Approximation Trade-off by TP Model Transformation

Zoltán Petres[1], Péter Baranyi[1], Fetah Kolonić[2] and Alen Poljugan[2]

[1] Computer and Automation Research Institute, Hungarian Academy of Sciences

[2] Faculty of Electrical Engineering and Computing, University of Zagreb

E-mail: petres@tmit.bme.hu, baranyi@sztaki.hu,
fetah.kolonic@fer.hr, 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 effectiveness of the LMI design depends on the LTI models of the convex combination. Therefore, the main objective of this paper is to study how the TP model transformation is capable of determining different types of convex hulls of the LTI models and how the optimal trade-off between the model's accuracy and the computational cost can be determined. The study is conducted trough the example of a translational electromechanical system, the Single Pendulum Gantry (SPG).

1 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 [5, 6, 9]. 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.

The TP model transformation is a recently proposed numerical method to transform LPV models into TP model form [3, 4]. It is capable of transforming different LPV model representations (such as physical model given by analytic equations, fuzzy, neural network, genetic algorithm based models) into TP model form in a uniform way. In this sense it replaces the analytical derivations and affine decompositions (that could be a very complex or even an unsolvable task). Execution of the TP model transformation takes a few minutes by a regular Personal Computer. The TP model transformation minimizes the number of the LTI components of the resulting TP model. Furthermore, the TP model transformation is capable of resulting different types of convex hulls of the given LPV model, and give the option to the user to define the trade-off between the model's approximation accuracy and the computational costs.

In this paper we study how the TP model transformation is applicable to generate different types of convex hulls of the given LPV models in different trade-off situations. The study is conducted through the example of a translational electromechanical system, the Single Pendulum Gantry (SPG).

2 Preliminaries

2.1 Linear Parameter-Varying state-space model

Consider the following parameter-varying state-space model:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(\mathbf{p}(t))\mathbf{x}(t) + \mathbf{B}(\mathbf{p}(t))\mathbf{u}(t), \tag{1}$$

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

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

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

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 $\Omega = [a_1, b_1] \times [a_2, b_2] \times \cdots \times [a_N, b_N] \subset \mathbb{R}^N$. $\mathbf{p}(t)$ 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 $\mathbf{S}_r$, $r = 1, \ldots, R$. 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:

$$\begin{pmatrix} \dot{\mathbf{x}}(t) \\ \mathbf{y}(t) \end{pmatrix} \approx \mathcal{S} \bigotimes_{n=1}^{N} \mathbf{w}_n(p_n(t)) \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \end{pmatrix} \tag{3}$$

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 $\mathbf{w}_n(p_n) \in \mathbb{R}^{I_n}$ $n = 1,\dots,N$ 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 $(n = 1,\dots,N)$ 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 $\mathbf{S}_{i_1 i_2 \dots i_N} \in \mathbb{R}^{O\times I}$. For further details we refer to [2–4]. The convex combination of the LTI vertex systems is ensured by the conditions:

Definition 1 *The TP model (3) is convex if:*

$$\forall n \in [1,N], i, p_n(t) : w_{n,i}(p_n(t)) \in [0,1]; \tag{4}$$

$$\forall n \in [1,N], p_n(t) : \sum_{i=1}^{I_n} w_{n,i}(p_n(t)) = 1. \tag{5}$$

This simply means that $\mathbf{S}(\mathbf{p}(t))$ is within the convex hull of the LTI vertex systems $\mathbf{S}_{i_1 i_2 \dots i_N}$ 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 (3)). However, exact finite element TP model representation does not exist in general ($\varepsilon > 0$ in (3)), see Ref. [10]. In this case $\varepsilon \mapsto 0$, when the number of the LTI systems involved in the TP model goes to ∞.

2.3 TP model transformation

The TP model transformation starts with the given LPV model (1) and results in the TP model representation (3), where the trade-off between the number of LTI vertex systems and the ε is optimized [3]. 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.*

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 (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 [1] 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.

In order to have a direct link between the TP model form and the typical form of LMI conditions, we define the following index transformation:

Definition 7 *(*Index transformation) *Let*

$$\mathbf{S}_r = \begin{pmatrix} \mathbf{A}_r & \mathbf{B}_r \\ \mathbf{C}_r & \mathbf{D}_r \end{pmatrix} = \mathbf{S}_{i_1,i_2,..,i_N},$$

where $r = ordering(i_1, i_2, \dots, i_N)$ *(*$r = 1 \dots R = \prod_n I_n$*). The function "ordering" results in the linear index equivalent of an N dimensional array's index* $i_1, i_2, \dots, i_N$*, when the size of the array is* $I_1 \times I_2 \times \cdots \times I_N$*. Let the weighting functions be defined according to the sequence of* r*:*

$$w_r(\mathbf{p}(t)) = \prod_n w_{n,i_n}(p_n(t)).$$

By the above index transformation one can write the TP model (3) in the typical form of:

$$\mathbf{S}(\mathbf{p}(t)) = \sum_{r=1}^{R} w_r(\mathbf{p}(t))\mathbf{S}_r.$$

Note that the LTI systems $\mathbf{S}_r$ and $\mathbf{S}_{i_1,i_2,\dots,i_N}$ are the same, only their indices are modified, therefore the hull defined by the LTI systems is the same in both forms.

3 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

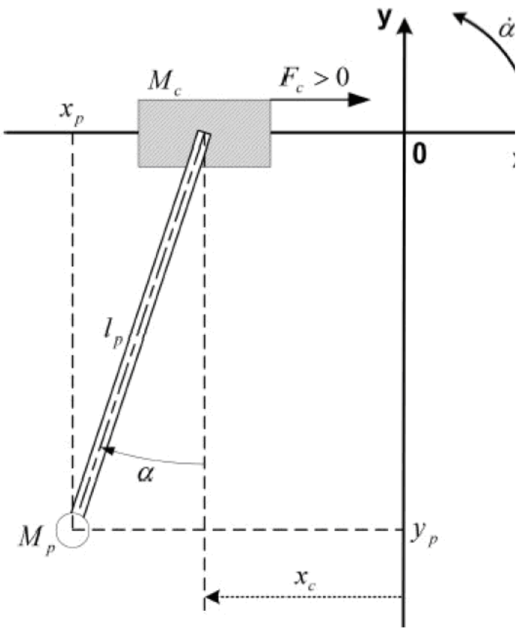

Figure 1: Schematic of the Single Pendulum Gantry model

and evaluate several controller approaches. For more details about the testbed we refer to [7, 8].

Let us consider the stabilization problem as shown in Figure 1. Here we give only a brief discussion, for detailed description we refer to [7, 8]. 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:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})\mathbf{x} + \mathbf{g}(\mathbf{x})u, \tag{6}$$

where

$$\mathbf{f}(\mathbf{x}) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & A_1/A_x & A_2/A_x & A_3/A_x \\ 0 & 0 & 0 & 1 \\ 0 & A_4/A_x & A_5/A_x & A_6/A_x \end{pmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{pmatrix} 0 \\ B_1/A_x \\ 0 \\ B_2/A_x \end{pmatrix}, \qquad \text{and}$$

Table 1: 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	—
Motor torque constant	K_t	0.007683	—
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

$$
\begin{aligned}
A_1 &= -(I_p + M_p l_p^2)\left(\frac{\eta_g K_g^2 \eta_m K_t K_m}{R_m r_{mp}^2} + B_{eq}\right) \\
A_2 &= \frac{M_p^2 l_p^2 g \cos(x_3) sin(x_3)}{x_3} \\
A_3 &= (M_p^2 l_p^3 + l_p M_p l_p)\sin(x_3) x_4 + M_p l_p B_p \cos(x_3) \\
A_4 &= M_p l_p \cos(x_3)\left(B_{eq} - \frac{\eta_g K_g^2 \eta_m K_t K_m}{R_m r_{mp}^2}\right) \\
A_5 &= \frac{-(M_c + M_p) M_p l_p \sin(x_3)}{x_3} \\
A_6 &= -(M_c + M_p) B_p - M_p^2 l_p^2 \cos(x_3)\sin(x_3) x_4 \\
A_x &= (M_c + M_p) I_p + M_c M_p l_p^2 + M_p^2 l_p^2 \sin^2(x_3) \\
B_1 &= -(I_p M_p l_p)^2 \frac{\eta_g K_g \eta_m K_t}{R_m r_{mp}} \\
B_2 &= -M_p l_p \cos(x_3) \frac{\eta_g K_g \eta_m K_t}{R_m r_m p}
\end{aligned}
$$

The parameters of the simulated system are given in Table 1.

3.1 TP model representations of the Single Pendulum Gantry

Observe that the nonlinearity is caused by $x_3(t)$ and $x_4(t)$. For the TP model transformation we define the transformation space as $\Omega = [-a, a] \times [-a, a]$ ($x_3(t) \in [-a, a]$ and $x_4(t) \in [-a, a]$), where $a = \frac{90}{180}\pi$rad (note that these intervals can be arbitrarily defined). Let the density of the sampling grid be 101×101. The sam-

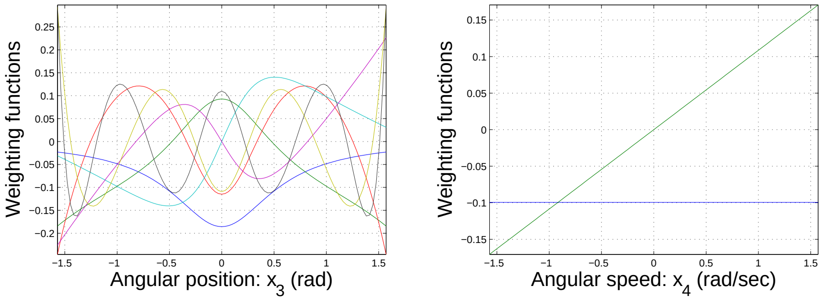

Figure 2: Weighting functions of the TP model 0 on dimensions $x_3(t)$ and $x_4(t)$

pling results in $\mathbf{A}^s_{i,j}$ and $\mathbf{B}^s_{i,j}$, where $i,j = 1 \dots 101$. Then we construct the matrix $\mathbf{S}^s_{i,j} = \begin{pmatrix} \mathbf{A}^s_{i,j} & \mathbf{B}^s_{i,j} \end{pmatrix}$, and after that the tensor $\mathcal{S}^s \in \mathbb{R}^{101\times101\times4\times5}$ from $\mathbf{S}^s_{i,j}$. 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 $7\times2 = 14$ 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:

$$\dot{\mathbf{x}}(t) = \sum_{i=1}^{7}\sum_{j=1}^{2} w_{1,i}(x_3(t))w_{2,j}(x_4(t))\left(\mathbf{A}_{i,j}\mathbf{x}(t) + \mathbf{B}_{i,j}u(t)\right). \tag{7}$$

In the followings we show that the type of the convex combination can readily be modified by the TP model transformation:

TP MODEL 0: The resulting weighting functions depicted on Figure 2 are directly obtained by the TP model transformation without any further modification. They are between -1 and $+1$ and orthogonal. The resulting LTI vertex systems do not define the convex hull of the LPV model, but their number is minimized.

TP MODEL 1: In order to have convex TP model to which the LMI control design conditions can be applied, let us generate SN and NN type weighting functions by the TP model transformation. The results are depicted on Figure 3.

TP MODEL 2: In many cases the convexity of the TP model is not enough, the further LMI design is not feasible. In order to relax the feasibility of the LMI conditions, let us define the tight convex hull of the LPV model via generating close to NO type weighting functions by the TP model transformation, see Figure 4.

TP MODEL 3: Let us further modify the weighting functions and define their INO–RNO type, see Figure 5.

The above resulting weighting functions can be derived analytically in some cases, but as the model become more and more complex, the analytical derivations

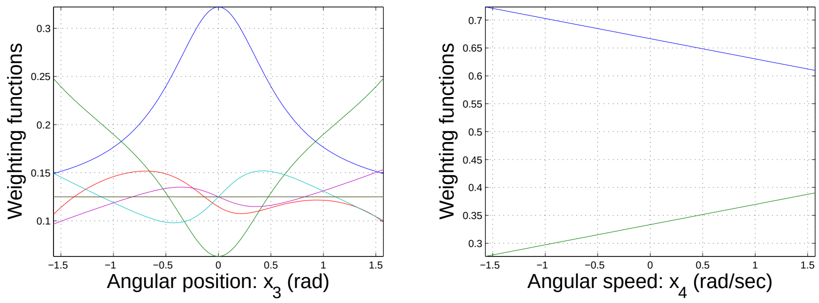

Figure 3: SN and NN type weighting functions of the TP model 1 on dimensions $x_3(t)$ and $x_4(t)$

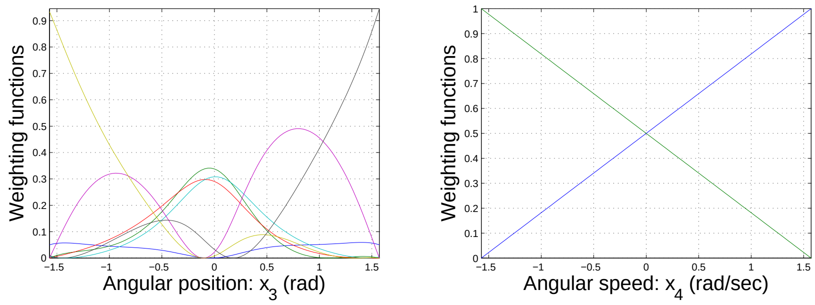

Figure 4: Close to NO type weighting functions of the TP model 2 on dimensions $x_3(t)$ and $x_4(t)$

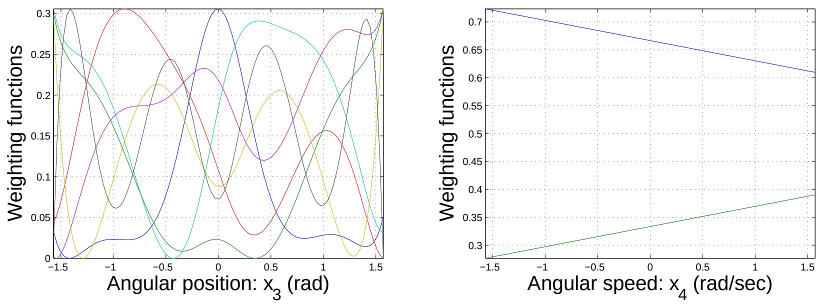

Figure 5: INO–RNO type weighting functions of the TP model 3 on dimensions $x_3(t)$ and $x_4(t)$

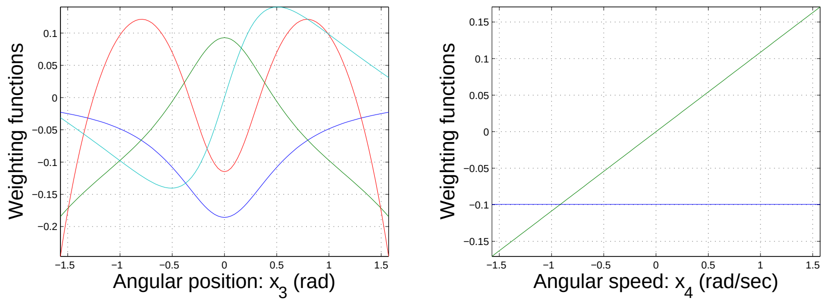

Figure 6: Weighting functions of the TP model 0 on dimensions $x_3(t)$ and $x_4(t)$ for the approximated system

needs more and more expertise. Moreover, the analytical derivations of the tight convex hull or INO–RNO type weighting functions need the analytical solution of the tight convex hull problem that is unavailable in general. In spite of this, the TP model transformation requires a few minutes and is not dependent on the actual analytical form of the given LPV model. If the model is changed we can simply execute the TP model transformation again.

3.2 Approximation trade-off

As the previous subsection described the TP model transformation is an efficient tool the transform analytical system into exact affine models. However when the complexity of the system is high, i.e. the number of nonlinear dimensions and/or number of nonzero singular values, the rank is high, then we can easily face to computation complexity problems. Therefore, during the TP model transformation, it is possible to control the complexity of resulting system by keeping less singular values than the rank of the system. In this case the TP model 3 only an approximation of the original system. The maximal approximation error of the system, ε is the sum of the discarded singular values σ_i.

As in most cases it is to expensive to work with 14 affine 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 affine models. The theoretically 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 1000 test points, an average 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 6–9.

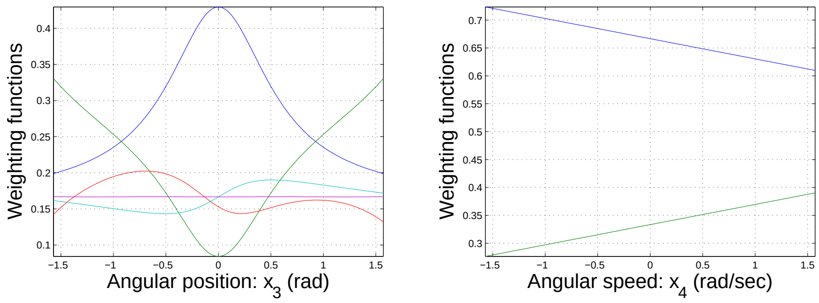

Figure 7: SN and NN type weighting functions of the TP model 1 on dimensions $x_3(t)$ and $x_4(t)$ for the approximated system

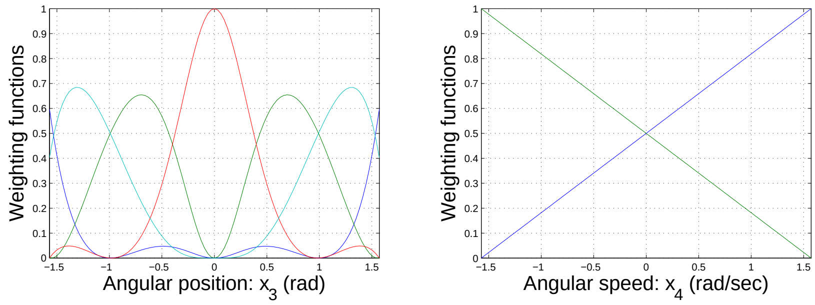

Figure 8: Close to NO type weighting functions of the TP model 2 on dimensions $x_3(t)$ and $x_4(t)$ for the approximated system

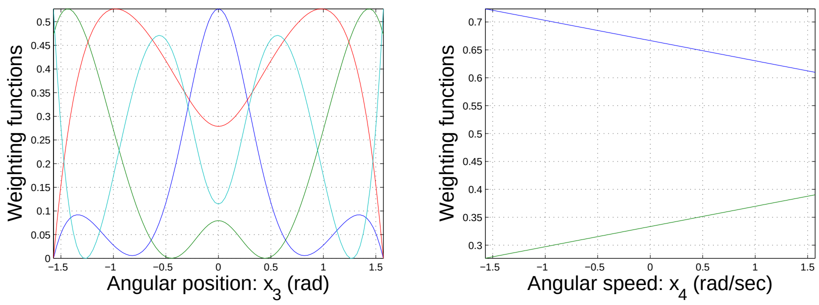

Figure 9: INO–RNO type weighting functions of the TP model 3 on dimensions $x_3(t)$ and $x_4(t)$ for the approximated system

4 Conclusion

This paper shows how the TP model transformation is capable of defining affine models with various types of convex hulls of a given LPV model in a few minutes without analytical derivations and also discussed the trade-off problem between approximation accuracy and model complexity. We may conclude that the TP model may replace the analytic affine model decomposition and can be an effective tool for generating models for real world situations. We studied the example of the LPV model of the Single Pendulum Gantry.

References

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

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

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

[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*, 51:281–297, 2003.

[5] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. Linear matrix inequalities in system and control theory. *Philadelphia PA:SIAM, ISBN 0-89871-334-X*, 1994.

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

[7] Fetah Kolonić, Alen Poljugan, and Alojz Slutej. Modern laboratory concept for mechatronic education. In *XXVII International Convention MIPRO 2004*, pages 143–146, Opatija, Croatia, May 2004.

[8] Fetah Kolonić, Alen Poljugan, and Željko Jakopović. Laboratory-based and industrial-oriented course in mehatronics. In *Proceedings of 13th International Conference on Electrical Drives and Power Electronics (EDPE'05)*, pages 1–8, Dubrovnik, 2005.

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

[10] D. Tikk, P. Baranyi, R. J. Patton, and J. Tar. Approximation Capability of TP model forms. *Australian Journal of Intelligent Information Processing Systems*, 8(3):155–163, 2004.