SVD-Based Reduction to MISO TS Models

Péter Baranyi, Yeung Yam, Annamária R. Várkonyi-Kóczy, *Senior Member, IEEE*, and Ron J. Patton, *Member, IEEE*

***Abstract*—The main objective of this paper is to expound the singular-value-decomposition (SVD)-based reduction technique proposed to single-input–single-output *Takagi-Sugeno* (TS) fuzzy models to multivariable cases. The use of *higher order singular value decomposition* is proposed in this paper for the complexity reduction of multiple-input–single-output TS fuzzy model approximation. A detailed illustrative example of a nonlinear dynamic model is also discussed.**

***Index Terms*—Complexity reduction, higher order singular value decomposition (SVD), SVD-based fuzzy rule base reduction.**

I. Introduction

The *Takagi-Sugeno* (TS) fuzzy model is one way to describe a nonlinear dynamic system using local linear models [1]–[10]. Its objective is that the system dynamics can be captured by a set of fuzzy implications, which characterize local regions in the state space. The main feature of a TS fuzzy model is to express the local dynamics of each fuzzy rule by a linear system model. The overall fuzzy model is achieved by fuzzy "blending" of the linear models. Recently, the issue of stability of fuzzy control systems has been investigated extensively in nonlinear stability frameworks [2]–[9], which helps us with designing TS models, controllers, and observers. The TS multiple-model scheme is used both for the feedback control [2]–[4] and the design of observers for fault diagnosis and isolation (FDI) [5]–[7]. Both are based on the principle of *parallel distributed compensation* (PDC) [8], [9]. By duality, the FDI structure is also based upon the TS fuzzy model system. Despite the above advantages, the use of TS fuzzy models is strictly limited by their exponentially growing complexity in respect to their approximation accuracy [10]. This complexity problem comes from two inevitable facts. Namely, the main structure of the TS model is, actually, a fuzzy rule base. Therefore, the TS model inherits the exponential complexity problem of fuzzy rule bases [11]–[14]. This leads to a requirement of finding a balance between two conflicting aims in achieving a good system performance by the TS model, such as reducing the fitting error of TS fuzzy models and reducing the model complexity.

In other words, the complexity problem of TS fuzzy models is due to the fact that fuzzy logic applications are suffering from exponentially growing computational complexity in respect to their approximation accuracy. This difficulty comes from two inevitable facts. The first is that the most adopted fuzzy inference techniques do not hold the universal approximation property, if the number of antecedent sets is limited, shown by Tikk in [15]. Furthermore, their explicit functions are sparse in the approximation function space. This fact inspires us to increase the density, namely, the number of fuzzy terms in pursuit of a good approximation. This, however, may soon lead to a conflict with the computational capacity available for the implementation, since the increasing number of antecedents exponentially explodes the computational cost. The computational explosion is the second fact as stated by Kóczy *et al.* in [14]. The effect of this contradiction is gained by the lack of a mathematical framework capable of estimating the necessary minimal number of antecedent sets. Therefore, a heuristic setting of antecedent sets is applied, which usually overestimates, in order to be on the safe side, the necessary number of antecedents resulting in an unnecessarily high computational cost. As a result, fuzzy rule base complexity reduction techniques emerged as a new topic in fuzzy theory. Some reduction techniques are classified regarding their concept in [11] and [16]. A fuzzy rule importance based technique is proposed by Song *et al.* in [32]. Another recent method is proposed by Sudkamp *et al.* [34], which combines rule learning with a region-merging strategy.

Recently, several publications have applied orthogonal transformation methods for selecting important rules from a given rule base. For instance, in 1999 Yen and Wang investigated various techniques in [11] for possible fuzzy rule base simplification techniques such as orthogonal least-squares, eigenvalue decomposition, singular value decomposition (SVD-QR) with column pivoting method, total least-squares method, and direct SVD method. [33] also proposes an SVD based technique with examples. The SVD-based fuzzy approximation technique was proposed in 1997 [13], which directly finds a minimal rule-base from sampled values. Shortly after, this concept was introduced as SVD fuzzy rule base reduction and structure decomposition in [12], [39], and [40]. Its key idea is conducting SVD of the consequents and generating proper linear combinations of the original membership functions to form new ones for the reduced set. [12], [13] characterizes membership functions by the conditions of sum-normalization (SN), nonnegativeness (NN), and

Manuscript received June 19, 2001; revised May 5, 2002. Abstract published on the Internet November 20, 2002. This work was supported by the Hungarian National Science Research Found (OTKA) under Grants F030056, T034233 and T034212, MKM KFKP 180/2001 and by the British-Hungarian Intergovernmental Scientific Co-operation Fund under Grant GB 37/98. The work of P. Baranyi was supported by the Zoltán Magyary Foundation.

P. Baranyi is with the Department of Telecommunication and Telematics, Integrated Intelligent Systems Japanese-Hungarian Laboratory, Budapest University of Technology and Economics H-1111 Budapest, Hungary (e-mail: baranyi@ttt-202.ttt.bme.hu).

Y. Yam is with the with the Department of Automation and Computer Aided Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: yyam@acae.cuhk.edu.hk).

A. R. Várkonyi-Kóczy is with the Department of Measurement and Information Systems, Integrated Intelligent Systems Japanese-Hungarian Laboratory, Budapest University of Technology and Economics, H-1111 Budapest, Hungary (e-mail: koczy@mit.bme.hu).

R. J. Patton is with the Control and Intelligent Systems Engineering Research Group, School of Engineering, University of Hull, Hull HU6 7RX, U.K. (e-mail: r.j.patton@eng.hull.ac.uk).

Digital Object Identifier 10.1109/TIE.2002.807673

normality (NO) and extends SVD reduction with further tools to preserve SN, NN, and NO conditions of the new membership functions. It may have a significant role if the purpose is not merely saving computational cost, but maintaining the fuzzy concept and proceeding further with a theoretical study on the reduced rule's features. Presumably, the SVD technique in this paper and in [12] and [13] can be replaced by other orthogonal techniques investigated by Yen and Wang in [11]. An extension of [11] to multidimensional cases may also be conducted in a similar fashion as the higher order SVD reduction technique proposed in [10], [12], and [13] and in this paper. Further developments of SVD-based fuzzy reduction [12], [13] are proposed in [10], [17], [18], and [38]. Examples of applying SVD reduction can be found in [35]–[37]. References [17] and [36] apply an Automatic Guided Vehicle system developed by Kovács [41]. The initial work in [12] and [13] can be applied regardless of the inference paradigm adopted for a fuzzy rule base as shown in [12] and [38]. Presumably, the product operation in this paper can be replaced by Rudas's generalized inference operators [20]–[23]. This would have a prominent role in developing the ability of finely tuning the TS models according to the application at hand and/or specific purposes of system performance.

SVD is not merely used as a way of reduction of fuzzy rule bases. A brief enumeration of some opportunities offered by SVD, development of which was started by Beltarmi about 200 years ago [19] and becomes one of the most fruitful tools in linear algebra, gives ideas about its promising role in complexity reduction in general. The key idea of using SVD in complexity reduction is that the singular values can be applied to decompose a given system and indicate the degree of significance of the decomposed parts. Reduction is conceptually obtained by the truncation of those parts, which have weak or no contribution at all to the output, according to the assigned singular values. This advantageous feature of SVD is used in this paper to minimize a given TS fuzzy model by discarding those local linear models, which have no significant role in the overall system. The complexity and its reduction is discussed in regard of the number of rules. However, reducing the number of rules does not imply the computational cost reduction in any cases, since the computation also depends on the number of simultaneously fired rules [42], [43]. Therefore, detailed investigation is given in the aspect of computational time reduction in this paper.

The present work constitutes a detailed investigation of the preliminary approaches outlined in [10] and gives a possible solution to the problem analyzed above. The algorithms proposed here are mostly developed in [12] and [13], but are restructured in terms of tensor description in order to facilitate further developments for TS fuzzy models. Concepts of higher order SVD (HOSVD) are investigated in tensor forms in the work of [24]–[28].

This paper is organized as follows. Section II defines various concepts to be utilized in the proposed method. Section III briefly summarizes the main concept of TS fuzzy models. Section IV examines the exponential complexity problem of TS fuzzy models in full accordance with [12]–[14]. Section V briefly summarizes those properties of HOSVD, which are significant in complexity reduction. Section VI presents the HOSVD based reduction of multiple-input–single-output (MISO) TS fuzzy models expounding the approaches defined in the preliminary work [10]. Section VII gives a detailed example of a dynamic system through numerical and analytical considerations to show the effectiveness of the proposed method.

II. Basic Definitions

This section will introduce some elementary definitions and concepts utilized in the further developments. Before starting with the definitions, some comments are enumerated on the notation to be utilized. To facilitate the distinction between the types of given quantities, they will be reflected by their representation: scalar values are denoted by lowercase letters $\{a, b, \ldots\}$; column vectors and matrices are given by boldface letters as $\{\mathbf{a}, \mathbf{b}, \ldots\}$ and $\{\mathbf{A}, \mathbf{B}, \ldots\}$, respectively. Tensors correspond to capital letters as $\{A, B, \ldots\}$, tensor 1 contains values 1 only. The transpose of matrix $\mathbf{A}$ is denoted as $\mathbf{A}^T$. A subscript is consistently used for a lower order of a given structure, e.g., an element of matrix $\mathbf{A}$ is defined by row–column number i, j symbolized as $(\mathbf{A})_{i,j} = a_{i,j}$. Systematically, the jth column vector of $\mathbf{A}$ is denoted as $\mathbf{a}_j$, i.e., $\mathbf{A} = [\,\mathbf{a}_1 \quad \mathbf{a}_2 \quad \cdots]$. Fuzzy sets are denoted by capital letters as $A : \mu_A(x)$. To enhance the overall readability characters $i, j, \ldots$ are in the meaning of indexes (counters), $I, J, \ldots$ are reserved to denote the index upper bounds, unless stated otherwise. $\Re^{I_1 \times I_2 \times \cdots \times I_N}$ is the vector space of real-valued $(I_1 \times I_2 \times \cdots \times I_N)$-tenors. Letter N serves to denote the number of dimensions of the space where the coefficient matrices of TS fuzzy model are approximated. M denotes local linear models.

Definition (n -mode Matrix of Tensor A): Assume an Nth order tensor $A \in \Re^{I_1 \times I_2 \times \cdots \times I_N}$. The n-mode matrix $\mathbf{A}_{(n)} \in \Re^{I_n \times J}$, $J = \prod_k I_l$ contains all the vectors in the nth dimension of tensor A . The ordering of the vectors is arbitrary, this ordering shall, however, be consistently used later on. $(\mathbf{A}_{(n)})_j$ is called an jth n -mode vector.

Note that any matrix of which the columns are given by n-mode vectors $(\mathbf{A}_{(n)})_j$ can evidently be restored to be tensor A. The restoring can be executed even in the case when some rows of $\mathbf{A}_{(n)}$ are discarded since the value of I_n has no role in the ordering of $(\mathbf{A}_{(n)})_j$ [24].

Definition (n-mode Subtensor of Tensor A): Assume an N th-order tensor $A \in \Re^{I_1 \times I_2 \times \cdots \times I_N}$. The n-mode sub-tensor $A_{i_n=\alpha}$ contains elements $a_{i_1, i_2, \ldots, i_{n-1}, \alpha, i_{n+1}, \ldots, i_N}$.

Definition (n-mode Tensor Partition): Assume an Nth order tensor $A \in \Re^{I_1 \times I_2 \times \cdots \times I_N}$. n -mode partitions of tensor A are $B_l \in \Re^{I_1 \times I_2 \cdots \times I_{n-1} \times J_l \times I_{n+1} \times \cdots I_N}$ denoted as $A = [\,B_1 \quad B_2 \quad \cdots \quad B_L\,]_n$, where $I_n = \sum_l J_l, l = 1 \cdots L$.

Definition (Scalar Product): The scalar product $\langle A, B\rangle$ of two tensors $A, B \in \Re^{I_1 \times I_2 \times \cdots \times I_N}$ is defined as $\langle A, B\rangle \stackrel{\text{def}}{=} \sum_{i_1} \sum_{i_2} \ldots \sum_{i_N} b_{i_1 i_2 \ldots i_N} a_{i_1 i_2 \ldots i_N}$.

Definition (Orthogonality): Tensors of which the scalar product equals 0, are mutually orthogonal.

Definition (Frobenius-norm): The Frobenius-norm of a tensor A is given by $\|A\| \stackrel{\text{def}}{=} \sqrt{\langle A, A\rangle}$.

Definition (n-mode Matrix–Tensor Product): The n-mode product of tensor $A \in \Re^{I_1 \times I_2 \times \cdots \times I_N}$ by a matrix $\mathbf{U} \in \Re^{J \times I_n}$, denoted by $A \times_n \mathbf{U}$ is an (

$I_1 \times I_2 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N$)-tensor of which the entries are given by $A \times_n \mathbf{U} = B$, where $B_{(n)} = \mathbf{U} \cdot A_{(n)}$. Let $A \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \cdots \times_N \mathbf{U}_N$ be noted for brevity as $A \overset{\otimes}{_n} \mathbf{U}_n$, $n = 1 \cdots N$.

Theorem (Matrix SVD): Every real-valued ($I_1 \times I_2$)-matrix $\mathbf{F}$ can be written as the product of $\mathbf{F} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T = \mathbf{S} \times_1 \mathbf{U} \times_2 \mathbf{V}$, in which:

1) $\mathbf{U} = [\mathbf{u}_1 \quad \mathbf{u}_2 \quad \cdots \quad \mathbf{u}_{I_1}]$ *is a unitary ($I_1 \times I_1$)-matrix*;
2) $\mathbf{V} = [\mathbf{v}_1 \quad \mathbf{v}_2 \quad \cdots \quad \mathbf{v}_{I_2}]$ *is a unitary ($I_2 \times I_2$)-matrix*;
3) $\mathbf{S}$ *is an ($I_1 \times I_2$)-matrix with the properties of:*
   a) *pseudodiagonality:* $\mathbf{S} = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_{\min(I_1, I_2)})$;
   b) *ordering:* $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min(I_1, I_2)} \geq 0$.

The σ_i are the singular values of $\mathbf{F}$ and the vectors $\mathbf{U}_i$ and $\mathbf{V}_i$ are, respectively, an ith left and an ith right singular vectors.

There are major differences between matrices and HO tensors when rank properties are concerned. These differences directly affect the way an SVD generalization could look like. As a matter of fact, there is no unique way to generalize the rank concept. In this paper, we restrict the description to n-mode rank only.

Definition (n-mode Rank of Tensor): The n-mode rank of A , denoted by $R_n = rank_n(A)$, is the dimension of the vector space spanned by the n-mode vectors as $rank_n(A) = rank(\mathbf{A}_{(n)})$.

Theorem [N-th Order SVD (HOSVD)]: Every tensor $A \in \Re^{I_1 \times I_2 \times \cdots \times I_N}$ can be written as the product $A = S \overset{\otimes}{_{n=1}} \mathbf{U}_n$, in which:

1) $\mathbf{U}_n = [\mathbf{u}_{1,n} \quad \mathbf{u}_{2,n} \quad \ldots \quad \mathbf{u}_{I_N,n}]$ *is a unitary ($I_N \times I_N$)-matrix called n-mode singular matrix*;
2) *tensor $S \in \Re^{I_1 \times I_2 \times \cdots \times I_N}$ of which the subtensors $S_{i_n=\alpha}$ have the properties of:*
   a) *all-orthogonality: two subtensors $S_{i_n=\alpha}$ and $S_{i_n=\beta}$ are orthogonal for all possible values of n, α and $\beta : \langle S_{i_n=\alpha}, S_{i_n=\beta} \rangle = 0$ when $\alpha \neq \beta$;*
   b) *ordering: $\|S_{i_n=1}\| \geq \|S_{i_n=2}\| \geq \cdots \geq \|S_{i_n=I_n}\| \geq 0$ for all possible values of n.*

The Frobenius-norm $\|S_{i_n=1}\|$, symbolized by $\sigma_i^{(n)}$, are n-mode singular values of A and the vector $\mathbf{u}_{i,n}$ is an i-th singular vector. S is called the core tensor.

More detailed discussion of matrix SVD and HOSVD is given in [24], [28].

III. TS Model Approximation

This section is intended to discuss the fundamental form of TS fuzzy models. For further detailed investigation of TS fuzzy models and closely related concepts see [1]–[10] and [30]. A TS model consists of a number of local linear models assigned to fuzzy regions, which are designed to approximate the dynamic features at the corresponding operating fuzzy points in vector space P. Fig. 1 shows the structure of a TS fuzzy model. The model varies according to vector $\mathbf{p} \in \Re^N$, which may contain some values of the state vector $\mathbf{x}$ as well. The TS fuzzy inference engine is responsible for combining the local linear models according to vector $\mathbf{p}$ in order to find a proper model, which is assumed to be the momentary linear descriptor of the system capable of generating output vector $\mathbf{y}$ from state vector $\mathbf{x}$ and input vector $\mathbf{u}$.

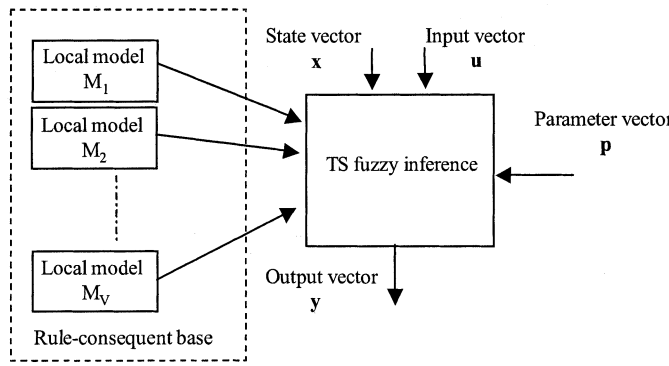

Fig. 1. Structure of a TS fuzzy model.

In the following, the adopted forms of the TS fuzzy model are discussed based on the forms of [6], [7], and [30].

Definition ($M(\mathbf{p})$ is a Model in Respect of Vector $\mathbf{p} \in \Re^N$): Assume a given model varies in the N-dimensional parameter space P

$$\left.\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}(\mathbf{p}) \cdot \mathbf{x}(t) + \mathbf{B}(\mathbf{p}) \cdot \mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}(\mathbf{p}) \cdot \mathbf{x}(t) + \mathbf{D}(\mathbf{p}) \cdot \mathbf{u}(t) \end{aligned}\right\} \Rightarrow M(\mathbf{p}). \quad (1)$$

In many cases, the rows of (1) consist of more than two terms like, for instance, in the case of observer design in [7] and [30], where the first row of model (1) has an extra term $\mathbf{L}(\mathbf{y}(t) - \hat{\mathbf{y}}(t))$. Therefore, to facilitate the further notation and have general description like in [10], the form of (2), shown at the bottom of the page, is applied, where K denotes the number of rows in the model (1) (i.e., the number of equations describing the model) and L indicates how many terms are in the rows of the equations, for instance, these are 2 in (1). Vector $\mathbf{x}_l \in \Re^{I_l}$ consists of the model input or state vectors, where I_l denotes the number of "input" elements in $\mathbf{x}_l$. Vector $\mathbf{z}_k \in \Re^{O_k}$ contains the output values of the kth row in (2), where O_k denotes the number of "output" values in $\mathbf{z}_k$. This implies that the size of $\mathbf{B}_{k,l}(\mathbf{p})$ is $O_k \times I_l$. For example, describing (1) by (2) the result is: $\mathbf{x}_1(t) = \mathbf{x}(t)$, $\mathbf{x}_2(t) = \mathbf{u}(t)$ and the outputs of the model are $\mathbf{z}_1(t) = \dot{\mathbf{x}}(t)$ and $\mathbf{z}_2(t) = \mathbf{y}(t)$. Coefficient matrices become: $\mathbf{B}_{1,1}(\mathbf{p}) = \mathbf{A}(\mathbf{p})$, $\mathbf{B}_{1,2}(\mathbf{p}) = \mathbf{B}(\mathbf{p})$, $\mathbf{B}_{2,1}(\mathbf{p}) = \mathbf{C}(\mathbf{p})$,

$$M(\mathbf{p}) \Rightarrow \begin{vmatrix} \mathbf{z}_1(t) \\ \mathbf{z}_2(t) \\ \vdots \\ \mathbf{z}_K(t) \end{vmatrix} = \begin{vmatrix} \mathbf{B}_{1,1}(\mathbf{p}) & \mathbf{B}_{1,2}(\mathbf{p}) & \cdots & \mathbf{B}_{1,L}(\mathbf{p}) \\ \mathbf{B}_{2,1}(\mathbf{p}) & \mathbf{B}_{2,2}(\mathbf{p}) & & \mathbf{B}_{1,L}(\mathbf{p}) \\ \vdots & & \ddots & \\ \mathbf{B}_{K,1}(\mathbf{p}) & \mathbf{B}_{K,2}(\mathbf{p}) & \cdots & \mathbf{B}_{K,L}(\mathbf{p}) \end{vmatrix} \begin{vmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \\ \vdots \\ \mathbf{x}_L(t) \end{vmatrix} \quad (2)$$

$\mathbf{B}_{2,2}(\mathbf{p}) = \mathbf{D}(\mathbf{p})$. For abbreviation, let us use the following notation of (2):

$$M(\mathbf{p}) \Rightarrow \{\mathbf{B}_{k,l}(\mathbf{p})\}.$$

In this paper, the firing probability of the fuzzy rules is based on a product operator like in [1]–[10] and [30]. Here, two formulas of a TS fuzzy model are discussed accordingly to the types of rule bases. The first one defines uncompleted rule base while the second takes all possible rules combined by antecedents into account [1]–[10], [30]. Before starting with the definition, let us initiate the following notation for the vth local linear model:

$$M_v \Rightarrow \{\mathbf{B}_{v,k,l}\}.$$

Definition (Uncompleted TS Fuzzy Model): Assume n-variable model consequent-based fuzzy rules as:

IF p_1 is $A_{1,v}$ AND p_2 is $A_{2,v}\ldots$ AND p_N is $A_{N,v}$ THEN model M_v, $v = 1\cdots V$. where $A_{n,v} : \mu_{A_{n,v}}(p_n)$ is the v -th antecedent fuzzy set on the n-th input universe. The approximated model in respect of p is

$$\hat{M}(\mathbf{p}) \Rightarrow \left\{\hat{\mathbf{B}}_{k,l}(\mathbf{p}) = \frac{\sum\limits_v \prod_n \mu_{A_{n,v}}(p_n)\cdot \mathbf{B}_{v,k,l}}{\sum\limits_v \prod_n \mu_{A_{n,v}}(p_n)}\right\}$$

which is written frequently as

$$\hat{M}(\mathbf{p}) \Rightarrow \left\{\mathbf{B}_{k,l}(\mathbf{p}) = \frac{\sum\limits_{v=1}^{V} f_v(\mathbf{p})\hat{\mathbf{B}}_{v,k,l}}{\sum\limits_{v=1}^{V} f_v(\mathbf{p})}\right\} \tag{3}$$

where the firing probability of the rules is $f_v(\mathbf{p}) = \prod_n \mu_{A_{n,v}}(p_n)$. Usually, the antecedent sets are given in Ruspini-partition, namely, $\forall_n\ \mathbf{p} : \sum_n \mu A_{n,v}(p_n) = 1$ and are normalized as

$$\forall \mathbf{p} : \sum_v f_v(\mathbf{p}) = 1 \tag{4}$$

in (3) *, thus, the denominator can be taken out of consideration.*

In the following, a TS fuzzy model is defined where all combinations of the antecedent sets yields a fuzzy rule, namely, where the rule base is completed.

Definition (Completed TS Fuzzy Model): The completed fuzzy rules are formed by all combinations of the antecedents as follows:

IF p_1 is A_{1,v_1} AND p_2 is $A_{2,v_2}\ldots$ AND p_N is A_{3,v_3} THEN model $M_{v_1,v_2,\ldots,v_N}$, $v_n = 1\cdots V_n$ where $M_{v_1,v_2,\ldots,v_N} \Rightarrow \{\mathbf{B}_{v_1,v_2,\ldots,v_N,k,l}\}$. Consequently, the approximation becomes as (5), shown at the bottom of the page, where V_n is the number of antecedent sets on the n-th antecedent universe. If the antecedent sets are defined in Ruspini-partition as above, then

$$\sum_{v_1=1}^{V_1}\sum_{v_2=1}^{V_2}\cdots\sum_{v_N}^{V_N}\prod_n \mu_{A_{n,v_n}}(p_n) = 1. \tag{6}$$

In order to facilitate the further development, let (3) and (5) be given in terms of tensor with assumption of (4) and (6). Thus, (3) can be formulated as [10]

$$\hat{M}(\mathbf{p}) \Rightarrow \left\{\hat{\mathbf{B}}_{k,l}(\mathbf{p}) = (B_{k,l}\times_1 \mathbf{f}(\mathbf{p}))_{(2)}\right\} \tag{7}$$

where $\mathbf{f}(\mathbf{p}) \in \Re^V$ is the vector of functions $f_v(\mathbf{p})$. Tensor $B_{k,l} \in \Re^{V\times O_k\times I_l}$ consists of matrices $\mathbf{B}_{v,k,l} \in \Re^{O_k\times I_l}$. The first dimension of $B_{k,l}$ is assigned to functions $f_v(\mathbf{p})$. The next two are assigned to the output and input vectors, respectively.

In the same way, (5) is reformulated as

$$\hat{M}(\mathbf{p}) \Rightarrow \left\{\hat{\mathbf{B}}_{k,l}(\mathbf{p}) = \left(B_{k,l}\times_1 \bigotimes_n \mathbf{m}_n(p_n))\right)_{(N+1)}\right\} \tag{8}$$

where $\mathbf{m}_n(p_n) \in \Re^{V_n}$ contains membership values $\mu A_{n,v_n}(p_n)$ and the $N+2$ dimensional tensor $B_{k,l} \in \Re^{V_1\times V_2\times\cdots\times V_N\times O_k\times I_l}$ is constructed from matrices $\mathbf{B}_{v_1,v_2,\ldots,v_{N.k.l}} \in \Re^{O_k\times I_l}$ of (5). The first N dimensions of $B_{k,l}$ are assigned to the dimensions of the parameter space P. The next two ones are assigned to the output and input vectors, respectively.

IV. Complexity Investigation

This section is intended to show the main motivation of the complexity reduction approach to be discussed here. The complexity of TS fuzzy models is proportional to the number of elements of tensor $B_{k,l}$, which will be detailed later. The computational reduction is hence based on the reduction of the size of tensor $B_{k,l}$. The first N dimensions of $B_{k,l}$ are assigned to the dimensions of the parameter space P, mentioned above and the next two are assigned to the output and input vectors, respectively. In the case of generalized form (7), the dimensionality of tensor $B_{k,l}$ is three regardless of the dimensionality of the parameter space. Therefore, its reduction can readily be traced back to the SISO model investigated in the preliminary work [10] and it is also included in the reduction of the completed MISO TS fuzzy model as a special case, namely, it can be considered in the same way as a completed SISO TS fuzzy model. Consequently, the discussion focuses on the completed MISO

$$\hat{M}(\mathbf{p}) \Rightarrow \left\{\hat{\mathbf{B}}_{k,l}(\mathbf{p}) = \frac{\sum\limits_{v_1=1}^{V_1}\sum\limits_{v_2=1}^{V_2}\cdots\sum\limits_{v_N}^{V_N}\prod_n \mu_{A_{n,v_n}}(p_n)\cdot \mathbf{B}_{v_1,v_2,\ldots,v_{N.k.l}}}{\sum\limits_{v_1=1}^{V_1}\sum\limits_{v_2=1}^{V_2}\cdots\sum\limits_{v_N}^{V_N}\prod_n \mu_{A_{n,v_n}}(p_n)}\right\} \tag{5}$$

TS fuzzy model from now on. First, the computational complexity of the model is examined. The output values are calculated by (8) as

$$\mathbf{z}_k = \left(\sum_l \left[B_{k,l} \underset{n}{\otimes} \mathbf{m}_n (p_n) \right] \times_{N+2} \mathbf{x}_l^T \right)_{(N+1)}. \quad (9)$$

Lemma (Complexity Explosion): The computational complexity P_{comp} of the completed TS fuzzy model grows exponentially with the number of antecedents and the size of the model coefficients. Considering the multiplication operation only the computational requirement is characterized as

$$P_{comp} = \prod_n V_n \left(\sum_k \sum_l O_k I_l + \sum_k O_k \right) + C_p \sum_n V_n \quad (10)$$

where C_p indicates the number of multiplications during the calculation of a membership function.

To arrive at (10), one notes that calculating the output of one linear local model to a given input needs $\sum_k \sum_l O_k I_l$ multiplications. The number of the local linear models is $\prod_n V_n$, which actually comes from the exponential complexity problem of fuzzy rule bases shown in [14]. The outputs of the $\prod_n V_n$ local linear models are weighted by the products of the membership values, which implies $\prod_n V_n \cdot \sum_k O_k$ further multiplications. $C_p \sum_n V_n$ indicates the calculation of the membership values, where C_p represents the number of multiplications in the calculation of one membership value. Consequently, (10) shows that increasing the rule density, namely, the number of antecedents in pursue of good approximation, leads to the explosion of the computational requirement fully according to [14]

V. Key Concept of HOSVD-Based Reduction

This section briefly discusses the fundamentals of HOSVD in the sense of complexity reduction. Many reduction properties of the HOSVD of HO tensors are investigated in the related literature. Let us briefly summarize those, which have prominent roles in this paper. First of all, let the computation of HOSVD be discussed. It is done by executing SVD on each dimeniosn of tensor A. Namely, $\mathbf{U}_n$ is determined by executing SVD on the n-mode matrix $\mathbf{A}_{(n)}$ of tensor A. For instance, let us determine $\mathbf{U}_n$ and $\mathbf{U}_{n+1}$

$$\begin{aligned}\mathbf{A}_{(n)} &= [\,\mathbf{U}^r \quad \mathbf{U}^d\,] \begin{bmatrix} \mathbf{D}^r & \mathbf{0} \\ \mathbf{0} & \mathbf{D}^d \end{bmatrix} [\,\mathbf{V}^r \quad \mathbf{V}^d\,]^T \\ &= [\,\mathbf{U}^r \quad \mathbf{U}^d\,]\,\mathbf{S}'_{(n)} = \mathbf{U}_n \mathbf{S}'_{(n)}\end{aligned}$$

where "r" denotes "reduced" and "d" means "discarded, " which we will see later. Thus, the result is

$$A = S' \times_n \mathbf{U}_n.$$

The $n+1$ dimension is done in the same way. It performs SVD on the $n+1$ mode matrix of S' and yields

$$\mathbf{S}'_{(n+1)} = \mathbf{U}_{n+1} \mathbf{S}_{(n+1)}.$$

Therefore,

$$A = S \times_n \mathbf{U}_n \times_{n+1} \mathbf{U}_{n+1}.$$

In multilinear algebra as well as in matrix algebra, the *Frobenius*-norm is unitary invariant. As a consequence, the fact that the squared *Frobenius*-norm of a matrix equals the sum of its squared singular values can be generalized.

Property (Approximation): Let the HOSVD of A be given as in the Theorem of HOSVD and let the n-mode rank of A be equal to R_n. Define a tensor $\hat{A}$ by discarding singular values $\sigma^{(n)}_{I'_n+1}, \sigma^{(n)}_{I'_n+2}, \ldots, \sigma^{(n)}_{R_n}$ for given values of I'_n, i.e., when SVD is performed on dimension n discard $\mathbf{U}^d$, $\mathbf{D}^d$ and $\mathbf{V}^d$, where $\mathbf{D}^d$ contains singular values $\sigma^{(n)}_{I'_n+1}, \sigma^{(n)}_{I'_n+2}, \ldots \sigma^{(n)}_{R_n}$. Then, we have

$$\left\| A - \hat{A} \right\|^2 \leq \sum_{n=1}^{N} \left(\sum_{i_n = I'_n+1}^{R_n} \left(\sigma^{(n)}_{i_n} \right)^2 \right). \quad (11)$$

This property is the HO equivalent of the link between the SVD of a matrix and its best approximation in a least-squares sense, by a matrix of lower rank. The situation is, however, quite different for tensors. By discarding the smallest n-mode singular values, one obtains a tensor $\hat{A}$ with n-mode rank of I'_n. Unfortunately, this tensor is in general not the best possible approximation under the given n-mode rank constrains [24]. Nevertheless, the ordering implies that the main components of A are mainly concentrated in the part corresponding to low values of the indexes. Consequently, if $\sigma^{(n)}_{I'_n} \gg \sigma^{(n)}_{I'_{n+1}}$, where actually I'_n corresponds to the numerical rank of A then the smaller n-mode singular values are not significant, which implies their discarding. In this case, the obtained $\hat{A}$ is still considered as a good approximation of A. According to the special terms in this topic the following naming has emerged [12], [13].

Definition 13 (Exact/Nonexact Reduction): Assume an N-th order tensor $A \in \Re^{I_1 \times I_2 \times \cdots \times I_N}$. ***Exact*** *reduced form $A = A^r \underset{n}{\otimes} \mathbf{U}_n$, where "$r$" denotes "reduced", is defined by tensor $A^r \in \Re^{I^r_1 \times I^r_2 \times \cdots \times I^r_N}$ and n-mode singular matrices $\mathbf{U}_n \in \Re^{I_n \times I^r_n}$, $\forall n : I^r_n \leq I_n$ which are the results of Theorem HOSVD, where only the zero singular values and the corresponding singular vectors are discarded.* ***Nonexact*** *reduced form $\hat{A} = A^r \underset{n}{\otimes} \mathbf{U}_n$, is obtained if not only zero singular values and the corresponding singular vectors are discarded.*

VI. SVD-Based Complexity Reduction of TS Fuzzy Models

The main objective of the complexity reduction proposed in this section is twofold, which is discussed via two methods. Method 1 is aimed to minimalize values V_n, which means the decrease of the size of $B_{k,l}$ in the first N dimension, namely, to find the minimal number of fuzzy rules/local linear model. The reduction conducts HOSVD on tensor $B_{k,l}$ to root out linear dependencies by truncating zero or small singular values. In the first case, exact and in the latter nonexact reduction is obtained [10], [12], [13], [24]. First an exact reduction is discussed in this section, which means that the output of the reduced TS fuzzy model does not differ from the output of the original model. Increasing the effectiveness of the reduction by discarding nonzero singular values in HOSVD, reduction error is obtained which will be bounded in Remark 2 at the end of this section. A subsequent aim of the reduction methods to be proposed is to decrease values O_k and I_l which also appear in the dominant term of (10). The number of input and output values are defined by the application at hand, which implies that O_k and I_l cannot directly be decreased. Similarly to [10] the key idea of reducing these values can be viewed as the transformation of the system model to a smaller computational space offline. The input values are also projected in each state step of the model and the output values are calculated in the reduced computational space. Finally, the output values are transformed back to the original space. The reduction is based on executing SVD reduction to the coefficient matrices. As a matter of fact exact reduction cannot be obtained in this step if the coefficient matrices are full in rank, which is usually guaranteed by modeling processes. Nonexact reduction is, however, still possible at the price of reduction error. First, let us characterize the concept and the goal of the reduction by the following theorem.

Theorem (TS Fuzzy Model Reduction): Equation (9) can always be transformed into the following form:

$$\mathbf{z}_k = \left(\sum_l \left[B^r_{k,l} \bigotimes_n \mathbf{m}^r_n(p_n) \right] \times_{N+1} \mathbf{A}_k \times_{N+2} \mathbf{x}_l^T \mathbf{C}_l \right)_{(N+1)}$$

which is equivalent to

$$\mathbf{z}_k = \mathbf{A}_k \left(\sum_l \left[B^r_{k,l} \bigotimes_n \mathbf{m}^r_n(p_n) \right] \times_{N+2} \mathbf{x}_l^T \mathbf{C}_l \right)_{(N+1)} \quad (12)$$

where the size of $B^r_{k,l} \in \Re^{V_1^r \times V_2^r \times \cdots \times V_N^r \times O_k^r \times I_l^r}$ *may be reduced as* $\forall n : V_n^r \leq V_n, \forall k : O_k^r \leq O_k$ *and* $\forall l : I_l^r \leq I_l$.

$\mathbf{m}^r_n(p_n) \in \Re^{V_n^r}$ consists of the new antecedents which define the rules in the reduced rule base. The number of antecedents on the n-th universe is V_n^r. $\mathbf{A}_k \in \Re^{O_k \times O_k^r}$ and $\mathbf{C}_l \in \Re^{I_l \times I_l^r}$ are applied to transform the inputs and the outputs between the reduced and the original computational space, which we will see later at Method 2.

The proof of the theorem can readily be derived from the following Methods 1 and 2. Before starting with the methods let us have a brief digression and represent the calculation of values $\mathbf{z}_k$ of the TS fuzzy model in respect of $\mathbf{x}_l$ in two different ways as discussed in [10]. Let tensor $G_k \in \Re^{V_1 \times V_2 \times \cdots \times V_N \times O_k \times (\sum_l I_l)}$ be given by the form of $G_k = [\, B_{k,1} \quad B_{k,2} \quad \cdots \quad B_{k,L} \,]_{N+2}$. The output value $\mathbf{z}_k$ of the TS fuzzy model in respect of $\mathbf{x}_k$ is

$$\mathbf{z}_k = \left(\left[G_k \bigotimes_n \mathbf{m}_n(p_n) \right] \times_{N+2} [\, \mathbf{x}_1^T \quad \mathbf{x}_2^T \quad \cdots \quad \mathbf{x}_L^T \,] \right)_{(N+1)}.$$

The second way utilizes matrix $H_l \in \Re^{V_1 \times V_2 \times \cdots \times V_N \times (\sum_k O_k) \times I_l}$ constructed as $H_l = [\, B_{1,l} \quad B_{2,l} \quad B_{K,l} \,]_{N+1}$. The output of the TS fuzzy model is

$$\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_L \end{bmatrix} = \left(\left[[\, H_1 \quad H_2 \quad \cdots \quad H_L \,] \bigotimes_n \mathbf{m}_n(p_n) \right] \times_{N+2} [\, \mathbf{x}_1^T \quad \mathbf{x}_2^T \quad \cdots \quad \mathbf{x}_L^T \,] \right)_{(N+1)}. \quad (13)$$

The first method shows how to find the minimal number of rules.

Method 1 (Determination of the Minimal Values of $V_1, V_2, \ldots, V_N$): Let tensor $K \in \Re^{V_1 \times V_2 \times \cdots \times V_N \times \sum_k O_k \times \sum_l I_l}$ be constructed, so it contains all tensors $B_{k,l}$, for instance, $K = [\, G_1 \quad G_2 \quad \cdots \quad G_K \,]_{N+1}$, where $G_k = [\, B_{k,1} \quad B_{k,2} \quad \cdots \quad B_{k,L} \,]_{N+2}$. Applying HOSVD to the $N+2$-dimensional tensor K in such a way that the SVD is executed only on dimensions $1 \ldots N$ yields

$$K = K^r \bigotimes_n \mathbf{T}_n \quad (14)$$

where "r" denotes "reduced." Tensors $B^r_{k,l} \in \Re^{V_1^r \times V_2^r \times \cdots \times V_N^r \times O_l \times I_l}$ are found accordingly to construction of K, thus, for instance, $K^r = [\, G_1^r \quad G_2^r \quad \cdots \quad G_K^r \,]_{N+1}$ and $G_k^r = [\, B^r_{k,1} \quad B^r_{k,2} \quad \cdots \quad B^r_{k,L} \,]_{N+2}$. If singular values are discarded then the size of $B^r_{k,l} \in \Re^{V_1^r \times V_2^r \times \cdots \times V_N^r \times O_l \times I_l}$ is less than the size of $B_{k,l} \in \Re^{V_1 \times V_2 \times \cdots \times V_N \times O_l \times I_l}$, so, $\forall n : V_n^r \leq V_n$, which is the key point of the reduction. Thus, for (14), we obtain

$$B_{k,l} = B^r_{k,l} \bigotimes_n T_n. \quad (15)$$

The new antecedent sets of the rules are constructed as

$$\mathbf{m}^r_n(p_n) = \mathbf{m}_n(p_n)\, \mathbf{T}_n. \quad (16)$$

Consequently, (9) can be written in the reduced form by substituting (15) and (16) into (9) which yields

$$\mathbf{z}_k = \left(\sum_l \left[B^r_{k,l} \bigotimes_n \mathbf{m}^r_n(p_n) \right] \times_{N+2} \mathbf{x}_l^T \right)_{(N+1)}$$

which is in full accordance with the theorem of TS fuzzy model reduction. This finally obtained form has the same structure as (9). Therefore, it represents the same structured fuzzy rule base, but with different antecedents and consequents.

The objectives of Method 2 are to decrease O_k and I_l.

Method 2 (Determination of the Minimal Computational Space): Again, in the following steps we use SVD in exact mode. Remark 2 discusses the error bound when SVD is executed in nonexact mode, namely, nonzero singular values are discarded as well.

1) Determination of matrices $\mathbf{A}_k$, namely, the reduction of O_k.

   Let $\mathbf{S}_k = (G_k)_{(1)}$. Applying SVD to $\mathbf{S}_k$ yields:

$$\mathbf{S}_k = \mathbf{A}_k \cdot \mathbf{D}_k \cdot \mathbf{V}_k = \mathbf{A}_k \cdot \mathbf{S}'_k.$$

   Matrix $\mathbf{S}'_k \in \Re^{O_l^r \times \prod_n V_n \cdot \sum_l I_l}$ can be restored to tensor $G'_k \in \Re^{V_1 \times V_2 \times \cdots \times V_N \times O_k^r \times (\sum_l I_l)}$.

2) 2) Determination of matrices $\mathbf{C}_l$, namely, the reduction of I_l

   Let tensor $H'_k \in \Re^{V_1 \times V_2 \times \cdots \times V_N \times \sum_k O_k \times I_l}$ be constructed like in (13) as $H'_l = [\, B'_{1,l} \quad B'_{2,l} \quad \cdots \quad B'_{K,l} \,]_{N+1}$, where tensors $B'_{k,l}$ are defined accordingly to the result

$G'_k = [B'_{k,1} \quad B'_{k,2} \quad \cdots \quad B'_{k,L}]_{N+2}$ by step 1). Then, let $\mathbf{M}_l = (H'_l)_{(N+2)}$ whereupon executing SVD yields

$$\mathbf{M}_l = \mathbf{C}_l \cdot \mathbf{D}'_l \cdot \mathbf{V}'_l = \mathbf{C}_l \cdot \mathbf{M}'_l.$$

Matrix $\mathbf{M}'_l$ defines tensors $B^r_{k,l} \in \Re^{V_1 \times V_2 \times \cdots \times V_N \times O^r_k \times I^r_l}$ accordingly to $\mathbf{M}'_l = (H''_l)_{(N+2)}$ and $H''_l = [B^r_{1,l} \quad B^r_{2,l} \quad \cdots \quad B^r_{K,l}]_{N+1}$.

The results of Method 2 are $\mathbf{A}_k$ and $\mathbf{C}_l$. $\mathbf{C}_l$ is applied to transform the input values $\mathbf{x}_l$ to a reduced space as: $\mathbf{x}^r_l = \mathbf{C}^T_l \cdot \mathbf{x}_l$. The output is calculated in the reduced computational space as: $\mathbf{z}^r_k = \left(\sum_l \left[B^r_{k,l} \overset{\otimes}{_n} \mathbf{m}_n(p_n)\right] \times_{N+2} (\mathbf{x}^r_l)^T\right)_{(N+1)}$. The output $\mathbf{z}^r_k$ is projected to the original space by $\mathbf{z}_k = \mathbf{A}_k \cdot \mathbf{z}^r_k$, which is in full accordance with the theorem of TS fuzzy model reduction.

The ordering of executing Methods 1 and 2 is arbitrary. In the following, some important issues and interpretability problem of the results are discussed.

Remark 1: The functions in (16) obtained by Method 1 may not be interpretable as fuzzy sets, since the transformation using matrix $\mathbf{T}_n$ may result in functions with negative values. Another crucial point is that the resulted antecedent functions do not guarantee *Ruspini*-partition, which means that the denominator in (5) may not be equal to 1. This fact would destroy the whole reduction concept since calculating the denominator with the new antecedents may get far from 1. However, if only the saving computational cost in final implementation is the purpose and the fuzzy concept does not have to be accommodated then (9) and (8) are directly applicable to the reduced form, namely, (12) is applicable directly. If the reduced form is for further studies in fuzzy theory and/or Lyapunov stability, then the reduced weighting functions should accommodate additional characterization pertaining to specific operations. This may require further transformations. To obtain matrices $\mathbf{T}_n$ in such a way that the reduced membership functions are bounded by [0, 1] and hold *Ruspini*-partition, *nonnegativeness* and *sum-normalization* transformation techniques are developed in [12], [13] as discussed in the introduction. If the SVD is accompanied by these transformations then the resulted functions remain interpretable as antecedent fuzzy sets. Furthermore, the denominator of (5) becomes 1 (if it were true in the case of original rule base as well), which ensures the theoretically correct use of (9) and (8) in fuzzy concept. Furthermore, in some theoretical points proposed by Dubois *et al.* [31] for Generalized Modus Ponent, it is highly desired that the fuzzy sets conserve normalization property, i.e., when at least one element exists in each fuzzy set whose membership value is one. It is also called *localization of rules*. In order to serve this concept *normalization* transformation is proposed in [12]. Consequently, the computational cost of the algorithm may be decreased via the proposed methods in final implementation, which serves our main goal, but its price is that the interpretability of the fuzzy sets may be degraded. Actually, this is also an interesting point itself in fuzzy theory—how to represent and extract a rule base in different ways.

Remark 2: An advantage of the proposed algorithm is that it has error controllable property, i.e., if the HOSVD is executed in nonexact mode then the original and the reduced approximation differ and the difference can be estimated during executing the reduction technique. In Section V, it is shown that discarding nonzero singular values results in reduction error, which can be bounded by (11). References [12], [13], and [16] bound the maximum reduction error by the sum of the discarded singular values. As a matter of fact, the reduction errors of the proposed methods also depend on the antecedent sets applied. In this regard, various cases of antecedents are discussed in [16]. Generally speaking, it can be said that if the original antecedents are given in *Ruspini*-partitions then the maximum reduction error is the sum of the discarded singular values. For more details about the error bound of SVD reduction see [12], [13], and [16].

Remark 3: Method 1 may result in membership functions which cannot be analytically simplified and, hence, their shapes are rather complicated and their computational loads may be greater than that of the original ones. Observing (10), it is concluded that C_p is not in the dominant part of (10) which implies that this computational increase is dispensable compared to the exponential feature of the dominant term. In the worst case, the membership values of the observations are calculated by the original functions and the membership values of the reduced antecedents are simply determined by (16) in each step of the system. Consequently, the worst case is bounded by

$$P = \prod_n V^r_n \left(\sum_k \sum_l O^r_k I^r_k + \sum_k O^r_k\right) + C_p \sum_n V_n + \sum_n V_n V^r_n + \sum_k O_k O^r_k + \sum_l I_l I^r_l \quad (17)$$

where extra term $V_n V^r_n$ indicates the extra computational load of calculating the membership values of the observation in the reduced antecedents on the n-th universe. $\sum_k O_k O^r_k$ and $\sum_l I_l I^r_l$ are form the computation requirement of the transformation between the original and the reduced computational spaces. Consequently, the effectiveness of the reduction is shown by the equation at the bottom of the next page.

In the case of a dense or higher dimensional rule base its dominant part is

$$\eta \approx \frac{\prod_n V^r_n \left(\sum\limits_k \sum\limits_l O^r_k I^r_k + \sum\limits_k O^r_k\right)}{\prod_n V_n \left(\sum\limits_k \sum\limits_l O_k I_k + \sum\limits_k O_k\right)}.$$

Remark 4: Method 1 could be modified in such a way that the reduction results in one fuzzy rule base for each row or column of (2) like in [10]. Furthermore, one rule base could be resulted for each coefficient tensor $B_{k,l}$. The advantage of the reduction of each $B_{k,l}$ is that the size of some $B^r_{k,l}$ may decrease. This is due to the fact that the n-mode rank of tensor $B_{k,l}$ is less or equal to the n-mode rank of tensor K in (14). In the worst case, its maximum could be $\min(\sum_{k,l} rank_n(B_{k,l}), rows((K)_{(n)}))$.

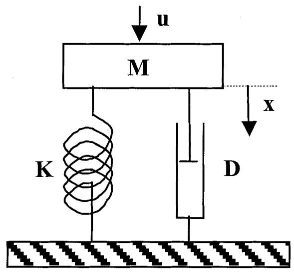

Fig. 2. Example: mass–spring–damper system.

Consequently, replacing K in (14) with $B_{k,l}$, the following is obtained:

$$B_{k,l} = B^r_{k,l} \underset{n}{\otimes} \mathbf{T}_{n,k,l}$$

and via (16) the new antecedents are: $\mathbf{m}^r_{n,k,l}(p_n) = \mathbf{m}_n(p_n)\mathbf{T}_{n,k,l}$, where antecedents defined by $\mathbf{m}^r_{n,k,l}(p_n)$ are assigned to the rule base approximating $B_{k,l}$. Again, the benefit is that the size of each $B^r_{k,l}$ is less or equal to the common $B^r_{k,l}$ resulted by Method 1. As a matter of fact, the calculation of the antecedents may increase since the membership values should be calculated for each rule base, however this extra calculation is not included in the exponentially dominant part of (10) and (17). This pinpointing of the reduction is burdened by the fact, that one has to check, whether performing the reduction for each coefficient tensor separately would yield a better computational reduction or not.

VII. Example

This example, taken from [4] and [10], is a design for a simple nonlinear mass–spring–damper mechanical system depicted in Fig. 2. The main goal of this example is to approximate the mass–spring–damper mechanical system (like a dynamically unknown one) by TS fuzzy model over a dense fuzzy partition. The reason for applying dense rule base is the goal of achieving a small approximation error. Then, the example performs the proposed reduction technique to find the minimal fuzzy partition. The differential equations of the mechanical system are analytically given in the minimal form of a TS fuzzy model as well as in order to evaluate the effectiveness of the reduction. The goal here is to show that the minimal form resulting from the proposed methods from training data is the same, in the sense of complexity, as the analytically derived TS model.

First let us discuss the dynamic model from "design example 2" of [4]. It is assumed that the stiffness coefficient of the spring, the damping coefficient of the damper and the input term have nonlinearity

$$m \cdot \ddot{x} + g(x, \dot{x}) + k(x) = \phi(\dot{x}) \cdot u \tag{18}$$

where m is the mass and u represents the force. $k(x)$ is the nonlinear or uncertain term with respect to the spring. $g(x, \dot{x})$ is the nonlinear or uncertain term with respect to the damper. $\phi(\dot{x})$ is the nonlinear term with respect to the input term. Assume that $g(x, \dot{x}) = d(c_1 x + c_2 \dot{x}^3)$, $k(x) = c_3 x + c_4 x^3$, and $\phi(\dot{x}) = 1 + c_5 \dot{x}^3$. Furthermore, assume that $x \in [-a, a]$, $\dot{x} \in [-b, b]$ and $a, b > 0$. The above parameters are set as follows [4]: $m = 1$, $d = 1$, $c_1 = 0.01$, $c_2 = 0.1$, $c_3 = 0.01$, $c_4 = 0.67$, $c_5 = 0$, $a = 1.5$ and $b = 1.5$. Equation (18) then becomes

$$\ddot{x} = -0.1\dot{x}^3 - 0.02x - 0.67x^3 + u. \tag{19}$$

The nonlinear terms are $-0.1\dot{x}^3$ and $-0.67x^3$. Let us proceed further in the same way as done in [4] and give a TS fuzzy model of (19) with minimal number of, namely, four fuzzy rules. x and $\dot{x}$ have the following conditions:

$$\begin{cases} -1.5075x \le -0.67x^3 \le 0 \cdot x, & x \ge 0 \\ 0 \cdot x \le -0.67x^3 \le -1.5075x, & x < 0 \end{cases}$$

and

$$\begin{cases} -0.225\dot{x} \le -0.1\dot{x}^3 \le \dot{x} \cdot 0, & x \ge 0 \\ 0 \cdot \dot{x} \le -0.1\dot{x}^3 \le -0.225\dot{x}, & x < 0 \end{cases}.$$

This fact means that the nonlinear term can be represented by the upper and the lower bounds:$-0.67x^3 = f_{1,1}(x)x \cdot 0 - (1 - f_{1,1}(x)) \cdot 1.5075x$ and $-0.1\dot{x}^3 = f_{2,1}(\dot{x})\dot{x} \cdot 0 - (1 - f_{2,1}(\dot{x})) \cdot 0.225\dot{x}$, where $f_{n,v_n}(\dot{x}) \in [0, 1]$, $V_n = 2$. This leads to fuzzy sets $F^a_{1,1} : f^a_{1,1}(x) = 1 - (x^2)/2.25$, ("$a$" means that the function is obtained analytically), $F^a_{1,2} : f^a_{1,2}(x) = (x^2)/2.25$; $F^a_{2,1} : f^a_{2,1}(\dot{x}) = 1 - (\dot{x}^2)/2.25$; $F^a_{2,2} : f^a_{2,2}(\dot{x}) = (\dot{x}^2)/2.25$. The antecedent functions are depicted in Fig. 4. Thus, the following rules are obtained analytically:

IF $x(t)$ is $F^a_{1,1}$ AND $\dot{x}(t)$ is $F^a_{2,1}$ THEN $\ddot{x} = -0.02x + u$
IF $x(t)$ is $F^a_{1,1}$ AND $\dot{x}(t)$ is $F^a_{2,2}$ THEN $\ddot{x} = -0.0225\dot{x} - 0.02x + u$
IF $x(t)$ is $F^a_{1,2}$ AND $\dot{x}(t)$ is $F^a_{2,1}$ THEN $\ddot{x} = -1.5275x + u$
IF $x(t)$ is $F^a_{1,2}$ AND $\dot{x}(t)$ is $F^a_{2,2}$ THEN $\ddot{x} = -0.0225\dot{x} - 1.5275x + u$.

Consequently, the TS fuzzy model in matrix representation takes the form

IF $x(t)$ is F^a_{1,v_1} AND $\dot{x}(t)$ is F^a_{2,v_2} THEN

$$\dot{\mathbf{x}}(t) = \mathbf{A}_{v_1,v_2}\mathbf{x}(t) + \mathbf{B}_{v_1,v_2}\mathbf{u}(t) \tag{20}$$

$$\eta = \frac{\prod_n V^r_n \left(\sum_k \sum_l O^r_k I^r_k + \sum_k O^r_k \right) + C_p \sum_n V_n + \sum_n V_n V^r_n + \sum_k O_k O^r_k + \sum_l I_l I^r_l}{\prod_n V_n \left(\sum_k \sum_l O_k I_k + \sum_k O_k \right) + C_p \sum_n V_n}$$

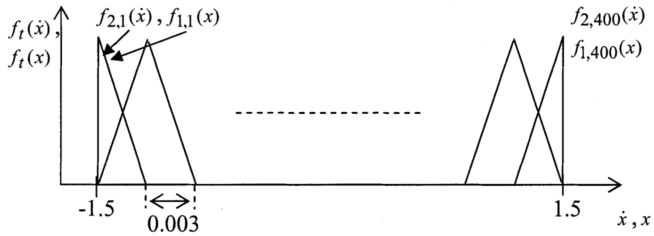

Fig. 3. Example: dense fuzzy partition to achieve a good approximation.

where

$$\mathbf{A}_{1,1}=\begin{bmatrix}0 & -0.02\\ 1 & 0\end{bmatrix}$$
$$\mathbf{B}_{1,1}=\begin{bmatrix}1\\ 0\end{bmatrix}$$
$$\mathbf{A}_{1,2}=\begin{bmatrix}-0.225 & -0.02\\ 1 & 0\end{bmatrix}$$
$$\mathbf{B}_{1,2}=\begin{bmatrix}1\\ 0\end{bmatrix}$$
$$\mathbf{A}_{2,1}=\begin{bmatrix}0 & -1.5275\\ 1 & 0\end{bmatrix}$$
$$\mathbf{B}_{2,1}=\begin{bmatrix}1\\ 0\end{bmatrix}$$
$$\mathbf{A}_{2,2}=\begin{bmatrix}-0.225 & -1.5275\\ 1 & 0\end{bmatrix}$$
$$\mathbf{B}_{2,2}=\begin{bmatrix}1\\ 0\end{bmatrix}.$$

The analytically obtained TS fuzzy model, consisting of four models, exactly represents the nonlinear system. The model has two antecedents in each parameter dimension, which is sufficient for the approximation. The next step is to approximate the model (18) with a dense rule base, after which we can assume that (18) is unknown and then go about generating a minimum rule base by the present technique. In order to simplify the example, let us define one of the simplest TS rule base types by simply sampling the differential equations over a 400 × 400 grid which yields 160 000 rules. This can imitate a fuzzy learning. As a matter of fact, learning from the training data set of the differential equation may result in a rule base which have a much less number of rules than 160 000. There is, however, no guarantee that the learning approaches lead to the minimum four rules as discussed in the introduction. The HOSVD technique can be executed on both the learned and on the sampled rule base in the same way. Therefore, without the loss of generality, we utilize the sampled rule base here. The aim is to show that the HOSVD technique finds the minimal four rules even from this over distended sampled rule base.

Let intervals $\dot{x}, x \in [-1.5, 1.5]$ be divided by 400 triangular shaped fuzzy sets (see Fig. 3).

The following rules are completed by the identification:

$$\begin{aligned}&\text{IF } x(t) \text{ is } F^b_{1,v_1} \text{ AND } \dot{x}(t) \text{ is } F^b_{2,v_2} \text{ THEN}\\ &\quad \dot{\mathbf{x}}(t) = \mathbf{A}_{v_1,v_2}\mathbf{x}(t) + \mathbf{B}_{v_1,v_2}\mathbf{u}(t), \text{ where}\\ &\quad V_n = 400.\end{aligned}$$

We sample the dynamic system at points $x_{v_1} = -1.5 + (v_1 - 1)3/400$ and $\dot{x}_{v_2} = -1.5 + (v_2 - 1)3/400$, which imitates the result of an identification algorithm like in [10]. The dense fuzzy model becomes

$$\begin{aligned}&\text{IF } x(t) \text{ is } F^b_{1,v_1} \text{ AND } \dot{x}(t) \text{ is } F^b_{2,v_2} \text{ THEN}\\ &\qquad\qquad \ddot{x}(t) = a_{v_1,v_2}\dot{x} + b_{v_1,v_2}x + c_{v_1,v_2}u\end{aligned}$$

where $a_{v_1,v_2} = -0.1(-1.5 + (v_2 - 1)3/400)^2$, $b_{v_1,v_2} = -0.02 - 0.67(-1.5 + (v_1 - 1)3/400)^2$ and $c_{v_1,v_2} = 1$. In matrix form,

$$\begin{aligned}&\text{IF } x(t) \text{ is } F^b_{1,v_1} \text{ AND } \dot{x}(t) \text{ is } F^b_{2,v_2} \text{ THEN}\\ &\quad \dot{\mathbf{x}}(t) = \mathbf{A}_{v_1,v_2}\mathbf{x}(t) + \mathbf{B}_{v_1,v_2}\mathbf{u}(t),\ V_n = 400.\end{aligned}$$

Executing Method 2 on matrices $\mathbf{A}_{v_1,v_2}$, namely, on tensor $\mathcal{A} \in \Re^{400\times400\times2\times2}$ (note that matrices $\mathbf{B}_{v_1,v_2}$ are equal) results in two nonzero singular values such as 461.6404... and 156.5663... to the first dimension and after performing SN and NN transformation two, such as 100.8708... and 1.8970... to the second dimension. The resulting coefficient matrices are

$$\begin{aligned}\mathbf{A}^r_{1,1} &= \begin{bmatrix}-169.595\ldots & -2.871\,86\ldots\\ 1 & 0\end{bmatrix}\\ \mathbf{A}^r_{1,2} &= \begin{bmatrix}338.965\ldots & -2.871\,86\ldots\\ 1 & 0\end{bmatrix}\\ \mathbf{A}^r_{2,1} &= \begin{bmatrix}-169.595\ldots & 3.895\,95\ldots\\ 1 & 0\end{bmatrix}\\ \mathbf{A}^r_{2,1} &= \begin{bmatrix}338.965\ldots & 3.895\,95\ldots\\ 1 & 0\end{bmatrix}.\end{aligned} \tag{21}$$

This means that two antecedent sets are sufficient on each dimension, which is in full accordance with the analytical TS fuzzy model design. As a result, we conclude that instead of applying the identified 400 × 400 rules only four rules are sufficient for the same approximation and the resulted antecedents maintain the *Ruspini*-partition. The PDC design and linear matrix inequality (LMI) computations can be restricted to the resulting four rules instead of the trained 160 000 rules.

We show analytically in the following that the obtained model is equivalent to (20). The new antecedent sets are piecewise linear. We approximate the break points of the pieces, which are actually the elements in the columns of $\mathbf{T}_n$ [16], by a polynomial fitting, which results in

$$\begin{aligned}F^r_{1,1} : f^r_{1,1}(x) &= \alpha_1 + \beta_1 x^2,\\ F^r_{1,2} : f^r_{1,2}(x) &= 1 - f^r_{1,1}(x),\\ F^r_{2,1} : f^r_{2,1}(\dot{x}) &= 1 - f^r_{2,2}(\dot{x})\\ F^r_{2,2} : f^r_{2,2}(\dot{x}) &= \alpha_2 + \beta_2 \dot{x}^2\end{aligned} \tag{22}$$

where $\alpha_1 = 0.578\,614\ldots$, $\beta_1 = 0.098\,997\ldots$, $\alpha_2 = 0.333\,480\ldots$ and $\beta_2 = -0.000\,442$. The antecedent functions are depicted in Fig. 4. Indeed, the rule base with antecedents given by (22) and consequents of (21) is a variant form of (20).

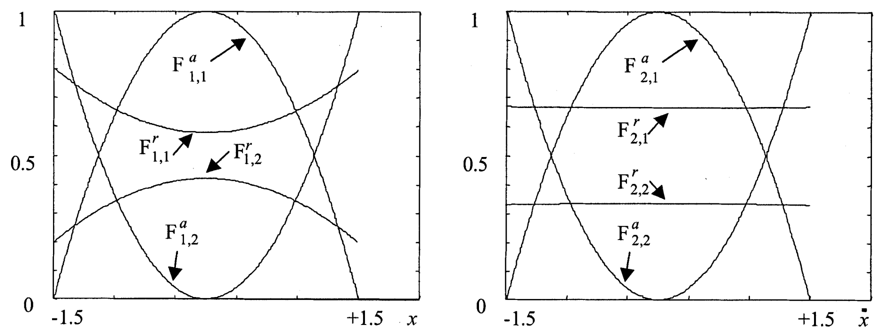

Fig. 4. Example: antecedent sets of the original rule base via analytical derivation and reduced sets extracted from training data.

VIII. CONCLUSION

In this paper, we have argued that the identification of TS fuzzy models from training data needs to consider an important feature between data fitness and model complexity. We emphasise the importance of these features by pointing out that a TS fuzzy model with a large number of fuzzy rules may encounter the risk of having an approximation capable of fitting training data well, but be incapable of running at low satisfactory computational cost. In order to help the developments of TS fuzzy models to find a balance between the two conflicting modeling objectives, we introduced a HOSVD-based TS fuzzy model reduction technique. Using the proposed method, we have demonstrated the application of HOSVD to constructing minimal sized local linear model consequent based fuzzy rule base. This approach is expounded from single-variable SVD-based reduction technique of SISO TS models proposed in [10] to HOSVD-based reduction capable of dealing with MISO TS models.

REFERENCES

[1] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. 15, no. 1, pp. 116–132, 1985.

[2] K. Tanaka, T. Ikeda, and O. H. Wang, "Fuzzy regulators and fuzzy observers: Relaxed stability conditions and LMI-based design," *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 250–265, May 1998.

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

[4] H. O. Wang, K. Tanaka, and M. F. P. Griffin, "An approach to fuzzy control of nonlinear systems: Stability and design issues," *IEEE Trans. Fuzzy Syst.*, vol. 4, pp. 14–23, Feb. 1996.

[5] R. J. Patton, J. Chen, and C. J. Lopez-Toribio, "Fuzzy observers for nonlinear dynamic systems fault diagnosis," in *Proc. 37th IEEE Conf. Decision and Control*, vol. 1, 1998, pp. 84–89.

[6] R. J. Patton, R. Clark, and P. M. Frank, *Issues of Fault Diagnosis for Dynamic Systems*, R. J. Patton, R. Clark, and P. M. Frank, Eds. Berlin, Germany: Springer-Verlag, 2000.

[7] J. Chen and R. J. Patton, *Robust Model Based Fault Diagnosis for Dynamic Systems*. Norwell, MA: Kluwer, 1999.

[8] H. O. Wang, K. Tanaka, and M. F. P. Griffin, "Parallel distributed compensation of nonlinear systems by Takagi and Sugeno fuzzy models," in *Proc. FUZZ-IEEE/IFES'95*, 1995, pp. 531–538.

[9] K. Tanaka, T. Ikeda, and H. O. Wang, "Robust stabilization of a class of uncertain nonlinear systems via fuzzy control: Quadratic stabilisability, $H\infty$ control theory and linear matrix inequalities," *IEEE Trans. Fuzzy Syst.*, vol. 4, pp. 1–13, Feb. 1996.

[10] P. Baranyi, Y. Yam, A. Várkonyi-Kóczy, R. J. Patton, P. Michelberger, and M. Sugiyama, "SVD based reduction to TS fuzzy models," *IEEE Trans. Ind. Electron.*, vol. 49, pp. 433–443, Apr. 2002.

[11] J. Yen and L. Wang, "Simplifying fuzzy rule-based models using orthogonal transformation methods," *IEEE Trans. Syst., Man, Cybern.*, pt. B, vol. 29, pp. 13–24, Feb. 1999.

[12] Y. Yam, P. Baranyi, and C. T. Yang, "Reduction of fuzzy rule base via singular value decomposition," *IEEE Trans. Fuzzy Syst.*, vol. 7, pp. 120–132, Apr. 1999.

[13] Y. Yam, "Fuzzy approximation via grid point sampling and singular value decomposition," *IEEE Trans. Syst., Man, Cybern.*, vol. 27, pp. 933–951, Dec. 1997.

[14] L. T. Kóczy and K. Hirota, "Size reduction by interpolation in fuzzy rule bases," *IEEE Trans. Syst., Man, Cybern.*, vol. 27, pp. 14–25, Feb. 1997.

[15] D. Tikk, "On nowhere denseness of certain fuzzy controllers containing prerestricted number of rules," *Tatra Mountains Math. Pub.*, vol. 16, pp. 369–377, 1999.

[16] 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. Norwell, MA: Kluwer, 2000, ch. 7, pp. 135–160.

[17] K. Lei, P. Baranyi, and Y. Yam, "Complexity minimalization of nonsingleton based fuzzy-neural network," *Int. J. Adv. Comput. Intell.*, vol. 4, no. 4, pp. 1–8, 2000.

[18] P. Baranyi, Y. Yam, P. Várlaki, and P. Michelberger, "Singular value decomposition of linguistically defined relations," *Int. J. Fuzzy Syst.*, vol. 2, no. 2, pp. 108–116, June 2000.

[19] G. W. Stewart, "On the early history of the singular value decomposition," *SIAM Rev.*, vol. 35, no. 4, pp. 551–566, Dec. 1993.

[20] I. J. Rudas and O. Kaynak, *New Types of Generalized Operations. Computational Intelligence: Soft Computing and Fuzzy-Neuro Integrationwith Applications*. ser. F: Computer and Systems, Sciences, O. Kaynak, L. A. Zadeh, B. Türksen, and I. J. Rudas, Eds. Berlin, Germany: Springer NATO ASI Series, 1998, vol. 192, pp. 128–156.

[21] I. J. Rudas and M. O. Kaynak, "Entropy-based operations on fuzzy sets," *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 33–40, Feb. 1998.

[22] ——, "Minimum and maximum fuzziness generalized operators," *Fuzzy Sets Syst.*, pp. 83–94, 1998.

[23] I. J. Rudas, "Toward the generalization of t-operators: A distance-based approach," *J. Inform. Org. Sci.*, vol. 23, no. 2, pp. 149–166, 1999.

[24] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A multi linear singular value decomposition," *SIAM J. Matrix Anal. Applicat.*, vol. 21, no. 4, pp. 1253–1278, 2000.

[25] A. Swami and G. Giannakis, "Editorial—Higher-order statistics," *Special Issue on Higher Order Statistics, Signal Process.*, vol. 53, no. 2–3, pp. 89–91, 1996.

[26] P. Comon, "Independent component analysis, a new concept?," *Special Issue on Higher Order Statistics, Signal Process.*, vol. 36, pp. 287–314, 1994.

[27] M. Moonen and B. D. Moor, *SVD and Signal Processing II—Algorithms, Applications and Architectures*, M. Moonen and B. D. Moor, Eds. Amsterdam, The Netherlands: Elsevier, 1995.

[28] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "An introduction to independent component analysis," *J. Chemometrics*, vol. 14, pp. 123–149.

[29] P. Baranyi, Y. Yam, A. R. Várkonyi-Kóczy, R. J. Patton, and P. Michelberger, "Increasing the density of approximation points in SVD reduced form," Chinese Univ. Hong Kong, Hong Kong, Rep. CUHK-ACAE-01–02, 2001.

[30] C. J. Lopez-Toribio, R. J. Patton, and S. Daley, "Takagi-Sugeno fuzzy fault tolerantcontrol of an induction motor," *J. Neuro-Fuzzy Syst., Special Issue on NeuroComputing and Applications*, 1999.

[31] D. Dubois and H. Prade, "What are fuzzy rules and how to use them," *Fuzzy Sets Syst.*, vol. 84, pp. 169–185, 1996.

[32] F. Song and S. M. Smith, "A simple based fuzzy logic controller rule base reduction mathod," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics (IEEE SMC 2000)*, Nashville, TN, 2000, pp. 3794–3798.

[33] M. Setnes and H. Hellendoorn, "Orthogonal transforms for ordering and reduction of fuzzy rules," in *Proc. 9th IEEE Int. Conf. Fuzzy Systems (FUZZ-IEEE 2000)*, San Antonio , TX, 2000, pp. 700–705.

[34] T. Sudkamp, A. Knapp, and J. Knapp, "A greedy approach to rule reduction in fuzzy models," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics (IEEE SMC 2000)*, Nashville, TN, 2000, pp. 3716–3721.

[35] D. Simon, "Design and rule base reduction of a fuzzy filter for the estimation of motor currents," *Int. J. Approx. Reasoning*, no. 2, pp. 145–167, Oct. 2000.

[36] C. T. Yang, P. Baranyi, Y. Yam, and S. Kovács, "SVD reduction of a fuzzy controller in an AGV steering system," in *Proc. EFDAN'99*, Dortmund, Germany, 1999, pp. 118–124.

[37] S. C.-T. Yang, "Fuzzy rule based identification via singular value decomposition," Ph.D. dissertation, Dept. Mech. Automat. Eng., Chinese Univ. Hong Kong, Hong Kong, 1999.

[38] P. Baranyi, Y. Yam, C. T. Yang, P. Várlaki, and P. Michelberger, "Inference algorithm independent SVD fuzzy rule base complexity reduction," *Int. J. Adv. Comput. Intell.*, vol. 5, no. 1, 2001.

[39] P. Baranyi and Y. Yam, "Singular value-based approximation with non-singleton fuzzy rule base," in *Proc. 7th Int. Fuzzy Systems Association World Congr. (IFSA'97)*, Prague, 1997, pp. 127–132.

[40] ——, "Singular value-based approximation with Takagi-Sugeno type fuzzy rule base," in *Proc. 6th IEEE Int. Conf. Fuzzy Systems (FUZZ-IEEE'97)*, Barcelona, Spain, 1997, pp. 265–270.

[41] S. Kovács and L. T. Kóczy, "Application of the approximate fuzzy reasoning based on interpolation in the vague environment of the fuzzy rule-base in the fuzzy logic controlled path tracking strategy of differential steered AGVS," in *Computational Intelligence—Theory and Applications*. Berlin, Germany: Springer, 1997, vol. 1226, Lecture Notes in Computer Science, pp. 456–467.

[42] Y. Yam, P. Baranyi, and C. T. Yang, "Author's reply," *IEEE Trans. Fuzzy Syst.*, vol. 9, pp. 676–677, Aug. 2001.

[43] C. W. Tao, "Comments on "Reduction of fuzzy rule base via singular value decomposition"," *IEEE Trans. Fuzzy Syst.*, vol. 9, pp. 675–676, Aug. 2001.

Péter Baranyi was born in Hungary in 1970. He received the M.Sc. degree in electrical engineering, the M.Sc. degree in education of engineering sciences, and the Ph.D. degree from the Technical University of Budapest, Budapest, Hungary, in 1994, 1995, and 1999, respectively.

He has held research positions at The Chinese University of Hong Kong (1996 and 1998), The University of New South Wales, Australia (1997), the CNRS LAAS Institute, Toulouse, France (1996), and the Gifu Research Institute, Japan (2000–2001). His research interests include fuzzy and neural network techniques.

Dr. Baranyi received the Youth Prize of the Hungarian Academy of Sciences and the International Dennis Gábor Award in 2000. He is the Secretary General of the Hungarian Society of the International Fuzzy Systems Association. He was a founding member of the Integrated Intelligent Systems Japanese-Hungarian Laboratory.

Yeung Yam received the B.S. degree in physics from The Chinese University of Hong Kong, Hong Kong, in 1975, the M.S. degree in physics from the University of Akron, Akron, OH, in 1977, and the M.S. and Sc.D. degrees in aeronautics and astronautics from Massachusetts Institute of Technology, Cambridge, in 1979 and 1983, respectively.

He is currently an Associate Professor in the Department of Automation and Computer-Aided Engineering at The Chinese University of Hong Kong. Before joining the university in 1992, he was with the Control Analysis Research Group of the Guidance and Control Section at the Jet Propulsion Laboratory, Pasadena, CA. His research interests include dynamics modeling and control, system identification, fuzzy design, analysis, and approximation

Annamária R. Várkonyi-Kóczy (M'95–SM'97) was born in Budapest, Hungary, in 1957. She received the M.Sc.E.E., M.Sc. M.E.-T., and Ph.D. degrees from the Technical University of Budapest (currently Budapest University of Technology and Economics), Budapest, Hungary, in 1981, 1983, and 1996, respectively.

She was a Researcher with the Research Institute for Telecommunication, Budapest, Hungary, for six years, followed by four years with the Hungarian Academy of Science, where she was a Research Associate in the Engineering Mechanics Group. She then transferred to the Technical University of Budapest, where, since 1991, she has been with the Department of Measurement and Information Systems, currently as an Associate Professor. Her research interests include digital signal processing, uncertainty handling, soft computing, anytime and hybrid techniques in complex measurement, diagnostics, and control systems.

Dr. Várkonyi-Kóczy is the Vice Chair of the Hungarian Fuzzy Association (member of the International Fuzzy Systems Association) and a member of the European Association for Signal Processing (EURASIP), Hungarian Academy of Engineers, John von Neumann Computer Society (Hungary), and Measurement and Automation Society (Hungary).

Ron J. Patton (M'78) was born in Peru in 1949. He received the M.Sc. and Ph.D. degrees in electrical and electronic engineering and control systems from Sheffield University, Sheffield, U.K., in 1974 and 1980, respectively.

He has worked in the hospital service in medical physics and was a founding member of the Electronics Laboratory at the Royal Free Hospital, London, U.K., in 1967–1968. During 1973–1974, he was with the BBC Research Department, Kingswood Warren, U.K. After completing his Ph.D. studies in 1976, he worked for GEC Electrical Projects, Rugby and Sheffield City Polytechnic on dynamic ship positioning control systems. He became a Lecturer at Sheffield Hallam University in 1978 and moved to the new Electronics Department at York in 1981, where he focused on fault diagnosis and aerospace control systems, with promotion to Senior Lecturer in 1987. In 1995. he was appointed Professor of Control and Intelligent Systems Engineering at the University of Hull, where his research interests are fault-tolerant control and the use of AI/soft computing methods for fault diagnosis in control systems. He has authored more than 250 papers and five books on eigenstructure assignment for control systems design, fault diagnosis and fault-tolerant control.

Prof. Patton is the Chair of the Technical Committee *SAFEPROCESS* of the Interational Federation of Automatic Control.