**Preprints of the 19th World Congress**
**The International Federation of Automatic Control**
**Cape Town, South Africa. August 24-29, 2014**

# Multilinear Approximation of Nonlinear State Space Models ⋆

**Kai Kruppa** * **Georg Pangalos** * **Gerwald Lichtenberg** *

* *Faculty of Life Sciences, Hamburg University of Applied Sciences, Ulmenliet 20, 21033 Hamburg, Germany (e-mail: {Kai.Kruppa, Georg.Pangalos, Gerwald.Lichtenberg}@haw-hamburg.de)*

**Abstract:** There are numerical tools for representation, editing and simulation of continuous-time state space models, e.g. numerous block-based tools like SIMULINK. Algorithms for numerical and symbolical linearization are available but structurally fail to compute models where nonlinear effects are essential. The paper presents a numerical method to find a multilinear model as approximation which is valid in appropriate regions usually given by the engineering application. The approximate model belongs to the class of Multilinear Time Invariant systems (MTI systems) which can be represented within a tensor framework and thus, tensor decomposition methods are applicable. The proposed approximation algorithm is applied to a test case of a chemical system, which is inherently nonlinear and simulation results show the improvement of the approximation of the behaviour compared to a linear model.

Keywords: Model reduction; Application of nonlinear analysis and design.

## 1. INTRODUCTION

This paper provides a multilinearization method for nonlinear state space models. This task is formulated as a minimization problem. The solution can be given in terms of an orthogonality condition yielding a multilinear model in tensor representation which was introduced in Lichtenberg [2011]. The method is implemented numerically for the case that no analytical representation of the model is known, but e.g. a SIMULINK model can be evaluated.

The validity of our approach is illustrated by an example. A model of a chemical system is derived. By investigation of mass and energy balances which form the basis of the model, one can observe many multilinear terms, i.e. different states or inputs are multiplied but no squares or higher orders of the states or inputs occur. Systems that are modeled e.g. by mass, energy or heat balances like chemical systems or heating systems often show an inherent multilinear structure. Thus multilinearization of similar systems should give good approximation results.

The multilinearity of heating systems was the inspiration to model them as tensor systems, cf. Pangalos et al. [2013]. Tensor calculus is an active field of research, see e.g. Hackbusch [2012]. We have already shown that standard tensor decomposition methods are applicable to tensor systems, Pangalos et al. [2013].

The paper is organized as follows. Section 2 gives an introduction to tensor systems and is followed by the multilinearization method in Section 3. An application example is provided in Section 4; conclusions are drawn in Section 5.

⋆ This work was partly supported by the project ModQS of the Federal Ministry of Economics and Technology, Germany.

## 2. TENSOR SYSTEMS

In this section tensor systems are introduced for which the definitions can be found in Lichtenberg [2011] or Lichtenberg [2010]. A first example states a multilinear system in matrix form. A translation to a tensor system is provided. Throughout the paper it is assumed that all states can be measured. An output function is not considered in this paper.

First the monomial vector is introduced to state the multilinear state space.

*Definition 1.* The *monomial vector* is defined as

$$\mathbf{m}(\mathbf{x},\mathbf{u}) = \begin{pmatrix}1\\u_m\end{pmatrix} \otimes \cdots \otimes \begin{pmatrix}1\\u_1\end{pmatrix} \otimes \begin{pmatrix}1\\x_n\end{pmatrix} \otimes \cdots \otimes \begin{pmatrix}1\\x_1\end{pmatrix} \quad (1)$$

where $\mathbf{x} \in \mathbb{R}^n$ with elements $x_i$, $i = 1,\ldots,n$ is the state vector and $\mathbf{u} \in \mathbb{R}^m$ with elements $u_j$, $j = 1,\ldots,m$ is the input vector and $\otimes$ denotes the Kronecker product.

For ease of notation $\beta$ will be an abbreviation for $2^{n+m}$. The state transition function of a multilinear state space model in matrix representation is given by

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{m}(\mathbf{x},\mathbf{u}) \,, \quad (2)$$

where $\mathbf{F} \in \mathbb{R}^{n\times\beta}$ is the transition matrix.

*Example 2.1.* A second order model with one input reads

$$\begin{pmatrix}\dot{x}_1\\\dot{x}_2\end{pmatrix} = \begin{pmatrix}f_{11}&f_{12}&f_{13}&f_{14}&f_{15}&f_{16}&f_{17}&f_{18}\\f_{21}&f_{22}&f_{23}&f_{24}&f_{25}&f_{26}&f_{27}&f_{28}\end{pmatrix} \begin{pmatrix}1\\x_1\\x_2\\x_1x_2\\u\\ux_1\\ux_2\\ux_1x_2\end{pmatrix} . \quad (3)$$

*Proposition 2.* Consider linear transformations for states and inputs $x_i = a_i\tilde{x}_i + b_i$ and $u_j = a_{n+j}\tilde{u}_j + b_{n+j}$, with indices $i = 1,\ldots,n$ and $j = 1,\ldots,m$. The state transition function in the transformed coordinates is given by

$$\widetilde{\mathbf{F}} = \underset{i=1,\ldots,n}{\text{diag}} \left({a_i}^{-1}\right) \mathbf{FT}$$

with diag denoting the diagonal matrix with elements $a_i^{-1}$ and a transformation matrix

$$\mathbf{T} = \begin{pmatrix} 1 & 0 \\ b_{n+m} & a_{n+m} \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} 1 & 0 \\ b_1 & a_1 \end{pmatrix}.$$

*Proof:* Inserting the state $x_i$ and input $u_j$ transformations with $\dot{x}_i = a_i \dot{\tilde{x}}_i$ in (2) leads to

$$\dot{\tilde{\mathbf{x}}} = \underset{i=1,\ldots,n}{\text{diag}} \left({a_i}^{-1}\right) \mathbf{FTm}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) = \widetilde{\mathbf{F}}\mathbf{m}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}). \qquad \square$$

The state transition function in matrix representation (2) is translated to a state transition tensor in the following. We need standard definitions of tensors and contracted products, which can be found e.g. in Kolda and Bader [2009] or Cichocki et al. [2009].

*Definition 3.* A Tensor

$$\mathsf{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$$

of order $n$ is an $n$-way array where elements $x_{i_1 i_2 \cdots i_n}$ are indexed by $i_j \in \{1, 2, \ldots, I_j\}$ for $j = 1, \ldots, n$.

*Definition 4.* A canonical polyadic (CP) tensor

$$\mathsf{K} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n] \cdot \boldsymbol{\lambda} \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_n}$$

is a tensor of dimension $r_1 \times \cdots \times r_n$, with elements given by the sums of the outer products of the column vectors of so-called *factor matrices* $\mathbf{X}_i \in \mathbb{R}^{r_i \times r}$, weighted by the elements of the so-called *weighting* or *parameter vector* $\boldsymbol{\lambda}$. An element of the tensor $\mathsf{K}$ is given by

$$K_{jk\ldots p} = \sum_{i=1}^{r} \lambda_i \, (X_1)_{ji} \;\; (X_2)_{ki} \;\; \ldots (X_n)_{pi} \, .$$

If no weighting vector is given, it is assumed to be a vector of ones, i.e. $\lambda = (1 \quad 1 \quad \cdots \quad 1)^T$.

*Definition 5.* The *contracted product* of two tensors $\mathsf{X}$ and $\mathsf{Y}$ with $\mathsf{X} \in \mathbb{R}^{I_1 \times \cdots \times I_n \times I_{n+1} \times \cdots \times I_{n+m}}$ and $\mathsf{Y} \in \mathbb{R}^{I_1 \times \cdots \times I_n}$

$$\langle \mathsf{X} \mid \mathsf{Y} \rangle (k_1, \ldots, k_m) = \sum_{i_1=1}^{I_1} \cdots \sum_{i_n=1}^{I_n} x_{i_1 \ldots i_n k_1 \ldots k_m} y_{i_1 \ldots i_n}, \quad (4)$$

with $k_i \in \{1, 2, \ldots, I_{n+i}\}$, $i = 1, \ldots, m$ is a tensor of dimension $I_{n+1} \times \cdots \times I_{n+m}$ . The notation $\mathbb{R}^{\times^{(n+m)}2}$ will denote the space $\mathbb{R}^{\overbrace{2 \times \ldots \times 2}^{n+m \text{ times}}}$. The state transition function of a tensor system is given in terms of a contracted tensor product

$$\dot{\mathbf{x}} = \langle \mathsf{F} \mid \mathsf{M}(\mathbf{x}, \mathbf{u}) \rangle \, , \quad (5)$$

with the state transition tensor $\mathsf{F} \in \mathbb{R}^{\times^{(n+m)}2 \times n}$ and the monomial CP tensor

$$\mathsf{M}(\mathbf{x}, \mathbf{u}) = \left[ \begin{pmatrix} 1 \\ u_m \end{pmatrix}, \ldots, \begin{pmatrix} 1 \\ u_1 \end{pmatrix}, \begin{pmatrix} 1 \\ x_n \end{pmatrix}, \ldots, \begin{pmatrix} 1 \\ x_1 \end{pmatrix} \right] \in \mathbb{R}^{\times^{(n+m)}2}.$$

The scalar elements of a tensor of dimension $\mathbb{R}^{\times^{(n+m)}2}$ are indexed by a subscript vector $\mathbf{i} = (i_1 \; \cdots \; i_\beta)$. Since every element of $\mathbf{i}$ can take 2 values, i.e. 1 or 2, there are $2^{n+m}$ different index vectors possible. They are denoted by $\mathbf{i}_i$, $i = 1, \ldots, 2^{n+m}$. Thus the elements of $\mathsf{M}(\mathbf{x},\mathbf{u})$ are given by $\mu_{\mathbf{i}_i}(\mathbf{x},\mathbf{u})$. Since the state transition tensor $\mathsf{F}$ has one additional dimension its scalar elements are denoted by $\varphi_{\mathbf{i}_i,j}$ with $j = 1, \ldots, n$ indexing the last dimension.

As contracted products can efficiently be computed in CP form, a decomposition

$$\mathsf{F} = [\mathbf{F}_{u_m}, \ldots, \mathbf{F}_{u_1}, \mathbf{F}_{x_n}, \ldots, \mathbf{F}_{x_1}, \mathbf{F}_\Phi] \cdot \boldsymbol{\lambda}_f \quad (6)$$

of the state transition tensor having $r_x$ factors is desirable as illustrated by Example 2.1.

Let $\mathbf{O} = (1 \; 0)^T$ and $\mathbf{I} = (0 \; 1)^T$ for better readability. With factor matrices

$$\begin{aligned} \mathbf{F}_{x_1} &= (\mathbf{O}\,\mathbf{I}\,\mathbf{O}\,\mathbf{I}\,\mathbf{O}\,\mathbf{I}\,\mathbf{O}\mathbf{I}\mathbf{O}\,\mathbf{I}\,\mathbf{O}\,\mathbf{I}\,\mathbf{O}\,\mathbf{I}\,\mathbf{O}\mathbf{I}) \\ \mathbf{F}_{x_2} &= (\mathbf{O}\mathbf{O}\,\mathbf{I}\,\mathbf{I}\,\mathbf{O}\mathbf{O}\,\mathbf{I}\,\mathbf{I}\mathbf{O}\mathbf{O}\,\mathbf{I}\,\mathbf{I}\,\mathbf{O}\mathbf{O}\,\mathbf{I}\,\mathbf{I}) \\ \mathbf{F}_u &= (\mathbf{O}\mathbf{O}\mathbf{O}\mathbf{O}\,\mathbf{I}\,\mathbf{I}\,\mathbf{I}\,\mathbf{I}\mathbf{O}\mathbf{O}\mathbf{O}\mathbf{O}\,\mathbf{I}\,\mathbf{I}\,\mathbf{I}\,\mathbf{I}) \\ \mathbf{F}_\Phi &= \begin{pmatrix} 1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1 \end{pmatrix}, \end{aligned}$$

and a 16 elements parameter vector

$$\boldsymbol{\lambda}_f = (\, f_{11} \;\; f_{12} \;\; f_{13} \;\; f_{14} \;\; f_{15} \;\; f_{16} \;\; f_{17} \;\; f_{18} \;\; \ldots \;\; f_{21} \;\; f_{22} \;\; f_{23} \;\; f_{24} \;\; f_{25} \;\; f_{26} \;\; f_{27} \;\; f_{28} \,)^T$$

the state transition tensor

$$\mathsf{F} = [\mathbf{F}_u, \mathbf{F}_{x_2}, \mathbf{F}_{x_1}, \mathbf{F}_\Phi] \cdot \boldsymbol{\lambda}_f$$

can be constructed. The contracted product of the decomposed tensors can be calculated by simple matrix operations see Lichtenberg [2011]

$$\langle \mathsf{F} \mid \mathsf{M}(\mathbf{x}, \mathbf{u}) \rangle = \mathbf{F}_\Phi \left( \boldsymbol{\lambda}_f \circledast \left( \mathbf{F}_u^T \begin{pmatrix} 1 \\ u \end{pmatrix} \right) \circledast \cdots \left( \mathbf{F}_{x_2}^T \begin{pmatrix} 1 \\ x_2 \end{pmatrix} \right) \circledast \left( \mathbf{F}_{x_1}^T \begin{pmatrix} 1 \\ x_1 \end{pmatrix} \right) \right) \right) , \quad (7)$$

where the Hadamard (elementwise) product is denoted by $\circledast$. The contracted product (7) is equal to the right hand side of the transition equation (3).

## 3. MULTILINEARIZATION METHOD

In this part we look for a multilinear approximation

$$\begin{aligned} h(\mathbf{x}) &= \langle \mathsf{F} \mid \mathsf{M}(\mathbf{x}) \rangle = \sum_{i=1}^{2^n} \varphi_{\mathbf{i}_i} \mu_{\mathbf{i}_i}(\mathbf{x}) \quad (8) \\ &= \varphi_{\mathbf{i}_1} \cdot 1 + \varphi_{\mathbf{i}_2} \cdot x_1 + \varphi_{\mathbf{i}_3} \cdot x_2 + \varphi_{\mathbf{i}_4} \cdot x_1 x_2 + \cdots + \varphi_{\mathbf{i}_\beta} \cdot x_1 x_2 \cdots x_n \end{aligned}$$

with $\mathbf{x} \in \mathbb{R}^n$ and $\mathsf{F} \in \mathbb{R}^{\times^{(n)}2}$ of a given nonlinear function $f(\mathbf{x})$. Here $\beta$ is equal to $2^n$. The approximation should be valid in a certain domain $D \subset \mathbb{R}^n$

$$x_i \in [x_{i,l}, x_{i,u}], i = 1, \ldots, n,$$

which is bounded by the fixed extreme values $x_{i,l} < x_{i,u}$ of the variables. The approach will be extended in subsection 3.3 to approximate nonlinear systems.

### *3.1 Approximation problem*

To find the best multilinear approximation $h(\mathbf{x})$ of a nonlinear function $f(\mathbf{x})$, a measure for the distance between approximation and original function has to be defined. The inner product space of the nonlinear functions $f(\mathbf{x})$ is denoted by $\mathcal{N}$ with the inner product of two functions $f_1(\mathbf{x}) \in \mathcal{N}$ and $f_2(\mathbf{x}) \in \mathcal{N}$

$$\langle f_1(\mathbf{x}), f_2(\mathbf{x}) \rangle_w := \int_D f_1(\mathbf{x}) f_2(\mathbf{x}) w(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \; w(\mathbf{x}) > 0, \quad (9)$$

where $w(\mathbf{x})$ is a weighting function used to define the importance of good approximation properties for certain regions in the domain $D$. With this inner product a norm

$$\|f(\mathbf{x})\|_w := \sqrt{\langle f(\mathbf{x}), f(\mathbf{x}) \rangle_w} \quad (10)$$

can be defined. The space of multilinear functions $\mathcal{M}$ is a subset of $\mathcal{N}$. The approximation problem can be formulated as follows

$$\min_{h(\mathbf{x})\in\mathcal{M}} \|h(\mathbf{x}) - f(\mathbf{x})\|_w \ . \tag{11}$$

The solution is the best approximation $h(\mathbf{x}) \in \mathcal{M}$ of the nonlinear function $f(\mathbf{x})$ with respect to the given norm (10). Since $\mathcal{N}$ is an inner product space it is strict convex and the approximation problem has a unique solution, Schaback and Werner [1992]. A polynomial $h(\mathbf{x})$ solves this problem (11), if and only if the orthogonality condition

$$\left\langle \hat{h}(\mathbf{x}), h(\mathbf{x}) - f(\mathbf{x}) \right\rangle_w = 0, \ \forall \hat{h}(\mathbf{x}) \in \mathcal{M} \tag{12}$$

is fulfilled, Kincaid and Cheney [1996]. This means that $h(\mathbf{x})$ is the best approximation of $f(\mathbf{x})$, if and only if the difference between $h(\mathbf{x})$ and $f(\mathbf{x})$ is orthogonal to all other elements $\hat{h}(\mathbf{x})$ of $\mathcal{M}$. A base $\{\mu_{\mathbf{i}_1}(\mathbf{x}), \dots, \mu_{\mathbf{i}_\beta}(\mathbf{x})\}$ of $\mathcal{M}$ is given by the elements of the monomial tensor $\mathsf{M}(\mathbf{x})$. This base is called monomial base in the following. The function $\hat{h}(\mathbf{x})$ can be expressed with respect to the monomial base $\{\mu_{\mathbf{i}_i}(\mathbf{x})\}$ like $h(\mathbf{x})$ in (8)

$$\hat{h}(\mathbf{x}) = \left\langle \hat{\mathsf{F}} \mid \mathsf{M}(\mathbf{x}) \right\rangle = \sum_{k=1}^{2^n} \hat{\varphi}_{\mathbf{i}_k} \mu_{\mathbf{i}_k}(\mathbf{x}), \tag{13}$$

with $\hat{\mathsf{F}} \in \mathbb{R}^{\times^{(n)}2}$. Inserting (13) in (12) leads to

$$\left\langle \sum_{k=1}^{2^n} \hat{\varphi}_{\mathbf{i}_k} \mu_{\mathbf{i}_k}(\mathbf{x}) \ , \ h(\mathbf{x}) - f(\mathbf{x}) \right\rangle_w = 0. \tag{14}$$

This results in a linear system of $2^n$ equations

$$\langle \mu_{\mathbf{i}_k}(\mathbf{x}), h(\mathbf{x}) - f(\mathbf{x}) \rangle_w = 0, \quad k = 1, \dots, 2^n. \tag{15}$$

The unknowns of these equations are the coefficients $\varphi_{\mathbf{i}_i}$ of the function $h(\mathbf{x})$. Inserting (8) and rearranging gives

$$\sum_{i=1}^{2^n} \varphi_{\mathbf{i}_i} \langle \mu_{\mathbf{i}_k}(\mathbf{x}), \mu_{\mathbf{i}_i}(\mathbf{x}) \rangle_w = \langle \mu_{\mathbf{i}_k}(\mathbf{x}), f(\mathbf{x}) \rangle_w, k = 1, \dots, 2^n. \tag{16}$$

Now it is possible to represent (16) as matrix equation

$$\mathbf{\Phi} \begin{pmatrix} \varphi_{\mathbf{i}_1} \\ \vdots \\ \varphi_{\mathbf{i}_\beta} \end{pmatrix} = \begin{pmatrix} \langle f(\mathbf{x}), \mu_{\mathbf{i}_1}(\mathbf{x}) \rangle_w \\ \vdots \\ \langle f(\mathbf{x}), \mu_{\mathbf{i}_\beta}(\mathbf{x}) \rangle_w \end{pmatrix}, \tag{17}$$

with

$$\mathbf{\Phi} = \begin{pmatrix} \langle \mu_{\mathbf{i}_1}(\mathbf{x}), \mu_{\mathbf{i}_1}(\mathbf{x}) \rangle_w & \cdots & \langle \mu_{\mathbf{i}_1}(\mathbf{x}), \mu_{\mathbf{i}_\beta}(\mathbf{x}) \rangle_w \\ \vdots & \ddots & \vdots \\ \langle \mu_{\mathbf{i}_\beta}(\mathbf{x}), \mu_{\mathbf{i}_1}(\mathbf{x}) \rangle_w & \cdots & \langle \mu_{\mathbf{i}_\beta}(\mathbf{x}), \mu_{\mathbf{i}_\beta}(\mathbf{x}) \rangle_w \end{pmatrix}. \tag{18}$$

The solution of (17) are the desired coefficients of the best approximation $h(\mathbf{x}) = \langle \mathsf{F} \mid \mathsf{M}(\mathbf{x}) \rangle$, Ackleh [2010]. The matrix $\mathbf{\Phi}$ is called Gramian matrix. It is difficult to solve (17) reliably because the Gramian matrix is ill conditioned, therefore we try to simplify the system by choosing an appropriate base of $\mathcal{M}$.

### *3.2 Orthonormalization*

An orthonormal base $\{\overline{\mu}_{\mathbf{i}_1}(\mathbf{x}), \dots, \overline{\mu}_{\mathbf{i}_\beta}(\mathbf{x})\}$ of $\mathcal{M}$ is computed to simplify (17) which means that

$$\left\langle \overline{\mu}_{\mathbf{i}_i}(\mathbf{x}), \overline{\mu}_{\mathbf{i}_j}(\mathbf{x}) \right\rangle_w = \begin{cases} 0 & \text{for } i \neq j \\ 1 & \text{for } i = j \end{cases} \ \forall i, j = 1, \dots, 2^n. \tag{19}$$

A tensor $\overline{\mathsf{M}}(\mathbf{x}) \in \mathbb{R}^{\times^{(n)}2}$ with the orthonormal base elements $\overline{\mu}_{\mathbf{i}_i}(\mathbf{x})$, $i = 1, \dots, 2^n$ is constructed in conformity with $\mathsf{M}(\mathbf{x})$ and its elements $\mu_{\mathbf{i}_i}(\mathbf{x})$. To orthonormalize the monomial base the orthogonalization algorithm of Gram-Schmidt is applied, Ackleh [2010]. It iteratively gives the new basis vectors

$$\overline{\mu}_{\mathbf{i}_1}(\mathbf{x}) = \frac{\mu_{\mathbf{i}_1}(\mathbf{x})}{\|\mu_{\mathbf{i}_1}(\mathbf{x})\|_w},$$

$$\overline{\mu}_{\mathbf{i}_k}(\mathbf{x}) = \frac{\mu_{\mathbf{i}_k}(\mathbf{x}) - \sum_{i=1}^{k-1} \left\langle \mu_{\mathbf{i}_k}(\mathbf{x}), \overline{\mu}_{\mathbf{i}_i}(\mathbf{x}) \right\rangle_w \overline{\mu}_{\mathbf{i}_i}(\mathbf{x})}{\left\| \mu_{\mathbf{i}_k}(\mathbf{x}) - \sum_{i=1}^{k-1} \left\langle \mu_{\mathbf{i}_k}(\mathbf{x}), \overline{\mu}_{\mathbf{i}_i}(\mathbf{x}) \right\rangle_w \overline{\mu}_{\mathbf{i}_i}(\mathbf{x}) \right\|_w}, k = 2, \dots, \beta.$$

Now the matrix (18) can be computed with the orthonormal base $\left\{\overline{\mu}_{\mathbf{i}_i}(\mathbf{x})\right\}$. Because of orthonormality all off-diagonal elements of $\mathbf{\Phi}$ are zero and the diagonal elements are equal to one. Therefore the matrix equation reads

$$\mathbf{I}_\beta \begin{pmatrix} \overline{\varphi}_{\mathbf{i}_1} \\ \vdots \\ \overline{\varphi}_{\mathbf{i}_\beta} \end{pmatrix} = \begin{pmatrix} \left\langle f(\mathbf{x}), \overline{\mu}_{\mathbf{i}_1}(\mathbf{x}) \right\rangle_w \\ \vdots \\ \left\langle f(\mathbf{x}), \overline{\mu}_{\mathbf{i}_\beta}(\mathbf{x}) \right\rangle_w \end{pmatrix}, \tag{20}$$

where $\mathbf{I}_\beta$ denotes a $2^n \times 2^n$ identity matrix. By solving this system of equations, the parameters $\overline{\varphi}_{\mathbf{i}_i}$ of the multilinear approximation $h(\mathbf{x})$ of the function $f(\mathbf{x})$ with respect to the orthonormal base $\{\overline{\mu}_{\mathbf{i}_i}(\mathbf{x})\}$ are identified resulting in

$$h(\mathbf{x}) = \sum_{i=1}^{2^n} \overline{\varphi}_{\mathbf{i}_i} \overline{\mu}_{\mathbf{i}_i}(\mathbf{x}) = \left\langle \overline{\mathsf{F}} \mid \overline{\mathsf{M}}(\mathbf{x}) \right\rangle. \tag{21}$$

The elements of the tensor $\overline{\mathsf{F}} \in \mathbb{R}^{\times^{(n)}2}$ are computed by (20) as

$$\overline{\varphi}_{\mathbf{i}_i} = \left\langle f(\mathbf{x}), \overline{\mu}_{\mathbf{i}_i}(\mathbf{x}) \right\rangle_w, \quad \forall i = 1, \dots, 2^n \ , \tag{22}$$

which is the orthogonal projection of the nonlinear function $f(\mathbf{x})$ on the orthonormal base polynomials, Kincaid and Cheney [1996]. The orthonormal and the monomial base span the same space $\mathcal{M}$, thus all elements of the orthonormal base can be expressed by a linear combination of the monomial base elements

$$\begin{pmatrix} \overline{\mu}_{\mathbf{i}_1}(\mathbf{x}) \\ \vdots \\ \overline{\mu}_{\mathbf{i}_\beta}(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} \vartheta_{11} & \cdots & \vartheta_{1\beta} \\ \vdots & \ddots & \vdots \\ \vartheta_{\beta 1} & \cdots & \vartheta_{\beta\beta} \end{pmatrix} \begin{pmatrix} \mu_{\mathbf{i}_1}(\mathbf{x}) \\ \vdots \\ \mu_{\mathbf{i}_\beta}(\mathbf{x}) \end{pmatrix}. \tag{23}$$

Inserting (23) in (21) leads to the function

$$h(\mathbf{x}) = \sum_{i=1}^{2^n} \varphi_{\mathbf{i}_i} \mu_{\mathbf{i}_i}(\mathbf{x}) = \langle \mathsf{F} \mid \mathsf{M}(\mathbf{x}) \rangle, \tag{24}$$

which is the same as (21) but is now expressed with respect to the base $\{\mu_{\mathbf{i}_i}\}$ given by the monomial tensor. This is the multilinear approximation $h(\mathbf{x})$ of the nonlinear function $f(\mathbf{x})$ that minimizes (11).

### *3.3 Extension to nonlinear systems*

The main purpose of this paper is to find a multilinear approximation

$$\dot{\mathbf{x}} = \langle \mathsf{F} \mid \mathsf{M}(\mathbf{x}, \mathbf{u}) \rangle = (h_1(\mathbf{x}, \mathbf{u}) \ \cdots \ h_n(\mathbf{x}, \mathbf{u}))^T \tag{25}$$

of a given nonlinear system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) = (f_1(\mathbf{x}, \mathbf{u}) \ \cdots \ f_n(\mathbf{x}, \mathbf{u}))^T \tag{26}$$

with $n$ states $\mathbf{x}$ and $m$ inputs $\mathbf{u}$. The approximation should be valid in a certain domain $D$, which is bounded by the intervals of the operating range of the variables

$$x_i \in [x_{i,l}, x_{i,u}], i = 1, \dots, n \ , \tag{27}$$

$$u_j \in [u_{j,l}, u_{j,u}], j = 1, \dots, m. \tag{28}$$

The interval boundaries $x_{i,l} < x_{i,u}$ and $u_{j,l} < u_{j,u}$ of the operation range are usually given by the engineering application. Because of the $n + m$ variables of the function $\mathbf{f}(\mathbf{x}, \mathbf{u})$, $\beta$ is equal to $2^{n+m}$ here.
The approach of sections 3.1 and 3.2 is applicable to each of the functions $f_j(\mathbf{x}, \mathbf{u}),\ j = 1, \ldots, n$ in (26) to compute the approximations $h_j(\mathbf{x}, \mathbf{u})$ elementwise

$$h_j(\mathbf{x},\mathbf{u}) = \langle\, \mathsf{F} \,|\mathsf{M}(\mathbf{x},\mathbf{u})\,\rangle(j) = \sum_{i=1}^{2^{n+m}} \varphi_{\mathbf{i}_i,j}\mu_{\mathbf{i}_i}(\mathbf{x},\mathbf{u}), \forall j = 1, \ldots, n,$$

with the monomial tensor $\mathsf{M}(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^{\times^{(n+m)}2}$ with scalar elements $\mu_{\mathbf{i}_i}$ and the state transition tensor $\mathsf{F} \in \mathbb{R}^{\times^{(n+m)}2\times n}$ with scalar elements $\varphi_{\mathbf{i}_i,j}$. Thus the approximation problem is given for each function $f_j(\mathbf{x}, \mathbf{u}), j = 1, \ldots, n$ by

$$\min_{h_j(\mathbf{x},\mathbf{u})\in\mathcal{M}} \|h_j(\mathbf{x}, \mathbf{u}) - f_j(\mathbf{x}, \mathbf{u})\|_w \ , \forall j = 1, \ldots, n\,. \quad (29)$$

The tensor system

$$\dot{\mathbf{x}} = \langle\, \mathsf{F} \mid \mathsf{M}(\mathbf{x}, \mathbf{u}) \,\rangle = \left\langle\, \overline{\mathsf{F}} \;\middle|\; \overline{\mathsf{M}}(\mathbf{x}, \mathbf{u}) \,\right\rangle \quad (30)$$

with the orthonormal base $\left\{\overline{\mu}_{\mathbf{i}_i}(\mathbf{x}, \mathbf{u})\right\}$ minimizes (29) analogous to (22) if the scalar elements of $\overline{\mathsf{F}}$ are equal to

$$\overline{\varphi}_{\mathbf{i}_i,j} = \left\langle f_j(\mathbf{x},\mathbf{u}), \overline{\mu}_{\mathbf{i}_i}(\mathbf{x},\mathbf{u})\right\rangle_w, \forall i = 1,\ldots,\beta, \forall j = 1,\ldots,n. \quad (31)$$

Inserting (23) in (30) gives the desired coefficients $\varphi_{\mathbf{i}_i,j}$ of $\mathsf{F}$ with respect to the monomial base $\{\mu_{\mathbf{i}_i}(\mathbf{x}, \mathbf{u})\}$ resulting in the tensor system (25) which is the best approximation of the nonlinear system (26) with respect to the given norm (10).

### *3.4 Numerical implementation*

Up to this point the system (26) to be approximated was assumed to be known analytically. Here the application of the multilinearization method to a nonlinear model realized in SIMULINK is investigated. One way to compute the introduced method is numerically. The resulting changes will be shown in the following.

The SIMULINK model has to be evaluated in a sufficiently large number $N$ of sampling points for each of the variables $x_1, \ldots, x_n$ and $u_1, \ldots, u_m$ in the operating range of the system to get the dataset

$$(\mathbf{x}_i, \mathbf{u}_j, \mathbf{f}(\mathbf{x}_i, \mathbf{u}_j)), \forall i = 1, \ldots, N^n,\ \forall j = 1, \ldots, N^m. \quad (32)$$

Since the nonlinear model is only known in the sampling points, numerical integration has to be used to compute the inner product (9), e.g. by using the trapezoidal rule, Phillips and Taylor [1996].

Because of the inaccuracies introduced by numerical integration the calculation of the parameters $\varphi_{\mathbf{i}_i,j}$ contains some errors. These errors gain influence if arbitrary large intervals are used for the variables $\mathbf{x}$ and $\mathbf{u}$. A large error occurs if the intervals (27) and (28) contain high offsets, e.g. if the states or inputs represent temperatures in Kelvin. As a consequence the polynomials of the orthonormal base $\{\overline{\mu}_{\mathbf{i}_i}(\mathbf{x}, \mathbf{u})\}$ get large coefficients in the offset terms. Calculating the parameters of the multilinear approximation by (31) numerically introduces some errors in the parameters $\overline{\varphi}_{\mathbf{i}_i,j}$. Multiplying these parameters with the large coefficients of the orthonormal base polynomials in $\left\langle\, \overline{\mathsf{F}} \;\middle|\; \overline{\mathsf{M}}(\mathbf{x}, \mathbf{u}) \,\right\rangle$ amplifies the error, such that there will be a large error of the parameters $\varphi_{\mathbf{i}_i,j}$ with respect to the monomial base polynomials $\mu_{\mathbf{i}_i}(\mathbf{x}, \mathbf{u})$ in $\langle\, \mathsf{F} \mid \mathsf{M}(\mathbf{x}, \mathbf{u}) \,\rangle$. To avoid such numerical problems it is necessary to scale the variables to fixed intervals as shown in Section 2 resulting in the scaled data set

$$(\tilde{\mathbf{x}}_i, \tilde{\mathbf{u}}_j, \mathbf{f}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{u}}_j)),\ \forall i = 1, \ldots, N^n,\ \forall j = 1, \ldots, N^m. \quad (33)$$

Computing the multilinear approximation of the system in the scaled variables $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{u}}$ gives the tensor system $\dot{\tilde{\mathbf{x}}} = \left\langle\, \tilde{\mathsf{F}} \mid \mathsf{M}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \,\right\rangle$, which can be transformed by (4) to get the system $\langle\, \mathsf{F} \mid \mathsf{M}(\mathbf{x}, \mathbf{u}) \,\rangle$ with variables $\mathbf{x}$ and $\mathbf{u}$ in the desired intervals (27) and (28).

Another problem occurs if a model with a large number of states $n$ and inputs $m$ is approximated, which is often the case for real plants. The increase of the number of states and inputs $n + m$ leads to an exponential increase of the number $n\ 2^{n+m}$ of parameters that have to be identified. The number of sampling points in the data set (32) increases by $N^{n+m}$. This leads to a high computational effort. Having a closer look at some examples of systems with an inherently multilinear structure one can see that often many parameters in $\mathsf{F}$ are equal to zero especially the coefficients related to monomials where many states and inputs are multiplied.

*Definition 6.* The *multilinear order* describes the number of variables that are multiplied in a multilinear monomial.

A function of maximal multilinear order $k$ has only monomials with an order lower or equal to $k$. Bounding the maximal multilinear order for the approximation leads to less parameters to be identified. The parameters concerning the higher order terms are set to 0. The number of elements in the multilinear bases $\{\mu_{\mathbf{i}_i}\}$ and $\{\overline{\mu}_{\mathbf{i}_i}\}$ are reduced too. Table 1 shows the number of parameters for different maximal multilinear orders $\eta_i(n)$, $i = 1, 2, 3$ of a function with $n$ variables. The index $i$ describes the maximal multilinear order.

| **n** | **full order** | $\boldsymbol{\eta_1(n)}$ | $\boldsymbol{\eta_2(n)}$ | $\boldsymbol{\eta_3(n)}$ |
|---|---|---|---|---|
| 1 | 2 | 2 | 2 | 2 |
| 2 | 4 | 3 | 4 | 4 |
| 3 | 8 | 4 | 7 | 8 |
| 4 | 16 | 5 | 11 | 15 |
| 5 | 32 | 6 | 16 | 26 |
| 6 | 64 | 7 | 22 | 42 |
| 7 | 128 | 8 | 29 | 64 |
| 8 | 256 | 9 | 37 | 93 |
| 9 | 512 | 10 | 46 | 130 |
| 10 | 1024 | 11 | 56 | 176 |
| 11 | 2048 | 12 | 67 | 232 |
| 12 | 4096 | 13 | 79 | 299 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

Table 1. Number of parameters for a system of certain order

The necessary steps to perform a multilinearization of a SIMULINK model can be summarized as follows.

*Algorithm 1.* Multilinearization

- Choose a maximal multilinear order,
- Specify the operation range $D$ of the system,
- Choose the number $N$ of sampling points for $\mathbf{x}$ and $\mathbf{u}$
- Evaluate the SIMULINK model in the sampling points to get the dataset (32),
- Orthonormalize the monomial base of chosen order
- Scale the determined data to the interval $[-1, 1]$,
- Calculate the parameters of the best multilinear approximation $\dot{\tilde{\mathbf{x}}} = \left\langle\, \tilde{\mathsf{F}} \mid \mathsf{M}(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \,\right\rangle$ of the scaled system

- Inverse transformation of the parameters in $\tilde{\mathsf{F}}$ by (4) to get the tensor $\mathsf{F}$ and thus the best multilinear approximation (25) of the nonlinear model in the desired operating range $D$.

## 4. APPLICATION

In this section the proposed multilinearization method is applied to a chemical system and simulation results are shown. The model belongs to a class of systems which can not be described appropriately by LTI models from an application point of view.

### *4.1 Chemical system*

The model of the chemical system consists of a reactor with inflow and outflow, a stirrer and jacket cooling, see Fig. 1 and is called constant stirred tank reactor (CSTR). It is often used in industrial applications, e.g. for cyclopentenol production, cf. Chen and Allgöwer [1998].

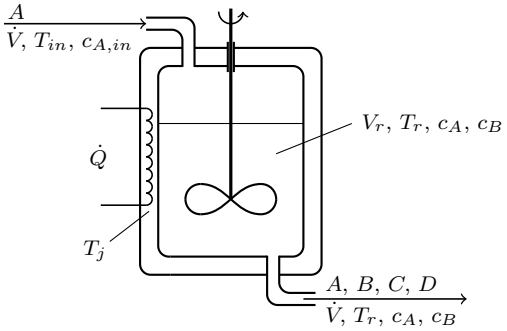


Fig. 1. Constant stirred tank reactor (CSTR)

The model was introduced by Engell and Klatt [1993] as benchmark process for nonlinear controllers. The reaction taking place inside the reactor follows the reaction scheme of van de Vusse

$$A \rightarrow B \rightarrow C,\ 2A \rightarrow D.$$

The educt $A$ reacts to the desired product $B$. Two byproducts C and D arise out of unwanted side reactions. The production of $B$ is influenced by the user by controlling the volume flow of the inflowing educt $A$ and the power of the jacket cooling. The constituent reactions inside the reactor are modeled by mass and energy balances resulting in a model with 4 states and 2 inputs

$$\mathbf{x} = [c_A\ c_B\ T_r\ T_j]^T,\ \mathbf{u} = \left[\frac{\dot{V}}{V_r}\ \dot{Q}\right]^T, \tag{34}$$

where $c_A$ is the concentration of educt $A$, $c_B$ is the concentration of product $B$, $T_r$ is the temperature inside the reactor and $T_j$ is the temperature of the jacket. The input $u_1$ is the incoming volume flow $\dot{V}$ related to the volume $V_r$ of the reactor. The inflow contains the educt $A$ with a concentration of $c_{A,in}$ and has the temperature $T_{in}$. Since the volume of the medium inside the reactor is constant, the outgoing volume flow is equal to the inflow. The power $\dot{Q}$ of the jacket cooling is used as second input.

Considering the mass balances of the elements inside the reactor the concentrations of educt $A$ and product $B$ are computed as in Rothfuss et al. [1996] by

$$\dot{c}_A = -k_1(T_r)c_A - k_2(T_r)c_A^2 + (c_{A,in} - c_A)u_1, \tag{35}$$
$$\dot{c}_B = k_1(T_r)c_A - k_1(T_r)c_B - c_B u_1, \tag{36}$$

where $k_1(T_r)$ and $k_2(T_r)$ are reaction rates depending on the temperature $T_r$ and are given by Arrhenius laws

$$k_i(T_r) = k_{i0}\exp\left(-\frac{E_i}{T_r}\right),\ i = 1, 2, \tag{37}$$

with parameters $k_{i0}$ and $E_i$, $i = 1, 2$. Using the energy balance of the reactor and the jacket the temperatures $T_r$ and $T_j$ are given by

$$\dot{T}_r = -\frac{1}{\rho C_p}\left[k_1(T_r)\left(\Delta H_{AB}c_A + \Delta H_{BC}c_B\right) + \cdots\right.$$
$$\left. k_2(T_r)\Delta H_{AD}c_A^2\right] + \alpha\left(T_r - T_j\right) + \left(T_{in} - T_r\right)u_1, \tag{38}$$
$$\dot{T}_j = \beta(T_r - T_j) + \gamma u_2, \tag{39}$$

with density $\rho$, heat capacity $C_p$, heat transfer coefficients $\alpha$, $\beta$ and $\gamma$ and reaction enthalpies $\Delta H_{AB}$, $\Delta H_{BC}$ and $\Delta H_{AD}$. The numerical values of all parameters can be found in Utz et al. [2006]. Inserting (37) to (35), (36) and (38) results in a system of four nonlinear differential equations, such that the behavior of the introduced chemical system is described by a nonlinear model

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}). \tag{40}$$

The example was chosen to check Algorithm 1 and to show the performance of multilinear approximations of the nonlinear model in comparison to linear approximations. Next, the differential equations (40) are implemented in SIMULINK by the block diagram shown in Fig. 2.

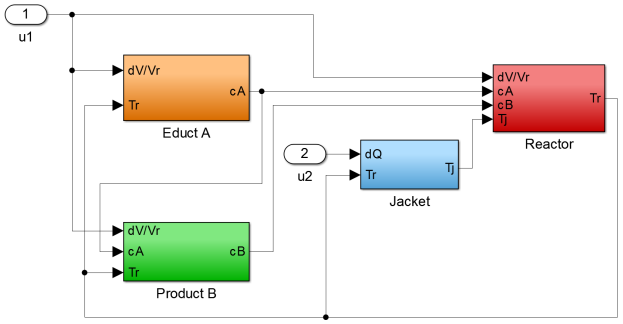


Fig. 2. CSTR system in Simulink

### *4.2 Multilinearization*

For application of the multilinearization method of Section 3 the user has to set up the parameters of the Algorithm 1. The weighting function of the inner product (9) is set to 1 for this example. The nonlinear model (40) of the CSTR system has 4 states and 2 inputs which add up to 6 variables of the multilinear functions $f_j(\mathbf{x}, \mathbf{u})$, $j = 1, \ldots, 4$ with $\mathbf{x} \in \mathbb{R}^4$ and $\mathbf{u} \in \mathbb{R}^2$.

The corresponding full state transition tensor $\mathsf{F}$ consists of $2^6 = 64$ parameters to be estimated. The bases $\{\mu_{\mathbf{i}_i}(\mathbf{x}, \mathbf{u})\}$ and $\{\overline{\mu}_{\mathbf{i}_i}(\mathbf{x}, \mathbf{u})\}$ would have 64 elements each. As applying Algorithm 1 would lead to a long computation time the maximal multilinear order is bounded as described in Section 3. For the CSTR system example it is sufficient to use a multilinear model with maximum order 2. By this restriction the number of parameters as well as base elements are reduced by factor 3 to 22, see Table 1.

The operating ranges of the systems states and inputs are defined as

$$c_A \in \left[0\frac{kmol}{l}, 1\frac{kmol}{l}\right],\ c_B \in \left[0\frac{mol}{l}, 0.75\frac{kmol}{l}\right],\ T_r \in [104°C, 116°C],$$
$$T_j \in [104°C, 116°C],\quad \frac{\dot{V}}{V_r} \in \left[0h^{-1}, 2.5h^{-1}\right],\quad \dot{Q} \in \left[0\frac{kJ}{h}, -1\frac{MJ}{h}\right].$$

A good trade-off between computation time and numerical accuracy for the number of sampling points for this Example has been shown to be $N = 10$ leading to $N^{n+m} = 10^6$ evaluations of the SIMULINK model stored in the dataset (32). Executing the Algorithm 1 results in a state transition function denoted by $\dot{\mathbf{x}} = \langle\, \mathsf{F}_{10} \mid \mathsf{M}(\mathbf{x}, u) \,\rangle$. A linear approximation $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$ of (40) around an operating point inside the operation range is computed by the MATLAB/SIMULINK built-in function `linmod`.

Fig. 3 shows the input trajectories and the simulated state trajectories of the nonlinear system and the multilinear and linear approximations.

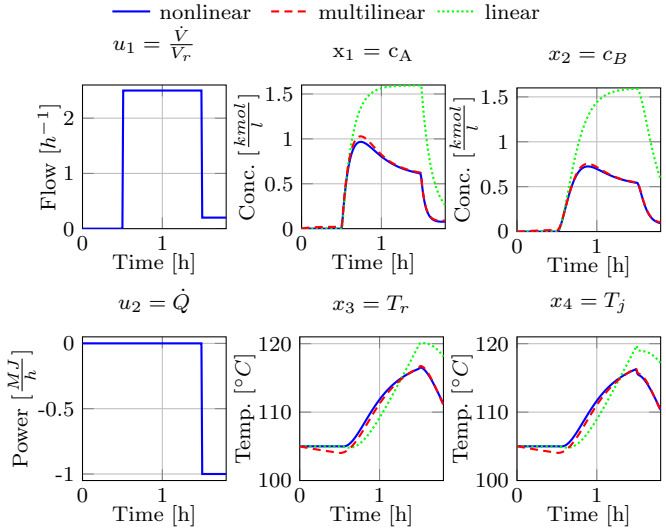


Fig. 3. Input trajectories and simulated state trajectories of nonlinear, multilinear and linear model

The simulation result shows the advantages of the multilinear approximation compared to the linearization of the original system. The differences between the trajectories of the original and the multilinear approximated system are much smaller than the differences between original and linear approximation. The error of the multilinear approximation results from the nonlinear terms of the CSTR system (40), e.g. the exponential terms (37), that cannot be described exactly by multilinear terms. But the approximation is much better than in the linear case. The numerical error of the multilinear approximation can be reduced further by increasing the number of sampling points $N$. The estimated tensor system is a good and physical meaningful approximation of the given SIMULINK model.

## 5. CONCLUSIONS AND FUTURE WORKS

This paper shows a method to multilinearize a nonlinear state space model. The result is a multilinear state space model represented as a tensor system. To find the best approximation, a minimization problem is formulated and solved by a system of linear equations which can be simplified by determining an orthonormal base of the multilinear functions. The parameters of the multilinear approximation can be computed by orthogonal projection of the orthonormal base elements on the nonlinear function. During the implementation some numerical problems occurred, which could be handled by scaling the data to fixed intervals reducing the numerical error. The computational effort was reduced by introducing a maximal multilinear order. The application of the presented multilinearization method on a chemical systems example shows its advantages over a linear approximation.

The multilinearization method is designed for systems with continuous signals. The multilinear behaviour of heating systems is shown in Pangalos et al. [2013] which justifies modeling them as tensor systems. Since heating systems often show hybrid behavior, see e.g. Pangalos and Lichtenberg [2012], the extension of the multilinearization method to hybrid tensor systems is of interest. Further research is going in the direction of finding efficient controller design procedures for MTI systems.

## REFERENCES

D. Ackleh. *Classical and Modern Numerical Analysis: Theory, Methods and Practice*. CRC Press, 2010.

H. Chen and F. Allgöwer. A Computationally Attractive Nonlinear Predictive Control Scheme with Guaranteed Stability for Stable Systems. *Journal of Process Control*, 8(5-6):475–485, 1998.

A. Cichocki, R. Zdunek, A. Phan, and S. Amari. *Nonnegative Matrix and Tensor Factorizations*. Wiley, Chichester, 2009.

S. Engell and K.-U. Klatt. Nonlinear Control of a Non-Minimum-Phase CSTR. In *American Control Conference*, pages 2941–2945. IEEE, 1993.

W. Hackbusch. *Tensor Spaces and Numerical Tensor Calculus*, volume 42 of *Springer Series in Computational Mathematics*. Springer-Verlag Berlin Heidelberg, 2012. ISBN 978-3-642-28026-9.

D. Kincaid and E. Cheney. *Numerical Analysis: Mathematics of Scientific Computing*. Brooks/Cole Publ., 1996.

T. Kolda and B. Bader. Tensor Decompositions and Applications. *SIAM Review*, 51(3):455–500, 2009. doi: 10.1137/07070111X.

G. Lichtenberg. Tensor Representation of Boolean Functions and Zhegalkin Polynomials. In *International Workshop on Tensor Decompositions*, 2010.

G. Lichtenberg. *Hybrid Tensor Systems*. Habilitation, Hamburg University of Technology, 2011.

G. Pangalos and G. Lichtenberg. Approach to Boolean Controller Design by Algebraic Relaxation for Heating Systems. In *4th IFAC Conference on Analysis and Design of Hybrid Systems*, 2012.

G. Pangalos, A. Eichler, and G. Lichtenberg. Tensor Systems: Multilinear Modeling and Applications. In *3rd Int. Conference on Simulation and Modeling Methodologies, Technologies and Applications*, 2013.

G. Phillips and P. Taylor. *Theory and Applications of Numerical Analysis*. San Diego Academic Press, 1996.

R. Rothfuss, J. Rudolph, and M. Zeitz. Flatness Based Control of a Nonlinear Chemical Reactor Model. *Automatica*, 32(10):1433–1439, 1996.

R. Schaback and H. Werner. *Numerische Mathematik*. Springer-Verlag Berlin Heidelberg, 4. edition, 1992.

T. Utz, V. Hagenmeyer, Bernd Mahn, and M. Zeitz. Nonlinear Model Predictive and Flatness-based Two-degree-of-freedom Control Design: A Comparison Evaluation in View of Industrial Application. In *International Conference on Control Applications*, pages 217–223. IEEE, 2006.