_k^2, \tag{2.6}$$

where σ_k are the discarded singular values.

2.3. TP model transformation

The TP model transformation starts with the given LPV model (2.1) and results in the TP model representation (2.3), where the trade-off between the number of LTI vertex systems and the ε is optimized [4]. The TP model transformation offers options to generate different types of the weighting functions $w(\cdot)$. For instance:

Definition 2. ***SN - Sum Normalization*** *Vector* $\mathbf{w}(p)$, *containing weighting functions* $w_i(p)$ *is SN if the sum of the weighting functions is 1 for all* $p \in \Omega$.

Definition 3. ***NN - Non Negativeness*** *Vector* $\mathbf{w}(p)$, *containing weighting functions* $w_i(p)$ *is NN if the value of the weighting functions is not negative for all* $p \in \Omega$.

Definition 4. ***NO - Normality*** *Vector* $\mathbf{w}(p)$, *containing weighting functions* $w_i(p)$ *is NO if it is SN and NN type, and the maximum values of the weighting functions are one. We say* $w_i(p)$ *is close to NO if it is SN and NN type, and the maximum values of the weighting functions are close to one.*

Definition 5. ***RNO - Relaxed Normality*** *Vector* $\mathbf{w}(p)$, *containing weighting functions* $w_i(p)$ *is RNO if the maximum values of the weighting functions are the same.*

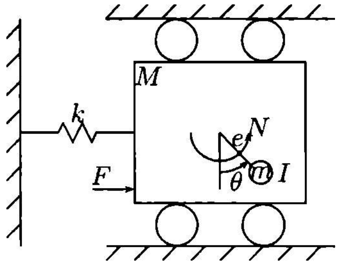

Figure 1. TORA system

Definition 6. ***INO - Inverted Normality*** *Vector* $\mathbf{w}(p)$*, containing weighting functions* $w_i(p)$ *is INO if the minimum values of the weighting functions are zero.*

All the above definitions of the weighting functions determine different types of convex hulls of the given LPV model. The SN and NN types guarantee (2.4), namely, they guarantee the convex hull. The TP model transformation is capable of always resulting SN and NN type weighting functions. This means that one can focus on applying LMI's developed for convex decompositions only, which considerably relaxes the further LMI design. The NO type determines a tight convex hull where as many of the LTI systems as possible are equal to the $\mathbf{S}(\mathbf{p})$ over some $\mathbf{p} \in \Omega$ and the rest of the LTI's are close to $\mathbf{S}(\mathbf{p}(t))$ (in the sense of L_2 norm). The SN, NN and RNO type guarantee that those LTI vertex systems which are not identical to $\mathbf{S}(\mathbf{p})$ are in the same distance from $\mathbf{S}(\mathbf{p}(t))$. INO guarantees that different subsets of the LTI's define $\mathbf{S}(\mathbf{p}(t))$ over different regions of $\mathbf{p} \in \Omega$.

These different types of convex hulls strongly effect the feasibility of the further LMI design. For instance paper [13] shows an example when determining NO is useful in the case of controller design while the observer design is more advantageous in the case of INO type weighting functions.

3. Case study of the TORA system

The Translational Oscillations with a Rotational Actuator (TORA) system[1] was developed as a simplified model of a dual-spin spacecraft [13]. Later, Bernstein and his colleagues at the University of Michigan, Ann Arbor, turned it into a benchmark problem for nonlinear control [14, 15, 16].

The system shown in Figure 1 represents a translational oscillator with an eccentric rotational proof-mass actuator. The oscillator consists of a cart of

[1] Also referred to as the rotational/translational proof-mass actuator (RTAC) system.

mass M connected to a fixed wall by a linear spring of stiffness k. The cart is constrained to have one-dimensional travel. The proof-mass actuator attached to the cart has mass m and moment of inertia I about its center of mass, which is located at distance e from the point about which the proof mass rotates. The motion occurs in a horizontal plane, so that no gravitational forces need to be considered. In Figure 1, N denotes the control torque applied to the proof mass, and F is the disturbance force on the cart.

Let q and $\dot{q}$ denote the translational position and velocity of the cart, and let θ and $\dot{\theta}$ denote the angular position and velocity of the rotational proof mass, where $\theta = 0\,\mathrm{deg}$ is perpendicular to the motion of the cart, and $\theta = 90\,\mathrm{deg}$ is aligned with the positive q direction. The equations of motion are given by

$$\begin{aligned} (M+m)\ddot{q} + kq \qquad & -me(\ddot{\theta}\cos\theta - \dot{\theta}^2\sin\theta) + F \\ (I+me^2)\ddot{\theta} \qquad & -me\ddot{q}\cos\theta + N \end{aligned}$$

With the normalization

$$\begin{aligned} \xi \triangleq \sqrt{\tfrac{M+m}{I+me^2}}\,q, \qquad & \tau \triangleq \sqrt{\tfrac{k}{M+m}}\,t, \\ u \triangleq \tfrac{M+m}{k(I+me^2)}N, \quad & w \triangleq \tfrac{1}{k}\sqrt{\tfrac{M+m}{I+me^2}}\,F, \end{aligned}$$

the equation of motion become

$$\begin{aligned} \ddot{\xi} + \xi \qquad & \varepsilon\left(\dot{\theta}^2\sin\theta - \ddot{\theta}\cos\theta\right) + w \\ \ddot{\theta} \qquad & -\varepsilon\ddot{\xi}\cos\theta + u \end{aligned}$$

where ξ is the normalized cart position, and w and u represent the dimensionless disturbance and control torque, respectively. In the normalized equations, the symbol $(\cdot)$ represents differentiation with respect to the normalized time τ The coupling between the translational and rotational motions is represented by the parameter ε which is defined by

$$\varepsilon \triangleq \frac{me}{\sqrt{(I+me^2)(M+m)}}$$

Letting $\mathbf{x} = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \end{pmatrix}^T = \begin{pmatrix} \xi & \dot{\xi} & \theta & \dot{\theta} \end{pmatrix}^T$, the dimensionless equations of motion in first-order form are given by

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u + \mathbf{d}(\mathbf{x})w, \tag{3.1}$$

where

$$\mathbf{f}(\mathbf{x}) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \frac{-1}{1-\varepsilon^2\cos^2 x_3} & 0 & 0 & \frac{\varepsilon x_4 \sin x_3}{1-\varepsilon^2\cos^2 x_3} \\ 0 & 0 & 0 & 1 \\ \frac{\varepsilon\cos x_3}{1-\varepsilon^2\cos^2 x_3} & 0 & 0 & \frac{-\varepsilon x_4\sin x_3}{1-\varepsilon^2\cos^2 x_3} \end{pmatrix}$$

Table 1. Parameters of the TORA system

Description	Parameter	Value	Units
Cart mass	M	1.3608	kg
Arm mass	m	0.096	kg
Arm eccentricity	e	0.0592	m
Arm inertia	I	0.0002175	kg m^2
Spring stiffness	k	186.3	N/m
Coupling parameter	ε	0.200	—

$$\mathbf{g}(\mathbf{x}) = \begin{pmatrix} 0 \\ \frac{-\varepsilon \cos x_3}{1-\varepsilon^2 \cos^2 x_3} \\ 0 \\ \frac{1}{1-\varepsilon^2 \cos^2 x_3} \end{pmatrix} \qquad \mathbf{d}(\mathbf{x}) = \begin{pmatrix} 0 \\ \frac{1}{1-\varepsilon^2 \cos^2 x_3} \\ 0 \\ \frac{-\varepsilon \cos x_3}{1-\varepsilon^2 \cos^2 x_3} \end{pmatrix}$$

Note that u, the control input, is the normalized torque N and w, the disturbance, is the normalized force F In the followings consider the case of no disturbance. The parameters of the simulated system are given in Table 1.

3.1. Convex state-space TP model forms of the TORA system

We execute the TP model transformation on the LPV model (3.1) of the TORA. As a first step of the TP model transformation we have to define the transformation space Ω. We define it as $\Omega = [-a, a] \times [-a, a]$ ($x_3(t) \in [-a, a]$ and $x_4(t) \in [-a, a]$), where $a = \frac{45}{180}\pi$ rad (note that these intervals can be arbitrarily defined). The TP model transformation starts with the discretization over a rectangular grid. Let the density of the discretization grid be 101×101 on ($x_3(t) \in [-a, a]$ and $x_4(t) \in [-a, a]$).

3.1.1. Exact finite TP model

The result of the TP model transformation shows that the rank of $\mathbf{S}(p)$ in the dimension of x_3 is 4, whilst in the dimension of x_4 is 2. The singular values in each dimensions are the following: $\sigma_{1,1} = 251.62, \sigma_{1,2}$ 5.7833, $\sigma_{1,3}$ 2.8396, $\sigma_{1,4}$ 0.030969; and $\sigma_{1,2} = 251.63, \sigma_{2,2}$ 5.7833. Therefore the TORA system can be exactly given as the combination of $4 \times 2 = 8$ LTI systems:

$$\mathbf{S}(p) = \sum_{i=1}^{4} \sum_{j=1}^{2} w_{1,i}(x_3) w_{2,j}(x_4) \left(\mathbf{A}_{i,j}\mathbf{x} + \mathbf{B}_{i,j} u \right). \tag{3.2}$$

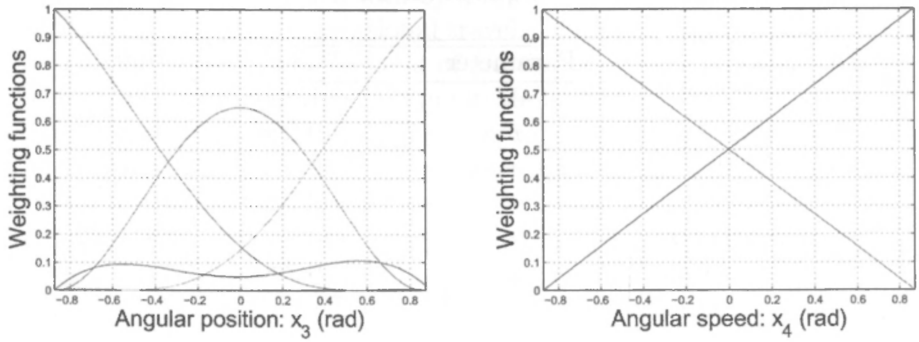

Figure 2. Close to NO type weighting functions of the exact TP model

Let us define the tight convex hull of the LPV model via generating close to NO type weighting functions by the TP model transformation, and depict them in Figure 2.

3.1.2. Approximation Trade-off of the TORA system

Even if the exact finite TP model exists, like in the case of TORA, some other reasons, such as the number of the resulting LMIs for controller design is unmanageable, or the accuracy of the actuator is much worse than the modeling accuracy, etc., can invoke the necessity to reduce the number of LTI systems.

The equation 2.6 gives a bound for the transformation error, but it is only a theoretical maximum and in most cases the resulting model has much better approximation properties. In case of the TORA system only the dimension of x_3 can be reduced, as in the dimension of x_4 if only one LTI system is kept, the Definition 1 cannot be satisfied. Thus, we repeated the TP model transformation and now only the three and two biggest singular values kept in the dimension of x_3. The result TP forms contained $3 \times 2 = 6$ and $2 \times 2 = 4$ LTI systems, respectively. During the transformation the theoretical maximum error is calculated by the equation (2.6), and also after the transformation the approximation error is measured over 10 000 sample points. Table 2 summarizes the results of the trade-off. Figure 3 and 4 show the resulting basis functions of the models.

The trade-off results showed that the size of TP model can be drastically reduced without causing unacceptable approximation error. However, it is worth noticing that the resulting reduced models might behave slightly differently

Table 2. Summary of approximation trade-off of the TORA system

Number of singular values kept	Number of LTIs	Reduction ratio	Theoretical maximal error	Measured maximal error
4	8	0%	0	10^{-12}
3	6	25%	0.0309	0.0007
2	4	50%	2.8699	0.3033

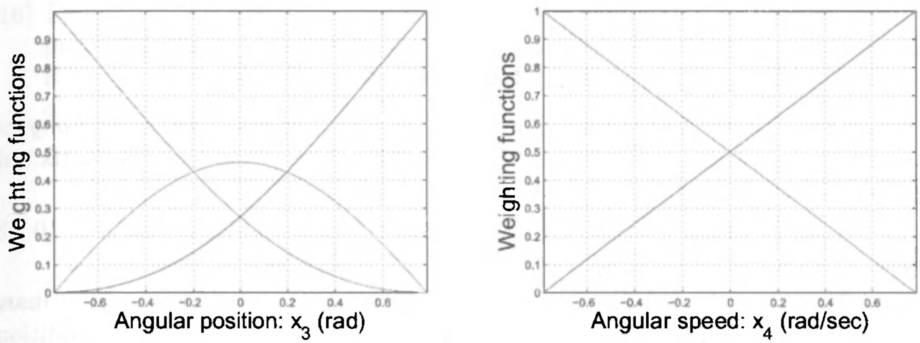

Figure 3. Close to NO type weighting functions of the reduced, 6 LTI TP model

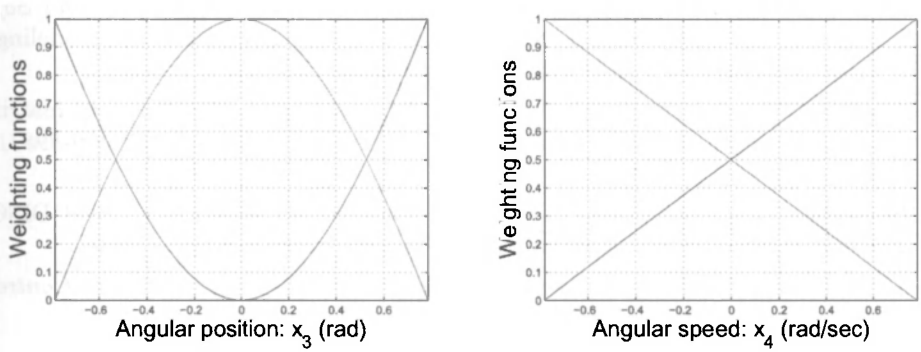

Figure 4. Close to NO type weighting functions of the reduced, 4 LTI TP model

than the exact finite model *e.g.* in controller design. These further checks are necessary to guarantee the needed behavior.

4. Conclusion

This paper shows how the TP model transformation is capable of solving the trade-off problem of the two contradictory goals, the dynamic TP model's accuracy and complexity through the case study of the TORA system. The TP model transformation gives a tool to define the theoretical maximal approximation error during the transformation. However, the case study shows that sometimes it overestimates the real error of the approximation and the model's complexity can be reduced with a large degree.

REFERENCES

[1] HILBERT, D.: Mathematische probleme. In *2nd Internation Congress of Mathematician*, Paris, France, 1900. http://www.mathematik.uni-bielefeld.de/~kersten/hilbert/rede.html.

[2] GRAY, J.: The Hilbert problems 1900–2000. *Newsletter*, **33**, (2000), 10–13. http://www.mathematik.uni-bielefeld.de/~kersten/hilbert/gray.html.

[3] KOLMOGOROV, A. N.: On the representation of continuous functions of many variables by superpositions of continuous functions of one variable and addition. *Dokl. Akad. SSSR*, **114**, (1957), 953–956. (in Russian).

[4] BARANYI, P TP model transformation as a way to LMI based controller design. *IEEE Transaction on Industrial Electronics*, **51**(2), (2004), 387–400.

[5] YAM, Y YANG, C. T. and BARANYI, P *Singular Value-Based Fuzzy Reduction with Relaxed Normalization Condition, Studies in Fuzziness and Soft Computing*, vol. 128. Springer-Verlag, interpretability issues in fuzzy modeling, J. Casillas, O. Cordon, F.Herrera, L.Magdalena (Eds.) edn. 2003.

[6] BOYD, S., GHAOUI, L. E., FERON, E., and BALAKRISHNAN, V Linear matrix inequalities in system and control theory. *Philadelphia PA:SIAM, ISBN 0-89871-334-X.*

[7] SCHERER, C. W and WEILAND, S.: *Linear Matrix Iequalities in Control.* DISC course lecture notes, http://www.cs.ele.tue.nl/SWeiland/lmid.pdf, 2000.

[8] GAHINET, P., NEMIROVSKI, A., LAUB, A. J., and CHILALI, M.: *LMI Control Toolbox.* The MathWorks, Inc., 1995.

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

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

[11] BARANYI, P., TIKK, D., YAM, Y., and PATTON, R. J.: From differential equations to PDC controller design via numerical transformation. *Computers in Industry, Elsevier Science*, **51**, (2003), 281–297.

[12] TIKK, D., BARANYI, P., PATTON, R. J., and TAR, J.: Approximation Capability of TP model forms. *Australian Journal of Intelligent Information Processing Systems*, **8**(3), (2004), 155–163.

[13] BARANYI, P Output-feedback design of 2-D aeroelastic system. *Journal of Guidance, Control, and Dynamics (in Press).*

[14] BERNSTEIN, D. S.: Special issue: A nonlinear benchmark problem. *International Journal of Robust and Nonlinear Control*, **8**.

[15] BUPP, R. T., BERNSTEIN, D. S., and COPPOLA, V T.: A benchmark problem for nonlinear control design. *International Journal of Robust and Nonlinear Control*, **8**, (1998), 307–310.

[16] BUPP, R. T., BERNSTEIN, D. S., and COPPOLA, V T Experimental implementation of integrator backstepping and passive nonlinear controllers on the RTAC testbed. *International Journal of Robust and Nonlinear Control*, **8**, (1998), 435–457.