Blind identification of under-determined mixtures based on the characteristic function

Pierre Comon*, Myriam Rajih

I3S laboratory, 2000 route des Lucioles, B.P.121, F-06903 Sophia-Antipolis, France

Abstract

Linear mixtures of independent random variables (the so-called sources) are sometimes referred to as under-determined mixtures (UDM) when the number of sources exceeds the dimension of the observation space. The algorithms proposed are able to identify algebraically a UDM using the second characteristic function (c.f.) of the observations, without any need of sparsity assumption on sources. In fact, by taking higher order derivatives of the multivariate c.f. core equation, the blind identification problem is shown to reduce to a tensor decomposition. With only two sensors, the first algorithm only needs a SVD. With a larger number of sensors, the second algorithm executes an alternating least squares (ALS) algorithm. The joint use of statistics of different orders is possible, and a LS solution can be computed. Identifiability conditions are stated in each of the two cases. Computer simulations eventually demonstrate performances in the absence of sparsity, and emphasize the interest in using jointly derivatives of different orders.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Tensor decomposition; Blind identification; Statistical Independence; Linear mixtures; Parafac

1. Introduction

This paper is devoted to *underdetermined* linear mixtures (UDM), that is, mixtures of independent random sources where the number of sources, N, always exceeds the number of sensors, P. In other words, underdetermined mixtures do not enjoy sparsity properties such as disjoint source spectra, or sources non-permanently present. The latter property is often exploited in speech applications; see [1] among others. The exploitation of sparsity is a promising technique, but is not applicable in the present framework.

Moreover, we are only interested in blind identification (BI), and not in source extraction. These two problems are closely related when the number of sources does not exceed the number of sensors. In fact, a noiseless linear mixture can then be linearly inverted, and looking for its inverse is an equivalent problem [2–4]. Techniques that have been utilized in this framework, such as second order pre-whitening, or deflation, are not applicable for UDM.

Identifiability of linear mixtures received on a *single sensor* requires source distributions to have an indecomposable characteristic function (c.f.) [5,6];

*Corresponding author. Tel.: +33 4 9294 2717/2793; fax: +33 4 9294 2896.

E-mail addresses: comon@i3s.unice.fr (P. Comon), rajih@i3s.unice.fr (M. Rajih).

URL: http://www.i3s.unice.fr.

for instance in digital communications, BPSK sources are indecomposable but QPSK are not. This condition can be deflated for underdetermined mixtures received on two sensors [7,8], as subsequently pointed out. See [9] for a recent analysis of UDM identifiability, and uniqueness of source densities. In contrast, for over-determined mixtures, the only pathological distributions are Gaussian [10,11,6]. In the sequel, it is assumed that an UDM is available on more than one sensor, viz $1 < P < N$. In addition, it is not assumed that spectral or multispectral differences can be exploited as in [12] for instance, and the time dimension is merely ignored, which is relevant in cases where sources are white for instance.

Blind source extraction from underdetermined mixtures is a difficult problem since these mixtures cannot be linearly inverted [13]. On the other hand, BI of the mixture matrix can be performed without extracting the sources (at least in a first stage), as in [14,13,15–17]. More precisely, the methods proposed in [14,13,15,12] only use the data fourth order statistics, whereas in [16] or [8], the information contained in the second c.f. of observations is exploited. We extend this kind of approach by using additional equations, which makes the solution much more stable. Contrary to cumulant based approaches such as [17] or [13], for a given number of sensors, the number of sources is theoretically not limited, which constitutes the main motivation in using the c.f.

The c.f. has been already utilized in [18] to blindly separate sources under the assumption that there are at most as many sources as sensors (i.e. over-determined mixtures). However, the advantage that the c.f. may allow to identify linear mixtures where the number of sources exceeds the number of sensors has neither been noticed nor exploited therein; hence the originality of the present contribution.

Some recent works have addressed the case of under-determined mixtures in a deterministic manner [19]. Such algorithms resort to the so-called Parafac algorithm, which needs the number of sources to be smaller than a particular bound (see [20] and Section 4.2), that depends on both dimensions and diversity of the data. This bound turns out to be often largely exceeded for a small number of sensors. The approach proposed in this paper allows to circumvent this problem by allowing to build another well-conditioned tensor of arbitrarily large order, which permits to enlarge that bound.

2. Assumptions and notation

In accordance with the remarks made in introduction, we assume the observation model below:

$$x = As + w, \qquad (1)$$

where array variables are distinguished from scalars by bold faces, x and s are random vectors of size P and N, respectively, A is a $P \times N$ full rank matrix, and w accounts for modeling errors and additive noise. From now on, its presence is just ignored in the remaining, except when running computer experiments. The entries s_n of vector s are assumed to be non-Gaussian and statistically independent.

For simplicity, we shall restrict our attention in this paper to variables and mixture with values in the real field, $\mathbb{R}$. As pointed out in [8], the immersion of a problem defined in the complex field, $\mathbb{C}$, into a real framework of larger size introduces some additional constraints, which makes the problem more difficult, but at the same time allows a better stability of the solution. In fact, a matrix–vector product in $\mathbb{C}^P$ can be written as a matrix–vector product in $\mathbb{R}^{2P}$, where the latter matrix is *constrained* to be block skew-symmetric. This constraint complicates the identification problem [8] because the mixing matrix has not independent entries anymore. Nevertheless, most of the reasoning developed in this paper applies to the complex case, up to some complication in notation and increase in computational complexity.

We also assume the following hypotheses:

H1 the columns of A are pairwise linearly independent;
H2 source distributions are unknown and non-Gaussian;
H3 the number N of sources is known;
H4 the moments of the sources are unknown, but finite up to some order larger than N.

Under H1, H2, and H3, A can be shown to be essentially unique [6].

Two practical algorithms are subsequently described. The first is a significant improvement of the approach described in [16] and [8], and the second is able to solve the derivative matching problem for $N > 2$ with the help of an ALS algorithm. Both are based on the core functional equation below, which is a direct consequence of source independence:

$$\Psi_x(u) = \sum_{n=1}^{N} \psi_n \left(\sum_{p=1}^{P} A_{pn} u_p \right) \qquad (2)$$

where $\Psi_x(\boldsymbol{u})$ denotes the second c.f. of $\boldsymbol{x}$ defined as $\Psi_x(\boldsymbol{u}) = \log E\{\exp(\jmath\,\boldsymbol{u}^{\mathsf{T}}\boldsymbol{x})\}$, and where $\psi_n(v)$ denotes the second c.f. of source s_n: $\psi_n(v) = \log E\{\exp(\jmath\,vs_n)\}$, where the dotless $\jmath$ denotes the square root of -1. This core equation can be used in an open neighborhood Ω of the origin where Ψ_x does not vanish, which always exists.

3. Algorithm ALGECAF: an algebraic solution

3.1. Description of the algorithm

It is easy to verify that any two derivatives of (2) can be combined in order to cancel the nth term of the sum. More precisely, for any triplet of indices, (n, i, j), $n \leqslant N$, $i, j \leqslant P$, define the differential operator:

$$D_{n,i,j} \stackrel{\text{def}}{=} A_{in}\frac{\partial \Psi_x}{\partial u_j} - A_{jn}\frac{\partial \Psi_x}{\partial u_i}.$$

In other words, $D_{n,i,j}\Psi(\boldsymbol{u})$ does not depend on ψ_n, for any values of (i, j). Thus, by applying such an operator N times for different n's and for arbitrary pairs (i_n, j_n), one eventually gets zero. In order to be able to estimate $\boldsymbol{A}$, it is interesting to fix the pair (i, j), which leads to

$$\left\{\prod_{n=1}^{N} D_{n,i,j}\right\}\Psi_x(\boldsymbol{u}) = \sum_{k=0}^{N} q_k[i,j]\frac{\partial^N \Psi_x(\boldsymbol{u})}{\partial u_j^{N-k}\partial u_i^k} = 0,$$
$$\forall \boldsymbol{u} \in \Omega, \tag{3}$$

where $q_k[i,j]$ are known functions of the (yet unknown) entries of $\boldsymbol{A}$. In order to obtain the exact relation between vector $\boldsymbol{q}[i,j]$ and rows i and j of $\boldsymbol{A}$, it suffices to plug equation (2) into (3), which yields:

$$\sum_{n=1}^{N}\left[\sum_{k=0}^{N} q_k[i,j]A_{jn}^{N-k}A_{in}^k\right]\psi_n^{(N)}\left(\sum_p A_{pn}u_p\right) = 0, \tag{4}$$

where $\psi_n^{(N)}$ denotes the Nth derivative of ψ_n. Since this holds true for any $\boldsymbol{u} \in \Omega$, one can deduce that

$$\sum_{k=0}^{N} q_k[i,j]A_{jn}^{N-k}A_{in}^k = 0, \quad \forall n. \tag{5}$$

The latter homogeneous polynomial equation has infinitely many solutions; but this is not a surprise since matrix $\boldsymbol{A}$ is identifiable only up to post-multiplication by a diagonal invertible matrix. In order to cope with this indeterminacy, one can solve for the ratios A_{in}/A_{jn} instead. However, since some entries can be null, a convenient approach to rooting homogeneous polynomials in several variables is to

immerse the problem in the projective space (i.e. the space including infinity) [21].

In these conditions, (5) shows that the N ratios A_{in}/A_{jn} can be obtained as the N roots in the projective space of a polynomial of degree N, once $\boldsymbol{q}$ has been obtained. So let us focus on the way to obtain $\boldsymbol{q}[i,j]$, and drop indices $[i,j]$ for the sake of simplicity, being understood that these indices are fixed.

Now, imposing (3) to be satisfied on a grid $\mathcal{G}$ of K values $\{\boldsymbol{u}[1], \ldots, \boldsymbol{u}[K]\} \in \Omega$, one can build the over-determined linear system $\boldsymbol{H}[N]\boldsymbol{q} = \boldsymbol{0}$, where $\boldsymbol{H}[N]$ is the $K \times N + 1$ matrix of Nth order derivatives given below

$$\boldsymbol{H}[N] \stackrel{\text{def}}{=} \begin{pmatrix} \frac{\partial^N \Psi_x(\boldsymbol{u}[1])}{\partial u_j^N} & \frac{\partial^N \Psi_x(\boldsymbol{u}[1])}{\partial u_j^{N-1}\partial u_i} & \cdots & \frac{\partial^N \Psi_x(\boldsymbol{u}[1])}{\partial u_i^N} \\ \frac{\partial^N \Psi_x(\boldsymbol{u}[2])}{\partial u_j^N} & \frac{\partial^N \Psi_x(\boldsymbol{u}[2])}{\partial u_j^{N-1}\partial u_i} & \cdots & \frac{\partial^N \Psi_x(\boldsymbol{u}[2])}{\partial u_i^N} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^N \Psi_x(\boldsymbol{u}[K])}{\partial u_j^N} & \frac{\partial^N \Psi_x(\boldsymbol{u}[K])}{\partial u_j^{N-1}\partial u_i} & \cdots & \frac{\partial^N \Psi_x(\boldsymbol{u}[K])}{\partial u_i^N} \end{pmatrix}. \tag{6}$$

Now equations (3), (5), (6) and (8) yield together an algebraic algorithm to blindly identify any two rows of matrix $\boldsymbol{A}$, up to scale and permutation factors

Algorithm ALGECAF(1)

(1) Fix the number N of sources sought (the algorithm can increment on N, starting with $N = P$).
(2) Select two sensor indices $[i,j]$, $1 \leqslant [i,j] \leqslant P$.
(3) Define a grid $\mathcal{G}$ of K values $\boldsymbol{u}[m]$ in a neighborhood Ω of the origin in $\mathbb{R}^P$, $1 \leqslant m \leqslant K$.
(4) Estimate the Nth order derivatives of the joint second characteristic function of observation $[x_i, x_j]$, $\psi_x(\boldsymbol{u})$ on this grid, and store them in a matrix $\boldsymbol{H}[N]$ as defined in (6).
(5) Compute the right singular vector $\boldsymbol{q}$ of $\boldsymbol{H}[N]$ associated with the smallest singular value.
(6) Root the Nth degree polynomial whose coefficients are q_k, $0 \leqslant k \leqslant N$ in the projective space (that includes infinity if necessary).
(7) Associate each root with the ratio A_{in}/A_{jn}.

We shall now show that this algorithm, proved to be unfortunately sometimes rather unstable, can be made more robust by adding extraneous equations. In fact, expression (3) is still null if we take further derivatives:

$$\frac{\partial}{\partial u_\ell}\sum_{k=0}^{N} q_k[i,j]\frac{\partial^N \Psi_x(\boldsymbol{u})}{\partial u_j^{N-k}\partial u_i^k} = 0, \quad \forall \boldsymbol{u} \in \Omega. \tag{7}$$

For instance, for $\boldsymbol{u} = \boldsymbol{0}$, $P = 2$, and $N = 3$, this yields the two fourth order cumulant equations used in [13].

An even more interesting result is that (3) and (7) involve the same unknown q_k, so that they can be combined to build a single larger over-determined system. Indeed, denote $\boldsymbol{H}[N+1, i]$ and $\boldsymbol{H}[N+1, j]$ the two $K \times N + 1$ matrices built from (7), when (i, j) is fixed and $\ell \in \{i, j\}$. Then $\boldsymbol{q}[i, j]$ satisfies the following linear system:

$$\begin{bmatrix} \boldsymbol{H}[N] \\ \boldsymbol{H}[N+1, i] \\ \boldsymbol{H}[N+1, j] \end{bmatrix} \cdot \boldsymbol{q}[i, j] = \boldsymbol{0}, \tag{8}$$

where matrices $\boldsymbol{H}[N, \ell]$, $\ell \in \{i, j\}$, are defined by (7). The corresponding algorithm is given by

Algorithm ALGECAF(2)

Run the same steps as Algorithm ALGECAF(1), but replace matrix $H[N]$ by the block matrix of (8).

More block equations can be obviously stacked at the bottom of (8), by adding higher order derivatives. Let us push one step further, and consider also derivatives of order $N + 2$, which will be proved to be useful especially when $P = 2$. For fixed $[i, j]$, there are three $(N+2)$nd order derivatives, obtained by applying operators $\partial^2 / \partial u_i^2$, $\partial^2 / \partial u_i u_j$, and $\partial^2 / \partial u_j^2$, to (3), which leads to three additional block equations, taking the form below, with obvious notations:

$$\begin{bmatrix} \boldsymbol{H}[N+2, i, i] \\ \boldsymbol{H}[N+2, i, j] \\ \boldsymbol{H}[N+2, j, j] \end{bmatrix} \cdot \boldsymbol{q}[i, j] = \boldsymbol{0}. \tag{9}$$

We then define a third algorithm

Algorithm ALGECAF(3)

Run the same steps as algorithm ALGECAF(1), but compute $\boldsymbol{q}$ as the right singular vector associated with the smallest singular value of the matrix containing the six blocks defined in (8) and (9).

3.2. Identifiability

Algorithms ALGECAF(k) estimate the ratio of any two rows i and j of A. If A has only two rows ($P = 2$), this means that A is estimated up to a scale factor, which is part of the inherent indetermination of the BI problem. We show in this section under what conditions one can make sure that there is a unique solution. If A has more than two rows, an association algorithm is necessary in addition to algorithms ALGECAF, in order to fix extraneous permutation indeterminacies, as pointed out in [8]. But the increase in complexity is important, and the ALESCAF algorithm described in Section 4 should be preferred in that case. On the other hand, ALGECAF algorithms are much more attractive when $P = 2$. So consider from now on in this section, that $P = 2$, so that indices (j, i) may be set to $(1, 2)$ once for all in this section.

Given a $q \times 1$ vector $\boldsymbol{a}$, denote by $\boldsymbol{a}^{\oslash k}$ the non-redundant symmetric Kronecker product of $\boldsymbol{a}$ with itself k times, as defined in [22]; for instance if $\boldsymbol{a}$ is of length q, then $\boldsymbol{a}^{\oslash 2}$ is of length $q(q+1)/2$ and contains all distinct degree 2 cross-products between the entries of $\boldsymbol{a}$. Also denote $A^{\odot k}$ the *column-wise* non-redundant symmetric Kronecker product of A with itself k times. If A is of size $2 \times N$ for instance, then $A^{\odot k}$ is of size $k + 1 \times N$.

Also define the $N \times K$ matrix $\boldsymbol{\Psi}_s^{(M)}$ containing in its (n, k) entry the Mth derivative of the second characteristic function of source n, $\psi_n^{(M)}(\boldsymbol{a}_n^{\mathsf{T}} \boldsymbol{u}[k])$, $1 \leqslant k \leqslant K$, taken at the (n, k)th point of the dual grid $A^{\mathsf{T}} \mathscr{G}$, where $\boldsymbol{a}_n$ is the nth column of A. Lastly denote in bold face $\boldsymbol{\psi}_s^{(M)}[k]$ the N-dimensional vector standing for the kth column of $\boldsymbol{\Psi}_s^{(M)}$.

Lemma 1. *The transpose of the $K \times M + 1$ matrix $\boldsymbol{H}[M]$ defined in* (6) *can be expressed as*

$$\boldsymbol{H}[M]^{\mathsf{T}} = A^{\odot M} \boldsymbol{\Psi}_s^{(M)}. \tag{10}$$

Proof. From the core equation (2), we know that for every point $\boldsymbol{u}[k]$ of grid $\mathscr{G}$, and $\forall \ell$, $0 \leqslant \ell \leqslant M$:

$$\frac{\partial^M \Psi_x}{\partial u_1^{M-\ell} \partial u_2^{\ell}}(\boldsymbol{u}[k]) = \sum_{n=1}^{N} A_{1n}^{M-\ell} A_{2n}^{\ell} \psi_n(\boldsymbol{a}_n^{\mathsf{T}} \boldsymbol{u}[k]). \tag{11}$$

Because of the definition of our column-wise non-redundant Kronecker product, the right-hand side can be seen to be the ℓth entry of the column vector $A^{\odot M} \boldsymbol{\psi}_s^{(M)}[k]$. Now grouping the K block equations one after the other, $1 \leqslant k \leqslant K$, and remarking that matrix $A^{\odot M}$ can be factored out on the left eventually yields the compact expression of Lemma 1. $\square$

Theorem 2. *In addition to hypotheses* H1, H2, *and* H3, *assume hypothesis*:

H4a: *the source second characteristic functions admit finite derivatives up to order N at every point of grid $\mathscr{G}$; in other words, matrix $\boldsymbol{\Psi}_s^{(N)}$ exists and is finite.*

Then model (1) *with* $P = 2$ *is identifiable with algorithm ALGECAF*(1) *if and only if the two conditions below hold true*

(i) *matrix* $A^{\odot N}$ *is of full rank* N,
(ii) *matrix* $\mathbf{\Psi}_s^{(N)}$ *is of full rank* N.

Note that condition (i) above is the same as that found in Theorem A.3.3 in [6]. Our additional condition (ii) is necessary because of the use of a particular numerical algorithm.

Proof. First, by definition (3), $\boldsymbol{H}[N]$ has $N+1$ columns. Yet, in order for the equation $\boldsymbol{H}[N]\boldsymbol{q} = 0$ to have a single solution, $\boldsymbol{H}[N]$ must admit a one-dimensional right singular space. This means that $\boldsymbol{H}[N]$ must be of rank exactly N. Now, from Lemma 1, $\boldsymbol{H}[N]$ is the product of two matrices of rank at most N. It is thus clear that they must both be of full rank, that is of rank N. The converse is derived along the same lines. $\square$

One of the consequences of this theorem is that the number of grid points, K, must be at least as large as N when using algorithm ALGECAF(1).

The same reasoning can be done with algorithms ALGECAF(2) and ALGECAF(3). In fact, let $\mathscr{A}[N] = A^{\odot N}$, let $\mathscr{A}[N+1, 1]$ denote the $N+1 \times N$ sub-matrix of $A^{\odot N+1}$ formed of its $N+1$ first rows, and $\mathscr{A}[N+1, 2]$ the sub-matrix of $A^{\odot N+1}$ formed of its $N+1$ last rows. Then we have for instance:

Corollary 3. *In addition to hypotheses* H1, H2, *and* H3, *assume hypothesis*:

H4b: *the source second characteristic functions admit finite derivatives up to order* $N+1$ *at every point of grid* $\mathscr{G} \subset \Omega$; *in other words, matrices* $\mathbf{\Psi}_s^{(N)}$ *and* $\mathbf{\Psi}_s^{(N+1)}$ *exist and are finite.*

Then model (1) *with* $P = 2$ *is identifiable with algorithm ALGECAF*(2) *if the following matrix product is of rank* N:

$$[\mathscr{A}[N], \mathscr{A}[N+1, 1], \mathscr{A}[N+1, 2]]$$
$$\times \begin{bmatrix} \Psi_s^{(N)} & 0 & 0 \\ 0 & \Psi_s^{(N+1)} & 0 \\ 0 & 0 & \Psi_s^{(N+1)} \end{bmatrix}.$$

The proof goes along the same lines as in that of Theorem 2. In practice, because of estimation and rounding errors, the matrix product above may be full rank, namely of rank $N+1$. In that case, as

vector $\boldsymbol{q}$ is chosen to be the singular vector associated with the $N+1$st singular value (the smallest), uniqueness is still ensured. On the other hand, the matrix product above should not have a rank strictly lower than N.

Now, it is important to note that K does not need to be larger than N anymore, as it was the case in ALGECAF(1); with ALGECAF(2), the necessary condition indeed deflates to $3K \geqslant N$.

Remark 4. Matrix $[\mathscr{A}[N+q, 1], \mathscr{A}[N+q, 2], \ldots, \mathscr{A}[N+q, q+1]]$, $q \geqslant 0$, can be rearranged into a set of Hankel blocks $\mathscr{H}_n$ by permuting the columns. More precisely, Eq. (11) may be rewritten as

$$[\boldsymbol{H}[N+q, 1]^\mathsf{T}, \ldots, \boldsymbol{H}[N+q, q+1]^\mathsf{T}]$$
$$= [\mathscr{H}_1, \mathscr{H}_2, \ldots, \mathscr{H}_N] \cdot (\mathbf{\Psi}_s^{(N+q)} \otimes \boldsymbol{I}_{q+1}),$$

where

$$\mathscr{H}_n \stackrel{\text{def}}{=} \begin{bmatrix} A_{1n}^{N+q} & \cdots & A_{1n}^N A_{2n}^q \\ A_{1n}^{N+q-1}A_{2n} & \cdots & A_{1n}^{N-1}A_{2n}^{q+1} \\ & & \\ A_{1n}^q A_{2n}^N & \cdots & A_{2n}^{N+q} \end{bmatrix}$$

is clearly of rank 1.

A similar corollary holds true if we also use derivatives of order $N+2$. The necessary condition then becomes $6K \geqslant N$. In order to illustrate this in a simple manner, just consider the particular case given by the corollary below when $\mathscr{G} = \{0\}$. The extension to $K > 1$ is straightforward and would just complicate the presentation. To simplify the notation, denote $\mathscr{C}_x^{(p,q)}$ the cumulants of $\boldsymbol{x}$ and $\mathscr{K}_n^{(r)}$ those of s_n, which are known to be the (weighted) derivatives of the c.f. at the origin:

$$\mathscr{C}_x^{(p,q)} \stackrel{\text{def}}{=} (-\jmath)^{p+q} \frac{\partial^{p+q}\Psi_x}{\partial u_1^p \partial u_2^q}(0), \quad \mathscr{K}_n^{(r)} \stackrel{\text{def}}{=} (-\jmath)^r \frac{\partial^r \psi_n}{\partial v^r}(0)$$

with $\jmath \stackrel{\text{def}}{=} \sqrt{-1}$, and define the hypothesis:

H4c: the source second characteristic functions admit finite derivatives up to order $N+2$ at the origin; in other words, matrices $\mathbf{\Psi}_s^{(N)}$, $\mathbf{\Psi}_s^{(N+1)}$, and $\mathbf{\Psi}_s^{(N+2)}$ exist and are finite.

When $\mathscr{G} = \{0\}$, then $K = 1$ and matrices $\mathbf{\Psi}_s^{(r)}$ are merely $N \times 1$ vectors, of the form

$$\mathbf{\Psi}_s^{(r)} = [\mathscr{K}_1^{(r)}, \ldots, \mathscr{K}_N^{(r)}]^\mathsf{T}$$

Then we have the following result:

Corollary 5. *Under assumptions* H1, H2, H3, *and* H4c, *ALGECAF*(3) *solves for* $\boldsymbol{q}$ *the over-determined*

system in the LS sense; $\boldsymbol{q}^{\mathsf{T}} \boldsymbol{C}_x = 0$, where $\boldsymbol{C}_x$ is formed of three Hankel blocks, each of them containing cumulants of order N, $N+1$, and $N+2$, respectively

$$\boldsymbol{C}_x \stackrel{\text{def}}{=} \begin{bmatrix} \mathscr{C}_x^{(N,0)} & \mathscr{C}_x^{(N+1,0)} & \mathscr{C}_x^{(N,1)} & \mathscr{C}_x^{(N+2,0)} & \mathscr{C}_x^{(N+1,1)} & \mathscr{C}_x^{(N,2)} \\ \mathscr{C}_x^{(N-1,1)} & \mathscr{C}_x^{(N,1)} & \mathscr{C}_x^{(N-1,2)} & \mathscr{C}_x^{(N+1,1)} & \mathscr{C}_x^{(N,2)} & \mathscr{C}_x^{(N-1,3)} \\ \vdots & & & & & \\ \mathscr{C}_x^{(0,N)} & \mathscr{C}_x^{(1,N)} & \mathscr{C}_x^{(0,N+1)} & \mathscr{C}_x^{(2,N)} & \mathscr{C}_x^{(1,N+1)} & \mathscr{C}_x^{(0,N+2)} \end{bmatrix}.$$

The solution $\boldsymbol{q}$ is unique if $\boldsymbol{C}_x$ is of rank N, and if $N \leqslant 6$. One can equivalently test the rank of the product below

$$[\mathscr{A}[N], \mathscr{A}[N+1,1], \mathscr{A}[N+1,2], \mathscr{A}[N+2,1],$$
$$\mathscr{A}[N+2,2], \mathscr{A}[N+2,3]]$$
$$\times \begin{bmatrix} \boldsymbol{\Psi}_s^{(N)} & 0 & 0 \\ 0 & \boldsymbol{I}_2 \otimes \boldsymbol{\Psi}_s^{(N+1)} & 0 \\ 0 & 0 & \boldsymbol{I}_3 \otimes \boldsymbol{\Psi}_s^{(N+2)} \end{bmatrix}$$

or the other form of this product, obtainable with the help of Remark 4.

Example 6. To make it clear, in order to identify a 2×3 mixture, one wishes to estimate a vector $\boldsymbol{q}$ of dimension $N+1 = 4$. To do this, one can either build a linear system with third order derivatives taken at (at least) three different points of Ω, or the two types of fourth order derivatives taken at (at least) two different points of Ω. But one can also build a system combining both as in (8), including then both third and fourth order derivatives, possibly taken at a single point of Ω (the linear system needs in fact at least $N = 3$ rows in order to have a null space of dimension at most 1). Adding (9) taken at the origin yields three additional lines to (8), which makes the null space much more stable.

Example 7. Let us take a more concrete example, and suppose that all sources are BPSK, that is, they take their values in the set $\{-1, 1\}$ with equal probabilities. Also suppose that we wish to use only the origin as a grid point. Then $\forall n$, $\psi_n(v) = \log \cos v$, and the cumulants of order r can be calculated as rth order derivatives of ψ at the origin, weighted by j^r. This yields zero odd order cumulants, and $\mathscr{K}^{(2)} = 1$, $\mathscr{K}^{(4)} = -2$, $\mathscr{K}^{(6)} = 16$. In that case, $\boldsymbol{I}_2 \otimes \boldsymbol{\Psi}_s^{(4)}$ and $\boldsymbol{I}_4 \otimes \boldsymbol{\Psi}_s^{(6)}$ are of rank 2 and 4, respectively, but odd order vectors $\boldsymbol{\Psi}_s^{(2q+1)}$ are null.

4. Algorithm ALESCAF: an alternating least squares solution

4.1. Description of the algorithm

As already pointed out, the ALGECAF algorithm is very attractive for the BI of $2 \times N$ mixtures, but more complicated to implement for $P > 2$ [8]. Therefore, there is a great interest in looking for other ways of exploiting the c.f. From (2), one can easily obtain that

$$\frac{\partial^3 \Psi_x(\boldsymbol{u})}{\partial u_i \partial u_j \partial u_p} = \sum_{n=1}^{N} A_{in} A_{jn} A_{pn} \psi_n^{(3)} \left(\sum_q A_{qn} u_q \right). \quad (12)$$

Again, take this equation on K points $\boldsymbol{u}[k] \in \mathscr{G} \subset \Omega$. Then, storing the left-hand side of (12) in a family of symmetric matrices $T_{ij}[p, k]$, and denoting $D_{kn} = \psi_n^{(3)}(\sum_q A_{qn} u_q[k])$, (12) can be arranged in compact form as

$$\boldsymbol{T}[p, k] = \boldsymbol{A} \operatorname{Diag}\{\boldsymbol{A}(p, :)\} \operatorname{Diag}\{\boldsymbol{D}(k, :)\} \boldsymbol{A}^{\mathsf{T}}, \quad (13)$$

with $1 \leqslant p \leqslant P, 1 \leqslant k \leqslant K$, where $\operatorname{Diag}\{\boldsymbol{v}\}$ denotes the diagonal matrix whose entries are those of vector $\boldsymbol{v}$, and where $\boldsymbol{A}(p, :)$ denotes the pth row of $\boldsymbol{A}$. Expression (13) is a four way PARAFAC model and can be solved using an ALS algorithm described in Appendix A.1. This procedure constitutes algorithm ALESCAF (Alternating LEast Squares identification based on the ChAracteristic Function) and is able to compute $\boldsymbol{A}$ and $\boldsymbol{D}$ from symmetric matrices $\boldsymbol{T}[p, k]$ (the implicit dependence of $\boldsymbol{D}$ on $\boldsymbol{A}$ is ignored).

4.2. Identifiability

According to [20], essential uniqueness (i.e. uniqueness up to scale and permutation among columns) of $\boldsymbol{A}$ is achieved by using the Parafac algorithm since Kruskal's inequality is verified

$$3 r_k(\boldsymbol{A}) + r_k(\boldsymbol{D}) \geqslant 2 \operatorname{rank}\{\boldsymbol{T}\} + 3, \quad (14)$$

where $r_k(\boldsymbol{A})$ is the Kruskal rank [20] of $\boldsymbol{A}$. This result can be traced back to 1977, and is not obvious. It indeed requires tedious calculations; we refer to [20,23] for more details.

Results are expected to be better when increasing the order of the statistics as we move away from the PARAFAC limit, but this needs to be verified by simulations.

More precisely, we have the theorem below.

Theorem 8. *Model* (1) *is identifiable with the help of algorithm ALESCAF if the following conditions are all satisfied*:

C1 *the unknown second characteristic functions of the sources and their derivatives up to order* 3 *exist, are finite and do not vanish in a neighborhood* Ω *of the origin*;

C2 *the Kruskal rank* [20] *of the mixing matrix* A *verifies* $r_k(A) = P$;

C3 $r_k(D) = N$;

C4 *for a given P the number of sources must verify*: $N \leqslant 3P - 3$.

Proof.

- Condition C1 allows us to derive expression (12) and build the four-way array T.
- If conditions C2, C3, and C4 are verified, then:

$$3r_k(A) + r_k(D) \geqslant 2\,\text{rank}\{T\} + 3 \qquad (15)$$

which means that the uniqueness conditions of the four-way PARAFAC model is achieved, leading to the uniqueness of A. $\square$

C3 is easy to achieve by taking a large number of points on a grid $\mathscr{G}: K \gg N$. But this increases the complexity of the ALS algorithm used to fit the PARAFAC model, as the fourth dimension of the tensor T will increase, leading to large size of the data. A good choice of the points of the grid can lead to an optimal value of K, thus reducing the complexity of computation.

Condition C2 can be relaxed to

C2′ A is full row rank and its columns are pairwise linearly independent.

But this implies a limitation on the number of sources and/or an increase of complexity with the use of arrays of higher order. Let us take an example.

Example 9. We consider the following mixing matrix:

$$A = \begin{pmatrix} 1 & 0 & 2 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

where A is full row rank (rank$(A) = 3$) but is not full Kruskal rank ($r_k(A) = 2$). Hence, relation (15),

Table 1
Maximum number of sources, N, as a function of the number of sensors, P, when PARAFAC is used with third order c.f. derivatives, under the conditions of Theorem 8

P	2	3	4	5	6	7	8
N	3	6	9	12	15	18	21

which ensures uniqueness of A, is no longer verified. To cope with this problem we build a tensor of order five, $T^{(5)}$, by deriving one more time the core equation (2). Then (15) becomes:

$$4r_k(A) + r_k(D) \geqslant 2\,\text{rank}\{T^{(5)}\} + 4$$

and uniqueness is achieved. At the same time the complexity of the PARAFAC model increases, since tensor $T^{(5)}$ is of higher order (5 instead of 4) as well as the derivatives (order 4 instead of 3).

In ALESCAF the number of sources N is limited by condition C4. Table 1 gives the maximum value of N as a function of P for $2 \leqslant P \leqslant 8$. In theory N is not limited when using ALGECAF, but as N increases the complexity of the algorithm increases since we need to derive the second joint characteristic function N times.

5. Computer results

Estimates of matrices $H[\cdot]$ are computed in the following manner. First, all derivatives of $\Psi_x(u)$ of required order are formally expressed as a function of moments $\mu(n, u[k]) = J^n E\{x^n e^{Jx^\top u[k]}\}$; see Appendix A.3. Then sample moments $\hat{\mu}(n, u[k]) = \frac{1}{M}\sum_{m=1}^{M} J^n \{x[m]^n e^{Jx[m]^\top u[k]}\}$ are computed, yielding eventually estimates of entries of H (here x^n stands for $\Pi_p x_p^{n_p}$). Other useful expressions are reported in Appendix A.3. This formal approach to take the derivatives of $\Psi_x(u)$ is more accurate, but turns out to be more sensitive to deviations of the sample mean from zero. Another possible approach for computing the derivatives of $\Psi_x(u)$ is a numerical one. It consists of estimating the characteristic function $\Psi_x(u)$ of the observation on a grid $\mathscr{G}$ containing the origin. We merely utilized the sample estimate:

$$\hat{\Psi}_x(u) = \log\left[\frac{1}{T}\sum_{t=1}^{T} \exp\{J\Re(x(t)^\top u)\}\right], \qquad (16)$$

where vectors $x(t)$ are realizations of the random variable x. The successive derivatives can then be obtained from the values of $\hat{\Psi}_x(u)$ over a grid included in $\mathscr{G}$ by central finite differences.

The number of sensors is taken to be $P = 2$, and the number of sources ranges from $N = 3$ to $N = 6$. Sources are either BPSK (that is, they take their values in $\{-1, 1\}$ with equal probabilities), or PAM4 (viz they take their values in $\{-3, -1, 1, 3\}$ with equal probabilities).

The 2×3 mixing matrix is

$$A = \begin{bmatrix} 1 & 0 & \cos(\pi/6) \\ 0 & 1 & \sin(\pi/6) \end{bmatrix}$$

and in the 2×6 case, the ith column of A is defined as $[\cos((i-1) * \pi/12), \sin((i-1) * \pi/12)]^{\mathsf{T}}$, $1 \leqslant i \leqslant 6$.

The performance criterion is that proposed in [13], namely the minimum Frobenius distance between the actual and estimated unit-column norm mixing matrices over the set of N-dimensional permutations.

Two types of results are reported for ALGECAF. First, the influence of the noise alone is analyzed. For this purpose, a block of BPSK data of length 2^N is generated with exactly all possible combinations of $\{-1, 1\}$; in this manner, sources are always seen as perfectly independent. Independent realizations of a Gaussian noise are added, with various noise level (SNR). Gaps averaged over 21 trials are reported in Figs. 3 and 4 with the label "infinite sample size". The same experiment is also run for PAM4 sources, with blocks of data of length exactly 4^N.

Second, other experiments are reported where source blocks are also randomly generated; therefore sources are seen as statistically independent only for asymptotically large block lengths. As reported in Figs. 1 and 2, one can observe a plateau for high SNRs. This plateau has a strictly positive level if derivatives of order N only are used in the presence of N sources (except in Fig. 3 for infinite data length and 3 sources). The joint use of derivatives of order $N + 1$ decreases significantly the level of the plateau, as can be seen in Figs. 1, 2, and 4.

In all these experiments, the grid utilized contained $K = 3$ points located close to the origin: $(0.001, 0.01)$, $(0.01, 0.001)$, and $(0.01, 0.01)$. This particular choice has been guided by the identifiability results proved in Section 4.2.
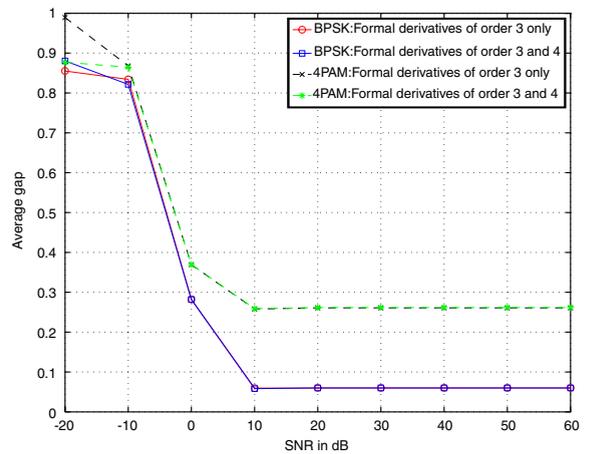
Fig. 1. Gap between estimated and actual mixing matrix for $(P, N) = (2, 3)$ and block lengths of 1000 samples; median values over 31 independent trials of sources and noise are plotted.
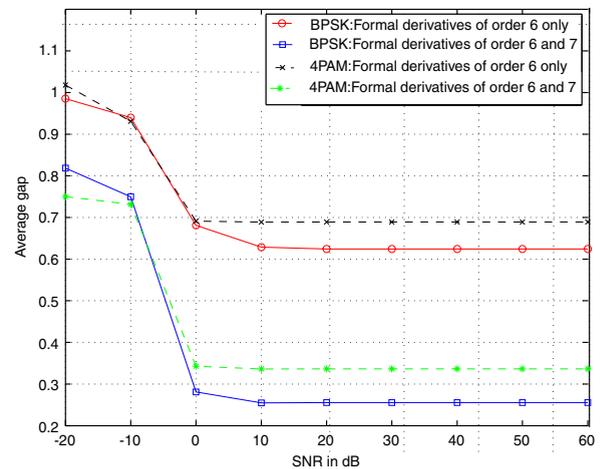
Fig. 2. Gap between estimated and actual mixing matrix for $(P, N) = (2, 6)$ and block lengths of 10 000 samples; median values over 21 independent trials of sources and noise are plotted.

Now, the influence of noise on ALESCAF algorithm applied on these "infinite" blocks of data is also analyzed this way. We start with SNR = 60 dB, check for convergence, and use the value of the corresponding loading matrices A and D to initialize the next ALESCAF algorithm for SNR = 50 dB and so on. By doing so, one expects to access ultimate performances (i.e. performances should be poorer in actual situations). The results are reported in Fig. 5; they are slightly worse than for algorithm ALGECAF, which makes sense. But the advantage of ALESCAF is that it is not limited to the case of 2 sensors.
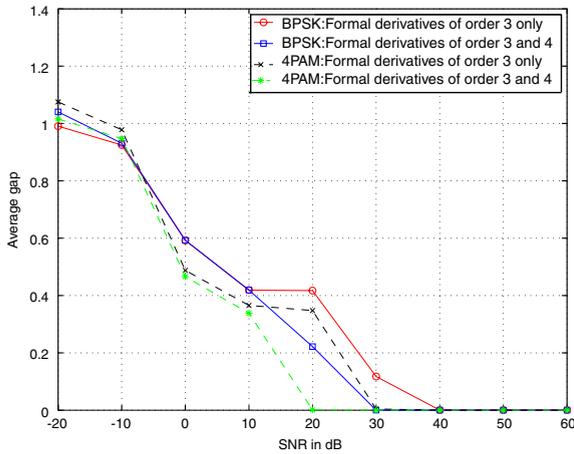
Fig. 3. Gap between estimated and actual mixing matrix for $(P, N) = (2, 3)$ and infinite block lengths; median values over 21 trials of noise are plotted.
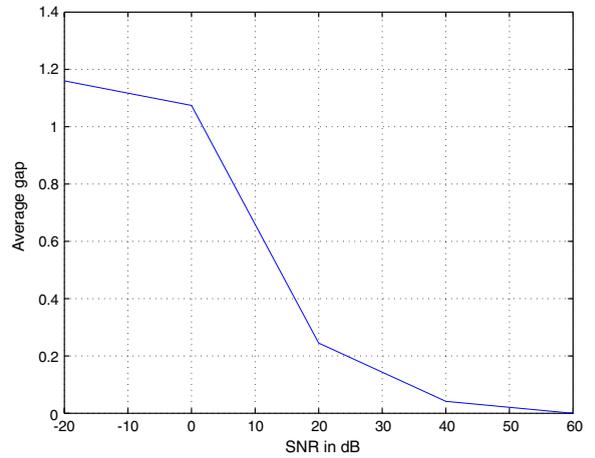
Fig. 5. Gap between estimated and actual mixing matrix for $(P, N) = (2, 3)$, with algorithm ALESCAF; median gap values over 21 independent trials.
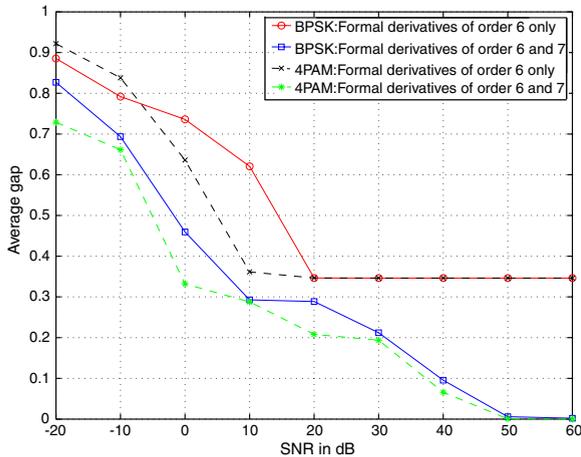
Fig. 4. Gap between estimated and actual mixing matrix for $(P, N) = (2, 6)$ and infinite block lengths; median values over 21 trials of noise are plotted; median values over 21 trials of noise are plotted.

6. Concluding remarks

Our contribution was three-fold:

(i) we have demonstrated that it was possible to derive an algebraic solution to the $2 \times N$ BI problem by simultaneously using derivatives of different orders, and that it improves the stability of the solution;

(ii) we have made the connection with cumulant-based approaches, and proved that improvement (i) also applies to the joint use of cumulants of different orders, larger than or equal to N;

(iii) we showed that an ALS algorithm of PARAFAC type could be utilized to identify a $P \times N$

mixture, and that only third order derivatives of the c.f. are necessary, although higher orders can also be used.

Future works include: (a) improvement of the convergence of ALS algorithms, slow for topological reasons (likely because of a lack of closure [24]), for instance along the lines of [25]; (b) in order to account for a possibly different variance in estimates of moments of different orders, a weighting can be rather easily introduced, and may improve on asymptotic performance; (c) one could try to take into account part of the symmetry in the PARAFAC algorithm, for instance as outlined in Appendix A.2.

Acknowledgment

Appendix A

A.1. The PARAFAC algorithm

Minimizing the gap between both sides of (13) consists of minimizing [26,27]:

$$\Upsilon = \sum_{p,k} \| T[p,k] - A \operatorname{Diag}\{C(p,:)\} \operatorname{Diag}\{D(k,:)\} B^{\top} \|^2$$

(17)

with respect to matrices A, B, C and D, if the symmetry constraint is relaxed. The set of matrices $T[p,k]$ defined in (13) can be stored in a tensor

T_{ijpk}. Then, this problem can be solved with the help of the Harshman's PARAFAC algorithm [20,28], originally developed for third order tensors, improved by Bro [29] and recently accelerated [25]. The computer experiments reported for ALESCAF in this paper have been run with the Enhanced Line Search (ELS) algorithm described in [25].

A.2. Towards a symmetric PARAFAC algorithm

Define the family of diagonal matrices $\Lambda[p,k] = \mathrm{Diag}\{A(p,:)\}\,\mathrm{Diag}\{D(k,:)\}$. Ignoring the dependence of Λ on A, one can try to impose the symmetry $A = B$ in (17); but things are more complicated because the optimization criterion (17) is not quadratic anymore in the unknown rectangular matrix B. Let $\lambda[k] = \mathbf{diag}\,\Lambda[k]$. Two writings are derived in order to obtain stationary values with respect to the rectangular matrix and to the diagonal one:

$$\Upsilon = \sum_k \| T[k] - B\Lambda[k]B^\dagger \|^2 \tag{18}$$

and, with $t[k] = \mathbf{vec}\,T[k]$ and an appropriate definition of $\mathscr{B}$:

$$\Upsilon = \sum_k \| t[k] - \mathscr{B}\lambda[k] \|^2. \tag{19}$$

Some manipulations would show that the stationary values $\lambda[k]$ are given by

$$\lambda[k] = \{\mathscr{B}^\dagger\mathscr{B}\}^{-1}\mathscr{B}^\dagger t[k]. \tag{20}$$

Last, the stationary value of each column $b[\ell]$ of matrix B is the dominant eigenvector of the Hermitian matrix

$$P[\ell] = \frac{1}{2}\sum_k \lambda_\ell[k]\{\tilde{T}[k;\ell]^\dagger + \tilde{T}[k;\ell]\}, \tag{21}$$

where $\tilde{T}[k;\ell] \stackrel{\text{def}}{=} T[k] - \sum_{n\neq\ell} \lambda_n[k]b[n]b[n]^\dagger$. A LS solution is computed when matrices involved are singular. This algorithm turns out to have strong similarities with a work of Yeredor [30], developed for diagonalizing a set of square matrices by an invertible transform, i.e. applicable when $\mathrm{rank}\{T\} \leqslant P$; however, this algorithm theoretically also works for square tensors of rank larger than P.

A.3. Formal derivatives of the second characteristic function

Although higher order derivatives have been considered (and calculated with MAPLE up to order 7), let us limit our illustration to the third order derivatives of $\Psi_x(u)$, for the sake of simplicity. Denote by E the first joint characteristic function of $x \stackrel{\text{def}}{=} (x_1, x_2)$ at $u \stackrel{\text{def}}{=} (u, v)$:

$$E = \Phi_{(x_1,x_2)}(u,v) = \mathrm{E}\{\exp(\jmath u x_1 + \jmath v x_2)\},$$

where the dotless $\jmath$ stands for $\sqrt{-1}$.

The successive derivatives of E with respect to u and v are given by

$$E_1 = \frac{\partial \Phi_{(x_1,x_2)}(u,v)}{\partial u} = \mathrm{E}\{\jmath x_1 \exp(\jmath u x_1 + \jmath v x_2)\},$$

$$E_2 = \frac{\partial^2 \Phi_{(x_1,x_2)}(u,v)}{\partial u^2} = \mathrm{E}\{-x_1^2 \exp(\jmath u x_1 + \jmath v x_2)\},$$

$$E_3 = \frac{\partial^3 \Phi_{(x_1,x_2)}(u,v)}{\partial u^3} = \mathrm{E}\{-\jmath x_1^3 \exp(\jmath u x_1 + \jmath v x_2)\},$$

$$F_1 = \frac{\partial \Phi_{(x_1,x_2)}(u,v)}{\partial v} = \mathrm{E}\{\jmath x_2 \exp(\jmath u x_1 + \jmath v x_2)\},$$

$$F_2 = \frac{\partial^2 \Phi_{(x_1,x_2)}(u,v)}{\partial v^2} = \mathrm{E}\{-x_2^2 \exp(\jmath u x_1 + \jmath v x_2)\},$$

$$F_3 = \frac{\partial^3 \Phi_{(x_1,x_2)}(u,v)}{\partial v^3} = \mathrm{E}\{-\jmath x_2^3 \exp(\jmath u x_1 + \jmath v x_2)\},$$

$$D_{11} = \frac{\partial^2 \Phi_{(x_1,x_2)}(u,v)}{\partial u\partial v} = \mathrm{E}\{-x_1 x_2 \exp(\jmath u x_1 + \jmath v x_2)\},$$

$$T_{21} = \frac{\partial^3 \Phi_{(x_1,x_2)}(u,v)}{\partial u^2\partial v} = \mathrm{E}\{-\jmath x_1^2 x_2 \exp(\jmath u x_1 + \jmath v x_2)\},$$

$$T_{12} = \frac{\partial^3 \Phi_{(x_1,x_2)}(u,v)}{\partial u\partial v^2} = \mathrm{E}\{-\jmath x_1 x_2^2 \exp(\jmath u x_1 + \jmath v x_2)\}.$$

As $\Psi_x(u,v) = \log(\Phi_x(u,v))$ we have:

$$\frac{\partial \Psi_x(u,v)}{\partial u} = \frac{\partial \log(E)}{\partial u} = \frac{E_1}{E},$$

$$\frac{\partial^2 \Psi_x(u,v)}{\partial u^2} = \frac{\partial}{\partial u}\left(\frac{E_1}{E}\right) = \frac{E_2 E - E_1^2}{E^2},$$

$$\frac{\partial^3 \Psi_x(u,v)}{\partial u^3} = \frac{\partial}{\partial u}\left(\frac{E_2 E - E_1^2}{E^2}\right)$$

$$= \frac{E_3 E^2 - 3E_2 E_1 E + 2E_1^3}{E^3},$$

$$\frac{\partial^3 \Psi_x(u,v)}{\partial u^2 \partial v} = \frac{\partial}{\partial v}\left(\frac{E_2 E - E_1^2}{E^2}\right)$$
$$= \frac{T_{21} E^2 - E_2 F_1 E - 2E_1 D_1 E + 2E_1^2 F_1}{E^3}.$$

Using the symmetry between the variables u and v we obtain derivatives with respect to v:

$$\frac{\partial \Psi_x(u,v)}{\partial v} = \frac{\partial \log(E)}{\partial v} = \frac{F_1}{E},$$

$$\frac{\partial^2 \Psi_x(u,v)}{\partial v^2} = \frac{\partial}{\partial v}\left(\frac{F_1}{E}\right) = \frac{F_2 E - F_1^2}{E^2},$$

$$\frac{\partial^3 \Psi_x(u,v)}{\partial v^3} = \frac{\partial}{\partial v}\left(\frac{F_2 E - F_1^2}{E^2}\right)$$
$$= \frac{F_3 E^2 - 3F_2 F_1 E + 2F_1^3}{E^3},$$

$$\frac{\partial^3 \Psi_x(u,v)}{\partial u \partial v^2} = \frac{\partial}{\partial u}\left(\frac{F_2 E - F_1^2}{E^2}\right)$$
$$= \frac{T_{12} E^2 - F_2 E_1 E - 2F_1 D_1 E + 2F_1^2 E_1}{E^3}.$$

References

[1] T.W. Lee, M.S. Lewicki, et. al., Blind source separation of more sources than mixtures using overcomplete representations, IEEE Signal Process. Lett. 6(4) (1999) 87–90.

[2] P. Comon, Independent component analysis, a new concept?, Signal Processing 36 (3) (1994) 287–314 (special issue on higher-order statistics.

[3] J.F. Cardoso, High-order contrasts for independent component analysis, Neural Comput. 11 (1) (1999) 157–192.

[4] A. Hyvärinen, J. Karhunen, E. Oja, Independent Component Analysis, Wiley, New York, 2001.

[5] E. Lukacs, Characteristic functions, Griffin, London, 1970.

[6] A.M. Kagan, Y.V. Linnik, C.R. Rao, Characterization Problems in Mathematical Statistics, Probability and Mathematical Statistics, Wiley, New York, 1973.

[7] A. Taleb, C. Jutten, On underdetermined source separation, in: ICASSP99, Phoenix, Arizona, 1999.

[8] P. Comon, M. Rajih, Blind identification of complex underdetermined mixtures, in: ICA Conference, Granada, Spain, 2004, pp. 105–112.

[9] J. Eriksson, V. Koivunen, Identifiability, separability and uniqueness of linear ICA models, IEEE Signal Process. Lett. (2004) 601–604.

[10] W. Feller, An Introduction to Probability Theory and its Applications, vol. II, Wiley, New York, 1966.

[11] G. Darmois, Analyse générale des liaisons stochastiques, Rev. Inst. Internat. Stochastic. 21 (1953) 2–8.

[12] A. Ferreol, L. Albera, P. Chevalier, Fourth order blind identification of underdetermined mixtures of sources (FOBIUM), in: ICASSP'03, Hong Kong, China, 2003, pp. 41–44.

[13] P. Comon, Blind identification and source separation in 2 × 3 under-determined mixtures, IEEE Trans. Signal Process. (2004) 11–22.

[14] J.F. Cardoso, Super-symmetric decomposition of the fourth-order cumulant tensor. Blind identification of more sources than sensors, in: Proceedings of the ICASSP, Toronto, 1991, pp. 3109–3112.

[15] L. de Lathauwer, P. Comon, B. de Moor, ICA algorithms for 3 sources and 2 sensors, in: Sixth Signal Processing Workshop on Higher Order Statistics, Caesarea, Israel, 1999, pp. 116–120.

[16] A. Taleb, An algorithm for the blind identification of n independent signals with 2 sensors, in: Sixth International Symposium on Signal Processing and its Applications (ISSPA'01), vol. 1, IEEE, Kuala Lumpur, 2001, pp. 5–8.

[17] L. Albera, A. Ferreol, P. Comon, P. Chevalier, Blind identification of overcomplete mixtures of sources (BIOME), Linear Algebra Appl.

[18] A. Yeredor, Blind source separation via the second characteristic function, Signal Processing 80 (2000) 897–902.

[19] N.D. Sidiropoulos, R. Bro, G.B. Giannakis, Parallel factor analysis in sensor array processing, IEEE Trans. Signal Process. 48 (8) (2000) 2377–2388.

[20] J.B. Kruskal, Three-way arrays: rank and uniqueness of trilinear decompositions, Linear Algebra Appl. 18 (1977) 95–138.

[21] I.R. Shafarevitch, Basic Algebraic Geometry 1 Varieties in Projective Space, Springer, Berlin, 1994.

[22] P. Comon, Tensor decompositions, in: J.G. McWhirter, I.K. Proudler (Eds.), Mathematics in Signal Processing, vol. V, Clarendon Press, Oxford, UK, 2002, pp. 1–24.

[23] N.D. Sidiropoulos, R. Bro, On the uniqueness of multilinear decomposition of N-way arrays, J. Chem. 14 (2000) 229–239.

[24] P. Comon, Canonical tensor decompositions, Research Report RR-2004-17, I3S, 17 June 2004.

[25] M. Rajih, P. Comon, Enhanced line search: a novel method to accelerate Parafac, in: Eusipco'05, Antalya, Turkey, 2005.

[26] R.A. Harshman, Foundations of the Parafac procedure: models and conditions for an explanatory multimodal factor analysis, UCLA Working Papers in Phonetics, vol. 16, 1970, pp. 1–84.

[27] A. Smilde, R. Bro, P. Geladi, Multi-way Analysis, with Applications in the Chemical Sciences, Wiley, New York, 2004.

[28] R.B.A.S. Leurgans, R.T. Ross, A decomposition for three-way arrays, Matrix Anal. Appl. 14 (4) (1993) 1064–1083.

[29] R. Bro, C.A. Andersson, Improving the speed of multiway algorithms, part II: compression, Chemometrics and Intelligent Laboratory Systems 42 (1998) 105–113.

[30] A. Yeredor, Non-orthogonal joint diagonalization in the LS sense with application in blind source separation, IEEE Trans. Signal Process. 50 (7) (2002) 1545–1553.