This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

Relaxed TS Fuzzy model transformation to improve the approximation accuracy / complexity trade-off and relax the computation complexity

Péter Baranyi

***Abstract*—The primary goal of the paper is to introduce the Relaxed TS Fuzzy model transformation, a method that enhances the original TS Fuzzy model transformation in two ways. First, it focuses on achieving a more efficient reduction of the number of antecedent fuzzy sets – hence, the fuzzy rules of the TS Fuzzy models – while minimizing the approximation error. Second, it aims to reduce the computational load required for the transformation process. With the first enhancement, the proposed transformation strikes a better balance between the number of fuzzy rules and the approximation accuracy of TS Fuzzy models. With the second enhancement, a unique pre- and post-processing of the TS Fuzzy model transformation is introduced leading to the radical computational improvements. The core part of the original TS Fuzzy model transformation is the Higher Order Singular Value Decomposition (HOSVD) used to balance the approximation quality with the number of fuzzy rules by truncating singular values. The HOSVD itself is a computationally intensive algorithm, the possibilities for advancements in its implementation seem to be limited as much research has focused on its optimization in the past and had reached its pinnacle in terms of computational complexity more than a decade ago. Therefore, the approach presented in this paper does not concentrates directly on enhancing HOSVD further, but instead proposes a unique pre- and post-processing technique for the tensor on which HOSVD is applied, tailored to the special characteristics of the TS Fuzzy model and the system model under consideration. Following a description of the proposed enhancements, the paper presents numerical examples and two examples of real-world engineering models to demonstrate the effectiveness of the Relaxed TS Fuzzy model transformation compared to the original TS Fuzzy model transformation.**

Keywords: TS Fuzzy model, TS Fuzzy model transformation, approximation-complexity trade-off.

I. Introduction

The TS Fuzzy model transformation, also known as TP model transformation, was first proposed in 1997 and has since undergone continuous development [1]–[9].

The mathematical perspective of the TP model transformation involves extending the concept of the Higher Order Singular Value Decomposition (HOSVD) [10] to continuous bounded functions. It serves as a method for numerically reconstructing the HOSVD of functions [1]–[3]. Besides generating higher order singular value based orthonormed structures, the TP model transformation is capable of deriving convex tensor product forms [1]–[9], [11] equivalent to the transfer functions of commonly used TS Fuzzy models, such that the antecedent fuzzy sets are defined as Ruspini-partitions. Therefore, the TP model transformation is also referred to as the TS Fuzzy model transformation in the literature on fuzzy modeling.

From the fuzzy design perspective, the TS Fuzzy model transformation allows for the generation of various alternative TS Fuzzy models based on either analytic functions or black-box models, with different types of convex hulls of the consequents [1]–[9], [11], which can enhance further design outcomes. This feature is particularly useful when further design steps need to be taken based on the consequents, as in the case of e.g. Linear Matrix Inequality (LMI) and Parallel Distributed Compensation (PDC) based control design [1]–[8], [12], [13]. Many research papers have documented improvements in control performance achieved by generating diverse alternatives to a given TS Fuzzy model, derived using the TS Fuzzy model transformation and followed by the application of LMI. Recent examples, mostly published in IEEE Transactions on Fuzzy Systems, include [14]–[30]. At the same time, the applicability of the TS Fuzzy model transformation is often limited by its heavy computational requirements.

For this reason, the development of the TS Fuzzy model transformation has centered around three main purposes over the course of the past 25 years:

- **TS Fuzzy model alternatives:** The goal is to vary the convex hull defined by the consequents through the manipulation of antecedent fuzzy sets and the input space to enhance further design outcomes [1]–[9], [11], [31], [32].
- **TS Fuzzy model complexity minimisation and associated trade-offs:** The goal is to derive a minimal number of fuzzy rules by compromising between the number of fuzzy rules and approximation accuracy to facilitate further design steps [1]–[9], [33]–[35], [35]–[41].
- **Computational relaxation:** The goal is to render the underlying algorithms themselves more computationally efficient. A key step of the transformation is based on the computationally expensive HOSVD, which is executed on a tensor in which the number of elements exponentially increases with the number of inputs during the numerical reconstruction of the antecedent fuzzy sets [1]–[3], [7].

P. Baranyi (peter.baranyi@uni-corvinus.hu) is with the Corvinus University of Budapest, Corvinus Institute for Advanced Studies (CIAS) and Institute of Data Analytics and Information Systems, Fővámtér 8, Budapest. Hungary and the Hungarian Research Network (HUN-REN). The author declares that there is no conflict of interest regarding the publication of this paper.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

A. Novel contributions of the paper

The paper proposes an improved variant of the TS Fuzzy model transformation, referred to as the Relaxed TS Fuzzy model transformation. Its key properties are:

- i) It provides improved accuracy and complexity trade-off for fuzzy rule base reduction.
- ii) It requires less computational load that considerably extends the class of models where the TS Fuzzy model transformation can efficiently be applied.

B. Challenges and the core idea of the solution

In the original TS Fuzzy model transformation, HOSVD is executed on a tensor representing the discretised variant of the given model over a hyper-rectangular grid. The resulting higher-order singular values express the relative importance – defined based on approximation error – of the different fuzzy rules. Therefore, rank reduction of the discretised tensor obtained by truncating the smallest singular values directly leads to a trade-off between the number of fuzzy rules and the approximation accuracy of the resulting TS Fuzzy model. The sum of the truncated singular values represents an upper bound on the resulting approximation error. Thus, the fundamental procedure in the original TS Fuzzy model transformation involves the application of HOSVD, a method that is inherently computationally intensive.

The evolution of HOSVD has been marked by extensive efforts to enhance its implementation, as detailed in [10]. This pursuit has led to incremental improvements in the associated computational methods. For instance, the Sequentially-truncated HOSVD (ST-HOSVD) algorithm, introduced in 2012 [42]. Approximately a decade ago, the development of HOSVD achieved its peak in terms of computational complexity and accuracy in handling singular values. Possibilities for further breakthroughs in this domain seem to be limited.

Therefore, the approach presented in this paper does not concentrate on enhancing HOSVD further, but instead proposes a unique pre-processing and post-processing technique for the tensor on which HOSVD is applied, tailored to the special characteristics of the TS Fuzzy model and the system model under consideration, which leads to radical improvements. The paper reveals and proves that the convexity of the output of the TS Fuzzy model transformation, achieved by further transforming the result of the HOSVD using convex transformation techniques, makes it possible to remove blocks of identical elements within a given tensor. Depending on the number of such elements, a considerable reduction on computational requirements can be achieved.

Based on the above, the proposed approach does not serve as a substitute computational model for HOSVD, but instead represents a supplementary step to HOSVD in the case of the TS Fuzzy model transformation, such that the data is tailored to the unique attributes of the TS Fuzzy model and the system model at hand.

C. Outlines of the Proposed Solution

The key idea behind the Relaxed TS Fuzzy model transformation is to modify the inputs to and the outputs from the HOSVD based step of the original TS Fuzzy model transformation.

In the proposed transformation, certain appropriately selected elements are temporarily separated from the discretised tensor. This results in a smaller values of the singular values and, further, the HOSVD is executed on a tensor that has fewer elements. In the final step, after the convex form is determined, the resulting decomposition is restructured and the previously separated elements are reinserted.

D. Structure of the Paper

Section II serves to define the notations used throughout the paper and establish the fundamental concepts necessary for the development of the Relaxed TS Fuzzy model transformation. Section III recalls the algorithm of the original TS Fuzzy model transformation. Section IV proposes an improved HOSVD based rank reduction for tensors having identical blocks. Section V develops the Relaxed TS Fuzzy model transformation based on the improved reduction proposed in Section IV. Section VI presents two demonstrative examples to provide further details on the execution of the original and the Relaxed TS Fuzzy model transformation, and to be able to obtain a comprehensive comparison of the two. Sections VII and VIII provide a further comparison of the two approaches based on the real-world engineering benchmark problems of Translational Oscil- lators with Rotating Actuator (TORA) and an aeroelastic wing section model. Section IX concludes the paper.

II. Notation and basic concepts

A. Notation

- $i, j, m, n, g \ldots$ are indices with the lower bound 1 and upper bounds $I, J, M, N, G \ldots$.
- $a \in \mathbb{R}$, $\mathbf{a} \in \mathbb{R}^I$, $\mathbf{A} \in \mathbb{R}^{I_1 \times I_2}$, $\mathcal{A} \in \mathbb{R}^{I^N}$ denote, scalars, vectors, matrices and tensors, respectively, where notation $\mathbb{R}^{I^N}$ is equivalent to $\mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$.
- $\mathbf{1}$ denotes a vector whose elements are all 1;
- $rank(\mathbf{A})$ denotes the rank of matrix $\mathbf{A}$;
- $rank_n(\mathcal{A})$ denotes the n-mode rank of tensor $\mathcal{A}$, see [10].
- $[\cdot]_{index}$ addresses elements, e.g. $[\mathcal{A}]_{i_1, i_2, \ldots i_N} = a_{i_1, i_2, \ldots i_N}$ of $\mathcal{A}$;
- $\{\mathcal{A}\}_{(i)}$ denotes the i-mode unfolding of $\mathcal{A}$, see [10]. $\{\mathcal{A}\}_{(i)}$ is a matrix whose columns are the vectors from dimension i of $\mathcal{A}$. Thus the size of $\{\mathcal{A}\}_{(i)}$ is $I_i \times \prod_{n=1, n \neq i}^{N} I_n$.
- $\mathcal{A} = \begin{bmatrix} \mathcal{B}_1 & \mathcal{B}_2 & \ldots & \mathcal{B}_K \end{bmatrix}_n$ represents the fact that tensor $\mathcal{A} \in \mathbb{R}^{I^N}$ is partitioned in the n-th dimension into K different tensors $\mathcal{B}_k \in \mathbb{R}^{I_1 \times \ldots \times I_{n-1} \times J_k \times I_{n+1} \times \ldots \times I_N}$, where $I_n = \sum_{k=1}^{K} J_k$.
- $\mathcal{A} \in co\{\forall n : \mathcal{B}_n\}$ represents the fact that $\mathcal{A}$ is within the convex hull defined by the vertices $\mathcal{B}_n$.
- $\mathbf{U}$ denotes an orthonormed matrix.
- $\mathbf{W}$ denotes the matrix such that:

$$\mathbf{1} = \mathbf{W}\mathbf{1} \quad \text{and} \quad \forall i, j : 0 \leq [\mathbf{W}]_{i,j}. \tag{1}$$

- $\mathbf{U} \rightarrow_{ctr} \mathbf{W}$ represents a convex transformation that transforms the orthonormed matrix $\mathbf{U}$ to $\mathbf{W}$. Examples of

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

such transformations in the literature related to the TS Fuzzy model transformation include Sum Normalisation – Non Negativeness (SNNN), Normalised (NO), Close to Normalised (CNO), Relaxed Normalised (RNO), Inverse Normalised (INO), and the Inverse Relaxed Normalised (IRNO) transformations [1], [2], [11]. These transformations guarantee further characteristics of $\mathbf{W}$ that are advantageous in TS Fuzzy model based design as discussed in the Introduction.

In this paper, $f(\mathbf{p})$ represents a bounded continuous function $f(\mathbf{p}) \in \mathbb{R}^{J^M}$, where $\mathbf{p} \in \Omega \subset \mathbb{R}^N$. Here, Ω is defined by the intervals $\omega_1 \times \omega_2 \times \cdots \times \omega_N$, where $p_n \in \omega_n$, and ω_n denotes the interval $[\omega_n^{min}, \omega_n^{max}]$.

Vector function $\mathbf{w}(p) = \begin{bmatrix} w_1(p) & w_2(p) & \dots & w_I(p) \end{bmatrix}$ has non-negative elements whose sum is equal to 1 for all p:

$$\forall p : \mathbf{w}(p)\mathbf{1} = 1; \quad \forall i, p : 0 \leq [\mathbf{w}(p)]_i. \tag{2}$$

B. Basic concepts

Definition 2.1: **Transfer function of TS Fuzzy model**

Let us consider a set of fuzzy rules in the form:

$$\text{IF } A_{1,i_1} \text{ AND } A_{2,i_2} \dots \text{AND } A_{N,i_N} \text{ THEN } B_{i_1,i_2,\dots i_N}. \tag{3}$$

Here, the membership values of the Ruspini-partitioned antecedent fuzzy sets A_{n,i_n} are defined by $w_{n,i_n}(p_n)$. The consequents $B_{i_1,i_2,\dots i_N}$ can represent scalar, vector, matrix, or even tensor elements, denoted as $\mathcal{S}_{i_1,i_2,\dots i_N} \in \mathbb{R}^{J^M}$ and structured in a tensor $\mathcal{S} \in \mathbb{R}^{I^N \times J^M}$.

The transfer function of the TS Fuzzy model, based on a product-sum-gravity approach and with singleton observations located at p_n, can be expressed as [1]–[3], [9]:

$$f(\mathbf{p}) = \sum_{i_1}^{I_1} \sum_{i_2}^{I_2} \dots \sum_{i_N}^{I_N} \prod_{n=1}^{N} w_{n,i_n}(p_n) [\mathcal{S}]_{i_1,i_2,\dots i_N}. \tag{4}$$

This transfer function with tensor-product operation takes the form of:

$$f(\mathbf{p}) = \mathcal{S} \mathop{\boxtimes}_{n=1}^{N} \mathbf{w}_n(p_n), \tag{5}$$

where $\mathbf{w}_n(p_n) = \begin{bmatrix} w_{n,1}(p_n) & w_{n,2}(p_n) \dots w_{n,I_N}(p_n) \end{bmatrix}$.

The tensor product operation utilized in Equ. (5) is a concept that was introduced in tensor algebra in the year 2000 [10]. Since then, Equ. (5) has been widely employed in the literature on TP model and TS Fuzzy model transformation.

Definition 2.2: **Discretised Tensor $\mathcal{F}$ of function $f(\mathbf{p})$ over grid $\mathcal{G}$**

The discretised tensor $\mathcal{F} \in \mathbb{R}^{G^N \times J^M}$ represents the discretised version of the function $f(\mathbf{p}) \in \mathbb{R}^{J^M}$ over a rectangular grid defined by the grid tensor $\mathcal{G} \in \mathbb{R}^{G^N \times N}$. It can be expressed as $[\mathcal{F}]_{g_1,g_2,\dots g_N} = f([\mathcal{G}]_{g_1,g_2,\dots g_N})$.

Here, the grid tensor $\mathcal{G}$ defines the coordinates of a hyper-rectangular grid as $[\mathcal{G}]_{g_1,g_2,\dots g_N} = \begin{bmatrix} g_{1,g_1} & g_{2,g_2} & \dots & g_{N,g_N} \end{bmatrix} \in \mathbb{R}^N$. It is constructed from the elements of grid vectors $\mathbf{g}_n = \begin{bmatrix} g_{n,1} & g_{n,2} & \dots & g_{n,G_n} \end{bmatrix} \in \mathbb{R}^{G_n}$, $\forall n, i : g_{n,i} < g_{n,i+1}$, defined on each interval ω_n of the N-dimensional hyperspace Ω. The grid covers Ω, meaning that the grid vectors satisfy the conditions , $g_{n,1} = \omega_n^{min}$, and $g_{n,G_n} = \omega_n^{max}$.

Definition 2.3: **Piece-wise Multi-linear approximation of function $f(\mathbf{p})$ over grid $\mathcal{G}$**

The piece-wise multi-linear approximation $\overline{f}(\mathbf{p})$, denoted by a bar on top, is defined using the discretised tensor $\mathcal{F}$ of $f(\mathbf{p})$ over grid $\mathcal{G}$ as:

$$f(\mathbf{p}) \approx \overline{f}(\mathbf{p}) = \mathcal{F} \mathop{\boxtimes}_{n=1}^{N} \mathbf{i}_n(p_n), \tag{6}$$

In this equation, the vector $\mathbf{i}_n(p_n)$ is given by:

$$\mathbf{i}_n(p_n) = \lambda_n [\mathbf{I}]_g + (1 - \lambda_n)[\mathbf{I}]_{g+1}; \quad \lambda_n = \frac{g_{n,g+1} - p_n}{g_{n,g+1} - g_{n,g}}, \tag{7}$$

where $[\mathbf{g}]_g \leq p_n \leq [\mathbf{g}]_{g+1}$, and $[\mathbf{I}]_g$ represents the g-th row of the identity matrix $\mathbf{I}$.

III. Original TS Fuzzy model transformation

The original TS Fuzzy model transformation is built on the following three methods [1]–[3].

Method 3.1: **CHOSVD based piece-wise Multi-linear approximation of function $f(\mathbf{p})$**

Let us determine $\mathcal{F}$ of $f(\mathbf{p})$ over $\mathcal{G}$. We perform the Compact HOSVD (CHOSVD) [10], also known as truncated HOSVD, on $\mathcal{F}$, where "Compact / truncated" means that all the zero singular values and their corresponding columns in the singular matrices are discarded:

$$\mathcal{F} \overset{chosvd}{=\!=} \mathcal{D} \mathop{\boxtimes}_{n=1}^{N} \mathbf{U}_n. \tag{8}$$

Here, $\mathcal{D} \in \mathbb{R}^{R^N \times J^M}$ and $\mathbf{U}_n \in \mathbb{R}^{G_N \times R_N}$, with R_n representing the rank of $\mathcal{F}$ along the n-th dimension $R_n = rank_n(\mathcal{F})$. Substituting Equ. (8) into (6), the approximation is defined as:

$$\overline{f}(\mathbf{p}) = \left(\mathcal{D} \mathop{\boxtimes}_{n=1}^{N} \mathbf{U}_n \right) \mathop{\boxtimes}_{n}^{N} \mathbf{i}_n(p_n) = \mathcal{D} \mathop{\boxtimes}_{n=1}^{N} \overline{\mathbf{u}}_n(p_n), \tag{9}$$

where $\overline{\mathbf{u}}_n(p_n) = \mathbf{i}_n(p_n)\mathbf{U}_n$. Because of Equ. (8) $f(\mathbf{p}) = \overline{f}(\mathbf{p})$ over the grid $\mathcal{G}$.

Method 3.2: **Accuracy-complexity trade-off**

The CHOSVD of $\mathcal{F}$ is determined via the execution of SVD for each dimension as follows:

$$\{\mathcal{F}\}_{(n)} \overset{svd}{=\!=} \mathbf{U}_n \mathbf{D}_n \mathbf{V}_n^T \tag{10}$$

Diagonal matrix $\mathbf{D}_n$ contains the singular values in decreasing order as follows:

$$diag(\mathbf{D}_n) = [\sigma_{n,1} \geq \sigma_{n,2} \geq \dots \geq \sigma_{n,R_n}]. \tag{11}$$

R_n – the number of singular values – determines the absolute minimum number of antecedent fuzzy sets in dimension n. Finally, the decomposition leads to:

$$\mathcal{F} = \mathcal{D} \mathop{\boxtimes}_{n=1}^{N} \mathbf{U}_n \quad \text{where} \quad \mathcal{D} = \mathcal{F} \mathop{\boxtimes}_{n=1}^{N} (\mathbf{U}_n)^+. \tag{12}$$

If further non-zero singular values and the corresponding columns of the singular matrices $\mathbf{U}_n$ are discarded then the following approximation error results:

$$\mathcal{F} \underset{\varepsilon}{\approx} \widehat{\mathcal{F}} = \mathcal{D} \mathop{\boxtimes}_{n=1}^{N} \mathbf{U}_n \tag{13}$$

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

where ε expresses the L_2 norm error, which is bounded by the sum of the discarded singular values, see [10]. Note that this is not the best approximation of tensor $\mathcal{F}$ under the decreased rank constraint if $n > 2$, see [10].

Method 3.3: **Determination of the Ruspini-partitioned antecedent fuzzy sets**

Each column of $\mathbf{U}_n$ determines a candidate for one antecedent fuzzy set. Each element in a given column defines the values of the possible antecedent fuzzy sets over the grid $\mathbf{g}_n$. In order to determine the values of the Ruspini-partitioned antecedent fuzzy set over the grid, we can apply one of many different kinds of convex transformations to $\mathbf{U}_n$ to derive $\mathbf{W}_n$:

$$\mathbf{U}_n \underset{ctr}{\rightarrow} \mathbf{W}_n. \tag{14}$$

These transformations guarantee various advantageous features of the resulting convex hull defined by the consequents. This leads to

$$\widehat{\mathcal{F}} = \mathcal{S} \mathop{\boxtimes}_{n=1}^{N} \mathbf{W}_n \quad \text{where} \quad \mathcal{S} = \mathcal{F} \mathop{\boxtimes}_{n=1}^{N} (\mathbf{W}_n)^+ . \tag{15}$$

Here, $\mathbf{W}_n$ possesses the properties defined in Equ. (1). Therefore each column of $\mathbf{W}_n$ determines one antecedent fuzzy set over the grid and for all inputs as:

$$\overline{\mathbf{w}}_n(p_n) = \mathbf{i}(p_n)\mathbf{W}_n. \tag{16}$$

It is important to note that the convex transformations may add one column to $\mathbf{U}_n$, resulting in $\mathbf{W}_n$. Therefore, in this case, $\mathcal{S} \in \mathbb{R}^{I^N \times J^M}$ and $\mathbf{W}_n \in \mathbb{R}^{G_N \times I_N}$, where

$$\forall n : rank_n(\widehat{\mathcal{F}}) \leq I_n \leq rank_n(\widehat{\mathcal{F}}) + 1. \tag{17}$$

Finally we arrive at the piece-wise Multi-linear TS Fuzzy model as

$$\overline{f}(\mathbf{p}) = \left(\mathcal{S} \mathop{\boxtimes}_{n=1}^{N} \mathbf{W}_n \right) \mathop{\boxtimes}_{n=1}^{N} \mathbf{i}_n(p_n) = \mathcal{S} \mathop{\boxtimes}_{n=1}^{N} \overline{\mathbf{w}}_n(p_n). \tag{18}$$

This is a convex form meaning that:

$$\forall \mathbf{p} : \overline{f}(\mathbf{p}) \in co\{\forall i_1, i_2, \dots i_N : [\mathcal{S}]_{i_1,i_2,\dots i_N}\}. \tag{19}$$

The L_2 norm error γ in $f(\mathbf{p}) \approx_\gamma \overline{f}(\mathbf{p})$ has two components:

$$\gamma = \varepsilon + \beta, \tag{20}$$

where ε is the approximation error over the grid caused by the rank reduction, as mentioned above, and β is caused by the piece-wise linear approximation of the antecedent fuzzy sets between the grid. When the grid density approaches infinity then $\beta \rightarrow 0$. The high resolution of antecedent fuzzy sets (the grid density over which the antecedents are defined) can be further enhanced as detailed in [7]. Therefore, $\beta \rightarrow 0$ is well supported and the paper focuses only on ε from this point on.

IV. Improved HOSVD-based tensor rank reduction

Before discussing the method used to achieve an improved accuracy and complexity trade-off, let us first introduce the following definitions.

Definition 4.1: **Block identical tensor $\mathcal{F}^b$**

A tensor $\mathcal{F}^b \in \mathbb{R}^{G^N \times J^M}$ is referred to as a block identical tensor, denoted by the superscript "b", if it is constructed from identical blocks $\mathcal{C}$, such that

$$\forall g_1, g_2, \dots g_N : [\mathcal{F}^b]_{g_1,g_2,\dots g_N} = \mathcal{C} \in \mathbb{R}^{J^M}. \tag{21}$$

Lemma 4.1: Consider block identical tensor $\mathcal{F}^b \in \mathbb{R}^{G^N \times J^M}$ constructed from block tensor $\mathcal{C}$ as Equ. (21). Any convex combination of tensor blocks $\mathcal{C}$ stored in block identical tensor $\mathcal{F}^b$ leads to a block identical tensor $\mathcal{F}'^b \in \mathbb{R}^{G'^N \times J^M}$ whose tensor blocks are $\mathcal{C}$:

$$\forall G_n, G'_n, \mathbf{W}_n \in \mathbb{R}^{G'_n \times G_n} : \mathcal{F}'^b = \mathcal{F}^b \mathop{\boxtimes}_{n=1}^{N} \mathbf{W}_n, \tag{22}$$

thus $\forall g_1, g_2, \dots g_N, g'_1, g'_2, \dots g'_N$:

$$[\mathcal{F}^b]_{g_1,g_2,\dots g_N} = [\mathcal{F}'^b]_{g'_1,g'_2,\dots g'_N} = \mathcal{C}. \tag{23}$$

Definition 4.2: **Partially block identical tensor $\mathcal{F}^p$**

A tensor $\mathcal{F}^p \in \mathbb{R}^{G^N \times J^M}$ is referred to as partially block identical, denoted by the superscript "p", if it is not block identical, but some of the elements of the blocks $[\mathcal{F}^p]_{g_1,g_2,\dots g_N}$ are identical for all $g_1, g_2, \dots g_N$, given by

$$\exists j_1, j_2, \dots j_M : [[\mathcal{F}^p]_{g_1,g_2,\dots g_N}]_{j_1,j_2,\dots j_M} = c_{j_1,j_2,\dots j_M}. \tag{24}$$

Definition 4.3: **Vector block variant tensor $\mathcal{F}^v$**

The rearranged variant $\mathcal{F}^v \in \mathbb{R}^{G^N \times L}$, denoted by the superscript "v", of tensor $\mathcal{F} \in \mathbb{R}^{G^N \times J^M}$ is constructed by rearranging the blocks $[\mathcal{F}]_{g_1,g_2,\dots g_N}$ of $\mathcal{F}$ into vectors of length L. This is given by

$$[[\mathcal{F}^v]_{g_1,g_2,\dots g_N}]_l = [[\mathcal{F}]_{g_1,g_2,\dots g_N}]_{j_1,j_2,\dots j_M} \tag{25}$$

where $L = \prod_{m=1}^{M} J_m$ and the ordering of the elements is defined by l that represents a linear index equivalent to the array index $j_1, j_2, \dots j_M$ as

$$l = ordering(j_1, j_2, \dots j_M). \tag{26}$$

Method 4.1: **Separation of identical elements**

Let us consider a partial block identical tensor $\mathcal{F}^p$. We can define the ordering in Equ. (26) in such a way that the resulting vectors of $\mathcal{F}^v$ can be partitioned into two vectors as

$$[\mathcal{F}^v]_{g_1,g_2,\dots g_N} = \begin{bmatrix} \mathbf{f}^\beta_{g_1,g_2,\dots g_N} & \mathbf{c} \end{bmatrix}, \tag{27}$$

where vector $\mathbf{c} \in \mathbb{R}^K$ contains the K number of elements that are identical for all $g_1, g_2, \dots g_N$. Consequently, tensor $\mathcal{F}^v$ can be partitioned in dimension $N+1$ into two tensors as

$$\mathcal{F}^v = \begin{bmatrix} \mathcal{F}^\beta & \mathcal{C}^b \end{bmatrix}_{N+1}, \tag{28}$$

where $\mathcal{F}^\beta \in \mathbb{R}^{I^N \times (L-K)}$ stores the non identical elements, and the block identical tensor $\mathcal{C}^b \in \mathbb{R}^{G^N \times K}$ stores the elements $[\mathcal{C}^b]_{g_1,g_2,\dots g_N} = \mathbf{c}$.

Method 4.2: **HOSVD-based rank reduction of partially block identical tensors**

Consider a partially block identical tensor $\mathcal{F}^p$. Furthermore, let its vector block variant $\mathcal{F}^v$ be derived and partitioned as shown in Equ. (28). The CHOSVD of $\mathcal{F}^\beta$ results in

$$\mathcal{F}^\beta = \mathcal{S}' \mathop{\boxtimes}_{n=1}^{N} \mathbf{U}^\beta_n. \tag{29}$$

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

Here, $\mathcal{S}' \in \mathbb{R}^{(R^\beta)^N \times (L-K)}$ and $\mathbf{U}'_n \in \mathbb{R}^{G_N \times R^\beta_N}$, where $R^\beta_n = rank_n(\mathcal{F}^\beta) \leq R_n = rank_n(\mathcal{F}^p)$. Based on Method 3.2 the CHOSVD is computed by executing SVD in each dimension as follows:

$$\{\mathcal{F}^\beta\}_{(n)} \overset{svd}{=\!=} \mathbf{U}^\beta_n \mathbf{D}^\beta_n \left(\mathbf{V}^\beta_n\right)^T \tag{30}$$

Diagonal matrix $\mathbf{D}^\beta_n$ contains the singular values in decreasing order as follows:

$$diag(\mathbf{D}^\beta_n) = [\sigma^\beta_{n,1} \geq \sigma^\beta_{n,2} \geq \ldots \geq \sigma^\beta_{n,R^\beta_n}]. \tag{31}$$

Because the elements of $\mathcal{C}^b$ are excluded from $\mathcal{F}^\beta$ the following properties hold for the singular values:

$$\exists n, r^\beta : \sigma^\beta_{n,r^\beta} \leq \sigma_{n,r^\beta}. \tag{32}$$

If non-zero singular values and the corresponding singular vectors of $\mathbf{U}^\beta_n$ are discarded then an approximation error ε^β results, however, it is bounded by the sum of the discarded singular values. Let us execute a convex transformation as follows:

$$\forall n : \mathbf{U}^\beta_n \underset{ctr}{\rightarrow} \mathbf{W}^\beta_n \tag{33}$$

which leads to:

$$\mathcal{F}^\beta \underset{\varepsilon^\beta}{\approx} \widehat{\mathcal{F}}^\beta = \mathcal{S}' \mathop{\boxtimes}_{n=1}^{N} \mathbf{W}^\beta_n \quad \text{where} \quad \mathcal{S}' = \mathcal{F}^\beta \mathop{\boxtimes}_{n=1}^{N} \left(\mathbf{W}^\beta_n\right)^+. \tag{34}$$

Here, $\mathcal{S}' \in \mathbb{R}^{(I^\beta)^N \times L}$ and $\mathbf{W}^\beta_n \in \mathbb{R}^{G_n \times I^\beta_n}$, where $\forall n$: $rank_n(\widehat{\mathcal{F}}^\beta) \leq I^\beta_n \leq rank_n(\widehat{\mathcal{F}}^\beta) + 1$.

The convex transformation does not affect the error and because of Equ. (32), finally we have a smaller upper bound for ε^β when smaller singular values $\sigma^\beta < \sigma$ are discarded.

In order to reinsert the elements of the excluded $\mathcal{C}^b$, let us create a block identical tensor $\mathcal{C}^\beta \in \mathbb{R}^{(I^\beta)^N \times K}$ from vector blocks $\mathbf{c}$ as follows:

$$\forall i^\beta_1, i^\beta_2, \ldots i^\beta_N : [\mathcal{C}^\beta]_{i^\beta_1, i^\beta_2, \ldots i^\beta_N} = \mathbf{c}. \tag{35}$$

Based on Lemma 4.1:

$$\mathcal{C}^b = \mathcal{C}^\beta \mathop{\boxtimes}_{n=1}^{N} \mathbf{W}^\beta_n. \tag{36}$$

By substituting Equ. (29) and (36) into Equ. (28) we obtain

$$\widehat{\mathcal{F}}^v = \left[\mathcal{S}' \mathop{\boxtimes}_{n=1}^{N} \mathbf{W}^\beta_n \quad \mathcal{C}^\beta \mathop{\boxtimes}_{n=1}^{N} \mathbf{W}^\beta_n\right] = \mathcal{S}^v \mathop{\boxtimes}_{n=1}^{N} \mathbf{W}^\beta_n, \tag{37}$$

where

$$\mathcal{S}^v = \left[\mathcal{S}' \quad \mathcal{C}^\beta\right]_{N+1} \in \mathbb{R}^{(I^\beta)^N \times L}. \tag{38}$$

If we rearrange the core tensor $\mathcal{S}^v$ using Equ. (26) as follows:

$$[[\mathcal{S}^\beta]_{i'_1, i'_2, \ldots i'_N}]_{j_1, j_2, \ldots j_M} = [[\mathcal{S}^v]_{i'_1, i'_2, \ldots i'_N}]_l, \tag{39}$$

then we arrive at:

$$\mathcal{F}^p \underset{\varepsilon^\beta}{\approx} \widehat{\mathcal{F}}^p = \mathcal{S}^\beta \mathop{\boxtimes}_{n=1}^{N} \mathbf{W}^\beta_n, \tag{40}$$

where $\mathcal{S}^\beta \in \mathbb{R}^{(I^\beta)^N \times J^M}$.

An alternative way of determining $\mathcal{S}^\beta$ is to directly derive it from $\mathcal{F}^p$ once we have matrices $\mathbf{W}^\beta_n$. Therefore, instead of Equ. (34-40), we can calculate:

$$\mathcal{S}^\beta = \mathcal{F}^p \mathop{\boxtimes}_{n=1}^{N} \left(\mathbf{W}^\beta_n\right)^+. \tag{41}$$

As a matter of fact, in this case there is no guarantee that the partially block identical elements will be the same or will even remain partially block identical, especially when a numerical computational error occurs.

The conclusion of this section is that if we separate the partially identical elements then the CHOSVD will find smaller singular values which leads to an improved complexity trade-off. Because of the convex form, the separated elements can be reinserted after the trade-off is performed.

V. Introducing the Relaxed TS Fuzzy model transformation

Based on the improved tensor rank reduction introduced in the previous section, this section proposes the Relaxed TS Fuzzy model transformation that is an extension of the original TS Fuzzy model transformation based on two additional steps. Figure 1 illustrates the block diagram of the Relaxed TS Fuzzy model transformation, which will be detailed in subsequent sections. For comparison, the original TS Fuzzy model transformation is also depicted in Figure 1. The blocks that are vertically aligned along the left-hand side of the figure reflect the original version, while the two additional blocks on the right show the extensions that make up the Relaxed TS Fuzzy model transformation. Furthermore, the equations delineating the crucial steps are provided in Figure 1.

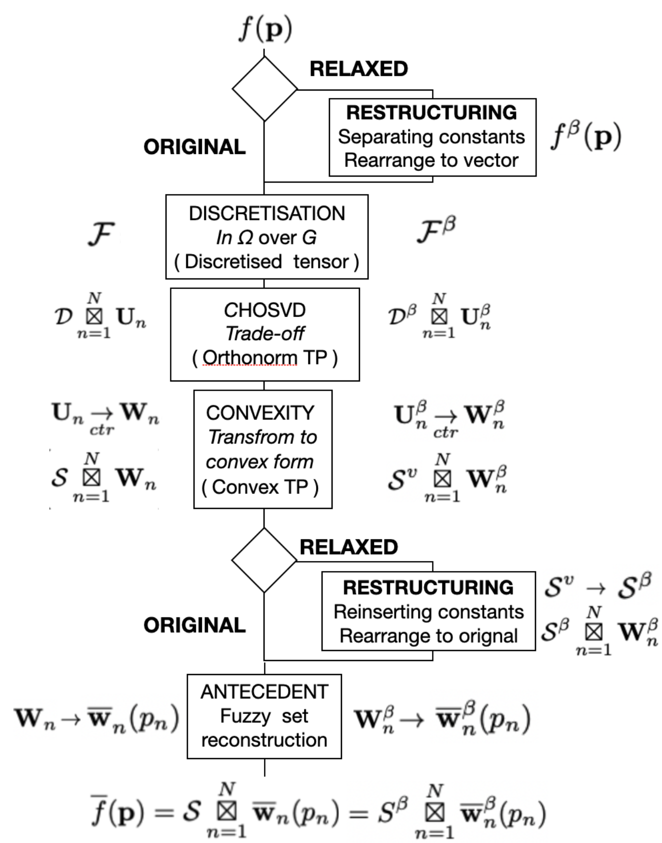

Fig. 1: The block diagram of the algorithms of the original and the Relaxed TS Fuzzy model transformation.

Method 5.1: **Relaxed TP model transformation**

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

Assume function $f(\mathbf{p})$ is given in a closed form, such that all of the the inner formulae of the function are known. Then, the following Relaxed TS Fuzzy model transformation can be executed.

Step 0 (additional step to the original TS Fuzzy model transformation)

Define $f^{\beta}(\mathbf{p})$ by rearranging the parameter-dependent elements of $f(\mathbf{p})$ into a vector and excluding the constant elements as

$$f^{v}(\mathbf{p}) = \begin{bmatrix} f^{\beta}(\mathbf{p}) & \mathbf{c} \end{bmatrix}. \tag{42}$$

Step 1: Discretisation

Define the discretisation grid based on G_n. Then derive the discretised tensor $\mathcal{F}^{\beta}$ of $f^{\beta}(\mathbf{p})$.

Step 2: Approximation and complexity trade-off

Execute the CHOSVD and perform further rank reduction – if required – by discarding non-zero singular values. This step results in:

$$\widehat{\mathcal{F}}^{\beta} = \mathcal{D}^{\beta} \boxtimes_{n=1}^{N} \mathbf{U}_n^{\beta}. \tag{43}$$

Step 3: Convex transformation

Execute a convex transformation as:

$$\mathbf{U}_n^{\beta} \underset{ctr}{\rightarrow} \mathbf{W}_n^{\beta}. \tag{44}$$

Step 4: Restructuring the core tensor and reinsertion of the partially identical elements (additional step to the original TS Fuzzy model transformation).

There are two possible approaches here, as discussed in the context of Equ. (41). One way is to derive the core tensor, see Equ. (34) as

$$\mathcal{S}' = \mathcal{F}^{\beta} \boxtimes_{n=1}^{N} \left(\mathbf{W}_n^{\beta}\right)^{+}. \tag{45}$$

Then, one can reinsert the separated constants as follows:

$$\mathcal{S}^{v} = \begin{bmatrix} \mathcal{S}' & \mathcal{C}^{\beta} \end{bmatrix}_{N+1} \in \mathbb{R}^{(I^{\beta})^{N} \times L}. \tag{46}$$

see Equ. (38). Here, $\mathcal{C}$ is defined by Equ. (35). Then, restructuring the core tensor $\mathcal{S}^{v}$ – counter to Step 0 – leads to $\mathcal{S}^{\beta}$ as follows:

$$\mathcal{F} \underset{\varepsilon^{\beta}}{\approx} \mathcal{S}^{\beta} \boxtimes_{n=1}^{N} \mathbf{W}_n^{\beta}. \tag{47}$$

The alternative way is to directly calculate the elements of the core tensor based on Equ. (41) as follows:

$$\mathcal{F} \underset{\varepsilon^{\beta}}{\approx} \mathcal{S}^{\beta} \boxtimes_{n=1}^{N} \mathbf{W}_n^{\beta}, \quad \text{where} \quad \mathcal{S}^{\beta} = \mathcal{F} \boxtimes_{n=1}^{N} \left(\mathbf{W}_n^{\beta}\right)^{+}. \tag{48}$$

This does not guarantee that the elements will be partially identical as mentioned in the context of Equ. (41).

Step 5: Determine the piece-wise linear antecedent fuzzy sets

The antecedent fuzzy set are determined over the grid $\mathbf{g}_n$ as follows:

$$\overline{\mathbf{w}}_n^{\beta}(p_n) = \mathbf{i}_n(p_n)\mathbf{W}_n^{\beta}. \tag{49}$$

Thus:

$$\overline{f}(\mathbf{p}) = S^{\beta} \boxtimes_{n=1}^{N} \overline{\mathbf{w}}_n^{\beta}(p_n). \tag{50}$$

The density of the discretisation grid in Step 1 is limited by the available computational capacity. In order to reveal all the ranks, a high-density grid is required. However, Step 2 executes HOSVD on the discretised tensor, which is computationally expensive. The overall computational complexity of the HOSVD of tensor $\mathcal{F} \in \mathbb{R}^{G^N \times J^M}$ is detailed in [10] and described as:

$$O\left(\sum_{n=1}^{N}\left(G_n \prod_{k=1}^{N} G_k \prod_{m=1}^{M} J_m\right)\right). \tag{51}$$

One can see that when N increases, G_n is strictly limited. The Relaxed TS Fuzzy model transformation restructures the elements into vectors and removes the constant elements. If K different elements are removed, see Equ. (27), then the computational complexity is reduced to:

$$O\left((L-K)\sum_{n=1}^{N}\left(G_n \prod_{k=1}^{N} G_k\right)\right), \tag{52}$$

where $L = \prod_{m=1}^{M} J_m$, see Equ. (25).

Another aspect is that the resolution (the grid density over which the antecedents are defined) of the piece-wise linear antecedent fuzzy sets is also determined by G_n, see Step 5. Thus, once the number of elements are decreased in the tensor, the grid density can be increased, which will lead to the improved resolution of the antecedent fuzzy sets as well. If further improvement of the resolution of the antecedents is needed, then the refining technique introduced in Section V of the recently published paper [7] can be applied. The end result is that the error β can be eliminated in a numerical sense.

Remark 5.1: The Relaxed TS Fuzzy model transformation does not yield a computational reduction in the absence of constant elements that can be separated. Its application is straightforward when the internal formulations of the provided functions are visible, allowing for the easy separation of constant elements. If the constant and non-constant components are obscured, for instance when a black-box form of the functions is given, it becomes necessary to use an additional algorithm to uncover the partially block-identical elements.

VI. Demonstrative Examples

The primary objective behind the examples presented in this Section is to show that the proposed relaxed TS Fuzzy model transformation offers better complexity and accuracy trade-offs than the original TS Fuzzy model transformation. Furthermore, this example serves as evidence that the proposed transformation requires less computational resources.

A. Example 1

Consider the tensor function $f(p_1, p_2) = \begin{bmatrix} \mathbf{A}(p_1,p_2) & \mathbf{B}(p_1,p_2) \end{bmatrix}_3 \in \mathbb{R}^{2\times3\times2}$, where $\mathbf{p} \in \Omega = [1,100]\times[1,100]$ and

$$\mathbf{A}(p_1,p_2) = \begin{bmatrix} p_1 & h & -10^3 h \\ p_2 & p_1p_2 & (p_1+p_2)^2 \end{bmatrix}; \tag{53}$$

$$\mathbf{B}(p_1,p_2) = \begin{bmatrix} 1 & p_2 & 10^2 h \\ 10h & (p_1+p_2)^2 & p_1p_2 \end{bmatrix}; \tag{54}$$

with $h = 1$. Let us begin by applying the original TS Fuzzy model transformation directly to the tensor function $f(\mathbf{p})$. In

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

the examples we utilise the TS Fuzzy model transformation and convex transformations available in the TS Fuzzy model transformation MATLAB ToolBox available on the Wikipedia site of the TP model transformation.

We define a $G_1 = G_2 = 137$ equidistant rectangular grid that covers the domain Ω. The first step of the transformation (CHOSVD) yields the TP structure (singular values less than 10^{-9} are discarded):

$$\mathcal{F} \underset{\varepsilon}{\approx} \mathcal{D} \underset{n=1}{\overset{2}{\boxtimes}} \mathbf{U}_n, \tag{55}$$

where $\mathcal{F} \in \mathbb{R}^{137\times137\times2\times3\times2}$, $\mathcal{D} \in \mathbb{R}^{6\times4\times2\times3\times2}$, $\mathbf{U}_1 \in \mathbb{R}^{137\times6}$, $\mathbf{U}_1 \in \mathbb{R}^{137\times4}$ and $\varepsilon = 1.2303\times10^{-9}$. The singular values are:

Dimension assigned to p_1*:* 2.1134×10^6; 1.614×10^5; 3.38×10^4; 2.3454×10^{-9}; 1.6608×10^{-9}; 1.0458×10^{-9}; $\ldots$.

Dimension assigned to p_2*:* 2.1134×10^6; 1.613×10^5; 3.38×10^4; 1.446×10^{-9}; $\ldots$.

To ensure a convex TP structure, the SNNN transformation is applied to Equ. (55). In the current scenario, the SNNN transformation does not increase the number of columns in $\mathbf{W}_n$. Therefore we have

$$\mathcal{F} \underset{\varepsilon}{\approx} \mathcal{S} \underset{n=1}{\overset{2}{\boxtimes}} \mathbf{W}_n. \tag{56}$$

where $\mathcal{S} \in \mathbb{R}^{6\times4\times2\times3\times2}$, $\mathbf{W}_1 \in \mathbb{R}^{137\times6}$ and $\mathbf{W}_2 \in \mathbb{R}^{137\times4}$.

We proceed with the proposed Relaxed TS Fuzzy model transformation. In Step 0, the given function is relaxed by rearranging its elements into a vector, excluding the constant elements $f^\beta(p_1,p_2) = \begin{bmatrix} p_1 & p_2 & p_1p_2 & (p_1+p_2)^2 \end{bmatrix}$.

Moving on to Step 1 of the proposed method, CHOSVD is executed on the discretized tensor $\mathcal{F}^\beta \in \mathbb{R}^{137\times137\times4}$, resulting in $\mathcal{F}^\beta \approx_{\varepsilon^\beta} \mathcal{D}^\beta \underset{n=1}{\overset{2}{\boxtimes}} \mathbf{U}_n^\beta$, where $\mathcal{D}^\beta \in \mathbb{R}^{3\times3\times2\times3\times2}$, $\mathbf{U}_n^\beta \in \mathbb{R}^{137\times3}$ and $\varepsilon^\beta = 8.5867\times10^{-10}$. Note that $\varepsilon^\beta < \varepsilon$ and the size of $\mathcal{D}^\beta$ is smaller than the size of $\mathcal{D}$ in the first two dimensions. The singular values obtained from this transformation are:

Dimensions assigned p_1 *and* p_2 *(they are same in the current scenario):* 1.493×10^6; 1.102×10^5; 2.36×10^4; $\ldots$. One can observe that the resulting singular values are considerably smaller than in the case of Equ. (55). When performing the SNNN transformation on $\mathbf{U}_n^\beta$ to obtain $\mathbf{W}_n^\beta$, it is observed that the transformation does not increase the number of columns in $\mathbf{W}_n^\beta$. Consequently, we have:

$$\mathcal{F}^\beta \underset{\varepsilon^\beta}{\approx} \mathcal{S}' \underset{n=1}{\overset{2}{\boxtimes}} \mathbf{W}_n^\beta, \tag{57}$$

where $\mathcal{S}^\beta \in \mathbb{R}^{3\times3\times4}$ and $\mathbf{W}_n^\beta \in \mathbb{R}^{137\times3}$. Finally we restructure the core tensor as:

$$[\mathcal{S}^\beta]_{i_1,i_2,:,:,1} = \begin{bmatrix} [\mathcal{S}']_{i_1,i_2,1} & h & -10^3h \\ [\mathcal{S}']_{i_1,i_2,2} & [\mathcal{S}']_{i_1,i_2,3} & [\mathcal{S}']_{i_1,i_2,4} \end{bmatrix}. \tag{58}$$

$$[\mathcal{S}^\beta]_{i_1,i_2,:,:,2} = \begin{bmatrix} 1 & [\mathcal{S}']_{i_1,i_2,2} & 10^2h \\ 10h & [\mathcal{S}']_{i_1,i_2,4} & [\mathcal{S}']_{i_1,i_2,3} \end{bmatrix}. \tag{59}$$

which leads to $\mathcal{F} \approx_{\varepsilon^\beta} \mathcal{S}^\beta \underset{n=1}{\overset{2}{\boxtimes}} \mathbf{W}_n$. Once again, the size of the core tensor $\mathcal{S}^\beta$ is considerably (37.5%) smaller ($3\times3\times2\times3\times2 = 108$) than the size of $\mathcal{S}$ ($6\times4\times2\times3\times2 = 288$) and, further, the resulting approximation error obtained via the Relaxed TS Fuzzy model transformation over the grid is also smaller, as $\varepsilon = 1.2303\times10^{-9} > \varepsilon^\beta = 8.5867\times10^{-10}$. This means that the number of fuzzy rules is decreased from $6\times4 = 24$ to $3\times3 = 9$, i.e., by 62.5% with better approximation accuracy over the grid.

Let us investigate the computational complexity. The CHOSVD is executed on a tensor with the size of $137\times137\times2\times3\times2$ that leads to 62×10^6 operational steps, see Equ. (51), in the case of the original TS Fuzzy model transformation, while in the case of the Relaxed TS Fuzzy model transformation the size of the tensor is only $137\times137\times4$, which leads to 20×10^6 operational steps, see Equ. (52). This is a reduction of around 67%.

Let us proceed by evaluating the complexity and accuracy trade-off. The results are summarized in Table I and II. Table I shows the L_2 norm error over the grid resulting from the original and the Relaxed TS Fuzzy model transformation. The first column denoted by R_1 and the first row denoted by R_2 show the kept number of singular values assigned to dimensions p_1 and p_2, respectively. The column denoted by I_1 and the row denoted by I_2 show the number of the antecedent fuzzy sets over dimensions p_1 and p_2, respectively, resulting from the convex transformation, see Equ. (56). Each cell is partitioned into two cells. The left one shows L_2 norm error ε while the right one shows the number of the resulting fuzzy rules $\#R$ as denoted in the third row. The 10th row denoted by R_2^β and the column denoted by R_1^β show the number of singular values that are kept when the Relaxed TS Fuzzy model transformation is executed. The 11th row denoted by I_2^β and its corresponding column denoted by I_1^β show the number of the antecedent fuzzy sets resulting from the convex transformation, see Equ. (57). The Tables demonstrate well that the greater the number of singular values that are kept, the better the approximation that results. Table II compares the result of the original and the Relaxed TS Fuzzy model transformation. The column denoted by $\varepsilon\%$ shows the reduction of the approximation error as $(1-\varepsilon^\beta/\varepsilon)\%$ and the column denoted by $R\%$ shows the reduction of the fuzzy rules as $(1-\#R^\beta/\#R)\%$. We can observe that a considerably better trade-off results from the Relaxed TS Fuzzy model transformation in contrast to the original TS Fuzzy model transformation.

For instance, the fourth column and the fifth row of Table I show that if we keep two singular values $R_1 = R_2 = 2$ in both dimensions, then the convex transformation increases the number of antecedent fuzzy sets to $I_1 = I_2 = 3$. The resulting TS Fuzzy model has 9 fuzzy rules and an L_2 norm error of 8×10^{-9} over the grid. One can see that in this case, the Relaxed TS Fuzzy model also results in 9 fuzzy rules, but with an L_2 norm approximation error of only 1.8×10^{-9}. Table II shows that this corresponds to a 77% reduction in approximation error.

In summary, the tables demonstrate that the Relaxed TS Fuzzy model leads to a considerable approximation error reduction in all variations and results in a significant fuzzy rule base reduction in many cases. Therefore, the Relaxed

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

	R_2	1		2,3		4	
	I_2	2		3		4	
R_1	I_1	ε	#R	ε	#R	ε	#R
1	2	39642	4	39517	6	39517	8
2,3	3	30811	6	8e-9	9	5e-9	12
4	4	30811	8	6e-9	12	5e-9	16
5	5	30811	10	6e-9	15	5e-9	20
6	6	30811	12	5e-9	18	4e-9	24
	R_2^β	1		2,3			
	I_2^β	2		3			
R_1^β	I_1^β	ε	#R	ε	#R		
1	2	28032	4	27944	6		
2,3	3	21786	6	1.8e-9	9		

TABLE I: Complexity and accuracy trade-off with the original and the Relaxed TS Fuzzy model transformation (Example 1).

TS Fuzzy model transformation provides significantly better approximation and complexity trade-offs than the original TS Fuzzy model transformation. Further, the computational complexity is also reduced by 67%.

Obviously, when h increases, this means that the L_2 norm of the separated constant elements also increases, and the comparison will be even more favourable to the Relaxed TS Fuzzy model transformation. Table III shows a case in which $h = 10000$. The result of the Relaxed TS Fuzzy model transformation is the same as above since the value of the constant element h is excluded in the computation. Table IV shows the approximation error reduction. Comparing Table IV to II, we can observe that the approximation error reduction increases with h.

			R_2	1		2,3		4	
			I_2	2		3		4	
			R_2^β	1		2,3		3	
			I_2^β	2		3		3	
R_1	I_1	R_1^β	I_1^β	$\varepsilon\%$	$R\%$	$\varepsilon\%$	$R\%$	$\varepsilon\%$	$R\%$
1	2	1	2	29	0	29	0	29	25
2,3	3	2,3	3	29	0	77	0	70	25
4	4	3	3	29	25	77	25	76	44
5	5	3	3	29	40	77	40	76	55
6	6	3	3	29	50	72	50	70	63

TABLE II: Reduction of the approximation error and the number of fuzzy rules (Example 1).

	R_2	1		2		3		4	
	I_2	2		2		3		4	
R_1	I_1	ε	#R	ε	#R	ε	#R	ε	#R
1	2	993860	4	895260	6	895260	6	895260	8
2	2	893940	6	39528	9	39383	9	39383	12
3	3	893940	6	29623	9	8e-7	9	2e-6	12
4	4	893940	8	29623	12	6e-7	12	2e-6	16
5	5	893940	10	29623	15	5e-7	15	1e-6	20
6	6	893940	12	29623	18	1e-6	18	5e-7	24

TABLE III: Complexity and accuracy trade-off with the original transformation when $h = 10000$ (Example 1).

			R_2	1		2		3		4	
			I_2	2		3		3		4	
			R_2^β	1		2		3		3	
			I_2^β	2		3		3		3	
R_1	I_1	R_1^β	I_1^β	$\varepsilon\%$	$R\%$	$\varepsilon\%$	$R\%$	$\varepsilon\%$	$R\%$	$\varepsilon\%$	$R\%$
1	2	1	2	97	0	97	0	97	0	97	25
2	3	2	3	97	0	100	0	100	0	100	25
3	3	3	3	97	0	100	0	100	0	100	25
4	4	3	3	97	25	100	25	100	0	100	44
5	5	3	3	97	40	100	40	100	40	100	55
6	6	3	3	97	50	100	50	100	50	100	63

TABLE IV: Reduction of the approximation error and the number of fuzzy rules when $h = 10000$ (Example 1).

B. Example 2

Consider the function frequently employed in the TP model transformation literature [1]–[3] $f_a(p_1,p_2) = (1 + p_1^{-2} + p_2^{-1.5})^2$, where $p_1, p_2 \in [1,5] \subset \mathbb{R}$. We construct a tensor function as: $f(p_1,p_2) = \begin{bmatrix}\mathbf{A}(p_1,p_2) & \mathbf{B}(p_1,p_2)\end{bmatrix}_3 \in \mathbb{R}^{2\times2\times2}$,

$$\mathbf{A}(\mathbf{p}) = \begin{bmatrix} 1 & 0 \\ f_a(\mathbf{p}) & 10 \end{bmatrix}, \quad \mathbf{B}(\mathbf{p}) = \begin{bmatrix} f_a(\mathbf{p}) & -10 \\ 0 & 1 \end{bmatrix} \tag{61}$$

Let us execute the accuracy and complexity trade-off with the original and the Relaxed TS Fuzzy model transformation as described in Example 1. The singular values resulting from the original TS Fuzzy model transformation are:

Dimension assigned to p_1*:* 1461.3; 100.18; 0.98.

Dimension assigned to p_2*:* 1461.4; 98.47; 0.95.

The singular values resulting from the Relaxed TS Fuzzy model transformation are:

Dimension assigned to p_1 *and* p_2*:* 250.1711; 12.16; 0.087.

Again, one can observe that the singular values are considerably smaller in the case of the Relaxed TS Fuzzy model transformation. Note that the number of elements of the discretised tensor upon which CHOSVD is executed in the case of the original TS Fuzzy model transformation is $137 \times 137 \times 2 \times 2 \times 2 = 150152$, which leads to a complexity of 41×10^6 operational steps, see Equ. (51), while in the case of the Relaxed TS Fuzzy model transformation, it has 18769 elements, which leads to a complexity of 5×10^6 operational steps, see Equ. (52). This is a reduction of about 88%. The results are summarized in Table V. The table demonstrates that the Relaxed TS Fuzzy model transformation provides a considerably improved approximation and complexity trade-off. For instance, if we decrease the number of fuzzy rules to 3×2 using the original TS Fuzzy model transformation, then the resulting L_2 norm error is 0.9685, while the Relaxed TS Fuzzy model transformation results in an L_2 norm error of only 0.6833.

VII. Example based on an engineering benchmark problem

In the literature related to TS Fuzzy model transformations, the real-word benchmark example of the Translational Oscillators with Rotating Actuator (TORA) often appears [1]–[3], [34]. It is an underactuated system which has one actuated rotor and one unactuated translational cart. For comparability, we also use this example in this paper.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

	R_2	1		2		3	
	I_2	2		3		3	
R_1	I_1	ε	#R	ε	#R	ε	#R
1	2	1.0218	4	1.0025	6	1.0025	6
2	3	0.9685	6	4.1451e-10	9	4.5808e-11	9
3	3	0.9685	6	2.7119e-10	9	≈ 0	9
	R_2^β	1		2		3	
	I_2^β	2		3		3	
R_1^β	I_1^β	ε	#R	ε	#R	ε	#R
1	2	0.7210	4	0.7072	6	0.7072	6
2	3	0.6833	6	2.1621e-12	9	≈ 0	9
3	3	0.6833	6	≈ 0	9	≈ 0	9

TABLE V: Complexity and accuracy trade-off with the original and the Relaxed TS Fuzzy model transformation (Example 2).

Assume the following qLPV model of the TORA system

$$\begin{bmatrix}\dot{\mathbf{x}}(t)\\ \mathbf{y}(t)\end{bmatrix} = \mathbf{S}(\mathbf{p}(t))\begin{bmatrix}\mathbf{x}(t)\\ \mathbf{u}(t)\end{bmatrix}, \tag{62}$$

where $\mathbf{x}(t)$, $\mathbf{u}(t)$ and $\mathbf{y}(t)$ are the state, input and output vectors respectively. Here $p_1(t) = x_3(t)$ and $p_2(t) = x_4(t)$. The system matrix $\mathbf{S}(\mathbf{p}(t))$ takes the form of

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0\\ -f_1(\mathbf{p}(t)) & 0 & 0 & f_2(\mathbf{p}(t)) & -f_3(\mathbf{p}(t))\\ 0 & 0 & 0 & 1 & 0\\ f_3(\mathbf{p}(t)) & 0 & 0 & f_4(\mathbf{p}(t)) & f_1(\mathbf{p}(t)) \end{bmatrix}, \tag{63}$$

$$f_1(\mathbf{p}(t)) = \frac{1}{f(p_1(t))}, \quad f_3(\mathbf{p}(t)) = \frac{\rho cos(p_1(t))}{f(p_1(t))}, \tag{64}$$

$$f_2(\mathbf{p}(t)) = \rho p_2(t) sin(p_1(t)) f_1(\mathbf{p}(t)), \tag{65}$$

$$f_4(\mathbf{p}(t)) = -\rho^2 p_2(t) sin(p_1(t)) cos(p_1(t)) f_1(\mathbf{p}(t)), \tag{66}$$

$$f(p_1(t)) = 1 - \rho^2 cos^2(p_1(t)) \quad \text{and} \quad \rho = 0.2. \tag{67}$$

Firs of all, let us define an equidistant 300×300 grid covering $\Omega = \begin{bmatrix}-a & a\end{bmatrix} \times \begin{bmatrix}-a & a\end{bmatrix}$, where $a = \pi/4$. Further, let us define $\mathbf{S}^\beta(\mathbf{p}(t)) = \begin{bmatrix}f_1(\mathbf{p}(t)) & f_2(\mathbf{p}(t)) & f_3(\mathbf{p}(t))\end{bmatrix}$. The HOSVD of $\mathcal{F}$, which is the discretised tensor of $\mathbf{S}(\mathbf{p}(t))$, leads to the following singular values:

Dimension assigned to $p_1(t)$*:* 615.33; 12.16; 8.15; 0.18; 0.07; 1×10^{-12}; ....

Dimension assigned to $p_2(1)$*:* 615.38; 12.16; 2×10^{-13}; ....

The HOSVD executed on $\mathcal{F}^\beta$ leads to the singular values:

Dimension assigned to $p_1(t)$*:* 315.15; 12.16; 5.38; 0.18; 1×10^{-12};....

Dimension assigned to $p_2(1)$*:* 315.19; 12.16; 1×10^{-13}; ....

Let us discard singular values until their sum does not exceed 8. In the case of $\mathcal{F}$, we arrive at $\mathcal{F} \approx_\varepsilon \mathcal{D} \boxtimes_{n=1}^{2} \mathbf{U}_n$, where $\mathcal{D} \in \mathbb{R}^{3\times 2\times 4\times 5}$, $\mathbf{U}_1 \in \mathbb{R}^{300\times 3}$, $\mathbf{U}_2 \in \mathbb{R}^{300\times 2}$ and $\varepsilon = 0.1837$. The execution of the convex transformation leads to $\mathcal{F} \approx \mathcal{S} \boxtimes_{n=1}^{2} \mathbf{W}_n$ where $\mathcal{S} \in \mathbb{R}^{4\times 2\times 4\times 5}$, $\mathbf{W}_1 \in \mathbb{R}^{300\times 4}$, $\mathbf{W}_2 \in \mathbb{R}^{300\times 2}$. Thus, the number of fuzzy rules is $4\times 2 = 8$.

In the case of $\mathcal{F}^\beta$, we arrive at: $\mathcal{F}^\beta \approx_{\varepsilon^\beta} \mathcal{D}^\beta \boxtimes_{n=1}^{2} \mathbf{U}_n^\beta$, where $\mathcal{D} \in \mathbb{R}^{2\times 2\times 4}$, $\mathbf{U}_1 \in \mathbb{R}^{300\times 2}$, $\mathbf{U}_2 \in \mathbb{R}^{300\times 2}$ and $\varepsilon^\beta = 0.1837$.

	R_2	2			R_2^β	2	
	I_2	2			I_2^β	2	
R_1	I_1	ε	#R	R_1^β	I_1^β	ε	#R
1	2	12.1622	4	1	2	12.1622	4
2	3	0.2212	6	2	3	0.1837	6
3	4	0.1837	8	3	4	0.1837	8
4	5	3.9139e-10	10	4	5	3.1248e-11	10
5	5	≈ 0	10	5	5	≈ 0	10

TABLE VI: Comparison of the original and the Relaxed TS Fuzzy model transformation (TORA benchmark example).

Applying the convex transformation and reinserting the separated constant elements, we have $\mathcal{F}^\beta \approx \mathcal{S}^\beta \boxtimes_{n=1}^{2} \mathbf{W}_n^\beta$, where $\mathcal{S}^\beta \in \mathbb{R}^{3\times 2\times 4\times 5}$, $\mathbf{U}_1 \in \mathbb{R}^{300\times 3}$, $\mathbf{U}_2 \in \mathbb{R}^{300\times 2}$. The resulting number of rules is $3\times 2 = 6$.

We can observe that the original TS Fuzzy model transformation results in 8 fuzzy rules with error 0.1837, while the Relaxed TS Fuzzy model transformation can achieve an error of 0.1837 with only 6 fuzzy rules. If we keep 2 singular values in the first dimension using the original TS Fuzzy model transformation, then the resulting TS Fuzzy model has 6 fuzzy rules, however the error increases to 0.2212. The different variations of this complexity-accuracy trade-off are given in Table VI.

Let us investigate the required computational complexity. The number of the elements of $\mathcal{F}$ is $300\times 300\times 4\times 5 = 1800000$, which leads to 1×10^9 operational steps, see Equ. (51), while the number of elements of $\mathcal{F}^\beta$ is $300\times 300\times 6 = 540000$, which leads to 324×10^6 operational steps, see Equ. (52). This is a reduction of 70%.

Thus, we can increase the resolution of the grid to 4484 $(1.5\times)$ in the case of the Relaxed TS Fuzzy model transformation with the same computational power as was used for the original TS Fuzzy model transformation, at least in the current example.

VIII. Example of a real-word engineering problem

In the literature related to TP model transformations, the real-world example of a very complex aeroelastic wing section often appears [4], [5], [31], [43], which is taken from a real engineering control problem. The data and parameters of the model were identified based on real-life measurements conducted by NASA. The above papers refer to further papers which detail the physical measurement system and the identification processes.

For comparability, we also use this example in this paper. The challenge when it comes to TS Fuzzy modelling and control design with respect to this problem lies in the strong non-linearities and complexity of the model. The state-space model of the 2D aeroelastic wing section has state vector $\mathbf{x}(t) \in \mathbb{R}^4$ as $\mathbf{x}(t) = \begin{bmatrix}x_1(t) & x_2(t) & x_3(t) & x_4(t)\end{bmatrix}^T = \begin{bmatrix}h(t) & \alpha(t) & \dot{h}(t) & \dot{\alpha}(t)\end{bmatrix}^T$, where $x_1(t)$ is the plunging displacement and $x_2(t)$ is the pitching displacement. The system matrix S(p(t)) of the state-space model depends on the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

parameter vector $\mathbf{p}(t) = \begin{bmatrix} U(t) & x_2(t) \end{bmatrix} \in \mathbb{R}^2$, $\mathbf{p} \in \Omega = \begin{bmatrix} 14 & 25 \end{bmatrix} \times \begin{bmatrix} -0.3 & 0.3 \end{bmatrix}$. Here, free stream velocity $U(t)$ is an external parameter. The entries of the system matrix are

$$\mathbf{S}(\mathbf{p}(t)) = \begin{bmatrix} \mathbf{0}_{2\times 2} & \mathbf{I}_{2\times 2} \quad \mathbf{0}_{2\times 1} \\ \mathbf{S}_1(\mathbf{p}(t)) & \mathbf{S}_2(\mathbf{p}(t)) \end{bmatrix}, \tag{68}$$

$$\mathbf{S}_1(\mathbf{p}(t)) = \begin{bmatrix} -k_1 & -k_2U^2(t) - p(k_\alpha(x_2(t))) \\ -k_3 & -k_4U^2(t) - q(k_\alpha(x_2(t))) \end{bmatrix}, \tag{69}$$

$$\mathbf{S}_2(\mathbf{p}(t)) = \begin{bmatrix} -c_1(U(t)) & -c_2(U(t)) & g_3U^2(t) \\ -c_3(U(t)) & -c_4(U(t)) & g_4U^2(t) \end{bmatrix}, \tag{70}$$

$$p(x_2(t)) = C_p k_\alpha(x_2(t)), \quad q(x_2(t)) = C_q k_\alpha(x_2(t)) \tag{71}$$

$$\begin{aligned} k_\alpha(x_2(t)) = 2.82(1 - 22.1x_2(t) + 1315.5x_2^2(t)+ \\ +8580x_2^3(t) + 17289.7x_2^4(t)), \end{aligned} \tag{72}$$

$$\begin{aligned} c_1(U(t)) &= (I_\alpha c_h + U(t)(I_\alpha \rho b c_{l_\alpha} + m x_\alpha \rho c_{m_\alpha}))/d \\ c_2(U(t)) &= (z\rho U(t)(I_\alpha b^2 c_{l_\alpha} + m x_\alpha b^4 c_{m_\alpha}) - m x_\alpha b c_\alpha)/d \\ c_3(U(t)) &= -m(x_\alpha b c_h + \rho U(t) b^2(x_\alpha c_{l_\alpha} + c_{m_\alpha}))/d \\ c_4(U(t)) &= m(c_\alpha - z\rho U(t) b^3(x_\alpha c_{l_\alpha} + c_{m_\alpha}))/d. \end{aligned} \tag{73}$$

Here $a = -0.673$, $b = 0.135$, $k_h = 2844.4$, $c_h = 27.43$, $c_\alpha = 0.036$, $\rho = 1.225$, $c_{l_\alpha} = 6.28$, $c_{l_\beta} = 3.358$, $c_{m_\alpha} = (0.5 + a) * c_{l_\alpha}$, $m = 12.387$, $c_{m_\beta} = -0.635$, $x_\alpha = -0.3533 - a$, $I_\alpha = 0.065$, $d = 0.5193$, $k_1 = 356$ $k_2 = 0.105$, $k_3 = -2928.1$. $k_4 = -0.4906$,$C_p = -1.0294$, $C_q = 23.851$, $g_3 = -0.054911$, $z = (1/2 - a)$ and $g_4 = 0.2335$.

Let us first execute the original TS Fuzzy model transformation. Let the grid density be 1000×1000. The resulting singular values are:

Dimension assigned to $p_1(t)$*:* 9.17×10^6, 6.26×10^4, 100.
Dimension assigned to $p_2(t)$*:* 8.82×10^6, 2.49×10^6.

The computational complexity has 5×10^9 operational steps, see Equ, (51).

As a next step, let us execute the Relaxed TS Fuzzy model transformation. After separating the constant elements of Equ. (68) we obtain a vector function that has 8 elements only. The Relaxed TS Fuzzy model results in the following singular values:

Dimension assigned to $p_1(t)$*:* 8.69×10^6, 5.64×10^4, 100.
Dimension assigned to $p_2(t)$*:* 8.69×10^6, 1×10^5.

The computational complexity has 2×10^9, see Equ. (52).

We can observe that the Relaxed TS Fuzzy model transformation results in smaller singular values, leading to a better trade-off and requires 60% less computational power.

IX. Conclusion

The primary goal of the TS Fuzzy model transformation is to convert a given model into various different alternative TS Fuzzy model representations with advantageous properties that can enhance subsequent design outcomes. A key feature of this transformation is its ability to identify the absolute minimum number of fuzzy rules and balance the trade-off between the number of the fuzzy rules and approximation accuracy, in case further reduction is necessary. At the same time, a notable limitation is the intensive computational resources required, particularly for high-resolution execution with a larger number of inputs. In this context, a new variant of the TS Fuzzy model transformation was introduced in this paper, which is referred to as the Relaxed TS Fuzzy model transformation. This variant aims to provide an improved balance between number of fuzzy rules and approximation accuracy, while significantly reducing computational complexity. These benefits are amplified with an increase in the number and values of constant elements within the function. To demonstrate the effectiveness of the proposed approach and to facilitate a thorough comparison with the original TS Fuzzy model transformation, the paper includes four numerical examples. The first two examples are well-known benchmark in the literature concerning the development of TS Fuzzy model transformations. The third and the fourth examples involve real engineering models frequently employed as benchmarks in related studies. As a conclusion, a practical design guideline can be advocated for through the use of the Relaxed TS Fuzzy model transformation over the original version in all scenarios, irrespective of the necessity for further reduction in the fuzzy rule base beyond the minimum requirement. This recommendation is underpinned by a significant reduction in computational complexity offered by the Relaxed TS Fuzzy model transformation. Future work on the further development of the Relaxed TS Fuzzy model transformation could focus on Interval Type-2 TS Fuzzy models and their convex hull manipulation possibilities, enhancing subsequent control design and improving the resulting control performance.

References

[1] P. Baranyi, Y. Yam, and P. Várlaki, *Tensor Product Model Transformation in Polytpic Model Based Control*, ser. Automation and Control Engineering. CRC Press, Taylor & Frances Group, March 2017, ISBN 9781138077782 - CAT K34341.

[2] P. Baranyi, *TP-Model Transformation Based Control Design Frameworks*, ser. Control Engineering. Springer book, July 2016, ISBN 978-3-319-19605-3.

[3] ——, *Dual-Control-Design TP and TS Fuzzy Model Transformation Based Control Optimisation and Design*, ser. Topics in Intelligent Engineering and Informatics (TIEI, volume 17). Springer book, December 2023.

[4] ——, "The generalized TP model transformation for T–S fuzzy model manipulation and generalized stability verification," *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 4, pp. 934–948, 2014.

[5] ——, "How to vary the input space of a T–S fuzzy model: A TP model transformation-based approach," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 2, pp. 345–356, 2022.

[6] ——, "Extracting LPV and qLPV structures from state-space functions: A TP model transformation based framework," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 3, pp. 499–509, 2020.

[7] ——, "Transition between TS Fuzzy models and the associated convex hulls by TS Fuzzy model transformation," *IEEE Transactions on Fuzzy Systems*, vol. 32, no. 4, pp. 2272–2282, 2024.

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

[9] 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, April 1999.

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

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

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in IEEE Transactions on Fuzzy Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2024.3418509

[12] 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.

[13] K. Tanaka and H. Wang, *Fuzzy control systems design and analysis: a linear matrix inequality approach*. John Wiley and Sons, Inc., U.S.A., 2001.

[14] F. Chang, J. P. Kang, S. R. N. Huang, and G. L. Zhao, "Quadrotors' double-loop controller design with tensor product model transformation and partial fully actuated method," *ISA Transactions*, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S001905782400226X

[15] X. Pan, S. Han, and S. Lee, "Novel mismatch parameter-dependent stabilization approach based on sampled-data fuzzy lyapunov function," *IEEE Transactions on Fuzzy Systems*, vol. 32, no. 4, pp. 1668–1680, 2024.

[16] Y. Wang, C. Hua, and P. Park, "A generalized reciprocally convex inequality on stability and stabilization for t–s fuzzy systems with time-varying delay," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 3, pp. 722–733, 2023.

[17] X. Xie, F. Yang, L. Wan, J. Xia, and K. Shi, "Enhanced local stabilization of constrained n-TS fuzzy systems with lighter computational burden," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 3, pp. 1064–1070, 2023.

[18] Y. Wang, S. Xu, and C. K. Ahn, "Finite-time composite antidisturbance control for t–s fuzzy nonhomogeneous markovian jump systems via asynchronous disturbance observer," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 11, pp. 5051–5057, 2022.

[19] W. Chen, Z. Fei, X. Zhao, and M. V. Basin, "Generic stability criteria for switched nonlinear systems with switching-signal-based lyapunov functions using takagi–sugeno fuzzy model," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 10, pp. 4239–4248, 2022.

[20] J. Song and X.-H. Chang, "Induced $\mathcal{L}_\infty$ quantized sampled-data control for t–s fuzzy system with bandwidth constraint," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 3, pp. 1031–1040, 2023.

[21] A. Ait Ladel, A. Benzaouia, R. Outbib, and M. Ouladsine, "Integrated state/fault estimation and fault-tolerant control design for switched t–s fuzzy systems with sensor and actuator faults," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 8, pp. 3211–3223, 2022.

[22] Y. Kang, L. Yao, and H. Wang, "Fault isolation and fault-tolerant control for takagi–sugeno fuzzy time-varying delay stochastic distribution systems," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 4, pp. 1185–1195, 2022.

[23] Y. Qiu, C. Hua, and Y. Wang, "Nonfragile sampled-data control of t–s fuzzy systems with time delay," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 8, pp. 3202–3210, 2022.

[24] X. Xie, C. Wei, Z. Gu, and K. Shi, "Relaxed resilient fuzzy stabilization of discrete-time takagi–sugeno systems via a higher order time-variant balanced matrix method," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 11, pp. 5044–5050, 2022.

[25] H.-Y. Sun, H.-G. Han, J. Sun, H.-Y. Yang, and J.-F. Qiao, "Security control of sampled-data t–s fuzzy systems subject to cyberattacks and successive packet losses," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 4, pp. 1178–1188, 2023.

[26] Y. Ren, D.-W. Ding, Q. Li, and X. Xie, "Static output feedback control for t–s fuzzy systems via a successive convex optimization algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 10, pp. 4298–4309, 2022.

[27] M. S. Aslam, P. Tiwari, H. M. Pandey, and S. S. Band, "Observer–based control for a new stochastic maximum power point tracking for photovoltaic systems with networked control system," *IEEE Transactions on Fuzzy Systems*, pp. 1–14, 2022.

[28] F. Sabbghian-Bidgoli and M. Farrokhi, "Polynomial fuzzy observer-based integrated fault estimation and fault-tolerant control with uncertainty and disturbance," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 3, pp. 741–754, 2022.

[29] S. Tiko and F. Mesquine, "Constrained control for a class of TS fuzzy systems," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 1, pp. 348–353, 2023.

[30] F. Ma, J. Li, and L. Wu, "Tensor product model HOSVD based polytopic LPV controller for suspension anti-vibration system," *Journal of Vibration and Control*, vol. 29, no. 1-2, pp. 5–20, 2022.

[31] P. Baranyi, Z. Petres, V. P.L., K. P., and Y. Yam, "Determination of different polytopic models of the prototypical aeroelastic wing section by TP model transformation," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 10, no. 4, pp. 486–493, 2006.

[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] 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.

[34] 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.

[35] P. Baranyi, Y. Yam, D. Tikk, and R. Patton, "Trade-off between approximation accuracy and complexity: TS controller design via HOSVD based complexity minimization," *Studies in Fuzziness and Soft Computing, Vol. 128. Interpretability Issues in Fuzzy Modeling, J. Casillas, O. Cordón, F. Herrera, L. Magdalena (Eds.), Springer-Verlag*, pp. 249–277, 2003.

[36] P. Baranyi and Y. Yam, "Fuzzy rule base reduction," in *Fuzzy IF-THEN Rules in Computational Intelligence: Theory and Applications*, D. Ruan and E. E. Kerre, Eds. Kluwer, 2000, ch. 7, pp. 135–160.

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

[38] K. Lei, P. Baranyi, and Y. Yam, "Complexity minimalisation of non-singleton based fuzzy-neural network," *International Journal of Advanced Computational Intelligence*, vol. 4, no. 4, pp. 1–7, 2000.

[39] D. Tikk, P. Baranyi, and R. Patton, "Approximation properties of TP model forms and its consequences to TPDC design framework," *Asian Journal of Control*, vol. 9, no. 3, pp. 221–231, 2008.

[40] D. Tikk, P. Baranyi, R. Patton, and J. Tar, "Approximation capability of TP model forms," *Australian Journal of Intelligent Information Processing Systems*, vol. 8, pp. 155–163, 2004.

[41] P. Baranyi, K. Lei, and Y. Yam, "Complexity reduction of singleton based neuro-fuzzy algorithm," in *IEEE international conference on Systems, Man and Cybernetics*, 2000, pp. 2503–2508.

[42] N. Vannieuwenhoven, R. Vandebril, and K. Meerbergen, "A new truncation strategy for the higher-order singular value decomposition," *SIAM Journal on Scientific Computing*, vol. 34, no. 2, pp. A1027–A1052, 2012.

[43] A. Szollosi and P. Baranyi, "Influence of the tensor product model representation of qLPV models on the feasibility of linear matrix inequality," *Asian Journal of Control*, vol. 18, no. 4, pp. 1328–1342, 2015.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/