

## Tensor nuclear norm LPV subspace identification

Gunes, Bilal; van Wingerden, Jan Willem; Verhaegen, Michel

**DOI**

[10.1109/TAC.2018.2800772](https://doi.org/10.1109/TAC.2018.2800772)

**Publication date**

2018

**Document Version**

Final published version

**Published in**

IEEE Transactions on Automatic Control

**Citation (APA)**

Gunes, B., van Wingerden, J. W., & Verhaegen, M. (2018). Tensor nuclear norm LPV subspace identification. *IEEE Transactions on Automatic Control*, 63(11), 3897-3903.  
<https://doi.org/10.1109/TAC.2018.2800772>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

# Tensor Nuclear Norm LPV Subspace Identification

Bilal Gunes<sup>ID</sup>, Jan-Willem van Wingerden<sup>ID</sup>, *Member, IEEE*, and Michel Verhaegen<sup>ID</sup>, *Member, IEEE*

**Abstract**—Linear parameter varying (LPV) subspace identification methods suffer from an exponential growth in number of parameters to estimate. This results in problems with ill-conditioning. In literature, attempts have been made to address the ill-conditioning by using regularization. Its effectiveness hinges on suitable a priori knowledge. In this paper, we propose using a novel, alternative regularization. That is, we first show that the LPV sub-Markov parameters can be organized into several tensors that are multilinear *low rank* by construction. Namely, their matricization along any mode is a low-rank matrix. Then, we propose a novel convex method with tensor nuclear norm regularization, which exploits this low-rank property. Simulation results show that the novel method can have higher performance than the regularized LPV-PBSID<sub>opt</sub> technique in terms of variance accounted for.

**Index Terms**—Closed-loop identification, identification, linear parameter varying (LPV) systems, subspace methods, tensor regression.

## I. INTRODUCTION

Linear parameter varying (LPV) systems have been used in many applications. Some examples are wind turbines [2], [12], aircraft applications [1], batteries [27], compressors [14], and wafer stages [34], [41]. These LPV systems are linear systems whose dynamics vary with a known time-varying parameter vector. They are suitable for describing many applications in greater detail than linear time invariant (LTI) systems can. Just like for LTI systems, there also exists a powerful control design framework for LPV systems [29], which can guarantee stability, performance, and robustness. This is generally not the case for nonlinear systems. Most control design frameworks do require an LPV state-space model of the system.

This model can be obtained from measurement data using system identification. There are many LPV identification methods that can be classified as global and local methods [7], [30], [32]. Local methods hinge on the property that for constant scheduling parameters the LPV system behaves as an LTI system. They identify LTI models at several fixed constant scheduling parameter conditions and then use interpolation techniques to obtain an LPV model. For applications where these experiments are possible, this can yield good results [7], [25], [28], [30], [32]. Global methods on the other hand do not have this requirement and only use one experiment. In this paper, we focus on global LPV identification methods.

These methods can be further divided into input–output and subspace methods. Input–output methods yield input–output LPV models and have received considerable attention in the literature [4], [24], [32]. But the preferred model structure for mainstream LPV control design

Manuscript received May 31, 2017; revised October 31, 2017 and January 1, 2018; accepted January 26, 2018. Date of publication February 1, 2018; date of current version October 25, 2018. This work was supported by the Design for Reliable Power Performance (D4REL) program, project code P91202. Recommended by Associate Editor C. M. Lagoa. (Corresponding author: Bilal Gunes.)

The authors are with the Delft Center for Systems and Control, Delft University of Technology, Delft 2628, The Netherlands (e-mail: B.Gunes@tudelft.nl; J.W.vanWingerden@tudelft.nl; M.Verhaegen@tudelft.nl).  
Digital Object Identifier 10.1109/TAC.2018.2800772

methodologies is state-space, and transformation from input–output to state-space models is problematic in the LPV case [33]. Subspace methods have also received considerable attention [6], [9], [10], [12], [23], [36], [37]. They directly produce state-space models, and can deal naturally with multiple input multiple output and closed-loop systems. In this paper, we focus on subspace methods. If the scheduling sequence has some special structure, for example, is periodic [10], white noise [9] or piecewise constant [36], then tailored methods can be used. However, this is not true for all applications. There exist several LPV subspace methods for this case [6], [13], [37], and they suffer from the “curse-of-dimensionality” during their first regression step. This means that the number of LPV sub-Markov parameters to be estimated can quickly, vastly exceed the number of data points. This results in ill-conditioned problems, memory and computational cost issues. In this paper, we focus on the ill-conditioning and continue to deal with the latter two issues in [16] using dual problems [37]. One way to tackle ill-condition of the problem is to use regularization. Methods with Tikhonov regularization [37] and matrix nuclear norms [12] have been proposed. The former penalizes the magnitude of estimate parameters, and the latter exploits the approximate low-rank property of the state-revealing matrix. In this paper, we propose using an alternative regularization, which arguably exploits more structure. Namely, we propose using tensor nuclear norms [31], in order to exploit the exact multilinear low-rank property of the parameter tensor.

Our proposed method can also be seen as an extension of the method presented in [21] to the LPV Multiple Input Multiple Output with tensors case.

A tensor is a multidimensional generalization of a matrix. That is, it can have more than two dimensions. This multidimensional structure can be exploited using techniques from the tensor framework such as tensor nuclear norms [31]. Tensor and matrix nuclear norms have received considerable attention in literature [20], [26], [31], [38], [39]. Matrix nuclear norms can be used to exploit knowledge that some matrices are low rank. Tensor nuclear norms can be used to exploit knowledge that some tensors have low-rank matricizations.

In this paper, we show that the LPV sub-Markov parameters can be organized into several tensors whose matricizations are all low rank. Then, this property is exploited through tensor nuclear norm regularization to obtain a novel convex LPV subspace identification method.

This paper relates to previous work as follows. In [17], [18], the problem of having a large number of LPV sub-Markov parameters was also tackled by using tensor techniques. However, in [17] the LPV sub-Markov parameters were organized in a single padded tensor and then a nonlinear (polyadic) decomposition and parametrization was used. In [18], a tensor train decomposition and parametrization is used. Both methods do not use multilinear low-rank properties or nuclear norms and are nonconvex refinement methods. The full LPV sub-Markov tensors used in this paper can be seen as generalizations of the tensors presented before in [18], however the decomposition and multilinear low-rank property are completely novel. The generalization details are provided later in this paper. Also, the proposed method is a convex method.

Before presenting the novel method, first essential background material is provided in Section II. Afterwards the LPV sub-Markov parameter tensors are presented explicitly in Section III and proven to be of multilinear low rank. Finally the novel method is presented in Section IV and a comparison with the regularized LPV-PBSID<sub>opt</sub> method of [37] through simulation results are presented in Section V.

## II. BACKGROUND

Prior to presenting the novel method, several related topics are reviewed in this section. These topics are LPV subspace identification, tensor decompositions, and matrix and tensor nuclear norms.

### A. First Regression Step of LPV Subspace Identification

In this section, a short review of the first regression step of LPV subspace identification [17], [37] is presented, because the exponential growth of parameters appears in this step. First define the following signals:

$$x_k \in \mathbb{R}^{\hat{n}}, u_k \in \mathbb{R}^r, y_k \in \mathbb{R}^l, e_k \in \mathbb{R}^l, \mu_k \in \mathbb{R}^m \quad (1)$$

as the state, input, output innovation and scheduling sequence signals at time instance  $k$ . The number  $\hat{n}$  is the system order. In this paper, it is assumed that  $\mu_k$  is known and the LPV system is affine. That is,  $\mu_k^{(1)} = 1$ .

We assume the following predictor-based LPV system as the data-generating system, such as in [5], [17], [37]:

$$x_{k+1} = \sum_{i=1}^m \mu_k^{(i)} \left( \tilde{A}^{(i)} x_k + \tilde{B}^{(i)} \begin{bmatrix} u_k \\ y_k \end{bmatrix} \right) \quad (2a)$$

$$y_k = C x_k + e_k \quad (2b)$$

where  $\tilde{A}^{(i)} \in \mathbb{R}^{\hat{n} \times \hat{n}}$  is  $A^{(i)} - K^{(i)}C$  and  $\tilde{B}^{(i)}$  is  $[B^{(i)}, K^{(i)}]$ , the variables  $A^{(i)}$ ,  $B^{(i)}$ , and  $C$  are the appropriately dimensioned state-space matrices. Notice that an LTI output equation with no direct feed-through (zero output matrix  $\tilde{D}$ ) is considered for sake of presentation and simplicity of derivation, similar to [37]. This will not trivialize the ‘‘curse-of-dimensionality’’. This representation allows predictor-based methods to deal with closed-loop data. Notice that the states here are the observer states. Define the discrete-time time-varying state transition matrix [37] as follows:

$$\phi_{j,k} = \tilde{A}_{k+j-1} \dots \tilde{A}_{k+1} \tilde{A}_k, \quad \tilde{A}_k = \sum_{i=1}^m \mu_k^{(i)} \tilde{A}^{(i)}. \quad (3)$$

The key approximation of predictor-based methods is that we assume that the state transition matrix  $\phi_{j,k} \approx 0 \forall j \geq p$  [5], [37]. This approximation is also used in several LTI methods [35]. If the predictor-based system is uniformly exponentially stable, then the approximation error can be made arbitrarily small by increasing  $p$  [22]. The introduced bias disappears as  $p$  goes to infinity, but is hard to quantify it for finite  $p$  [5]. The integer  $p$  is also known as the past window. Under this assumption, the effect of  $x_{k-p}$  on  $y_k$  is negligible. Then the outputs can be described using past inputs and outputs without the states as follows:

$$y_{k+p} \approx CK^p \mathcal{Z}_{k+p} + e_{k+p} \quad (4)$$

where  $CK^p$  contains the (scalar) LPV sub-Markov parameters, and  $\mathcal{Z}_{k+p}$  contains the effective data with past inputs and outputs [37]. Define

$$CK^p = C \begin{bmatrix} \mathcal{L}_p & \dots & \mathcal{L}_1 \end{bmatrix} \quad (5a)$$

$$\mathcal{L}_1 = \begin{bmatrix} \tilde{B}^{(1)} & \dots & \tilde{B}^{(m)} \end{bmatrix} \quad (5b)$$

$$\mathcal{L}_{i+1} = \begin{bmatrix} \tilde{A}^{(1)} \mathcal{L}_i & \dots & \tilde{A}^{(m)} \mathcal{L}_i \end{bmatrix}. \quad (5c)$$

The number of LPV sub-Markov parameters  $q$ , or number of elements (or entries) of  $CK^p$ , scales polynomially with  $m$  and exponentially with  $p$

$$q = l(l+r) \sum_{j=1}^p m^j. \quad (6)$$

The large size of  $CK^p$  is inherent to LPV predictor-based methods and requires special care. It can result in ill-conditioned problems, memory and computational cost issues. In this paper, we focus on the ill-conditioning and continue to deal with the latter two issues in [16] using dual problems [37].

The ‘‘effective data matrix’’  $\mathcal{Z}_k$  (4) is constructed from the input-output data and the scheduling sequence

$$\mathcal{Z}_k = N_{k-p}^p \mathcal{Z}_k \quad (7)$$

where  $\mathcal{Z}_k$  only contains input-output data

$$\mathcal{Z}_k = \begin{bmatrix} z_{k-p} \\ \vdots \\ z_{k-1} \end{bmatrix}, \quad z_k = \begin{bmatrix} u_k \\ y_k \end{bmatrix} \quad (8)$$

and  $N_{k-p}^p$  is a function of the scheduling sequence

$$N_k^p = \begin{bmatrix} \tilde{P}_{p|k} & 0 & \dots & 0 \\ 0 & \tilde{P}_{p-1|k+1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \tilde{P}_{1|k+p-1} \end{bmatrix} \quad (9a)$$

$$\tilde{P}_{p|k} = \mu_{k+p-1} \otimes \dots \otimes \mu_k \otimes I_{r+l} \quad (9b)$$

where the operator  $\otimes$  is the Kronecker product [3]. Define the data matrix and output matrix with all samples as

$$\mathcal{Z} = \begin{bmatrix} \mathcal{Z}_{p+1} & \dots & \mathcal{Z}_N \end{bmatrix} \quad (10a)$$

$$Y = \begin{bmatrix} y_{p+1} & \dots & y_N \end{bmatrix}. \quad (10b)$$

With these results, an estimate of the LPV state-space matrices can be obtained as follows. First the matrix  $CK^p$  is estimated in the first regression step. For example, this can be done by solving

$$\min_{CK^p} \|Y - CK^p \mathcal{Z}\|_F^2. \quad (11)$$

In this paper, we are mainly interested in the first regression step, because  $CK^p$  can be large. Afterwards, this estimate is used to form a rank-revealing matrix. This allows choosing a model order with the assistance of a singular value decomposition (SVD). Then the state sequence is reconstructed. With this state sequence the state-space matrices can be readily estimated, see for example [37].

In the following section, we introduce several tensor-related definitions.

## B. General Tensor-Related Definitions

In this section, some general tensor-related definitions are presented. These definitions are needed in the next section in order to present a tensor perspective on the LPV subspace identification problem.

We formally define a tensor and its sizes as follows:

*Definition 2.1:* Consider a tensor  $\mathbf{T}$ . Let  $D$  be the number of dimensions. Denote the size of the  $i$ -th dimension as  $J_i$ . Then the tensor is in  $\mathbb{R}^{J_1 \times \dots \times J_D}$ . Or in other words, the size of the tensor is  $J_1$ -by- $\dots$ -by- $J_D$  and contains real values. In this paper, we use bold upper case characters to denote tensors.

We define how we access entries from vectors and matrices:

*Definition 2.2:* Consider a matrix  $M$  and a vector  $v$ . Define  $[M]_{i,j}$  as the entry of  $M$  at row  $i$  and column  $j$ . Let  $[M]_{:,j}$  and  $[M]_{i,:}$  respectively be the  $j$ th column and  $i$ th row vector. For a two-by-two matrix  $M$  this means

$$M = \begin{bmatrix} [M]_{1,1} & [M]_{1,2} \\ [M]_{2,1} & [M]_{2,2} \end{bmatrix} = \begin{bmatrix} [M]_{1,:} \\ [M]_{2,:} \end{bmatrix} = \begin{bmatrix} [M]_{:,1} & [M]_{:,2} \end{bmatrix} \quad (12)$$

For row and column vectors, define  $[v]_i$  as the  $i$ th entry of  $v$ .

In the following section, tensor nuclear norms are reviewed.

## C. Matrix Nuclear Norm and the Tensor Nuclear Norm

In this section the matrix nuclear norm [26] and the tensor nuclear norm [31] are reviewed. They can be used to exploit a priori knowledge on low-rank properties of matrices and matricizations of tensors. These norms will play a key role in the method proposed in this paper.

First we review the matrix nuclear norm. The matrix nuclear norm has received considerable attention in the literature [26], [39], [42]. It can be used in a regularization term in order to exploit low-rank properties in a convex manner [26]. The matrix nuclear norm of a matrix is the largest convex lower bound of the rank of that matrix [26] as follows:

$$\|M\|_* \leq \text{rank}(M), \quad \|M\|_2 \leq 1 \quad (13)$$

where  $\|M\|_*$  is the nuclear norm of  $M$ , and  $M$  has been normalized. The matrix nuclear norm itself is defined as the sum of singular values.

Next we review the tensor nuclear norm. The tensor nuclear norm of a tensor relates to the multilinear rank of a tensor [31]. The multilinear rank of a tensor is a tuple of numbers. Each number is the matrix rank of a different matricization of the tensor. We formally define the  $n$ -mode matricization of a tensor as follows:

*Definition 2.3:* Consider a tensor  $\mathbf{T}$  (Definition 2.1) of size  $\mathbb{R}^{J_1 \times J_2 \times \dots \times J_D}$ . This tensor is of  $D$ th order. Let  $n$  be an integer  $\in \{1, \dots, D\}$  and represent the mode number at hand. Then the  $n$ -mode matricization of  $\mathbf{T}$  is denoted  $\mathbf{T}_{\langle n \rangle}$  and can be constructed as follows. First rearrange the dimensions of the tensor in the ordering  $[n, n+1, \dots, D, 1, 2, \dots, n-1]$ . The entry of  $\mathbf{T}$  at position  $(i_1, i_2, \dots, i_D)$  is now put on position  $(i_n, i_{n+1}, \dots, i_{n-1})$ . Then reshape the result into a matrix with  $J_n$  rows. The resulting matrix is the  $n$ -mode matricization or tensor unfolding (with forward cycling). This  $n$ -mode matricization can be transformed back into the original tensor by performing the two operations in reverse.

Define also the  $n$ -mode product:

*Definition 2.4:* The  $n$ -mode product of a tensor  $\mathbf{T}$  with a matrix  $M$  is  $\mathbf{T} \bullet_n M$  and can be computed using

$$[\mathbf{T} \bullet_n M]_{\langle n \rangle} = M \mathbf{T}_{\langle n \rangle} \quad (14)$$

where the right hand side requires a matricization and a matrix multiplication. Returning to  $\mathbf{T} \bullet_n M$  can be done using Definition 2.3. This

and other tensor operations can also be computed using the TensorLab toolbox [40].

Also define

*Definition 2.5:* The  $n$ -rank of a tensor is defined as the rank of the  $n$ -mode matricization of that tensor:

$$\text{rank}_n(\mathbf{T}) = \text{rank}(\mathbf{T}_{\langle n \rangle}) \quad (15)$$

Now it is possible to define the multilinear rank explicitly:

*Definition 2.6:* The multilinear rank of a tensor is the  $D$ -tuple of all  $n$ -ranks of that tensor.

The multilinear rank notion is computationally attractive because it is a tuple of matrix ranks. For contrast, the polyadic rank is NP-hard to determine [19]. This multilinear rank relates to the tensor nuclear norm as defined in [31]. In this paper, we use the definition of [31] and not the different one of [11], because the former is computationally much cheaper to compute. Both have received considerable attention in the literature [11], [20], [38]. The tensor nuclear norm is defined as

$$\|\mathbf{T}\|_* = \frac{1}{D} \sum_{n=1}^D \|\mathbf{T}_{\langle n \rangle}\|_* \quad (16)$$

and has the property

$$\|\mathbf{T}\|_* \leq \frac{1}{D} \sum_{n=1}^D \text{rank}_n(\mathbf{T}), \quad \|\mathbf{T}_{\langle n \rangle}\|_2 \leq 1 \quad \forall n \quad (17)$$

however it is not proven to be the best convex heuristic. Just like the matrix nuclear norm, the tensor nuclear norm can be used as a regularization term.

In the following section, the multilinear SVD of a tensor is reviewed.

## D. Multi-Linear Singular Value Decomposition (MLSVD)

In this section, we review the SVD for matrices and the MLSVD for tensors, to be able to prove the multilinear low-rank property of the parameter tensors in the next section.

First we present the more simple matrix SVD for illustration. The SVD [15] of a real matrix can be seen as a decomposition with special properties

$$M = U \Sigma V^T \quad (18)$$

where  $U$  and  $V$  are unitary and  $\Sigma$  contains the (nonnegative) singular values of  $M$  along its diagonal in descending order. Notice that the matrix  $M$  is low rank if and only if some singular values are zero. Suppose only  $\bar{r}$  singular values are nonzero. Then this allows truncating without error, to

$$M = U \Sigma V^T = \bar{U} \bar{\Sigma} \bar{V}^T \quad (19)$$

where  $\bar{U}$  and  $\bar{V}$  are the first  $\bar{r}$  columns of  $U$  and  $V$  and  $\bar{\Sigma}$  is the top-left  $\bar{r}$ -by- $\bar{r}$  sub-matrix of  $\Sigma$ . Notice that  $\bar{\Sigma}$  is smaller than  $M$  for low rank  $M$ .

The MLSVD [8] can be seen as an extension of the SVD to tensors. It decomposes a tensor as follows:

$$\mathbf{T} = \mathcal{S} \bullet_1 U^{(1)} \bullet_2 U^{(2)} \bullet_3 \dots \bullet_D U^{(D)} \quad (20)$$

where the  $n$ -mode product  $\bullet_n$  is defined in Definition 2.4,  $\mathcal{S}$  is the all-orthogonal core tensor with ordered multilinear singular values and the matrices  $U^{(*)}$  have orthonormal columns. The conditions for multilinear low rankness of a tensor are

*Definition 2.7:* A tensor is multilinear low rank if and only if all its  $n$ -mode matricizations are low rank.

or alternatively

*Definition 2.8:* A tensor is multilinear low rank if the core tensor of its MLSVD (20) is smaller than the full tensor in all dimensions. More specifically, the multilinear rank of a tensor is the minimal size of its core tensor.

In fact, there is no need for an MLSVD to prove multilinear low rankness. A *multilinear decomposition* is sufficient to prove some tensor to be multilinear low rank. This decomposition is defined as follows:

$$\mathbf{T} = \mathcal{A} \bullet_1 M^{(1)} \bullet_2 M^{(2)} \bullet_3 \cdots \bullet_D M^{(D)} \quad (21)$$

where  $\mathcal{A}$  is a  $D$ th order tensor and  $M$  are matrices with corresponding sizes. With this decomposition the multilinear rank of the tensor is entrywise bounded from above by the size of  $\mathcal{A}$ . This is because  $\mathcal{A}$  and  $M^{(*)}$  can be normalized to obtain an MLSVD with the same sizes [40].

In the following section, it will be proven that the parameter tensor is multilinear low rank.

### III. MULTILINEAR LOW-RANK PARAMETER TENSORS

In this section, we show that the LPV sub-Markov parameters can be organized into several multilinear low-rank tensors. That is, all their matricizations are low rank. This property is proven using the multilinear decompositions discussed in Section II-D. This low-rank property will be exploited by the novel method in the next section.

First we present a three-dimensional example to illustrate the parameter tensors for a simple case. Consider an LPV state-space system (2) with  $m = 2$ . Let this system be single-input single-output output-error with LTI input matrix. Furthermore we choose the size of the parameter tensor to be  $3 \times 3 \times 3$ , such that it contains all LPV sub-Markov parameters to be estimated for  $p = 4$ . Then, the parameter tensor for this example system is as follows:

$$\begin{array}{|c|c|c|} \hline & C\tilde{A}^{(2)}B & \cdots & C\tilde{A}^{(2)}\tilde{A}^{(2)}B' \\ \hline & C\tilde{A}^{(1)}\tilde{A}^{(2)}B & \cdots & C\tilde{A}^{(1)}\tilde{A}^{(2)}\tilde{A}^{(2)}B \\ \hline & C\tilde{A}^{(2)}\tilde{A}^{(2)}B & \cdots & C\tilde{A}^{(2)}\tilde{A}^{(2)}\tilde{A}^{(2)}B \\ \hline & C\tilde{A}^{(1)}B & \cdots & C\tilde{A}^{(2)}\tilde{A}^{(1)}B \\ \hline & C\tilde{A}^{(1)}\tilde{A}^{(1)}B & \cdots & C\tilde{A}^{(1)}\tilde{A}^{(2)}\tilde{A}^{(1)}B \\ \hline & C\tilde{A}^{(2)}\tilde{A}^{(1)}B & \cdots & C\tilde{A}^{(2)}\tilde{A}^{(2)}\tilde{A}^{(1)}B \\ \hline CB & C\tilde{A}^{(1)}B & C\tilde{A}^{(2)}B \\ \hline C\tilde{A}^{(1)}B & C\tilde{A}^{(1)}\tilde{A}^{(1)}B & C\tilde{A}^{(1)}\tilde{A}^{(2)}B \\ \hline C\tilde{A}^{(2)}B & C\tilde{A}^{(2)}\tilde{A}^{(1)}B & C\tilde{A}^{(2)}\tilde{A}^{(2)}B \\ \hline \end{array}$$

Notice that this tensor contains the LPV sub-Markov parameters and has some similarities with Hankel matrices. For example, the front slice is exactly a Hankel matrix. Also, the LTI variant would be exactly a Hankel tensor. More specifically, this tensor has some block-Hankel tensor structure. Later in this section we will show that we can make statements about its multilinear ranks.

Next we present the parameter tensors in multilinear decomposition form for the general case. This decomposition will also prove (an upper bound on) the multilinear rank (Section II-D). The parameter tensors are denoted  $\mathbf{H}_{o,\kappa}$  and are defined per output  $o$  and column of  $\tilde{B}(\kappa)$ . The dimensions of a parameter tensor are all equal. Let the scalar  $t$  be this dimension (value), and  $D$  the number of dimensions. Both variables are user-chosen and will be defined explicitly later in this section. The relation of the two variables with the past window is discussed at the

end of this section. First we present their multilinear decomposition

$$\mathbf{H}_{o,\kappa} = \mathcal{A}_H \bullet_1 F_o \bullet_2 F \bullet_3 \cdots \bullet_{D-1} F \bullet_D F_\kappa \in \mathbb{R}^{t \times \cdots \times t} \quad (22)$$

where the tensor  $\mathcal{A}_H$  and matrices  $F_o$ ,  $F$  and  $F_\kappa$  will be defined next. The matrix  $F_o$  contains the output matrix and products of  $\tilde{A}^{(*)}$  and is

$$F_o = \begin{bmatrix} [C]_{o,:} \\ [C]_{o,:}\tilde{A}^{(1)} \\ \vdots \\ [C]_{o,:}\tilde{A}^{(m)} \\ [C]_{o,:}\tilde{A}^{(1)}\tilde{A}^{(1)} \\ [C]_{o,:}\tilde{A}^{(2)}\tilde{A}^{(1)} \\ \vdots \\ [C]_{o,:}\tilde{A}^{(m)}\tilde{A}^{(m)} \\ \vdots \end{bmatrix} \in \mathbb{R}^{t \times \hat{n}} \quad (23)$$

where we let  $t$  be the height of this matrix, and the brackets with subscript are defined in Definition 2.2. Notice that  $t$  is (by definition) also the size of the first dimension of  $\mathbf{H}_{o,\kappa}$ . The matrix  $F$  contains products of  $\tilde{A}^{(*)}$  in a vectorized form as follows:

$$F = \begin{bmatrix} \text{vec}(I_{\hat{n}})^T \\ \text{vec}(\tilde{A}^{(1)})^T \\ \vdots \\ \text{vec}(\tilde{A}^{(m)})^T \\ \text{vec}(\tilde{A}^{(1)}\tilde{A}^{(1)})^T \\ \text{vec}(\tilde{A}^{(2)}\tilde{A}^{(1)})^T \\ \vdots \\ \text{vec}(\tilde{A}^{(m)}\tilde{A}^{(m)})^T \\ \vdots \end{bmatrix} \in \mathbb{R}^{t \times \hat{n}^2}. \quad (24)$$

Notice that the width of  $F$  is  $\hat{n}^2$ , but this will be dealt with in the definition of  $\mathcal{A}_H$ . The matrix  $F_\kappa$  contains the matrix  $\tilde{B}$  and is defined as:

$$F_\kappa = \begin{bmatrix} ([\tilde{B}]_{:, \kappa})^T \\ (\tilde{A}^{(1)}[\tilde{B}]_{:, \kappa})^T \\ \vdots \\ (\tilde{A}^{(m)}[\tilde{B}]_{:, \kappa})^T \\ (\tilde{A}^{(1)}\tilde{A}^{(1)}[\tilde{B}]_{:, \kappa})^T \\ (\tilde{A}^{(2)}\tilde{A}^{(1)}[\tilde{B}]_{:, \kappa})^T \\ \vdots \\ (\tilde{A}^{(m)}\tilde{A}^{(m)}[\tilde{B}]_{:, \kappa})^T \\ \vdots \end{bmatrix} \in \mathbb{R}^{t \times \hat{n}}. \quad (25)$$

These three matrices and the tensor  $\mathcal{A}_H$  together form the LPV sub-Markov parameters. This tensor  $\mathcal{A}_H$  consists purely of ones and zeros,

and discards all inadmissible products between elements of the matrices  $F_*$ . These inadmissible products appear because instead of matrix products we have products of the vectorizations of those matrices. Therefore, this tensor is named here as the admissibility tensor. The algorithm to generate  $\mathcal{A}_H$  as a function of  $\hat{n}$  and  $D$  is given in [16]. Proof of the decomposition follows through straightforward computations. The size of this tensor  $\mathcal{A}_H$  matches with the widths of  $F_*$  as shown in (22) and is  $[\hat{n} \hat{n}^2 \hat{n}^2 \dots \hat{n}^2 \hat{n}]$ . Using the definition of the multilinear decomposition (Section II-D), we can state that this size is the upper bound of the multilinear rank of  $\mathbf{H}_{o,\kappa}$ .

Now that we know the multilinear rank of  $\mathbf{H}_{o,\kappa}$ , we can state under which conditions it is multilinear low rank. Using Definition 2.8, this is the case if  $\mathcal{A}_H$  is smaller than  $\mathbf{H}_{o,\kappa}$  in all dimensions. Filling in their sizes gives

*Theorem 3.1:* The parameter tensor  $\mathbf{H}_{o,\kappa}$  is multilinear low rank if  $t > \hat{n}^2$ .

Since  $t$  is user-chosen, we will implicitly assume throughout this paper that  $t > \hat{n}^2$ .

In this paragraph, we give details on the size of  $\mathbf{H}_{o,\kappa}$  for completeness. The dimension  $t$  is

$$t = \sum_{j=0}^{\bar{f}-1} m^j \quad (26)$$

where  $\bar{f}$  is named the incremental window. The maximum number of  $\hat{A}^{(*)}$  in sequence in  $F_*$  is exactly  $\bar{f} - 1$ . It is interesting to note that for fixed  $\bar{f} = 2$ , the full tensors are similar to the ones in our previous work [18]. Furthermore, the number of dimensions  $D$  of  $\mathbf{H}_{o,\kappa}$  is also user-chosen. Together they determine the size of  $\mathbf{H}_{o,\kappa}$ . For purpose of simple notation, we omit  $D$  and  $\bar{f}$  from the notation of  $\mathbf{H}_{o,\kappa}$  and other affected variables. These two also determine which LPV sub-Markov parameters will appear in  $\mathbf{H}_{o,\kappa}$ . Let the “order” of an LPV sub-Markov parameter be the number of  $\hat{A}^{(*)}$  in sequence that it has plus one. Then the highest order of LPV sub-Markov parameters in  $\mathbf{H}_{o,\kappa}$  is

$$h = 1 + (\bar{f} - 1)D \quad (27)$$

where  $h$  is named the regularization window. A natural choice would be  $h = p$ . However, we keep the option of choosing  $h$  open. In the remainder of this paper we let  $h \leq p$ . The availability of the choice of  $h$  will allow reducing computational load where needed.

We would like to remark that for  $\bar{f} = 2$  the full tensor equals the tensor presented before in [18]. The decomposition and multilinear low-rank property are completely novel.

In the following section, we present the novel method.

#### IV. PREDICTOR-BASED TENSOR NUCLEAR NORM REGRESSION (PBTNNR)

Using the results from the previous sections we now have all the ingredients to derive the new Predictor-Based Tensor Nuclear Norm Regression method algorithm. The proposed convex LPV subspace identification method exploits the multilinear low-rank property of the parameter tensor for improved problem conditioning.

The difference between the proposed method and the method of [37] and [12] is that they use different regularization terms during the critical, first regression step. Therefore we only present the first regression step in this section and refer to Section II-A for details on subsequent steps. The starting point is the objective function (11)

$$\min_{\theta} \|Y - [CK^p](\theta)\mathcal{Z}\|_F^2 \quad (28)$$

where  $CK^p$  and  $\mathcal{Z}$  are defined and motivated in Section II-A. The proposed method adds tensor nuclear norm regularization terms to this objective function.

The arguments of the tensor nuclear norms are the multilinear low-rank parameter tensors (22). Every scalar LPV sub-Markov parameter is parametrized by one optimization parameter. This applies to both  $[CK^p](\theta)$  and the parameter tensors  $\mathbf{H}_{o,\kappa}(\theta)$ . This also implies that some optimization parameters ( $\theta_i$ ) appear multiple times in  $\mathbf{H}_{o,\kappa}(\theta)$ . For simplicity, all regularization terms are weighed in the objective function by a single tuning parameter  $\lambda$ . Furthermore, we let  $h = p$  (27) for ease of notation. This results in the following objective function

$$\min_{\theta} \|Y - [CK^p](\theta)\mathcal{Z}\|_F^2 + \lambda \sum_{o=1}^l \sum_{\kappa=1}^{m(l+\tau)} \|\mathbf{H}_{o,\kappa}(\theta)\|_* \quad (29)$$

which exploits the multilinear low-rank property of the parameter tensors to improve problem conditioning.

In the following section, the simulation results are presented.

## V. SIMULATION RESULTS

In this section simulation results are presented in order to compare the proposed method PBTNNR with the method of [37] in terms of variance accounted for.

### A. Simulation Settings

In this section, the simulation settings and some related definitions are presented.

For statistical significance, the results presented in this paper are based on 100 Monte Carlo simulations. For every Monte Carlo simulation a different realization of both the input and the innovation vector is used. The scheduling sequence is kept the same. All methods use a model order equal to the system order, and are supplied the information that  $\bar{D} = 0$  and that the output equation is LTI. For completeness, the future window variable for the SVD step is chosen equal to the past window.

In this section, we compare the novel method with the (Tikhonov) regularized LPV-PBSID<sub>opt</sub> (kernel) method of [37]. We use the kernel variant of the proposed method presented in [16] to perform simulations. The quality of the estimates is evaluated by investigating the Variance Accounted For (VAF) on a validation data set which is *different* from the one used for identification, in the sense that different realizations of both the input and the innovation vector are used. The VAF for single-output systems is defined as follows [37]:

$$\text{VAF}(\bar{y}_k, \hat{y}_k) = \max \left\{ 1 - \frac{\text{var}(\bar{y}_k - \hat{y}_k)}{\text{var}(\bar{y}_k)}, 0 \right\} 100\%.$$

The noise-free simulated output of the system is used when evaluating the VAF, because this allows the VAF to reach 100% when the model is equal to the true system modulo global state-coordinate transformations. Notice that this is possible because the data is generated using simulations. The noise-free simulated output of the system is here denoted as  $\bar{y}_k$ . In similar sense, the noise-free (simulated) model output is denoted as  $\hat{y}_k$ . The operator  $\text{var}(\ast)$  denotes the variance of its argument. The tuning parameter  $\lambda$  of the proposed method has been chosen as follows. The  $\lambda$  with single significant digit which yields the highest mean validation VAF is chosen. The computations are performed on an Intel i7 quad-core processor running at 2.7 GHz with 8 GB RAM. We provide the computation time both of the proposed method and the method of [37].

Several cases and their results are presented in the following sections.

TABLE I  
MEAN VAF FOR DIFFERENT METHODS FOR CASE 1

Method	VAF
LPV-PBSID <sub>opt</sub> (kernel)	96.6
PBTNNR (kernel, $\bar{f} = 2, D = 2$ )	97.9 with $\lambda = 0.1$
PBTR (refinement method)	98.0

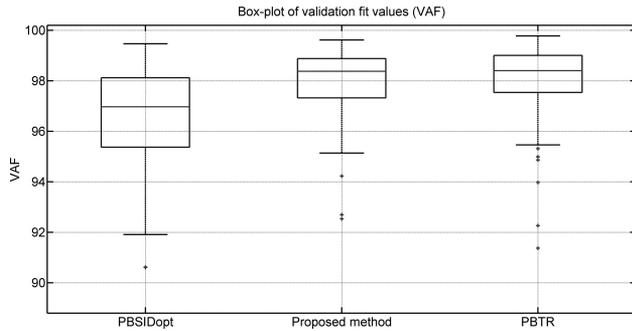


Fig. 1. Box-plot of the validation VAF of the 100 Monte Carlo simulations for three methods. Only the first two methods are convex; the last method is a nonconvex refinement method.

### B. Simulation Results Case 1

In this section, simulation results are presented for a case of [17] in order to compare the proposed method with both the method of [37] and [17] (PBTR).

This case uses the following LPV state-space system (2):

$$[A^{(1)}, A^{(2)}] = \left[ \begin{array}{cc|cc} \frac{4}{15} & \frac{1}{15} & \frac{3}{20} & -\frac{1}{60} \\ -\frac{1}{6} & \frac{1}{30} & -\frac{1}{60} & \frac{3}{20} \end{array} \right]$$

$$[B^{(1)}, B^{(2)}] = \left[ \begin{array}{c|c} 1 & 0.2 \\ 0 & 0.2 \end{array} \right], C = [1 \quad 0]$$

and  $D$  is zero and  $K$  is LPV. The matrix  $K^{(i)}$  for  $i = \{1, \dots, m\}$  is obtained from the Discrete Algebraic Riccati Equation (DARE) with  $A^{(i)}$ ,  $C$  and identity covariance of the concatenated process and measurement noise. Both the input vector  $u_k$  and the innovation vector  $e_k$  are white noise with unit power. The data size  $N$  is chosen as 200. Both methods are run with past window  $p$  equal to 6.

The system is evaluated at the scheduling sequence

$$\mu_k^{(2)} = \cos\left(2\pi k \frac{20}{N}\right) / 2 + 0.2.$$

We present the VAF of the proposed method, the method of [37] and the nonconvex refinement method of [17] (PBTR) in Table I, and a box-plot of the VAF in Fig. 1. We present the computation times for the convex methods. The average computation times are 6.6 seconds for the proposed method and 38 milliseconds for the method of [37] per simulation.

From Table I and Fig. 1 it can be concluded that for this case the proposed method has higher performance than the method of [37] in terms of VAF. Additionally, the nonconvex refinement method of [17] has a higher VAF than the two convex methods.

### C. Simulation Results Case 2

In this section, simulation results are presented for a larger past window with a case which relates to the flapping dynamics of a wind turbine. This case has been used before in [10], [37].

TABLE II  
MEAN VAF FOR DIFFERENT METHODS FOR CASE 2

Method	VAF
LPV-PBSID <sub>opt</sub> (kernel)	97.0
PBTNNR (kernel, $\bar{f} = 3, D = 3$ )	98.1 with $\lambda = 0.01$

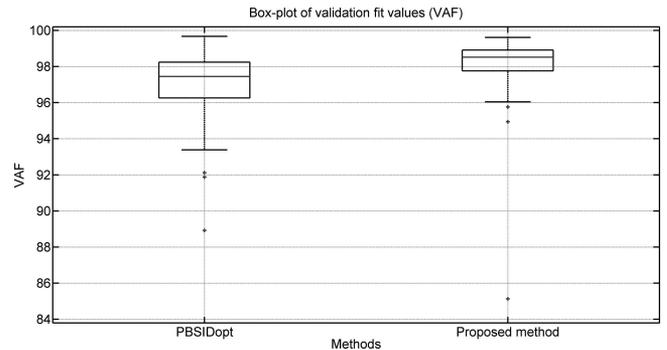


Fig. 2. Box-plot of the validation VAF of the 100 Monte Carlo simulations for two methods.

This case uses the following LPV state-space system (2):

$$[A^{(1)}, A^{(2)}] = \left[ \begin{array}{ccc|c} 0 & 0.0734 & -0.0021 & 0 \\ -6.5229 & -0.4997 & -0.0138 & 0.5196 \end{array} \right]$$

$$[B^{(1)}, B^{(2)}] = \left[ \begin{array}{c|c} -0.7221 & 0 \\ -9.6277 & 0 \end{array} \right], C = [1 \quad 0]$$

and  $D$  is zero and  $K$  is LPV. The matrix  $K^{(i)}$  for  $i = \{1, \dots, m\}$  is obtained from the DARE with  $A^{(i)}$ ,  $C$  and identity covariance of the concatenated process and measurement noise. The input vector  $u_k$  is white noise with unit power, the innovation vector  $e_k$  is also white noise and the signal-to-noise ratio is 40 dB. The data size  $N$  is chosen as 100. Both methods are run with past window  $p$  equal to 15.

The system is evaluated at the scheduling sequence

$$\mu_k^{(2)} = \cos\left(2\pi k \frac{20}{N}\right) / 2 + 0.2.$$

We present the VAF of both the novel method and the method of [37] in Table II, and a box-plot of the VAF in Fig. 2. The average computation times are 23 seconds for the proposed method and 50 milliseconds for the method of [37] per Monte Carlo simulation.

From Table II and Fig. 2 it can be concluded that for this case the proposed method has higher performance than the method of [37] in terms of VAF.

The conclusions are presented in the following section.

## VI. CONCLUSION

In this paper, we showed that the LPV sub-Markov parameters can be organized into several multilinear low-rank tensors. Then we proposed exploiting this property using a novel tensor nuclear norm regularization term. This resulted in a novel convex LPV subspace identification method. Finally, simulation results showed that the novel method can have higher performance than the regularized LPV-PBSID<sub>opt</sub> technique in terms of variance accounted for.

## REFERENCES

- [1] G. J. Balas, "Linear, parameter-varying control and its application to aerospace systems," in *Proc. Int. Council Aeronaut. Sci. Congr.*, Vol. 12, no. 9, pp. 763–796, Jul. 2002.
- [2] F. D. Bianchi, R. J. Mantz, and C. F. Christiansen, "Gain scheduling control of variable-speed wind energy conversion systems using quasi-LPV models," *Control Eng. Pract.*, vol. 13, no. 2, pp. 247–255, 2005.
- [3] J. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Trans. Circuits Syst.*, vol. 25, no. 9, pp. 772–781, Sep. 1978.
- [4] M. Butcher, A. Karimi, and R. Longchamp, "On the consistency of certain identification methods for linear parameter varying systems," *IFAC Proc. Vol.*, vol. 41, no. 2, pp. 4018–4023, 2008.
- [5] A. Chiuso, "The role of vector autoregressive modeling in predictor-based subspace identification," *Automatica*, vol. 43, no. 6, pp. 1034–1048, 2007.
- [6] P. Cox and R. Tóth, "Alternative form of predictor based identification of LPV-SS models with innovation noise," in *Proc. IEEE 55th Conf. Decis. Control*, 2016, pp. 1223–1228.
- [7] J. D. Caigny, J. F. Camino, and J. Swevers, "Interpolating model identification for SISO linear parameter-varying systems," *Mech. Syst. Signal Process.*, vol. 23, no. 8, pp. 2395–2417, 2009.
- [8] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.
- [9] W. Favoreel, B. D. Moor, and P. V. Overschee, "Subspace identification of bilinear systems subject to white inputs," *IEEE Trans. Autom. Control*, vol. 44, no. 6, pp. 1157–1165, Jun. 1999.
- [10] F. Felici, J.-W. V. Wingerden, and M. Verhaegen, "Subspace identification of MIMO LPV systems using a periodic scheduling sequence," *Automatica*, vol. 43, no. 10, pp. 1684–1697, 2007.
- [11] S. Friedland and L.-H. Lim, "Computational complexity of tensor nuclear norm," arXiv:1410.6072, 2014.
- [12] P. M. Gebraad, J.-W. van Wingerden, P. A. Fleming, and A. D. Wright, "LPV subspace identification of the edgewise vibrational dynamics of a wind turbine rotor," in *Proc. IEEE Int. Conf. Control Appl.*, 2011, pp. 37–42.
- [13] P. M. Gebraad, J.-W. van Wingerden, G. J. van der Veen, and M. Verhaegen, "LPV subspace identification using a novel nuclear norm regularization method," in *Proc. Amer. Control Conf.*, 2011, pp. 165–170.
- [14] L. Giarré, D. Bauso, P. Falugi, and B. Bamieh, "LPV model identification for gain scheduling control: An application to rotating stall and surge control problem," *Control Eng. Pract.*, vol. 14, no. 4, pp. 351–361, 2006.
- [15] G. H. Golub, M. Heath, and G. Wahba, "Generalized cross-validation as a method for choosing a good ridge parameter," *Technometrics*, vol. 21, no. 2, pp. 215–223, 1979.
- [16] B. Gunes, J.-W. van Wingerden, and M. Verhaegen, "Kernel method for tensor nuclear norm LPV subspace identification," Intern. Rep., 2017. [Online]. Available: <https://drive.google.com/open?id=0BxAriINeXhF5UEIUX3BOYWotWkE>
- [17] B. Gunes, J.-W. van Wingerden, and M. Verhaegen, "Predictor-based tensor regression (PBTR) for LPV subspace identification," *Automatica*, vol. 79, pp. 235–243, 2017.
- [18] B. Gunes, J.-W. van Wingerden, and M. Verhaegen, "Tensor networks for MIMO LPV system identification," *Int. J. Control*, submitted to 2017.
- [19] J. Håstad, "Tensor rank is np-complete," *J. Algorithms*, vol. 11, no. 4, pp. 644–654, 1990.
- [20] X. He, D. Cai, and P. Niyogi, "Tensor subspace analysis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 499–506.
- [21] H. Hjalmarsson, J. S. Welsh, and C. R. Rojas, "Identification of box-jenkins models using structured ARX models and nuclear norm relaxation," *IFAC Proc. Vol.*, vol. 45, no. 16, pp. 322–327, 2012.
- [22] T. Knudsen, "Consistency analysis of subspace identification methods based on a linear regression approach," *Automatica*, vol. 37, no. 1, pp. 81–89, 2001.
- [23] W. E. Larimore and M. Buchholz, "ADAPT-LPV software for identification of nonlinear parameter-varying systems," *IFAC Proc. Vol.*, vol. 45, no. 16, pp. 1820–1825, 2012.
- [24] V. Laurain, M. Gilson, R. Tóth, and H. Garnier, "Refined instrumental variable methods for identification of LPV box-jenkins models," *Automatica*, vol. 46, no. 6, pp. 959–967, 2010.
- [25] D. J. Leith and W. E. Leithead, "Survey of gain-scheduling analysis and design," *Int. J. Control*, vol. 73, no. 11, pp. 1001–1025, 2000.
- [26] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *Soc. Ind. Appl. Math., Rev.*, vol. 52, no. 3, pp. 471–501, 2010.
- [27] J. Remmlinger, M. Buchholz, and K. Dietmayer, "Identification of a bilinear and parameter-varying model for lithium-ion batteries by subspace methods," in *Proc. IEEE Amer. Control Conf.*, 2013, pp. 2268–2273.
- [28] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, no. 10, pp. 1401–1425, 2000.
- [29] C. W. Scherer, "LPV control and full block multipliers," *Automatica*, vol. 37, no. 3, pp. 361–375, 2001.
- [30] J. S. Shamma, "An overview of LPV systems," in *Control of Linear Parameter Varying Systems with Applications*, New York, NY, USA: Springer, 2012, pp. 3–26.
- [31] M. Signoretto, L. D. Lathauwer, and J. A. K. Suykens, "Nuclear norms for tensors and their use for convex multilinear estimation," Tech. Rep. 10-186, ESAT-SISTA, KU Leuven, 2010.
- [32] R. Tóth, *Modeling and Identification of Linear Parameter-Varying Systems*, vol. 403, New York, NY, USA: Springer, 2010.
- [33] R. Tóth, H. S. Abbas, and H. Werner, "On the state-space realization of LPV input-output models: Practical approaches," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 1, pp. 139–153, Jan. 2012.
- [34] R. van der Maas, A. van der Maas, and T. Oomen, "Accurate frequency response function identification of LPV systems: A 2D local parametric modeling approach," in *Proc. IEEE 54th Annu. Conf. Decis. Control*, 2015, pp. 1465–1470.
- [35] G. van der Veen, J.-W. van Wingerden, M. Bergamasco, M. Lovera, and M. Verhaegen, "Closed-loop subspace identification methods: An overview," *IET Control Theory Appl.*, vol. 7, no. 10, pp. 1339–1358, Jul. 2013.
- [36] J.-W. van Wingerden, F. Felici, and M. Verhaegen, "Subspace identification of MIMO LPV systems using a piecewise constant scheduling sequence with hard/soft switching," in *Proc. IEEE Eur. Control Conf.*, 2007, pp. 927–934.
- [37] J.-W. van Wingerden and M. Verhaegen, "Subspace identification of bilinear and LPV systems for open-and closed-loop data," *Automatica*, vol. 45, no. 2, pp. 372–381, 2009.
- [38] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear analysis of image ensembles: Tensorfaces," in *Proc. Eur. Conf. Comput. Vis.*, 2002, pp. 447–460.
- [39] M. Verhaegen and A. Hansson, "Nuclear norm subspace identification (N2SID) for short data batches," vol. 47, no. 3, pp. 9528–9533, 2014.
- [40] N. Vervliet, O. Debals, L. Sorber, M. Van Barel, and L. D. Lathauwer, "Tensorlab 3.0," 2016. [Online]. Available URL: <http://www.tensorlab.net>
- [41] M. G. Wassink, M. van de Wal, C. Scherer, and O. Bosgra, "LPV control for a wafer stage: Beyond the theoretical solution," *Control Eng. Pract.*, vol. 13, no. 2, pp. 231–245, 2005.
- [42] C. Yu and M. Verhaegen, "Local subspace identification of distributed homogeneous systems with general interconnection patterns," *Proc. 13th IFAC Symp. Syst. Identif., Beijing*, 2015, pp. 585–589.