# SECOND-ORDER TENSOR-BASED CONVOLUTIVE ICA: DECONVOLUTION VERSUS TENSORIZATION

*Frederik Van Eeghem*[⋆†‡] *Lieven De Lathauwer*[⋆†‡]

[⋆]Group Science, Engineering and Technology, KU Leuven Kulak, E. Sabbelaan 53, B-8500 Kortrijk, Belgium. [†]Department of Electrical Engineering (ESAT), KU Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium.

## ABSTRACT

Independent component analysis (ICA) research has been driven by various applications in biomedical signal separation, telecommunications, speech analysis, and more. One particular class of algorithms for instantaneous ICA uses tensors, which have useful properties. In an attempt to port these properties to convolutive methods, we zoom in on an existing method that uses second-order statistics. By pointing out links in the literature, we show that this method is in fact a typical tensor-based method, even though this was not recognized by the authors at the time.

The existing method mentioned above can be interpreted as a tensorization step followed by a deconvolution step. However, as sometimes done in literature, one may consider using the opposite approach; starting with a deconvolution step and then tensorizing the remaining instantaneous mixture. Because subspace-based deconvolution can be slow, we propose a fast variant which uses only partial information. We then use this variant to compare the approach starting with tensorization and the one starting with deconvolution.

***Index Terms*—** tensor, convolutive independent component analysis, tensorization, deconvolution, second-order statistics

## 1. INTRODUCTION

Various signals in biomedical data processing and array processing can be interpreted as an instantaneous mixture of statistically independent components. To retrieve these components, one can turn to independent component analysis (ICA) [1]. However, in several applications, the underlying independent components are not mixed instantaneously because of path length differences and reflections. Among others, this is the case for speech signals, telecommunications, and seismic signals [2]. For these applications, a convolutive mixture model is more appropriate.

Tensors are higher-dimensional generalizations of vectors and matrices. A key strength of tensors is the uniqueness of their decompositions under mild conditions [3, 4]. Moreover, several decompositions can be computed algebraically [5]. These properties make tensors useful tools in chemometrics, psychometrics, and signal separation [6–8].

For both instantaneous as convolutive ICA, a variety of algorithms have been developed [1, 9]. Of particular interest are tensor-based methods, which are well-established for instantaneous ICA [1, 10, 11]. In an attempt to port the favorable tensor properties to convolutive ICA, several tensor-based algorithms have been presented for convolutive mixtures [12, 13]. In these methods, the convolution gives rise to additional structure, which is often not exploited. Since taking this structure into account may lead to faster or more accurate methods, recent advances focus on exploiting the structure [14]. In this paper, we focus on convolutive mixtures of temporally coherent and mutually independent components. A separation algorithm for this model using second-order statistics was coined in [13], and can actually be seen as the convolutive extension of the popular SOBI algorithm for instantaneous ICA [10]. Though not recognized by the authors, the method in [13] is in fact a tensor-based approach. As a first contribution, we will clarify the link between the method from [13] and tensor decompositions.

Most convolutive ICA methods involve a deconvolution step and a separation step. The method from [13] first separates the mixture and subsequently deconvolves the resulting filtered versions of the original sources. The opposite approach, in which the data are deconvolved first, is quite popular because it allows one to use any instantaneous ICA algorithm after the deconvolution. Deconvolving is typically done using second-order statistics [15, 16] or subspace-based techniques [17–19]. Subspace-based techniques avoid computing any statistics, but they involve large-size matrices and may be slow [20]. For single-input multiple-output (SIMO) systems, this has been countered by using only part of the data [21]. As a second contribution, we extend this idea to MIMO systems to obtain faster deconvolution. We give a lower bound for the number of needed observations and provide a relative measure to determine reasonable values for the number of observations in practice. Numerical experiments support our analysis. Due to space limitations, further comparisons are done in a follow-up paper.

*Notations:* scalars are represented by normal lowercase letters (e.g., $a$), vectors by bold lowercase letters (e.g. $\mathbf{a}$), matrices by bold uppercase letters (e.g. $\mathbf{A}$) and tensors by calligraphic letters (e.g. $\mathcal{A}$). The Kronecker product is denoted by $\otimes$. The transpose and conjugate transpose are represented by $\cdot^{\mathrm{T}}$ and $\cdot^{\mathrm{H}}$, respectively. Commas and semicolons are used for horizontal and vertical concatenation, respectively. Bars are used to represent dropped block-rows. For instance, consider $\mathbf{A} = \left[\mathbf{A}^{(1)}; \cdots; \mathbf{A}^{(N)}\right]$, then $\overline{\mathbf{A}} = \left[\mathbf{A}^{(2)}; \cdots; \mathbf{A}^{(N)}\right]$ and $\underline{\mathbf{A}} = \left[\mathbf{A}^{(1)}; \cdots; \mathbf{A}^{(N-1)}\right]$. The mathematical expectation is denoted by $\mathbb{E}\{\cdot\}$.

Frederik Van Eeghem is supported by an Aspirant Grant from the Research Foundation – Flanders (FWO). This research is funded by (1) Research Council KU Leuven: C1 project c16/15/059-nD and CoE PFV/10/002 (OPTEC), (2) F.W.O.: project G.0830.14N and G.0881.14N, (3) the Belgian Federal Science Policy Office: IUAP P7/19 (DYSCO II, Dynamical systems, control and optimization, 2012-2017), (4) EU: The research leading to these results has received funding from the European Research Council under the European Unions 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.

## 2. PROBLEM STATEMENT

A convolutive mixture with $M$ outputs $x_m(t)$ and $R$ inputs $s_r(t)$ can be written as

$$x_m(t) = \sum_{r=1}^{R} \sum_{l=0}^{L} h_{mr}(l) s_r(t-l) \qquad \text{for } m \in \{1, \ldots, M\}. \quad (1)$$

In this equation, $h_{mr}(l)$ with $l \in \{0, \ldots, L\}$ represents the filter between the $r$th input and $m$th output. The maximum filter delay is represented by $L$.

If the system is strictly overdetermined, i.e., if there are strictly more outputs than inputs, equation (1) can be written as an overdetermined matrix equation

$$\mathbf{X} = \mathbf{H}\mathbf{S}, \quad (2)$$

in which $\mathbf{X} \in \mathbb{C}^{ML' \times N}$ represents the outputs and $\mathbf{S} \in \mathbb{C}^{RL_{\text{tot}} \times N}$ contains (lagged versions of) the inputs. The value $L'$ represents the number of lagged outputs taken into account and is chosen such that the mixing matrix $\mathbf{H} \in \mathbb{C}^{ML' \times RL_{\text{tot}}}$ is tall, i.e., such that $ML' \geq RL_{\text{tot}}$, with $L_{\text{tot}} = L + L'$. Note that including lagged versions of the outputs to obtain a tall matrix $\mathbf{H}$, sometimes called temporal smoothing, is only possible because the system is strictly overdetermined.

The columns $\mathbf{x}(n)$ and $\mathbf{s}(n)$ of the matrices $\mathbf{X}$ and $\mathbf{S}$ take the form

$$\begin{aligned} \mathbf{x}(n) = &[x_1(n), \ldots, x_M(n), x_1(n-1), \ldots, \\ &x_M(n-1), \ldots, x_M(n-L'+1)]^{\mathrm{T}}, \\ \mathbf{s}(n) = &[s_1(n), s_1(n-1), \ldots, s_1(n-L_{\text{tot}}+1), \ldots, \\ &s_R(n), \ldots, s_R(n-L_{\text{tot}}+1)]^{\mathrm{T}}. \end{aligned} \quad (3)$$

Equation (3) shows that if the rows of $\mathbf{S}$ are properly permuted, it has block-Toeplitz structure for $L > 0$. Denote the row-permuted version of $\mathbf{S}$ having block-Toeplitz structure by $\widetilde{\mathbf{S}} = \left[\widetilde{\mathbf{S}}^{(1)}; \ldots; \widetilde{\mathbf{S}}^{(L_{\text{tot}})}\right] \in \mathbb{C}^{RL_{\text{tot}} \times N}$, in which the $\widetilde{\mathbf{S}}^{(l)} \in \mathbb{C}^{R \times N}$ are shifted versions of each other.

The mixing matrix $\mathbf{H} \in \mathbb{C}^{ML' \times RL_{\text{tot}}}$ from (2) is given by

$$\mathbf{H} = [\mathbf{H}_1 \quad \mathbf{H}_2 \quad \cdots \quad \mathbf{H}_R],$$

in which

$$\mathbf{h}_q(l) = \begin{bmatrix} h_{1q}(l) & h_{2q}(l) & \cdots & h_{Mq}(l) \end{bmatrix}^{\mathrm{T}} \in \mathbb{C}^M,$$

and

$$\mathbf{H}_q = \begin{bmatrix} \mathbf{h}_q(0) & \cdots & \mathbf{h}_q(L) & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & \cdots & \mathbf{h}_q(0) & \cdots & \mathbf{h}_q(L) \end{bmatrix} \in \mathbb{C}^{ML' \times L_{\text{tot}}}.$$

Finally, throughout the text we assume that (1) the system is strictly overdetermined, i.e., $M > R$, (2) $\mathbf{H}$ has full column rank, (3) the inputs have zero mean and are temporally coherent but mutually independent and (4) $\mathbf{S}^{\mathrm{T}}$ has full column rank. Using these assumptions and the structural information described above, the goal is to retrieve the original inputs $s_r(t)$ for $r \in \{1, \ldots, R\}$ from the observed mixtures.

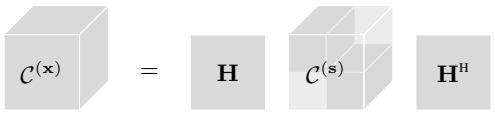


**Fig. 1**. The unmixing step in convolutive ICA can be written as a block term decomposition in multilinear rank-$(L_{\text{tot}}, L_{\text{tot}}, \cdot)$ terms.

## 3. DECONVOLUTION AFTER TENSORIZATION

In [13, 22], a convolutive generalization of the popular SOBI-algorithm for instantaneous mixtures [10] is presented, as mentioned in the introduction. This convolutive method starts by computing lagged second-order statistics of $\mathbf{x}(n)$, which can be stacked into a third-order tensor $\mathcal{C}^{(\mathbf{x})}$. The $k$th frontal slice of this tensor is given by

$$\begin{aligned} \mathbf{C}_k^{(\mathbf{x})} &= \mathbb{E}\left\{\mathbf{x}(n)\mathbf{x}^{\mathrm{H}}(n+\tau_k)\right\} \in \mathbb{C}^{ML' \times ML'} \\ &= \mathbf{H}\,\mathbb{E}\left\{\mathbf{s}(n)\mathbf{s}^{\mathrm{H}}(n+\tau_k)\right\}\mathbf{H}^{\mathrm{H}}. \end{aligned}$$

Because the inputs are temporally coherent and mutually independent, it follows from (3) that the (lagged) covariance matrix $\mathbb{E}\left\{\mathbf{s}(t)\mathbf{s}^{\mathrm{H}}(t+\tau_k)\right\}$ will be block-diagonal. Consequently, the frontal slices of $\mathcal{C}^{(\mathbf{x})}$ can be jointly block-diagonalized [13]. From this joint block-diagonalization, $\mathbf{H}$ can be found up to two indeterminacies: (1) its block columns $\mathbf{H}_i$ can be arbitrarily permuted and (2) each block column $\mathbf{H}_i$ can be multiplied with a nonsingular matrix $\mathbf{E}_i \in \mathbb{C}^{L_{\text{tot}} \times L_{\text{tot}}}$. Though the authors did not realize it at the time, this simultaneous block-diagonalization is in fact a block-term decomposition in multilinear rank-$(L_{\text{tot}}, L_{\text{tot}}, \cdot)$ terms [4], as illustrated in Figure 1. Algorithms for this decomposition can be found in [23–25]. Because of the indeterminacies of joint block-diagonalization, computing the decomposition only allows us to retrieve the inputs up to a FIR filter [13]. One can deal with these filters using SIMO deconvolution techniques [21], which eventually return the inputs up to scaling and permutation.

In instantaneous ICA, all tensor-based methods compute statistics which are stored in a tensor. By subsequently decomposing this tensor, the mixing system and sources are extracted [10, 11]. Conceptually, the approach for convolutive systems above is exactly the same, which implies that this method nicely fits in a tensor-based framework for convolutive ICA.

## 4. DECONVOLUTION BEFORE TENSORIZATION

One of the drawbacks of the method from the previous section lies in the computation time. By taking lagged outputs into account, more statistics have to be estimated. Moreover, joint block-diagonalization may take quite some time. By first deconvolving the outputs, we reduce the convolutive mixture to an instantaneous one, which can be solved more quickly. In this article, the main focus lies on the deconvolution step, which has to be both accurate and computationally inexpensive.

Subspace methods exploiting block-Toeplitz structure are well-established in signal and array processing [17–19, 26]. We briefly recall a technique from [26]. Compute the dominant row space of $\mathbf{X}$, which will be equal to the row space of $\mathbf{S}$ since $\mathbf{H}$ is assumed to have full column rank. Store the $RL_{\text{tot}}$ basis vectors for the dominant subspace of $\mathbf{X}$ in the matrix $\mathbf{U} \in \mathbb{C}^{RL_{\text{tot}} \times N}$. We now have

$$\mathbf{U}^{\mathrm{T}} = \widetilde{\mathbf{S}}^{\mathrm{T}}\mathbf{M} = \left[\left(\widetilde{\mathbf{S}}^{(1)}\right)^{\mathrm{T}}, \ldots, \left(\widetilde{\mathbf{S}}^{(L_{\text{tot}})}\right)^{\mathrm{T}}\right]\mathbf{M}, \quad (4)$$

in which $\mathbf{M} \in C^{RL_{\text{tot}} \times RL_{\text{tot}}}$. Since $\widetilde{\mathbf{S}}$ has block-Toeplitz structure, equation (4) represents a block-Toeplitz factorization. Methods to solve this type of factorization can be found in [21, 26, 27]. We will use the result from [26], in which Lemma II.4 states that the decomposition in (4) is essentially unique if $\mathbf{M}$ and $\mathbf{T}$ have full column rank, in which $\mathbf{T} \in \mathbb{C}^{(N-1)\times R(L_{\text{tot}}+1)}$ is defined by

$$\mathbf{T} = \left[\overline{\left(\widetilde{\mathbf{S}}^{(1)}\right)^{\mathsf{T}}}, \ldots, \overline{\left(\widetilde{\mathbf{S}}^{(L_{\text{tot}})}\right)^{\mathsf{T}}}, \underline{\left(\widetilde{\mathbf{S}}^{(L_{\text{tot}})}\right)^{\mathsf{T}}}\right].$$

In this problem, essential uniqueness means that the mixture is deconvolved up to an instantaneous mixture. Mathematically, this implies that if the conditions hold, we obtain an input matrix estimate $\mathbf{K} \in \mathbb{C}^{RL_{\text{tot}} \times N}$ that has block-Toeplitz structure and that is related to the real inputs as follows:

$$\mathbf{K} = (\mathbf{I}_{L_{\text{tot}} \times L_{\text{tot}}} \otimes \mathbf{A})\, \widetilde{\mathbf{S}}, \tag{5}$$

in which $\mathbf{A} \in \mathbb{C}^{R \times R}$ is a nonsingular matrix. Since this deconvolution step is deterministic, only few data samples are needed in principle. Reducing the number of used data samples leads to a faster deconvolution. However, we do need the full deconvolved signals to properly estimate the statistics for the resulting instantaneous ICA problem. This problem is solved by using only a part of the samples $N_{\text{part}}$ in the deconvolution step. The method described above applied to the partial subspace $\mathbf{U}_{\text{part}} \in \mathbb{C}^{RL_{\text{tot}} \times N_{\text{part}}}$ leads to an input matrix estimate $\mathbf{K}_{\text{part}} \in \mathbb{C}^{RL_{\text{tot}} \times N_{\text{part}}}$. Since both matrices share the same row space, they are related through

$$\mathbf{U}_{\text{part}}{}^{\mathsf{T}}\mathbf{G} = \mathbf{K}_{\text{part}}{}^{\mathsf{T}}, \tag{6}$$

with $\mathbf{G} \in \mathbb{C}^{RL_{\text{tot}} \times RL_{\text{tot}}}$. The transformation matrix $\mathbf{G}$ can be computed from this equation and subsequently applied to the full subspace $\mathbf{U}$ to obtain the full-length deconvolved signals $\mathbf{K} = \mathbf{G}^{\mathsf{T}}\mathbf{U}$.

**Minimum number of samples** To see how many samples must be retained in theory, we derive a lower bound on the number of observations $N_{\text{part}}$ such that such that (4) is essentially unique. For generic inputs, $\mathbf{T}$ will have full column rank if

$$N_{\text{part}} \geq R(L_{\text{tot}} + 1) + 1. \tag{7}$$

It then follows that the input matrix $\widetilde{\mathbf{S}}$ has full row rank. Since $\mathbf{H}$ is assumed to have full column rank, it follows that $\mathbf{M}$ also has full column rank and subsequently that the decomposition in (4) is essentially unique. Also note that the linear system (6) is overdetermined if this condition holds.

**Determining the number of samples in practice** The number of samples $N_{\text{part}}$ is a trade-off between computational speed and accuracy of the deconvolution. Estimating $R^2L_{\text{tot}}^2$ parameters from the $N_{\text{part}}RL_{\text{tot}}$ equations in (6) can be done in least-squares sense. The standard deviation of the least-squares estimator is proportional to

$$\sigma_{\text{LS}} \propto \frac{1}{\sqrt{N_{\text{part}}RL_{\text{tot}} - R^2L_{\text{tot}}^2}}.$$

To get an idea of the (relative) accuracy, one can use this equation to assess the influence of reducing the number of observations $N_{\text{part}}$. For instance, if $R = 2$ and $L_{\text{tot}} = 4$, then reducing the number of observations from 1000 to 100 increases the standard deviation on the estimator with a factor 3.28, which may still be reasonable depending on the application.

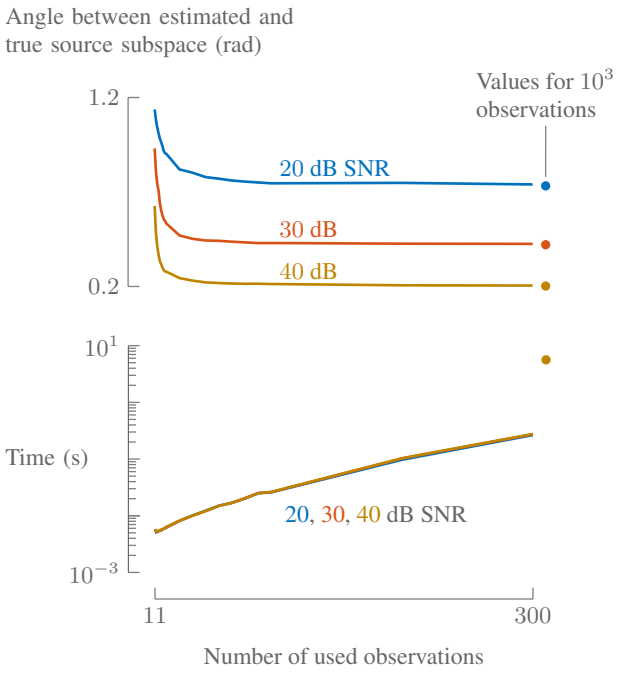


**Fig. 2**. Using fewer observations enables faster deconvolution without sacrificing much accuracy for reasonable sample sizes. Here, the theoretical minimum number of observations needed is 11.

**Resolving the remaining mixture** Once $\mathbf{K}$ is estimated, one can extract the first block-row $\mathbf{K}^{(1)}$, for which it follows from (5) that

$$\mathbf{K}^{(1)} = \mathbf{A}\widetilde{\mathbf{S}}^{(1)} \in \mathbb{C}^{R \times N}.$$

Since $\widetilde{\mathbf{S}}^{(1)}$ contains all sources without lags, $\mathbf{K}^{(1)}$ is an instantaneous mixture of independent sources. Any existing technique for instantaneous ICA can be used to separate this remaining mixture up to scaling and permutation of the original inputs [1, 9].

## 5. NUMERICAL EXPERIMENTS

### 5.1. Synthetic data

In both experiments below, the synthetic inputs are generated by passing a complex signal with components randomly sampled from a uniform distribution over $[0, 1]$ through a convolutive filter, of which the 20 coefficients have been randomly sampled from a standard normal distribution. This procedure yields mutually independent but temporally coherent inputs. The actual system coefficients making up the convolutive mixture are randomly sampled from a standard normal distribution. All additive noise is Gaussian.

#### *5.1.1. Effect of partial information for deconvolution*

To illustrate the effect of partial information, consider a convolutive mixture with 4 outputs, 2 inputs and a maximum filter delay of 2. The available output signals have $10^3$ samples. Following the approach outlined in Section 4, we estimate the input subspace using only part of the available data. The mean result over 100 experiments is shown in Figure 2. The theoretical minimum number of samples can be computed using equation (7) and is equal to 11 in this example. In the neighborhood of this theoretical minimum, the

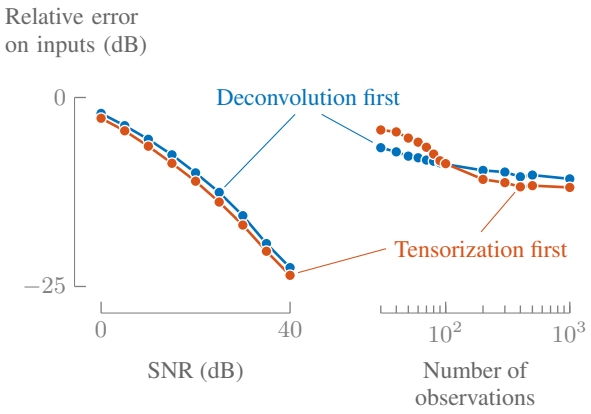


**Fig. 3**. Using deconvolution or tensorization as the first step in the algorithm yields comparable results. The small difference between both approaches arises from the number of available observations.

accuracy is low due to the big influence of noise. For a higher number of observations, the figure shows that using partial information enables a much faster deconvolution, with only little loss in accuracy when compared to the values for $10^3$ observations. Note that none of the subspace estimates is exact, this can be explained by noting that we are trying to estimate the long space of a matrix, which cannot be estimated consistently in noisy conditions [28].

#### 5.1.2. *Comparison of both strategies*

To see whether it is better to deconvolve first and tensorize afterwards or the other way around, we compare both approaches. Consider a system with 4 outputs, 2 inputs, a maximum filter delay of $L = 1$ and signals of length $10^4$. For the fast deconvolution presented in Section 4, 200 observations were used. After deconvolution, SOBI is used with the same time lags as its convolutive extension in this experiment (which are the first 20 nonzero lags). Figure 3 shows the mean performance over 1000 experiments of both approaches for various signal-to-noise ratio (SNR) values and for a varying number of available observations. The performance is expressed as the relative error on the inputs, after optimal permutation and scaling of the estimates.

The left part of the figure shows that the approach from [13], which starts by tensorizing, performs slightly better. However, this comes at the costs of a slower method. On average, the method takes 0.121 s on a standard laptop[1], whereas deconvolution before tensorization takes 0.061 s, both for signals of length $10^4$. The difference in accuracy can be explained as follows: since the available signals are long, the lagged statistics can be estimated accurately. Therefore, tensorizing the data first has a denoising effect without much loss of accuracy. If this is the case, we expect the accuracy to drop when fewer data samples are available because the estimation error of the statistics will be larger. This is indeed the case, as illustrated in the right part of Figure 3, which has been simulated with 20 dB SNR. The approach starting with deconvolution is less influenced by the number of data samples because the deconvolution step is deterministic. The crossover point lies around 100 samples. So when few data points are available, and statistics cannot be estimated accurately, one should deconvolve first.

[1]Using Matlab on a laptop containing an Intel Core i7-4810MQ CPU.

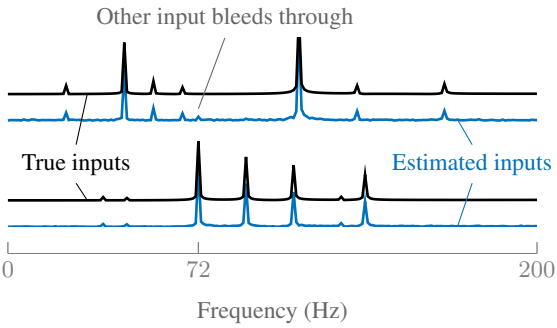


**Fig. 4**. Apart from a slight influence at 72 Hz, the fast deconvolution approach followed by SOBI cleanly separates a convolutive mixture of vibrational signals stemming from rotational machines.

### 5.2. Application to rotating machines

Vibration analysis is often used in rotating machines to find faults such as bearing defects without stopping production [29]. From the frequencies of the machine vibrations, one can pinpoint which component is faulty. However, if several machines are working side by side, their vibrations may travel through the foundation and disturb measurements. This is where blind source separation comes in [30]. If the machines are sufficiently different, their vibrations will be statistically independent. This motivates the use of convolutive ICA techniques. To illustrate our method in this context, we simulate rotating machine vibrations using a model for gearbox vibrations used in [30][2]. The simulated vibrational signals are mixed convolutively with filters of maximum length $L = 3$ to obtain 5 outputs, to which Gaussian noise is added such that the SNR is 30 dB. From these outputs, the inputs are estimated using the method that first deconvolves and then tensorizes, using 200 observations for the deconvolution and SOBI with 20 nonzero time lags as instantaneous ICA method. The results were computed in 0.16 s on a standard laptop. The obtained relative error on the original vibration signals is $-18.8$ dB. Figure 4 shows that at 72 Hz, a small error is noticeable in the estimated input where the effect of the other input slightly bleeds through. Apart from this, the frequency peaks of both signals are cleanly separated.

## 6. CONCLUSION

First, we linked an existing method for convolutive ICA with tensor-based methods. Next we contrasted this method with its opposite approach that starts by deconvolving the data before tensorizing. To do the deconvolution efficiently, a new subspace-based method was presented which uses only part of the available information. Through numerical experiments, this fast deconvolution was shown to be almost as accurate as methods using all information, while being much faster. Its effectiveness was illustrated in a simulation of rotational machine vibrations. Finally, we compared the speed and accuracy of the approach starting with tensorization and its opposite approach.

[2]For completeness, we give all parameter values. First source: $a = 3$, $\mathbf{A} = [-0.2792, -1.7251, -0.2326]$, $f_m = 55$, $\boldsymbol{\phi}_i = [0.0027, 0.1546, 0.0808]$, $b = 2$, $\mathbf{B} = [1.8883, -0.8930]$, $f_p = 77$, $\boldsymbol{\phi}_j = [-0.3361, 0.288]$. Second source: $a = 3$, $\mathbf{A} = [-0.1479, -2.7398, -1.6482]$, $f_m = 45$, $\boldsymbol{\phi}_i = [0.2198, 0.1782, -0.0626]$, $b = 2$, $\mathbf{B} = [-1.6716, -1.0965]$, $f_p = 81$, $\boldsymbol{\phi}_j = [0.0842, -0.2716]$.

## 7. REFERENCES


[1] P. Comon and C. Jutten, Eds., *Handbook of blind source separation, independent component analysis and applications*, Academic Press, 2010.

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

[3] I. Domanov and L. De Lathauwer, "On the uniqueness of the canonical polyadic decomposition of third-order tensors — Part II: Overall uniqueness," *SIAM J. Matrix Anal. Appl.*, vol. 34, no. 3, pp. 876–903, 2013.

[4] L. De Lathauwer, "Decomposition of a Higher-Order Tensor in Block Terms –Part II: Definitions and Uniqueness," *SIAM J. Matrix Anal. Appl.*, vol. 30, pp. 1033–1066, 2008.

[5] I. Domanov and L. De Lathauwer, "Canonical polyadic decomposition of third-order tensors: reduction to generalized eigenvalue decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 35, no. 2, pp. 636–660, 2014.

[6] A. Cichocki, C. Mandic, A. H. Phan, C. Caifa, G. Zhou, Q. Zhao, and L. De Lathauwer, "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.

[7] T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.

[8] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," Tech. Rep. 16-34, ESAT-STADIUS, KU Leuven (Leuven, Belgium), 2016.

[9] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural networks*, vol. 13, no. 4, pp. 411–430, 2000.

[10] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines, "A Blind Source Separation Technique Using Second-Order Statistics," *IEEE Trans. Signal Process.*, vol. 45, no. 2, pp. 434–444, 1997.

[11] J.-F. Cardoso and A. Souloumiac, "Blind beamforming for non-gaussian signals," in *Radar and Signal Processing, IEE Proc.-F*. IET, 1993, vol. 140, pp. 362–370.

[12] C. E. R. Fernandes, G. Favier, and J. C. M. Mota, "Blind multipath MIMO channel parameter estimation using the Parafac decomposition," in *IEEE International Conference on Communications, 2009. ICC'09.* IEEE, 2009, pp. 1–5.

[13] H. Bousbia-Salah, A. Belouchrani, and K. Abed-Meraim, "Jacobi-like algorithm for blind signal separation of convolutive mixtures," *Electronics Letters*, vol. 37, no. 16, pp. 1049–1050, 2001.

[14] F. Van Eeghem, M. Sørensen, and L. De Lathauwer, "Tensor decompositions with several block-Hankel factors and application in blind system identification," Tech. Rep. 16-39, ESAT-STADIUS, KU Leuven (Leuven, Belgium), 2016.

[15] A. Gorokhov and P. Loubaton, "Subspace-based techniques for blind separation of convolutive mixtures with temporally correlated sources," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 44, no. 9, pp. 813–820, 1997.

[16] A. Mansour, C. Jutten, and P. Loubaton, "Subspace method for blind separation of sources in convolutive mixture," *Proc. EUSIPCO, Trieste, Italy*, pp. 2081–2084, 1996.

[17] H. Liu and G. Xu, "Multiuser blind channel estimation and spatial channel pre-equalization," in *International Conference on Acoustics, Speech, and Signal Processing, 1995. ICASSP-95*. IEEE, 1995, vol. 3, pp. 1756–1759.

[18] A.-J. Van der Veen, S. Talwar, and A. Paulraj, "Blind estimation of multiple digital signals transmitted over fir channels," *IEEE Signal Process. Lett.*, vol. 2, no. 5, pp. 99–102, 1995.

[19] A.-J. Van Der Veen, S. Talwar, and A. Paulraj, "A subspace approach to blind space-time signal processing for wireless communication systems," *IEEE Trans. Signal Process.*, vol. 45, no. 1, pp. 173–190, Jan. 1997.

[20] A. Mansour, A. K. Barros, and N. Ohnishi, "Blind separation of sources: Methods, assumptions and applications," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 83, no. 8, pp. 1498–1512, 2000.

[21] H. Liu and G. Xu, "Closed-form blind symbol estimation in digital communication," *IEEE Trans. Signal Process.*, vol. 43, no. 11, pp. 2714–2723, 1995.

[22] H. Bousbia-Salah, A. Belouchrani, and K. Abed-Meraim, "Blind separation of convolutive mixtures using joint block diagonalization," in *Sixth International Symposium on Signal Processing and its Applications*. IEEE, 2001, vol. 1, pp. 13–16.

[23] K. Abed-Meraim and A. Belouchrani, "Algorithms for joint block diagonalization," in *Signal Processing Conference, 2004 12th European*. IEEE, 2004, pp. 209–212.

[24] L. De Lathauwer and D. Nion, "Decompositions of a Higher-Order Tensor in Block Terms — Part III: Alternating least squares algorithms," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1067–1083, 2008.

[25] 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 J. Optim.*, vol. 23, no. 2, pp. 695–720, 2013.

[26] M. Sørensen, "Convolutive low-rank factorizations via coupled low-rank and Toeplitz structured matrix/tensor decompositions," Tech. Rep. 16-37, ESAT-STADIUS, KU Leuven, Belgium, 2016.

[27] H. Liu and G. Xu, "Smart antennas in wireless systems: uplink multiuser blind channel and sequence detection," *IEEE Trans. Communications*, vol. 45, no. 2, pp. 187–199, 1997.

[28] B. De Moor, "The singular value decomposition and long and short spaces of noisy matrices," *IEEE Trans. Signal Process.*, vol. 41, no. 9, pp. 2826–2838, 1993.

[29] N. Tandon and A. Choudhury, "A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings," *Tribology international*, vol. 32, no. 8, pp. 469–480, 1999.

[30] G. Gelle, M. Colas, and C. Serviere, "Blind source separation: a tool for rotating machine monitoring by vibrations analysis?," *Journal of sound and vibration*, vol. 248, no. 5, pp. 865–885, 2001.