The Generalized TP Model Transformation for TS Fuzzy Model Manipulation and Generalized Stability Verification

Péter Baranyi

***Abstract*— The paper integrates various ideas about the TP (Tensor Product) model transformation into one conceptual framework and formulates it in terms of the TS fuzzy model manipulation and control design framework. Several new extensions of the TP model transformation are proposed, such as the quasi and "full", Compact and Rank-reduced Higher Order Singular Value Decomposition based canonical form of TS fuzzy models, and the bi-linear-, Multi -, Pseudo -, Convex -, Partial TP model transformations. All of these extensions together form the generalized TP model transformation, which provides an effective tool to freely and readily manipulate the antecedent sets and rules of TS fuzzy models, and also provides main fuzzy rule component analysis, as well as a means for complexity and accuracy trade-offs. It is demonstrated in the paper that the proposed manipulation forms a new, effective and necessary optimization step of TS fuzzy or polytopic models and Linear Matrix Inequality based control design, and can also decrease conservativeness. Identification techniques are typically constructed according to the available data and measurement set, as well as the type of system to be identified. As a result, they may not always provide good representations for control design frameworks. The paper demonstrates that the proposed TP model transformation is unique in that it bridges between various soft-computing based identification techniques and TS fuzzy model based approaches. Finally, the paper proposes the Multi TP model transformation, which is a tractable and non-heuristic framework to verify the stability of the result of fuzzy or various soft-computing based control designs. The Multi TP model transformation could provide an answer to the frequently emerging criticisms regarding the lack of mathematical stability verification techniques in soft-computing based control design. Control examples are provided in the paper.**

***Index Terms*— TP model transformation, TS fuzzy model, complexity trade-off, control optimization, stability verification, TP model transformation, parallel distributed compensation (PDC)**

I. Introduction

The Tensor Product (TP) model transformation was originally proposed in [1], [2] and summarized in [3] for qLPV control theories. It transforms a function (which can be given via closed formulas or neural networks, fuzzy logic, etc.) into TP function form if such a transformation is possible. If an exact transformation is not possible, then the method determines a TP function that is an approximation of the given function. The TP model transformation also provides a trade-off between approximation accuracy and complexity [4] of the resulting TP function. Besides being a transformation of functions, the TP model transformation is also a new concept in qLPV based control which plays a central role in providing a valuable means of bridging between identification and polytopic systems theories [1]–[3]. The TP model transformation is uniquely effective in manipulating the convex hull of polytopic forms, and, as a result has revealed and proved the fact that convex hull manipulation is a necessary and crucial step in achieving optimal solutions and decreasing conservativeness in modern LMI based control theory. Thus, although it is a transformation in a mathematical sense, it has established a conceptually new direction in control theory and has laid the ground for further new approaches towards optimality.

The core of the TP model transformation was first introduced as an approach to complexity reduction of fuzzy systems [4], [5]. Soon, it was extended to TP model transformation for system control design [1], [2], [6] and to a framework for polytopic model and LMI based system control [7]–[15]. A MATLAB toolbox was also created, see [16]. Some papers began to investigate the convex hull manipulation property [8], [17], [17]–[19] and the approximation trade-off property [9], [20]–[26] of the TP model transformation. New convex hull manipulation techniques were proposed in [8], [18], [19]. The HOSVD based canonical form of TP models was initiated in [27], [28], and it was also proved that the TP model transformation is capable of numerically reconstructing this form, see [29]. A computationally relaxed variant of the TP model transformation was proposed in [23], [30]. A centralized variant of the transformation was given in [31]. Relying on the above properties, we can find a variety of control solutions, see e.g. some key directions in [6], [32]–[55]. Further applications of the TP model transformation in sliding-mode control were presented in [11], [56].

The main goal of the paper is to introduce and conceptually restructure the TP model transformation to a generalized form for fuzzy modeling and to propose new features and several new variants of the TP model transformation, in order to achieve flexible means for TS fuzzy modeling, in the form of a manipulation and design tool extended with a general stability theorem.

First, in Section II, the paper recalls the transfer function of TS fuzzy models. Section III redefines the HOSVD based canonical form and introduces the HOSVD based canonical form as a unique representation. A simplified variant of the form, the quasi HOSVD based canonical form, is also presented. Both representations serve the task of main fuzzy

The research was supported by the Hungarian National Development Agency, (ERC-HU-09-1-2009-0004, MTASZTAK) (OMFB-01677/2009)

P. Baranyi is with the Computer and Automation Research Institute of the Hungarian Academy of Sciences, H-1111 Budapest, Kende u. 13-17, Hungary, baranyi@sztaki.hu

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

rule component analysis via the higher order singular value based ordering of the contribution of the fuzzy rules. Section IV introduces the generalized TP model transformation, which is a framework based on the extension and restructuring of the TP model transformation. This general form is capable of systematically combining different algorithms as a framework to achieve a bridging between various identification and modeling methods through the use of the TS fuzzy model representation. The resulting TS fuzzy model is amenable to flexible manipulation. To show this, a number of new concepts are introduced: convex TP model transformation for LMI based design, pseudo TP model transformation as an elementary technique for convex hull manipulation leading to control optimization, and finally multi TP model transformation as a way to transform a set of TS fuzzy models into a unified antecedent system. This section also integrates the complexity and accuracy trade-off capability of the TP model transformation, which leads to the ability to perform main fuzzy rule component analysis. Section V discusses the central role of the generalized TP model transformation in terms of two aspects. The first aspect is that the TP model transformation could be a final step of identification, and as a generalized "interface", could at the same time serve as a preprocessing step for further design requirements (e.g., to optimize control performance). The second aspect is that the multi TP model transformation can be used as a general framework for stability verification of soft computing based solutions. These discussions are demonstrated via numerical examples in Section VI. The use of the Multi TP model transformation as a stability verification framework is demonstrated on a strongly non-linear, 3DOF problem of an aeroleastic wing section. The effectiveness of the pseudo TP model transformation in LMI based design is also demonstrated in an academic numerical example.

II. Transfer function of TS fuzzy model

Let us recall the transfer function of the Takagi-Sugeno fuzzy operator and product-sum-gravity defuzzyfication based fuzzy model. Assume that we have a set of fuzzy rules with N inputs such as:

$$\begin{array}{ccccc} \text{IF} & A_{1,i_1} & \text{AND} & A_{2,i_2} & \text{AND}\cdots \\ \cdots & \text{AND} & A_{N,i_N} & \text{THEN} & B_{i_1,i_2,\ldots,i_N} \end{array}. \tag{1}$$

Let $w_{n,i}(x_n) \in [0,1]$, $x_n \in \mathbb{R}$ be the membership function of antecedent fuzzy set $A_{n,i}$, $i = 1,\ldots,I_n$ on input universe X_n, $n = 1,\ldots,N$. Let the observation fuzzy sets be singleton sets with elements x_n, and the consequent fuzzy sets $B_{i_1,i_2,\ldots,i_N}$ be singletons as well, represented by their single elements $b_{i_1,i_2,\ldots,i_N}$ on the output universe Y. If the output of the TS fuzzy model is not a scalar value, but a vector, matrix or even a tensor $\mathcal{Y}$ of dimensions $L_1 \times L_2 \times \ldots \times L_K$, then the consequent fuzzy sets represent vectors, matrices or tensors, respectively. The consequent sets can also represent parametrized functions as $f(\mathbf{b}_{i_1,i_2,\ldots,i_N}, \mathbf{x}')$. For the sake of simplicity, and without the loss of generality, we do not distinguish between what the consequent sets symbolize, and we simply assume that the consequents are assigned to parameters arranged into tensors $\mathcal{B}_{i_1,i_2,\ldots,i_N} \in \mathbb{R}^{L_1\times L_2\times\ldots\times L_K}$. In order to have a more general form, we turn to multi-output fuzzy rule bases. Let the number of outputs be denoted by $o = 1,\ldots,O$, such that each of the outputs are of the form $\mathcal{Y}_o \in \mathbb{R}^{L_1\times L_2\times\ldots\times L_K}$. We merge the output tensors into a single tensor $\mathcal{Y}$ along the $K+1$-th dimension, and whenever we need to extract a single output, we can work with separate partitions of $\mathcal{Y} \in \mathbb{R}^{L_1\times L_2\times\ldots\times L_K\times O}$. In the same way, we construct $\mathcal{B}$ from $\mathcal{B}_{i_1,i_2,\ldots,i_N,o}$ assigned to the outputs.

A very typical requirement in fuzzy modeling is the Ruspini-partition:

Definition 1 (Ruspini-partition): The antecedent membership functions are given in Ruspini-partitions if they satisfy $\forall p_n : \sum_{i=1}^{I} w_{n,i}(p_n) = 1$. Membership functions which satisfy this property are denoted by $w_{n,i}^{RP}(p_n)$.

Based on the above, we arrive at the following general transfer function for TS fuzzy models:

$$\mathcal{Y} = \Sigma_{i_1=1}^{I_1}\Sigma_{i_2=1}^{I_2}\cdots\Sigma_{i_N=1}^{I_N}\left[\Pi_{n=1}^{N} w_{n,i_n}^{RP}(x_n)\right]\mathcal{B}_{i_1,i_2,\ldots,i_N}, \tag{2}$$

where $\mathbf{x} \in \mathbb{R}^N$. This transfer function is specifically a Tensor Product (TP) function, therefore, it can be given in the form [1]–[3]:

$$\mathcal{Y} = \mathcal{B} \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n), \tag{3}$$

where $\mathbf{w}_n(x_n) = \begin{pmatrix} w_{n,1}^{RP}(x_n) & \cdots & w_{n,I_n}^{RP}(x_n) \end{pmatrix}$.

For the sake of brevity, we will refer to the above transfer function as a TS fuzzy model later on. In later sections, we will focus on fuzzy control design as well, therefore, a brief introduction to the above discussed TS fuzzy model in the context of dynamic systems modeling is also in order here. Let us consider a Linear Parameter-Varying (LPV) state-space model:

$$\begin{pmatrix} \dot{\mathbf{x}}(t) \\ \mathbf{y}(t) \end{pmatrix} = \mathbf{S}(\mathbf{p}(t)) \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \end{pmatrix}. \tag{4}$$

with input $\mathbf{u}(t)$, output $\mathbf{y}(t)$ and state vector $\mathbf{x}(t)$. The system matrix $\mathbf{S}(\mathbf{p}(t)) \in \mathbb{R}^{L_1\times L_2}$ is a parameter-varying object, where $\mathbf{p}(t) \in \Omega$ is a time varying N-dimensional parameter vector which is an element of 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 $\mathbf{x}(t)$, and, hence this model belongs to the class of non-linear systems. This kind of form is often referred to as a quasi LPV (qLPV) model. Further parameter dependent channels, which represent various control performance requirements, can be inserted into $\mathbf{S}(\mathbf{p}(t))$. The qLPV model (5) can then be defined using a TS fuzzy model as follows:

$$\begin{pmatrix} \dot{\mathbf{x}}(t) \\ \mathbf{y}(t) \end{pmatrix} = \mathcal{S} \boxtimes_{n=1}^{N} \mathbf{w}_n(p_n(t)) \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \end{pmatrix}. \tag{5}$$

The $N+2$-dimensional core tensor $\mathcal{S} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N\times L_1\times L_2}$ is constructed from the consequent system matrices ($K=2$), also known as LTI vertex system matrices $\mathbf{S}_{i_1,i_2,\ldots,i_N} \in \mathbb{R}^{L_1\times L_2}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

If we have Ruspini-partition for all n, then the transfer functions of the TS fuzzy model become a polytopic representation, and in consequence $\mathbf{S}(\mathbf{p}(t))$ is always within $co\{\forall n, i_n : \mathbf{S}_{i_1,i_2,\dots,i_N}\}$.

The advantage of this form is that a large set of LMI based system control design theories can immediately be applied to this kind of TS fuzzy model.

III. HOSVD and quasi HOSVD based canonical form of TS fuzzy models

This section redefines the previously published HOSVD based canonical form. It highlights the fact that the published HOSVD based canonical form is not unique when the dimensionality of the core tensor is larger than N, thus, it is quasi-unique only. In order to resolve this shortcoming, a "full" canonical form is proposed. The key idea is that the quasi HOSVD based canonical form is resulted when the HOSVD is executed only for dimensions assigned to the antecedent part, and "full" HOSVD is resulted when HOSVD is also executed on dimensions assigned to consequents and outputs. Since the HOSVD based canonical form determines the contribution (in decreasing order) of the fuzzy rules (in the sense of L_2 norm) via the higher order singular values (or via the singular values by dimensions) we can view this canonical form as a tool for main fuzzy rule component analysis.

Theorem 1 (HOSVD-based canonical form of TS models): For brevity one may say HOSVD of TS fuzzy models. A TS fuzzy model with output $\mathcal{Y}$ has the HOSVD canonical form:

$$\mathcal{Y} = \left(\left(\mathcal{S} \underset{n=1}{\overset{N}{\boxtimes}} \mathbf{w}_n(x_n) \right) \underset{N+(k=1)}{\overset{N+K}{\boxtimes}} \mathbf{T}^C_k \right) \times_{N+K+1} \mathbf{T}^O \quad (6)$$

in which

1) singular functions $w_{n,i_n}(x_n)$, $i_n = 1, \dots, I_n$ contained in singular vectors $\mathbf{w}_n(x_n)$ form an orthonormal system in the sense of
$$\forall n : \int_{min(\omega_n)}^{max(\omega_n)} w_{n,i}(x_n) w_{n,j}(x_n) dx_n = \delta_{i,j}, \quad (7)$$
where $1 \le i, j \le I_n$ and $\delta_{i,j}$ is the Kronecker-function ($\delta_{i,j} = 1$, if $i = j$ and $\delta_{i,j} = 0$, if $i \ne j$).
2) Transformation matrices $\mathbf{T}^C_{k=1\dots K}$ and $\mathbf{T}^O$ are unitary $(I_n \times I_n)$-matrices. Here, superscript "C" means "Consequent transformation" and "O" means "Output transformation".
3) Core tensor $\mathcal{S}$ is a real $(I_1 \times I_2 \times \cdots \times I_{N+K+1})$-tensor. Its subtensors $\mathcal{S}_{i_n=\alpha}$ (which can be obtained by fixing the nth index to α) have the properties of
   a) all-orthogonality: two subtensors $\mathcal{S}_{i_n=\alpha}$ and $\mathcal{S}_{i_n=\beta}$ are orthogonal for all possible values of n, α and β subject to $\alpha \ne \beta$: $\langle \mathcal{S}_{i_n=\alpha} \mathcal{S}_{i_n=\beta} \rangle = 0$, when $\alpha \ne \beta$,
   b) ordering: $\|\mathcal{S}_{i_n=1}\| \ge \|\mathcal{S}_{i_n=2}\| \ge \cdots \ge \|\mathcal{S}_{i_n=I_n}\| \ge 0$, for all possible values of n.

Based on the analogy of the HOSVD of tensors, we refer to the Frobenius-norms $\|\mathcal{S}_{i_n=i}\|$, symbolized by $\sigma_i^{(n)}$, as the n-mode singular values of the TS fuzzy model.

Proof 1: The proof of the existence of the HOSVD based canonical form in cases where a TS fuzzy model has a single scalar output is given in [27]. The existence of the HOSVD based canonical form for multi-output and / or non-scalar functions can be proved in the same way, the only difference being that in dimensions larger than N, we simply have the original tensor HOSVD.

Proof 2: The proof of uniqueness of the HOSVD based canonical form in cases where a TS fuzzy model has a single scalar output is given in [27]. The uniqueness of the HOSVD based canonical form for general cases can be proved in the same way, the only difference being that in dimensions larger than N of the core tensor, the HOSVD itself guarantees the unique decomposition [57]. Thus, this property of the HOSVD of tensors is true for the HOSVD based canonical form as well. As a matter of fact, the decomposition is unique to the extent of the signs of the singular functions and the columns of the transformation matrices, which can be systematically switched, just like in the case of HOSVD of tensors. If there are equal singular values on any dimension, then the HOSVD based canonical form is not unique. In this case the n-mode singular functions or vectors corresponding to the same n-mode singular value can be replaced by orthonormal linear combinations. This property is proved in the original paper on the HOSVD concept itself, see [57].

Remark 1: Transformation matrix $\mathbf{T}^C_{k=N+1,\dots,N+K}$ transforms the consequent tensors to the minimal $R_{N+1} \times R_{N+2} \times \dots \times R_{N+K}$ orthonormal subspace that is structured via higher order singular values. This transformation indicates whether or not the consequent tensors have linear dependencies. Thus we may deal only with the linearly independent tensors, and once we have the canonical conclusions, we can restructure the expected tensor. T^O defines the minimal number of linearly independent outputs structured by higher order singular values. Once we have computed the minimal number of outputs, we can simply reconstruct the originally expected outputs using $\mathbf{T}^O$.

Remark 2: Note that the HOSVD based canonical form is not a real TS fuzzy model as its antecedent membership functions may assume negative values.

Theorem 2 (Quasi HOSVD-based canonical form): If we do not execute SVD on dimensions $n > N$, we arrive at the quasi HOSVD canonical form. Specifically, if we multiply by $\mathbf{T}^C$ and $\mathbf{T}^O$ in the HOSVD canonical form (6), then we arrive at the quasi HOSVD canonical form:

$$\mathcal{Y} = \mathcal{S} \underset{k=1}{\overset{N}{\boxtimes}} \mathbf{w}_n(x_n). \quad (8)$$

Proof 3: In case of the quasi-HOSVD based canonical form (when the number of the dimensions of the core tensor is larger than N), the unique representation is not guaranteed since not all dimensions are decomposed by SVD, which would be the step that can guarantee unique decomposition.

Note that even in case we use quasi-HOSVD based canonical forms, where uniqueness is not guaranteed, the decomposition still gives the higher order ranking of the fuzzy rules for main fuzzy rule component analysis.

Definition 2 (n-mode rank of TS fuzzy model): The n-mode rank of a TS fuzzy model, where $\mathbf{x} \in \Omega$, denoted by $R_n = rank_n(f(\mathbf{x}), \Omega)$ is the number of non-zero singular values in the n-th dimension, thus $R_n = rank_n(f(\mathbf{x}), \Omega) = rank_n(\mathcal{S})$. We can also indicate the rank of the consequent parameters in dimensions $n = N+1, \ldots, N+K$ and for the outputs if $n = N+K+1$.

Definition 3 (CHOSVD/RHOSVD-based canonical form): This definition is about the complexity trade-off property of the HOSVD. We arrive at the Compact HOSVD (CHOSVD) of TS fuzzy models, when we keep the first R_n, $n = 1, \ldots, N+K+1$ singular values only in all dimensions. Accordingly, the size of the core tensor is $R_1 \times R_2 \times \ldots \times R_{N+K+1}$, where R_n is the n-mode rank of the TS fuzzy model. We have Rank reduced HOSVD (RHOSVD), when we keep $J_1 \times J_2 \times \ldots \times J_N$ nonzero singular values only, where $\forall n : J_n \leq R_n$ and $\exists n : J_n < R_n$. The RHOSVD canonical form of TS fuzzy models is only an approximation, where the error (in L_2 norm) is bounded by the sum of the discarded singular values as in the case of the HOSVD of tensors (see the proof in [57]). Using HOOI we can further tune the core tensor to decrease the error [58], [59]. A comprehensive analysis on the approximation properties are given in [25], [26].

IV. Generalized TP model transformation

A. Numerical reconstruction of the HOSVD / qHOSVD of TS fuzzy models

This section recalls the TP model transformation from [1]–[3] and restructures it, in order to have a core algorithm that can readily involve further extensions to be introduced in the next subsections. This section also discuses the bi-linear TP model transformation, which has already been used in practical cases, but has not yet been formally introduced.

Definition 4 (discretization space Ω): $\Omega = \omega_1 \times \omega_2 \times \cdots \times \omega_N$ is a space in which we intend to perform the discretization of a given TS fuzzy model $f(\mathbf{x})$.

Definition 5 (Discretization grid M fit to Ω): Let N-dimensional $M_1 \times M_2 \times \ldots \times M_N$ sized hyper rectangular discretization grid M be defined by vectors

$$\mathbf{g}_n = \begin{pmatrix} g_{n,1} & \cdots & g_{n,m_n} & \cdots & g_{n,M_n} \end{pmatrix}^T \in \mathbb{R}^{M_n} \quad (9)$$

assigned to the dimensions $n = 1, \ldots, N$, where $\omega_n = [g_{n,1}, g_{n,M_n}]$. M_n denotes the number of grids on dimension n. The elements of these vectors define the points of M as $\mathbf{p}_{m_1,\ldots m_n,\ldots m_N} = \begin{pmatrix} g_{1,m_1} & \cdots & g_{n,m_n} & \cdots & g_{N,m_N} \end{pmatrix}^T \in \Omega$.

Definition 6: (Discretised function). The discretized form of function $y = f(\mathbf{x})$, $\mathbf{x} \in \Omega$, over discretization grid M is denoted by $\mathfrak{F}^{D(M,\Omega)}$ (where superscript "D" is an abbreviation for "discretized"). $\mathfrak{F}^{D(M,\Omega)} \in \mathbb{R}^{M_1 \times M_2 \times \ldots \times M_N}$ contains elements $a_{m_1,\ldots,m_n,\ldots,m_N} = f(\mathbf{p}_{m_1,\ldots,m_n,\ldots,m_N})$.

Lemma 1: The discretization of a given $f(\mathbf{x}) = B \mathop{\boxtimes}_{n=1}^{N} \mathbf{w}_n(x_n)$ simplifies to the discretization of the weighting functions $w_{n,i}(x_n)$ as $\mathfrak{F}^{D(M,\Omega)} = B \mathop{\boxtimes}_{n=1}^{N} \mathfrak{W}_n^{D(M_n,\omega_n)}$, where

$$\mathfrak{W}_n^{D(M_n,\omega_n)} = \begin{pmatrix} \mathfrak{w}_{n,1}^{D(M_n,\omega_n)} & \cdots & \mathfrak{w}_{n,I_n}^{D(M_n,\omega_n)} \end{pmatrix}, \quad (10)$$

where $\mathfrak{W}_n^{D(M_n,\omega_n)} \in \mathbb{R}^{M_n \times I_n}$ and are constructed from discretized functions as:

$$\mathfrak{w}_{n,i_n}^{D(M_n,\omega_n)} = \begin{pmatrix} w_{n,i_n}(g_{n,1}) & \cdots & w_{n,i_n}(g_{n,M_n}) \end{pmatrix}^T, \quad (11)$$

where $i_n = 1 \ldots I_n$.

Note that the result of HOSVD has the same structure as the discretized TS fuzzy models. Thus, the key idea is that executing HOSVD on the discretized function, we obtain the discretized form of the HOSVD based canonical form of the TS fuzzy model:

Algorithm 1: (TP model transformation) Assume a given TS fuzzy model $f(\mathbf{x}) = \mathcal{B} \mathop{\boxtimes}_{n=1}^{N} \mathbf{v}_n(x_n)$, $\mathbf{x} \in \mathbb{R}^N$. The goal of the algorithm is to numerically reconstruct the HOSVD/CHOSVD canonical form

$$\mathcal{Y} = \left(\left(\mathcal{S} \mathop{\boxtimes}_{n=1}^{N} \mathbf{w}_n(x_n) \right) \mathop{\boxtimes}_{N+(k=1)}^{N+K} \mathbf{T}_k^C \right) \times_{N+K+1} \mathbf{T}^O \quad (12)$$

in Ω:

- STEP 0: Numerical initialization: Define discretisation grid M by fitting $M_1 \times M_2 \times \ldots \times M_N$ gridpoints to Ω.
- STEP 1: Discretisation: Determine $\mathfrak{F}^{D(M,\Omega)}$.
- STEP 2: Reconstruct the core of the model: Determine $\mathcal{S}$ and $\mathbf{U}_n$ by executing HOSVD, CHOSVD on $\mathfrak{F}^{D(M,\Omega)}$ (in case of rank reduction or complexity trade-off RHOSVD is executed in this step). This results in $\mathfrak{F}^{D(M,\Omega)} = \hat{\mathcal{S}} \mathop{\boxtimes}_{n=1}^{N+K+1} \mathbf{U}_n$. Thus $\mathbf{T}_k^C = \mathbf{U}_{N+k}$, $k = 1 \ldots K$ and $\mathbf{T}^O = \mathbf{U}_{N+K+1}$.
- STEP 3: Determine $\mathbf{w}_n(x_n)$: Let $\hat{\mathfrak{W}}_n^{D(M_n,\omega_n)} = \mathbf{U}_n$. Antecedent membership functions $\hat{\mathbf{w}}_n(x_n)$ of

$$\mathcal{Y} = \left(\left(\hat{\mathcal{S}} \mathop{\boxtimes}_{n=1}^{N} \hat{\mathbf{w}}_n(x_n) \right) \mathop{\boxtimes}_{N+(k=1)}^{N+K} \mathbf{T}_k^C \right) \times_{N+K+1} \mathbf{T}^O \quad (13)$$

can be reconstructed over any point in ω_n. For instance, let us calculate the antecedent membership functions $\hat{\mathbf{w}}_d(x_d)$ on dimension d over a given point x_d. Let us define a new discretisation grid M' as $M_1 \times \ldots \times M_{d-1} \times 1 \times M_{d+1} \times \ldots \times M_N$ and restrict the discretization space to x_d as $\Omega' = \omega_1 \times \ldots \times \omega_{d-1} \times x_d \times \omega_{d+1} \times \ldots \times \omega_N$, then define $\mathfrak{F}^{D(M',\Omega')}$. Then for x_d:

$$\hat{\mathbf{w}}_d(x_d) = \mathfrak{F}_{(d)}^{D(M',\Omega')} \left(\mathcal{Q}_{(d)} \right)^+ . \quad (14)$$

where

$$\mathcal{Q} = \left(\hat{\mathcal{S}} \mathop{\boxtimes}_{\substack{n=1 \\ n \neq d}}^{N} \mathbf{U}_n(x_n) \right) \mathop{\boxtimes}_{N+(k=1)}^{N+K} \mathbf{T}_k^C \times_{N+K+1} \mathbf{T}^O. \quad (15)$$

lower case "$()_{(d)}$" denoted the unfolding of that dimension, see the works of Lathauwer about HOSVD.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

- STEP +1: Transformation error: This step is a numerical checking of the accuracy of the resulting TS fuzzy model over a huge number of random points in Ω.

Proof 4: Szeidl et al. [29] proves that the TP model transformation numerically reconstructs the HOSVD canonical form in case of single scalar output functions, namely if $M_n \to \infty$ then $\hat{\mathcal{S}} \to \mathcal{S}$ and $\mathbf{U}_n \to \mathfrak{W}_n^{D(M_n,\omega_n)}$. If we consider matrices T^C and T^O resulting from SVD in the same way as matrices U_n are considered for $n = 1 \ldots N$ in the proof presented in paper [29] (but without transforming them to functions), we arrive at a proof of the claim that the TP model transformation numerically reconstructs the HOSVD based canonical form, as well as a quasi-HOSVD based canonical form of multiple output and/or non-scalar functions when we multiple by T^C and T^O (see Theorem 2). Paper [29] also gives various theorems for the speed of the convergence for the numerical reconstruction depending on whether we use equidistant or non-equidistant rectangular grids for discretization.

Remark 3: The numerical implementation limits the grid density as $\forall n = 1, \ldots, N : M_n \to M_n^{max} < \infty$. Furthermore, the computational load of HOSVD can easily explode as M_n and N grow larger. These factors form the bottlenecks of this algorithm. Thus, we say that the TP model transformation numerically reconstructs quasi-$\mathcal{S}$. Papers [23], [30] propose very effective computational complexity reduction techniques for the TP model transformation.

Remark 4: The paper of Szeidl et al. also derives theorems for the smallest grid density necessary for finding all the ranks of the TS fuzzy model, thus, the discretization density should be set according to Szeidl's theorems and based on the fact that the maximum rank is determined by the number of the antecedent membership functions by dimensions of the given TS fuzzy model. If we do not see the structure of the given TS fuzzy model, or the given model is not a TS fuzzy model (see later), then we can practically use a grid with the highest density made possible by the numerical implementation.

Remark 5: If the density of the discretisation grid is not sufficiently high to find the rank of the given TS fuzzy model, then the TP model transformation results in an approximation only. In this case the transformation works like in the case of given non-TS fuzzy models (see details later and [25], [26]).

In various engineering cases we have different accuracy requirements of different components of the TS fuzzy model, hence, it is not always necessary to find all points of antecedent membership functions in Step 3. For instance, in case of robust control design the precise core tensor $\mathcal{S}$ is important as much as the control design is based on it, however, in the final implementation of the TS fuzzy controller, we can accept a good piece-wise approximation of the antecedent membership functions. This leads to a practically useful engineering implementation, where we simply use their piece-wise linear variant of the antecedent sets in the controller:

Definition 7: (Piece-wise linear function system denoted by $\bar{\mathbf{w}}(x)$) Function $\bar{\mathbf{w}}(x)$ is defined by matrix $\mathbf{U}$ and grid M over $x \in \omega$ in such a way that $\mathbf{U} = \mathfrak{W}^{D(M,\omega)}$. A linear interpolation between neighboring values of each column of $\mathbf{U}$ fullly defines the piece-wise linear functions.

Algorithm 2: (Bi-linear TP model transformation) The Bi-linear TP model transformation results in a bi-linear approximation $f(\mathbf{x}) \approx \hat{\mathcal{S}} \boxtimes_{n=1}^{N} \bar{\mathbf{w}}_n(x_n)$ of the given function fit to a given grid M. It differs only in Step 3 as:

STEP 3: $\bar{\mathbf{w}}_n(x_n)$ is directly defined by $\mathbf{U}_n$ and grid M.

Remark 6: If the grid density is sufficient to find the precise core tensor, but is too sparse to determine good membership functions $\bar{\mathbf{w}}_n(x_n)$, then we may combine the third steps of the TP model transformation and the bi-linear TP model transformation. Step 3 of the TP model transformation does not require the execution of HOSVD. Only the available memory limits the off-line storage of a number of points of $\hat{\mathbf{w}}_n(x_n)$ in Step 3, which can readily be calculated over any $\mathbf{x}$. Therefore, we may simply determine H_n new gridpoints on dimension n in Step 3 of the TP model transformation, where H_n can be considerably larger than M_n^{max}, and we determine $\hat{\mathfrak{W}}_n^{D(H_n,\omega_n)} = U_n$ for Step 3 of the Bi-linear TP model transformation that leads to better resolution of $\bar{\mathbf{w}}_n(x_n)$.

B. Convex TP model transformation - Incorporating Ruspini-partition

The goal is to transform the given model to a convex TS fuzzy model $\mathcal{S} \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$ where the antecedent membership functions are in Ruspini-partitions. This section focuses on the step where the antecedents of the TS fuzzy model can be manipulated and where the already published convex hull generation methods can be inserted in the algorithm of the TP model transformation.

Algorithm 3: (Convex TP model transformation) Assume a given TS fuzzy model $f(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^N$. The goal is to numerically reconstruct TS fuzzy model $f(\mathbf{x}) = \mathcal{S} \boxtimes_{n=1}^{N} \mathbf{w}_n^{RP}(x_n)$ in $\mathbf{x} \in \Omega$ and include complexity trade-off if needed. The steps of this transformation are the same as in the TP model transformation. Only Step 2 is extended by the convex hull generation. We also add Step +2 to the algorithm to be executed after Step 3.

- STEP 2: Reconstruct the core of the TP structure: Determine $\mathcal{S}$ and $\mathbf{U}_n$, then use the SN and NN transformation introduced by Yam in [5], [19], which transform $\mathbf{U}_n$ to $\mathbf{U}_n^{RP}$, and define $\mathcal{S}$ for $\mathfrak{F}^{D(M,\Omega)} = \mathcal{S} \boxtimes_{n=1}^{N} \mathbf{U}_n^{RP}$.
- STEP +2: The antecedent membership functions satisfy the Ruspini-partition criteria over the grid, however one has to check this between gridpoints. Obviously, in case of bilinear TP model transformation the Ruspini-partition is guaranteed for all $x_n \in \Omega$.

Further types of convex TS fuzzy models can be generated by using normalized, close to normalized, inverse normaized etc. transformations in the same way as SN and NN transformation is used in Step 2, see for instance [7], [8], [17]–[19].

C. Pseudo TP model transformation

We may want to find an equivalent TS fuzzy model with a predefined antecedent membership function system, namely by transforming a given TS fuzzy model to an other TS fuzzy

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

model with given antecedent membership functions. For such purposes, we propose the pseudo TP model transformation as follows:

Algorithm 4: (Pseudo TP model transformation, TP$^+$ model transformation for short) Assume a given TS fuzzy model $y = f(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^N$ and antecedent membership function system $\mathbf{w}_n(x_n)$, $\mathbf{x} \in \Omega$. The goal is to determine $\mathcal{S}$ such that $f(\mathbf{x}) = \mathcal{S} \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$, or if this is not possible, then the goal is to find $\hat{f}(\mathbf{x}) = \mathcal{S} \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$, where $\hat{f}(\mathbf{x})$ is the best or at least a good approximation under the rank constraints implicitly given by $\mathbf{w}_n(x_n)$ (e.g. the number of linearly independent new antecedent membership functions may be less in dimension n than $rank_n(f(\mathbf{x}))$). Steps 0 and +1 are the same as in the TP model transformation.

- STEP 1: Discretization: Determine $\mathfrak{F}^{D(M,\Omega)}$ and $\mathfrak{W}_n^{D(M_n,\omega_n)}$.
- STEP 2: Determine the core tensor:

$$\mathcal{S} = \mathfrak{F}^{D(M,\Omega)} \boxtimes_{n=1}^{N} \left(\mathfrak{W}_n^{D(M_n,\omega_n)}\right)^+. \tag{16}$$

If $\mathfrak{W}_n^{D(M_n,\omega_n)}$ introduces rank reduction here then we arrive at $\hat{f}(\mathbf{x}) = \mathcal{S} \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$. This works like in the case of complexity trade-off via TP model transformation. If we have predefined transformations $\mathbf{T}^C$ or $\mathbf{T}^O$ then:

$$\mathcal{S} = \left(\mathcal{Q} \boxtimes_{N+(k=1)}^{N+K} \left(\mathbf{T}_k^C\right)^+\right) \left(\mathbf{T}^O\right)^+, \tag{17}$$

where

$$\mathcal{Q} = \mathfrak{F}^{D(M,\Omega)} \boxtimes_{n=1}^{N} \left(\mathfrak{W}_n^{D(M_n,\omega_n)}\right)^+. \tag{18}$$

- STEP +2: Checking the antecedent membership functions: Once we have the core tensor $\mathcal{S}$ we may recalculate the antecedent membership functions between the points of $\mathfrak{W}_n^{D(M_n,\omega_n)}$ through Step 3 and compare to the given $\mathbf{w}_n(x_n)$.

Algorithm 5: (Partial TP$^+$ model transformation) Assume a given TS fuzzy model $y = f(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^N$. Further, assume a given antecedent membership function system $\mathbf{w}_d(x_d)$, $\mathbf{x} \in \Omega$, $d \in D$. The goal is to determine $\mathcal{S}$ such that $f(\mathbf{x}) = \mathcal{S} \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$, where antecedent membership functions $\mathbf{w}_n(x_n)$, $n \notin D$, are the same as in the case of the TP model transformation. If this is not possible, or if we need a complexity trade-off, then the goal is to find $\hat{f}(\mathbf{x}) = \mathcal{S} \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$, where $\hat{f}(\mathbf{x})$ is the best or at least a good approximation under the rank constraint implicitly given by $\mathbf{w}_n(x_n)$, $n \notin D$. Steps 0,1,3,+1 and +2 are the same as in the case of the TP$^+$ model transformation:

STEP 2: Determine the core tensor as $\mathcal{K} = \mathfrak{F}^{D(M,\Omega)} \boxtimes_{d\in D} \left(\mathfrak{W}_d^{D(M_d,\omega_d)}\right)^+$. Execute HOSVD on $\mathcal{K}$ in all dimensions except $n \notin D$ to obtain:

$$\mathcal{K} = \mathcal{S} \boxtimes_{\substack{n=1 \\ n\notin D}}^{N} \mathbf{U}_n. \tag{19}$$

Let $\mathfrak{W}_n^{D(M_n,\omega_n)} = \mathbf{U}_n$, $n \notin D$, in which case:

$$\mathfrak{F}^{D(M,\Omega)} = \left(\mathcal{S} \boxtimes_{\substack{n=1 \\ n\notin D}}^{N} \mathbf{U}_n\right) \boxtimes_{d\in D} \mathfrak{W}_d^{D(M_d,\omega_d)} = \tag{20}$$

$$= \mathcal{S} \boxtimes_{n=1}^{N} \mathfrak{W}^{D(M_n,\omega_n)_n}. \tag{21}$$

D. Multi TP model transformation

We may want to transform a set of TS fuzzy models simultaneously to the same antecedent system.

Algorithm 6: (Multi TP model transformation) Assume that we have parameter dependent scalar, vector, matrix or tensor functions $\mathbf{S}_k(\mathbf{x})$, $k = 1..K$, $\mathbf{x} \in \Omega$ (an important property is that they may have different sizes). The goal is to find their TS fuzzy model representations over the same antecedent membership function system $\mathbf{S}_k(\mathbf{x}) = \mathcal{S}_k \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$:

- STEP 0): Define discretization grid M fit to Ω.
- STEP 1): Discretization: store all the elements of $\mathbf{S}_k(\mathbf{x})$ in a vector $f(\mathbf{x}) = \begin{pmatrix} h_1(\mathbf{x}) & h_2(\mathbf{x}) & \dots & h_Z(\mathbf{x}) \end{pmatrix}$ or directly the discretized values according to ordering of this vector in $N+1$ dimensional tensor $\mathfrak{F}^{D(M,\Omega)}$ of size $M_1 \times M_2 \times \ldots \times M_N \times Z$.
- STEP 2-3: These two steps are the same as in the case of the TP model transformation (including trade-off and convex manipulation etc). As a result we have $f(\mathbf{x}) = \mathcal{B} \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$, where $\mathcal{B}$ is $N+1$ dimensional.
- STEP 4: By repartitioning tensor $\mathcal{B}$ in the $N+1$-th dimension by elements into tensors $\mathcal{H}_z$, we have $h_z(\mathbf{x}) = \mathcal{H}_z \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$. Finally, we can reconstruct $\mathcal{S}_k$ from tensors $\mathcal{H}_z$ according to the ordering of vector in Step 1, which leads to $\mathbf{S}_k(\mathbf{x}) = \mathcal{S}_k \boxtimes_{n=1}^{N} \mathbf{w}_n(x_n)$.
- STEP +1 and +2) These steps have the same error checking role as in the case of the TP model transformation.

E. Summary

The previous sections proposed various TP model transformations. The variants essentially have differences in Steps 2 and 3. Step 2 is responsible for determining HOSVD, convex forms or executing pseudo transformations and complexity trade-offs. Step 3 is executed depending on whether or not we need a Bi-linear TS fuzzy model form. In Step 2, the core tensor and the discretized variants of the expected antecedent membership functions are manipulated by dimensions. Thus, we may combine the above TP model transformations and quickly find very flexible variations that are suitable to our needs. All these combinations can be carried through the use of the Multi TP model transformation, with the key idea of constructing a vector function.

For instance, we could define the bi-linear convex multi partial pseudo TP model transformation, in which case Step 2 would consist in either keeping the orthonormal system or executing SN and NN or other transformations on various dimensions (we could also use the pseudo inverse based product for dimensions where we have predefined antecedent membership functions), while Step 3 would consist in selecting the direct determination of piece-wise linear antecedent membership functions on the selected dimensions. Finally, we would obtain a set of TS fuzzy models where the antecedent membership functions are the same.

V. GATEWAY TO TS FUZZY MODELS AND GENERLIZED STABILITY VERIFICATION

The TP model transformation works even in case the rules and the entire TS model structure of the given function or model are hidden. The only requirement of the presented algorithms is that the model given at hand should be discretizable. In case of models which have TS fuzzy model structure (with bounded number of components) once we find all the ranks through the TP model transformation, then irrespective of how many extra gridpoints we add to the discretization, the number of the nonzero singular values will not increase when HOSVD is executed. If we have a model that has no TS fuzzy model representation (with bounded number of components) then the rank of the discretized tensor will increase (at least in one dimension) with the density, such that the rank will always be M_n. Since the computational power available limits M_n, when using the TP model transformation in engineering applications, it is irrelevant whether the given model is a TS fuzzy model with a higher rank than M_n, or if it is a model that does not have an exact TS fuzzy model representation. We are faced with the same uncertainty if we have a limitation on the number of resulting antecedent membership functions and we have to execute RHOSVD in any case. If we find that the given function and the resulting TS fuzzy model are equivalent in a numerical sense, then we may suppose that we have found all the ranks. Therefore, it should be kept in mind that in a mathematical sense, we are always dealing with approximations unless we perform further analysis, however, in engineering the possible cases will be numerically equivalent.

A. *TP model transformation as a gateway to TS fuzzy models*

Identification and modeling techniques based on fuzzy theory, neural networks, genetic algorithms or any combination of these approaches (referred to as soft-computing techniques) are extremely powerful in solving modern model identification engineering tasks, especially in cases where the derivation of closed formulae through the consideration of physical and engineering laws would prove to be much too difficult. As a result, a number of different identification techniques have emerged. Due to differences in the structure and many times unique and problem-dependent representation of the identification techniques, it is not trivial in these cases how we can continue the identification phase with the well developed system design frameworks.

The goal of having a gateway to TS fuzzy models is motivated by the fact that TS fuzzy model based design is well developed, has various frameworks and is widely adopted approach towards finding routine-like solutions to engineering problems. Further, the structure of the TS fuzzy model and its transfer function is well-adapted to the polytopic and LMI based modern control theories, which means that once we have a TS fuzzy model representation of a problem, we can almost directly use the mathematical approaches of convex optimisation and modern control design theories. In this regards, the reader is referred to the early papers of GAHINET, BOKOR, CHILAI, BOYD, and APKARIAN, see [60]–[69], who pioneered the polytopic model and LMI based design; and also to the works of Sheerer, Balas, Pakard discussing the whole structure of these control design concepts [63], [70], [71]. From the fuzzy theory side, the reader is referred to the early papers and book of Tanaka et al. about Parallel Distributed Compensation (PDC) design [72]–[75], [76], which clearly shows the conceptual similarities and common points of these theories.

The measurement based identification of given functions and systems, or derivation of the model via physical considerations typically (with the exception of some special cases, we always make simplifying assumptions here) entails considerably larger errors than the TP model transformation, thus we can execute the TP model transformation in engineering cases without checking whether the result of the identification has a TS fuzzy model structure or not, in order to reveal the TS fuzzy structure or propose a TS fuzzy model to be validated. If the validation is positive, then we accept the TS fuzzy model as the output of the identification, irrespective of the kinds of identification applied. Note that in many cases the relatively small (but non-zero) singular values may actually represent noise in the identification process, and hence, RHOSVD can be considered in the TP model transformation as a noise filtering step of the identification. After this, the TP model transformation to various further types of TS fuzzy models would be exact, which is an important requirement in e.g. control design.

In conclusion **the TP model transformation could be a final step of the identification, and as a generalized "interface" could be, simultaneously, a preprocessing step for further design requirements** (e.g. convex hull manipulation).

B. *Generalized TS Fuzzy model based stability verification*

If a system has various different components given in different representations (i.e. the model is given by equations, the controller by fuzzy logic rules and the observer by neural network) - we say hybrid representations - then the stability proof is extremely difficult to derive. In this case **the Multi TP model transformation can be used to find the convex TS fuzzy model representation of all components, such that all TS fuzzy models have the same antecedent system, in order to apply LMI based stability or performance analysis in a straightforward way.** As a matter of fact if there is no exact TS fuzzy model representation for a component, then the complexity trade-off should be carefully executed while considering the best achievable accuracy and/or performing validation as discussed in the previous subsection.

VI. EXAMPLES

A. *Convex hull manipulation via TP$^+$ model transformation*

This example shows how the TP$^+$ model transformation can be used to interpolate between different exact convex TS

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

fuzzy model representations of a given model. The example of this section starts with an HOSVD based canonical from (e.g. transformed from the result of an identification) of a very simple qLPV state-space model. The example goes through a typical PDC design strategy, where the controller and observer are searched for in TS fuzzy model form and their consequent vertices are designed via LMIs from the consequent vertices of the derived TS fuzzy model. The example uses the TP$^+$ model transformation to systematically interpolate between the antecedent systems, hence, the convex hull of the convex TS fuzzy model representations of the given model for the design, and will show that the LMI based design leads to considerably different solutions or in many cases the LMIs are not feasible. Hence, this example also demonstrates that the LMI based design is very sensitive to convex hull manipulation in full agreement with paper [17] in which the necessity of convex hull manipulation in LMI-based control design is proved. This example reveals an additional novelty, namely that if different convex TS fuzzy models are applied for controller and observer design (if the separation theory is applicable) then we can further improve the resulting control performance. Since the example is very simple we can drop the tensor notation.

1) The given TS fuzzy model: Assume the following TS fuzzy model $\mathbf{S}(p(t)) = \sum_{r=1}^{3} w_r(p(t))\mathbf{S}_r$ given in HOSVD based canonical form, and assume that it has been validated for the identified qLPV system:

$$\begin{pmatrix}\dot{\mathbf{x}}(t)\\ \mathbf{y}(t)\end{pmatrix} = \mathbf{S}(\mathbf{p}(t))\begin{pmatrix}\mathbf{x}(t)\\ \mathbf{u}(t)\end{pmatrix}, \tag{22}$$

where $p(t) \in \Omega = [0, 0.04]$. The system matrices are:

$$\mathbf{S}_1 = 10^4\begin{pmatrix}1.1487 & 1.1924 & 0.0009\\ -1.1495 & -1.1915 & -0.0009\end{pmatrix}$$

$$\mathbf{S}_2 = 10^4\begin{pmatrix}-134.3021 & 129.5591 & 2.1529\\ 132.1521 & -127.3150 & -2.1495\end{pmatrix}$$

$$\mathbf{S}_3 = 10^4\begin{pmatrix}-0.2175 & 0.2135 & 0.4384\\ -0.2281 & 0.2247 & -0.4379\end{pmatrix}$$

and the assigned antecedent membership functions are shown on Fig. 1.

2) Manipulation of the Convex TS fuzzy model: Let us execute the convex TP model transformation with SN, NN transformation (in Step 2) introduced by Yam. This leads to a Ruspini-partitioned antecedent function system that defines the vertices to form a convex hull around the given qLPV model. We also execute the CNO transformation that leads to an antecedent system such that the vertices form a tight convex hull around the given model, for more details about SN, NN and CNO transformations see [3], [5], [8], [18], [19]:

$$\mathbf{S}(p(t)) = \sum_{r=1}^{3} w_r^{SNN}(p(t))S_r^{SNN} = \sum_{r=1}^{3} w_r^{CNO}(p(t))S_r^{CNO}. \tag{23}$$

The antecedent membership functions are given in Fig. 2 and 3 .

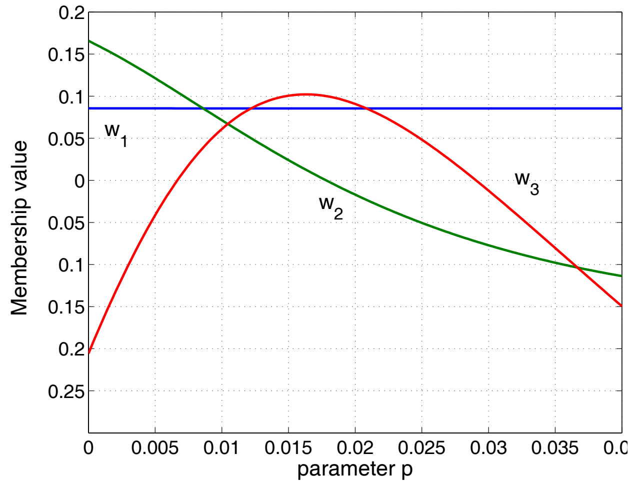

Fig. 1. Antecedent membership functions of the HOSVD based canonical form

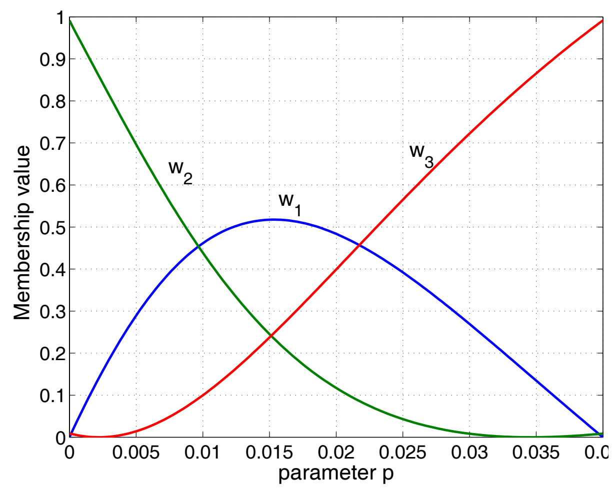

Fig. 2. CNO type antecedent membership function system

In the next step the TP$^+$ model transformation is executed. We define the linear interpolation between the antecedent membership function systems for all $p(t) \in \Omega$ such that:

$$\mathbf{w}_n^{\lambda}(p(t)) = \lambda\mathbf{w}_n^{CNO}(p(t)) + (1-\lambda)\mathbf{w}_n^{SNN}(p(t)), \tag{24}$$

where $\lambda \in [0, 1]$. This means that we are tightening the convex hull with λ. Then, using the TP$^+$ model transformation, we determine the interpolated and exact TS fuzzy model such that:

$$\mathbf{S}(\mathbf{p}(t)) = \sum_{r=1}^{3} w_r^{\lambda}(p(t))S_r^{\lambda}. \tag{25}$$

3) Control design: In order to provide an example of how the convex hull of the vertices influences the resulting control, we simply enlarge and tighten the convex hull by tuning λ. We search for the controller and the observer in the following structure:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

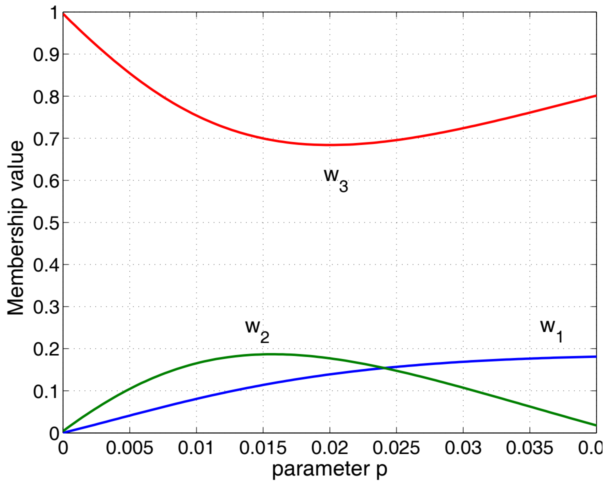

Fig. 3. SNNN type antecedent membership function system

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}(\mathbf{p}(t))\hat{\mathbf{x}}(t) + \mathbf{B}(\mathbf{p}(t))\mathbf{u}(t) + \mathbf{K}(\mathbf{p}(t))(\mathbf{y}(t) - \hat{\mathbf{y}}(t)) \tag{26}$$

$$\hat{\mathbf{y}}(t) = \mathbf{C}(\mathbf{p}(t))\hat{\mathbf{x}}(t) \quad \text{and} \quad \mathbf{u}(t) = -\mathbf{F}(\mathbf{p}(t))\hat{\mathbf{x}}(t). \tag{27}$$

Thus, the goal of the design is to find the vertices of the observer and controller as follows:

$$\mathbf{F}(\mathbf{p}(t)) = \sum_{r=1}^{R} w_r^{\lambda}(\mathbf{p}(t))\mathbf{F}_r^{\lambda} \tag{28}$$

and

$$\mathbf{K}(\mathbf{p}(t)) = \sum_{r=1}^{R} w_r^{\lambda}(\mathbf{p}(t))\mathbf{K}_r^{\lambda}. \tag{29}$$

Let us apply a very simple LMIs taken from [7], [8] and substitute the model vertices $\mathbf{S}_r^{\lambda}$ to determine $\mathbf{F}_r^{\lambda}$ and $\mathbf{K}_r^{\lambda}$. Let us investigate the results at $z = 1, \ldots, Z = 31$ equidistantly located points of λ within $[0, 1]$. We focus on the comparison of the control performance of the controllers resulting for all z in terms of the maximum control value and the stabilization time. First we focus only on the state feedback controller design.

The first conclusion is that the LMIs are not feasible in the range of $z = 1, \ldots, 20$. All of these Ruspini partitioned TS fuzzy models are not good representations for the present LMI based design. When we further tighten the convex hull for $z = 21, \ldots, 31$, we find that the LMIs are feasible, which means that we can determine the TS fuzzy controller. When we compare the resulting control performance for these controllers we find that the stabilization time is almost the same, however the maximum control value (which reflects on the conservativeness from this aspect) really depends on the applied convex hull. The maximum control value decreases as we get closer to the CNO type tight convex hull, see Fig. 4 (control value 0 on Fig. 4 means that there is no feasible solution). $z_{controller}$ on the Fig. 4 refers to the interpolated model $\mathbf{S}_r^{\lambda}$ which is used for the controller design. The figure shows the maximum of the control values for the same $\mathbf{x}(0)$, and one can see that when z increases to 31, in other words λ increases to 1 (which means that we are using increasingly tight convex hulls), then the maximum of the control value shows a drastically decreasing tendency. In conclusion, the results demonstrate the theoretical statements of [17] and clearly show that the manipulation of the antecedent fuzzy sets is necessary in control design, because the LMI optimizes the solution for the given TS fuzzy model representation only, but not for the given system.

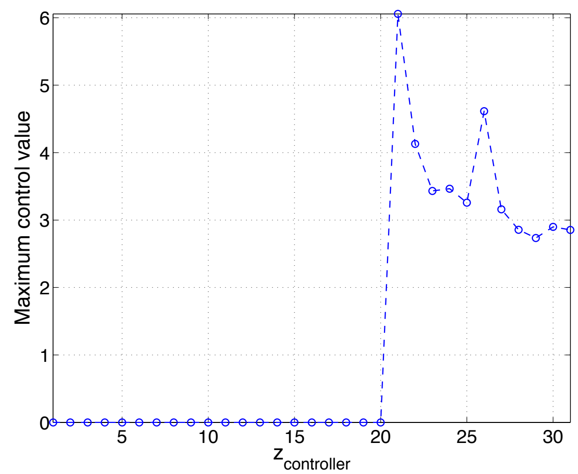

Fig. 4. The maximum of the control value for a given convex hull type (for the same $\mathbf{x}(0)$)

When we include the observer design, we find that the LMIs are not feasible in the range $z = 1, \ldots, 14$ (an interesting point here is that the observer has decreased the infeasible range). Moving between $z = 15, \ldots, 30$ the LMIs are feasible, and we find that at $z = 16$ we have the "best" result with regards to our comparison objectives, namely the smallest maximum of the control value, see Fig. 6. The stabilization time is almost the same in all cases. $z_{controller,obsrever}$ means that we use the same TP model representation (i.e. the same convex hull determined by the antecedent membership functions) for both the controller and observer design. Thus, the conclusion is that the tightest hull does not necessarily lead to the best solution. The antecedent sets at $z = 16$ are depicted in Fig. 5. The vertexes are:

$$\mathbf{S}_1^{z=16} = 10^3 \begin{pmatrix} 0.9809 & 1.0186 & 0.0010 \\ -0.9819 & -1.0176 & -0.0010 \end{pmatrix}$$

$$\mathbf{S}_2^{z=16} = 10^3 \begin{pmatrix} 0.9373 & 1.0640 & 0.0014 \\ -0.9387 & -1.0626 & -0.0014 \end{pmatrix}$$

$$\mathbf{S}_3^{z=16} = 10^3 \begin{pmatrix} 0.9982 & 1.0017 & 0.0004 \\ -0.9986 & -1.0013 & -0.0004 \end{pmatrix}.$$

An important point here is that including the observation with a proper convex hull improves the resulting control performance.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

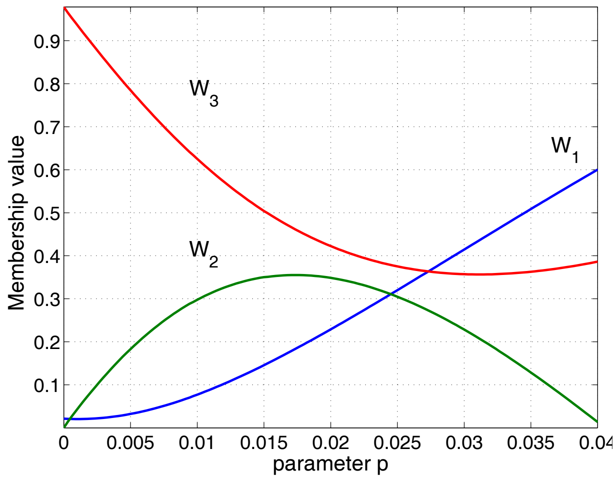

Fig. 5. Interpolated antecedent membership functions at $z = 16$

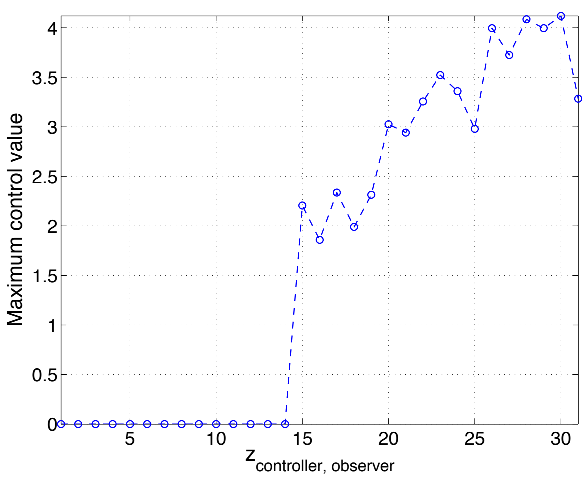

Fig. 6. Maximum of the control value in case of different convex hulls

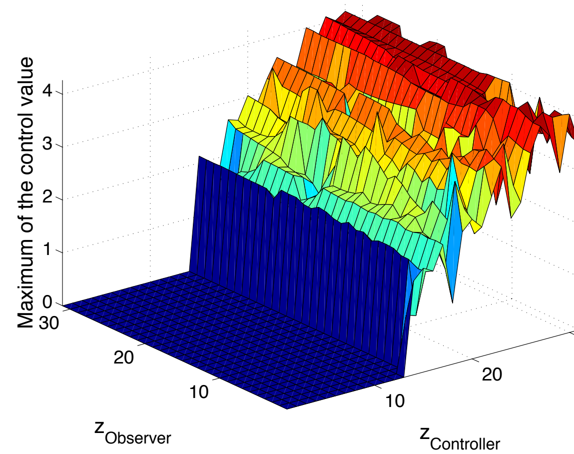

Fig. 7. Maximum of the control value at given convex hulls for the controller and observer

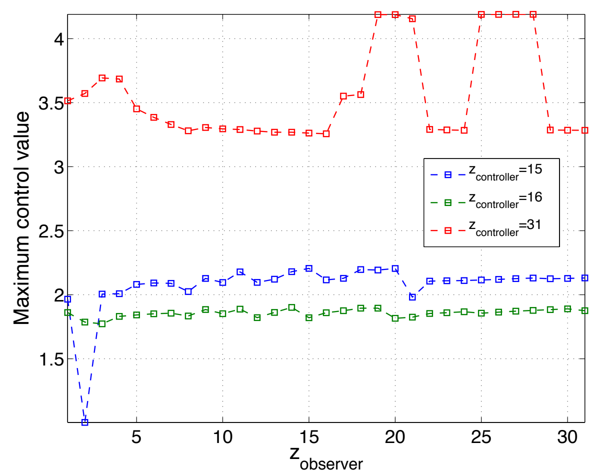

Fig. 8. Maximum of the control value when the convex hull for the observer is modified

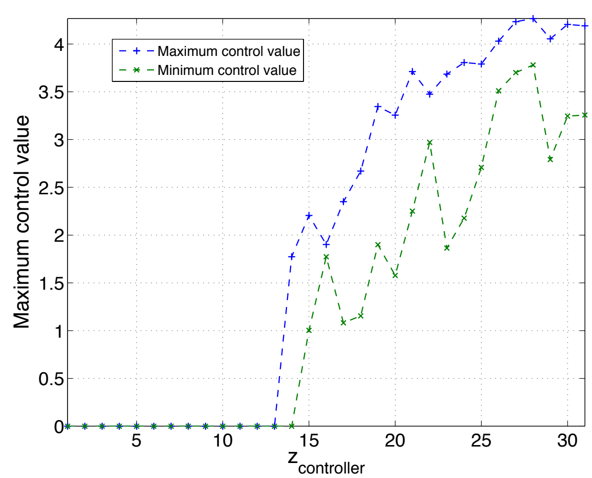

Fig. 9. Maximum of the control value when the convex hull for the controller is modified

Since the separation theory applies here, we may simply use different TP models for the controller design and for the observer design. Fig. 7 shows how the maximum of the control values varies (control value 0 means that there is no feasible solution) for the same $\mathbf{x}(0)$ when the TS fuzzy model used for the observer design ($z_{observer}$) is different from the TS fuzzy model used for controller design ($z_{controller}$). Fig. 8 shows some intersections of the surface depicted on Fig. 7. It shows the changing of the control performance when the controller is fixed but the convex hull of the observer is modified. Fig. 9 shows that the performance is rather variable when the observer is fixed and the convex hull of the controller is tightened.

One may continue and readily try various types of convex hulls systematically, since the execution of the TP$^+$ model transformation in the present case takes less than a minute on a regular computer. The conclusion of this simple example is that the convex hull can readily be manipulated by the TP$^+$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

model transformation and this may have considerable effects on the resulting control performances and the conservativeness of the solution. An additional conclusion is that we can achieve even better performance when we derive different TS fuzzy models for controller and observer design.

B. General framework for stability verification via Multi TP model transformation

We investigate the example of the 3DoF aeroelastic wing section. Very rich and deep investigations on the control design of the 2DoF and 3DoF aeroelastic wing section are available in a series of papers published in the Journal of Guidance and Dynamics and Control [77]–[83]. The goal of the design is to stabilize the pitch and the plunge motion of the wing by controlling the dynamics of the trailing edge actuator.

The qLPV model of the wing section is given in Appendix 1. The challenge in the control design of the wing section is the strong nonlinearity and various phenomena such as limit cycle oscillation and even chaotic behaviour emerging in the uncontrolled case. Assume that we face a situation where the observer based output feedback design is complete and we have a fuzzy logic controller (see Appendix 2) and neural network observer (see Appendix 3) in the structure given in Fig. 10 for $\Omega = [-0.3, 0.3]rad \times [8, 20]m/s$. Further, assume that the performance of the controlled system is acceptable, see the results for a very critical wind speed on Fig. 11, where the pitch and the plunge are shown alongside the controlled trailing edge, which has a direct effect on the dynamic motion of the wing and the control value of the dynamics of the trailing edge. However, no stability proof of the system is given. This example shows how the multi TP model transformation is capable of transforming the given system components to convex TS fuzzy models, such that the antecedent membership function systems are the same. Then we can check the stability via the feasibility test of LMIs. We may also substitute the consequent systems into LMIs which can indicate a certain performance as well.

First of all, we execute the multi TP model transformation (with grid $M = 137 \times 137$) on the system model, the observer and the controller. As a result, we find that all of these can be given exactly by 2×3 vertices with the same CNO type antecedent membership functions system as follows:

$$\mathbf{S}(\mathbf{p}(t)) = \mathcal{S} \boxtimes_{n=1}^{2} \mathbf{w}_n^{CNO}(p_n(t)), \tag{30}$$

$$\mathbf{F}(\mathbf{p}(t)) = \mathcal{F} \boxtimes_{n=1}^{2} \mathbf{w}_n^{CNO}(p_n(t)), \tag{31}$$

$$\mathbf{K}(\mathbf{p}(t)) = \mathcal{K} \boxtimes_{n=1}^{2} \mathbf{w}_n^{CNO}(p_n(t)). \tag{32}$$

The CNO antecedent membership functions are shown on Fig. 12. Once we have $\mathbf{S}_{r=1..6}$, $\mathbf{F}_{r=1..6}$ and $\mathbf{K}_{r=1..6}$, we can use, for instance, quadratic stability analysis. In this case, when we substitute these vertices into the MATLAB quad stab function, for instance, we see that the stability is guaranteed. The execution of the multi TP model transformation and the quad stab functions in MATLAB take a few minutes, in contrast the analytical derivations may be quite hard or even practically impossible in a reasonable amount of time.

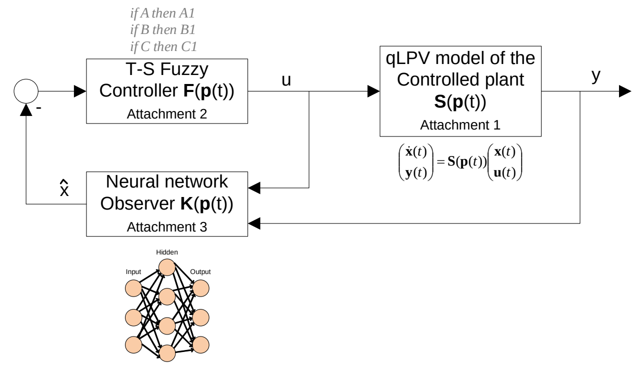

Fig. 10. Observer based output feedback control structure

VII. Conclusion

The paper generalized the TP model transformation and integrated various features in order to obtain a tractable, non-heuristic numerical framework for fuzzy rule base manipulation, such as HOSVD based main component analysis, complexity reduction and manipulation of antecedent fuzzy sets. Since the TP model transformation can be executed on various different kinds of representations, it can serve as a bridge to TS fuzzy model based theories of design and analysis. From a system control design perspective, the paper demonstrated that this new framework can be used to manipulate the TS fuzzy model representation specifically for LMI based control performance optimzation. Thus, the paper came to the conclusion that the TP model transformation can be considered as a final step of various identification and modeling processes, and can at the same time be regarded as a general "interface" – i.e. a preprocessing step – towards fulfilling further design requirements. It was shown that the TP model transformation can also be used to find TS fuzzy representations of system components with the same parameter dependent convex weightings, a property which directly leads to LMI based system analysis and stability verification frameworks. In this sense, the results presented in the paper represent an attempt to resolve the main stability verification criticisms which are often leveled at fuzzy or any soft-computing based control solutions.

VIII. Appendix 1: Model of the 3DOF aeroelastic wing section

The equations of motion (detailed description is in [84], [85]):

$$\begin{pmatrix} \mathbf{S}_1 & \mathbf{S}_2 & \mathbf{S}_3 \end{pmatrix} \begin{pmatrix} \ddot{h} \\ \ddot{\alpha} \\ \ddot{\beta} \end{pmatrix} + \tag{33}$$

$$+\mathbf{S}_4 \begin{pmatrix} \dot{h} \\ \dot{\alpha} \\ \dot{\beta} \end{pmatrix} + \mathbf{S}_5 \begin{pmatrix} h \\ \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} -L \\ M \\ k_{\beta_{servo}} \beta_{des} \end{pmatrix}.$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

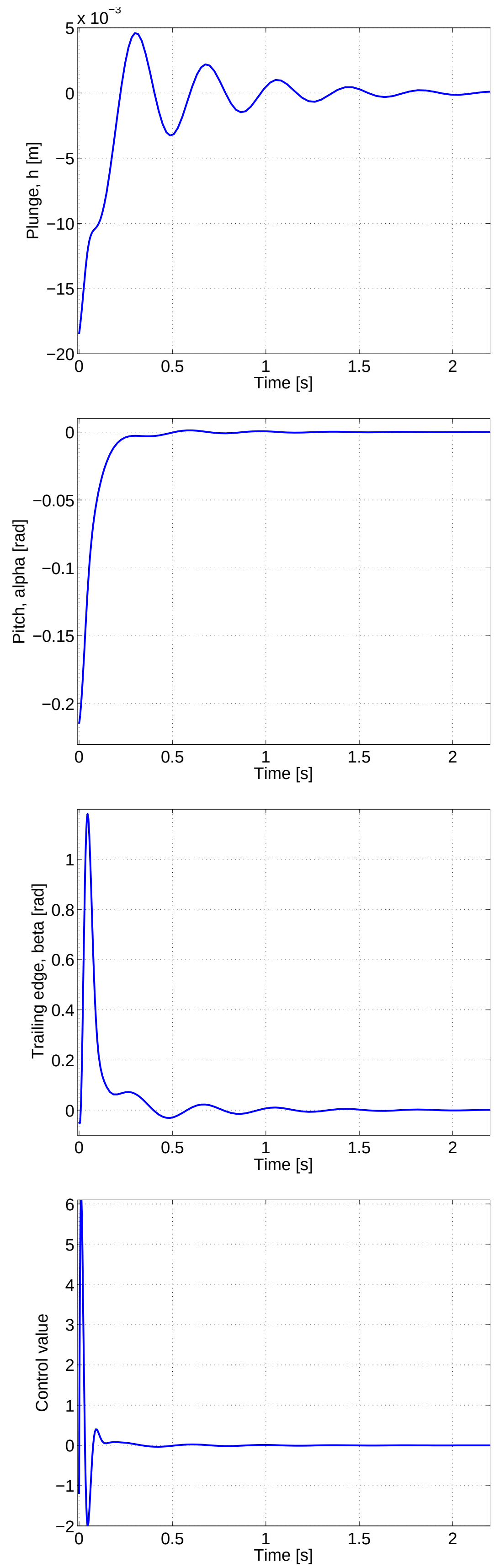

Fig. 11. Time response of controlled system for $U = 14.4m/s$.

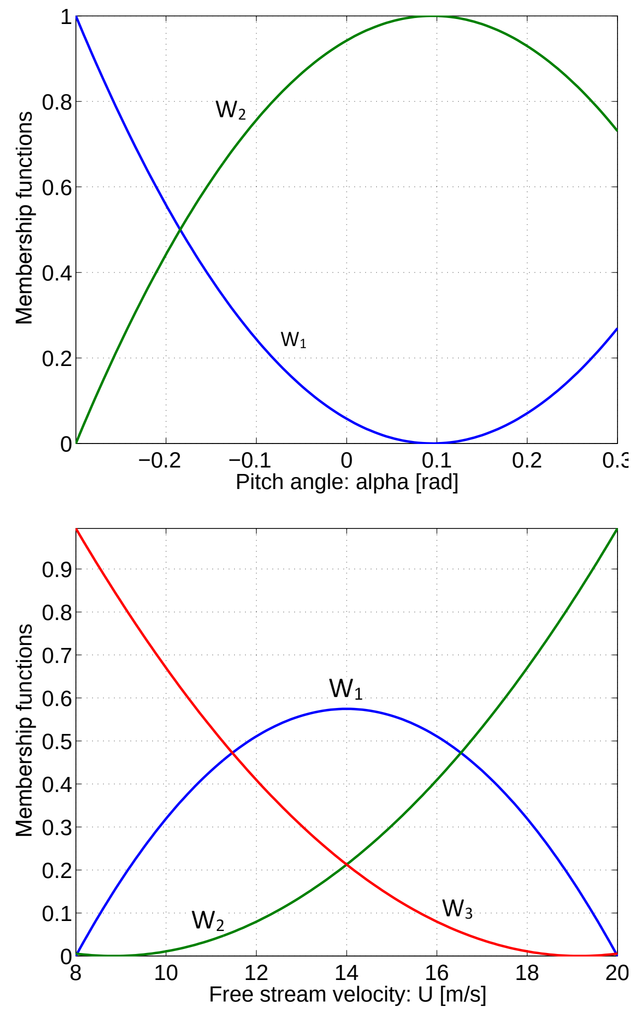

Fig. 12. CNO type antecedent membership functions of the dimensions U and α.

$$\mathbf{S_1} = \begin{pmatrix} m_h + m_\alpha + m_\beta \\ m_a x_a b + m_\beta r_\beta + m_\beta x_\beta \\ m_\beta r_\beta \end{pmatrix} \tag{34}$$

$$\mathbf{S_2} = \begin{pmatrix} m_a x_a b + m_\beta r_\beta + m_\beta x_\beta \\ \hat{I}_\alpha + \hat{I}_\beta + m_\beta r_\beta^2 + 2x_\beta m_\beta r_\beta \\ \hat{I}_\beta + x_\beta m_\beta r_\beta \hat{I}_\beta m x_\alpha b \end{pmatrix} \tag{35}$$

$$\mathbf{S_3} = \begin{pmatrix} m_\beta r_\beta \\ \hat{I}_\beta + x_\beta m_\beta r_\beta \\ I_\alpha \end{pmatrix} \tag{36}$$

$$\mathbf{S}_4 = \begin{pmatrix} c_h & 0 & 0 \\ 0 & c_\alpha & 0 \\ 0 & 0 & c_{\beta_{servo}} \end{pmatrix} \tag{37}$$

$$\mathbf{S}_5 = \begin{pmatrix} k_h & 0 & 0 \\ 0 & k_\alpha(\alpha) & 0 \\ 0 & 0 & k_{\beta_{servo}} \end{pmatrix} \tag{38}$$

$k_\alpha(\alpha)$ is obtained by curve fitting on the measured displacement-moment data for a non-linear spring as $k_\alpha(\alpha) =$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

$25.55 - 103.19\alpha + 543.24\alpha^2$. Quasi-steady aerodynamic force and moment is derived as:

$$L = \rho U^2 b C_{l_\alpha}\left(\alpha + \frac{\dot{h}}{U} + \left(\frac{1}{2} - a\right) b \frac{\dot{\alpha}}{U}\right) + \rho U^2 b c_{l_\beta} \beta \quad (39)$$

$$M = \rho U^2 b^2 C_{m_{\alpha,eff.}}\left(\alpha + \frac{\dot{h}}{U} + \left(\frac{1}{2} - a\right) b \frac{\dot{\alpha}}{U}\right) + \\ + \rho U^2 b C_{m_{\beta,eff.}} \beta.$$

L and M above are accurate for the low-velocity regime.

The trailing-edge servo-motor dynamics is represented using a second-order system of the form $\hat{I}_\beta \ddot{\beta} + c_{\beta_{servo}} \dot{\beta} + k_{\beta_{servo}} \beta = k_{\beta_{servo}} \mathbf{u}_\beta$. Combining equations (33) and (39) one can obtain the qLPV model of the system:

$$\begin{pmatrix} \dot{\mathbf{x}}(t) \\ \mathbf{y}(t) \end{pmatrix} = \mathbf{S}(\mathbf{p}(t)) \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \end{pmatrix}, \quad (40)$$

where

$$\mathbf{x}(t) = \begin{pmatrix} \dot{h} & \dot{\alpha} & \dot{\beta} & h & \alpha & \beta \end{pmatrix}^T \quad \text{and} \quad \mathbf{u}(t) = u_\beta.$$

It is assumed that only state variable α is measurable, thus $\mathbf{y} = \alpha$ is chosen as the output. The output matrix in the system matrix is the following:

$$\mathbf{C} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}. \quad (41)$$

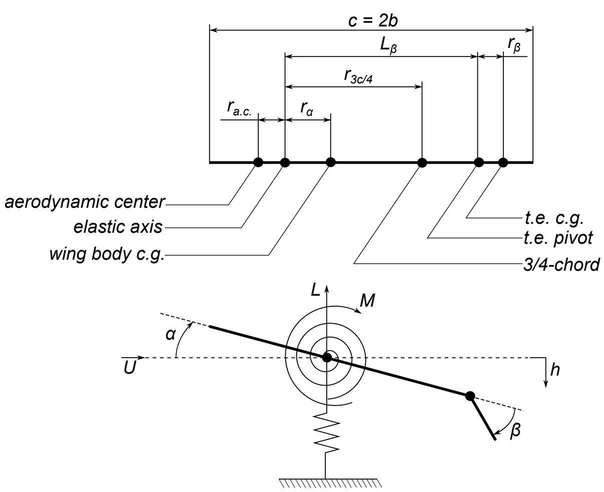

Fig. 13. Two-dimension flat plate airfoil small deflection, force notation and schematic diagram

The nonlinear system matrix $\mathbf{S}(\mathbf{p}(t))$ depends on the free stream velocity U and pitch α, thus parameter vector $\mathbf{p}(t)$ can be written as $\mathbf{p}(t) = \begin{pmatrix} U(t) & \alpha(t) \end{pmatrix}^T$.

IX. Appendix 2: Fuzzy logic controller

The fuzzy logic controller has two inputs representing the elements of the parameter vector as α and U. 9 antecedent fuzzy sets are given on each input dimension. Thus 81 linguistic fuzzy rule describes the control law. The output of the fuzzy controller is calculated by TS fuzzy model strategy. The antecedent sets are depicted on Figure 14. The output of the fuzzy engine is the feedback gains $\mathbf{F}(\mathbf{p}(t))$. The output of the controller is the $-\mathbf{F}(\mathbf{p}(t))\hat{\mathbf{x}}(t)$ estimated by the observer except the measurable α.

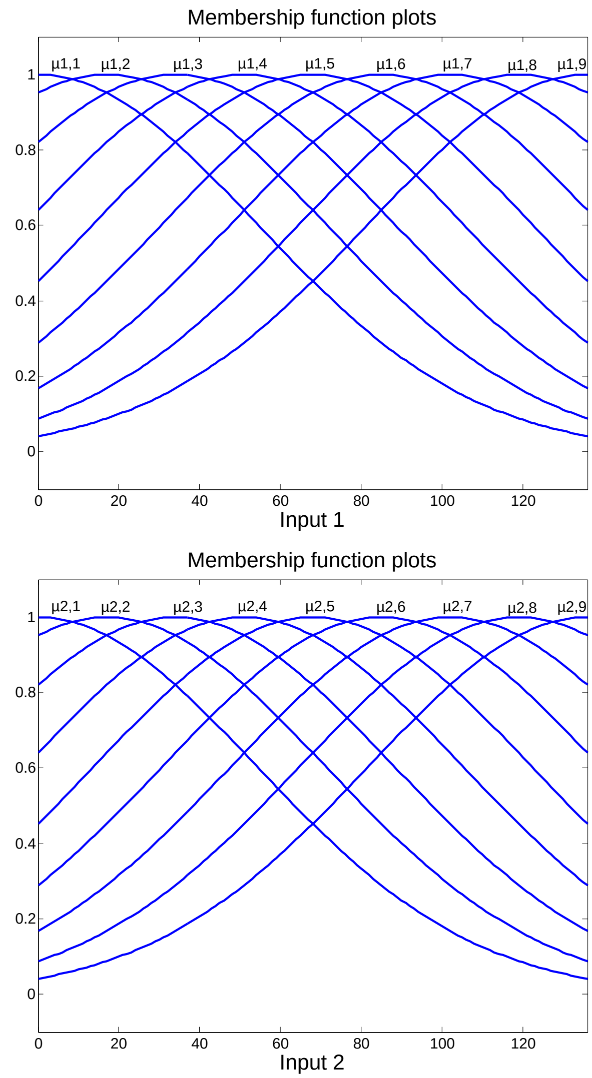

Fig. 14. Antecedent membership functions of the fuzzy controller

X. Appendix 3: Neural Network observer

The observer is given by a 3 layered neural networks. Each layer has 25 neurons. Actually each element of the $\mathbf{K}(\mathbf{p}(t))$ is computed by one neural network so in total 6 neural networks are given.

References

[1] P. Baranyi, "TP model transformation as a way to LMI-based controller design," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 2, pp. 387–400, 2004.

[2] P. Baranyi, D. Tikk, Y. Yam, and R. Patton, "From differential equations to PDC controller design via numerical transformation," *Computers in Industry*, vol. 51, no. 3, pp. 281–297, 2003.

[3] P. Baranyi, Y. Yam, and P. Varlaki, *TP model transformatoin in polytopic model-based control*, ser. Automation and Control Engineering. Taylor and Francis, in print, 2013.

[4] Y. Yam, "Fuzzy approximation via grid point sampling and singular value decomposition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 27, no. 6, pp. 933–951, 1997.

[5] Y. Yam, P. Baranyi, and C. Yang, "Reduction of fuzzy rule base via singular value decomposition," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 2, pp. 120–132, 1999.

[6] Z. Szabó, P. Gáspár, S. Nagy, and P. Baranyi, "TP model transformation for control-oriented qLPV modeling," *Australian Journal of Intelligent Information Processing Systems*, vol. 10, no. 2, pp. 36–53, 2008.

[7] P. Baranyi, "Tensor-product model-based control of two-dimensional aeroelastic system," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 2, pp. 391–400, 2006.

[8] ——, "Output feedback control of two-dimensional aeroelastic system," *Journal of Guidance, Control and Dynamics*, vol. 29, no. 3, pp. 762–767, 2006.

[9] Z. Petres, P. Baranyi, and H. Hashimoto, "Approximation and complexity trade-off by TP model transformation in controller design: A case study of the TORA system," *Asian Journal of Control*, vol. 12, no. 5, pp. 575–585, 2010.

[10] Z. Petres, P. Baranyi, P. Korondi, and H. Hashimoto, "Trajectory tracking by TP model transformation: Case study of a benchmark problem," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 3, pp. 1654–1663, 2007.

[11] B. Takarics, "TP model transformation based sliding mode control and friction compensation," PhD Thesis, Budapest University of Technology and Economics, Budapest, Hungary, 2011.

[12] B. Takarics and P. Baranyi, "Friction compensation in TP model form - aeroelastic wing as an example system," *Acta Polytechnica Hungarica*, in print.

[13] P. Baranyi, P. Korondi, and H. Hashimoto, "Global asymptotic stabilization of the prototypical aeroelastic wing section via TP model transformation," *Asian Journal of Control*, vol. 7, no. 2, pp. 99–111, 2005.

[14] P. Baranyi, P. Korondi, and K. Tanaka, "Parallel distributed compensation based stabilization of a 3-DOF RC helicopter: A tensor product transformation based approach," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 13, pp. 25–34, 2009.

[15] P. Baranyi and A. Varkonyi-Koczy, "TP transformation based dynamic system modeling for nonlinear control," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 6, pp. 2191–2203, 2005.

[16] "TPtool — Tensor Product MATLAB Toolbox," Website, 2007, www.tp-control.hu.

[17] P. Baranyi, "Convex hull generation methods for polytopic representations of LPV models," in *Applied Machine Intelligence and Informatics, 2009. SAMI 2009. 7th International Symposium on*. IEEE, 2009, pp. 69–74.

[18] P. Várkonyi, D. Tikk, P. Korondi, and P. Baranyi, "A new algorithm for RNO-INO type tensor product model representation," in *Proceedings of the IEEE* 9^{th} *International Conference on Intelligent Engineering Systems*, 2005, pp. 263–266.

[19] Y. Yam, C. Yang, and P. Baranyi, "Singular value-based fuzzy reduction with relaxed normality condition," in *Interpretability issues in fuzzy modeling*, ser. Studies in Fuzziness and Soft Computing, J. Casillas, O. Cordon, F. Herrera, and L. Magdalena, Eds. Springer Verlag, 2003, vol. 128, pp. 325–354.

[20] P. Baranyi, Y. Yam, A. Várkonyi-Kóczy, R. Patton, P. Michelberger, and M. Sugiyama, "SVD-based complexity reduction to TS fuzzy models," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 2, pp. 433–443, 2002.

[21] P. Baranyi, Y. Yam, A. Várkonyi-Kóczy, and R. Patton, "SVD-based reduction to MISO TS models," *IEEE Transactions on Industrial Electronics*, vol. 50, no. 1, pp. 232–242, 2003.

[22] P. Baranyi, P. Korondi, R. Patton, and H. Hashimoto, "Trade-off between approximation accuracy and complexity for TS fuzzy models," *Asian Journal of control*, vol. 6, no. 1, pp. 21–33, 2004.

[23] P. Baranyi, Z. Petres, P. Korondi, Y. Yam, and H. Hashimoto, "Complexity relaxation of the tensor product model transformation for higher dimensional problems," *Asian Journal of Control*, vol. 9, no. 2, pp. 195–200, 2007.

[24] D. Tikk, P. Baranyi, R. Patton, and J. Tar, "Approximation capability of TP model forms," *Australian Journal of Intelligent Information Processing Systems*, vol. 8, no. 3, pp. 155–163, 2004.

[25] D. Tikk, P. Baranyi, and R. Patton, "Polytopic and TS models are nowhere dense in the approximation model space," in *Proceedings of the 2002 IEEE International Conference on Systems, Man and Cybernetics*, vol. 7, Yasmine Hammamet, Tunisia, 2002.

[26] ——, "Approximation properties of TP model forms and its consequences to TPDC design framework," *Asian Journal of Control*, vol. 9, no. 3, pp. 221–231, 2007.

[27] P. Baranyi, L. Szeidl, P. Várlaki, and Y. Yam, "Definition of the HOSVD based canonical form of polytopic dynamic models," in *Proceedings of the 2006 IEEE International Conference on Mechatronics*, Budapest, Hungary, July 3-5 2006, pp. 660–665.

[28] P. Baranyi, L. Szeidl, and P. Várlaki, "Numerical reconstruction of the HOSVD based canonical form of polytopic dynamic models," in *Proceedings of the* 10^{th} *International Conference on Intelligent Engineering Systems, 2006*, London, UK, June 26-28 2006, pp. 196–201.

[29] L. Szeidl and P. Várlaki, "HOSVD based canonical form for polytopic models of dynamic systems," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 13, no. 1, pp. 52–60, 2009.

[30] S. Nagy, Z. Petres, P. Baranyi, and H. Hashimoto, "Computational relaxed TP model transformation: restricting the computation to subspaces of the dynamic model," *Asian Journal of Control*, vol. 11, no. 5, pp. 461–475, 2009.

[31] S. Nagy, P. Baranyi, and Z. Petres, "Centralized tensor product model form," in *6th International Symposium on Applied Machine Intelligence and Informatics, 2008*, 2008, pp. 189–193.

[32] P. Baranyi and Y. Yam, "Case study of the TP-model transformation in the control of a complex dynamic model with structural nonlinearity," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 3, pp. 895–904, 2006.

[33] C. Sun, Y. Huang, C. Qian, and L. Wang, "On modeling and control of a flexible air-breathing hypersonic vehicle based on LPV method," *Frontiers of Electrical and Electronic Engineering*, pp. 1–13.

[34] S. Rangajeeva and J. Whidborne, "Linear parameter varying control of a quadrotor," in *Proceedings of the* 6^{th} *IEEE International Conference on Industrial and Information Systems (ICIIS), 2011*, 2011, pp. 483–488.

[35] W. Gai, H. Wang, T. Guo, and D. Li, "Modeling and LPV flight control of the canard rotor/wing unmanned aerial vehicle," in *Proceedings of the* 2^{nd} *International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011*, 2011, pp. 2187–2191.

[36] W. Qin, Z. Zheng, G. Liu, J. Ma, and W. Li, "Robust variable gain control for hypersonic vehicles based on LPV," *Systems Engineering and Electronics*, vol. 33, no. 6, pp. 1327–1331, 2011.

[37] S. Chumalee and J. Whidborne, "LPV autopilot design of a jindivik uav," 2009.

[38] S. Chumalee, "Robust gain-scheduled H_∞ control for unmanned aerial vehicles," 2010.

[39] T. Luspay, T. Péni, and B. Kulcsar, "Constrained freeway traffic control via linear parameter varying paradigms," *Control of Linear Parameter Varying Systems With Applications*, p. 461, 2012.

[40] T. Luspay, B. Kulcsar, T. Péni, and I. Varga, "Freeway ramp metering: an LPV set theoretical analysis," in *Proceedings of the 2011 American Control Conference*, 2011, pp. 733–738.

[41] B. Al-Hadithi, A. Jiménez, and F. Matía, "New methods for the estimation of Takagi–Sugeno model based extended kalman filter and its applications to optimal control for nonlinear systems," *Optimal Control Applications and Methods*, 2011.

[42] A. Rovid, L. Szeidl, and P. Varlaki, "On tensor-product model based representation of neural networks," in *Proceedings of the 15th IEEE International Conference on Intelligent Engineering Systems (INES), 2011*, 2011, pp. 69–72.

[43] Z. Szabó, P. Gáspár, and J. Bokor, "A novel control-oriented multi-affine qLPV modeling framework," in *Proceedings of the 18th Mediterranean Conference on Control & Automation (MED), 2010*, 2010, pp. 1019–1024.

[44] R. Precup, S. Preitl, B. Ursache, P. Clep, P. Baranyi, and J. Tar, "On the combination of tensor product and fuzzy models," in *Proceedings of the IEEE International Conference on Automation, Quality and Testing, Robotics, 2008*, vol. 2, 2008, pp. 48–53.

[45] R. Precup, C. Dragoş, S. Preitl, M. Radac, and E. Petriu, "Novel tensor product models for automatic transmission system control," 2012.

[46] R. Precup and H. Hellendoorn, "A survey on industrial applications of fuzzy control," *Computers in Industry*, vol. 62, no. 3, pp. 213–226, 2011.

[47] Š. Ileš, J. Matuško, and F. Kolonić, "Tensor product transformation based speed control of permanent magnet synchronous motor drives," in *Proceedings of the 17th International Conference on Electrical Drives and Power Electronics, EDPE 2011 (5th Joint Slovak-Croatian Conference)*, 2011.

[48] R. Precup, L. Dioanca, E. Petriu, M. Radac, S. Preitl, and C. Dragos, "Tensor product-based real-time control of the liquid levels in a three

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2013.2278982, IEEE Transactions on Fuzzy Systems

tank system," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), 2010*, 2010, pp. 768–773.

[49] Š. Ileš, F. Kolonić, and J. Matuško, "Linear matrix inequalities based H_∞ control of gantry crane using tensor product transformation," in *Proceedings of the 18th International Conference on Process Control*, 2011.

[50] F. Kolonic and A. Poljugan, "Experimental control design by TP model transformation," in *Proceedings of the 2006 IEEE International Conference on Mechatronics*, Budapest, Hungary, 2006, pp. 666–671.

[51] F. Kolonic, A. Poljugan, and I. Petrovic, "Tensor product model transformation-based controller design for gantry crane control system–an application approach," *Acta Polytechnica Hungarica*, vol. 3, no. 4, pp. 95–112, 2006.

[52] Y. Kunii, B. Solvang, G. Sziebig, and P. Korondi, "Tensor product transformation based friction model," in *Proceedings of the 11th International Conference on Intelligent Engineering Systems, 2007*, 2007, pp. 259–264.

[53] E. Tognetti, R. Oliveira, and P. Peres, "An LMI-based approach to static output feedback stabilization of TS fuzzy systems," in *World Congress*, vol. 18, no. 1, 2011, pp. 12 593–12 598.

[54] L. A. B. T. V. C. S. Campos, F. O. Souza and R. M. Palhares, "New stability conditions based on piecewise fuzzy lyapunov functions and tensor product transformations," *IEEE Trans. on Fuzzy Systems*, p. 12, 2013.

[55] B. Takarics and P. Baranyi, "Tensor product model based control of a three degrees-of-freedom aeroelastic model," *Journal of Guidance, Control and Dynamics*, 2013, in print.

[56] B. Takarics, "Parallel distributed compensation based sector sliding mode control of takagi-sugeno type polytopic models." IEEE, Jan. 2012, pp. 501–506.

[57] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[58] ——, "Dimensionality reduction in higher-order-only ICA," in *Proceedings of the IEEE Signal Processing Workshop on Higher-Order Statistics, 1997*, Banff, Alberta, Canada, 1997, pp. 316–320.

[59] M. Ishteva, L. De Lathauwer, P. Absil, and S. Van Huffel, "Dimensionality reduction for higher-order tensors: algorithms and applications," *International Journal of Pure and Applied Mathematics*, vol. 42, no. 3, p. 337, 2008.

[60] P. Apkarian, P. Gahinet, and G. Becker, "Self-scheduled H_∞ linear parameter-varying systems," in *Proceedings of the 1994 American Control Conference*, vol. 1, Baltimore, Maryland, USA, 1994, pp. 856–860.

[61] E. Feron, P. Apkarian, and P. Gahinet, "S-procedure for the analysis of control systems with parametric uncertainties via parameter-dependent Lyapunov functions," in *Proceedings of the 1995 American Control Conference*, vol. 1, Seattle, Washington, USA, 1995, pp. 968–972.

[62] A. Nemirovskii and P. Gahinet, "The projective method for solving linear matrix inequalities," in *Proceedings of the 1994 American Control Conference*, vol. 1, Baltimore, Maryland, USA, 1994, pp. 840–844.

[63] C. Scherer, "H_∞-optimization without assumptions on finite or infinite zeros," *SIAM Journal on Control and Optimization*, vol. 30, no. 1, pp. 143–166, 1992.

[64] J. Doyle, K. Glover, P. Khargonekar, and B. Francis, "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Transactions on Automatic Control*, vol. 34, no. 8, pp. 831–847, 1989.

[65] P. Gahinet, "Explicit controller formulas for LMI-based H_∞ synthesis," in *Proceedings of the 1994 American Control Conference*, vol. 3, Baltimore, Maryland, USA, 1994, pp. 2396–2400.

[66] P. Gahinet and A. Laub, "Reliable computation of γ_{opt} in singular H_∞ control," in *Proceedings of the 33^{rd} IEEE Conference on Decision and Control, 1994*, vol. 2, Orlando, Florida, USA, 1994, pp. 1527–1532.

[67] P. Apkarian and P. Gahinet, "A convex characterization of gain-scheduled H_∞ controllers," *IEEE Transactions on Automatic Control*, vol. 40, no. 5, pp. 853–864, 1995.

[68] S. Boyd, V. Balakrishnan, and P. Kabamba, "A bisection method for computing the H_∞ norm of a transfer matrix and related problems," *Mathematics of Control, Signals, and Systems (MCSS)*, vol. 2, no. 3, pp. 207–219, 1989.

[69] I. Kaminer, P. Khargonekar, and M. Rotea, "Mixed H_2/H_∞ control for discrete-time systems via convex optimization," *Automatica*, vol. 29, no. 1, pp. 57–70, 1993.

[70] C. Scherer and S. Weiland, "Linear matrix inequalities in control," *Lecture Notes, Dutch Institute for Systems and Control, Delft, The Netherlands*, 2000, http://www.cs.ele.tue.nl/sweiland/lmi.htm.

[71] P. Seiler, G. J. Balas, and A. Packard, "Linear parameter-varying control for the x-53 active aeroelastic wing," in *Control of Linear Parameter Varying Systems with Applications*, J. Mohammadpour and C. W. Scherer, Eds. Boston, MA: Springer US, 2012, pp. 483–512.

[72] K. Tanaka and M. Sugeno, "Stability analysis and design of fuzzy control systems," *Fuzzy sets and systems*, vol. 45, no. 2, pp. 135–156, 1992.

[73] K. Tanaka, T. Ikeda, and H. Wang, "Fuzzy regulators and fuzzy observers: relaxed stability conditions and LMI-based designs," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 2, pp. 250–265, 1998.

[74] K. Tanaka and H. Wang, *Fuzzy control systems design and analysis: a linear matrix inequality approach*. Wiley-Interscience, 2001.

[75] K. Tanaka and M. Sugeno, "Stability analysis and design of fuzzy control systems," *Fuzzy sets and systems*, vol. 45, no. 2, pp. 135–156, 1992.

[76] H. Wang, K. Tanaka, and M. Griffin, "Parallel distributed compensation of nonlinear systems by Takagi-Sugeno fuzzy model," in *Proceedings of the International Joint Conference of the 4^{th} IEEE International Conference on Fuzzy Systems and The 2nd International Fuzzy Engineering Symposium, 1995*, vol. 2, Yokohama, Japan, 1995, pp. 531–538.

[77] V. Mukhopadhyay, "Transonic flutter suppression control law design and wind-tunnel test results," *Journal of Guidance, Navigation, and Control*, vol. 23, no. 5, pp. 930–937, 2000.

[78] J. Ko, T. Strganac, and A. Kurdila, "Adaptive feedback linearization for the control of a typical wing section with structural nonlinearity," *Nonlinear Dynamics*, vol. 18, no. 3, pp. 289–301, 1999.

[79] J. Ko, A. Kurdila, and T. Strganac, "Nonlinear control of a prototypical wing section with torsional nonlinearity," *Journal of Guidance, Control and Dynamics*, vol. 20, no. 6, pp. 1180–1189, 1997.

[80] W. Yim, S. Singh, and W. Wells, "Nonlinear control of a prototypica aeroelastic wing section: State-dependent Riccati Equation Method," in *Proceedings of the 4^{th} International Conference on Nonlinear Problems in Aviation and Aerospace (ICNPAA'02)*, Melbourne, Florida, USA, 2002, pp. 543–550.

[81] R. Scott and L. Pado, "Active control of wind-tunnel model aeroelastic response using neural networks," *Journal of Guidance, Control and Dynamics*, vol. 23, no. 6, pp. 1100–1108, 2000.

[82] J. Block and H. Gilliatt, "Active control of an aeroelastic structure," Master's thesis, Texas A&M University, 1996.

[83] J. Block and T. Strganac, "Applied active control for a nonlinear aeroelastic structure," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 6, pp. 838–845, 1998.

[84] Z. Prime, B. Cazzolato, and C. Doolan, "A mixed H_2/H_∞ scheduling control scheme for a two degree-of-freedom aeroelastic system under varying airspeed and gust conditions," in *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Honolulu, Hawaii, USA, 2008, pp. 18–21.

[85] Z. Prime, B. Cazzolato, C. Doolan, and T. Strganac, "Linear-parameter-varying control of an improved three-degree-of-freedom aeroelastic model," *Journal of Guidance, Control and Dynamics*, vol. 33, no. 2, pp. 615–619, 2010.

Peter Baranyi Peter Baranyi received his M.Sc and Ph.D. degrees in 1995 and in 1999, respectively, and holds a D.Sc. degree since 2006. He is member of the Hungarian Academy of Engineering. His key research topics are qLPV, LMI and TP model transformation based control design. He has established the 3D Internet based Control and Communication laboratory at MTA SZTAKI, and has also initiated the ITM Norwegian and Hungarian Lab. He has more than 380 publications, and was invited on over 50 occasions to give seminars at various universities and research institutions. He has organized a number of conferences as general chair and was guest editor of special issues in several scientific journals. He also took part in the organization committees of more than 70 conferences, and supervised a number of large-scale projects, both at a domestic and international level.