Methods

Gerwald Lichtenberg*, Georg Pangalos, Carlos Cateriano Yáñez, Aline Luxa, Niklas Jöres, Leona Schnelle and Christoph Kaufmann

Implicit multilinear modeling

Implizite multilineare Modellierung

An introduction with application to energy systems

Eine Einführung mit Anwendung auf Energiesysteme

https://doi.org/10.1515/auto-2021-0133
Received September 16, 2021; accepted November 19, 2021

Abstract: The paper introduces a subclass of nonlinear differential-algebraic models of interest for applications. By restricting the nonlinearities to multilinear polynomials, it is possible to use modern tensor methods. This opens the door to new approximation and complexity reduction methods for large scale systems with relevant nonlinear behavior. The modeling procedures including composition, decomposition, normalization, and multilinearization steps are shown by an example of a local energy system with a nonlinear electrolyzer, a linear buck converter and a PI controller with saturation.

Keywords: multilinear algebra, differential-algebraic models, tensor decompositions, energy systems

Zusammenfassung: Eine für Anwendungen interessante Unterklasse differential-algebraischer Modelle wird eingeführt. Die Beschränkung auf multilineare Polynome erlaubt den Einsatz moderner Tensormethoden. Dies öffnet die Tür zu neuen Approximations- und Reduktionsmethoden für großskalige Systeme mit relevantem nichtlinearen Verhalten. Die Modellierungsschritte Komposition, Dekomposition, Normalisierung sowie Multilinearisierung werden am Beispiel eines lokalen Energiesystems, bestehend aus einem nichtlinearen Elektrolyseur, einem linearen Buck-Converter und einem PI-Regler mit Sättigung, gezeigt.

Schlagwörter: Multilineare Algebra, differential-algebraische Modelle, Tensor-Dekomposition, Energiesysteme

Dedicated to the 70th birthday of Prof. Dr. Ing. Jan Lunze.

1 Introduction

1.1 Motivation

Automation systems in general are hybrid, i. e., some signals are discrete-valued and some are continuous-valued [20]. The discrete-valued part can be modeled by binary functions, which are generically in the multilinear subclass of polynomials. But multilinear functions can also be defined over real numbers, thus including linear functions, see Figure 1.

Linear modeling will not be sufficient for applications where larger deviations from an operating point have to be considered and which show essential nonlinear effects within their region of operation. Multilinear functions generically enable the usage of tensor algorithms and multilinear algebra is the mathematical domain of interest. In the last decade this domain has delivered results in the field of tensor decompositions which have been enabling break-throughs in many disciplines like quantum information theory or thermodynamic models of atomic alloy structures [5, 25]. The research community of control and automation is only slightly aware of this: an open invited session was organized at the IFAC World Congress 2017, but e. g., the number of papers at IFAC WC 2020 dealing with tensors methods still was low.

***Corresponding author: Gerwald Lichtenberg,** HAW Hamburg, Faculty Life Sciences, Hamburg, Germany, e-mail: gerwald.lichtenberg@haw-hamburg.de
Georg Pangalos, Aline Luxa, Niklas Jöres, Christoph Kaufmann, Fraunhofer IWES, Application Center Integration of Local Energy Systems (ILES), Hamburg, Germany, e-mails: georg.pangalos@iwes.fraunhofer.de, aline.luxa@iwes.fraunhofer.de, niklas.joeres@iwes.fraunhofer.de, christoph.kaufmann@iwes.fraunhofer.de
Carlos Cateriano Yáñez, Fraunhofer IWES, Application Center Integration of Local Energy Systems (ILES), Hamburg, Germany; and Universitat Politècnica de València, Instituto Universitario de Automática e Informática Industrial, València, Spain, e-mail: carlos.cateriano.yanez@iwes.fraunhofer.de
Leona Schnelle, HAW Hamburg, Faculty Life Sciences, Hamburg, Germany, e-mail: leona.schnelle@haw-hamburg.de

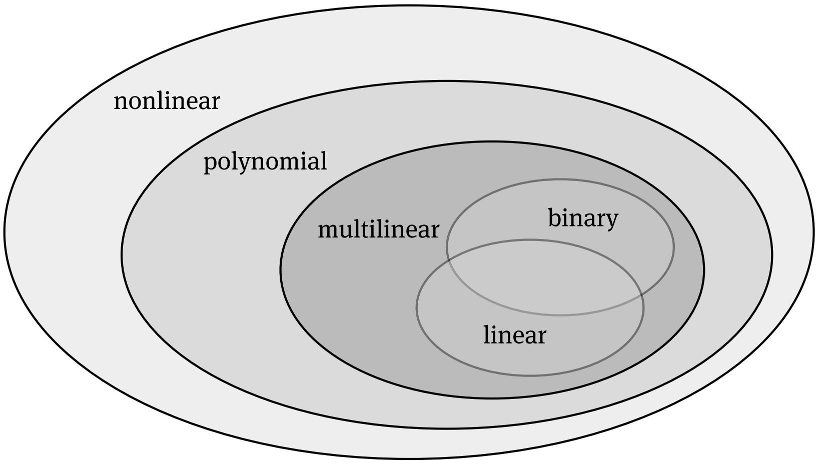


Figure 1: Classes of functions.

This paper follows the idea of using tensor decomposition techniques for parameter spaces of models. It further extends the class of multilinear time-invariant (MTI) models from explicit state space models to implicit descriptor models. This opens the door to reduced efficient modeling of huge nonlinear complex systems – where many challenging problems of today and in future are coming from.

1.2 Known results

Although tensor decompositions were defined a century ago [11] there are still open problems [3], some of them related to the field of algebraic geometry [6]. But applied mathematics and informatics research have found efficient strategies for approximative solutions [13]. The attention in engineering sciences has been growing with the availability of computational toolboxes [30, 2, 1, 24]. The connection of tensor decomposition and multilinear systems is established [27].

Remark: Multiple or piecewise linear systems should be called *multi-linear* to distinguish them from *multilinear* systems. The first class needs switches and linear algebra, the latter involves multilinear algebra. Tensor representations can be found for both, but this paper is focussed on the second.

In the past years, explicit multilinear time-invariant (eMTI) models have been developed and used for control of HVAC systems in buildings [14, 18, 19, 15]. The class of eMTI models is not closed w. r. t. basic compositions, but only the larger class of polynomial models [26]. Descriptor models which combine algebraic and differential equations are intrinsically composable. They describe the system dynamics implicitly and as well have been proven to be useful in many applications [22]. To the knowledge of the authors, extensions of eMTI models to implicit multilinear time-invariant (iMTI) models haven't been investigated so far. This will be the main and methodological contribution of this paper which will be illustrated by an example of an energy system.

Current and future problems of energy systems are the integration of new components in power networks – in the real world as well as in the models. Basic relevant literature for the energy system components discussed later can be found in [21]. The power system is undergoing a deep transformation towards power electronic dominated generation and consumption, which brings challenges regarding the safe operation of the grid. Therefore new techniques are needed to access the safe operation. The MTI class shows several aspects which makes it promising for power systems description, e. g., switching and discrete dynamics, multilinearity and scale.

1.3 Open questions

The non-closedness of eMTI models w. r. t. compositions has several drawbacks for complex system modeling which this paper overcomes by answering the questions:

1. How are basic connections (parallel, series, feedback) best represented?
2. Which model class holds the pros of multilinearity but offers composability?
3. Can complex energy system components be modeled within this class?
4. What complexity and accuracy do tensor energy network models have?

The paper is structured as follows: the next Section 2 recaps the current state of research of explicit multilinear modeling and introduces implicit models in matrix representation. Section 3 shows by an counterexample why explicit multilinear models are not closed w. r. t. composition of subsystems and gives the important result that implicit multilinear models are closed w. r. t. series, parallel and feedback compositions. In Section 4, tensor representations are given for explicit as well as implicit multilinear models together with one relevant decomposition method. How the canonical polyadic (CP) decomposed tensor representation can be further reduced by normalization is shown in Section 5. All methods are applied in Section 6 to a small scale energy system example for demonstration of the capabilities of the approach. The paper closes with conclusion and outlook.

2 Multilinear models

In this section the description of multilinear models is given. This class of models extends the linear class.

2.1 Multilinear functions

A function $f(\mathbf{x})$ with $\mathbf{x} = (x_1, \dots, x_n)$ is called multilinear, if the function is linear in each individual variable x_i, meaning all other variables are held constant. This means, that multiplicative combinations of the scalars in $\mathbf{x}$ are feasible for the function to be multilinear. All combinations of the scalars can be generated by the so called monomial vector $\mathbf{m}(\mathbf{x})$, which is given by

$$\mathbf{m}(\mathbf{x}) = \begin{pmatrix} 1 \\ x_n \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} 1 \\ x_1 \end{pmatrix}, \tag{1}$$

where $\otimes$ denotes the Kronecker product.

Example 2.1. For $x \in \mathbb{R}^3$ the monomial is given by

$$\mathbf{m}((x_1, x_2, x_3)) = \begin{pmatrix} 1 \\ x_1 \\ x_2 \\ x_1x_2 \\ x_3 \\ x_1x_3 \\ x_2x_3 \\ x_1x_2x_3 \end{pmatrix}. \tag{2}$$

Thus, the general form of a multilinear function can be given by the inner product

$$f(\mathbf{x}) = \mathbf{f}^T \mathbf{m}(\mathbf{x}), \tag{3}$$

i. e., the multiplication of a row vector $\mathbf{f}^T$ of coefficients times the column vector $\mathbf{m}(\mathbf{x})$ of monomials.

A very interesting property of multilinear functions is, that all Boolean functions belong to that class. In the following, the set denoted by $\mathbb{B} = \{0, 1\} \equiv \{\mathtt{FALSE}, \mathtt{TRUE}\}$ refers to binary variables. Any Boolean function b of n variables can be represented by 2^n rows of a truth table, corresponding to n-element binary vectors given in lexicographical order. Assuming this order, the Boolean function can be represented by the truth vector $\mathbf{b} \in \mathbb{B}^{2^n}$ of the last column of the truth table. To show this, a vector of literals is introduced as the Kronecker product of the literals of the variables of the Boolean function, given as

$$\mathbf{l}(\mathbf{x}) = \begin{pmatrix} \bar{x}_n \\ x_n \end{pmatrix} \otimes \cdots \otimes \begin{pmatrix} \bar{x}_1 \\ x_1 \end{pmatrix} \in \mathbb{R}^{2^n}, \tag{4}$$

where $\bar{x} := 1 - x \in \mathbb{R}$.

Definition 2.1. A Zhegalkin polynomial [32]

$$f(\mathbf{x}) = \mathbf{b}^T \mathbf{l}(\mathbf{x}) \tag{5}$$

of order n is a multilinear square-free polynomial given by the scalar product of a Boolean truth vector $\mathbf{b} \in \mathbb{B}^{2^n}$ and a vector $\mathbf{l}(\mathbf{x}) \in \mathbb{R}^{2^n}$ of literals. Zhegalkin Polynomials are sometimes also referred to as *Reed-Muller Forms* or *Algebraic Normal Forms* (ANF).

The following example of order $n = 2$, taken from [28], illustrates the idea.

Example 2.2. For the Boolean function $b = \neg(x_1 \wedge x_2)$ the truth table is given as

x_2	x_1	b
0	0	1
0	1	1
1	0	1
1	1	0

thus the Zhegalkin Polynomial is

$$\begin{aligned} f(\mathbf{x}) = \mathbf{b}^T \mathbf{l}(\mathbf{x}) &= \begin{pmatrix} 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} \bar{x}_2\bar{x}_1 \\ \bar{x}_2x_1 \\ x_2\bar{x}_1 \\ x_2x_1 \end{pmatrix} \\ &= (1 - x_1)(1 - x_2) + x_1(1 - x_2) + (1 - x_1)x_2 + 0x_2x_1 \\ &= 1 - x_1x_2 = \begin{pmatrix} 1 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ x_1 \\ x_2 \\ x_1x_2 \end{pmatrix} = \mathbf{f}^T \mathbf{m}(\mathbf{x}). \end{aligned}$$

As Zhegalkin polynomials (5) are multilinear, they can be equivalently represented by the parameter vector $\mathbf{f}$ of (3).

If Boolean values are inserted for x_1 and x_2 this results in Boolean values which correspond to the truth table.

2.2 Explicit multilinear models

With the monomial vector, an eMTI state space model can be written in continuous time

$$\dot{\mathbf{x}} = \mathbf{F}\,\mathbf{m}(\mathbf{x}, \mathbf{u}) \,, \tag{6}$$

$$\mathbf{y} = \mathbf{G}\,\mathbf{m}(\mathbf{x}, \mathbf{u}) \,, \tag{7}$$

where $\mathbf{u} \in \mathbb{R}^m$ is the input vector, $\mathbf{y} \in \mathbb{R}^p$ the output vector, $\mathbf{F} \in \mathbb{R}^{n\times 2^{(n+m)}}$ the transition matrix and $\mathbf{G} \in \mathbb{R}^{p\times 2^{(n+m)}}$ the output matrix. The monomial vector of more than one vector treats the input vectors as concatenated, for two vectors i. e., $\mathbf{m}(\mathbf{x}, \mathbf{u}) = \mathbf{m}((x_1, x_2, u_1, u_2))$.

Remark. In contrast to multilinear models, linear models are restricted to the subclass of linear functions going through the origin ($f(0) = 0$), whereas the first element of the monomial vector contains a constant.

Remark. Bilinear models are also a subclass of multilinear models where all monomials are restricted to order 2 and combinations of inputs and states.

Boolean functions can be represented as given in (5). If the Boolean variable is depended on a continuous variable x, the result of (5) will be continuous. Thus, continuous multilinear function and Boolean functions can be modeled individually by (6) and (7). On the other side, if continuous and Boolean functions are connected with each other a quantizer is needed, see [27] for more details. A simple solution is to use a Heaviside function

$$\sigma(x) = \begin{cases} 1 & \text{if } x \geq 0 \,, \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

as quantizer. Using this function, a saturation can be modeled as given in the following example.

Example 2.3. Assuming the input to the saturation is a state, and the output is saturated between two values, e. g., the output of a controller needs to be between 0 and 1. The saturation function is depicted in Figure 2.

To model this as an eMTI model two discrete-valued variables are introduced

$$b_l = \sigma(u - u_l) \tag{9}$$

$$b_u = \sigma(u - u_u) \,. \tag{10}$$

The output is then calculated by

$$y = (1 - b_l)y_l + b_l(1 - b_u)((u - u_l)s + y_l) + b_u y_u \,, \tag{11}$$

with the slope $s = \frac{y_u - y_l}{u_u - u_l}$ of the middle part.

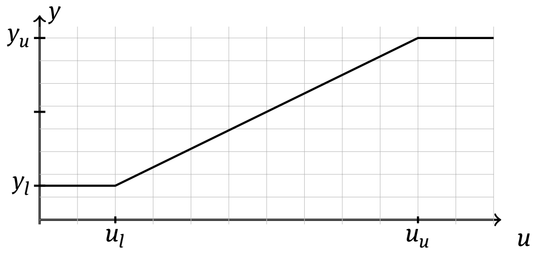


Figure 2: Saturation function.

2.3 Implicit multilinear models

The class of explicit multilinear models has the drawback, that by connecting multiple eMTI models the overall model is not necessarily an eMTI model which will be shown in section 3. To overcome this obstacle we introduce iMTI models similar to descriptor systems in the next subsection.

For iMTI models a kernel representation is chosen and the monomial vector is extended. An iMTI model in kernel representation is given as

$$\mathbf{H}\,\mathbf{m}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, \mathbf{y}) = \mathbf{0} \,, \tag{12}$$

where the states are $\dot{\mathbf{x}} \in \mathbb{R}^n$, $\mathbf{x} \in \mathbb{R}^n$, the inputs are $\mathbf{u} \in \mathbb{R}^m$, the outputs are $\mathbf{y} \in \mathbb{R}^p$ and the model matrix is $\mathbf{H} \in \mathbb{R}^{N\times 2^{2n+m+p}}$. The dimension of the model matrix is not only determined by the number of states, inputs and outputs but by the application specific number N of equations. A designated output equation is in implicit form not required, but output equations can be included in the model matrix $\mathbf{H}$.

Remark. Note that all eMTI models can be written as iMTI models by subtracting the right hand sides of (6) and (7) on both sides, leading to kernel representations

$$\dot{\mathbf{x}} - \mathbf{F}\,\mathbf{m}(\mathbf{x}, \mathbf{u}) = \mathbf{0} \tag{13}$$

$$\mathbf{y} - \mathbf{G}\,\mathbf{m}(\mathbf{x}, \mathbf{u}) = \mathbf{0} \,, \tag{14}$$

given as a semiexplicit differential algebraic equation, [4]. These equations can then be written in the form (12), because their left hand sides are obviously multilinear functions in $\mathbf{m}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, \mathbf{y})$.

Since the equations for the states are no longer in explicit form, the mapping from continuous-valued variables to discrete-valued variables can no longer be performed by quantization (8), but discrete valued states have to be defined in pairs, because the single kernel equation $b(1-b) = b - b^2 = 0$ of a Boolean constraint is polynomial, but not multilinear.

Example 2.4. Assume the states b_1 and b_2 are Boolean and can therefore only take the values 0 or 1, meaning FALSE or TRUE, respectively. Introducing the equations

$$b_1 b_2 = 0\ , \tag{15}$$

$$b_1 + b_2 - 1 = 0 \tag{16}$$

forces at least one of the variables to be 0 by (15), but prevents both to be 0, because of (16), and thus forcing one to be 0 and the other one to be 1.

As the example shows, an iMTI model is capable to define Boolean variables. But to, e. g., model saturation, inequalities are no longer avoidable – which leads to the generic multilinear model

$$\mathbf{L}\,\mathbf{m}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, \mathbf{y}) \geq \mathbf{0}\ . \tag{17}$$

Note that (17) includes the kernel representation (12) by setting

$$\mathbf{L} = \begin{pmatrix} \mathbf{H} \\ -\mathbf{H} \end{pmatrix} \in \mathbb{R}^{2N\times(2n+m+p)}\ , \tag{18}$$

and thus, the iMTI model (17) has the broadest spectrum of applicability.

Next, saturation is modeled generically as iMTI.

Example 2.5. To model saturation, we use the discrete variables b_1 and b_2 of Example 2.4 and append a second pair of discrete variables by

$$b_3 b_4 = 0\ , \tag{19}$$

$$b_3 + b_4 - 1 = 0\ . \tag{20}$$

Then the variables need to be connected to the ranges in Figure 2 given as

$$(b_1 - b_2)(u - u_l) \geq 0\ , \tag{21}$$

$$(b_3 - b_4)(u - u_u) \geq 0\ . \tag{22}$$

In case u is above the lower limit u_l, the second term in (21) is positive, thus $b_1 = 1$ and $b_2 = 0$ for the inequality to hold. If u is below the lower limit u_l, then $b_1 = 0$ and $b_2 = 1$. The same holds analogously for the upper limit in (22), see Table 1.

Table 1: Values of the discrete variables corresponding to the input ranges.

$u - u_l$	$u - u_u$	b_1	b_2	b_3	b_4
< 0	< 0	0	1	0	1
> 0	< 0	1	0	0	1
> 0	> 0	1	0	1	0

With the discrete variables connected to the input ranges, it is now possible to give the output equation of the saturation as

$$y = b_2 y_l + b_1 b_4((u - u_l)s + y_l) + b_3 y_u\ , \tag{23}$$

again with the slope $s = \frac{y_u - y_l}{u_u - u_l}$.

3 Composition

Systems can be connected in three different ways: series, parallel, and feedback. In general, any eMTI system can be connected to another eMTI system using one of the mentioned connection types. Whether the coupled system is within the class of eMTI systems depends on the subsystems and the connection type [26]. The following example shows a simple series connection where the overall system does not remain within the eMTI class.

Example 3.1. Consider two first order eMTI systems connected in series. The first system has one input u_1 and two outputs $\mathbf{y}_1$ described by

$$\dot{x}_1 = f_{12} x_1 + f_{14}\, x_1\, u_1\ , \tag{24}$$

$$\mathbf{y}_1 = \begin{pmatrix} g_{12} x_1 \\ g_{18}\, x_1\, u_1 \end{pmatrix}\ . \tag{25}$$

The second system has a 2-dimensional input $\mathbf{u}_2$ and one output y_2 described by

$$\dot{x}_2 = f_{21} + f_{27}\, u_{21}\, u_{22}\ , \tag{26}$$

$$y_2 = g_{26}\, u_{22}\, x_2\ . \tag{27}$$

A series connection of the two subsystems can be represented as state space model

$$\dot{\mathbf{x}} = \begin{pmatrix} f_{12}\, x_1 + f_{14}\, x_1\, u_1 \\ f_{21} + f_{27}\, g_{12}\, g_{18}\, x_1^2\, u_1 \end{pmatrix}\ , \tag{28}$$

$$y = g_{26}\, g_{18}\, x_1\, x_2\, u_1\ . \tag{29}$$

Due to the term x_1^2, it is no longer possible to model the overall system explicitly multilinear, but only by more general polynomial functions. See Appendix B for a detailed development and the parameter matrices **F** and **G**.

In contrast, if the systems are represented in iMTI form, the overall system will still be within this class, which is formally stated by the following proposition.

Lemma 3.1. *The class of iMTI models (12) is closed w. r. t. all connections: series, parallel, and feedback.*

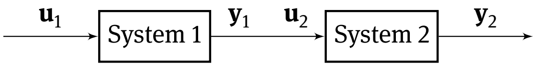


Figure 3: Series connection.

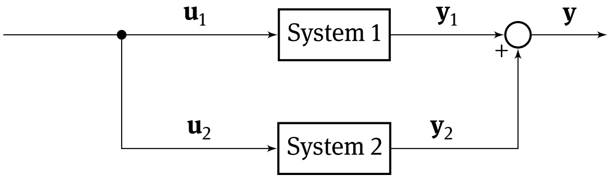


Figure 4: Parallel connection.

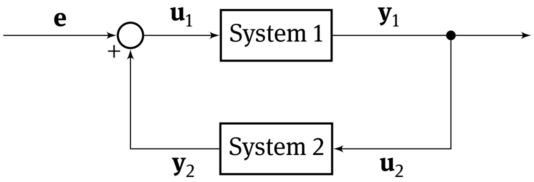


Figure 5: Feedback connection.

Proof. Given two model matrices $\mathbf{H}_1$ and $\mathbf{H}_2$ of the subsystems, the model matrix $\mathbf{H}$ of the overall system has a column dimension according to the total number of variables of the overall system. The row dimension takes the sum of the row dimensions of $\mathbf{H}_1$ and $\mathbf{H}_2$ plus the number of additional connecting equations, which is dependent on the connection type. The overall dimensions of $\mathbf{H}$ are given in subsection 2.3. The equations to append for different connection types are given next.

1. Series connection (Figure 3) has the implicit equation

$$\mathbf{u}_2 - \mathbf{y}_1 = \mathbf{0}\,. \tag{30}$$

2. Parallel connection (Figure 4) has the implicit equations

$$\mathbf{y} - \mathbf{y}_1 - \mathbf{y}_2 = \mathbf{0}\,, \tag{31}$$

$$\mathbf{u}_2 - \mathbf{u}_1 = \mathbf{0}\,. \tag{32}$$

3. Feedback connection (Figure 5) has the implicit equations

$$\mathbf{u}_2 - \mathbf{y}_1 = \mathbf{0}\,, \tag{33}$$

$$\mathbf{e} + \mathbf{y}_2 - \mathbf{u}_1 = \mathbf{0}\,. \tag{34}$$

□

Example 3.2. If Example 3.1 is written in implicit form, it can be seen that the overall system remains in the iMTI class. The iMTI formulation of system 1 is

$$\begin{pmatrix} f_{12}\,x_1 + f_{14}\,x_1\,u_1 - \dot{x}_1 \\ g_{12}\,x_1 - y_{11} \\ g_{18}\,x_1\,u - y_{12} \end{pmatrix} = \mathbf{0}\,, \tag{35}$$

and for system 2

$$\begin{pmatrix} f_{21} + f_{27}\,u_{21}\,u_{22} - \dot{x}_2 \\ g_{26}\,u_{22}\,x_2 - y \end{pmatrix} = \mathbf{0}\,. \tag{36}$$

Using (30) for series connection, the implicit form of the overall model is

$$\begin{pmatrix} f_{12}\,x_1 + f_{14}\,x_1\,u_1 - \dot{x}_1 \\ g_{12}\,x_1 - y_{11} \\ g_{18}\,x_1\,u_1 - y_{12} \\ f_{21} + f_{27}\,u_{21}\,u_{22} - \dot{x}_2 \\ g_{26}\,u_{22}\,x_2 - y \\ u_{21} - y_{11} \\ u_{22} - y_{12} \end{pmatrix} = \mathbf{0}\,. \tag{37}$$

It is clear that the last two connection equations allow to reduce the model to

$$\begin{pmatrix} f_{12}\,x_1 + f_{14}\,x_1\,u_1 - \dot{x}_1 \\ g_{12}\,x_1 - u_{21} \\ g_{18}\,x_1\,u_1 - u_{22} \\ f_{21} + f_{27}\,u_{21}\,u_{22} - \dot{x}_2 \\ g_{26}\,u_{22}\,x_2 - y \end{pmatrix} = \mathbf{0}\,. \tag{38}$$

See Appendix B for a more detailed development of the equations.

Equation (38) shows that in general, auxiliary variables are required in addition to the overall input and output variables and state variables of the subsystems. Symbolic methods for reduction of the number of auxiliary variables and thus, equations are known and provided by standard algorithms, e. g., as implemented for the *Modelica* language in *OpenModelica*. This paper does not focus on these, but on the structure of the iMTI models, which additional enable the use of tensor decomposition methods for reduction which will be discussed in the next section.

4 Decomposition

4.1 Tensors

Real-valued tensors $\mathsf{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$ are multidimensional arrays well known from physics and various other application domains, e. g., in data sciences. Well established

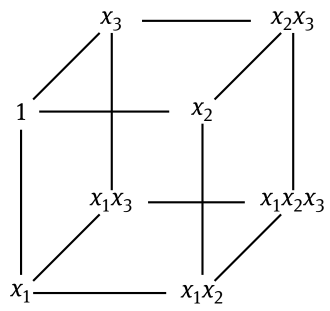


Figure 6: Monomial tensor of 3rd order.

linear control engineering methods mostly rely on matrices, i. e., 2D tensors and also current extensions to the nonlinear domain, e. g., linear-parameter-varying models do this, too. But for multilinear models, matrices are less appropriate parameter formats than general tensors. This is because of the intrinsic tensor structure of multilinear monomials which can be seen best by the small example in Figure 6.

The contracted product $\langle\cdot|\cdot\rangle$ of a tensor $\mathsf{F} \in \mathbb{R}^{I_1\times\dots\times I_n\times J_1\dots\times J_m}$ and a tensor $\mathsf{G} \in \mathbb{R}^{I_1\times\dots\times I_n}$ is defined as a tensor $\mathsf{H} = \langle\, \mathsf{F} \mid \mathsf{G}\, \rangle \in \mathbb{R}^{J_1\times\dots\times J_m}$ with elements

$$\begin{aligned} &H(j_1,\dots,j_m) \\ &\quad = \sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}\cdots\sum_{i_n=1}^{I_n} g(i_1,\dots,i_n)f(i_1,\dots,i_n,j_1,\dots,j_m)\,. \end{aligned} \tag{39}$$

The contracted product for tensors of the same size is called inner (or element-wise) product. This can be used to define a multilinear function

$$f(\mathbf{x}) = \langle\, \mathsf{F} \mid \mathsf{M}(\mathbf{x})\, \rangle \in \mathbb{R}\,, \tag{40}$$

with $\mathbf{x} \in \mathbb{R}^n$ represented by the inner (elementwise) product of a parameter tensor $\mathsf{F} \in \mathbb{R}^{\times^n 2}$, using the simple notation for tensor spaces as

$$\mathbb{R}^{\times^{(n+m)}2} := \mathbb{R}^{\times^{(n+m)}2} := \mathbb{R}^{\overbrace{2\times\ldots\times 2}^{n+m}}\,,$$

with the monomial tensor $\mathsf{M}(\mathbf{x})$ of the same dimension, which is the tensor analogon to (3), [13].

4.2 Tensor decomposition

The representation of the coefficients of a multilinear polynomial as tensor F as in (40) enables the application of tensor decomposition methods which have proven to be extremely powerful in many other application domains. This paper focusses on the so called canonical polyadic (CP) decomposition, but all other decomposition methods are also possible in principle [14].

A tensor X can be decomposed in a sum of a minimum number of r outer products, where r is the so called *tensor rank*. All factors can be represented as matrices $\mathbf{F}_i \in \mathbb{R}^{I_i\times r}$ for each dimension i of the original tensor F, which can be abbreviated as

$$\mathsf{X} = [\mathbf{F}_1, \mathbf{F}_2, \cdots, \mathbf{F}_n]\ . \tag{41}$$

An element of X is then given by

$$x(i_1, i_2, \dots, i_n) = \sum_{j=1}^{r} \mathbf{F}_1(i_1, j)\cdot\mathbf{F}_2(i_2, j)\cdots\mathbf{F}_n(i_n, j)\,, \tag{42}$$

which can be seen as the generalization of the dyadic product for dimensions larger 2. Figure 7 shows how the red element $x(3,1,2)$ of the tensor is computed as $F_1(3,1)F_2(1,1)F_3(2,1) + F_1(3,2)F_2(1,2)F_3(2,2) + \cdots + F_1(3,r)F_2(1,r)F_3(2,r)$ by $r\cdot(n-1)$ multiplications and $(r-1)$ summations.

Determing the exact rank of certain tensors is known to be a relevant but hard problem in mathematics [7]. For engineering applications, the problem of finding a CP decomposition of a predefined rank as approximation of an original tensor F is more relevant. Minimizing the norm $\langle\, \mathsf{E} \mid \mathsf{E}\, \rangle$ of the error tensor

$$\mathsf{E} = \mathsf{F} - [\mathbf{F}_1, \cdots, \mathbf{F}_n] \tag{43}$$

still is a non-convex hard problem, but sub-optimal solutions can be computed by tools like *Tensorlab* [30] and used for approximation in applications like the one given in this paper in Section 6.

4.3 Explicit CP models

Any eMTI model can represented by contracted tensor products [26]

$$\dot{\mathbf{x}} = \langle\, \mathsf{F} \mid \mathsf{M}(\mathbf{x},\mathbf{u})\, \rangle\ , \tag{44}$$

$$\mathbf{y} = \langle\, \mathsf{G} \mid \mathsf{M}(\mathbf{x},\mathbf{u})\, \rangle\ , \tag{45}$$

with the parameter tensors $\mathsf{F} = \mathbb{R}^{\times^{(n+m)}2\times n}$ and $\mathsf{G} = \mathbb{R}^{\times^{(n+m)}2\times p}$. Whereas the parameter tensors can have arbitrary rank, monomial tensors are always rank 1, so for a state vector $\mathbf{x} \in \mathbb{R}^n$ and an input vector $\mathbf{u} \in \mathbb{R}^m$ the mono-

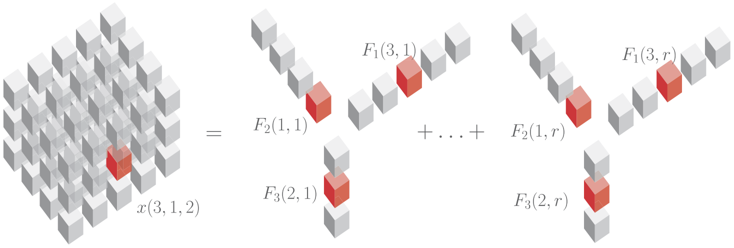


Figure 7: CP Decomposition of a tensor with dimensions 5 × 3 × 4.

mial tensor has the decomposition

$$\mathsf{M}(\mathbf{x},\mathbf{u}) = \left[\left[\begin{pmatrix}1\\x_1\end{pmatrix},\cdots,\begin{pmatrix}1\\x_n\end{pmatrix},\begin{pmatrix}1\\u_1\end{pmatrix},\cdots,\begin{pmatrix}1\\u_m\end{pmatrix}\right]\right] . \quad (46)$$

After CP decomposition of the parameter tensors as well as the monomial tensor, the eMTI model reads

$$\dot{\mathbf{x}} = \left\langle \left[\mathbf{F}_1,\cdots,\mathbf{F}_{n+m},\mathbf{F}_\phi\right] \middle| \left[\left[\begin{pmatrix}1\\x_1\end{pmatrix},\cdots,\begin{pmatrix}1\\u_m\end{pmatrix}\right]\right] \right\rangle , \quad (47)$$

$$\mathbf{y} = \left\langle \left[\mathbf{G}_1,\cdots,\mathbf{G}_{n+m},\mathbf{G}_\phi\right] \middle| \left[\left[\begin{pmatrix}1\\x_1\end{pmatrix},\cdots,\begin{pmatrix}1\\u_m\end{pmatrix}\right]\right] \right\rangle , \quad (48)$$

with the factor matrices $\mathbf{F}_i, \mathbf{G}_i \in \mathbb{R}^{2\times r}$ for all states and inputs $i = 1,\cdots,(n+m)$ and $\mathbf{F}_\phi \in \mathbb{R}^{n\times r}$ and $\mathbf{G}_\phi \in \mathbb{R}^{p\times r}$.

The next step is the most important one for applicability and large-scale models, because here the curse of dimensionality is broken by factorization. If a CP decomposition of the parameter tensor F with arbitrary rank r is available, the computation of the right hand side of the state space model (6)–(7) is possible by using the factors and the rank-1 structure of the monomial tensor, e. g., for the state equation by

$$\dot{\mathbf{x}} = \mathbf{F}_\phi\left(\left(\mathbf{F}_1^{\mathrm{T}}\begin{pmatrix}1\\x_1\end{pmatrix}\right)\circledast\cdots\circledast\left(\mathbf{F}_n^{\mathrm{T}}\begin{pmatrix}1\\x_n\end{pmatrix}\right)\right.$$
$$\left.\circledast\left(\mathbf{F}_{n+1}^{\mathrm{T}}\begin{pmatrix}1\\u_1\end{pmatrix}\right)\circledast\cdots\circledast\left(\mathbf{F}_{m+n}^{\mathrm{T}}\begin{pmatrix}1\\u_m\end{pmatrix}\right)\right). \quad (49)$$

The overall number of operations for each factor are $2r$ multiplications and $n+m$ additions, together with $n+m$ multiplications for the element-wise Hadamard product $\circledast$ and the final multiplication with the $n\times r$ matrix $\mathbf{F}_\phi$ these are $(n+2)\,r$ multiplications and $r+n+m$ additions which is linear in all dimensions n and m of the variables whereas already the number 2^{n+m} of monomials has exponential complexity. Thus, full tensor computations are reasonable only for small dynamic order n and number m of inputs, whereas computations with decomposed representations are easily done up to very large dimensions. Similar arguments hold for the output equation (48).

4.4 Implicit CP models

An iMTI model (12) represented in tensor form reads

$$\langle\, \mathsf{H} \mid \mathsf{M}(\dot{\mathbf{x}},\mathbf{x},\mathbf{u},\mathbf{y})\,\rangle = \mathbf{0} , \quad (50)$$

and the same holds true for the representation of the inequality constraints

$$\langle\, \mathsf{L} \mid \mathsf{M}(\dot{\mathbf{x}},\mathbf{x},\mathbf{u},\mathbf{y})\,\rangle \geq \mathbf{0} . \quad (51)$$

As discussed, for larger systems only decomposed parameter tensors make sense, i. e., the general model is given by

$$\left\langle \left[\mathbf{L}_1,\cdots,\mathbf{L}_{2n+m+p},\mathbf{L}_\phi\right] \middle| \left[\left[\begin{pmatrix}1\\\dot{x}_1\end{pmatrix},\cdots,\begin{pmatrix}1\\y_p\end{pmatrix}\right]\right] \right\rangle \geq \mathbf{0} , \quad (52)$$

as mentioned in subsection 2.3, (50) can be included in (51) and therefore the decomposed version of (50) is not given. Computations with large scale models always have to be done on the factors like

$$\mathbf{L}_\phi\left(\left(\mathbf{L}_1^{\mathrm{T}}\begin{pmatrix}1\\\dot{x}_1\end{pmatrix}\right)\circledast\cdots\circledast\left(\mathbf{L}_{2n+m+p}^{\mathrm{T}}\begin{pmatrix}1\\y_p\end{pmatrix}\right)\right) \geq \mathbf{0} , \quad (53)$$

because even constructing the full parameter tensor would lead to enormous memory and runtime requirements.

The factor matrix $\mathbf{L}_\phi \in \mathbb{R}^{N\times r}$ where the row dimension N is the number of inequality constraints and r is the decomposed tensor rank. All other factor matrices $\mathbf{L}_i \in \mathbb{R}^{2\times r}$ for $i = 1,\cdots 2n+m+p$ have a row dimension of 2. The next Section will show how this structure can be exploited further and how the storage needed for the model can nearly be reduced by another 50 %.

5 Normalization

In this section, a linear state and input transformation for eMTI models is given and a normalization of iMTI models in CP tensor representation are discussed.

5.1 Linear transformation

For linear state and input variable transformations of multilinear models (6) in continuous time given by

$$\widetilde{x}_i = \frac{x_i - b_i}{a_i} \; \forall i = 1, \dots, n, \tag{54}$$

$$\widetilde{u}_i = \frac{u_i - b_{n+i}}{a_{n+i}} \; \forall i = 1, \dots, m, \tag{55}$$

the transformed state transition matrix is given by

$$\widetilde{\mathbf{F}} = \operatorname*{diag}_{i=1,\dots,n} \left(\frac{1}{a_i}\right)(\mathbf{FT}), \tag{56}$$

where diag denotes the diagonal matrix with elements a_i and

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

is a transformation matrix. Linear transformations allow a numerical preconditioning of models which is used in the application example. The linear transformation can as well be done in tensor representation.

5.2 Normalized implicit CP tensor models

The CP tensor H of the decomposed iMTI model from section 4 can be represented in a normalized form. Several norms, such as the euclidean norm (2-norm) or the manhattan norm (1-norm) could be used in principle, but the latter enables very efficient computations and thus is given priority in this paper [17]. In [13] it is proposed to normalize the columns of the factors of a CP tensor which is applied in [12] using the norm 1 condition.

Definition 5.1. The iMTI model with CP Tensor

$$\widetilde{\mathsf{H}} = [\widetilde{\mathbf{H}}_1, \dots, \widetilde{\mathbf{H}}_{2n+m+p}, \widetilde{\mathbf{H}}_\phi] \tag{57}$$

is called 1-norm normalized, if all factors $\widetilde{\mathbf{H}}_i \in \mathbb{R}^{2\times r}$ for $i = 1, \dots, (2n+m+p)$ have columns $k = 1, \dots, r$ of norm

$$||\widetilde{\mathbf{H}}_i(:,k)||_1 = \sum_{j=1}^{2} |\widetilde{\mathbf{H}}_i(j,k)| = |\widetilde{\mathbf{H}}_i(1,k)| + |\widetilde{\mathbf{H}}_i(2,k)| = 1\,. \tag{58}$$

A minimal representation of $\widetilde{\mathsf{H}}$ by the factor $\widetilde{\mathbf{H}}_\phi \in \mathbb{R}^{N\times r}$ and only vectors instead of matrices for the other factors thus is possible, because the other elements of the matrix $\widetilde{\mathbf{H}}_i$ can be computed by (58). Moreover, any CP decomposed tensor H of an iMTI model can be transformed in a minimal normalized representation with factor vectors

$$\widetilde{\mathbf{h}}_i(k) = \frac{\mathbf{H}_i(2,k)}{||\mathbf{H}_i(:,k)||_1} \cdot (2\sigma(\mathbf{H}_i(1,k)) - 1) \tag{59}$$

and

$$\widetilde{\mathbf{H}}_\phi = \mathbf{H}_\phi \circledast \left[\prod_{i=1}^{2n+m+p} ||\mathbf{H}_i(:,1)||_1, \dots, \prod_{i=1}^{2n+m+p} ||\mathbf{H}_i(:,r)||_1 \right]$$
$$\circledast \left[\prod_{i=1}^{2n+m+p} (2\sigma(\mathbf{H}_i(1,1)) - 1), \dots, \prod_{i=1}^{2n+m+p} (2\sigma(\mathbf{H}_i(1,r)) - 1) \right]. \tag{60}$$

Some remarks:

- For computational efficiency, $\widetilde{\mathbf{H}}_\phi$ is allowed to have arbitrary column vectors,
- for all reasonable iMTI models, $||\mathbf{H}_i(:,k)||_1 > 0$ holds,
- the Heaviside function (8) allows to construct a two-valued sign function,
- all factor vectors $\widetilde{\mathbf{h}}_i$ can be stored in a matrix of dimension $\mathbb{R}^{(2n+m+p)\times r}$.

In the following, an example of CP tensor normalization is given by an example of an easy linear system relevant for the next section, too.

Example 5.1. A PI controller can be represented by the implicit equations

$$\begin{aligned} \dot{x} - K_I\, u &= 0, \\ y - x - K_P\, u &= 0, \end{aligned} \tag{61}$$

with integral gain K_I, proportional gain K_P, input u, state x and output y. The CP iMTI model

$$\left\langle [\mathbf{H}_{\dot{x}}, \mathbf{H}_x, \mathbf{H}_u, \mathbf{H}_y, \mathbf{H}_\phi] \middle| \left[\begin{pmatrix} 1 \\ \dot{x} \end{pmatrix}, \begin{pmatrix} 1 \\ x \end{pmatrix}, \begin{pmatrix} 1 \\ u \end{pmatrix}, \begin{pmatrix} 1 \\ y \end{pmatrix} \right] \right\rangle = \mathbf{0} \tag{62}$$

can be represented as a rank 5 CP tensor H with factor matrices for each term

$$\mathbf{H}_{\dot{x}} = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\mathbf{H}_x = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 \\ 0 & -1 & 0 & 0 & 0 \end{pmatrix},$$

$$\mathbf{H}_u = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & -K_I & -K_P & 0 \end{pmatrix},$$

$$\mathbf{H}_y = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$\mathbf{H}_\phi = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

Using (59) and (60), the factors in minimal normalized form are

$$\begin{aligned}
\widetilde{\mathbf{h}}_{\dot{x}} &= \begin{pmatrix}1 & 0 & 0 & 0 & 0\end{pmatrix},\\
\widetilde{\mathbf{h}}_{x} &= \begin{pmatrix}0 & -1 & 0 & 0 & 0\end{pmatrix},\\
\widetilde{\mathbf{h}}_{u} &= \begin{pmatrix}0 & 0 & -1 & -1 & 0\end{pmatrix},\\
\widetilde{\mathbf{h}}_{y} &= \begin{pmatrix}0 & 0 & 0 & 0 & 1\end{pmatrix},\\
\widetilde{\mathbf{H}}_{\phi} &= \begin{pmatrix}1 & 0 & K_I & 0 & 0\\ 0 & 1 & 0 & K_P & 1\end{pmatrix}.
\end{aligned}$$

Because the third and the fourth columns of all $\mathbf{h}_i$ are equal and $\mathbf{H}_\phi$ has only one parameter in the third row for the first state equation and one parameter in the forth row for the second state equation, it is obvious, that the rank can be reduced to 4 by storing the third and fourth columns as one:

$$\begin{aligned}
\widetilde{\mathbf{h}}_{\dot{x}} &= \begin{pmatrix}1 & 0 & 0 & 0\end{pmatrix},\\
\widetilde{\mathbf{h}}_{x} &= \begin{pmatrix}0 & -1 & 0 & 0\end{pmatrix},\\
\widetilde{\mathbf{h}}_{u} &= \begin{pmatrix}0 & 0 & -1 & 0\end{pmatrix},\\
\widetilde{\mathbf{h}}_{y} &= \begin{pmatrix}0 & 0 & 0 & 1\end{pmatrix},\\
\widetilde{\mathbf{H}}_{\phi} &= \begin{pmatrix}1 & 0 & K_I & 0\\ 0 & 1 & K_P & 1\end{pmatrix}.
\end{aligned}$$

It is now possible to crosscheck, that by (53) the model (61) can be recovered

$$\begin{aligned}
((1-1)\cdot 1 + 1\dot{x})\cdot 1 + ((1-1)\cdot 1 - 1u)\cdot K_I &= 0\,,\\
((1-1)\cdot 1 - 1x)\cdot 1 + ((1-1)\cdot 1 - 1u)\cdot K_P&\\
+((1-1)\cdot 1 + 1y)\cdot 1 &= 0\,.
\end{aligned}$$

To illustrate the normalization of the factor matrices from Example 5.1, the Figure 8 shows the original column vector of $\mathbf{H}_u(:,3)$ with dashed line and the normalized column vector of $\widetilde{\mathbf{H}}_u(:,3)$ with solid line in blue. Red vectors belong to a factor $\mathbf{H}_u(:,k) = \begin{pmatrix}2 & 1\end{pmatrix}'$ to demonstrate normalization with arbitrary angle.

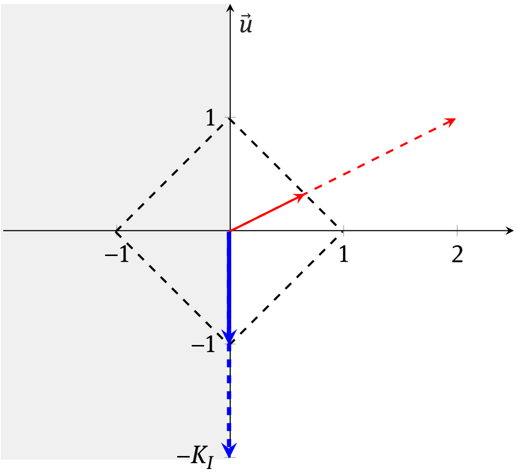


Figure 8: Normalization of a CP tensor factor to norm-1.

6 Application example

The previous section showed that the non-closedness of the eMTI is overcome by introducing the iMTI form. The following section presents a first showcase example about the usage of this new representation by connecting an electrolyzer with a buck converter powered by an infinite DC bus. These applications exemplify local energy systems, where several multiphysical dynamics are present. Three main aspects are demonstrated here. Firstly, the nonlinear model of the electrolyzer is multilinearized and compared with a typical linear approximation. Then this multiphysical model is connected with a buck converter, where a linear averaged model is assumed, since the focus in this paper is on the connection of the two models. Lastly, this paper derives a representation of the nonlinear saturation limits of the buck converter in an iMTI form. Thereby, the basic functionalities of the new implicit form are demonstrated and the prospective of large-scale multilinear energy systems modeling is shown.

6.1 Electrolyzer

The nonlinear model of the polymer exchange membrane (PEM) electrolyzer is based on a 46 kW model from literature [8]. Because in this example the connection to power electronics is crucial, also the electric dynamic behavior should be modeled [23]. Considering an equivalent electric circuit to model the electrolyzer with an RC element for each electrode, the ordinary differential equation (ODE) describing the activation overpotential v_{act} on anode and cathode, respectively, is expressed as

$$\frac{\mathrm{d}v_{\mathrm{act}}}{\mathrm{d}t} = \frac{1}{C_{\mathrm{DL}}} i_{\mathrm{ely}} - \frac{1}{C_{\mathrm{DL}} R_{\mathrm{act}}} v_{\mathrm{act}}, \qquad (63)$$

with the total electrolyzer current i_{ely} in A [31]. The double layer capacity C_{DL}, defined in (A.1), and activation resistance R_{act} are often measured by a step function of the input current or voltage [10]. Since the base model of [8] does not include values for the capacity, simplifying assumptions will be made to include a value. The electric dynamic behavior is mainly caused by the double layer at the electrode-electrolyte interface. Due to the geometric arrangement of the electrode and electrolyte, the capacity can be estimated by a simple plate capacitor [29]. The total activation overpotential is mainly caused by the reaction at the anode ($v_{\mathrm{act,an}} = v_{\mathrm{act}}$), wherefore the cathode is neglected and the resistance R_{act} can be calculated as displayed in (A.2) [8]. Therefore, R_{act} is already nonlinear, but in addition the temperature dependency of the electrode exchange current density j_0 is expressed by an exponential function [8]. The temperature influence on the open circuit voltage v_{ocv}, defined by the Nernst-Equation, is also nonlinear (A.3). By adding up v_{ocv} with the ohmic overpotential v_{ohm} based on (A.4) and the activation overpotential, the total electrolyzer voltage can be described as

$$v_{\mathrm{ely}} = v_{\mathrm{ocv}} + v_{\mathrm{act}} + v_{\mathrm{ohm}}. \tag{64}$$

Again, the temperature dependency of the protonic membrane conductivity σ_{mem} is nonlinear and contains an exponential expression. For describing the thermal dynamics, an ODE is used as in

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \left(\sum_{k=1}^{s_1} \dot{W}_{\mathrm{k}} + \sum_{k=1}^{s_2} \dot{Q}_{\mathrm{k}} + \sum_{k=1}^{s_3} \dot{H}_{\mathrm{k}}\right)/C_{\mathrm{th}}, \tag{65}$$

with s_1 power ($\dot{W}$), s_2 heat ($\dot{Q}$) and s_3 enthalpy ($\dot{H}$) streams, and the lumped thermal capacity C_{th} [8]. The cooling heat $\dot{Q}_{\mathrm{cool}}$ is directly controlled by a PI controller, and described by a state space model as

$$\begin{aligned} \dot{Q}_{\mathrm{cool}} &= x_{\mathrm{PI}} + K_{\mathrm{P}}\Delta T \\ \frac{\mathrm{d}x_{\mathrm{PI}}}{\mathrm{d}t} &= K_{\mathrm{I}}\Delta T, \end{aligned} \tag{66}$$

with $K_{\mathrm{P}} = -152.023$, $K_{\mathrm{I}} = -0.218$ and the temperature difference as control deviation $\Delta T = T_{\mathrm{set}} - T$. Because of the ODEs describing the thermal, electric and controller dynamics, the model has three states ($n = 3$). Since the electrolyzer is current-controlled, the input current i_{ely} ($m = 1$) results into an output voltage v_{ely}, which is set by the converter. Additionally, the generated hydrogen substance stream $\dot{n}_{\mathrm{H}_2}$ in mol s^{-1} is considered as an output ($p = 2$), described by Farady's law in

$$\dot{n}_{\mathrm{H}_2} = \frac{n_{\mathrm{c}} i_{\mathrm{ely}}}{2F} n_{\mathrm{F}}, \tag{67}$$

containing the number of cells n_{c}, Faraday's constant F and faradaic efficiency $n_{\mathrm{F}} = 1$. With the state and output vectors

$$\mathbf{x} = \begin{pmatrix} T \\ v_{\mathrm{act}} \\ x_{\mathrm{PI}} \end{pmatrix}, \mathbf{y} = \begin{pmatrix} v_{\mathrm{ely}} \\ \dot{n}_{\mathrm{H}_2} \end{pmatrix}, \tag{68}$$

a nonlinear state space model can be derived.

Considering the broad operating range of electrolyzers, linearization of this nonlinear model at one operation point could be insufficient regarding the model accuracy. Thus, the projection multilinearization method from [16] was chosen, as it can be defined for the operating range of states, input and outputs. In that operating range the number of values per variable is $N = 30$ and the maximal multilinear order is $m_n = 2$. The lower and upper bound for the multilinearization were estimated by the minimum and maximum values of the nonlinear model in the simulated range of 200 to 400 A, which corresponds to 50 to 100 % of the nominal load of the electrolyzer. This leads to the matrices $\mathbf{F} \in \mathbb{R}^{3\times16}$ and $\mathbf{G} \in \mathbb{R}^{2\times16}$ of the eMTI model. The eMTI model in the form of (6) and (7) can easily be transferred to implicit form, as described in (13) and (14) in subsection 2.3.

The eMTI model was decomposed by the MATLAB tensor toolbox, resulting in a reduced rank of $r_{\mathsf{F}} = 9$ and $r_{\mathsf{G}} = 8$. A linear transformation was applied to the model according to (54)–(56) for getting an operating region between 0 and 0.5. For numerical comparison, linearization at 300 A, corresponding to 75 % of the nominal load was done by the MATLAB function `linmod`.

At the operation point, the linear time-invariant (LTI) model fits the nonlinear dynamics better than the eMTI model, as it can be observed for v_{act} in Figure 9 (c). After 10 min of electrolyzer operation at 75 %, the current jumps to nominal load, where the eMTI model fits better. The same behavior can be observed at the second jump to 200 A. Especially the states of the thermal dynamics T and x_{PI} of the eMTI model show good accordance with the nonlinear model in Figure 9 (b) and (d). This example shows the benefits of the eMTI modeling approach for wider operating ranges compared to linearization.

Additionally, the computational time of the simulations are compared. The simulations are performed on a conventional laptop, while using the automatic solver and step-size selection within Simulink. The simulation time of the LTI model is the smallest (21 s), followed by the eMTI (26 s) and the nonlinear model (35 s). These results are expected as the LTI and eMTI are approximations of different complexity of the nonlinear model. In addition, the memory requirements are compared for the LTI and the eMTI models. For the comparison the matrices **A**, **B**, **C**, **D**

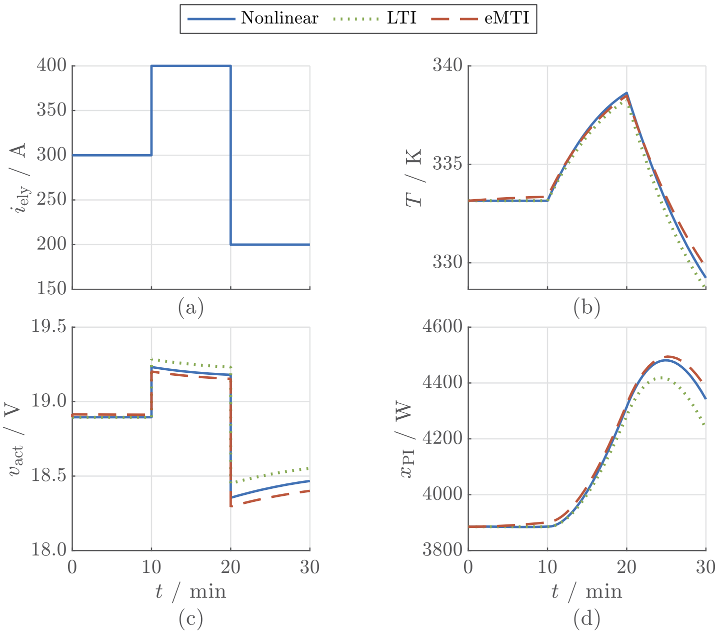


Figure 9: Electrolyzer model comparison for: (a) current input i_{ely}, (b) temperature state T, (c) activation voltage state v_{act}, and (d) cooling power state x_{PI}.

of the LTI model and the normalized decomposed tensors F, G of the eMTI model in double precision are used. The required memory of the eMTI model (888 bytes) is significantly higher than that of the LTI model (160 bytes). However, the advantages of decomposed tensors for MTI models will be noticeable for larger systems as the increase of elements is linear compared to an exponential increase for full tensors and for the LTI model. An example for this is given in Figure 10. The intersections of the graphs depend on the rank of the decomposed tensors. The memory requirements of the nonlinear model are not taken into account due to the significantly larger memory of the overall Simulink file. Therefore, a reasonable comparison is not possible here.

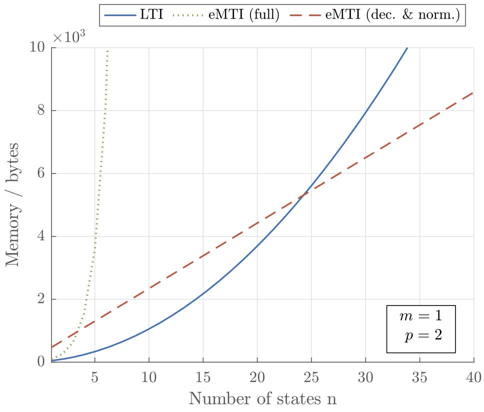


Figure 10: Comparison of the memory consumption of an LTI model, eMTI model using full tensors and eMTI model using decomposed and normalized tensors with $r_{\mathrm{F}} = 9$ and $r_{\mathrm{G}} = 8$ as a function of number of states.

6.2 Buck converter

The buck converter steps down the voltage of 800 V provided by the infinite DC bus V_{DC}. This paper takes an ideal model of a buck converter as depicted in Figure 11. At the load side of the converter, since the electrolyzer model provides the voltage v_{ely} for a given current input i_{ely}, a controlled voltage source is used to connect the electrolyzer to the DC bus. It is connected in series with a small resistance R_{S} of 10 µΩ. The switching power electronic S, e. g., an IGBT, is modelled as an ideal switch. The diode, inductor L, and capacitor C, have no internal, or parasitic, resistances. The inductor is sized to 1 mH and the capacitor to 680 µF similarly to the filter design used in [9].

This paper assumes an average model of the switching, and thus, the buck converter excluding the controller can be described by the LTI state-space model

$$\begin{pmatrix} \dot{i}_{\mathrm{L}} \\ \dot{v}_{\mathrm{C}} \end{pmatrix} = \begin{pmatrix} 0 & -\frac{1}{L} \\ \frac{1}{C} & -\frac{1}{R_{\mathrm{s}}C} \end{pmatrix} \begin{pmatrix} i_{\mathrm{L}} \\ v_{\mathrm{C}} \end{pmatrix} + \begin{pmatrix} \frac{V_{\mathrm{DC}}}{L} & 0 \\ 0 & \frac{1}{R_{\mathrm{s}}C} \end{pmatrix} \begin{pmatrix} d \\ v_{\mathrm{ely}} \end{pmatrix}, \tag{69}$$

$$\begin{pmatrix} i_{\mathrm{ely}} \end{pmatrix} = \begin{pmatrix} 0 & \frac{1}{R_{\mathrm{s}}} \end{pmatrix} \begin{pmatrix} i_{\mathrm{L}} \\ v_{\mathrm{C}} \end{pmatrix} + \begin{pmatrix} 0 & -\frac{1}{R_{\mathrm{s}}} \end{pmatrix} \begin{pmatrix} d \\ v_{\mathrm{ely}} \end{pmatrix}, \tag{70}$$

where the duty cycle d is the input coming from the controller. This LTI model can be easily transformed into an iMTI model as outlined in (13)–(14) in subsection 2.3.

6.3 Local energy system

Finally, the coupled system of the buck converter and electrolyzer including a controller is analyzed. Figure 12 shows the closed control loop of the coupled system. The controller tracks the difference between the reference power P_{ref} and the actual power P delivered to the electrolyzer. Using the tracking error, the duty cycle d is calculated with a PI controller, with gains $K_{\mathrm{P}} = 5.0 \cdot 10^{-4}$ and $K_{\mathrm{I}} = 1.0 \cdot 10^{-2}$, and saturated to $[0, 1]$. The buck converter takes the duty cycle d and the output voltage of the electrolyzer v_{ely} to provide the input current i_{ely} to the electrolyzer.

As mentioned in section 3 the connection of eMTI systems can lead to an overall system, which is not in the eMTI class. Due to Lemma 3.1 the application example is formulated in implicit form, which guarantees the overall model to be within the iMTI class. The connection equations for the system are given as

$$u - u_{\mathrm{I}} - K_{\mathrm{P}} e = 0, \quad (71)$$

$$e - P_{\mathrm{ref}} + v_{\mathrm{ely}} i_{\mathrm{ely}} = 0, \quad (72)$$

$$y - d = 0. \quad (73)$$

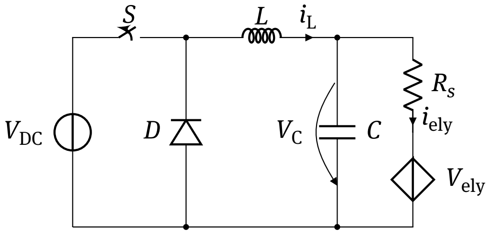


Figure 11: Ideal buck converter scheme and connection to the electrolyzer.

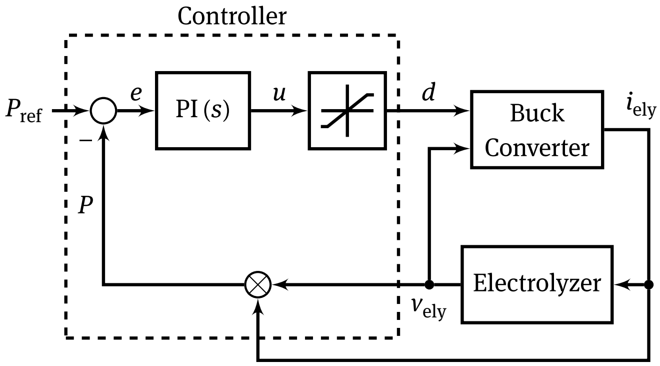


Figure 12: Closed control loop.

The iMTI representation of the controller can be written in CP tensor form

$$\langle\, \mathsf{H}_{\mathrm{c}} \mid \mathsf{M}(y_{\mathrm{c}}, \mathbf{b}_{\mathrm{c}}, \dot{x}_{\mathrm{c}}, x_{\mathrm{c}}, \mathbf{u}_{\mathrm{c}}) \,\rangle = \mathbf{0}, \quad (74)$$

$$\langle\, \mathsf{L}_{\mathrm{c}} \mid \mathsf{M}(y_{\mathrm{c}}, \mathbf{b}_{\mathrm{c}}, \dot{x}_{\mathrm{c}}, x_{\mathrm{c}}, \mathbf{u}_{\mathrm{c}}) \,\rangle \geq \mathbf{0}, \quad (75)$$

where $y_{\mathrm{c}} = d$, $\mathbf{b}_{\mathrm{c}} = \begin{bmatrix} b_1 & b_2 & b_3 & b_4 \end{bmatrix}^{\mathrm{T}}$, $x_{\mathrm{c}} = u_{\mathrm{I}}$, and $\mathbf{u}_{\mathrm{c}} = \begin{bmatrix} P_{\mathrm{ref}} & v_{\mathrm{ely}} & i_{\mathrm{ely}} \end{bmatrix}^{\mathrm{T}}$. In this representation H_{c} corresponds to the equalities of the controller (61), the saturation (15), (16), (19), (20), (23), the buck converter (69), (70), the electrolyzer (63)–(68) as well as the connection equations (71)–(73), while L_{c} corresponds to the inequalities (21) and (22).

Special differential algebraic equations (DAE) solvers for descriptor models can be used for simulation of CP iMTI models like the one derived for the closed loop energy system here [4]. For this paper, the closed loop model is implemented in Simulink with CP eMTI submodels. Three different setups are compared, where the electrolyzer is either modeled as an LTI, eMTI or nonlinear system. The formulation of the electrolyzer w.r.t these three model classes is given in subsection 6.1. The rest of the system remains unchanged during the comparison shown in Figure 13.

A total simulation time of 30 minutes is used. During the simulation the reference power P_{ref} is changed twice using a step function. The step is performed w. r. t. subsection 6.1. The initial reference power P_{ref} until 10 minutes simulation time is related to 75 % of the nominal power of the electrolyzer. After the first step the reference power is increased by 1/3. After another 10 minutes

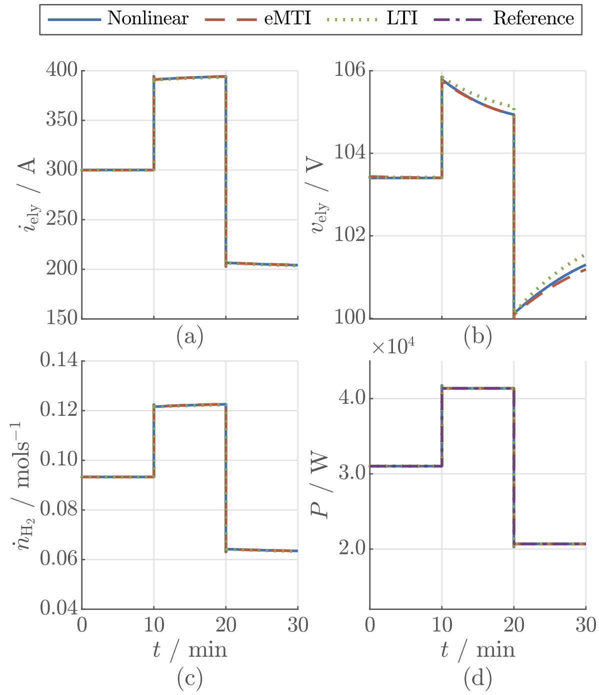


Figure 13: Closed loop simulation results: (a) electrolyzer input current i_{ely}, (b) electrolyzer output voltage v_{ely}, (c) electrolyzer output hydrogen stream $\dot{n}_{\text{H}_2}$, and (d) electrolyzer power P tracking.

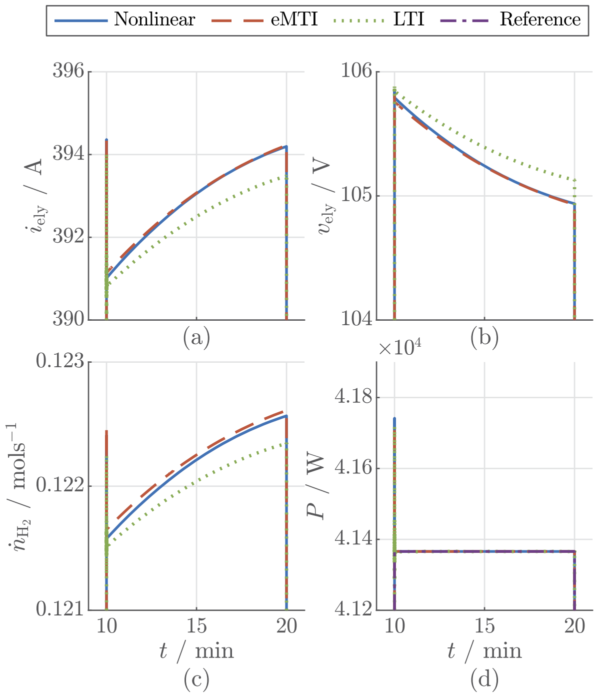


Figure 14: Zoom of closed loop simulation results: (a) electrolyzer input current i_{ely}, (b) electrolyzer output voltage v_{ely}, (c) electrolyzer output hydrogen stream $\dot{n}_{\text{H}_2}$, and (d) electrolyzer power P tracking.

of simulation time the reference power is decreased by 2/3 of the initial power. Figure 13 gives an overview of the entire simulation regarding the time and operational range. Here, all models show comparable results.

A more detailed view of the results is given in Figure 14. An overshoot for the controller can be seen for all models after the first step. During the simulation from 10 to 20 minutes the eMTI model is outperforming the LTI model. Especially for an advanced simulation time after the first step, the deviation for the LTI model is large compared to the eMTI model.

The whole model can also be represented by a single tensor and inequalities

$$\langle\, \mathsf{L}_{\text{tot}} \mid \mathsf{M}\,(y_{\text{c}}, \mathbf{b}_{\text{c}}, \dot{x}_{\text{c}}, x_{\text{c}}, \mathbf{u}_{\text{c}})\,\rangle \;\geq\; \mathbf{0}, \tag{76}$$

where L_{tot} appends the equalities to the inequalities given by L_{c}. This can be achieved by expressing each equality by two opposing inequalities as given in (18). Although the number of inequalities gets larger, such a representation could lead to reductions because of the possible good low rank tensor approximations exploiting the internal structure of L_{tot}.

7 Conclusion

This paper introduces iMTI models as a new class – which is interesting for large-scale hybrid complex engineering applications because of three reasons. The parameters can be represented as normalized tensors – which allow tremendous model reduction by modern decomposition methods. Nonlinear effects can be modeled – which is essential for the large signal behaviour of real world systems. Boolean dynamics is exactly modeled because its intrinsically multilinear – which is necessary to describe all discrete-valued subsystems, switches, and automata. Possible white-box modeling steps are shown by a small energy system example, including multilinearization, decomposition of the multilinear model into factors, normalization and composition back to overall models. Research on multilinear black-box and grey-box parameter identification is parallel ongoing.

Future research activities will apply the iMTI form to modeling large power networks, where discrete-valued dynamics, such as the switches in power converters, become more present with the growth of renewables. This includes the quick composition of multiphysical energy systems models by using the implicit form. This will add a powerful tool to address the challenges arising in the European power system growing in complexity, for which efficient models are needed.

Funding: This work was partly supported by the project SONDE of the Federal Ministry of Education and Research, Germany (Grant-No.: 13FH144PA8) and partly supported by the Free and Hanseatic City of Hamburg.

Appendix A. Electrolyzer model

In this section some equations of the nonlinear electrolyzer model are further described to point out the nonlinearity of the problem. The equations calculate

- the total capacity

$$C_{\mathrm{DL}} = \frac{C_{\mathrm{DL,cell}} A}{n_{\mathrm{c}}}, \tag{A.1}$$

  with electrode surface A, specific cell capacity $C_{\mathrm{DL,cell}}$, and number of cells n_{c},

- the activation resistance

$$R_{\mathrm{act}} = \frac{v_{\mathrm{act,cell}}}{i_{\mathrm{ely}}} n_{\mathrm{c}} = \frac{\frac{RT}{2\alpha_{\mathrm{an}} F} \operatorname{asinh}\left(\frac{j}{2 j_{0,\mathrm{an}}}\right)}{i_{\mathrm{ely}}} n_{\mathrm{c}}, \tag{A.2}$$

  including the Volmer-Butler-Equation with current density j, electrode exchange current density j_0, charge transfer coefficient α, temperature T, Faraday's constant F, and universal gas constant R [8, 31],

- the open-circuit cell voltage

$$\begin{aligned} v_{\mathrm{ocv,cell}} &= (v_{\mathrm{std}} - 0.0009(T - T_{\mathrm{std}})) \\ &\quad + \frac{RT}{2F}\left[\ln\left(\frac{p_{\mathrm{H_2}} p_{\mathrm{O_2}}^{0.5}}{p_{\mathrm{H_2O}}(T)}\right)\right], \end{aligned} \tag{A.3}$$

  with voltage at standard conditions v_{std}, standard temperature T_{std}, and partial pressure p_s of the substance s [8],

- and the Ohmic cell overpotential

$$v_{\mathrm{ohm,cell}} = \frac{\delta_{\mathrm{mem}}}{\sigma_{\mathrm{mem,std}} \exp\left(-\frac{E_{\mathrm{pro}}}{R}\left(\frac{1}{T} - \frac{1}{T_{\mathrm{std}}}\right)\right)} j, \tag{A.4}$$

  with membrane thickness δ_{mem}, standard protonic conductivity $\sigma_{\mathrm{mem,std}}$, and activation energy required for the proton transport in the membrane E_{pro} [8].

For further description of the parameters and the thermal dynamics see [8].

Appendix B. Composition

This section provides the parameter matrices **F** and **G** of the eMTI model example in section 3. The state equation of system 1 is given by

$$\begin{aligned} \dot{x}_1 &= f_{12} x_1 + f_{14}\, x_1\, u_1 = \mathbf{F}_1 \mathbf{m}_1(x_1, u_1) \\ &= \begin{pmatrix} 0 & f_{12} & 0 & f_{14} \end{pmatrix} \begin{pmatrix} 1 \\ x_1 \\ u_1 \\ x_1\, u_1 \end{pmatrix} \end{aligned}$$

and the output equations

$$\begin{aligned} \mathbf{y}_1 &= \begin{pmatrix} y_{11} \\ y_{12} \end{pmatrix} = \begin{pmatrix} g_{12} x_1 \\ g_{18}\, x_1\, u_1 \end{pmatrix} = \mathbf{G}_1 \mathbf{m}_1(x_1, u_1) \\ &= \begin{pmatrix} 0 & g_{12} & 0 & 0 \\ 0 & 0 & 0 & g_{18} \end{pmatrix} \begin{pmatrix} 1 \\ x_1 \\ u_1 \\ x_1\, u_1 \end{pmatrix}. \end{aligned}$$

For system 2, the corresponding equations are

$$\begin{aligned} \dot{\mathbf{x}}_2 &= f_{21} + f_{27}\, u_{21}\, u_{22} = \mathbf{F}_2 \mathbf{m}_2(x_2, u_{21}, u_{22}) \\ &= \begin{pmatrix} f_{21} & 0 & 0 & 0 & 0 & 0 & f_{27} & 0 \end{pmatrix} \begin{pmatrix} 1 \\ x_2 \\ u_{21} \\ x_2\, u_{21} \\ u_{22} \\ u_{22}\, x_2 \\ u_{22}\, u_{21} \\ u_{22}\, x_2\, u_{21} \end{pmatrix}, \end{aligned}$$

and

$$\begin{aligned} y_2 &= g_{26}\, u_{22}\, x_2 = \mathbf{G}_2 \mathbf{m}_2(x_2, u_{21}, u_{22}) \\ &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & g_{26} & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ x_2 \\ u_{21} \\ x_2\, u_{21} \\ u_{22} \\ u_{22}\, x_2 \\ u_{22}\, u_{21} \\ u_{22}\, x_2\, u_{21} \end{pmatrix}. \end{aligned}$$

A series connection $\mathbf{u}_2 = \begin{pmatrix} u_{21} \\ u_{22} \end{pmatrix} = \begin{pmatrix} y_{11} \\ y_{12} \end{pmatrix} = \mathbf{y}_1$ of the two models is not within the eMTI class, but the overall iMTI model can be derived from the iMTI submodels, where System 1 can be written by $\mathbf{H}_1 \in \mathbb{R}^{3\times 32}$ as

$$\mathbf{H}_1\,\mathbf{m}_1(\dot{x}_1, x_1, u_1, y_{11}, y_{12}) = \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 0 \\ f_{12} & g_{12} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ f_{14} & 0 & g_{18} \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} 1 \\ \dot{x}_1 \\ x_1 \\ \dot{x}_1 x_1 \\ u_1 \\ \dot{x}_1 u_1 \\ x_1 u_1 \\ \dot{x}_1 x_1 u_1 \\ y_{11} \\ \dot{x}_1 y_{11} \\ \vdots \\ \dot{x}_1 x_1 u_1 y_{11} y_{12} \end{pmatrix} = \mathbf{0}$$

and system 2 with $\mathbf{H}_2 \in \mathbb{R}^{2\times 32}$ as

$$\mathbf{H}_2\,\mathbf{m}_2(\dot{x}_2, x_2, u_{21}, u_{22}, y_2) = \begin{pmatrix} f_{21} & 0 \\ -1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & g_{26} \\ 0 & 0 \\ f_{27} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -1 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} 1 \\ \dot{x}_2 \\ x_2 \\ \dot{x}_2 x_2 \\ u_{21} \\ \dot{x}_1 u_{21} \\ x_1 u_{21} \\ \dot{x}_1 x_1 u_{21} \\ u_{22} \\ \dot{x}_2 u_{22} \\ x_2 u_{22} \\ \dot{x}_2 x_2 u_{22} \\ u_{21} u_{22} \\ \dot{x}_1 u_{21} u_{22} \\ x_1 u_{21} u_{22} \\ \dot{x}_1 x_1 u_{21} u_{22} \\ y_2 \\ \dot{x}_2 y_2 \\ \vdots \\ \dot{x}_2 x_2 u_{21} u_{22} y_2 \end{pmatrix} = \mathbf{0}\,,$$

where the transposed forms of $\mathbf{H}$ are chosen for better visualization.

References

1. Bader, B. W. and T. G. Kolda. 2008. Efficient MATLAB computations with sparse and factored tensors. *SIAM Journal on Scientific Computing* 30(1): 205–231.
2. Bader, B. W., T. G. Kolda, et al. April 2021. Tensor Toolbox for MATLAB, Version 3.2.1. Available at https://www.tensortoolbox.org/.
3. Bhaskara, A., M. Charikar, A. Moitra and A. Vijayaraghavan. 2014. Open problem: Tensor decompositions: Algorithms up to the uniqueness threshold? *JMLR: Workshop and Conference Proceedings* 35: 1–3.
4. Brenan, K. E., S. L. Campbell and L. R. Petzold. 1996. *Numerical solution of initial-value problems in differential-algebraic equations*. Classics in applied mathematics. Society for Industrial and Applied Mathematics (SIAM).
5. Coutinho, Y. A., N. Vervliet, L. de Lathauwer and N. Moelans. 2020. Combining thermodynamics with tensor completion techniques to enable multicomponent microstructure prediction. *npj Computational Materials* 6(1).
6. Cox, D. A., J. B. Little and D. O'Shea. 2015. *Ideals, Varieties, and Algorithms*. 4th edition. Springer International Publishing.
7. de Silva, V. and L.-H. Lim. 2008. Tensor rank and the ill-posedness of the best low-rank approximation problem. *SIAM Journal on Matrix Analysis and Applications* 30(3): 1084–1127.
8. Espinosa-López, M., C. Darras, P. Poggi, R. Glises, P. Baucour, A. Rakotondrainibe, S. Besse and P. Serre-Combe. 2018. Modelling and experimental validation of a 46 kw pem high pressure water electrolyzer. *Renewable Energy* 119: 160–173.
9. Gao, F., B. Blunier, M. G. Simões and A. Miraoui. 2011. Pem fuel cell stack modeling for real-time emulation in hardware-in-the-loop applications. *IEEE Transactions on Energy Conversion* 26(1): 184–194.
10. Guilbert, D. and G. Vitale, Experimental validation of an equivalent dynamic electrical model for a proton exchange membrane electrolyzer. In: *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*, 12.06.2018–15.06.2018. IEEE, pp. 1–6.
11. Hitchcock, F. L. 1927. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics* 6: 164–189.
12. Jouni, M., M. Dalla Mura and P. Comon, July 2018. Some issues in computing the CP decomposition of NonNegative Tensors. In: *LVA/ICA 2018 – 14th International Conference on Latent Variable Analysis and Signal Separation* (Guildford, United Kingdom), pp. 57–66.
13. Kolda, T. G. and B. W. Bader. 2009. Tensor decompositions and applications. *SIAM Review* 51(3): 455–500.
14. Kruppa, K. 2018. *Multilinear Design of Decentralized Controller Networks for Building Automation Systems*. Ph.d. dissertation, HafenCity Universität Hamburg.
15. Kruppa, K. and G. Lichtenberg. 2020. A heating systems application of feedback linearization for mti systems in a tensor framework. In: (M.S. Obaidat, T. Ören and F.D. Rango, eds) *Simulation and Modeling Methodologies, Technologies and Applications*. Springer International Publishing, Cham, pp. 126–152.
16. Kruppa, K., G. Pangalos and G. Lichtenberg. 2014. Multilinear approximation of nonlinear state space models. *IFAC Proceedings Volumes* 47(3): 9474–9479.
17. Laub, A. 2005. *Matrix Analysis for Scientists and Engineers*. Society for Industrial and Applied Mathematics, Davis, CA.
18. Lautenschlager, B. 2019. *Data-Driven Learning and Model Predictive Control for Heating Systems*. Ph.d. dissertation, Universität Hamburg, HafenCity.
19. Lautenschlager, B., K. Kruppa and G. Lichtenberg. 2015.

Convexity properties of the model predictive control problem for subclasses of multilinear time-invariant systems. *IFAC-PapersOnLine* 48(23): 148–153. 5th IFAC Conference on Nonlinear Model Predictive Control NMPC 2015.

20. Lunze, J. and F. Lamnabhi-Lagarrigue. 2009. *Handbook of hybrid systems control: Theory, tools, applications*. Cambridge University Press, Cambridge, UK and New York.
21. Milano, F., F. Dörfler, G. Hug, D. J. Hill and G. Verbič. 2018. Foundations and challenges of low-inertia systems. In: *2018 Power Systems Computation Conference (PSCC)*, pp. 1–25.
22. Modelica Association. 2021. Modelica – a unified object-oriented language for systems modeling. Available at https://modelica.org/documents/MLS.pdf.
23. Olivier, P., C. Bourasseau and P. B. Bouamama. 2017. Low-temperature electrolysis system modelling: A review. *Renewable and Sustainable Energy Reviews* 78: 280–300.
24. Oseledets, I., A. Boyko and D. Savostyanov. June 2014. TT-Toolbox 2.3. Available at https://github.com/oseledets/TT-Toolbox.
25. Pang, Y., T. Hao, A. Dugad, Y. Zhou and E. Solomonik. 2020. Efficient 2d tensor network simulation of quantum systems. In: *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE Press.
26. Pangalos, G. 2016. *Model-based controller design methods for heating systems*. Ph.d. dissertation, Technische Universität Hamburg.
27. Pangalos, G., A. Eichler and G. Lichtenberg. 2015. Hybrid multilinear modeling and applications. Vol. 319 of *Advances in Intelligent Systems and Computing*. Springer International Publishing, pp. 71–85.
28. Pangalos, G. and G. Lichtenberg. 2012. Approach to Boolean Controller Design by Algebraic Relaxation for Heating Systems. In: *4th IFAC Conference on Analysis and Design of Hybrid Systems* (Eindhoven).
29. Schmidt, V. M. 2003. *Elektrochemische Verfahrenstechnik: Grundlagen, Reaktionstechnik, Prozeszoptimierung*. Wiley-VCH, Weinheim.
30. Vervliet, N., O. Debals, L. Sorber, M. Van Barel and L. De Lathauwer. Mar. 2016. Tensorlab 3.0. Available at https://www.tensorlab.net.
31. Yodwong, B., D. Guilbert, M. Hinaje, M. Phattanasak, W. Kaewmanee and G. Vitale. 2021. Proton exchange membrane electrolyzer emulator for power electronics testing applications. *Processes* 9(3): 498.
32. Zhegalkin, I. 1928. Arithmetics of symbolic logic. *Mat. Sb.* 35(3-4): 311–377.

Bionotes

Gerwald Lichtenberg
HAW Hamburg, Faculty Life Sciences,
Hamburg, Germany
gerwald.lichtenberg@haw-hamburg.de

Gerwald Lichtenberg is Professor of Physics and Control Engineering at the Faculty of Life Sciences at the HAW Hamburg and scientific director of the Fraunhofer application center for Integration of Local Energy Systems ILES. His research areas are in the fields of model-based and learning control as well as fault diagnosis of complex systems such as local energy networks, building systems, or particle accelerators. The methodological focus of his work is on tensor decomposition methods and multilinear models.

Georg Pangalos
Fraunhofer IWES, Application Center
Integration of Local Energy Systems (ILES),
Hamburg, Germany
georg.pangalos@iwes.fraunhofer.de

Georg Pangalos received the Dipl.-Ing. degree in electrical engineering and his Ph.D. in control engineering from the Hamburg University of Technology in 2011 and 2015 respectively. Since 2015 he has been working at the Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. in Hamburg, Germany. His research interests include the areas of renewable energy integration and energy systems transition.

Carlos Cateriano Yáñez
Fraunhofer IWES, Application Center Integration of Local Energy Systems (ILES), Hamburg, Germany
Universitat Politècnica de València, Instituto Universitario de Automática e Informática Industrial, València, Spain
carlos.cateriano.yanez@iwes.fraunhofer.de

Carlos Cateriano Yáñez received the B.Eng. degree in industrial and systems engineering from the University of Piura, Lima, Peru, in 2012 and the M.Eng. in renewable energy systems from the Hamburg University of Applied Sciences, Hamburg, Germany, in 2017. He is currently pursuing the Ph.D. degree in automation, robotics, and industrial computer science at Universitat Politècnica de València, Valencia, Spain, in cooperation with the Hamburg University of Applied Sciences, Hamburg, Germany. Since 2017 he has been working at the Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. in Hamburg, Germany. His research interests include model predictive control and its applications in the field of power systems integrated with renewable energy sources.

Aline Luxa
Fraunhofer IWES, Application Center Integration of Local Energy Systems (ILES), Hamburg, Germany
aline.luxa@iwes.fraunhofer.de

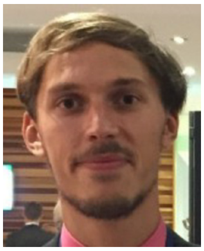
Aline Luxa received the B.Sc. and M.Sc. degree in energy and process engineering from the Technical University of Berlin in 2017 and 2020 respectively. Since 2020 she has been part of the Application Center for Integration of Local Energy Systems at the Fraunhofer Institute for Wind Energy Systems. In the scope of the Application Center, she is in close cooperation with the Hamburg University of Applied Sciences. Her research topics include modelling and control of grid supporting hydrogen systems.

Niklas Jöres
Fraunhofer IWES, Application Center Integration of Local Energy Systems (ILES), Hamburg, Germany
niklas.joeres@iwes.fraunhofer.de

Niklas Jöres received his B.Sc. in Renewable Energy Systems in 2016 from the University of Stuttgart. At the same university he received his M.Sc in Energy Engineering in 2020 with the focus on thermal turbomachinery and wind energy systems. Since 2020 he has been working at the Fraunhofer Institute for Wind Energy Systems. His research focuses on modelling and optimal operation of local energy systems.

Leona Schnelle
HAW Hamburg, Faculty Life Sciences, Hamburg, Germany
leona.schnelle@haw-hamburg.de

Leona Schnelle received her M.Eng. in Renewable Energy Systems from the HAW Hamburg in 2019. Since 2020 she has been working as research associate at the HAW Hamburg for the BMBF project SONDE on normalized multilinear modelling for diagnosis applications in the building energy sector.



Christoph Kaufmann
Fraunhofer IWES, Application Center Integration of Local Energy Systems (ILES), Hamburg, Germany
christoph.kaufmann@iwes.fraunhofer.de

Christoph Kaufmann received his B.Sc. in Industrial Engineering and Management from the University of Hamburg in 2016 and his M.Eng. in Renewable Energy Systems at the HAW Hamburg in 2019. He is currently pursuing a PhD with a focus on stability of multilinear systems with applications to power systems.