K4SID: Large-Scale Subspace Identification With Kronecker Modeling

Baptiste Sinquin [ID] and Michel Verhaegen [ID]

Abstract—In this paper, we consider the identification of matrix state-space models (MSSM) of the following form:

$$\mathbf{X}(k + 1) = \mathbf{A_2}\mathbf{X}(k)\mathbf{A_1}^T + \mathbf{B_2}\mathbf{U}(k)\mathbf{B_1}^T$$

$$\mathbf{Y}(k) = \mathbf{C_2}\mathbf{X}(k)\mathbf{C_1}^T + \mathbf{E}(k)$$

for all time dependent quantities and matrices of appropriate dimensions. Due to the large size of these matrices, vectorization does not allow the use of standard multivariable subspace methods such as N4SID or MOESP. In this paper, the resulting Kronecker structure that appears in the system matrices due to vectorization is exploited for developing a scalable subspace-like identification approach. This approach consists of first estimating the Markov parameters associated to the MSSM via the solution of a regularized bilinear least-squares problem that is solved in a globally convergent manner. Second, a bilinear low-rank minimization problem is tackled which allows to write a three-dimensional low-rank tensor and consequently to estimate the state sequence and the lower-dimensional matrices $\mathbf{A_1}, \mathbf{A_2}, \mathbf{B_1}, \mathbf{B_2}, \mathbf{C_1}, \mathbf{C_2}$. A numerical example on a large-scale adaptive optics system demonstrates the ability of the algorithm to handle the identification of state-space models within the class of Kronecker structured matrices in a scalable manner which results in more compact models.

Index Terms—Bilinear optimization, kronecker product, large-scale networks, spatial-temporal modeling, subspace identification.

I. INTRODUCTION

THE identification of multidimensional systems with a large number of input-output data has started receiving interest from the image processing community. The first attempts to model two-dimensional (2-D) systems in state-space form happened in 1975 with the Roesser model, [1], [2]. Although the time dimension is left out, the state-space representation models the spatial dynamics by introducing horizontal and vertical states that are coupled together through some unknown interconnection. Assuming the denominator of the transfer function

to be separable, also known as causal recursive separable in denominator assumption, has helped to identify the deterministic 2-D [3], the stochastic 2-D [4] and the deterministic-stochastic [5] systems.

When, the time dimension is considered along with the spatial coupling between the nodes in the network, it gives rise to a $(2+1)$D model. Examples of applications for such spatial-temporal systems are numerous in control for high resolution imaging, e.g., adaptive optics [6]. The large number of actuators to be controlled and sensor channels to be processed at kilohertz rate calls for scalable algorithms that enable real-time control. Although different structures of networks have been investigated to cope with different spatial interconnection patterns [8], e.g., sparse, circulant or decomposable to name but a few, in this work, the focus is laid on 2-D spatial-temporal networks with actuator-sensor nodes regularly located on a grid.

More precisely, let a 2-D network be defined from a grid of $N \times N$ nodes. Each node corresponds to a local subsystem of order n with m inputs and p outputs. We assume $p, m < n \ll N$. The subsystems influence each other through some in general unknown interconnection. The temporal dynamics of the network as a whole are modeled with the state-space representation

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \\ \mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{e}(k) \end{cases} \tag{1}$$

in which $\mathbf{x}(k) \in \mathbb{R}^{nN^2}$ and $\mathbf{u}(k) \in \mathbb{R}^{mN^2}, \mathbf{y}(k) \in \mathbb{R}^{pN^2}$ are, respectively, the lifted inputs and outputs. The measurement noise $\mathbf{e}(k)$ is zero-mean white and with an unknown covariance matrix. Identifying the global matrices corresponding to the lifted state (1) from standard subspace methods such as N4SID and MOESP [7] is not possible for large N as these methods scale at the very least with $\mathcal{O}(N^6)$. The difficulties for these centralized methods to handle large-scale 2-D networks has already been noticed in [9]. A more thorough explanation of the computational cost is found in Section II. Scalable subspace identification from data collected in a network may exploit information about the interconnection pattern between the subsystems. The two main approaches that have been followed so far in the literature are either based on *local* or *modal* decompositions.

The former assumes knowledge of the topology of the dynamics. In this case, we might impose a certain structure on the system matrices, e.g, the state-transition matrix $\mathbf{A}$ having a block-tridiagonal pattern. Writing now the state equation

locally, for each node located at the position (i, j)

$$\mathbf{x}_{i,j}(k+1) = \sum_{i'=1}^{N} \sum_{j'=1}^{N} \left(\mathbf{A}_{i',j'}^{(i,j)} \mathbf{x}_{i',j'}(k) + \mathbf{B}_{i',j'}^{(i,j)} \mathbf{u}_{i',j'}(k) \right) \quad (2)$$

where $\mathbf{x}_{i,j}(k) \in \mathbb{R}^n, \mathbf{u}_{i,j}(k) \in \mathbb{R}^m$. When the local matrix $\mathbf{A}_{i',j'}^{(i,j)}$ is nonzero, the neighboring unknown state at position (i', j') influences the local state $\mathbf{x}_{i,j}(k+1)$. Each subsystem influences the state of very few neighbors through some unknown interconnection state. Let $\mathcal{N}_{i,j}$ denote the collection of these neighboring systems for the node (i, j), and such that $\text{card}(\mathcal{N}_{i,j}) \ll N^2$. The local state space equation (2) reduces to

$$\mathbf{x}_{i,j}(k+1) = \sum_{(i',j') \in \mathcal{N}_{i,j}} \left(\mathbf{A}_{i',j'}^{(i,j)} \mathbf{x}_{i',j'}(k) + \mathbf{B}_{i',j'}^{(i,j)} \mathbf{u}_{i',j'}(k) \right).$$

$$(3)$$

A challenge is that often, the states leaking into the local subsystems are not measurable. Identification of the state matrices in a sparse network has been addressed in [10] and [11] when the adjacency matrix has a banded pattern. Both works in [12] and [13] tackle subspace identification for more general interconnection patterns where the subsystems are organized in clusters and use only local measurements to estimate the states of the neighbouring subsystems. However, these approaches require the topology to be known. Moreover, in these local models, the global stability of the identified network can neither be guaranteed nor influenced.

The second approach decouples the large-scale state-space model (1) into small-scale *modal* models by means of a block-diagonalizing similarity transformation. The success of this approach depends on the efficiency of applying this similarity transformation. For example, for the special class of 1-D interconnected circular systems [14], the Fourier transform enables an efficient application. Another Fourier transform based block-diagonalization method is the extension to the class of decomposable systems in [15]. However, the identification of the local systems required the solution of a bilinear optimization problem.

The major constraint of both types of approaches is that they require the large-scale network to have a very specific and known network topology such as 1-D homogeneous [11] or decomposable [15]. In this paper, we develop a subspace type of identification approach that is scalable, does not assume sparsity in the interconnection pattern and breaks away from assuming the system network topology to be known. A new modeling paradigm is introduced to model large-scale matrices of state-space models, which builds on the paper [16] where the class of Kronecker networks has been described. The latter does not require any knowledge on the interconnection pattern in the network nor the subsystems to be identical. In the state-space model stated in the abstract, the products such as $\mathbf{A}_2 \mathbf{X}(k) \mathbf{A}_1^T$ require separability of the column operations from those of the row operation on the matrix $\mathbf{X}(k)$. This separability assumption is common in image processing to model static input-output maps [17].

Although exploiting Kronecker structures to tackle large-scale problems is well-known in tensor-based scientific computing [22], its use has been mainly restricted to forward modeling

with, e.g., partial differential equation or for solving large-scale linear systems. System identification of multidimensional large-scale systems is however in its infancy. A class of multilinear dynamical systems (MLDS) is introduced in [20] for modeling tensor-time series and an expectation-maximization algorithm is presented for estimating parameters. The well-known drawbacks of such methods are the *a priori* selection of the order and the high computational cost, i.e $\mathcal{O}(N^6)$ per iteration. Therefore, they are often used in combination with subspace methods that provides them with initial estimates. In the sequel, we propose to overcome the computational complexity issue of the parameter estimation methods and present a subspace type of identification method for Kronecker state-space models. The system matrices in (1) are assumed to have a Kronecker rank equal to 1

$$\mathbf{A} = \mathbf{A}_1 \otimes \mathbf{A}_2, \quad \mathbf{B} = \mathbf{B}_1 \otimes \mathbf{B}_2, \quad \mathbf{C} = \mathbf{C}_1 \otimes \mathbf{C}_2. \quad (4)$$

We present an algorithm to estimate the system matrices with $\mathcal{O}(N^3 N_t)$ where N_t is the number of temporal samples rather than $\mathcal{O}(N^6)$ computational complexity. This paper is the companion to the paper [16] that considers the identification of large-scale Kronecker vector-autoregressive models, abbreviated with QUARKS. This methodology now serves as a first step (out of three) in the identification of state-space models when the matrices are of Kronecker rank one.

The QUARKS identification is relevant when each coefficient matrix is modeled as a sum of few Kronecker products of low-dimensional matrices, which enables high data compression. Such models are interesting when the variance of the prediction error is similar to the one obtained with the unstructured estimation. This Kronecker parametrization relies on the low-rank property of a certain reshuffling of the global system matrices. Reshuffling the sensor data or the coefficient matrices to exhibit low rank matrices and handle the blind system identification of large datasets was studied in [18]. In this paper, we take advantage of the special 2-level structure of the coefficient matrices that arise in 2-D spatial systems. Data-driven methods for identifying a Kronecker-based model also include [19]. The latter reshuffles the data in order to exhibit a large $N^2 \times N^2$ low-rank matrix and therefore, is not a potential candidate for deriving a scalable method for this first step. Furthermore, a significant difference is that it addresses the identification of vectors in a Kronecker product rather than matrices.

The main contributions of this work are the formulation of a new class of 2-D spatial-temporal models within the state-space framework and the formulation of a three-stage algorithm with lower computational complexity than existing methods. This algorithm Kronecker-Structured large-Scale SubSpace IDentification is abbreviated as K4SID. Moreover, we highlight the performances in terms of data compression and prediction-error with an application to turbulence prediction for large-scale adaptive optics systems. K4SID is compared with SSARX [35] and the estimates are refined with MLDS [20] for small sizes of the sensor.

The paper has the following outline. Section II formulates the identification problem and introduces theoretical results related to the Kronecker state-space model. Section III summarizes the identification of QUARKS models for estimating a high-order

FIR filter in Kronecker form. The estimates feature two sequences of impulse response -with terms of size $pN \times mN$- that are related via a bilinear equation. Section IV analyzes, the question why realizing the state-space matrices from these estimates requires to first solve a bilinear low-rank optimization. A method is proposed in Section V to estimate the factor matrices $\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}_1, \mathbf{B}_2, \mathbf{C}_1, \mathbf{C}_2$ using two consecutive singular value decompositions (SVD). A realistic numerical example for predicting large-scale wavefront aberrations in an adaptive optics setting is presented in Section VI.

Notations: Scalars are denoted by lower or uppercase letters or symbols. $\lfloor x \rfloor$ denotes the floor of the real number x. $mod(a,m)$ denotes the remainder after division of a by m.

Vectors are written as boldface lower-case letters such as $\mathbf{x}$. The boldface is used to make a distinction between indexing a set of vectors, such as $\mathbf{x}_1, \mathbf{x}_2$, and referring to the elements of a single vector $\mathbf{x} \in \mathbb{R}^n$, such as $x_1, \ldots, x_n$. The null vector and the vector of ones is denoted by $\mathbf{0}$ and $\mathbf{1}$, respectively, where an index can be used to explicitly show its size, e.g., $\mathbf{1}_n \in \mathbb{R}^n$. The Euclidean norm of a vector $\mathbf{x}$ is written as $\|\mathbf{x}\|_2 = \sqrt{x_1^2 + \ldots + x_n^2} = \langle \mathbf{x}, \mathbf{x} \rangle$.

Matrices are represented by boldface uppercase letters such has $\mathbf{X}$. The element located at the ith row and j-column of the matrix $\mathbf{X}$ is written as $x_{i,j}$. The identity matrix of size $N \times N$ is written with $\mathbf{I}_N$. The inverse, pseudo-inverse and transpose are written as $\mathbf{X}^{-1}$, $\mathbf{X}^{\dagger}$, and $\mathbf{X}^T$, respectively. MATLAB-like notations are used to denote columns and rows of matrices, e.g., $\mathbf{X}(:,\mathtt{i})$ refers to the ith column of $\mathbf{X}$, $\mathbf{X}(\mathtt{i},:)$ the ith row. The matrix $\mathbf{X}(\mathtt{a:i:b,c:j:d})$ selects a submatrix from $\mathbf{X}$ that consists of the intersection of all entries with row index $a + ki$ (until it reaches b) and column index $c + kj$ (until d) where k is an integer starting at 0. It is standard MATLAB notation. The vectorization operator applied on $\mathbf{X}$ is written with $\mathrm{vec}(\mathbf{X}) = \begin{bmatrix} x_{1,1} & x_{2,1} & \ldots & x_{m,n} \end{bmatrix}^T$. The Kronecker product of two matrices $\mathbf{X}, \mathbf{Y}$ is represented by the symbol $\otimes$ such as $\mathbf{X} \otimes \mathbf{Y}$. The Frobenius norm for a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ is denoted with $\|\mathbf{X}\|_F^2 = \sqrt{\sum_{i=1}^m \sum_{j=1}^n x_{i,j}^2}$. The nuclear norm for a matrix $\mathbf{X}$ is indicated with $\|\mathbf{X}\|_\star$.

Let m, n, s, s_1 be strictly positive integer's such that $s = 2s_1 - 1$, and define $\mathbf{x} \in \mathbb{R}^s$ and $\mathbf{Y}_i \in \mathbb{R}^{m \times n}$ for all $i = 1..s$. The block-Hankel matrix $\mathcal{H}(x_i \mathbf{Y}_i)$ is such that

$$\mathcal{H}(x_i \mathbf{Y}_i) = \begin{bmatrix} x_1 \mathbf{Y}_1 & x_2 \mathbf{Y}_2 & x_3 \mathbf{Y}_3 & \ldots & x_{s_1} \mathbf{Y}_{s_1} \\ x_2 \mathbf{Y}_2 & x_3 \mathbf{Y}_3 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ x_{s_1} \mathbf{Y}_{s_1} & & \ldots & & x_s \mathbf{Y}_s \end{bmatrix}.$$

The big-$\mathcal{O}$ notation is used for describing computational complexities and indicates the asymptotic growth rate of the computational cost for a given mathematical operation. For example, an operation costing $\mathcal{O}(n)$ floating-point operations (flops) finishes in at most $c \cdot n$ flops, for some constant c.

Other section-specific notations are introduced in the respective section.

Nomenclature: A matrix $\mathbf{X}$ written as $\mathbf{X} = \mathbf{X}_1 \otimes \mathbf{X}_2$ is said to have Kronecker rank 1. The matrices $\mathbf{X}_1$ and $\mathbf{X}_2$ are called the factor matrices. The matrices $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i$ are called the factored state-space matrices. The terms $\mathbf{C}\mathbf{A}^i\mathbf{B} = \mathbf{C}_1\mathbf{A}_1^i\mathbf{B}_1 \otimes \mathbf{C}_2\mathbf{A}_2^i\mathbf{B}_2$ are the Markov parameters while $\mathbf{C}_j\mathbf{A}_j^i\mathbf{B}_j$ are called the factored Markov parameters. The computational rules related to the Kronecker product are described in [21].

II. PROBLEM FORMULATION

We consider a 2-D array with $N \times N$ nodes, that each has the same number of input and output data. The input data are collected at time instant k into the matrix $\mathbf{U}(k) \in \mathbb{R}^{mN \times N}$

$$\mathbf{U}(k) = \begin{bmatrix} \mathbf{u}_{1,1}(k) & \ldots & \mathbf{u}_{1,N}(k) \\ \vdots & & \vdots \\ \mathbf{u}_{N,1}(k) & \ldots & \mathbf{u}_{N,N}(k) \end{bmatrix}$$

where, for $i, j = 1 \ldots N, \mathbf{u}_{i,j} \in \mathbb{R}^m$. The output matrix $\mathbf{Y}(k)$ is defined similarly from local signals $\mathbf{y}_{i,j}(k) \in \mathbb{R}^p$. Denote the lifted quantities with $\mathbf{u}(k) = \mathrm{vec}(\mathbf{U}(k))$. The temporal dynamics of the network are modeled with the state-space model (1) in which the state-space matrices have a Kronecker rank equal to 1

$$\mathbf{A} = \mathbf{A}_1 \otimes \mathbf{A}_2, \quad \mathbf{B} = \mathbf{B}_1 \otimes \mathbf{B}_2, \quad \mathbf{C} = \mathbf{C}_1 \otimes \mathbf{C}_2 \quad (5)$$

with,

$$\begin{array}{llll} \mathbf{A}_1 \in \mathbb{R}^{n_1 \times n_1}, & \mathbf{B}_1 \in \mathbb{R}^{n_1 \times N}, & \mathbf{C}_1 \in \mathbb{R}^{N \times n_1} \\ \mathbf{A}_2 \in \mathbb{R}^{n_2 \times n_2}, & \mathbf{B}_2 \in \mathbb{R}^{n_2 \times mN}, & \mathbf{C}_2 \in \mathbb{R}^{pN \times n_2}. \end{array} \quad (6)$$

When the global state-space model (1) has the Kronecker structure (5), it is equivalently written in matrix form as

$$\begin{cases} \mathbf{X}(k+1) = \mathbf{A}_2\mathbf{X}(k)\mathbf{A}_1^T + \mathbf{B}_2\mathbf{U}(k)\mathbf{B}_1^T \\ \mathbf{Y}(k) = \mathbf{C}_2\mathbf{X}(k)\mathbf{C}_1^T + \mathbf{E}(k). \end{cases} \quad (7)$$

Definition 1: The set of generators $\mathcal{S}$ for the Kronecker MSSM (7) is defined with the factored state-space matrices as follows:

$$\mathcal{S} = \{\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}_1, \mathbf{B}_2, \mathbf{C}_1, \mathbf{C}_2\}$$

with the dimensions of the corresponding matrices given in (6).

In Lemma 1, Lemma 2, and Corollary 1, we relate the stability, the observability and the minimal realization associated to the large-scale matrices $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ to the sets defined from $(\mathbf{A}_1, \mathbf{B}_1, \mathbf{C}_1)$ and $(\mathbf{A}_2, \mathbf{B}_2, \mathbf{C}_2)$. First, we establish a relationship between the spectral radius of $\mathbf{A}_1 \otimes \mathbf{A}_2$ and the factor matrices $\mathbf{A}_1, \mathbf{A}_2$.

Lemma 1: If the systems associated with $(\mathbf{A}_1, \mathbf{B}_1, \mathbf{C}_1)$ and $(\mathbf{A}_2, \mathbf{B}_2, \mathbf{C}_2)$ are both stable, then the system associated with $(\mathbf{A}_1 \otimes \mathbf{A}_2, \mathbf{B}_1 \otimes \mathbf{B}_2, \mathbf{C}_1 \otimes \mathbf{C}_2)$ is stable. The reverse is not true in general.

Proof: Let $(i, j) \in \{1, \ldots, n_1\} \times \{1, \ldots, n_2\}$. Assume that $\mathbf{A}_1$ and $\mathbf{A}_2$ have eigenvalues, respectively, $\mu_{1,i}$ and $\mu_{2,j}$, lying strictly within the unit circle. The eigenvalues of $\mathbf{A}_1 \otimes \mathbf{A}_2$ are $\mu_{1,i}\mu_{2,j}$. If $|\mu_{1,i}| < 1$ and $|\mu_{2,j}| < 1$, then $|\mu_{1,i}\mu_{2,j}| < 1$. However, if $|\mu_{1,i}\mu_{2,j}| < 1$, it does not guarantee that both $|\mu_{1,i}| < 1$ and $|\mu_{2,j}| < 1$. $\square$

Let s_1 denote an integer such that $s_1 N \min(p, m) > \max(n_1, n_2)$. The integer s is defined as $2s_1 - 1$ for reasons

that are explained in Section IV. Let the observability matrix built from two matrices $\mathbf{C}_i, \mathbf{A}_i$ be denoted with $\mathcal{O}_{i,s}$ such that:

$$\mathcal{O}_{i,s} = \begin{bmatrix} \mathbf{C}_i \\ \mathbf{C}_i \mathbf{A}_i \\ \vdots \\ \mathbf{C}_i \mathbf{A}_i^{s-1} \end{bmatrix}$$

$$\mathcal{O}_s = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A} \\ \vdots \\ \mathbf{C}\mathbf{A}^{s-1} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_1 \otimes \mathbf{C}_2 \\ \mathbf{C}_1 \mathbf{A}_1 \otimes \mathbf{C}_2 \mathbf{A}_2 \\ \vdots \\ \mathbf{C}_1 \mathbf{A}_1^{s-1} \otimes \mathbf{C}_2 \mathbf{A}_2^{s-1} \end{bmatrix}.$$

The extended controllability matrix built from $\mathbf{B}_i, \mathbf{A}_i$ is similarly denoted with $\mathcal{C}_{i,s}$:

$$\mathcal{C}_{i,s} = \begin{bmatrix} \mathbf{B}_i & \mathbf{A}_i\mathbf{B}_i & \dots & \mathbf{A}_i^{s-1}\mathbf{B}_i \end{bmatrix}$$

$$\mathcal{C}_s = \begin{bmatrix} \mathbf{B} & \mathbf{A}\mathbf{B} & \dots & \mathbf{A}^{s-1}\mathbf{B} \end{bmatrix}.$$

Lemma 2: If $(\mathbf{A}_1 \otimes \mathbf{A}_2, \mathbf{C}_1 \otimes \mathbf{C}_2)$ is observable, then each of the pairs $(\mathbf{A}_1, \mathbf{C}_1)$ and $(\mathbf{A}_2, \mathbf{C}_2)$ is observable.

The reverse is not true in general.

Proof: Let $n = n_1 n_2$. We start by partitioning the columns of $\mathcal{O}_n$ blockwise:

$$\mathcal{O}_n = \begin{bmatrix} \mathbf{L}_0 & \dots & \mathbf{L}_{n_1-1} \end{bmatrix}$$

where $\mathbf{L}_j \in \mathbb{R}^{npN^2 \times n_2}$ for $j = 0 \dots n_1 - 1$. Each block-matrix $\mathbf{L}_j$ is such that

$$\mathbf{L}_j = \begin{bmatrix} c_{1,(1,j+1)}\mathbf{C}_2 \\ \vdots \\ c_{1,(N,j+1)}\mathbf{C}_2 \\ \vdots \\ m_{n-1,(N,j+1)}\mathbf{W}_{n-1} \end{bmatrix}$$

where $\mathbf{M}_{n-1} = \mathbf{C}_1 \mathbf{A}_1^{n-1}, \mathbf{W}_{n-1} = \mathbf{C}_2 \mathbf{A}_2^{n-1}$. If $\mathcal{O}_n$ is full column rank, so is $\mathbf{L}_j$. It yields the following equalities for the column ranks:

$$\text{rank}(\mathbf{L}_j) = \text{rank}(\mathcal{O}_{2,n}). \tag{8}$$

Therefore, $\text{rank}(\mathcal{O}_{2,n}) = n_2$. A similar reasoning on the submatrix $\mathcal{O}_n(1 : pN : pnN^2, 1 : n_2 : n)$ holds to prove that $\text{rank}(\mathcal{O}_{1,n}) = n_1$.

We provide with a counter example for the reverse side

$$\mathbf{C}_1 = \begin{bmatrix} 1 & 1 \end{bmatrix} \quad \mathbf{A}_1 = \begin{bmatrix} 0.4 & 0 \\ 0 & 0.6 \end{bmatrix}$$

$$\mathbf{C}_2 = \begin{bmatrix} 1 & 1 \end{bmatrix} \quad \mathbf{A}_2 = \begin{bmatrix} 0.6 & 0 \\ 0 & 0.4 \end{bmatrix}.$$

Both $\mathcal{O}_{1,n}$ and $\mathcal{O}_{2,n}$ are full column rank which is not the case for $\mathcal{O}_n$. □

Controllability is the dual notion from observability, and therefore, a similar statement can be made for the pairs $(\mathbf{A}_1, \mathbf{B}_1)$ and $(\mathbf{A}_2, \mathbf{B}_2)$ to be controllable.

Definition 2: A minimal realization of (7) corresponds to a set $\mathcal{S}$ such that the extended observability and controllability

matrices built, respectively, from the pairs $(\mathbf{A}_1 \otimes \mathbf{A}_2, \mathbf{C}_1 \otimes \mathbf{C}_2)$ and $(\mathbf{A}_1 \otimes \mathbf{A}_2, \mathbf{B}_1 \otimes \mathbf{B}_2)$ are of minimal rank $n_1 n_2$.

Corollary 1: If the set of generators $\mathcal{S} = \{\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}_1, \mathbf{B}_2, \mathbf{C}_1, \mathbf{C}_2\}$ corresponds to a minimal realization of the MSSM (7), then both sets $\{\mathbf{A}_1, \mathbf{B}_1, \mathbf{C}_1\}$ and $\{\mathbf{A}_2, \mathbf{B}_2, \mathbf{C}_2\}$ correspond to a minimal realization.

The reverse is not true in general.

Proof: The proof follows from Lemma 2. □

As shown in [21] for standard matrix properties and in the above results, a particularity of the Kronecker product is that the properties relating the global matrices to the factors are often one-sided. We now investigate the state-space (7) from the input-output relationship, which matrix form reads

$$\mathbf{Y}(k) = \mathbf{C}_2 \mathbf{A}_2^{k-1} \mathbf{X}(1) \mathbf{A}_1^{k-1^T} \mathbf{C}_1^T$$
$$+ \sum_{i=1}^{k-1} \mathbf{C}_2 \mathbf{A}_2^{k-i-1} \mathbf{B}_2 \mathbf{U}(i) \mathbf{B}_1^T \mathbf{A}_1^{k-i-1^T} \mathbf{C}_1^T + \mathbf{E}(k). \tag{9}$$

Definition 3: Let N_t be the number of temporal samples. For $i \in \{1, 2\}$, denote:

$$\mathcal{S}_i = \{\mathbf{A}_1^{(i)}, \mathbf{A}_2^{(i)}, \mathbf{B}_1^{(i)}, \mathbf{B}_2^{(i)}, \mathbf{C}_1^{(i)}, \mathbf{C}_2^{(i)}\}.$$

The two sets of generators $\mathcal{S}_1$ and $\mathcal{S}_2$ are said to be equivalent if the input-output behavior of the associated state-space model (7) is identical for all $k = 1 \dots N_t$.

It is well-known that the state-space matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ in (1) modeling the input-output relationship are not unique because of the existence of a nonsingular similarity transformation $\mathbf{T} \in \mathbb{R}^{n_1 n_2 \times n_1 n_2}$. Reshuffling (1) yields (7) if and only if the state-space matrices are all of Kronecker rank one. It is not the case when allowing similarity transformations not written as $\mathbf{T} = \mathbf{T}_1 \otimes \mathbf{T}_2$.

We characterize the similarity transformation in the case of Kronecker state-space models and relate equivalent sets of generators in the next Lemma.

Lemma 3: The sets of generators $\mathcal{S}_1$ and $\mathcal{S}_2$ for the Kronecker MSSM equivalently model (7) if and only if there exist $\mathbf{T}_1 \in \mathbb{R}^{n_1 \times n_1}, \mathbf{T}_2 \in \mathbb{R}^{n_2 \times n_2}$ nonsingular, $\mathbf{P}_1 \in \mathbb{R}^{mN \times mN}, \mathbf{P}_2 \in \mathbb{R}^{N \times N}$ and nonzero scalars η, c_t that satisfy

$$\forall k \in \{1, \dots, N_t\}, \quad \mathbf{P}_2 \mathbf{U}(k)\mathbf{P}_1^T = \mathbf{U}(k) \tag{10}$$

$$\left.\begin{array}{l} \mathbf{A}_1^{(1)} = \eta \mathbf{T}_1^{-1} \mathbf{A}_1^{(2)} \mathbf{T}_1 \\[2mm] \mathbf{B}_1^{(1)} = \mathbf{T}_1^{-1} \mathbf{B}_1^{(2)} \mathbf{P}_1 \\[2mm] \mathbf{C}_1^{(1)} = c_t \mathbf{C}_1^{(2)} \mathbf{T}_1 \end{array}\right\} \tag{11}$$

$$\left.\begin{array}{l} \mathbf{A}_2^{(1)} = \frac{1}{\eta} \mathbf{T}_2^{-1} \mathbf{A}_2^{(2)} \mathbf{T}_2 \\[2mm] \mathbf{B}_2^{(1)} = \mathbf{T}_2^{-1} \mathbf{B}_2^{(2)} \mathbf{P}_2 \\[2mm] \mathbf{C}_2^{(1)} = \frac{1}{c_t} \mathbf{C}_2^{(2)} \mathbf{T}_2 \end{array}\right\}. \tag{12}$$

Proof: If (11) and (12) hold, then the input-output model (9) built from $\mathcal{S}_1$ is equivalent to the model built from $\mathcal{S}_2$:

$$\tilde{\mathbf{X}}(k+1) = \mathbf{T}_2^{-1}\frac{1}{\eta}\mathbf{A}_2^{(2)}\mathbf{T}_2\tilde{\mathbf{X}}(k)\mathbf{T}_1^T\eta\mathbf{A}_1^{(2)^T}\mathbf{T}_1^{-T}$$

$$+ \mathbf{T}_2^{-1}\mathbf{B}_2^{(2)}\mathbf{P}_2\mathbf{U}(k)\mathbf{P}_1^T\mathbf{B}_1^{(2)^T}\mathbf{T}_1^{-T}$$

$$\mathbf{Y}(k) = \frac{1}{c_t}\mathbf{C}_2^{(2)}\mathbf{T}_2\tilde{\mathbf{X}}(k)\mathbf{T}_1^T c_t\mathbf{C}_1^{(2)^T} \qquad (13)$$

where $\tilde{\mathbf{X}}(k) = \mathbf{T}_2^{-1}\mathbf{X}(k)\mathbf{T}_1^{-T}$. This model yields the same input-output behavior provided that

$$\mathbf{P}_2\mathbf{U}(k)\mathbf{P}_1^T = \mathbf{U}(k). \qquad (14)$$

This assumes a global similarity transformation with a Kronecker structure: $\mathbf{T} = \mathbf{T}_1 \otimes \mathbf{T}_2$. $\square$

The matrices $\mathbf{P}_1, \mathbf{P}_2$ in Lemma 3 are particular to MSSM and can be further characterized by assuming persistency of excitation. For all temporal samples until N_t, we form the matrix $\tilde{\mathbf{U}} \in \mathbb{R}^{mN^2 \times N_t}$ from concatenated input data with

$$\tilde{\mathbf{U}} = \left[\,\text{vec}(\mathbf{U}(1))\ldots\text{vec}(\mathbf{U}(N_t))\,\right].$$

Definition 4: **[7]** The input is persistency exciting of order 1 if the matrix $\tilde{\mathbf{U}}$ is full row rank.

Let us assume $N_t \geq mN^2$ and that the input is persistently exciting. Then, (14) is equivalently written with

$$(\mathbf{P}_1 \otimes \mathbf{P}_2 - \mathbf{I})\text{vec}(\mathbf{U}(k)) = 0.$$

Concatenating data for all k in $\{1, \ldots, N_t\}$, it follows:

$$(\mathbf{P}_1 \otimes \mathbf{P}_2 - \mathbf{I})\tilde{\mathbf{U}} = 0. \qquad (15)$$

Under the assumption that the matrix $\tilde{\mathbf{U}}$ is full column rank

$$\mathbf{P}_1 \otimes \mathbf{P}_2 = \mathbf{I}$$

which implies that both $\mathbf{P}_1$ and $\mathbf{P}_2$ are diagonal, and

$$p_{1,(i,i)}p_{2,(j,j)} = 1$$

for all $i = 1\ldots mN, j = 1..N$. Hence, $\mathbf{P}_1 = b_t\mathbf{I}_{mN}$ and $\mathbf{P}_2 = \frac{1}{b_t}\mathbf{I}_N$ for some nonzero scalar b_t.

Remark 1: When the similarity transformation $\mathbf{T}$ is unstructured, the Kronecker structure in the vectorized state-space model is lost. Denote $\mathbf{x}_{\mathbf{T}}(k) := \mathbf{T}^{-1}\mathbf{x}(k)$. Then, the Kronecker rank of $\mathbf{T}^{-1}(\mathbf{A}_1 \otimes \mathbf{A}_2)\mathbf{T}$ is in general not one unless $\mathbf{T} = \mathbf{T}_1 \otimes \mathbf{T}_2$. Therefore, the vector state-space model with matrices $(\mathbf{T}^{-1}(\mathbf{A}_1 \otimes \mathbf{A}_2)\mathbf{T}, \mathbf{T}^{-1}(\mathbf{B}_1 \otimes \mathbf{B}_2), (\mathbf{C}_1 \otimes \mathbf{C}_2)\mathbf{T})$ cannot be rewritten in general with a matrix state-space model as in (7).

The above discussion is of particular interest as the identification algorithm we propose relies on the matrix state-space model (7) and the global matrices are never formed. The global similarity transformation is not involved.

In order to derive a scalable identification algorithm, the following assumptions on the data and system matrices in (7) are made.

1) **A1:** The pair $(\mathbf{A}_1 \otimes \mathbf{A}_2, \mathbf{C}_1 \otimes \mathbf{C}_2)$ is observable.
2) **A2:** The pair $(\mathbf{A}_1 \otimes \mathbf{A}_2, \mathbf{B}_1 \otimes \mathbf{B}_2)$ is controllable.
3) **A3:** The eigenvalues of both $\mathbf{A}_1$ and $\mathbf{A}_2$ are strictly within the unit circle.

4) **A4:** The input is persistently exciting.
5) **A5:** The measurement noise $\mathbf{e}(k)$ is zero-mean white noise with unknown covariance matrix.
6) **A6:** The measurement noise is uncorrelated with all past inputs:

$$\text{for all } k \leq j, E\left[\mathbf{u}(k)\mathbf{e}(j)^T\right] = \mathbf{0}$$

The assumptions **A1** to **A4** and **A6** are related to the global system properties and are commonly used in subspace identification, [7]. The assumption **A6** is also made in [12]–[13] in order to focus on the essential building of the subspace identification method(s) like the vector autoregressive with exogenous inputs (VARX) modeling and state sequence approximations. The generality of the method is illustrated in Section IV in which a model in innovation form is identified. We are now ready to state the problem formulation.

Problem Formulation: *Assuming* **A1** *to* **A6***, and given the input-output data* $\mathbf{U}(k), \mathbf{Y}(k)$ *from the state-space model in (7) for* $k = \{1, \ldots, N_t\}$*, estimate, up to the similarities transformation* $\mathbf{T}_1, \mathbf{T}_2$ *and the ambiguity scaling factors* η, c_t, b_t *defined in Lemma 3, the matrices* $\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}_1, \mathbf{B}_2, \mathbf{C}_1, \mathbf{C}_2$ *that correspond to a minimal realization. The challenge lies on deriving an algorithm with* $\mathcal{O}(N^3N_t)$ *computational complexity.*

Such requirements on the computational cost exclude a centralized identification of (1) with standard subspace methods such as MOESP [7] or SSARX [35]. Such methods fail for three main reasons. First, they rely on a QR decomposition of the concatenated block-Hankel matrix built from the input-output sequence, whose size is $(p+m)sN^2 \times M$ (where $M = N_t - s + 1$ and s is a scalar). A square lower-triangular Gram–Schimdt matrix is only obtained when $N_t \geq psN^2$ which requires storing huge data samples. Second, with a global system order of n_1n_2, computing the QR decomposition and the SVD of $N^2 \times N^2$ matrices is very costly, $\mathcal{O}(N^6)$ flops. If a prior knowledge of the system order is available, then a rank-n_1n_2 SVD can be computed at a cost of $\mathcal{O}(n_1n_2N^4)$. More efficient methods [23] for computing SVD do not break the curse of dimensionality (that appears with multidimensional systems) and still require $\mathcal{O}(log(n_1n_2)N^4)$ flops. Last, forming the global matrices is a drawback for storage and, e.g., subsequent control design for real-time applications. For example, computing a matrix-vector multiplication with the dense unstructured matrix requires $\mathcal{O}(N^4)$ instead of $\mathcal{O}(N^3)$ in the matrix form.

The algorithm PBSID [24] provides with an alternative route that estimates first a high-order VARX and then computes the SVD of a large-matrix. The computational cost associated with the latter operation along with the estimation of unstructured (and dense) estimates of the state-space matrices reaches $\mathcal{O}(N^6)$ and is reduced in this paper by working rather with the factored Markov parameters $\mathbf{C}_1\mathbf{A}_1^i\mathbf{B}_1$ and $\mathbf{C}_2\mathbf{A}_2^i\mathbf{B}_2$.

In the three following sections, we describe the subspace-like method which is decomposed in three major steps. We first identify the factored Markov parameters using a globally convergent algorithm. Such parameters are however estimated up to an unknown scaling factor. Second, these estimates are analyzed so as to estimate the factored Markov parameters $w\mathbf{C}_1(\eta\mathbf{A}_1)^{i-1}\mathbf{B}_1$ and $\frac{1}{w}\mathbf{C}_2(\frac{1}{\eta}\mathbf{A}_2)^{i-1}\mathbf{B}_2$ (for w, η nonzero scalars). This paves

the way for the third step in Section V where we identify the factored state-space matrices by estimating the state sequence.

III. HIGH-ORDER FIR ESTIMATION

Based on Assumption **A3**, the output $\mathbf{y}(k)$ can be approximated with a high-order FIR model, for all $k > s$,

$$
\mathbf{y}(k) \approx \sum_{i=1}^{s} \mathbf{C}\mathbf{A}^{i-1}\mathbf{B}\mathbf{u}(k-i) + \mathbf{e}(k)
$$

$$
\approx \sum_{i=1}^{s} \Big(\mathbf{C}_1\mathbf{A}_1^{i-1}\mathbf{B}_1 \otimes \mathbf{C}_2\mathbf{A}_2^{i-1}\mathbf{B}_2\Big)\mathbf{u}(k-i) + \mathbf{e}(k).
$$

$$(16)$$

Denote $\mathbf{M}_{i,\ell} := \mathbf{C}_1\mathbf{A}_1^{i-1}\mathbf{B}_1 \in \mathbb{R}^{N \times N}$ and $\mathbf{M}_{i,r} := \mathbf{C}_2\mathbf{A}_2^{i-1}\mathbf{B}_2 \in \mathbb{R}^{pN \times mN}$. The matrix $\mathbf{M}_\ell = \begin{bmatrix} \mathbf{M}_{1,\ell} & \dots & \mathbf{M}_{s,\ell} \end{bmatrix}$ is denoted as the left-factor impulse response. Similarly, the matrix $\mathbf{M}_r$ is built from the factor matrices $\mathbf{M}_{i,r}$ and called the right-factor impulse response.

By appropriately selecting the parameters as standardly done in the subspace identification literature [7], the approximation error can be made arbitrarily small, [25].

A computationally efficient and globally convergent algorithm has been derived in [16] to estimate structured large-scale VARX models when the coefficient-matrices have large dimensions but low-Kronecker rank. In the FIR approximation (16), each Markov parameter $\mathbf{M}_i$ has Kronecker rank equal to 1, and hence the (16) can be recast into a minimization on the factor matrices $\mathbf{M}_{i,\ell}, \mathbf{M}_{i,r}$ only. We review in the next paragraph the main features of the *QUARKS* identification method of [16]. The cost function of the optimization problem features a least-squares term and a regularization prior to induce temporal stability of the estimated MIMO system.

The stability of the impulse responses built from factored matrices is imposed by using kernel regularization methods. The general problem of fitting impulse responses given data while guaranteeing stability has been studied in a number of papers. See e.g. [26], [27] for surveys on these kernel methods applied to function estimation. The kernel matrix $\mathbf{P}_t \in \mathbb{R}^{s \times s}$ is here introduced along with the decomposition of its inverse with a square-root matrix $\mathbf{K}_t$. Both the decay rate and the smoothness of the impulse responses can be tuned using hyperparameters as shown in [26], [27]. Adding the following cost as regularization to a cost function induces stable VARX models:

$$
r_t(\mathbf{M}_{i,\ell}, \mathbf{M}_{i,r}) = \left\| \mathbf{Q}_t \begin{bmatrix} \mathrm{vec}(\mathbf{M}_{1,\ell})\mathrm{vec}(\mathbf{M}_{1,r})^T \\ \vdots \\ \mathrm{vec}(\mathbf{M}_{p,\ell})\mathrm{vec}(\mathbf{M}_{p,r})^T \end{bmatrix} \right\|_F^2
$$

where $\mathbf{Q}_t = \mathbf{W}_t \otimes \mathbf{I}_{N^2}$. The factor matrices $\mathbf{M}_{i,\ell}, \mathbf{M}_{i,r}$ are estimated using alternating least squares (ALS) on the following least-squares bilinear minimization problem:

$$
\min_{\mathbf{M}_{i,r}, \mathbf{M}_{i,\ell}} \sum_{k=s+1}^{N_t} \left\| \mathbf{Y}(k) - \sum_{i=1}^{s} \mathbf{M}_{i,r}\mathbf{U}(k-i)\mathbf{M}_{i,\ell}^T \right\|_F^2
$$

$$
+ \lambda_{\mathrm{ALS}} r_t(\mathbf{M}_{i,\ell}, \mathbf{M}_{i,r}) \tag{17}
$$

Algorithm 1: Summary of *QUARKS*.

Input: $\{\mathbf{u}_k\}_{1:N_t}, \{\mathbf{y}_k\}_{1:N_t}, s, \lambda_{ALS}, \kappa_{\max}, e_{\max}$
Output: $\widehat{\mathbf{M}}_\ell, \widehat{\mathbf{M}}_r$
  /* QUARKS identification   */
1: $\kappa \leftarrow 0$
2: **foreach** $i \leq s$ **do**
3: $\quad \Big| \quad \mathbf{M}_{i,\ell}^{(\kappa)} \leftarrow$ randn(N, N)
4: **end**
5: **while** $\kappa \leq \kappa_{\max} - 1$ and $e > e_{\max}$ **do**
6: $\quad \Big| \quad \mathbf{M}_r^{(\kappa+1)} \leftarrow$ argmin $\mathcal{L}_{\mathbf{M}_\ell^{(\kappa)}}(\mathbf{M}_r)$.
7: $\quad \Big| \quad \mathbf{M}_\ell^{(\kappa+1)} \leftarrow$ argmin $\mathcal{L}_{\mathbf{M}_r^{(\kappa+1)}}(\mathbf{M}_\ell)$.
8: $\quad \Big| \quad$ Evaluate the residual $c^{(\ell)}$
9: $\quad \Big| \quad e \leftarrow |c^{(\ell)} - c^{(\ell-1)}|$
10: $\quad \Big| \quad \kappa \leftarrow \kappa + 1$
11: **end**
12: $\widehat{\mathbf{M}}_r \leftarrow \mathbf{M}_r^{(\kappa_{\max})}$
13: $\widehat{\mathbf{M}}_\ell \leftarrow \mathbf{M}_\ell^{(\kappa_{\max})}$

---

where λ_{ALS} is a regularization parameter. An ALS algorithm is proposed and described in Algorithm 1. The notation $\mathcal{L}_{\mathbf{M}_r}(\mathbf{M}_\ell)$ is introduced and refers to the the cost function in (17) when the optimization variables are only $\mathbf{M}_{i,\ell}$ for all i while $\mathbf{M}_{i,r}$ is fixed. As was the case for *QUARKS* identification in [16] in which similar priors were used to induce stability, random search [28] can be used for finding a set of valid hyperparameters.

The convergence of the regularized ALS to the global minimum of the ALS provided the input is persistently exciting and the initial guesses are nonzero is proved in [16]. The solution to (17) is not unique as summarized in the following Lemma.

Lemma 4: **[16].** Let s and N_t tend towards infinity. Let $i \in \{1, \dots, s\}$ and $t_i \in \mathbb{R}\backslash\{0\}$. Denote a solution to (17) with the parameters $\mathbf{M}_{i,\ell}, \mathbf{M}_{i,r}$.

The set of all solutions $\widehat{\mathbf{M}}_{i,\ell}, \widehat{\mathbf{M}}_{i,r}$ that yield the same optimum value of the cost function (17) is such that

$$
\mathrm{vec}(\widehat{\mathbf{M}}_{i,\ell}) = \mathrm{vec}(\mathbf{M}_{i,\ell})t_i, \qquad \mathrm{vec}(\widehat{\mathbf{M}}_{i,r}) = \frac{1}{t_i}\mathrm{vec}(\mathbf{M}_{i,r})
$$

Proof: See [16]. □

Denote the estimates from (17) with $\widehat{\mathbf{M}}_{i,\ell}, \widehat{\mathbf{M}}_{i,r}$. They are related to the nonscaled parameters $\mathbf{M}_{i,\ell}, \mathbf{M}_{i,r}$ with

$$
\widehat{\mathbf{M}}_{i,\ell} \approx t_i \mathbf{M}_{i,\ell}, \qquad \widehat{\mathbf{M}}_{i,r} \approx v_i \mathbf{M}_{i,r} \tag{18}
$$

where $t_i v_i \approx 1$. The nonzero ambiguity constants t_i are however unknowns and *different* for each i.

A. Computational Complexity

The computational complexity for the QUARKS has been analyzed in [16]. We assume that the number of iterations $\kappa_{\max}$ are independent of N, which is however not the case for the number of temporal samples N_t. The algorithm scales with $\mathcal{O}(N^3 N_t)$, where $N_t > s$.

IV. ESTIMATION OF THE IMPULSE RESPONSES UP TO A SCALING FACTOR

In this section, we assume that the matrices $\mathbf{M}_{i,\ell}$ and $\mathbf{M}_{i,r}$ are estimated up to a different nonzero scaling factor as highlighted in Lemma 4 in the previous section. This relationship between the matrices estimated with QUARKS in (17) and their true variants hold in an asymptotic consistent manner. We now study how to estimate the factored Markov parameters $w\mathbf{C}_1(\eta\mathbf{A}_1)^{i-1}\mathbf{B}_1$ (for w, η nonzero scalars). Although, the forthcoming analysis is performed on the left factor matrices $\mathbf{M}_{i,\ell}$, it is equally valid for the right factor matrices $\mathbf{M}_{i,r}$.

A. A Low-Rank Block-Hankel Matrix

From the matrices $\begin{bmatrix} \mathbf{M}_{1,\ell} & \dots & \mathbf{M}_{s,\ell} \end{bmatrix}$, the realization theory consists in forming a block-Hankel low-rank matrix $\mathcal{H}(\mathbf{M}_{1,\ell})$ that is equal to $\mathcal{O}_{1,s_1}\mathcal{C}_{1,s_1}$. A SVD is then computed to estimate the range space of the observability matrix $\mathcal{O}_{1,s_1}$. In the analysis from this section, the approximations in (18) are considered as equalities. Because, the unknowns t_i are dependent on the temporal index i, the block-Hankel matrix built from the estimated matrices $\begin{bmatrix} \widehat{\mathbf{M}}_{1,\ell} & \dots & \widehat{\mathbf{M}}_{s,\ell} \end{bmatrix}$ is in general not low rank.

Each of the factored Markov parameters need to be multiplied with a scalar on which we formulate conditions such that the resulting block-Hankel matrix has rank equal to n_1. We describe this statement in Theorem 1 for which Lemma 5 is needed.

Lemma 5 (Partial Realization Problem [29]): Let $\delta \in \mathbb{N}$. Let $s = 2s_1 - 1$ with s_1 an integer strictly larger than δ. For $i = 1 \dots s$, let $x_i \in \mathbb{R}$ such that $\operatorname{rank}(\mathcal{H}(x_i)) = \delta$.

Then, there exists a realization $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ of minimal degree δ with $\mathbf{a} \in \mathbb{R}^{\delta \times \delta}, \mathbf{b} \in \mathbb{R}^{\delta \times 1}, \mathbf{c} \in \mathbb{R}^{1 \times \delta}$ such that for $i = 1 \dots s, x_i = \mathbf{c}\mathbf{a}^{i-1}\mathbf{b}$. This decomposition is unique up to a similarity transformation.

The triplet $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ defines a partial realization on the finite sequence $\begin{bmatrix} x_1 & \dots & x_s \end{bmatrix}$.

Theorem 1: Let $s = 2s_1 - 1$ with s_1 an integer such that $s_1 N \min(p, m) \geq \max(n_1, n_2)$. For $i \in \{1, \dots, s\}$, let (α_i, t_i) be nonzero scalars and let the matrices $\widehat{\mathbf{M}}_{i,\ell}$ satisfy

$$\widehat{\mathbf{M}}_{i,\ell} = t_i \mathbf{M}_{i,\ell}$$

with $\operatorname{rank}(\mathcal{H}(\mathbf{M}_{i,\ell})) = n_1$.
1) *If $\operatorname{rank}(\mathcal{H}(\alpha_i\widehat{\mathbf{M}}_{i,\ell})) = n_1$, then $\operatorname{rank}(\mathcal{H}(\alpha_i t_i)) = 1$.*
2) *If $\alpha_i t_i = \eta^{i-1}$ for a nonzero scalar η, then*

$$\operatorname{rank}(\mathcal{H}(\alpha_i\widehat{\mathbf{M}}_{i,\ell})) = n_1$$

Proof: We derive the proof using the contraposition. In the sequel, we denote $x_i = \alpha_i t_i$ and $\mathbf{X}_i = \mathbf{M}_{i,\ell}$.

Let $\delta \in \mathbb{N}$ such that $1 < \delta \leq s_1$ and suppose that

$$\operatorname{rank}(\mathcal{H}(x_i)) = \delta. \tag{19}$$

From Lemma 5, there exists a realization $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ of minimal degree δ with $\mathbf{a} \in \mathbb{R}^{\delta \times \delta}, \mathbf{b} \in \mathbb{R}^{\delta \times 1}, \mathbf{c} \in \mathbb{R}^{1 \times \delta}$ such that for $i = 1 \dots s, x_i = \mathbf{c}\mathbf{a}^{i-1}\mathbf{b}$. If every eigenvalue of $\mathbf{a}$ is 0, then $\mathbf{a}$ is nilpotent (via Cayley–Hamilton) which is forbidden by the assumption $x_i \neq 0$ for all i. Therefore, the matrix $\mathbf{a}$ has at least one nonzero eigenvalue.

We divide the proof in two cases. If $\mathbf{a}$ is diagonalizable, there exists an invertible matrix $\mathbf{P}$ such that $\mathbf{a} = \mathbf{P}\mathbf{D}\mathbf{P}^{-1}$ where $\mathbf{D}$ is a diagonal matrix containing the eigenvalues. All eigenvalues λ_i are distinct. Let $k \in \mathbb{N}$. We have

$$\mathbf{c}\mathbf{a}^k\mathbf{b}\mathbf{C}\mathbf{A}^k\mathbf{B} = \mathbf{C}(\mathbf{c}\mathbf{a}^k\mathbf{b})\mathbf{A}^k\mathbf{B}$$

$$= \mathbf{C}\mathbf{c}\mathbf{P}\mathbf{D}^k\mathbf{P}^{-1}\mathbf{b}\mathbf{A}^k\mathbf{B}.$$

Denote $\tilde{\mathbf{c}} = \mathbf{c}\mathbf{P}, \tilde{\mathbf{b}} = \mathbf{P}^{-1}\mathbf{b}$ and $r_i = \tilde{\mathbf{c}}_i\tilde{\mathbf{b}}_i \neq 0$. It yields:

$$\mathbf{c}\mathbf{a}^k\mathbf{b}\mathbf{C}\mathbf{A}^k\mathbf{B} = \mathbf{C}\tilde{\mathbf{c}}\mathbf{D}^k\tilde{\mathbf{b}}\mathbf{A}^k\mathbf{B}$$

$$= \mathbf{C}\sum_{i=1}^{\delta}\tilde{\mathbf{c}}_i\lambda_i^k\tilde{\mathbf{b}}_i\mathbf{A}^k\mathbf{B}$$

$$= \sum_{i=1}^{\delta}r_i\mathbf{C}(\lambda_i\mathbf{A})^k\mathbf{B} \tag{20}$$

Then, without loss of generality, consider $\delta = 2$. These Markov parameters are associated with the state-space matrices

$$\tilde{\mathbf{A}} = \begin{bmatrix} \lambda_1\mathbf{A} & \mathbf{0} \\ \mathbf{0} & \lambda_2\mathbf{A} \end{bmatrix}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ \mathbf{B} \end{bmatrix}, \quad \tilde{\mathbf{C}} = \begin{bmatrix} r_1\mathbf{C} & r_2\mathbf{C} \end{bmatrix}.$$

Let $\mathbf{W}_i$ be an eigenvector of $\mathbf{A}$. Then both $\begin{bmatrix} \mathbf{W}_i \\ \mathbf{0} \end{bmatrix}$ and $\begin{bmatrix} \mathbf{0} \\ \mathbf{W}_i \end{bmatrix}$ are eigenvectors of $\tilde{\mathbf{A}}$. The condition $\tilde{\mathbf{C}}\begin{bmatrix} \mathbf{W}_i \\ \mathbf{0} \end{bmatrix} = \mathbf{0}$ or $\tilde{\mathbf{C}}\begin{bmatrix} \mathbf{0} \\ \mathbf{W}_i \end{bmatrix} = \mathbf{0}$ is equivalently written with

$$r_i\mathbf{C}\mathbf{W}_i = \mathbf{0} \tag{21}$$

for $i \in \{1, 2\}$. Using the Popov–Belevitch–Hautus (PBH) test and with the assumption that the pair $(\mathbf{A}, \mathbf{C})$ is observable, it implies that $\mathbf{W}_i = \mathbf{0}$. Therefore, the pair $(\tilde{\mathbf{A}}, \tilde{\mathbf{C}})$ is observable following the PBH test. It follows that the rank of $\mathcal{H}(x_i\mathbf{X}_i)$ is strictly larger than n_1 and we have a contradiction.

Suppose now that the matrix $\mathbf{a}$ is not diagonalizable. There exists an invertible matrix $\mathbf{P}$ such that $\mathbf{a} = \mathbf{P}\mathbf{J}\mathbf{P}^{-1}$, where $\mathbf{J}$ is the Jordan matrix. The latter matrix is block-diagonal: each of the so-called Jordan blocks has a size equal to the algebraic multiplicity of the associated eigenvalue. Let q denote the number of blocks (also equal to the number of different eigenvalues) and h_i the multiplicity of the ith eigenvalue.

Without loss of generality, we assume that λ_i has multiplicity 2. The Jordan blocks $\mathbf{J}_i$ have then the following form:

$$\mathbf{J}_i = \begin{bmatrix} \lambda_i & 1 \\ 0 & \lambda_i \end{bmatrix}$$

It can be proven (using, e.g., induction) that $\mathbf{J}_i^k$ is expressed as

$$\mathbf{J}_i^k = \begin{bmatrix} \lambda_i^k & k\lambda_i^{k-1} \\ 0 & \lambda_i^k \end{bmatrix}.$$

The expression in (20) reads

$$\mathbf{c} a^k \mathbf{b} \mathbf{C} \mathbf{A}^k \mathbf{B} = \mathbf{C} \sum_{i=1}^{q} \tilde{c}_i \begin{bmatrix} \lambda_i^k & k\lambda_i^{k-1} \\ 0 & \lambda_i^k \end{bmatrix} \tilde{\mathbf{b}}_i \mathbf{A}^k \mathbf{B}$$

$$= \sum_{i=1}^{q} \tilde{\mathbf{C}}_i \tilde{\mathbf{A}}_i^k \tilde{\mathbf{B}}_i$$

where

$$\tilde{\mathbf{A}}_i = \begin{bmatrix} \lambda_i \mathbf{A} & \mathbf{A} \\ \mathbf{0} & \lambda_i \mathbf{A} \end{bmatrix}$$

$$\tilde{\mathbf{B}}_i = \begin{bmatrix} \tilde{b}_{i,1} \mathbf{B} \\ \tilde{b}_{i,2} \mathbf{B} \end{bmatrix}, \quad \tilde{\mathbf{C}}_i = \begin{bmatrix} \tilde{c}_{i,1} \mathbf{C} & \tilde{c}_{i,2} \mathbf{C} \end{bmatrix}.$$

The matrix $\mathbf{a}$ has at least one nonzero eigenvalue from the assumption that all x_i are different from 0. The following therefore assumes $\lambda_i \neq 0$.

The observability matrix associated to the triplet $(\tilde{\mathbf{A}}_i, \tilde{\mathbf{B}}_i, \tilde{\mathbf{C}}_i)$ is written as

$$\mathcal{V}_{n,i} = \begin{bmatrix} \tilde{c}_{i,(1)} \mathbf{C} & \tilde{c}_{i,(2)} \mathbf{C} \\ \tilde{c}_{i,(1)} \mathbf{C}(\lambda_i \mathbf{A}) & \tilde{c}_{i,(1)} \mathbf{C}\mathbf{A} + \tilde{c}_{i,(2)} \mathbf{C}(\lambda_i \mathbf{A}) \\ \vdots & \vdots \end{bmatrix} \quad (22)$$

whose rank is equal to the rank of $\frac{1}{\tilde{c}_{i,(1)}} \mathcal{V}_n$ for $\lambda_i \neq 0, \tilde{c}_{i,(1)} \neq 0$

$$\frac{1}{\tilde{c}_{i,(1)}} \mathcal{V}_{n,i} = \begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{C}(\lambda_i \mathbf{A}) & \mathbf{C}(\lambda_i \mathbf{A}) \\ \mathbf{C}(\lambda_i A)^2 & 2\mathbf{C}(\lambda_i \mathbf{A})^2 \\ \vdots & \vdots \end{bmatrix} \begin{bmatrix} \mathbf{I} & \frac{\tilde{c}_{i,(2)}}{\tilde{c}_{i,(1)}} \mathbf{I} \\ \mathbf{0} & \frac{1}{\lambda_i} \mathbf{I} \end{bmatrix}. \quad (23)$$

From Sylvester's inequality [7], the rank of $\frac{1}{\tilde{c}_{i,1}} \mathcal{V}_{n,i}$ is equal to the rank of the following matrix:

$$\begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{C}(\lambda_i \mathbf{A}) & \mathbf{C}(\lambda_i \mathbf{A}) \\ \mathbf{C}(\lambda_i \mathbf{A})^2 & 2\mathbf{C}(\lambda_i \mathbf{A})^2 \\ \vdots & \vdots \end{bmatrix}. \quad (24)$$

If the pair $(\mathbf{C}, \lambda_i \mathbf{A})$ is observable, then the matrices

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{C}(\lambda_i \mathbf{A}) \\ \mathbf{C}(\lambda_i \mathbf{A})^2 \\ \vdots \end{bmatrix}, \begin{bmatrix} \mathbf{I} & & & \\ & 2\mathbf{I} & & \\ & & \ddots & \\ & & & (s-1)\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{C} \\ \mathbf{C}(\lambda_i \mathbf{A}) \\ \mathbf{C}(\lambda_i \mathbf{A})^2 \\ \vdots \end{bmatrix}$$

are both full column rank. Owing to the zero block in the upper right part of the matrix (24), the rank of $\mathcal{V}_{n,i}$ is strictly larger than n_1 and again we have a contradiction.

The proof for the second bullet is as follows. If $\alpha_i t_i = \eta^{i-1}$ for all i, then

$$\mathcal{H}(\alpha_i \widehat{\mathbf{M}}_{i,\ell}) = \mathcal{H}(\mathbf{C}_1 (\eta \mathbf{A}_1)^{i-1} \mathbf{B}_1)$$

$$= \mathcal{O}_{1,s_1} \mathcal{C}_{1,s_1}$$

where $\mathcal{O}_{1,s_1}$ and $\mathcal{C}_{1,s_1}$ are, respectively, the observability and controllability matrices associated with the pairs $(\mathbf{C}_1, \eta \mathbf{A}_1)$ and $(\eta \mathbf{A}_1, \mathbf{B}_1)$. From Sylvester's inequality, the rank of $\mathcal{O}_{1,s_1} \mathcal{C}_{1,s_1}$ is equal to n_1, which proves that the rank of $\mathcal{H}(\alpha_i \widehat{\mathbf{M}}_{i,\ell})$ is n_1 under the conditions specified in the Theorem. $\square$

Corollary 2: With the notations introduced in Theorem 1, if $\mathrm{rank}(\mathcal{H}(\alpha_i \widehat{\mathbf{M}}_{i,\ell})) = n_1$, then, there exist nonzero scalars $(a_\alpha, b_\alpha, c_\alpha)$ such that

$$\alpha_i = \frac{c_\alpha a_\alpha^{i-1} b_\alpha}{t_i}$$

Proof: It follows from Theorem 1 by using $x_i = \alpha_i t_i$. $\square$

Corollary 3: With the notations introduced in Theorem 1, if $\mathrm{rank}(\mathcal{H}(\beta_i \widehat{\mathbf{M}}_{i,r})) = n_2$, then, there exist $(a_\beta, b_\beta, c_\beta)$ nonzero scalars such that

$$\beta_i = c_\beta a_\beta^{i-1} b_\beta t_i.$$

Proof: It follows from Theorem 1 adapted to $\mathcal{H}(\beta_i \widehat{\mathbf{M}}_{i,r})$ and by using $x_i = \frac{\beta_i}{t_i}$. $\square$

To summarize, the matrix $\mathcal{H}(\widehat{\mathbf{M}}_{i,\ell})$ is in general not low-rank as indicated in Theorem 1 because the scaling factor t_i is different for each factor matrix. The properties of the block-Hankel $\mathcal{H}(\alpha_i \widehat{\mathbf{M}}_{i,\ell})$ have then been studied. If the rank of the latter matrix is minimal, then

$$\mathrm{rank}(\mathcal{H}(\alpha_i t_i)) = 1.$$

There are, however, an infinite number of sequences $\boldsymbol{\alpha}$ for which such a condition is valid. In the next theorem, we analyze a rank minimization problem featuring both low-rank block-Hankel matrices $\mathcal{H}(\alpha_i \widehat{\mathbf{M}}_{i,\ell})$ and $\mathcal{H}(\beta_i \widehat{\mathbf{M}}_{i,r})$ and study the uniqueness when the scalings α_i, β_i are related with a bilinear constraint.

Theorem 2: The solution to the multicriteria feasibility problem

find $(\boldsymbol{\alpha}, \boldsymbol{\beta})$

s.t. $\{\mathrm{rank}(\mathcal{H}(\alpha_i \widehat{\mathbf{M}}_{i,\ell})) = n_1, \mathrm{rank}(\mathcal{H}(\beta_i \widehat{\mathbf{M}}_{i,r})) = n_2\}$

$$\forall i \in \{1, \ldots, s\}, \quad \alpha_i \beta_i = 1 \quad (25)$$

is not unique and feasible values for (25) are obtained for all α, β as described in Corollary 2 and 3 with the additional conditions that $a_\alpha a_\beta = 1$ and $c_\alpha c_\beta b_\alpha b_\beta = 1$.

Proof: Using Corollary 2 and 3, the above rank conditions are satisfied for all $i = 1 \ldots s$:

$$\alpha_i = \frac{c_\alpha a_\alpha^{i-1} b_\alpha}{t_i}$$

$$\beta_i = c_\beta a_\beta^{i-1} b_\beta t_i.$$

Replacing these expressions inside the bilinear constraint (25) yields

$$\alpha_i \beta_i = c_\alpha c_\beta (a_\alpha a_\beta)^{i-1} b_\alpha b_\beta = 1$$

which implies $a_\alpha a_\beta = 1$ and $c_\alpha c_\beta b_\alpha b_\beta = 1$. $\square$

The nonuniqueness of the feasibility problem in (25) will be studied further on in Section IV. A when estimating the state-space matrices and related to Lemma 3.

Remark 2: With the bilinear constraint $\alpha_i \beta_i = 1$, both α_i and β_i cannot be 0. Moreover, the scalars t_i, v_i are related with $t_i v_i = 1$ using Lemma 4. Therefore, both sequences $\alpha_i t_i$ and $\frac{\beta_i}{t_i}$ are nonzero which fulfills the condition expressed in Theorem 2.

B. A Bilinear Constrained Low-Rank Optimization

In this paragraph, we propose a method to estimate a set of vectors $\boldsymbol{\alpha}, \boldsymbol{\beta}$ using the constraints that have been derived in the previous paragraph and the estimates $\widehat{\mathbf{M}}_{i,\ell}, \widehat{\mathbf{M}}_{i,r}$ that have been obtained with the *QUARKS* identification. From (25), and without knowing *a priori* the system orders (n_1, n_2), we formulate a bilinear rank optimization problem

$$\min_{\boldsymbol{\alpha},\boldsymbol{\beta}} \quad \text{rank}(\mathcal{H}(\alpha_i \widehat{\mathbf{M}}_{i,\ell})) + \text{rank}(\mathcal{H}(\beta_i \widehat{\mathbf{M}}_{i,r}))$$

$$\text{s.t.} \quad \forall i \in \{1,\ldots,s\}, \quad \alpha_i \beta_i = 1 \tag{26}$$

where λ is a regularization parameter that trades between the low-rank priors and the stability constraint. The minimization problem (26) is bilinear and features the rank operator which is nonconvex. When convexifying the rank operator with the nuclear norm, it belongs to the class of multiconvex optimization problems as described in [30]. The works in [31] and [32] both propose an iterative algorithm. In this paper, use is made of a block-coordinate updates (BCU) algorithm with slack variables, [31]. The slack variables are used to relax the bilinear constraint and are denoted with q_i for $i \in \{1,\ldots,s\}$. The optimization (26) is transformed into

$$\min_{\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{q}} \quad \|\mathcal{H}(\alpha_i \widehat{\mathbf{M}}_{i,\ell})\|_\star + \|\mathcal{H}(\beta_i \widehat{\mathbf{M}}_{i,r})\|_\star + \mu \sum_{i=1}^{s} q_i^2$$

$$\text{s.t.} \quad \forall i \in \{1,\ldots,s\}, \quad \alpha_i \beta_i - 1 = q_i \tag{27}$$

where μ is a regularization parameter related to the slack variables. The higher μ is, the more weight is laid on setting $\mathbf{q}$ to 0. The optimization problem is solved iteratively with nonzero initial guesses. At each iteration, the optimization successively solves over $(\boldsymbol{\alpha}, \mathbf{q})$ and then $(\boldsymbol{\beta}, \mathbf{q})$. The number of variables is moreover only $2s$. Algorithm 2 details the steps. The notation $\mathcal{B}_{\widehat{\mathbf{M}}_r, \alpha, \mu}(\boldsymbol{\beta}, \mathbf{q})$ is introduced for the cost function in (27) when the optimization variables are $\boldsymbol{\beta}, \mathbf{q}$ only while $\boldsymbol{\alpha}$ is fixed, and with the regularization parameter μ. The regularization parameter μ shall be gradually increased throughout the iterations to

---

Algorithm 2: Summary of BCU.

Input: $\widehat{\mathbf{M}}_\ell, \widehat{\mathbf{M}}_r, \mu^{(0)}, d, \tau, \kappa_{\max}, e_{\max}$
Output: $\widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}}$
   /* Bilinear low-rank optimization   */
1: $\kappa \leftarrow 0$
2: **foreach** $i \leq s$ **do**
3:   $\alpha_i^{(\kappa)} \leftarrow 1$
4: **end**
5: **while** $\kappa \leq \kappa_{\max} - 1$ and $e > e_{\max}$ **do**
6:   $\boldsymbol{\beta}^{(\kappa+1)} \leftarrow \text{argmin } \mathcal{B}_{\widehat{\mathbf{M}}_r, \boldsymbol{\alpha}^{(\kappa)}, \mu^{(\kappa)}}(\boldsymbol{\beta}, \mathbf{q})$.
7:   $\boldsymbol{\alpha}^{(\kappa+1)} \leftarrow \text{argmin } \mathcal{B}_{\widehat{\mathbf{M}}_\ell, \boldsymbol{\beta}^{(\kappa+1)}, \mu^{(\kappa)}}(\boldsymbol{\alpha}, \mathbf{q})$.
8:   **if** $\text{mod}(\kappa, d) = 0$ **then**
9:     $\mu^{(\kappa+1)} \leftarrow \tau \mu^{(\kappa)}$
10:   **end**
11:   $e \leftarrow \sum_{i=1}^{s}(\alpha_i \beta_i - 1)^2$
12:   $\kappa \leftarrow \kappa + 1$
13: **end**
14: $\widehat{\boldsymbol{\alpha}} = \boldsymbol{\alpha}^{(\kappa^{\max})}$
15: $\widehat{\boldsymbol{\beta}} = \boldsymbol{\alpha}^{(\kappa^{\max})}$

---

ensure that the bilinear constraint (27) is met, [31]. We highlight the prominent role of the initial value $\mu^{(0)}$ for μ. If it is set too large when optimizing over $(\boldsymbol{\alpha}, \mathbf{q})$, respectively $(\boldsymbol{\beta}, \mathbf{q})$, the variable α_i is fixed to $1/\beta_i$ by the constraint. If it is set too low, α_i goes to 0 and q_i to -1. In that respect, $\mu^{(0)}$ plays the role of a regularization parameter whose optimal value is determined by grid search.

Standard alternating direction method of multipliers (ADMM) techniques apply here and a detailed analysis of the optimization updates are inspired from [33] using the linear operator framework to enforce the block-Hankel structure. The nuclear norm minimization is performed via singular value soft thresholding. The details are however not reproduced here to focus on the subspace algorithm.

C. Computational Complexity

The complexity of Algorithm 2 lies in solving each ADMM problem, $\mathcal{B}_{\widehat{\mathbf{M}}_r, \boldsymbol{\alpha}^{(\tau)}, \mu^{(\tau)}}(\boldsymbol{\beta}, \mathbf{q})$ and $\mathcal{B}_{\widehat{\mathbf{M}}_\ell, \boldsymbol{\beta}^{(\tau+1)}, \mu^{(\tau)}}(\boldsymbol{\alpha}, \mathbf{q})$. We assume that the number of iterations is independent from N. Prior to performing the ADMM updates, a Gramian matrix is computed based on the sequences $\{\widehat{\mathbf{M}}_{i,r}\}, \{\widehat{\mathbf{M}}_{i,\ell}\}$ with $\mathcal{O}(N^2)$

$$\boldsymbol{\mathcal{V}} = \begin{bmatrix} \mathbf{C}_r \mathbf{X}(1) \mathbf{C}_\ell^T & \ldots & \mathbf{C}_r \mathbf{X}(M) \mathbf{C}_\ell^T \\ \mathbf{C}_r \mathbf{A}_r \mathbf{X}(1)(\mathbf{C}_\ell \mathbf{A}_\ell)^T & \ldots & \vdots \\ \vdots & \ldots & \vdots \\ \mathbf{C}_r \mathbf{A}_r^{s-1} \mathbf{X}(1)(\mathbf{C}_\ell \mathbf{A}_\ell^{s-1})^T & \ldots & \mathbf{C}_r \mathbf{A}_r^{s-1} \mathbf{X}(M)(\mathbf{C}_\ell \mathbf{A}_\ell^{s-1})^T \end{bmatrix} \tag{28}$$

$$\boldsymbol{\mathcal{T}_u} = \begin{bmatrix} \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{M}_{1,r} \mathbf{U}(1) \mathbf{M}_{1,\ell}^T & \ldots & \mathbf{M}_{1,r} \mathbf{U}(M) \mathbf{M}_{1,\ell}^T \\ \sum_{i=0}^{1} \mathbf{M}_{i+1,r} \mathbf{U}(2-i) \mathbf{M}_{i+1,\ell}^T & & \vdots \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{s-2} \mathbf{M}_{i+1,r} \mathbf{U}(s-1-i) \mathbf{M}_{i+1,\ell}^T & \ldots & \sum_{i=0}^{s-2} \mathbf{M}_{i+1,r} \mathbf{U}(N_t - i) \mathbf{M}_{i+1,\ell}^T \end{bmatrix}. \tag{29}$$

flops. Two operations that appear in each of the above ADMM algorithm are detailed below. The number of unknowns in only $2s$ and hence, the cost of the primal variable update in each of the ADMM algorithm is not dominated by the matrix inversion but rather forming the matrices prior to solving the least squares, which scales with $\mathcal{O}(N^2)$ only. However, at each iteration of the ADMM algorithm, a SVD of the block-Hankel matrix $\mathcal{H}(\alpha_i^{(\tau)}\widehat{\mathbf{M}}_{i,\ell})$ (respectively, $\mathcal{H}(\beta_i^{(\tau)}\widehat{\mathbf{M}}_{i,r})$) is computed with a singular value soft thresholding, which is the bottleneck in Algorithm 2 as this operation scales with $\mathcal{O}(N^3)$.

D. Stability

The feasibility problem (25) (similarly, the rank minimization (26)) can lead to unstable *factored* models. For α_i as described in Corollary 2, the sequence $\alpha_i\widehat{\mathbf{M}}_{i,\ell}$ becomes

$$\alpha_i\widehat{\mathbf{M}}_{i,\ell} = c_\alpha\eta^{i-1}b_\alpha\mathbf{C}_1\mathbf{A}_1^{i-1}\mathbf{B}_1$$
$$= c_\alpha\mathbf{C}_1(\eta\mathbf{A}_1)^{i-1}b_\alpha\mathbf{B}_1$$

The stability of $\eta\mathbf{A}_1$ is not guaranteed depending on the value for η. Although this does not affect the estimation of $\boldsymbol{\alpha},\boldsymbol{\beta}$, we wish to recover two stable impulses built from the factor matrices. This is however done using an scaling/couterscaling approach of the matrices $\widehat{\mathbf{A}}_1$ and $\widehat{\mathbf{A}}_2$ as described in Section V.

V. ESTIMATION OF THE STATE-SPACE MATRICES

In this section, we estimate the state sequence and then perform a bilinear least-squares optimization on the MSSM model (7). Similarly to standard subspace identification methods, a data equation is first written in the form

$$\mathcal{Y} = \mathcal{V} + \mathcal{T}_u + \mathcal{E} \qquad (30)$$

where the block-Hankel matrix $\mathcal{Y} \in \mathbb{R}^{pNs \times NM}$ is as follows:

$$\mathcal{Y} = \begin{bmatrix} \mathbf{Y}(1) & \mathbf{Y}(2) & \dots & \mathbf{Y}(M) \\ \mathbf{Y}(2) & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \mathbf{Y}(s) & \mathbf{Y}(s+1) & \dots & \mathbf{Y}(N_t) \end{bmatrix}$$

with $M = N_t - s + 1$. The block-Hankel matrix $\mathcal{E}$ is similarly built from the noise matrices $\mathbf{E}(k)$. The matrices $\mathcal{V}$ and $\mathcal{T}_u$ are, respectively, defined in (28) and (29) are shown at the bottom of previous page.

The *matrix form* of the data equation features matrices of sizes of the order N (rather than N^2) but with no key structural properties like low rank as is enforced in standard subspace identification, [7]. For example, the matrix $\mathcal{V}$ is in general not low rank, hence the estimation of the state sequence is not straightforward.

In the following, the terms in $\mathcal{V}$ are embedded into a structured 3-D tensor, denoted with $\mathcal{A}$. Let $\varphi \in \mathbb{N}$ such that $\varphi pN > n_2$ and $\varphi N > n_1$. For all $k = 1\ldots M$, a slice $\mathcal{A}(:,:,k) \in \mathbb{R}^{p\varphi N \times \varphi N}$

is described with

$$\begin{bmatrix} \mathbf{C}_2\mathbf{X}(k)\mathbf{C}_1^T & \dots & \mathbf{C}_2\mathbf{X}(k)(\mathbf{C}_1\mathbf{A}_1^{\varphi-1})^T \\ \vdots & & \vdots \\ \mathbf{C}_2\mathbf{A}_2^{\varphi-1}\mathbf{X}(k)\mathbf{C}_1^T & \dots & \mathbf{C}_2\mathbf{A}_2^{\varphi-1}\mathbf{X}(k)(\mathbf{C}_1\mathbf{A}_1^{\varphi-1})^T \end{bmatrix}. \qquad (31)$$

We first justify the use of the tensor $\mathcal{A}$ before focusing on estimating the entries in the tensor $\mathcal{A}$. The rank properties of the tensor $\mathcal{A}$ are related to its matricizations.

Definition 5: Let $\mathcal{A} \in \mathbb{R}^{p\varphi N \times \varphi N \times M}$ be an order-3 tensor. The unfolding $\mathcal{A}_{(1)}$ is defined with

$$\mathcal{A}_{(1)} = \begin{bmatrix} \mathcal{A}(:,:,1) & \dots & \mathcal{A}(:,:,M) \end{bmatrix} \in \mathbb{R}^{p\varphi N \times \varphi NM}$$

Consequently, using (31) along with Definition 5:

$$\mathcal{A}_{(1)} = \begin{bmatrix} \mathbf{C}_2 \\ \mathbf{C}_2\mathbf{A}_2 \\ \vdots \\ \mathbf{C}_2\mathbf{A}_2^{\varphi-1} \end{bmatrix} \begin{bmatrix} \mathbf{X}(1)\mathbf{C}_1^T & \dots & \mathbf{X}(M)(\mathbf{C}_1\mathbf{A}_1^{\varphi-1})^T \end{bmatrix}.$$

The rank of the tensor unfolding $\mathcal{A}_{(1)}$ is equal to n_2

$$\text{rank}(\mathcal{A}_{(1)}) = n_2 < p\varphi N. \qquad (32)$$

Computing an SVD of $\mathcal{A}_{(1)}$ yields

$$\mathcal{A}_{(1)} = \mathbf{U}_2\mathbf{V}_2$$
$$\mathbf{U}_2 = \mathcal{O}_{\varphi,2}\mathbf{T}_2$$
$$\mathbf{V}_2 = \mathbf{T}_2^{-1}\begin{bmatrix} \mathbf{X}(1)\mathbf{C}_1^T & \dots & \mathbf{X}(M)(\mathbf{C}_1\mathbf{A}_1^{\varphi-1})^T \end{bmatrix} \qquad (33)$$

for a nonsingular $\mathbf{T}_2$. From (33) and more precisely from $\mathbf{U}_2$, the matrices $\mathbf{A}_2$ and $\mathbf{C}_2$ are estimated. By reshaping the matrix $\mathbf{V}_2$, we can write

$$\mathbf{H} = \begin{bmatrix} \mathbf{T}_2^{-1}\mathbf{X}(1)\mathbf{C}_1^T & \dots & \mathbf{T}_2^{-1}\mathbf{X}(1)(\mathbf{C}_1\mathbf{A}_1^{\varphi-1})^T \\ \vdots & & \vdots \\ \mathbf{T}_2^{-1}\mathbf{X}(M)\mathbf{C}_1^T & \dots & \mathbf{T}_2^{-1}\mathbf{X}(M)(\mathbf{C}_1\mathbf{A}_1^{\varphi-1})^T \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{T}_2^{-1}\mathbf{X}(1) \\ \vdots \\ \mathbf{T}_2^{-1}\mathbf{X}(M) \end{bmatrix} \begin{bmatrix} \mathbf{C}_1^T & \dots & (\mathbf{C}_1\mathbf{A}_1^{\varphi-1})^T \end{bmatrix}.$$

The following rank equality holds:

$$\text{rank}(\mathbf{H}) = n_1 < \varphi N. \qquad (34)$$

An SVD on the low-rank matrix $\mathbf{H}$ gives

$$\mathbf{H} = \mathbf{U}_1\mathbf{V}_1$$
$$\mathbf{U}_1 = \begin{bmatrix} \mathbf{T}_2^{-1}\mathbf{X}(1)\mathbf{T}_1 \\ \vdots \\ \mathbf{T}_2^{-1}\mathbf{X}(M)\mathbf{T}_1 \end{bmatrix}$$
$$\mathbf{V}_1 = \mathbf{T}_1^{-1}\begin{bmatrix} \mathbf{C}_1^T & \dots & (\mathbf{C}_1\mathbf{A}_1^{\varphi-1})^T \end{bmatrix}. \qquad (35)$$

Hence, $\mathbf{U}_1$ provides with an estimate for the state sequence up to two similarity transformations as presented initially in the proof of Lemma 3. The matrix $\mathbf{V}_1$ is equal to the extended

observability matrix $\mathcal{O}_{\varphi,1}$ up to a similarity transformation $\mathbf{T}_1$, and the matrices $\mathbf{A}_1$ and $\mathbf{C}_1$ can then be estimated. From both rank equalities (32) and (34), we conclude that two consecutive SVDs on, respectively, $\mathcal{A}_{(1)}$ and $\mathbf{H}$ enable to estimate the state sequence.

Remark 3: Contrary to the Multilinear-SVD presented in [34] in which an SVD of each of the three unfoldings are computed independently to estimate the column space of the factors, the first SVD (33) is performed on an unfolded tensor $\mathcal{A}_{(1)}$ while the second one (35) deals with a reduced size matrix $\mathbf{H}$ obtained from the right singular vectors in (33). The method proposed here is advantageous for extending the subspace algorithm to higher tensor orders.

Remark 4 (Stability): The matrices $\widehat{\mathbf{A}}_1$ and $\widehat{\mathbf{A}}_2$ are estimated up to a scaling/couterscaling factor which might lead to unstability. In this case, it is suggested to compute an eigenvalue decomposition of both matrices and scale the matrix with the largest spectral radius with a quantity strictly smaller than the inverse of its spectral radius and counterscale the other matrix.

In the following, we propose a method to estimate the tensor $\mathcal{A}$. In the noise-free case $\mathcal{E} = 0$, the terms $\mathbf{C}_2 \mathbf{A}_2^i \mathbf{X}(k)(\mathbf{C}_1 \mathbf{A}_1^j)^T$ for $i = j$ and located on the main block diagonal of $\mathcal{A}(:,:,k)$ are available from

$$\mathcal{V} \approx \mathcal{Y} - \widehat{\mathcal{T}_u} \qquad (36)$$

where $\widehat{\mathcal{T}_u}$ is obtained from $\mathcal{T}_u$ by replacing all $\mathbf{M}_{i,\ell}, \mathbf{M}_{i,r}$ with $\widehat{\mathbf{M}}_{i,\ell}, \widehat{\mathbf{M}}_{i,r}$.

Furthermore, the block entries away from the main block diagonal of $\mathcal{A}(:,:,k)$ feature cross terms such as $\mathbf{C}_2 \mathbf{A}_2^i \mathbf{X}(k)(\mathbf{C}_1 \mathbf{A}_1^j)^T$ for $i \neq j$. To cope with both the measurement noise and the estimation of cross terms, we introduce the so-called *virtual* outputs, denoted with $\mathbf{Y}^\sharp$, as follows:

$$\begin{cases} \mathbf{X}(k+1) = \mathbf{A}_2 \mathbf{X}(k)\mathbf{A}_1^T + \mathbf{B}_2 \mathbf{U}(k)\mathbf{B}_1^T \\ \mathbf{Y}_{g,h}^\sharp(k) = \mathbf{C}_{2,g}^\sharp \mathbf{X}(k)\mathbf{C}_{1,h}^{\sharp\ T} \end{cases} \qquad (37)$$

where $g, h \in \mathbb{N}$, $\mathbf{C}_{2,g}^\sharp = \mathbf{C}_2 \mathbf{A}_2^g$ and $\mathbf{C}_{1,h}^\sharp = \mathbf{C}_1 \mathbf{A}_1^h$.

In other words, $\mathbf{Y}_{g,h}^\sharp(k)$ is a virtual output associated with the set of Kronecker generators $\mathcal{S}_{g,h} = \{\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}_1, \mathbf{B}_2, \mathbf{C}_1 \mathbf{A}_1^h, \mathbf{C}_2 \mathbf{A}_2^g\}$. The virtual outputs $\{\mathbf{Y}_{g,h}^\sharp(k)\}_{k=1\ldots N_t}$ are not known when both g, h are not zero and are in the sequel approximated with a high-order FIR filter.

Let $z \in \mathbb{N}$. For all $k \geq z$

$$\mathbf{Y}_{g,h}^\sharp(k) \approx \sum_{i=0}^{z-1} \mathbf{C}_2 \mathbf{A}_2^{i+g} \mathbf{B}_2 \mathbf{U}(k-i-1)(\mathbf{C}_1 \mathbf{A}_1^{i+h} \mathbf{B}_1)^T$$

$$\approx \sum_{i=0}^{z-1} \mathbf{M}_{i+g+1,r} \mathbf{U}(k-i-1)\mathbf{M}_{i+h+1,\ell}^T. \qquad (38)$$

Using the estimates $\widehat{\mathbf{M}}_{i,\ell}, \widehat{\mathbf{M}}_{i,r}$ for the factored left and right coefficient matrices in the VARX model in Section III along with the estimates $\widehat{\alpha}, \widehat{\beta}$ in Section IV, the (38) reads

$$\mathbf{Y}_{g,h}^\sharp(k)$$
$$\approx \sum_{i=0}^{z-1} \widehat{\beta}_{i+g+1} \widehat{\mathbf{M}}_{i+g+1,r} \mathbf{U}(k-i-1)\widehat{\alpha}_{i+h+1} \widehat{\mathbf{M}}_{i+h+1,\ell}^T. \qquad (39)$$

$\mathbf{Y}_{0,0}^\sharp(k)$ is an FIR approximation of the noise free model (7), hence the coefficients $\widehat{\alpha}, \widehat{\beta}$ are not needed for computing the virtual output $\mathbf{Y}_{0,0}^\sharp(k)$.

We now investigate the requirements on the indices g, h to fill the tensor $\mathcal{A}$ according to (31). Both sequences $\widehat{\alpha}$ and $\widehat{\beta}$ have been estimated with s entries, therefore the (39) implies the following ranges for choosing the triplet (z, g, h):

$$z + g \leq s, \quad z + h \leq s.$$

For φ strictly smaller than s and larger than $\frac{n}{p}$ so that the rank strict inequalities (32) and (34) hold, the combinations of the pair (g, h) for filling $\mathcal{A}$ are obtained with
  1) $g = 0, h \in \{1, \ldots, \varphi - 1\}$
  2) $g = 0, h = 0$
  3) $g \in \{1, \ldots, \varphi - 1\}, h = 0$.

The maximum value of g is obtained for $g = \varphi - 1$, which implies $z + \varphi - 1 = s$. A total of $2\varphi - 1$ virtual outputs are available within the temporal range $\{z, \ldots, N_t\}$. For each of the associated subsystems in (37), a data equation in matrix form similar to (36) is written

$$\mathcal{Y}_{g,h}^\sharp = \mathcal{V}_{g,h} + \mathcal{T}_{g,h} \qquad (40)$$

$$\mathcal{V}_{g,h} = \begin{bmatrix} \mathbf{C}_{2,g}^\sharp \mathbf{X}(z)\mathbf{C}_{1,h}^{\sharp\ T} & \cdots & \mathbf{C}_{2,g}^\sharp \mathbf{X}(M_z)\mathbf{C}_{1,h}^{\sharp\ T} \\ \vdots & \ddots & \vdots \\ \mathbf{C}_{2,g}^\sharp \mathbf{A}_2^{\varphi-1} \mathbf{X}(z)(\mathbf{C}_{1,h}^\sharp \mathbf{A}_1^{\varphi-1})^T & \cdots & \mathbf{C}_{2,g}^\sharp \mathbf{A}_2^{\varphi-1} \mathbf{X}(M_z)(\mathbf{C}_{1,h}^\sharp \mathbf{A}_1^{\varphi-1})^T \end{bmatrix}$$

$$\mathcal{T}_{g,h} = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{C}_{2,g}^\sharp \mathbf{B}_2 \mathbf{U}(z)(\mathbf{C}_{1,h}^\sharp \mathbf{B}_1)^T & \cdots & \mathbf{C}_{2,g}^\sharp \mathbf{B}_2 \mathbf{U}(M_z)(\mathbf{C}_{1,h}^\sharp \mathbf{B}_1)^T \\ \sum_{i=0}^{1} \mathbf{C}_{2,g}^\sharp \mathbf{A}_2^i \mathbf{B}_2 \mathbf{U}(z+1-i)(\mathbf{C}_{1,h}^\sharp \mathbf{A}_1^i \mathbf{B}_1)^T & \cdots & \vdots \\ \vdots & & \vdots \\ \sum_{i=0}^{\varphi-2} \mathbf{C}_{2,g}^\sharp \mathbf{A}_2^i \mathbf{B}_2 \mathbf{U}(z+\varphi-2-i)(\mathbf{C}_{1,h}^\sharp \mathbf{A}_1^i \mathbf{B}_1)^T & \cdots & \sum_{i=0}^{\varphi-2} \mathbf{C}_{2,g}^\sharp \mathbf{A}_2^i \mathbf{B}_2 \mathbf{U}(N_t-i)(\mathbf{C}_{1,h}^\sharp \mathbf{A}_1^i \mathbf{B}_1)^T \end{bmatrix}$$
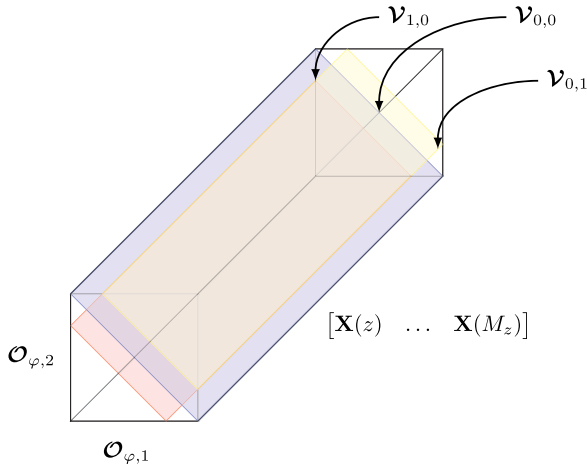
Fig. 1. Schematic of the tensor $\mathcal{A}$. The position of the observability matrices obtained by writing the data equation for each virtual system are indicated for (g, h) equal to $(0, 1), (0, 0), (1, 0)$.
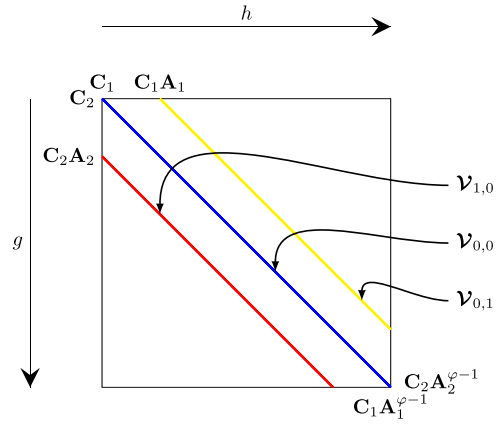

Fig. 2. Schematic of a slice $\mathcal{A}(:, :, k)$. The position of the observability matrices obtained by writing the data equation for each virtual system are indicated for (g, h) equal to $(0, 1), (0, 0), (1, 0)$.

where

$$\mathcal{Y}^\sharp_{g,h} = \begin{bmatrix} \mathbf{Y}^\sharp_{g,h}(z) & \mathbf{Y}^\sharp_{g,h}(z+1) & \ldots & \mathbf{Y}^\sharp_{g,h}(M_z) \\ \mathbf{Y}^\sharp_{g,h}(z+1) & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \mathbf{Y}^\sharp_{g,h}(z+\varphi-1) & \ldots & \ldots & \mathbf{Y}^\sharp_{g,h}(N_t) \end{bmatrix}.$$

$$M_z = N_t - \varphi + 1$$

The matrices $\mathcal{V}_{g,h}$ and $\mathcal{T}_{g,h}$ are, respectively, defined in (28) and (29). The matrices $\mathcal{V}_{g,h}$ are estimated with

$$\mathcal{V}_{g,h} \approx \mathcal{Y}^\sharp_{g,h} - \widehat{\mathcal{T}}_{g,h} \tag{41}$$

and are contained in the tensor $\mathcal{A}$. A diagonal slice of the tensor $\mathcal{A}$ is defined as follows:

$$\mathcal{A}((i-1)pN+1 : ipN, (i-1)N+1 : iN, :).$$

Then each diagonal slice of the tensor $\mathcal{A}$ contains a matrix $\mathcal{V}_{g,h}$ as illustrated in Figs. 1 and 2. For example, the block diagonal terms for each $\mathcal{A}(:, :, k)$ are contained in $\mathcal{V}_{0,0}$, the block subdiagonal terms are contained in $\mathcal{V}_{1,0}$, the block super-diagonal terms are contained in $\mathcal{V}_{0,1}$, etc. To summarize, one diagonal slice is provided by one data equation (with $\mathcal{E} = 0$) corresponding to one MSSM (37). The main block diagonal can be computed without the bilinear low-rank optimization in Section IV contrary to all other slices that use the estimates $(\widehat{\alpha}, \widehat{\beta})$ from the BCU estimation.

From the estimates of the state sequence $\widehat{\mathbf{X_T}}(k) = \mathbf{T}_2^{-1}\mathbf{X}(k)\mathbf{T}_1$, the following bilinear least squares is formulated to recover the matrices $\mathbf{B}_1, \mathbf{B}_2$

$$\min_{\mathbf{B}_1, \mathbf{B}_2} \sum_{k=z+1}^{M_z-1} \|\left(\widehat{\mathbf{X_T}}(k+1) - \mathbf{A}_2\widehat{\mathbf{X_T}}(k)\mathbf{A}_1^T\right) - \mathbf{B}_2\mathbf{U}(k)\mathbf{B}_1^T\|_F^2. \tag{42}$$

The minimization problem (42) is solved using ALS and starting with a random nonzero initial guess.

Algorithm 3: Estimation of the State Sequence.

Input: $\{\mathbf{u}_k\}_{1:N_t}, \{\mathbf{y}_k\}_{1:N_t}, \widehat{\alpha}, \widehat{\beta}, \widehat{\mathbf{M}}_\ell, \widehat{\mathbf{M}}_r, z, \varphi$
Output: $\widehat{\mathbf{A}}_1, \widehat{\mathbf{A}}_2, \widehat{\mathbf{B}}_1, \widehat{\mathbf{B}}_2, \widehat{\mathbf{C}}_1, \widehat{\mathbf{C}}_2$
1: **foreach** $\eta = 1 : 2\varphi - 1$ **do**
2:    **if** $\eta < \varphi$ **then**
3:      | $g = 0, h = \eta$
4:    **else if** $\eta = \varphi$ **then**
5:      | $g = 0, h = 0$
6:    **else if** $\eta > \varphi$ **then**
7:      | $g = \eta - \varphi, h = 0.$
8:
9:    **foreach** $k = z + 1 : N_t$ **do**
10:      Compute the virtual outputs with (39)
11:    **end**
12:    Compute $\mathcal{V}_{g,h}$ with (41)
    /* Fill-in the tensor $\mathcal{A}$ /*
13:    **foreach** $k = z : N_t$ **do**
14:      | Fill block-diag$(\mathcal{A}(:, :, k), \eta - \varphi)$ with the kth block column of $\mathcal{V}_{g,h}$ according to (31).
15:    **end**
16: **end**
  /* Compute the state sequence   /*
17: Form the unfolding $\mathcal{A}_{(1)}$ and compute the SVD (33)
18: Estimate $\widehat{\mathbf{A}}_2, \widehat{\mathbf{C}}_2$
19: Form $\mathbf{H}$ and compute the SVD (35)
20: Estimate $\widehat{\mathbf{A}}_1, \widehat{\mathbf{C}}_1$
  /* Compute the input state-space matrices   /*
21: Solve (42) iteratively with ALS and estimate $\widehat{\mathbf{B}}_1, \widehat{\mathbf{B}}_2$

The steps for estimating the state sequence are summarized in Algorithm 3.

Computing the virtual outputs requires $(2\varphi - 1)N^3N_tz$ flops and therefore, scales with $\mathcal{O}(N^3N_t)$. The first and second SVD cost, respectively, $\mathcal{O}(N^3M)$ and $\mathcal{O}(\widehat{n_2}N^2M)$. Last, solving the

bilinear least-squares in (42) requires $N^3 M_z$ flops. The overall computational complexity for Algorithm 3 is $\mathcal{O}(N^3 N_t)$.

VI. NUMERICAL EXAMPLE FOR LARGE-SCALE ATMOSPHERIC TURBULENCE PREDICTION

A. The Model

The subspace algorithm is now illustrated with an adaptive optics application. When a light beam with an originally flat wavefront passes through a turbulent medium, the wavefront gets distorted both in time and space. Adaptive optics is the system that corrects in real-time atmospheric aberrations to obtain high-resolution images. A description of the signal-generation model is described in [16].

We aim here at illustrating the subspace identification of Kronecker state-space models in innovation form defined with

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{K}\mathbf{e}(k) \\ \mathbf{s}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{e}(k) \end{cases}$$

where $\mathbf{e}(k)$ is a zero-mean white Gaussian noise sequence. The prediction model reads

$$\begin{cases} \widehat{\mathbf{x}}(k+1|k) = \tilde{\mathbf{A}}\widehat{\mathbf{x}}(k|k) + \mathbf{K}\mathbf{s}(k) \\ \mathbf{s}(k) = \mathbf{C}\widehat{\mathbf{x}}(k|k) + \mathbf{e}(k) \end{cases} \tag{43}$$

where we assume that the matrices $\tilde{\mathbf{A}} = \mathbf{A} - \mathbf{KC}$, $\mathbf{K}$, and $\mathbf{C}$ have a Kronecker rank equal to 1, hence giving rise to the MSSM as follows:

$$\begin{cases} \widehat{\mathbf{X}}(k+1|k) = \tilde{\mathbf{A}}_2\widehat{\mathbf{X}}(k|k)\tilde{\mathbf{A}}_1^T + \mathbf{K}_2\mathbf{S}(k)\mathbf{K}_1^T \\ \mathbf{S}(k) = \mathbf{C}_2\widehat{\mathbf{X}}(k|k)\mathbf{C}_1^T + \mathbf{E}(k) \end{cases} \tag{44}$$

In the following simulation, we evaluate the performance of Algorithm 1 for varying size of the lenslet-array and compare it with the SSARX method developed in [35].

Remark 5 (Structured innovation forms): A drawback that appears when considering the state-space model in innovation form of networked systems such as [11] or the Kronecker-based subspace algorithm is that the structure of the Kalman gain is restricted to the structure of the matrix $\mathbf{A}$. In our example, $\tilde{\mathbf{A}}$ and $\mathbf{K}$ are assumed to have a Kronecker rank equal to 1.

We investigate here, the performances for 20 independent turbulence realizations. For each of these, the following methods are compared:

1) **SSARX**: The centralized identification scheme used for adaptive optics purposes in [35] and tackles the model (43). The number of temporal samples is set to the minimum required for the method, $4N^2 s + 2s$. The integer s is set to 15.
2) **K4SID (Algorithms 1 + 2 + 3)** The detailed set of parameters is found below.
3) **K4SID+MLDS**: The estimates obtained with K4SID are used for initializing the non-linear expectation-maximization algorithm for handling multilinear dynamical systems, [20], [36]. $10sN$ temporal samples are used for identification and ten iterations are computed. However, only small sizes of networks with $N \leq 10$ could be handled because of the high computational

complexity of MLDS. Note that we have computed a suboptimal model from K4SID by fixing the system orders n_1, n_2 to lower values than the optimal ones such that MLDS could handle the optimization, i.e, n_1, n_2 are set to $2N$.

We summarize the building blocks of the algorithm K4SID along with the chosen parameters used in the simulations below.

1) The *QUARKS* identification (Algorithm 1): The number of temporal points in the identification set (for Kronecker-based models) is $10sN$. The initial guesses for the matrices $\mathbf{M}_{i,\ell}$ are chosen following a Gaussian distribution with zero mean and identity covariance matrix. A maximum of ten iterations is fixed along with a stopping bound $e_{\max}$ set to 10^{-5}. The temporal stability of the *QUARKS* model is ensured by fitting a DC-kernel

$$p_{t,(m,n)} = e^{-\eta|m-n|}e^{-\xi/2(m+n)}$$

for $m, n = 1 \ldots s$. The optimization is performed for different hyperparameters λ, η, ξ; the optimal set is found by random search, [28] with ten runs.

2) The bilinear low-rank optimization in (27) (Algorithm 2): The initial guesses for $\alpha^{(0)}$, $\beta^{(0)}$ are chosen equal to 1. The regularization parameter μ to gradually set the slack variables to 0 is defined as follows. The initial value for $\mu^{(0)}$ is taken within the range linspace$(1, 50, 5)$ and its influence is studied in Section F. Although, there is no clear theoretical guideline on how to update $\mu^{(\ell)}$, a heuristic rule for its update is to increase it gradually, for example multiply it by five every four iterations. The necessity for a similar update rule is found in [32]. A maximum of 15 iterations is fixed and the stopping bound $e_{\max}$ is set to 10^{-3}.

3) State-sequence estimation (Algorithm 3): The integers φ and z are, respectively, fixed to $\lfloor \frac{s+1}{2} \rfloor$ and $\varphi - 1$. The system orders $\widehat{n_1}$ and $\widehat{n_2}$ correspond to the index of the singular value that in logarithm is closest to the logarithmic mean of the maximum and minimum singular values for both $\widehat{n_1}$ (35) and $\widehat{n_2}$ (33). The maximum number of iterations in the ALS for minimizing (42) is set to ten.

The quality criteria is the variance accounted for (VAF) between the slopes measurements $\mathbf{s}_{i,j}$ and the predicted $\widehat{\mathbf{s}_{i,j}}$ and is defined with

$$\max\left(0, \left(1 - \frac{\frac{1}{N_t}\sum_{k=1}^{N_t} \|\mathbf{s}_{k,(i,j)} - \widehat{\mathbf{s}}_{k,(i,j)}\|_2^2}{\frac{1}{N_t}\sum_{k=1}^{N_t} \|\mathbf{s}_{k,(i,j)}\|_2^2}\right) \times 100\right)$$

for $N_t = 5 \times 10^3$ time samples from a validation set independent from the identification set. The VAF is computed for each sensor channel independently, and the mean is taken over the whole measurement grid afterwards. The fit between two identical signals $\mathbf{s}_{i,j}$ and $\widehat{\mathbf{s}_{i,j}}$ reaches 100%.

B. The Simulation Parameters

The experiments have been carried out using MATLAB R2015b on a desktop computer with a CPU Intel Xeon E5-1620V3/3.5 GHz with 24GB of RAM. The dimensions of the problem are summarized in Table I.

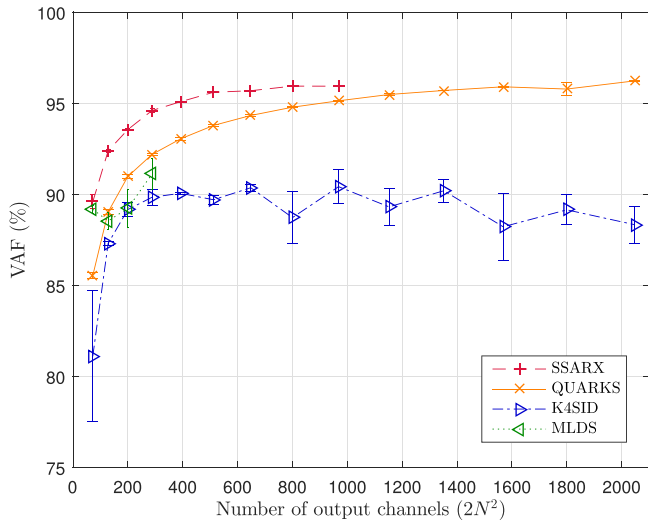|	
Fried parameter, r_0	$0.2\,[m]$
Outer scale, L_0	$20\,[m]$
Number of lenslet in 1D, N	$6:2:32$
Number of sensor outputs, $2N^2$	From 72 to 2048
Telescope aperture, D	$N/4\,[m]$
Number of phase points per lenslet, n_ϕ	4
Frozen-flow shift in phase points per sampling time	3
Upper bound for system order, s	15
Signal-to-Noise Ratio, SNR	$20[dB]$
Number of Monte-Carlo simulations	20

Fig. 3. VAF (%) on validation data as a function of the number of outputs. No model was computed for SSARX for $N > 20$.

Fig. 4. Storage complexity as a function of the size of the network, in loglog scale. The linear model plotted corresponds to $log10(\text{Storage complexity}) = a \times log10(N) + b$. The dashed part of the model corresponds to extrapolated data, as no model could be identified with these sizes.

C. Analyzing the Prediction Error

Fig. 3 displays the VAF on validation data for the four algorithms as a function of the total number of outputs, $2N^2$. We stress that we are not aiming at reaching lower prediction errors than SSARX as structural assumptions are made on the matrices. However, we show that the proposed Kronecker-based modeling handles networks of much larger sizes and with a slight decrease of performance w.r.t the centralized version (when the latter model can be computed).

When $N \leq 20$ (that is, $2N^2 \leq 800$), SSARX obtains lower prediction errors than K4SID. A centralized identification could no longer be carried out for $N > 20$ because of lack of memory, which is a well-known problem as explained in Section II and in, e.g., [9]. The QUARKS estimation reaches SSARX-like performances and handles many more outputs. More temporal samples would improve the performance with some impact on the scalability. The performances using K4SID are lower than QUARKS because the low-rank bilinear algorithm (Algorithm 2) does not in general converge to the global minimum of (26). The difficulties are twofold: both the rank operator and the bilinear constraint have been relaxed to solve a sequence of convex minimizations rather than a rank minimization problem. The estimates obtained with K4SID serve as good initial values for
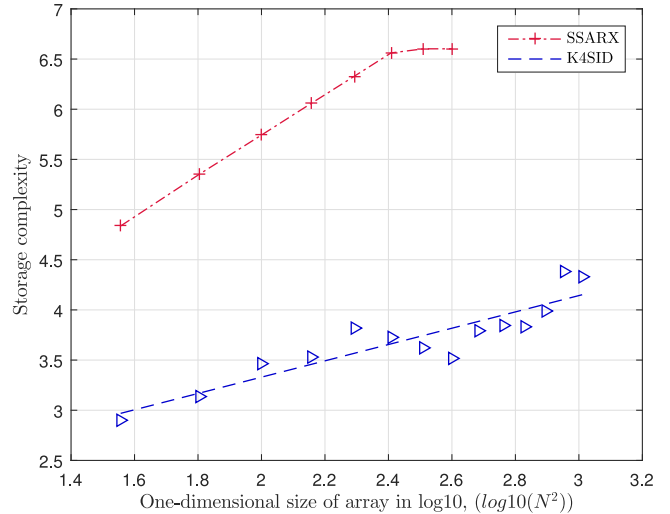
further optimization using output-error algorithms for Kronecker models as highlighted with the method that combines K4SID and MLDS (in green). However, the latter is computationally cumbersome. Increasing the system order would result in higher performances but in much longer optimization times as MLDS scales with the third power of the global order, that is $\mathcal{O}(N^6)$ for the example considered. Systems could no longer be identified with MLDS for sizes strictly larger than 10×10 and this is corroborated with the timing measurements in Section VI-E.

D. Storage Complexity

The storage is defined as the number of entries to construct the state-transition matrix $\mathbf{A}$, that is, n^2 in the centralized case and $n_1^2 + n_2^2$ in the Kronecker model. We analyze these results further by plotting the storage as a function of the size, which are directly related to the system orders.

Fig. 4 illustrates a dependency of the storage complexity with $N^{1.86}$ for global models while it is only $N^{0.81}$ for the Kronecker-structured model. The order stops increasing for SSARX when $N > 20$ as reaching a user-chosen upper-bound, $n = 2 \cdot 10^3$.

E. Timing Experiments

In this section, we investigate how the computational time for the identification algorithms evolves with N. We lay the emphasis on relative results rather than absolute as the latter are very much hardware dependent. The SSARX algorithm consists mainly of a QR decomposition, a SVD and a least squares, while the Kronecker-based methods contains many loops in the QUARKS identification, the bilinear low-rank algorithm and the state-sequence estimation. Consequently, the performances would benefit from a C-based implementation. A similar observation is done for the MATLAB code used for MLDS, [36]. We
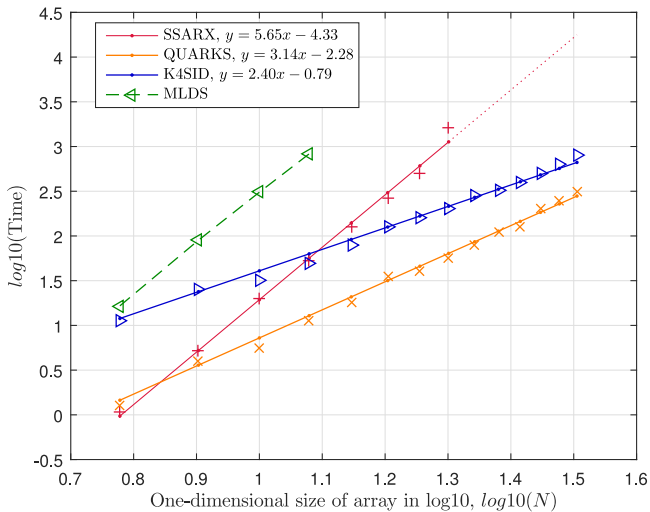
Fig. 5. Computational time of the model identification as a function of the size of the network, in loglog scale. The linear model plotted corresponds to $\log 10(Time) = a \times \log 10(N) + b$. The dashed part of the model corresponds to extrapolated data, as no model could be identified with these sizes.
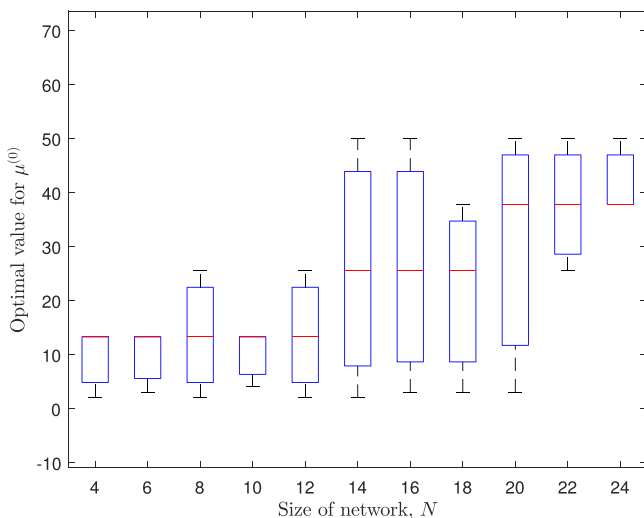

Fig. 6. Optimal value of the initial value for the regularization parameter, μ, in Algorithm 2 as a function the network size, N.

nonetheless focus on the difference of scaling capabilities of the three algorithms in Fig. 5. The time for SSARX to identify a model scales with $N^{5.65}$ while it is only $N^{2.40}$ for K4SID. The M-step in the MLDS computes a Kalman smoother, which scales at the very least with $\mathcal{O}(N^6)$ as no Kronecker structure could be exploited in every step and large matrices were used for computations. For example, it requires computing the inverse of the global state covariance matrix at each iteration for each time sample. Consequently, the computational time increases sharply even for moderate sizes of networks.

F. Regularization Parameter $\mu^{(0)}$ in Algorithm 2

Algorithm 2 for estimating the parameters $\widehat{\alpha}$ and $\widehat{\beta}$ requires an initial value for the regularization parameter, $\mu^{(0)}$. Its value impacts whether the bilinear constraint $\alpha_i \beta_i = 1$ is met. If it is

set too low, then both α and β are estimated with $\mathbf{0}$, and the bilinear constraint is not met. Therefore, we look for an optimal $\mu^{(0)}$ by grid search. We call an optimal value for $\mu^{(0)}$ the one that minimizes the prediction error for the method K4SID for a particular set of data and network size.

The optimal values are plotted in Fig. 6 and it shows that lower $\mu^{(0)}$ are expected when N is small, while it should be set higher when increasing the size of the network. Studying a systematic way to determine $\mu^{(0)}$ is outside the scope of this paper.

VII. CONCLUSION

In this paper, we presented a new framework to analyze large scale networks and identify with $\mathcal{O}(N^3 N_t)$ complexity the state-space matrices when they exhibit a Kronecker rank-1 structure. The algorithm consists of first identifying a QUARKS model in which we estimate the left and right factor matrices up to an unknown parameter that is different for each factored Markov parameter. Next, we formulated some low-rank conditions on a block-Hankel matrix such that the left and right factored impulse responses are retrieved up to a scaling factor. This subproblem was tackled by formulating a regularized bilinear low-rank optimization. A proposal has been made to use a block-coordinate descent algorithm with slack variables that is solved iteratively and gradually ensures that the bilinear constraint is met. We estimated the state sequence using two consecutive SVD on a tensor which enabled to estimate the Kronecker generators from an ALS on the matrix state-space model. The benefits of large-scale modeling with the Kronecker structure have been illustrated with an adaptive optics example and are threefold. First, the Kronecker-based subspace algorithm handles larger systems than the benchmark SSARX allow. Second, although the method we propose leads to a higher prediction-error than the centralized version, the number of time samples required for using the SSARX method increases with N^2. Last, timing experiments (MATLAB-based) have shown a dependency with $N^{5.65}$ for SSARX instead of $N^{2.40}$ for K4SID.

The estimates obtained with the subspace algorithm can be further refined by using them as initial estimates for Kronecker-based output-error optimization algorithms.

We have not investigated in this paper the identification of state-space models in which the matrices (especially $\mathbf{B}$, $\mathbf{C}$) have a Kronecker rank strictly larger than one, which could help to reduce further on the orders n_1, n_2 while maintaining the same prediction-error performances. Moreover, the algorithm has been presented for 2-D dynamical systems with temporal dynamics and can be generalized to higher dimensions by using a Kronecker product of multiple matrices instead of only two, in which case larger compression rates are achieved and a larger number of inputs and outputs can be handled.

REFERENCES

[1] R. P. Roesser, "A discrete state-space model for linear image processing," *IEEE Trans. Autom. Control*, vol. AC-20, no. 1, pp. 1–10, Feb. 1975.

[2] E. Fornasini and G. Marchesini, "Doubly-indexed dynamical systems: State-space models and structural properties," *Math. Syst. Theory*, vol. 12, pp. 59–72, 1978.

[3] J. A. Ramos, A. Alenany, H. Shang, and P. J. L. Dos Santos, "Subspace algorithms for identifying separable-in-denominator two dimensional systems with deterministic inputs," *IET Control Theory Appl.*, vol. 5, no. 15, pp. 1748–1765, Oct. 2011.

[4] J. A. Ramos and G. Mercère, "Image modeling based on a 2D stochastic subspace system identification algorithm," *Multidimens. Syst. Signal Process.*, pp. 1–33, 2016.

[5] A. Alenany, G. Mercère, and J. A. Ramos, "Subspace identification of 2-D CRSD roesser models with deterministic-stochastic inputs: A state computation approach," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 3, pp. 1108–1115, May 2017.

[6] C. Kulcsár, H-F Raynaud, C. Petit, and J.-M. Conan, "Minimum variance prediction and control for adaptive optics," *Automatica*, vol. 48, pp. 1939–1954, 2012.

[7] M. Verhaegen and V. Verdult, "Filtering and system identification: A least-squares approach," Cambridge, U.K.: Cambridge. Univ. Press, 2007.

[8] M. Benzi *et al.*, "Exploiting hidden structure in matrix computations: Algorithms and applications," *(Lecture Notes in Mathematics 2173, CIME Found. Subseries).* Berlin, Germany: Springer, 2015.

[9] K. Hinnen, "Data-driven optimal control for adaptive optics," PhD dissertation, Delft Center Syst. Control, TU Delft, Delft, The Netherlands, 2005.

[10] A. Haber and M. Verhaegen, "Subspace identification of large-scale interconnected systems," arXiv:1309.5105, 2013.

[11] C. Yu, M. Verhaegen, and A. Hansson, "Subspace identification of local systems in 1D homogeneous networks," *IEEE Trans. Autom. Control*, vol. 62, no, 10, pp. 2754–2759, Apr. 2014.

[12] C. Yu and M. Verhaegen, "Subspace identification of distributed clusters of homogeneous systems," *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 463–468, Jan. 2017.

[13] C. Yu and M. Verhaegen, "Subspace identification of individual systems operating in a network (SI^{2}ON)," *IEEE Trans. Autom. Control*, vol. 63, no. 4, pp. 1120–1125, Apr. 2018.

[14] P. Massioni and M. Verhaegen, "Subspace identification of circulant systems," *Automatica*, vol. 44, no. 11, pp. 2825–2833, 2008.

[15] P. Massioni and M. Verhaegen, "Subspace identification of distributed, decomposable systems," in *Proc. Joint 48th IEEE Conf. Decision Control, 28th Chin. Control Conf.*, 2009, pp. 3364–3369.

[16] B. Sinquin and M. Verhaegen, "QUARKS: Identification of large-scale Kronecker vector autoregressive models," *IEEE Trans. Autom. Control*, arXiv:1609.07518 [cs.SY], 2018.

[17] C. Hansen, J. G. Nagy, and D. P. O'Leary, "Deblurring images: Matrices, spectra, and filtering," *Fundamentals of Algorithms 3*. Philadelphia, PA, USA: SIAM, 2006.

[18] M. Boussé, O. Debals, and L. De Lathauwer, "Tensor-based large-scale blind system identification using segmentation," *IEEE Trans. Signal Process.*, vol. 65, no. 21, pp. 5770–5784, Nov. 2017.

[19] R. T. Suryaprakash, B. E. Moore, and R. R. Nadakuditi, "Algorithms and performance analysis for estimation of low-rank matrices with Kronecker structured singular vectors," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 3776–3780.

[20] M. Rogers, L. Li, and S. Russell, "Multilinear dynamical systems for tensor time series," in *Proc. Adv. Neural Inform. Process. Syst.*, 2013, pp. 2634–2642.

[21] C. F. van Loan, "The ubiquitous Kronecker product," *J. Comput. Appl. Mathematics*, vol. 123, pp 85–100, 2000.

[22] L. Grasedyck, D. Kressner, and C. Tobler, "A literature survey of low-rank approximation techniques," in *GAMM-Mitteilungen*, vol. 36, no. 1, pp. 53–78, 2013.

[23] N. Halko, P. G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix," *Soc. Ind. Appl. Math. Rev.*, vol. 53, no. 2, pp. 217–288, 2011.

[24] A. Chiuso, "The role of vector autoregressive modeling in predictor-based subspace identification," *Automatica*, vol. 43, pp. 1034–1048, 2007.

[25] T. Knusden, "Consistency analysis of subspace identification methods based on a linear regression approach," *Automatica*, vol. 37, no. 1, pp. 81–89, 2001.

[26] T. Chen, H. Ohlsson, and L. Ljung, "On the estimation of transfer functions, regularizations ANF Gaussian processes-Revisited," *Automatica*, vol. 48, pp. 1525–1535, 2012.

[27] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, pp. 657–682, 2014.

[28] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, 2012.

[29] W. B. Gragg and A. Lindquist, "On the partial realization problem," *Linear Algebra Appl.*, vol. 50, pp. 277–319, 1983.

[30] J. Nocedal and S. Wright, "Numerical optimization," Berlin, Germany: Springer Science & Business Media, 2006.

[31] Y. Xu and W. Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *SIAM J. Imag. Sci.*, vol. 6, no. 3, pp. 1758–1789, 2016.

[32] R. Doelman and M. Verhaegen, "Sequential convex relaxation for convex optimization with bilinear matrix equalities," in *Proc. Eur. Control Conf.*, 2016, pp. 1946–1951.

[33] M. Verhaegen and A. Hansson, "N2SID: Nuclear norm subspace identification of innovation models," *Automatica*, vol. 72, pp. 57–63, 2016.

[34] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *Siam J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.

[35] K. Hinnen, M. Verhaegen, and N. Doelman, "A data-driven $\mathcal{H}_2$-optimal control approach for adaptive optics," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 3, pp. 381–395, May 2008.

[36] L. Li, "Multilinear dynamical systems." [Online] Available: http://www.cs.cmu.edu/leili/mlds/, Accessed on: Oct. 2017.

Baptiste Sinquin received the electrical engineering degree from Ecole Centrale Lyon, Écully, France, in 2014. He is currently working toward the Ph.D. degree at the Delft Center for Systems and Control, TU Delft, Delft, The Netherlands.

His research interest include the identification of multi-dimensional and large-scale systems, and validation of the derived tensor-based identification and control methods in a laboratory environment dedicated for large-scale adaptive optics.

Michel Verhaegen received the Engineering degree in aeronautics from the Delft University of Technology, Delft, The Netherlands, in 1982 and the Doctoral degree in applied sciences from the Catholic University Leuven, Leuven, Belgium, in 1985.

From 1985 to 1994, he was a Research Fellow with the U.S. National Research Council (NRC), NASA Ames Research Center in California, and with the Dutch Academy of Arts and Sciences, Network Theory Group, Delft University of Technology. From 1994 to 1999, he was an Associate Professor with the Control Laboratory, Delft University of Technology and was appointed as a Full Professor with the Faculty of Applied Physics, University of Twente, The Netherlands, in 1999. His main research interest is the interdisciplinary domain of numerical algebra and system theory. In this field he has authored and coauthored more than 130 papers. Current activities focus on the transfer of knowledge about new identification, fault tolerant control and data-driven controller design methodologies to industry. Application areas include adaptive optics.

Prof. Verhaegen joint the University of Delft, in 2001, and is now a member of the Delft Center for Systems and Control. He has held short sabbatical leaves at the University of Uppsala, McGill, Lund, and the German Aerospace Research Center in Munich and is participating in several European Research Networks.