A Tensor-Based Method for Large-Scale Blind System Identification Using Segmentation

Martijn Boussé
Department of Electrical Engineering
ESAT, KU Leuven, Leuven, Belgium.
Email: martijn.bousse@esat.kuleuven.be

Otto Debals
Department of Electrical Engineering
ESAT, KU Leuven, Leuven, Belgium.
Group Science, Engineering and Technology, KU Leuven Kulak, Kortrijk, Belgium.
Email: otto.debals@esat.kuleuven.be

Lieven De Lathauwer
Department of Electrical Engineering
ESAT, KU Leuven, Leuven, Belgium.
Group Science, Engineering and Technology, KU Leuven Kulak, Kortrijk, Belgium.
Email: lieven.delathauwer@kuleuven-kulak.be

***Abstract*—A new method for the blind identification of large-scale finite impulse response (FIR) systems is presented. It exploits the fact that the system coefficients in large-scale problems often depend on much fewer parameters than the total number of entries in the coefficient vectors. We use low-rank models to compactly represent matricized versions of these compressible system coefficients. We show that blind system identification (BSI) then reduces to the computation of a structured tensor decomposition by using a deterministic tensorization technique called segmentation on the observed outputs. This careful exploitation of the low-rank structure enables the unique identification of both the system coefficients and the inputs. The approach does not require the input signals to be statistically independent.**

I. Introduction

In blind system identification (BSI) one wishes to identify the system coefficients of an unknown system that is driven by unknown inputs resorting only to measured output values [1]. In contrast with instantaneous blind source separation (BSS), the outputs are convolutive mixtures of the inputs [2]. In this paper we limit ourselves to the blind identification of finite impulse response (FIR) systems. BSI occurs in many applications within signal processing, image processing, and sensor array processing, see, e.g., [3], [4], [5].

BSS methods have been developed that make some statistical assumption on the sources and then use shifted second-order or higher-order statistics to tensorize the problem. This allows one to uniquely determine the mixture and the sources under some mild conditions. Similar methods have been developed for BSI, see [1] for an overview. In the BSS case several deterministic methods have recently been proposed [6], [7]. The authors have earlier developed a tensor-based method for large-scale BSS using segmentation as a deterministic tensorization technique [8], [9]. In this paper, we generalize the approach to large-scale convolutive BSI.

The key idea is known from compressed sensing [10]: when considering large-scale signals or big data, it is clear that there is often an excessive number of entries compared to the actual amount of information contained in the signal. In other words, there is some structure in the signal allowing one to model it much more compactly. Such signals are called compressible and they can be represented by parsimonious models. A way to do this is by using (higher-order) low-rank models [11], [12]. In this paper, we adopt this strategy to the system coefficients in BSI. Our approach consists of tensorization of the observed outputs using segmentation and computation of a structured tensor decomposition of the resulting tensor. By exploiting the hypothesized compressibility of the system coefficients in this way, we obtain the first big data variant of BSI, allowing us to uniquely retrieve the system coefficients and the inputs.

In the remainder of this section we introduce the notation and basic definitions. We present our method in Section II. In Section III and IV we discuss numerical experiments and possible applications, respectively. We conclude in Section V.

A. Notation and definitions

Vectors and matrices, denoted by bold lowercase (e.g., $\mathbf{a}$) and bold uppercase (e.g., $\mathbf{A}$) letters, respectively, can be generalized to tensors, denoted by calligraphic letters (e.g., $\mathcal{A}$). The $(i_1, i_2, \ldots, i_N)$th entry of an Nth-order tensor $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$ ($\mathbb{K}$ meaning $\mathbb{R}$ or $\mathbb{C}$) is denoted by $a_{i_1 i_2 \ldots i_N}$. The jth column of a matrix $\mathbf{A} \in \mathbb{K}^{I \times J}$ is indicated as $\mathbf{a}_j$. A superscript between parentheses denotes an element in a sequence: $\{\mathbf{A}^{(n)}\}_{n=1}^{N}$. The transpose is indicated by $\bullet^T$.

A mode-n vector of a tensor $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$ is a natural generalization of the rows and columns of a matrix. The former is defined by fixing every index except the nth, e.g., $\mathbf{a}_{i_1 \cdots i_{n-1} : i_{n+1} \cdots i_N}$. An Nth-order slice is obtained by fixing all but N indices. For example, the second-order slices of a third-order tensor $\mathcal{X} \in \mathbb{K}^{I \times J \times K}$ are denoted by $\mathbf{X}_i$, $\mathbf{X}_j$, and $\mathbf{X}_k$ and are called the horizontal, lateral, and frontal slices, respectively. $\mathbf{A}_{(n)}$ denotes the mode-n unfolding of $\mathcal{A}$ and has the mode-n vectors as its columns, following the ordering convention in [13]. The vectorization of $\mathcal{A}$ is a vector $\text{vec}(\mathcal{A})$ obtained by mapping each element $a_{i_1 i_2 \cdots i_N}$ onto $\text{vec}(\mathcal{A})_j$ with $j = 1 + \sum_{k=1}^{N} (i_k - 1) J_k$ and $J_k = \prod_{m=1}^{k-1} I_m$. The Kronecker and outer product are denoted by $\otimes$ and $\otimes$, respectively. They are related through a vectorization: $\text{vec}\,(\mathbf{a} \otimes \mathbf{b}) = \mathbf{b} \otimes \mathbf{a}$.

B. Tensor decompositions

An Nth-order tensor of rank one is defined as the outer product of N nonzero vectors. The rank of a tensor is defined

as the minimal number of rank-1 tensors that generate the tensor as their sum. The mode-n rank of a tensor equals the rank of its mode-n unfolding. The multilinear rank of an Nth-order tensor is defined as the N-tuple of these mode-n ranks.

Definition 1. A *polyadic decomposition* (PD) writes an Nth-order tensor $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$ as a sum of R rank-1 terms:

$$\mathcal{A} = \sum_{r=1}^{R} \mathbf{u}_r^{(1)} \otimes \mathbf{u}_r^{(2)} \otimes \cdots \otimes \mathbf{u}_r^{(N)}. \tag{1}$$

The columns of the factor matrices $\mathbf{U}^{(n)} \in \mathbb{K}^{I_n \times R}$ are equal to the factor vectors $\mathbf{u}_r^{(n)}$ for $1 \leq r \leq R$. The PD is called *canonical* (CPD) when R is equal to the rank of $\mathcal{A}$.

The CPD is used in several applications within signal processing, biomedical sciences, and data mining [13], [14]. An important advantage of the CPD for higher-order tensors is its uniqueness under rather mild conditions. We call the decomposition *essentially unique* if it is unique up to trivial permutation of the rank-1 terms and scaling and counterscaling of the factors in the same term, see, e.g., [15].

Definition 2. A *block term decomposition* (BTD) of a third-order tensor $\mathcal{X} \in \mathbb{K}^{I \times J \times K}$ *in multilinear rank-*$(P_r, P_r, 1)$ *terms* for $1 \leq r \leq R$ is a decomposition of the form:

$$\mathcal{X} = \sum_{r=1}^{R} (\mathbf{A}_r \mathbf{B}_r^{\mathrm{T}}) \otimes \mathbf{c}_r, \tag{2}$$

in which $\mathbf{A}_r \in \mathbb{K}^{I \times L_r}$ and $\mathbf{B}_r \in \mathbb{K}^{J \times L_r}$ have full column rank P_r and $\mathbf{c}_r$ is nonzero.

The block terms in (2) are more general than the simple rank-1 terms of the PD and allow one to capture more complex phenomena, see e.g., [6], [16], [17]. Other types of BTDs as well as uniqueness conditions can be found in [6], [18].

II. Segmentation-based blind system identification

We present a new tensor-based BSI method, allowing one to uniquely identify the system coefficients and inputs in (very) large-scale problems. We will explain that this is possible by tensorizing the observed outputs which allows us to exploit the hypothesized compressibility of the system coefficients. We define the BSI problem, motivate the working hypothesis, derive our method, and discuss selection of parameters, respectively, in Subsection II-A, II-B, II-C, and II-D.

A. Blind system identification

Consider a discrete linear time-invariant system with M outputs, R inputs, and memory L. The system coefficients of the filter from input r to output m are denoted by $h_{mr}[l]$ for $0 \leq l \leq L$. The mth output of FIR system is defined as:

$$x_m[k] = \sum_{r=1}^{R} \sum_{l=0}^{L} h_{mr}[l] s_r[k-l] + n_m[k] \tag{3}$$

for $1 \leq k \leq K$. Here $s_r[k]$ and $n_m[k]$ denote the rth input and additive noise on the mth output, respectively. In this paper we will use the matrix formulation of (3). First define the matrices $\mathbf{H}^{(l)} \in \mathbb{K}^{M \times R}$ and $\mathbf{S}^{(l)} \in \mathbb{K}^{R \times K}$ element-wise as $h_{mr}^{(l)} = h_{mr}[l]$ and $s_{rk}^{(l)} = s_r[k-l]$ for $0 \leq l \leq L$, respectively. Then the output data matrix $\mathbf{X} \in \mathbb{K}^{M \times K}$ can be written as:

$$\mathbf{X} = \sum_{l=0}^{L} \mathbf{H}^{(l)} \mathbf{S}^{(l)}. \tag{4}$$

The goal of BSI is to identify the system coefficients using *only* the output data. It is clear that (4) reduces to instantaneous BSS if $L = 0$. We ignore noise for notational simplicity in the derivation of our method. Its influence will be examined later by means of simulations.

B. Low-rank coefficient vectors

In this paper we assume that a matricized version of the (r,l)th coefficient vector $\mathbf{h}_r^{(l)}$ in (4), i.e., the rth column of $\mathbf{H}^{(l)}$, admits a low-rank model. We denote such vectors as *low-rank coefficient vectors* and we will show that this working hypothesis enables a unique identification of the system coefficients and inputs. This assumption is satisfied in many real-life large-scale applications because in large-scale situations the coefficients are often compressible, i.e., they can be described in terms of much fewer parameters than the actual number of coefficients [11]. The high compressibility of such low-rank models makes our method applicable in large-scale BSI problems. The authors adopted a similar strategy to solve large-scale instantaneous BSS in [8], [9]. We generalize the strategy here to the convolutive BSI case.

A vector with Vandermonde structure is an example of an exact rank-1 coefficient vector. For example, take $h_{mr}^{(l)} = az^m$ for $m = 0, 1, \ldots 5$. Reshaping this vector into a (2×3) matrix:

$$a \begin{pmatrix} 1 & z^2 & z^4 \\ z & z^3 & z^5 \end{pmatrix} = a \begin{pmatrix} 1 \\ z \end{pmatrix} \begin{pmatrix} 1 & z^2 & z^4 \end{pmatrix}, \tag{5}$$

clearly illustrates the rank-1 structure. It is known that sums and products of exponentials and trigonometric terms admit a low-rank model [6]. For example, a sine is a linear combination of two (complex conjugated) exponentials and, hence, admits a rank-2 model. It can be proven that matricized versions of functions that depend smoothly on m can be well approximated with a low-rank model [9]. This is illustrated in Figure 1 for a Gaussian, rational function, and sigmoid.

More formally, reshape coefficient vector $\mathbf{h}_r^{(l)}$ in (4) into a $(I \times J)$ matrix $\mathbf{H}_r^{(l)}$ such that $\mathrm{vec}(\mathbf{H}_r^{(l)}) = \mathbf{h}_r^{(l)}$ with $M = IJ$. Our working hypothesis now states that the matricized coefficient vectors should admit a low-rank representation, hence:

$$\mathbf{H}_r^{(l)} = \sum_{p=1}^{P_r^{(l)}} \mathbf{a}_{pr}^{(l)} \otimes \mathbf{b}_{pr}^{(l)} = \mathbf{A}_r^{(l)} \mathbf{B}_r^{(l),\mathrm{T}} \tag{6}$$

with $\mathbf{a}_{pr}^{(l)} \in \mathbb{K}^I$ and $\mathbf{b}_{pr}^{(l)} \in \mathbb{K}^J$. This is equivalent to assuming that $\mathbf{h}_r^{(l)}$ can be written as a sum of Kronecker products:

$$\mathbf{h}_r^{(l)} = \mathrm{vec}(\mathbf{H}_r^{(l)}) = \sum_{p=1}^{P_r^{(l)}} \mathbf{b}_{pr}^{(l)} \otimes \mathbf{a}_{pr}^{(l)}. \tag{7}$$

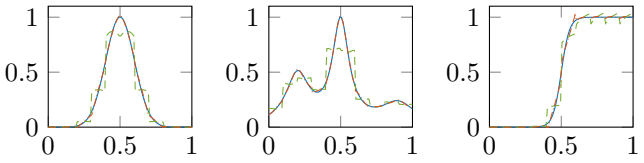


Fig. 1. Consider a Gaussian (left), a rational function (middle), and a sigmoid (right) evaluated in 100 equidistant samples in $[0, 1]$. We reshaped the original functions (——) into (10×10) matrices and subsequently approximated them by a low-rank matrix. This is done by truncating the singular value decomposition. The reconstructed functions are obtained by vectorizing the best rank-1 (– – – –) and rank-2 (–·–·–) approximation. It is clear that the functions can be better approximated by a rank-2 than a rank-1 model. Also note that the rank-2 model approximately coincides with the original function.

This is clearly interesting for large-scale BSI because of the possibly large compressions. We only need $P_r^{(l)}(I + J - 1)$ parameters instead of $M = IJ$ to model $\mathbf{h}_r^{(l)}$. For example, the number of parameters needed in the model is one order of magnitude lower than the total number of values when $I \approx J$.

C. Segmentation and decomposition

We show that the BSI problem in (4) can be reformulated as the computation of a structured tensor decomposition given low-rank system coefficients of the form in (7). First, each column of $\mathbf{X}$ is reshaped into a $(I \times J)$ matrix $\mathbf{X}_k$ with $M = IJ$, see Subsection II-D for the choice of I and J. Second, we stack the matricized columns into the frontal slices of a third-order tensor $\mathcal{X} \in \mathbb{K}^{I \times J \times K}$, i.e., we have that $\text{vec}(\mathbf{X}_k) = \mathbf{x}_k$. Since this operation is linear, the M reshaped observed outputs are linear combinations of the RL shifted sources $\mathbf{s}_r^{(l)}$ using the matricized system coefficients $\mathbf{H}_r^{(l)} \in \mathbb{K}^{I \times J}$, i.e., we have:

$$\mathcal{X} = \sum_{r=1}^{R} \sum_{l=0}^{L} \mathbf{H}_r^{(l)} \otimes \mathbf{s}_r^{(l)}.$$

This particular deterministic tensorization technique is denoted by *segmentation*, see [8], [9], [19]. Assume that the system coefficients admit a low-rank representation as in (6), i.e., $\mathbf{H}_r^{(l)} = \sum_{p=1}^{P_r^{(l)}} \mathbf{a}_r^{(l)} \otimes \mathbf{b}_r^{(l)}$, for $1 \leq r \leq R$ and $0 \leq l \leq L$:

$$\mathcal{X} = \sum_{r=1}^{R} \sum_{l=0}^{L} \left(\mathbf{A}_r^{(l)} \mathbf{B}_r^{(l),\mathrm{T}} \right) \otimes \mathbf{s}_r^{(l)}. \tag{8}$$

Equation (8) is a BTD in RL multilinear rank-$(P_r^{(l)}, P_r^{(l)}, 1)$ terms as in (2) with the additional superscript l due to the convolution. Hence, if the reshaped system coefficients admit a low-rank representation, BSI boils down to the computation of a tensor decomposition. We explicitly note that the compressibility of the system coefficients has enabled their blind identification. It is also possible to additionally impose statistical independence on the inputs when applicable, but this is out of the scope of this paper. Moreover, note that our method *also* works for inputs that are *not* statistical independent or i.i.d. Uniqueness properties of this particular BTD have been mentioned in Subsection I-B. Finally, the factor matrix of the third mode, i.e., $\mathbf{S} = \begin{bmatrix} \mathbf{S}^{(0)} & \cdots & \mathbf{S}^{(L)} \end{bmatrix} \in \mathbb{K}^{K \times R(L+1)}$, has a block Toeplitz structure, see [20], [21] for uniqueness properties of block Toeplitz constrained decompositions.

The proposed method blindly identifies both the system coefficients and the inputs by applying segmentation and then computing a structured decomposition of the resulting tensor, benefiting from the mild uniqueness properties of (structured) tensor decompositions. Consequently, we carefully exploit the compressibility of the system coefficients, enabling BSI in large-scale applications. Finally, our method is deterministic, hence, it also works well if only a limited number of samples are available. Note that if the problem is memoryless (i.e., $L = 0$), BSI boils down to instantaneous BSS, see [8], [9].

D. Parameter selection

We investigate a simple example in order to give some intuition on how to choose the segmentation parameters I and J and the number of rank-1 terms $P_r^{(l)}$ in our approach. Consider the rational function from Figure 1 (middle) but uniformly discretized in $M = 2^{14}$ samples in $[0, 1]$. First, let us reshape the resulting vector into an $(I \times J)$ matrix with $I = 2^q$ and $J = 2^{14-q}$ for $2 \leq q \leq 12$ such that $IJ = M$. Next, we compute the best rank-P approximation by truncating the singular value decomposition (SVD) for $P = \{1, 2, 3\}$. Define the normalized number of parameters as follows: $\hat{M} = P(I + J - 1)/M$. Figure 2 plots $\hat{M}$ as a function of the relative error ϵ of the rank-P model.

There is a clear trade-off between compression and accuracy. Hence, what is considered a "good" choice of parameters depends on the needs in a particular application [9]. Note that segmentation is not symmetric in the modes it creates: for a fixed rank, $I < J$ is a better choice than $I > J$. (Analogous, note that the column and row in (5) have a different Vandermonde generator.) In this example, the compression can be increased by taking $I \approx J$ and choosing P not too large. The accuracy on the other hand can be improved by changing I and J and/or increasing P. We indicate an example in Figure 2 with an arrow. First, we increase q from 3 to 5, obtaining a slightly more square matrix which improves the compression but reduces the accuracy (for the same rank value). Next, we increase the rank from 2 to 3, greatly improving the accuracy and only slightly reducing the compression. Overall we have both a better compression and accuracy.

In practice, one can start with a trial-and-error approach to find some reasonable segmentation parameters and overestimate P; the latter is not so critical anyway [9]. Subsequently, one can perform a similar analysis as here with ranks smaller than P and refine the choice of the parameters I, J, and P.

III. Numerical experiments

We illustrate our method with a simple proof-of-concept and an inspection of the influence on accuracy when explicitly imposing block Toeplitz structure on the decomposition in (8). We use Gaussian additive noise unless stated otherwise.

The system coefficients as well as the inputs can only be determined up to scaling and permutation, which are the standard indeterminacies in BSI. Hence, we first optimally scale and

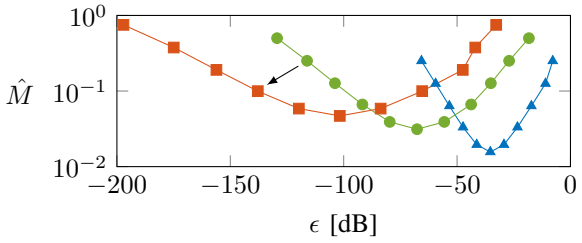


Fig. 2. Normalized number of parameters $\hat{M} = P(I+J-1)/M$ as a function of the relative error (dB) of the rank-P approximation of a matricized version of the rational function depicted in Figure 1 (middle) discretized in $M = 2^{14}$ equidistant samples in $[0, 1]$. First, the resulting vector is reshaped into an $(I \times J)$ matrix with $I = 2^q$ and $J = 2^{14-q}$ such that $IJ = M$. Subsequently, the best approximation is obtained by truncating the singular value decomposition for $P = 1$ (—▲—), 2 (—●—), and 3 (—■—). The parameter $2 \leq q \leq 12$ increases from left to right on each curve. The arrow indicates a beneficial change of parameters, improving both the accuracy and compression.

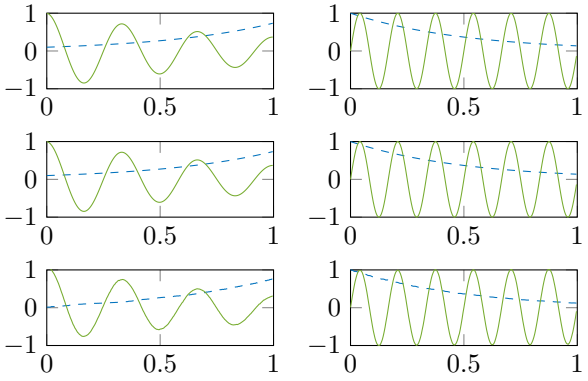


Fig. 3. Original (top) and reconstructed system coefficients in the noiseless (middle) and noisy (bottom) case (15 dB SNR) for the first (left) and second (right) input and both shifts ($l = 0$ (- - -) and $l = 1$ (——)). Note a perfect and excellent reconstruction for the noiseless and noisy case, respectively.

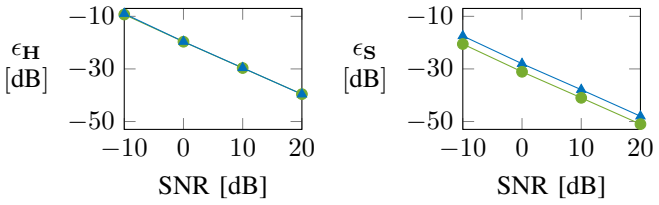


Fig. 4. Median across 100 experiments of the relative error (dB) on the coefficient vectors (left) and the inputs (right) as a function of SNR (dB). The relative errors are shown for the results extracted from the unstructured (—▲—) decomposition and the block Toeplitz constrained (—●—) decomposition.

permute them with respect to the true ones in order to compute the relative error. We then define the relative error $\epsilon_{\mathbf{A}}$ as the relative difference in Frobenius norm $||\mathbf{A} - \hat{\mathbf{A}}||_{\mathrm{F}}/||\mathbf{A}||_{\mathrm{F}}$ with $\hat{\mathbf{A}}$ an optimally scaled and permuted estimate for the matrix $\mathbf{A}$. The CPD and the BTD in multilinear rank-$(L_r, L_r, 1)$ terms are computed using `cpd_nls` and `lll_nls` from Tensorlab, respectively, see [18], [22], [23]. We use a generalized eigenvalue decomposition as initialization for the latter. The block Toeplitz constrained version can be easily implemented using the structured data fusion (SDF) framework in Tensorlab [24].

In a first experiment we consider an FIR system with $M = 2500$ outputs, $R = 2$ i.i.d. zero-mean unit-variance Gaussian random inputs of sample size $K = 100$, and system order $L = 2$. We use the following low-rank coefficient vectors: $h_1^{(0)}(\xi) = 0.1e^{2\xi}$, $h_1^{(1)}(\xi) = e^{-\xi}\cos(6\pi\xi)$, $h_2^{(0)}(\xi) = e^{-2\xi}$, and $h_2^{(0)} = \sin(12\pi\xi)$ evaluated in M equidistant samples in $[0, 1]$. We use a rank-1 ($P_1^{(0)} = P_2^{(0)} = 1$) and rank-2 ($P_1^{(1)} = P_2^{(1)} = 2$) approximation for the matricized exponential and sinusoidal coefficient vectors, respectively, with $I = J = 50$. Hence, we decompose the $(50 \times 50 \times 20)$ segmented version of the observed data matrix $\mathbf{X}$ in a sum of multilinear rank-$(P_r^{(l)}, P_r^{(l)}, 1)$ terms as in (8) using `lll_nls`. We do not impose block Toeplitz structure on the decomposition. Note that one only needs $P_r^{(l)}(I+J-1)$ values to approximate the (r, l)th coefficient vector, i.e., only 99 or 198 values instead of 2500 depending on the rank. This results in a compression of $1-P_r^{(l)}(I+J-1)/M = 96.04\%$ or 92.08%, respectively. See Figure 3 for an illustration of the original and reconstructed coefficient vectors in both the noiseless and noisy case.

We now compare the accuracy of the results when extracted from the unstructured versus the block Toeplitz constrained decomposition. Consider a FIR system with $M = 2500$ outputs and system order $L = 2$. We have $R = 2$ (deterministic) complex exponential inputs: $s_r[k] = e^{2\pi rik}$ with $K = 20$ equidistant samples in $[0, 1]$. The low-rank coefficient vectors are constructed as the vectorization of rank-1 ($P_r^{(l)} = 1\ \forall r, l$) matrices using (7) with zero-mean unit-variance Gaussian random factor vectors of size $I = J = 50$. Consequently we also use a second-order rank-1 approximation for all coefficient vectors with $I = J = 50$. Hence, the unstructured decomposition is simply a CPD as in (1) which can be computed with `cpd_nls`. We use the result of the latter as the initialization of the block Toeplitz constrained decomposition algorithm which is often a good initialization strategy. We report the relative error on the coefficient vectors $\epsilon_{\mathbf{H}}$ and the inputs $\epsilon_{\mathbf{S}}$ in Figure 4. Note that the results are very accurate compared to the SNR. It is clear that $\epsilon_{\mathbf{S}}$ improves by imposing the block Toeplitz structure. Even for negative SNR, a good accuracy is obtained, thanks to the exploitation of the low-rank structure. On the other hand, the relative error on the coefficients $\epsilon_{\mathbf{H}}$ does not improve significantly. Finally, note that $\epsilon_{\mathbf{S}}$ is lower than $\epsilon_{\mathbf{H}}$ because this is the shorter factor in the decomposition, i.e., $K < I, J$.

IV. Applications

There is a trend in biomedical applications to higher density sensor grids. Examples include surface electromyogram (sEMG) and wireless body area networks (WBANs) based on electroencephalography (EEG) and electrocorticography (ECoG) [3], [25], [26]. BSS and BSI are typical problems in these applications [9]. For example, one of the first models proposed to separate the action potentials of the muscle's motor units in high-density sEMG (HD-sEMG) was BSS. The BSI model of (4) is, however, closer to the real phenomenon.

In array processing and telecommunications, FIR models can be used to model uniform linear arrays (ULAs) and rectangular arrays (URAs) with far-field sources that emit narrowband signals [27]. In this case the coefficient vectors have a Vandermonde structure and thus have rank-1 structure, as shown in (5). In the near-field and/or multipath case the matricized system coefficients typically have a low-rank structure [9]. Here as well we see an increase in the number of antennas. This is also known as massive MIMO [28].

V. Conclusion

We presented the first BSI method that can be used for (very) large-scale FIR systems. It exploits the hypothesized compressibility of system coefficients in large-scale problems by using segmentation on the observed outputs. Computation of a structured decomposition of the resulting tensor allowed us to uniquely identify both the system coefficients and the inputs. Statistical independence of the inputs is not needed. Actually only a few samples are needed because the method is deterministic. We investigated parameter selection and concluded the paper with numerical experiments and applications.

In this paper we focused on second-order segmentation, i.e., matricization of the coefficient vectors. This can be extended to higher-order segmentation as in [8], [9] for instantaneous BSS. Also, a more extensive inspection of the block Toeplitz structure and its influence on the uniqueness properties and accuracy is needed. In a follow-up paper we will present more simulations and validate the method on real-life applications.

Acknowledgment

This research is funded by (1) a Ph.D. grant of the Agency for Innovation by Science and Technology (IWT), (2) Research Council KUL: C1 project C16/15/059-nD, CoE PFV/10/002 (OPTEC). (3) FWO: project: G.0830.14N, G.0881.14N, (4) the Belgian Federal Science Policy Office: IUAP P7/19 (DYSCO, Dynamical systems, control and optimization, 2012-2017), (5) EU: The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC Advanced Grant: BIOTENSORS (no. 339804). This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information.

References

[1] K. Abed-Meraim, W. Qiu, and Y. Hua, "Blind system identification," *Proceedings of the IEEE*, vol. 85, no. 8, pp. 1310–1322, Aug. 1997.

[2] P. Comon and C. Jutten, *Handbook of blind source separation: Independent component analysis and applications*. Academic press, 2009.

[3] A. Holobar and D. Farina, "Blind source identification from the multichannel surface electromyogram," *Physiological Measurement*, vol. 35, no. 7, p. R143.

[4] D. Kundur and D. Hatzinakos, "Blind image deconvolution," *IEEE Signal Processing Magazine*, vol. 13, no. 3, pp. 43–64, May 1996.

[5] A.-J. van der Veen, "Algebraic methods for deterministic blind beamforming," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 1987–2008, Oct. 1998.

[6] L. De Lathauwer, "Blind separation of exponential polynomials and the decomposition of a tensor in rank-$(L_r, L_r, 1)$ terms," *SIAM Journal on Matrix Analysis and Applications*, vol. 32, no. 4, pp. 1451–1474, 2011.

[7] O. Debals, M. Van Barel, and L. De Lathauwer, "Löwner-based blind signal separation of rational functions with applications," *Technical Report 15–44, ESAT-STADIUS, KU Leuven, Leuven, Belgium*, 2015, (accepted for publication in IEEE TSP).

[8] M. Boussé, O. Debals, and L. De Lathauwer, "A novel deterministic method for large-scale blind source separation," in *Proceedings of the 23rd European Signal Processing Conference (EUSIPCO 2015, Nice, France)*, Aug. 2015, pp. 1935–1939.

[9] ——, "A tensor-based method for large-scale blind source separation using segmentation," *Technical Report 15-59, ESAT-STADIUS, KU Leuven, Leuven, Belgium*, 2015.

[10] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

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

[12] N. Vervliet, O. Debals, L. Sorber, and L. De Lathauwer, "Breaking the curse of dimensionality using decompositions of incomplete tensors: Tensor-based scientific computing in big data analysis," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 71–79, Sept. 2014.

[13] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, Aug. 2009.

[14] A. Cichocki, D. P. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. F. Caiafa, and A. H. Phan, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 145–163, Mar. 2015.

[15] I. Domanov and L. De Lathauwer, "Canonical polyadic decomposition of third-order tensors: Reduction to generalized eigenvalue decomposition," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, no. 2, pp. 636–660, Apr.-May 2014.

[16] L. De Lathauwer, "Block component analysis, a new concept for blind source separation," in *Latent Variable Analysis and Signal Separation*, ser. Lecture Notes in Computer Science. Springer Berlin/Heidelberg, 2012, vol. 7191, pp. 1–8.

[17] L. De Lathauwer and A. De Baynast, "Blind deconvolution of DS-CDMA signals by means of decomposition in rank-$(1, L, L)$ terms," *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1562–1571, Apr. 2008.

[18] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms — Part II: Definitions and uniqueness," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1033–1066, Sept. 2008.

[19] O. Debals and L. De Lathauwer, "Stochastic and deterministic tensorization for blind signal separation," in *Latent Variable Analysis and Signal Separation*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2015, vol. 9237, pp. 3–13.

[20] M. Sørensen and L. De Lathauwer, "Convolutive low-rank factorizations via coupled low-rank and Toeplitz structured matrix/tensor decompositions," *Technical Report 16–37, ESAT-STADIUS, KU Leuven, Leuven, Belgium*.

[21] M. Sørensen, F. Van Eeghem, and L. De Lathauwer, "Tensor decompositions with block-Hankel factors with application in blind system identification," *Technical Report 16–39, ESAT-STADIUS, KU Leuven, Leuven, Belgium*.

[22] L. Sorber, M. Van Barel, and L. De Lathauwer, "Optimization-based algorithms for tensor decompositions: canonical polyadic decomposition, decomposition in rank-$(L_r, L_r, 1)$ terms and a new generalization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 695–720, Apr. 2013.

[23] ——, "Tensorlab v2.0," Jan. 2014, available online at http://www.tensorlab.net/.

[24] ——, "Stuctured data fusion," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 4, pp. 586–600, June 2015.

[25] A. Bertrand, "Distributed signal processing for wireless EEG sensor networks," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2015, (accepted for publication).

[26] B. Rubehn, C. Bosman, R. Oostenveld, P. Fries, and T. Stiegliz, "A MEMS-based flexible multichannel ECoG-electrode array," *Journal of Neural Engineering*, vol. 6, no. 3, pp. 1–10, 2009.

[27] H. Krim and M. Viberg, "Two decades of array signal processing: The parametric approach," *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, 1996.

[28] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, Feb. 2014.