BLIND IDENTIFICATION OF UNDERDETERMINED MIXTURES OF COMPLEX SOURCES BASED ON THE CHARACTERISTIC FUNCTION

Xavier Luciani[(1)], André L. F. de Almeida[(2)], Pierre Comon[(1)]

[(1)] I3S Laboratory, University of Nice-Sophia Antipolis (UNS), CNRS, France.
Federal University of Ceará, Fortaleza, Brazil.
E-mails:{luciani,pcomon}@i3s.unice.fr, andre@gtel.ufc.br

ABSTRACT

In this work we consider the problem of blind identification of underdetermined mixtures using the generating function of the observations. This approach has been successfully applied on real sources but had not been extended to the more attractive case of complex mixtures of complex sources. This is the main goal of the present study. By developing the core equation in the complex case, we arrive at a particular tensor stowage which involves an original tensor decomposition. Exploiting this decomposition, an algorithm is proposed to blindly estimate the mixing matrix. Three versions of this algorithm based on 2nd, 3rd and 4th-order derivatives of the generating function are evaluated on complex mixtures of 4-QAM and 8-PSK sources and compared to the 6-BIOME algorithm by means of simulation results.

1. INTRODUCTION

Blind Identification (BI) methods have been successfully applied in various scientific areas, including for instance telecommunications [1], acoustic [2] or biomedical signal processing [3]. A large family of BI methods relies on the theory of Independent Component Analysis (ICA) [4] and thereby involves second or higher-order statistics. BI of underdetermined mixtures (when the number of sources exceeds the number of sensors) is an important subcategory of BI problems which arises in many practical situations, especially in telecommunications.

Several solutions have been proposed in the literature to solve this problem (see, e.g. [5, 6, 7, 8, 9]). Notably, some original methods, which do not exploit cumulants but the second characteristic function of the observations, have been proposed in [10, 11, 12, 13] . We are interested here by the approach proposed in [12], leading to a class of efficient algorithms such as the ALESCAF algorithm [13]. In that work, the authors showed that partial derivatives of the second ChAracteristic Function (CAF) can be stored in a symmetric tensor. The Canonical Decomposition (CanD) of this tensor provides a direct estimation of the mixing matrix up to trivial scaling and permutation indeterminacies. The ALESCAF algorithm resorts to an Alternating LEast Squares procedure in order to perform the CanD.

The CAF approach has a nice advantage, which makes it very attractive for the identification of underdetermined mixtures. Indeed, for a given number of sensors, the number of sources is theoretically not limited.

In [13], ALESCAF has been successfully applied on under-determined mixtures of real sources such as BPSK or 4-PAM. It can be shown easily that the method holds for complex mixtures of real sources and that ALESCAF can be applied to the case of real mixtures of complex sources within few modifications. However, these are very specific cases, which are rarely encountered in practice. On the other hand, as far as we know, this approach has not been extended to the case of complex mixtures of complex sources although this scenario [1] is far more relevant from a practical point of view. Most cumulant based algorithms can be directly applied in both situations whereas it rapidly turns out that the ALESCAF algorithm is not pertinent in the complex case. As a consequence, the present work aims at extending the CAF approach to the complex case, which often occurs in practice.

In this paper, we firstly transpose the theory of the CAF approach to the complex case: a new core equation is obtained. By differentiating this core equation, we obtain a new tensor decomposition from which an estimation of the mixing matrix can be obtained. In order to implement this more general approach a new algorithm is proposed. The CAF approach is available for most applications involving BI. Computer results obtained from simulated telecommunications signals are presented in the last part of the paper as an application example.

2. THEORY

2.1 Notations

Vectors, matrices and tensors are denoted by lower case boldface ($\mathbf{a}$), upper case boldface ($\mathbf{A}$) and upper case calligraphic ($\mathscr{A}$) letters respectively. a_i is the i^{th} coordinate of vector $\mathbf{a}$ and $\mathbf{a}_i$ is the i^{th} column of matrix $\mathbf{A}$. The (i,j) entry of matrix $\mathbf{A}$ is denoted A_{ij} and the (i,j,k) entry of the third order tensor $\mathscr{A}$ is denoted $\mathscr{A}_{ijk}$. Complex objects are underlined, their real and imaginary parts are denoted $\Re\{\cdot\}$ and $\Im\{\cdot\}$ respectively. E[.] denotes the mean value of a random variable.

2.2 Blind Identification Problem

Consider a noisy mixture of K statistically independent narrowband sources received by an array of N sensors. The vector $\underline{\mathbf{y}}(m)$ containing discrete observations of the received signal at the sensor outputs is modelled according to the following linear model:

$$\underline{\mathbf{y}}(m) = \underline{\mathbf{H}}\underline{\mathbf{s}}(m) + \underline{\mathbf{n}}(m),\ m = 1 \cdots M$$

[1] Note that in the following we refer to the real or complex case when both mixture and sources belong to $\mathbb{R}$ ("real case") or $\mathbb{C}$ ("complex case").

 ISSN 2076-1465

where $\underline{\mathbf{H}} = [\underline{\mathbf{h}}_1, \ldots, \underline{\mathbf{h}}_K] \in \mathbb{C}^{N\times K}$, $\underline{\mathbf{s}} = [\underline{s}_1, \ldots, \underline{s}_K]^T \in \mathbb{C}^K$ and $\underline{\mathbf{n}} \in \mathbb{C}^N$ are the mixing matrix, source and Gaussian noise random vectors, respectively. It is assumed that for any fixed sample index m, $\underline{\mathbf{s}}$ and $\underline{\mathbf{n}}$ are statistically independent. The problem is to identify the mixing matrix $\underline{\mathbf{H}}$ (up to trivial column permutation and scaling) from the only knowledge of the observation vector $\underline{\mathbf{y}}(m)$, using its characteristic function. Recall that we are interested in the so-called underdetermined case, which means that we have $K > N$.

Before to proceed, we describe our working hypotheses:

H1. The mixing matrix $\underline{\mathbf{H}}$ does not contain collinear columns.

H2. The sources $\underline{s}_1(m), \ldots, \underline{s}_K(m)$ are mutually independent and non-Gaussian

H3. The number of sources is known.

The theoretical justification of the present approach is similar to that of the real case. It consists in successively differentiating the second generating function[2] of the observations at different points of the observation space. By working with complex mixtures of complex sources, this leads to a new core equation following a particular tensor decomposition. By exploiting the structure of this tensor decomposition, the mixing matrix is estimated.

2.3 The new core equation

The first step is to obtain the new core equation. This is achieved by decomposing the second generating function of the observations as a sum of the individual second generating functions of the sources. Generating functions of a complex variable are actually defined by assimilating $\mathbb{C}$ to $\mathbb{R}^2$. Thus the second generating function of the k^{th} source φ_k taken at the point $\underline{z}$ of $\mathbb{C}$ is defined as a function of two real variables (real and imaginary parts of $\underline{z}$):

$$\varphi_k(\Re\{\underline{z}\}, \Im\{\underline{z}\}) \stackrel{\text{def}}{=} \log \mathrm{E}[\exp(\Re\{\underline{s}_k\}\Re\{\underline{z}\} + \Im\{\underline{s}_k\}\Im\{\underline{z}\})]$$

In a more compact form we have:

$$\varphi_k(\Re\{\underline{z}\}, \Im\{\underline{z}\}) = \log \mathrm{E}[\exp(\Re\{\underline{z}^* \underline{s}_k\})] \qquad (2.1)$$

In the same way, the second generating function of the observations Φ_y taken at the point $\underline{\mathbf{w}}$ of $\mathbb{C}^{2N}$ is actually defined in $\mathbb{R}^{2N}$ by

$$\Phi_y(\Re\{\underline{\mathbf{w}}\}, \Im\{\underline{\mathbf{w}}\}) \stackrel{\text{def}}{=} \log \mathrm{E}[\exp(\Re\{\underline{\mathbf{y}}\}^{\mathrm{H}}\Re\{\underline{\mathbf{w}}\} + \Im\{\underline{\mathbf{y}}\}^{\mathrm{H}}\Im\{\underline{\mathbf{w}}\})]$$

thus we have

$$\Phi_y(\Re\{\underline{\mathbf{w}}\}, \Im\{\underline{\mathbf{w}}\}) = \log \mathrm{E}[\exp(\Re\{\underline{\mathbf{w}}^{\mathrm{H}}\underline{\mathbf{y}}\})]$$

Replacing $\underline{\mathbf{y}}$ by its model yields:

$$\Phi_y(\Re\{\underline{\mathbf{w}}\}, \Im\{\underline{\mathbf{w}}\}) = \log \mathrm{E}[\exp(\Re\{\underline{\mathbf{w}}^{\mathrm{H}}\underline{\mathbf{H}}\underline{\mathbf{s}}\})]$$

and from the sources mutual statistical independence hypothesis we can deduce:

$$\Phi_y(\Re\{\underline{\mathbf{w}}\}, \Im\{\underline{\mathbf{w}}\}) = \sum_k \log \mathrm{E}[\exp(\Re\{\underline{\mathbf{w}}^{\mathrm{H}}\underline{\mathbf{h}}_k\underline{s}_k\})]$$

[2] In order to simplify notations and calculations, without any theoretical impact, we prefer using the generating function instead of the characteristic function.

where $\underline{\mathbf{h}}_k$ is the k^{th} column of matrix $\underline{\mathbf{H}}$. Then, (2.1) yields:

$$\Phi_y(\Re\{\underline{\mathbf{w}}\}, \Im\{\underline{\mathbf{w}}\}) = \sum_k \varphi_k\left(\Re\{\underline{\mathbf{w}}^{\mathrm{T}}\underline{\mathbf{h}}_k^*\}, \Im\{\underline{\mathbf{w}}^{\mathrm{T}}\underline{\mathbf{h}}_k^*\}\right)$$

Finally, we define two real matrices $\mathbf{A}$ and $\mathbf{B}$ so that $\underline{\mathbf{H}} = \mathbf{A} + j\mathbf{B}$. This leads to the the new core equation that copes with the complex case:

$$\Phi_y(\Re\{\underline{\mathbf{w}}\}, \Im\{\underline{\mathbf{w}}\}) = \sum_k \varphi_k\left(\sum_n A_{nk}\Re\{\underline{w}_n\} + B_{nk}\Im\{\underline{w}_n\}, \sum_n A_{nk}\Im\{\underline{w}_n\} - B_{nk}\Re\{\underline{w}_n\}\right) \qquad (2.2)$$

Note that defining φ_k, Φ_y in $\mathbb{R}^{2N}$ and $\mathbb{R}^2$ respectively instead of $\mathbb{C}^N$ and $\mathbb{C}^2$ allows their differentiation. Hence, the next step is the differentiation of (2.2).

2.4 Differentiation of $\Phi_y(\Re\{\underline{\mathbf{w}}\}, \Im\{\underline{\mathbf{w}}\})$

We define $\mathbf{u} = \Re\{\underline{\mathbf{w}}\}$, $\mathbf{v} = \Im\{\underline{\mathbf{w}}\}$ and $\mathbf{w} = (\mathbf{u}, \mathbf{v})$. $\mathbf{w}$ is an element of $\mathbb{R}^{2N}$ and (2.2) can be rewritten as:

$$\Phi_y(\mathbf{w}) = \sum_k \varphi_k\left(\sum_n A_{nk}u_n + B_{nk}v_n, \sum_n A_{nk}v_n - B_{nk}u_n\right) \qquad (2.3)$$

We also introduce three functions g_1, g_2 and g respectively defined by:

$$g_1(\mathbf{w}) = \sum_n A_{nk}u_n + B_{nk}v_n \; ; \; g_2(\mathbf{w}) = \sum_n A_{nk}v_n - B_{nk}u_n$$

$$\begin{array}{rccc} g: & \mathbb{R}^{2N} & \longrightarrow & \mathbb{R}^2 \\ & \mathbf{w} & \longmapsto & g(\mathbf{w}) = (g_1(\mathbf{w}), g_2(\mathbf{w})) \end{array}$$

The φ_k functions map $\mathbb{R}^2$ to $\mathbb{R}$ and we have:

$$\Phi_y(\mathbf{w}) = \sum_k \varphi_k(g(\mathbf{w}))$$

Let us compute the partial derivatives of $\Phi_y(\mathbf{w})$ with respect to the real $(u_n, n = 1\cdots N)$ and imaginary parts $(v_n, n = 1\cdots N)$ of $\underline{\mathbf{w}}$ coordinates. The differentiations of (2.3) with respect to u_p and v_p, $p = 1\cdots N$ yield:

$$\frac{\partial \Phi_y(\mathbf{w})}{\partial u_p} = \sum_{k=1}^K \frac{\partial \varphi_k(g)}{\partial g_1}A_{pk} - \frac{\partial \varphi_k(g)}{\partial g_2}B_{pk} \qquad (2.4)$$

$$\frac{\partial \Phi_y(\mathbf{w})}{\partial v_p} = \sum_{k=1}^K \frac{\partial \varphi_k(g)}{\partial g_1}B_{pk} + \frac{\partial \varphi_k(g)}{\partial g_2}A_{pk} \qquad (2.5)$$

In order to have a sufficient diversity of equations we have to use higher differentiating orders. In the theoretical part of this study, we limit ourselves to the second order. The associated equations at higher orders can be obtained in a similar manner. Hence, we can differentiate (2.4) and (2.5) with respect to u_q and v_q, $q = 1\cdots N$. For instance,

$$\frac{\partial^2 \Phi_y(\mathbf{w})}{\partial u_p \partial u_q} = \frac{\partial}{\partial g_1}\sum_{k=1}^K \frac{\partial \varphi_k(g)}{\partial u_q}A_{pk} - \frac{\partial}{\partial g_2}\sum_{k=1}^K \frac{\partial \varphi_k(g)}{\partial u_q}B_{pk} \qquad (2.6)$$

Substituting (2.4) and (2.5) in (2.6) yields:

$$\begin{aligned}\frac{\partial^2\Phi_y(\mathbf{w})}{\partial u_p\partial u_q} &= \sum_{k=1}^{K} A_{pk}\left[\frac{\partial^2\varphi_k(g)}{\partial g_1\partial g_1}A_{qk}-\frac{\partial^2\varphi_k(g)}{\partial g_1\partial g_2}B_{qk}\right]\\ &- \sum_{k=1}^{K} B_{pk}\left[\frac{\partial^2\varphi_k(g)}{\partial g_2\partial g_1}A_{qk}-\frac{\partial^2\varphi_k(g)}{\partial g_2\partial g_2}B_{qk}\right]\end{aligned}$$

2.5 Tensor stowage and decomposition

In practice, the partial derivatives of Φ_y are computed at S points of $\mathbb{R}^{2N}$ denoted $\mathbf{w}^{(s)}$. The objective is again to increase the order of the tensor, aiming at achieving a better estimation quality. Let us define $G_{sk}^{ij}=\frac{\partial^2\varphi_k(g(\mathbf{w}^{(s)}))}{\partial g_i(\mathbf{w}^{(s)})\partial g_j(\mathbf{w}^{(s)})}$ $i=1,2$; $j=1,2$ (one can note that $G_{sk}^{12}=G_{sk}^{21}$). This leads to the three distinct relations:

$$\begin{aligned}\frac{\partial^2\Phi_y(\mathbf{w}^{(s)})}{\partial u_p\partial u_q} &= \sum_{k=1}^{K}A_{pk}A_{qk}G_{sk}^{11}-\sum_{k=1}^{K}A_{pk}B_{qk}G_{sk}^{12}-\\ &\sum_{k=1}^{K}B_{pk}A_{qk}G_{sk}^{12}+\sum_{k=1}^{K}B_{pk}B_{qk}G_{sk}^{22}\quad(2.7)\end{aligned}$$

$$\begin{aligned}\frac{\partial^2\Phi_y(\mathbf{w}^{(s)})}{\partial v_p\partial v_q} &= \sum_{k=1}^{K}B_{pk}B_{qk}G_{sk}^{11}+\sum_{k=1}^{K}B_{pk}A_{qk}G_{sk}^{12}+\\ &\sum_{k=1}^{K}A_{pk}B_{qk}G_{sk}^{12}+\sum_{k=1}^{K}A_{pk}A_{qk}G_{sk}^{22}\quad(2.8)\end{aligned}$$

$$\begin{aligned}\frac{\partial^2\Phi_y(\mathbf{w}^{(s)})}{\partial u_p\partial v_q} &= \sum_{k=1}^{K}A_{pk}B_{qk}G_{sk}^{11}+\sum_{k=1}^{K}A_{pk}A_{qk}G_{sk}^{12}-\\ &\sum_{k=1}^{K}B_{pk}B_{qk}G_{sk}^{12}-\sum_{k=1}^{K}B_{pk}A_{qk}G_{sk}^{22}\quad(2.9)\end{aligned}$$

Since all values of p and q are taken into consideration, equations (2.7)-(2.9) cover all the partial second order derivatives. In the real case, the second order derivatives of Φ_y are stored in a third order tensor whose CanD gives a direct estimation of the mixing matrix. This situation is quite different in the complex case. Indeed, each of the three previous equations can be seen as a sum of four CanD of third-order tensors (p,q,s), involving the elements of the mixing matrix in different ways. It appears that the CanD of these tensors or of any combination of those is insufficient here. Therefore CanD based algorithms such as ALESCAF are not pertinent in this case. However it is still possible to use a tensor approach by jointly exploiting the three forms of derivatives in order to build a fourth-order tensor $(N,N,S,3)$ with increased diversity, noted $\mathscr{T}^{\Phi}$. The last mode of $\mathscr{T}^{\Phi}$ contains the following elements:

$$\begin{aligned}&\mathscr{T}^{\Phi}_{pqs1}\overset{\text{def}}{=}\frac{\partial^2\Phi_y(\mathbf{w}^{(s)})}{\partial u_p\partial u_q}\quad;\quad\mathscr{T}^{\Phi}_{pqs2}\overset{\text{def}}{=}\frac{\partial^2\Phi_y(\mathbf{w}^{(s)})}{\partial v_p\partial v_q}\\ &\mathscr{T}^{\Phi}_{pqs3}\overset{\text{def}}{=}\frac{\partial^2\Phi_y(\mathbf{w}^{(s)})}{\partial u_p\partial v_q}\end{aligned}\quad(2.10)$$

3. ALGORITHM

3.1 Building $\mathscr{T}^{\Phi}$

We explain in this section how to build $\mathscr{T}^{\Phi}$ from the realizations of $\underline{\mathbf{y}}$.

The entries of $\mathscr{T}^{\Phi}$ are computed one by one just like in the real case. We call Γ_y the first generating function of $\underline{\mathbf{y}}$ defined by:

$$\Gamma_y(\mathbf{w}^{(s)})\overset{\text{def}}{=}\mathrm{E}[\exp(\mathbf{u}^{(s)\mathsf{T}}\Re\{\underline{\mathbf{y}}\}+\mathbf{v}^{(s)\mathsf{T}}\Im\{\underline{\mathbf{y}}\})]\quad(3.11)$$

so that $\Phi_y=\log\Gamma_y$. In practice, the expected value is estimated by the mean value on all the realisations. Note that this estimator is consistent but it leads to a biased estimation of the partial derivatives of Φ_y, if the latter are computed by finite differences of (3.11). As in [13], it is preferred to compute formal derivatives, and estimate the obtained expressions with the help of sample means.

Let us define $D(\mathbf{w}^{(s)})$ as the partial derivatives of $\Gamma_y(\mathbf{w}^{(s)})$ with respect to the components of $\mathbf{u}^{(s)}$ and $\mathbf{v}^{(s)}$. Examples of first and second order derivatives are:

$$D_p^u(\mathbf{w}^{(s)})\overset{\text{def}}{=}\frac{\partial\Gamma_y(\mathbf{w}^{(s)})}{\partial u_p}=\Re\{\underline{y}_p\}\Gamma_y(\mathbf{w}^{(s)})$$

$$D_{pq}^{vu}(\mathbf{w}^{(s)})\overset{\text{def}}{=}\frac{\partial^2\Gamma_y(\mathbf{w}^{(s)})}{\partial v_p\partial u_q}=\Im\{\underline{y}_p\}\Re\{\underline{y}_q\}\Gamma_y(\mathbf{w}^{(s)})$$

Thus, the elements of $\mathscr{T}^{\Phi}$ (i.e. second order derivatives) are given by:

$$\begin{aligned}\mathscr{T}^{\Phi}_{pqs1} &= \frac{D_{pq}^{uu}(\mathbf{w}^{(s)})}{\Gamma_y(\mathbf{w}^{(s)})}-\frac{D_p^u(\mathbf{w}^{(s)})D_q^u(\mathbf{w}^{(s)})}{\Gamma_y^2(\mathbf{w}^{(s)})}\\ \mathscr{T}^{\Phi}_{pqs2} &= \frac{D_{pq}^{vv}(\mathbf{w}^{(s)})}{\Gamma_y(\mathbf{w}^{(s)})}-\frac{D_p^v(\mathbf{w}^{(s)})D_q^v(\mathbf{w}^{(s)})}{\Gamma_y^2(\mathbf{w}^{(s)})}\\ \mathscr{T}^{\Phi}_{pqs3} &= \frac{D_{pq}^{uv}(\mathbf{w}^{(s)})}{\Gamma_y(\mathbf{w}^{(s)})}-\frac{D_p^u(\mathbf{w}^{(s)})D_q^v(\mathbf{w}^{(s)})}{\Gamma_y^2(\mathbf{w}^{(s)})}\end{aligned}\quad(3.12)$$

3.2 Estimation of the mixing matrix

The proposed algorithm is named LEMACAFC-O, where O is the order of differentiation. Hence, LEMACAFC-2 consists of iteratively fitting the tensor $\widehat{\mathscr{T}^{\Phi}}$ built from the estimated parameters and model equations (2.7)-(2.9) to $\mathscr{T}^{\Phi}$ using the Levenberg-Marquardt (LM) method. The LM method has been used to perform the CanD of multi-way arrays in [14, 15] for example.

We consider the minimization of the following quadratic cost function:

$$\mathrm{f}_{\mathscr{T}}(\mathbf{p})=\frac{1}{2}\|\mathbf{e}(\mathbf{p})\|_F^2=\frac{1}{2}\mathbf{e}^{\mathsf{H}}(\mathbf{p})\mathbf{e}(\mathbf{p})$$

where $\mathbf{e}(\mathbf{p})=\mathrm{vec}\{\widehat{\mathscr{T}^{\Phi}}(\mathbf{p})-\mathscr{T}^{\Phi}\}\in\mathbb{C}^{3SN^2\times1}$ is the residue and $\mathbf{p}$ is the parameter vector defined as:

$$\mathbf{p}=\begin{bmatrix}\mathbf{p}_{\widehat{\mathbf{A}}}\\ \mathbf{p}_{\widehat{\mathbf{B}}}\\ \mathbf{p}_{\widehat{\mathbf{G}}^{11}}\\ \mathbf{p}_{\widehat{\mathbf{G}}^{12}}\\ \mathbf{p}_{\widehat{\mathbf{G}}^{22}}\end{bmatrix}=\begin{bmatrix}\mathrm{vec}(\widehat{\mathbf{A}}^{\mathsf{T}})\\ \mathrm{vec}(\widehat{\mathbf{B}}^{\mathsf{T}})\\ \mathrm{vec}(\widehat{\mathbf{G}}^{11\mathsf{T}})\\ \mathrm{vec}(\widehat{\mathbf{G}}^{12\mathsf{T}})\\ \mathrm{vec}(\widehat{\mathbf{G}}^{22\mathsf{T}})\end{bmatrix}\in\mathbb{C}^{(2N+S+3)K\times1}$$

Ns	Simulation parameters				Median values (10^{-2}) and number of acceptable values (%) of f_H											
	Mod.	K	M	SNR	LEMC-2		LEMC-2		LEMC-2		LEMC-3		LEMC-4		6-BIOME	
					Med.	Na	Med.	Na	Med.	Na	Med.	Na	Med.	Na	Med.	Na
1	4-QAM	4	10000	20	13	16	×	×	×	×	0.21	90	0.26	90	0.43	76
2	4-QAM	4	5000	50	×	×	×	×	4.4	18	0.34	78	0.42	74	0.6	74
3	4-QAM	4	5000	20	24	4	4.5	18	2.9	18	0.45	76	0.5	76	0.58	68
4	4-QAM	5	10000	20	NC	0	×	×	×	×	1.5	40	1.2	40	1.2	46
5	4-QAM	5	5000	30	NC	0	×	×	×	×	1.7	34	1.5	32	1.9	16
6	4-QAM	6	20000	20	×	×	×	×	NC	0	2.5	26	1.6	40	1.9	28
7	4-QAM	5	5000	20	NC	0	×	×	×	×	6.8	12	3.2	14	4.2	6
8	8-PSK	4	10000	20	×	×	×	×	1.6	28	0.38	82	0.34	90	NC	0

Table 1: **Some comparisons between LEMACAFC algorithms (denoted LEMC here) and 6-BIOME**. Ns is the simulation number. NC means that the corresponding algorithm has never converged; × means that it has not been evaluated in this situation.

and where vec$\{\cdot\}$ yields a column vector by stacking the columns of its matrix argument. The LM update is given as follows:

$$\mathbf{p}(i+1) = \mathbf{p}(i) - \left[\mathbf{J}^H(i)\mathbf{J}(i) + \lambda(i)\mathbf{I}\right]^{-1}\mathbf{g}(i)$$

where $\mathbf{J}(i)$ denotes the Jacobian matrix, $\mathbf{g}(i)$ the gradient vector computed at iteration i and $\lambda(i)$ is a positive regularization parameter. $\mathbf{p}$ and λ are updated at every iteration. There are many ways to proceed with the LM updates. We retained the scheme described in [16]. After convergence, an estimate of the mixture is obtained by $\widehat{\underline{\mathbf{H}}} = \text{unvec}\{\mathbf{p}_{\widehat{\mathbf{A}}} + j\mathbf{p}_{\widehat{\mathbf{B}}}\}$ (up to column permutation and scaling).

4. COMPUTER RESULTS

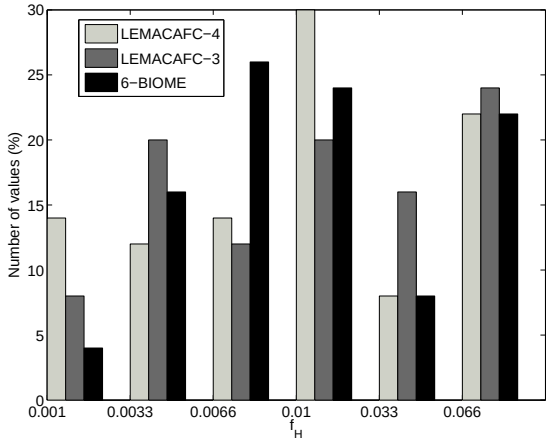

Figure 1: Distribution (in %) of the relative estimation error (f_H) computed from simulation 4 results.

The performance of the proposed approach is evaluated for blind identification of underdetermined complex mixtures of 4-QAM or 8-PSK sources. Our goal here is to highlight the performance of LEMACAFC through a limited number of key simulations.

Three versions of the algorithm (LEMACAFC-2, LEMACAFC-3, LEMACAFC-4) have been implemented and compared in various situations to the well known 6-BIOME (Blind Identification of Overcomplete MixturEs) algorithm [7], also referred to as "BIRTH" (Blind Identification of mixtures of sources using Redundancies in the daTa Hexacovariance matrix).
Algorithms were evaluated with respect to the estimation error, according to the following normalized measure:

$$f_H(\underline{\mathbf{H}}, \widehat{\underline{\mathbf{H}}}) = \frac{vec(\underline{\mathbf{H}} - \widehat{\underline{\mathbf{H}}})^{\mathrm{H}} vec(\underline{\mathbf{H}} - \widehat{\underline{\mathbf{H}}})}{vec(\underline{\mathbf{H}})^{\mathrm{H}} vec(\underline{\mathbf{H}})}$$

For each situation a median value (Med.) of f_H and a number of "acceptable results" (Na) is obtained from 50 Monte-Carlo runs. We chose to define Na as the percentage of Monte-Carlo results for which $f_H < 10^{-2}$. These values are reported in Table 1 according to simulation parameters and algorithms. Figure 1 focuses on simultion 4 results, giving all LEMACAFC-3, LEMACAFC4 and 6-BIOME results in an histogram form. At each run, the source vectors and the mixing matrix were changed and the derivatives were computed at 10 different points ($S = 10$) whose real and imaginary parts were randomly drawn in the range $[-1;1]^N$. Our iterative algorithms were all initialized with the same random entries and consistently stopped after 60 iterations. Simulation parameters are source modulation (Mod.), source number (K), sample number (M) and Signal to Noise Ratio (SNR). The number of sensors is fixed to 3 for each simulations.
Three variations of LEMACAFC-2 were actually computed with an eye to evaluate the impact of an "unlucky" initialization. The first one uses only one random initialization while for the second and third ones we compared five and ten different initializations respectively and we kept the initialization corresponding to the smallest value of $f_{\mathscr{J}}(\mathbf{p})$ after the 60 iterations. Only one random initialization was used for LEMACAFC-3 and LEMACAFC-4. Simulation 3 results show that increasing the number of random initialization appreciably improves our two performance criteria.
Our simulations on 4-QAM sources can be rank in several categories. First of all, LEMACAFC-2 only converge in the most favourable situations (simulations 1,2,3). Actually, LEMACAFC-2 seems to be not suitable when the number of sources exceeds 4. Our simulations are ordered in the as-

cending order of LEMACAFC-3 median values. Hence it clearly appears that for simulations 1 to 3 (*i.e.*: the easiest cases) LEMACAFC-3 provides slightly better results than LEMACAFC-4 as opposed to simulations 4 and 5 (middle cases). Finally LEMACAFC-4 is sensibly better for simulations 6 and 7 (difficult cases), indicating that the latter is still interesting in some difficult situations, notably when the underdeterminacy level (*i.e.*: the ratio between source number and sensor number) is high.

Taking into account both criteria, LEMACAFC-3 provides better or comparable results than 6-BIOME in most situations (simulations 1,2,3,5,7) while LEMACAFC-4 is consistently better than the cumulant based approach at the exeption of simulation 4, for which 6-BIOME provides 46 % of acceptable values against 40 % for LEMACAFC-3 and LEMACAFC-4. However the histogram plotted in figure 1 shows that when converging, LEMACAFC provides a better estimation of the mixing matrix. For instance, 14 % of LEMACAFC-4 error values are smaller than 0.0033 against 8 % for LEMACAFC-3 and only 4 % for 6-BIOME. In this sense, figure 1 is typical because a similar observation could have been done from every simulation histogram. Furthermore the number of LEMACAFC-3 and LEMACAFC-4 acceptable values could be increase by trying several random initialization entries.

Finally simulation 8 shows that in this "easy" case all LEMACAFC algorithms provide satisfactory results with 8-PSK sources as opposed to 6-BIOME.

5. CONCLUSION

We have addressed the problem of blind identification of underdetermined complex mixtures of complex sources using the second generating function of the observations. We detailed the theoretical background and proposed an algorithm relying on an original tensor decomposition. Finally, three versions of this algorithm, based on several differentiation order, have been evaluated on simulated complex mixtures of telecommunications complex sources.

It has been shown that second order version provides some satisfying results in the least difficult cases, especially if several initialization entries are compared. In these conditions, it can be an option if one is looking for a fast algorithm. On the contrary, the fourth order version appears as a possible solution for the most complicated cases. In this connection, a deeper investigation should clarify the respective influences of SNR, underdeterminacy level and sample number. In other cases we recommend the use of LEMACAFC-3 which is enough to overpass a classical 6 order cumulant based approach in most situations while being less time consuming than LEMACAFC-4.

Moreover, the LEMACAFC algorithm also worked fine in the case of 8-PSK sources.

REFERENCES

[1] C. ESTEVAO and R. FERNANDES and G. FAVIER and J. C. MOTA, "Blind channel identification algorithms based on the PARAFAC decomposition of cumulant tensors: the single and multiuser cases," *Signal Processing*, 88, 6, pp. 1382–1401, 2008.

[2] F. ASANO and S. IKEDA and M. OGAWA and H. ASOH and N. KITAWAKI , "Combined approach of array processing and independent component analysis for blind separation of acoustic signals," *IEEE Transactions On Speech and Audio Processing*, 11, 3, pp. 204–215, 2003.

[3] A. KACHENOURA and L. ALBERA and L. SENHADJI and P. COMON , "ICA: a potential tool for BCI systems," *IEEE Signal Processing Magazine, special issue on Brain-Computer Interfaces*, 25, 3, pp. 57–68, 2008.

[4] P. COMON and C. JUTTEN, *Handbook of Blind Source Separation, Independent Component Analysis and Applications*. Academic Press, 2010.

[5] J. F. CARDOSO, "Super-symmetric decomposition of the fourth-order cumulant tensor. Blind identification of more sources than sensors," in *Proc. ICASSP'91*, Toronto, 1991, pp. 3109–3112.

[6] P. COMON, "Blind identification and source separation in 2x3 under-determined mixtures," *IEEE Trans. Signal Process.*, pp. 11–22, Jan. 2004.

[7] L. ALBERA, A. FERREOL, P. COMON, and P. CHEVALIER, "Blind identification of overcomplete mixtures of sources (BIOME)," *Lin. Algebra Appl.*, vol. 391, pp. 1–30, Nov. 2004.

[8] L. DE LATHAUWER, J. CASTAING, and J-. F. CARDOSO, "Fourth-order cumulant-based blind identification of underdetermined mixtures," *IEEE Trans. Signal Process.*, vol. 55, no. 2, pp. 2965–2973, Feb. 2007.

[9] L. DE LATHAUWER and J. CASTAING, "Blind identification of underdetermined mixtures by simultaneous matrix diagonalization," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1096–1105, Mar. 2008.

[10] A. YEREDOR, "Blind Source Separation via the Second Characteristic Function, " *Signal Processing*, vol. 80, no. 5 pp.897–902, May 2000.

[11] A. TALEB, "An algorithm for the blind identification of *n* idependent signals with 2 sensors," *ISSPA'01*, Kuala Lumpur, 2001, vol 1, pp. 5–8.

[12] P. COMON and M. RAJIH, "Blind identification of complex underdetermined mixtures," *ICA Conference*, Granada, 2004, pp. 105–112.

[13] P. COMON and M. RAJIH, "Blind identification of under-determined mixtures based on the characteristic function," *Signal Processing*, vol. 86, no. 9, pp. 2271–2281, Sept. 2006.

[14] G. TOMASI and R. BRO, "A comparison of algorithms for fitting the parafac model," *Comp. Stat. Data Anal.*, vol. 50, pp. 1700–1734, 2006.

[15] A.L.F. DE ALMEIDA, X. LUCIANI, P. COMON, "Blind identification of underdetermined mixtures based on the hexacovariance and higher-order cyclostationarity," *Proc. SSP'09*, Cardiff, 2009, pp. 669–672.

[16] K. MADSEN and H. B. NIELSEN and O. TINGLEFF, "Methods for Non-Linear Least Squares Problems," *Technical University of Denmark, Informatics and mathematical Modelling*, 2