EUSIPCO 2013 1569741959

FAST JACOBI LIKE ALGORITHMS FOR JOINT DIAGONALIZATION OF COMPLEX SYMMETRIC MATRICES

Victor MAURANDI [1,2], *Christophe DE LUIGI* [1,2] *and Eric MOREAU* [1,2]

[1] Aix Marseille Université, CNRS, ENSAM, LSIS, UMR 7296
13397 Marseille, France.
[2] Université de Toulon, CNRS, LSIS, UMR 7296
83957 La Garde, France.
Email : {maurandi, deluigi, moreau}@univ-tln.fr

ABSTRACT

In this paper, we consider the problem of non-orthogonal joint diagonalization of a set of complex symmetric matrices. This appears in many signal processing problems, especially in source separation. We propose three new algorithms based on LU decomposition of the matrix we are looking for and based on a Jacobi like procedure. The algorithms are based on a coupled and a decoupled parameter estimation. Numerical simulations are provided to compare the performances with a similar one existing in the real case and adapted to the complex symmetric case. Finally, we propose a mixed algorithm combining two proposed ones which allows more balanced performances.

Index Terms**—** Non-Orthogonal Joint Diagonalization, Source Separation, Complex Symmetric Matrix

1. INTRODUCTION

Joint Diagonalization of sets of matrices is an important issue in blind signal processing and more particularly in source separation [1]-[12]. The unitary case has been first considered, see [1, 2] for hermitian matrices and [3] for complex symmetric matrices. Nowadays the non-unitary case has become important mainly because it allows to skip a first processing step (whitening of observations in source separation) which limits the performances in practical cases. Recently a number of algorithms have been proposed [4]-[12]. Some of them are based on a direct criterion or based on an inverse criterion or even on a combined one. We consider the inverse criterion.

The complex symmetric matrix case is of importance due to the non-circularity property that may appear in some wireless telecommunication applications [13]. For the joint diagonalization problem of complex symmetric matrices, we consider a Jacobi like scheme for the minimization of the inverse criterion. Our main purpose is to use an LU decomposition on the searched separation matrix in order to avoid the critical problem of degenerate solutions as e.g. the estimation of non invertible matrix.

The paper is organized as follows. First, we introduce the joint diagonalization problem. Then we develop the Jacobi like algorithms for non-orthogonal joint diagonalization of complex symmetric matrices based on a LU decomposition. We propose an improvement by a combination of two of these algorithms. Finally, by numerical simulations, we compare the performances of each algorithm.

The authors thank the *Direction Générale de l'Armement* for its financial support.

2. PROBLEM FORMULATION

We consider K ($K \geq 2$) complex symmetric matrices, $\mathbf{M}_k$, $k = 1, \dots, K$, defined as:

$$\mathbf{M}_k = \mathbf{A}\mathbf{D}_k\mathbf{A}^T, \quad (1)$$

where $(\cdot)^T$ is the transposed operator. Throughout the paper, we only consider the case of square matrices, all of them of size $N \times N$. $\mathbf{D}_k$, $k = 1, \dots, K$, are diagonal complex matrices and $\mathbf{A}$ is complex invertible. Here $\mathbf{A}$ is the so-called mixing matrix.

From the matrix set $\{\mathbf{M}_k\}$ the objective is to estimate a complex separation matrix $\mathbf{B}$ (equal to the inverse of the mixing matrix $\mathbf{A}$ up to the product of a diagonal matrix and a permutation matrix) such that the matrices $\mathbf{B}\mathbf{M}_k\mathbf{B}^T$ are (approximately) jointly diagonal. The cost function used to jointly diagonalize the $\mathbf{M}_k$ matrices is the inverse criterion which is defined as:

$$J(\mathbf{B}) = \sum_{k=1}^{K} ||\mathbf{B}\mathbf{M}_k\mathbf{B}^T - \mathsf{Diag}\{\mathbf{B}\mathbf{M}_k\mathbf{B}^T\}||_F^2, \quad (2)$$

where $\mathsf{Diag}\{\cdot\}$ is the diagonal matrix defined by the diagonal of $\mathbf{M}$ and where $|| \cdot ||_F$ is the Frobenius norm. In order to optimize this cost function we have to find the matrix $\mathbf{B}$ minimizing $J(\mathbf{B})$ but avoiding the obvious solution $\mathbf{B} = \mathbf{0}$. For this purpose, we propose three Jacobi like algorithms based on the LU decomposition.

3. PROPOSED ALGORITHMS

All square matrices can be decomposed as $\mathbf{DPLU}$, where $\mathbf{D}$ is a diagonal matrix, $\mathbf{P}$ is a permutation matrix and $\mathbf{L}$ and $\mathbf{U}$ are, respectively lower and upper triangular matrices with diagonal coefficients equal to 1. In the source separation problem, matrices $\mathbf{D}$ and $\mathbf{P}$ corresponds to classical indeterminacies and can thus be dropped. It then remains to determine both triangular matrices $\mathbf{L}$ and $\mathbf{U}$ giving now the separating matrix as $\mathbf{B} = \mathbf{LU}$. A great advantage of this parameterization is that $\mathbf{B}$ will be an invertible matrix.

We consider three iterative processes for the estimation of matrices $\mathbf{L}$ and $\mathbf{U}$. In order to simplify this estimation, we consider a Jacobi like procedure. This procedure consists of decomposing a problem of size $N \times N$ as a product of $\frac{N(N-1)}{2}$ sub-problems of "size" 2×2 considering all couples with the same indexes of rows and columns. For example, in case $N = 3$, this can be written as:

$$\mathbf{B} = \begin{pmatrix} B_{11}^1 & B_{12}^1 & 0 \\ B_{21}^1 & B_{22}^1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} B_{11}^2 & 0 & B_{13}^2 \\ 0 & 1 & 0 \\ B_{31}^2 & 0 & B_{33}^2 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & B_{22}^3 & B_{23}^3 \\ 0 & B_{32}^3 & B_{33}^3 \end{pmatrix},$$

where B_{ij}^q with $(i,j) \in \{1,\dots,N\}^2$ and $q = 1,\dots,\frac{N(N-1)}{2}$, are complex coefficients.

Hence, from now, we only focus on the 2×2 case. Each sub-matrices 2×2 will be written as the product of the following 2×2 $\mathbf{L}$ and $\mathbf{U}$ matrices

$$\mathbf{L} = \begin{pmatrix} 1 & 0 \\ l & 1 \end{pmatrix} \qquad \mathbf{U} = \begin{pmatrix} 1 & u \\ 0 & 1 \end{pmatrix} \tag{3}$$

where l and u are the two unknown complex parameters. We develop two iterative ways for the estimation of $\mathbf{L}$ and $\mathbf{U}$:

- The first way consists of estimating $\mathbf{L}$ and $\mathbf{U}$ all together, *i.e.* considering there is a unique matrix $\mathbf{LU}$. This is the so-called coupled method.
- The second way consists of estimating $\mathbf{L}$ and $\mathbf{U}$ one after the other, *i.e.* fixing one of the two matrices in order to compute the other one and vice versa. This is the so-called decoupled method.

We can highlight that the Jacobi scheme minimizes successively the $\frac{N(N-1)}{2}$ restricted 2×2 cases. Hence, the proposed scheme does not minimize the full criterion J directly. Computer simulations in section 4 will show the effectiveness of the proposed approach.

3.1. The coupled estimation

3.1.1. Separation phase

In this first phase, the separation matrix $\mathbf{B}$, in the 2×2 case, is defined as

$$\mathbf{B} = \begin{pmatrix} 1 & 0 \\ l & 1 \end{pmatrix} \begin{pmatrix} 1 & u \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & u \\ l & 1 + lu \end{pmatrix}. \tag{4}$$

Let $\mathbf{M}'_k$, $k = 1,\dots,K$, the matrices of size 2×2 defined as

$$\mathbf{M}'_k = \mathbf{B}\mathbf{M}_k\mathbf{B}^T. \tag{5}$$

By construction of the $\mathbf{M}_k$ matrices in (1), the $\mathbf{M}'_k$ matrices are symmetric too. Thus $M'_{k,21} = M'_{k,12}$. Hence, the cost function in (2), in the 2×2 case, named $J_{2\times 2}(\mathbf{B})$ can be rewritten as:

$$J_{2\times 2}(\mathbf{B}) = 2 \sum_{k=1}^{K} |M'_{k,12}|^2. \tag{6}$$

In using (5), we have $M'_{k,12} = \mathbf{c}_k^T \mathbf{p}$ with

$$\mathbf{c}_k = \begin{pmatrix} M_{k,11} \\ M_{k,12} \\ M_{k,22} \end{pmatrix} \text{ and } \mathbf{p} = \begin{pmatrix} l \\ 1 + 2lu \\ u(1 + lu) \end{pmatrix}. \tag{7}$$

The cost function can be now written as the quadratic form:

$$J_{2\times 2}(\mathbf{B}) = 2\, \mathbf{p}^H \sum_{k=1}^{K} \mathbf{c}_k^* \mathbf{c}_k^T \mathbf{p}, \tag{8}$$

where $(\cdot)^*$ is conjugate operator. Thus, the minimization of $J_{2\times 2}(\mathbf{B})$ can be solved by finding the unit-norm minor eigenvector, noted $\mathbf{e}$, of $\sum_{k=1}^{K} \mathbf{c}_k^* \mathbf{c}_k^T$. Now we have to find the parameters of $\mathbf{p}$ in such a way that

$$\mathbf{e} = \begin{pmatrix} e_1 & e_2 & e_3 \end{pmatrix}^T = \alpha \mathbf{p}, \tag{9}$$

where α is a supplementary unknown complex parameter. The parameters l, u and α are calculated by resolving the following non linear system of three equations derived from (9)

$$\begin{cases} 0 = e_1 u^2 - e_2 u + e_3 \\ \alpha = e_2 - 2ue_1 \\ l = \dfrac{e_1}{\alpha}. \end{cases} \tag{10}$$

In this system, as the first equation is of the second order, there are two solutions for u. We propose to choose the one with the smallest magnitude. Indeed the cost function depends, *inter alia*, on the magnitude of u, and so the smallest the magnitude is more the cost function is minimized. Hence, this choice yields a best possible minimization of $J_{2\times 2}(\mathbf{B})$.

3.1.2. Balancing phase

In order to improve the robustness of this iterative method based on a minor eigenvector research, we have to pay attention that the separation matrix norm will not become too high. That is why we propose to normalize $\mathbf{B}$ in using a postmultiplication by a diagonal weighted matrix defined as:

$$\mathbf{W} = \begin{pmatrix} w & 0 \\ 0 & w^{-1} \end{pmatrix}, \tag{11}$$

where w is a complex parameter. This parameter is computed by minimizing the norm of $\mathbf{WB}$. Straightforward derivation yields to:

$$w = \frac{(|l|^2 + |1 + lu|^2)^{1/4}}{(1 + |u|^2)^{1/4}}. \tag{12}$$

The resulting coupled LU algorithm named CLU is built with the alternate scheme. This alternate strategy involves, for each global sweep, to apply for each pair the first phase and then second phase.

3.2. The decoupled estimation

In this second way, the separation matrix $\mathbf{B}$ is first imposed equal to $\mathbf{U}$. Then its optimal value is derived as above. After $\mathbf{B}$ is imposed equal to $\mathbf{L}$ and its optimal value is also derived. The main point is that it is not done all together, but successively. The main difference with [9] is that we restrict to the 2×2 case for the successive estimation of l and u, so we do not minimize the global criterion but the 2×2 corresponding "sub" criterion.

3.2.1. The $\mathbf{U}$ part

By considering $\mathbf{B} = \mathbf{U} = \begin{pmatrix} 1 & u \\ 0 & 1 \end{pmatrix}$, in (5) then the criterion in (6), in the 2×2 case, named $J_{2\times 2}$ reads

$$J_{2\times 2}(\mathbf{U}) = \sum_{k=1}^{K} |M_{k,12} + uM_{k,22}|^2. \tag{13}$$

The optimal value for u minimizing $J_{2\times 2}(\mathbf{U})$ follows as

$$u = -\frac{\sum_{k=1}^{K} M_{k,12} M_{k,22}^*}{\sum_{k=1}^{K} |M_{k,22}|^2}. \tag{14}$$

3.2.2. The $\mathbf{L}$ part

By considering $\mathbf{B} = \mathbf{L} = \begin{pmatrix} 1 & 0 \\ l & 1 \end{pmatrix}$, in (5) then the criterion $J_{2\times 2}$ reads

$$J_{2\times 2}(\mathbf{L}) = \sum_{k=1}^{K} |M_{k,12} + lM_{k,11}|^2. \tag{15}$$

The optimal value for l minimizing $J_{2\times 2}(\mathbf{L})$ follows as

$$l = -\frac{\sum_{k=1}^{K} M_{k,12} M_{k,11}^*}{\sum_{k=1}^{K} |M_{k,11}|^2}. \tag{16}$$

3.2.3. Two different strategies

This decoupled procedure can be built in two global optimization strategies. One way consists of applying the $\mathbf{U}$ part first for all pairs and after the $\mathbf{L}$ part for all pairs. This is the so-called block strategy with the following acronym DLU1. The second way is to do the same scheme as in the coupled procedure, *i.e.* for each fixed pairs, the 2×2 $\mathbf{U}$ part first and the 2×2 $\mathbf{L}$ part are derived and applied. This is the so-called alternate strategy with the following acronym DLU2.

3.3. Mixed algorithm

As it will be shown in the computer simulation section 4.1, on the first iterations (*i.e.* the sweeps of all pairs) DLU1 convergence rate outperforms CLU one and inversely after few iterations. So we propose a third algorithm that mixes both previous algorithms in order to have the behaviour of DLU1 on the first iterations and then the CLU one. Let $n_0 \in \mathbb{N}^*$, this mixed LU (MLU) algorithm consists in using DLU1 algorithm for the first n_0 iterations and then using CLU algorithm. In the simulation results section we will test this algorithm for different values of n_0.

We can notice that all these proposed algorithms have the same numerical complexity because of the algebraic calculus of the parameters in the 2×2 case. Moreover, we can highlight that the algorithms DLU1 and DLU2 are an improvement of the Afsari's algorithm [9] and CLU algorithm is an original one using LU parametrisation.

4. SIMULATION RESULTS

4.1. Algorithm comparisons

In order to evaluate the algorithm performances, we use the performance index proposed in [2][11][12]. It compares the global matrix $\mathbf{S} = \mathbf{BA} = (S_{ij})$ to the product of a permutation matrix and a diagonal matrix as follow:

$$I(\mathbf{S}) = \frac{1}{2N(N-1)} \left(\sum_{i=1}^{N} \left(\sum_{j=1}^{N} \frac{|S_{ij}|^2}{\max_\ell |S_{i\ell}|^2} - 1 \right) + \sum_{j=1}^{N} \left(\sum_{i=1}^{N} \frac{|S_{ij}|^2}{\max_\ell |S_{\ell j}|^2} - 1 \right) \right). \tag{17}$$

This non-negative index is zero if $\mathbf{S}$ satisfies $\mathbf{B} = \mathbf{DPA}^{-1}$. We consider 25 complex symmetric matrices of size 5×5 in the two following cases:

- without noise
- with additive noise (which corresponds to a model error on the decomposition of the observation matrices) of the form $\mathbf{M}_k + t\ \mathbf{N}_k$ where $\mathbf{N}_k$, $k = 1, \ldots, 25$, are complex symmetric matrices of size $N \times N$ and $t = 10^{-4}$.

The components of $\mathbf{M}_k$, the diagonal components of $\mathbf{D}_k$ and the components of $\mathbf{A}$, follow a zero mean unit variance normal distribution for both their real part and their imaginary

part. We do the same for the components of $\mathbf{N}_k$.
Finally, we display the median value of the global criterion $J(\mathbf{B})$ (cf fig. 1) and the median value of the performance index $I(\mathbf{S})$ (cf fig. 2 - fig.5) *w.r.t.* the number of iteration over one hundred independent draws.

First, in figure 1, we can see that the algorithms CLU, DLU1 and DLU2 converge for the global criterion (2) althought they are built on the minimization of the corresponding successive 2×2 problems. Moreover they have a better accuracy than the algorithm LUJ1D.

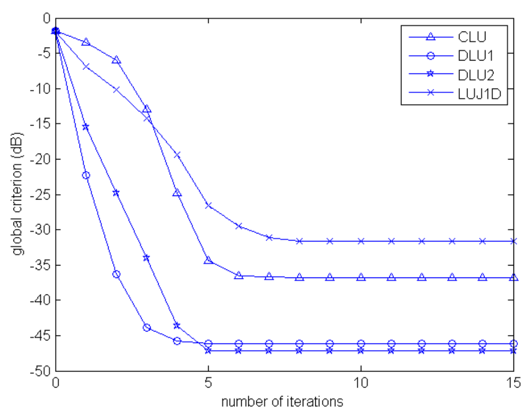

Fig. 1. Algorithms convergence for the global criterion for 25 matrices of size 5×5 in the noisy context ($t = 10^{-4}$).

Now, we compare the results given by the proposed algorithms CLU, DLU1, DLU2 in comparison to the result given by a similar algorithm, denoted LUJ1D and proposed by Afsari in the real case [9]. Notice that it is adapted here to the complex symmetric case.

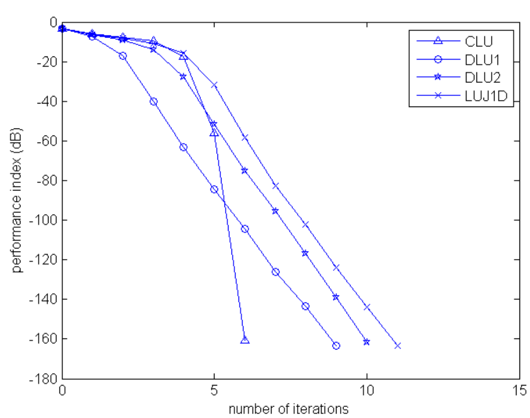

Fig. 2. Algorithm comparisons for 25 matrices of size 5×5 in the noiseless context.

In the noiseless case, see figure 2, we verify that all algorithms joint diagonalize all matrices $\mathbf{M}_k$ perfectly. All proposed algorithms outperform the LUJ1D w.r.t. the convergence speed. Moreover, until the fourth iteration DLU1 outperforms the other algorithms and after CLU becomes better than the others. Concerning DLU2 algorithm, it gives some results similar but less efficient to DLU1.

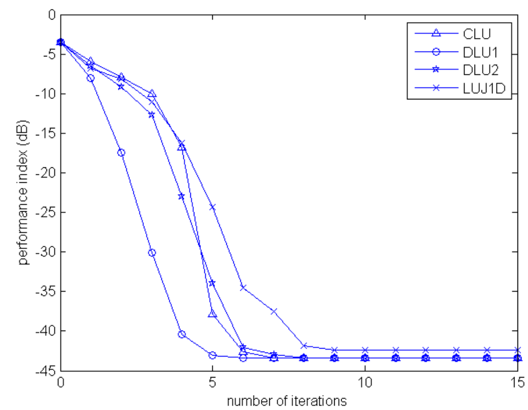

Fig. 3. Algorithm comparisons for 25 matrices of size 5×5 in the noisy context ($t = 10^{-4}$).

The case with noise (see figure 3) confirms that LUJ1D is really less efficient than the others. Indeed the performance index of CLU, DLU1 and DLU2 all converge to $-43\ dB$ whereas LUJ1D converges around $-42\ dB$. Concerning the convergence speed, DLU1 is always more efficient during the first four iterations compared to DLU2 and CLU. But from the fifth iteration CLU becomes better than the others.

4.2. Combination of DLU1 and CLU algorithms

As we have seen previously DLU1 and CLU present a good convergence speed but with a rather different behavior. In the first iterations DLU1 is better than CLU and after it is the contrary. That is why it is certainly interesting to propose a basic combination of both algorithms. Using exactly the same scenarios studied in paragraph 4.1, we test MLU algorithm with different values of n_0 (defined in paragraph 3.3) and we compare it with CLU and DLU1. We point out that MLU algorithm for $n_0 = 2$ and $n_0 = 4$ gives similar results to those for $n_0 = 3$ (the results are not plotted in figure 4 and figure 5).

For $n_0 = 1$, in the noiseless (figure 4) and the noisy contexts (figure 5), MLU has a better convergence speed than CLU and DLU1.

We can notice that for $n_0 = 3$, in the noisy case, the performance index increases when MLU goes from the DLU1 to the CLU. This behavior is not acceptable and it never has been noticed for $n_0 = 1$. That is why we can retain MLU algorithm with $n_0 = 1$ as possibly the best combination of CLU and DLU1 algorithms.

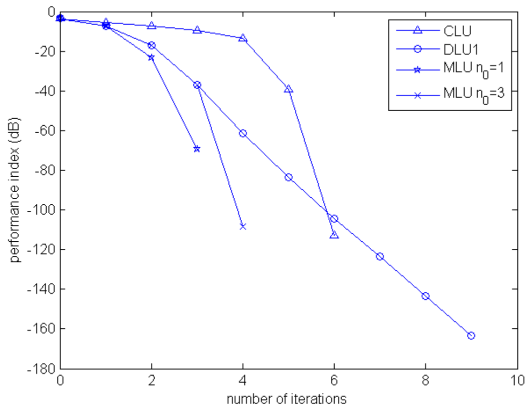

Fig. 4. Performance of MLU algorithm versus n_0 parameter in the noiseless context.

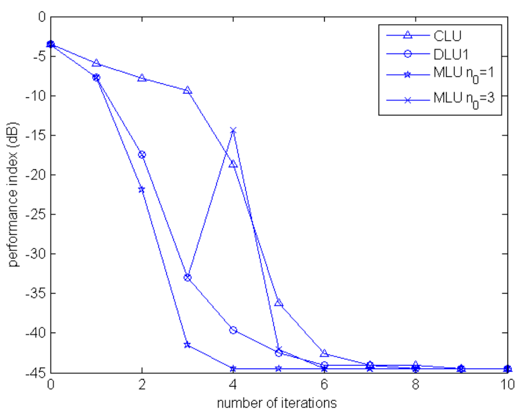

Fig. 5. Performance of MLU algorithm versus n_0 parameter in the noisy context ($t = 10^{-4}$).

5. CONCLUSION

For the non-orthogonal joint diagonalization of complex symmetric matrices, we have proposed three new algorithms based on LU decomposition of the separation matrix and based on an Jacobi like procedure. Thus, all of them rely on the analytic resolution of the elementary 2×2 problems. Computer simulations have illustrated the good behavior of proposed algorithms DLU1 and CLU with additive noise or without. Finally, thanks to these results, we have proposed a mixed algorithm (MLU) that takes advantages of both previous algorithms.

6. REFERENCES

[1] J.-F. Cardoso and A. Souloumiac, “Blind beamforming for non Gaussian signals”, *IEE Proceedings-F*, Vol. 40, pp 362-370, 1993.

[2] E. Moreau, “A generalization of joint-diagonalization criteria for source separation”, *IEEE Trans. Signal Processing*, Vol. 49, No. 3, pp 530-541, March 2001.

[3] L. De Lathauwer, B. De Moor and J. Vandewalle, “ICA techniques for more sources than sensors”, in *Proc. of the IEEE Workshop on Higher-Order Statistics (HOS’99)*, Caesarea, Israel, pp. 121-124, June 1999.

[4] A. Yeredor, “Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation”, *IEEE Trans. Signal Processing*, Vol. 50, No. 7, pp 1545-1553, July 2002.

[5] H. Ghennioui, E.-M. Fadaili, N. Thirion-Moreau, A. Adib and E. Moreau, “A non-unitary joint block diagonalization algorithm for blind separation of convolutive mixtures of sources,” *IEEE Signal Process. Letters*, Vol. 14, No. 11, pp. 860-863, Nov. 2007.

[6] X.-L. Li and X.-D. Zhang, “Nonorthogonal joint diagonalization free of degenerate solution,” *IEEE Trans. on Sig. Pro.*, Vol. 55, No. 5, May 2007

[7] X. Guo, S. Zhu, S. Miron and D. Brie, “Approximate joint diagonalization by nonorthogonal nonparametric Jacobi transformations,” in *Proc. ICASSP’2010*, Dallas, USA, March 2010.

[8] A. Souloumiac, “Nonorthogonal joint diagonalization by combining Givens and hyperbolic rotations,” *IEEE Trans. Signal Processing*, vol. 57, no. 6, pp. 2222-2231, June 2009.

[9] B. Afsari, “Simple LU and QR based non-orthogonal matrix joint diagonalization”, in *Proc. ICA’2006, Springer LNCS series*, pp 1-7, 2006.

[10] T. Trainini and E. Moreau, “Variations around gradient like algorithms for joint diagonalization of Hermitian matrices,” in *Proc. EUSIPCO’2012*, Bucharest, Romania, August 2012.

[11] E. Moreau and O. Macchi, “A one stage self-adaptive algorithm for source separation”, in Proc. ICASSP, april 1994, vol. 3, pp. 4952.

[12] E. Moreau and O. Macchi, “High order contrasts for self-adaptive source separation”, International Journal of Adaptive Control and Signal Processing, vol. 10, no. 1, pp. 1946, January 1996.

[13] B. Picinbono and P. Bondon, “Second-order statistics of complex signals”, *IEEE Transactions on Signal Processing*, vol. 45, no. 2, pp. 411419, 1997.