A RELATIVE GRADIENT ALGORITHM FOR JOINT DECOMPOSITIONS OF COMPLEX MATRICES

Tual Trainini(∗), *Xi-Lin Li*(+), *Eric Moreau*(∗), *and Tülay Adalı*(+)

(∗) Université du Sud Toulon Var
LSEET, UMR CNRS 6017
Av. G. Pompidou, BP 56, 83162
La Valette du Var, Cédex, France
Fax: (+33) 4 94 14 26 71 {trainini,moreau@univ-tln.fr}

(+) University of Maryland Baltimore County
MLSP Laboratory
Department of CSEE
Baltimore, MD 21250, USA
{lixilin,adali@umbc.edu}

ABSTRACT

The problem of joint decomposition of sets of complex matrices arises in many problems in signal processing. In this paper, we address the problem for the general case where the matrices can be Hermitian and/or complex symmetric. As such, complete statistical information in the complex domain can be taken into account for the given signal processing problem. The proposed algorithm is based on an optimal step size relative gradient approach and computer simulations are provided to illustrate the behavior of this algorithm in different contexts and to establish a comparison with other algorithms.

1. INTRODUCTION

Joint decomposition of sets of complex matrices provides an important tool for a number of signal processing problems, see *e.g.*, [2–5, 9, 11, 13]. In [2], joint diagonalization of a particular cumulant matrix has been used for the blind beamforming problem, which has led to the popular joint approximate diagonalization of eigenmatrices (JADE) algorithm. Note that this algorithm is also useful for source separation and array processing. A generalization of the criterion to any order cumulant can be found in [9]. In this approach, the searched joint diagonalizer happens to be a unitary matrix. Thus, in practice, first a projection (often called prewhitening) stage is required. However, this first stage induces a bound on the attainable performance and this is why, recently, joint diagonalization algorithms for the "non unitary" case have been suggested, see *e.g.* [3, 5, 11, 12].

Besides, when noncircular complex signals are considered [10] , one can exploit additional matrix decompositions, see, *e.g.*, [8,11,13]. Noncircular signals can arise in many applications such as communications, radar, and medical imaging [1]. For example, in order to account for second-order noncircularity, both the covariance matrix, which is Hermitian, and the pseudo-covariance matrix, which is complex symmetric, need to be taken into account.

The goal of this paper is to provide an algorithm for joint decompositions of complex matrices for this most general case, *i.e.*, including the presence of Hermitian as well as complex symmetric matrices. Hence, in the application in question, we can expect improved performance by taking the full statistical information in the complex domain into account. We can also expect an increase in robustness due to a further use of the available diversity in signal statistics. Moreover, the diagonalization we propose is introduced through a common framework in order to not to significantly increase the computational cost. This is achieved by considering a classical joint quadratic criterion. The proposed optimization algorithm is an optimal step-size gradient algorithm with a multiplicative update.

The paper is organized as follows: the general problem of joint matrix decompositions of sets of Hermitian and/or symmetric complex matrices is stated in Section 2. In Section 3, the suggested approach based on a(n) (optimal step size) relative gradient approach is detailed. Computer simulations are provided to illustrate the good performance of the suggested method and to compare it with other "state-of-the-art" approaches. The purpose of Section 4 is to enhance the usefulness of the suggested algorithm. Finally, in Section 5, conclusions are drawn.

2. JOINT MATRIX DECOMPOSITIONS

2.1 Problem statement

The problem that we consider is stated as follows. We consider two sets $\mathscr{M}_j$ for $j = 1,2$ of N_j square matrices $\mathbf{M}_i^{(j)} \in \mathbb{C}^{M\times M}$, for all $i \in \{1,\ldots,N_j\}$. The N_1 matrices in $\mathscr{M}_1$ all admit the following decomposition:

$$\mathbf{M}_i^{(1)} = \mathbf{A}\mathbf{D}_i^{(1)}\mathbf{A}^H , \qquad (1)$$

while the N_2 matrices in $\mathscr{M}_2$ all admit the following decomposition:

$$\mathbf{M}_i^{(2)} = \mathbf{A}\mathbf{D}_i^{(2)}\mathbf{A}^T , \qquad (2)$$

where $(\cdot)^H$ stands for the transpose conjugate operator and $(\cdot)^T$ for the transpose operator. The matrices $\mathbf{D}_i^{(j)}$, for $j = 1,2$, for all $i = 1,\ldots,N_j$, are complex diagonal matrices. We further assume that $\mathbf{A}$ is full column rank and belongs to $\mathbb{C}^{M\times N}$ with $M \geq N$ (**Assumption** $\mathbf{A}_0$). For all $j = 1,2$, the set of the N_j square matrices $\mathbf{D}_i^{(j)} \in \mathbb{C}^{N\times N}$ is denoted $\mathscr{D}_j$. The general joint matrix decompositions problem that we consider consists of estimating the matrix $\mathbf{A}$ and the two diagonal matrices sets $\mathscr{D}_1$ and $\mathscr{D}_2$ from only the matrix sets $\mathscr{M}_1$ and $\mathscr{M}_2$. We finally remark that when only the first of these two sets is considered (see (1)), the treated problem simplifies into a well-known joint-diagonalization problem [2, 3, 7]–[12].

2.2 An optimization problem

In what follows, the pseudo-inverse (Moore-Penrose generalized matrix inverse) $\mathbf{A}^+$ of $\mathbf{A}$ is denoted by $\mathbf{B}$. Due to the matrix factorization, a rather classical way to solve the given problem minimizes a positive cost function. Thus, we suggest to use:

$$\mathscr{J}(\mathbf{B}) = \alpha \underbrace{\sum_{i=1}^{N_1} \|\mathsf{offDiag}(\mathbf{B}\mathbf{M}_i^{(1)}\mathbf{B}^H)\|_F^2}_{\mathscr{C}(\mathbf{B})} + (1-\alpha)\underbrace{\sum_{i=1}^{N_2} \|\mathsf{offDiag}(\mathbf{B}\mathbf{M}_i^{(2)}\mathbf{B}^T)\|_F^2}_{\mathscr{D}(\mathbf{B})}, \quad (3)$$

where $\alpha \in [0,1]$, $\|\cdot\|_F$ stands for the Frobenius norm. $\mathsf{offDiag}\{\mathbf{M}\} = \mathbf{M} - \mathsf{Diag}\{\mathbf{M}\}$, *i.e.*, the $\mathsf{offDiag}\{\cdot\}$ operator sets to zero all the diagonal elements of the matrix in argument while the $\mathsf{Diag}\{\cdot\}$ operator sets to zero all the off-diagonal elements of the matrix in argument. When $\alpha = 1$, $\mathscr{J}(\mathbf{B}) = \mathscr{C}(\mathbf{B})$ which is the cost function used in [2, 3, 7]–[12]. The case $\alpha = 0.5$, with an additional third term $\log|\det(\mathbf{B})|$ ($\det(\cdot)$ is the determinant of a square matrix) was studied in [13]. Even if the purpose of this additional "constraint" term is to insure that $\mathbf{B}$ does not become singular, it has two drawbacks: first, it implies the search for a square matrix $\mathbf{B}$, and second, the $\log(\cdot)$ function is not bounded which may induce numerical problems too. Finally, in order to avoid the trivial zero matrix solution, a normalization has to be imposed. This will be done in constraining the norm of the searched matrix to be equal to one.

3. PROPOSED RELATIVE GRADIENT ALGORITHM

To estimate the matrix $\mathbf{B} \in \mathbb{C}^{N\times M}$, the cost function $\mathscr{J}(\mathbf{B})$ given in (3) has to be minimized. To that aim, we propose, as in [5], to use a relative gradient algorithm. The main interest of such an approach is that for small enough step sizes the invertibility of the matrix $\mathbf{B}$ can be guaranteed which is not the case with the standard gradient algorithm, see, *e.g.*, [12]. We also provide an alternative algorithm in which the step size is no more fixed but computed algebraically at each iteration.

3.1 Fixed step size relative gradient approach

We consider a relative gradient approach written as $\triangle\mathbf{B} = -\mu_r \nabla_r \mathscr{J}(\mathbf{B})\mathbf{B}$ since $\mathscr{J}(\mathbf{B})$ has to be minimized versus $\mathbf{B}$. μ_r is a positive small enough number called the step size or adaptation coefficient and $\nabla_r \mathscr{J}(\cdot)$ is defined as:

$$\nabla_r \mathscr{J}(\mathbf{B}) = 2\frac{\partial \mathscr{J}(\mathbf{B})}{\partial \mathbf{B}^*}\mathbf{B}^H = \nabla_a \mathscr{J}(\mathbf{B})\mathbf{B}^H, \quad (4)$$

where $\mathbf{B}^*$ stands for the complex conjugate of the complex matrix $\mathbf{B}$, ∂ is the partial derivative operator and $\nabla_a \mathscr{J}(\mathbf{B}) = 2\frac{\partial \mathscr{J}(\mathbf{B})}{\partial \mathbf{B}^*}$ is the complex gradient matrix of the real-valued scalar cost function given in (3). Subsequently, the suggested relative gradient-based algorithm can be derived, $\mathbf{B}$ is then updated at each iteration k (for all $k = 1, 2, \ldots$) according to the following scheme:

$$\begin{aligned}\mathbf{B}^{(k)} &= \mathbf{B}^{(k-1)} - \mu_r \nabla_r \mathscr{J}(\mathbf{B}^{(k-1)})\mathbf{B}^{(k-1)} \\ &= \left(\mathbf{I}_N - \mu_r \nabla_r \mathscr{J}(\mathbf{B}^{(k-1)})\right)\mathbf{B}^{(k-1)}.\end{aligned} \quad (5)$$

This updating relation is followed by a normalization of matrix $\mathbf{B}^{(k)}$ to unit norm at each iteration. In what follows, the resulting algorithm will be denoted by $\mathsf{JMD}_{\mathsf{RG}}$.
To be able to derive this algorithm, the complex gradient matrix $\nabla_a \mathscr{J}(\mathbf{B})$ has to be evaluated. Using (3), we have:

$$\nabla_a \mathscr{J}(\mathbf{B}) = \alpha \nabla_a \mathscr{C}(\mathbf{B}) + (1-\alpha)\nabla_a \mathscr{D}(\mathbf{B}). \quad (6)$$

In [7], it has been demonstrated that $\nabla_a \mathscr{C}(\mathbf{B})$ equals:

$$\begin{aligned}\nabla_a \mathscr{C}(\mathbf{B}) = 2\sum_{i=1}^{N_1} \Big[& \mathsf{offDiag}\{\mathbf{B}\mathbf{M}_i^{(1)}\mathbf{B}^H\}\mathbf{B}\mathbf{M}_i^{(1)H} \\ & + \left(\mathsf{offDiag}\{\mathbf{B}\mathbf{M}_i^{(1)}\mathbf{B}^H\}\right)^H \mathbf{B}\mathbf{M}_i^{(1)}\Big].\end{aligned} \quad (7)$$

Focusing now, on the second term $\nabla_a \mathscr{D}(\mathbf{B})$, it was shown [13] that it equals:

$$\nabla_a \mathscr{D}(\mathbf{B}) = 4\sum_{i=1}^{N_2}\left[\mathsf{offDiag}\{\mathbf{B}\mathbf{M}_i^{(2)}\mathbf{B}^T\}\mathbf{B}^*\mathbf{M}_i^{(2)*}\right]. \quad (8)$$

3.2 Seek of the optimal step size

To eliminate the difficult problem of the choice of the step size, while decreasing the total number of iterations N_i needed by the previous algorithm to reach convergence, it is possible to compute its optimal step size μ_{opt} at each iteration k which means that the algebraical calculation of the following quantity:

$$\begin{aligned}\mathscr{J}\left(\mathbf{B}^{(k)}\right) &= \alpha\mathscr{C}\left(\mathbf{B}^{(k)}\right) + (1-\alpha)\mathscr{D}\left(\mathbf{B}^{(k)}\right) \\ &= \mathscr{J}\left(\mathbf{B}^{(k-1)} - \mu\nabla_r \mathscr{J}(\mathbf{B}^{(k-1)})\mathbf{B}^{(k-1)}\right),\end{aligned} \quad (9)$$

has to be performed to be minimized with respect to μ. To simplify, we opt to omit the dependency with respect to the iteration k in what follows. The quantity defined in (9) is found to be a fourth order polynomial whose expression is given by:

$$\begin{aligned}&\mathscr{J}(\mathbf{B} - \mu\nabla_r\mathscr{J}(\mathbf{B})\mathbf{B}) \\ &= \alpha\mathscr{C}(\mathbf{B} - \mu\nabla_r\mathscr{J}(\mathbf{B})\mathbf{B}) + (1-\alpha)\mathscr{D}(\mathbf{B} - \mu\nabla_r\mathscr{J}(\mathbf{B})\mathbf{B}) \\ &= c_0 + \mu c_1 + \mu^2 c_2 + \mu^3 c_3 + \mu^4 c_4 \\ &= \alpha(a_0 + \mu a_1 + \mu^2 a_2 + \mu^3 a_3 + \mu^4 a_4) \\ &\quad + (1-\alpha)(b_0 + \mu b_1 + \mu^2 b_2 + \mu^3 b_3 + \mu^4 b_4),\end{aligned} \quad (10)$$

where the coefficients a_k, $\forall k = 0,\ldots,4$, b_k, $\forall k = 0,\ldots,4$ (and consequently c_k, $\forall k = 0,\ldots,4$), are defined below:

$$a_k = (-1)^k \textstyle\sum_{i=1}^{N_1} (\mathsf{vec}\{\mathbf{M}_i^{(1)}\})^H \mathbf{A_k} \mathsf{vec}\{\mathbf{M}_i^{(1)}\}, \quad (11)$$

and:

$$\begin{aligned}\mathbf{A_0} &= \mathbf{P}\mathbf{T_{off}}\mathbf{P}^H \\ \mathbf{A_1} &= \mathbf{P}\mathbf{T_{off}}\mathbf{Q}^H + \mathbf{Q}\mathbf{T_{off}}\mathbf{P}^H \\ \mathbf{A_2} &= \mathbf{P}\mathbf{T_{off}}\mathbf{R}^H + \mathbf{R}\mathbf{T_{off}}\mathbf{P}^H + \mathbf{Q}\mathbf{T_{off}}\mathbf{Q}^H \\ \mathbf{A_3} &= \mathbf{Q}\mathbf{T_{off}}\mathbf{R}^H + \mathbf{R}\mathbf{T_{off}}\mathbf{Q}^H \\ \mathbf{A_4} &= \mathbf{R}\mathbf{T_{off}}\mathbf{R}^H\end{aligned}$$

$$b_k = (-1)^k \sum_{j=1}^{N_2} (\mathsf{vec}\{\mathbf{M}_j^{(2)}\})^H \mathbf{B_k} \mathsf{vec}\{\mathbf{M}_j^{(2)}\}, \qquad (12)$$

and

$$\begin{aligned}
\mathbf{B_0} &= \mathbf{L}\mathbf{T_{off}}\mathbf{L}^H \\
\mathbf{B_1} &= \mathbf{L}\mathbf{T_{off}}\mathbf{M}^H + \mathbf{M}\mathbf{T_{off}}\mathbf{L}^H \\
\mathbf{B_2} &= \mathbf{L}\mathbf{T_{off}}\mathbf{N}^H + \mathbf{N}\mathbf{T_{off}}\mathbf{L}^H + \mathbf{M}\mathbf{T_{off}}\mathbf{M}^H \\
\mathbf{B_3} &= \mathbf{M}\mathbf{T_{off}}\mathbf{N}^H + \mathbf{N}\mathbf{T_{off}}\mathbf{M}^H \\
\mathbf{B_4} &= \mathbf{N}\mathbf{T_{off}}\mathbf{N}^H
\end{aligned}$$

where:

- $\mathbf{P} = \mathbf{B}^T \otimes \mathbf{B}^H$
- $\mathbf{Q} = \mathbf{B}^T \otimes (\nabla_r \mathscr{C}(\mathbf{B}))^H + (\nabla_r \mathscr{C}(\mathbf{B}))^T \otimes \mathbf{B}^H$
- $\mathbf{R} = (\nabla_r \mathscr{C}(\mathbf{B}))^T \otimes (\nabla_r \mathscr{C}(\mathbf{B}))^H$
- $\mathbf{L} = \mathbf{B}^H \otimes \mathbf{B}^H$
- $\mathbf{M} = \mathbf{B}^H \otimes (\nabla_r \mathscr{D}(\mathbf{B}))^H + (\nabla_r \mathscr{D}(\mathbf{B}))^H \otimes \mathbf{B}^H$
- $\mathbf{N} = (\nabla_r \mathscr{D}(\mathbf{B}))^H \otimes (\nabla_r \mathscr{D}(\mathbf{B}))^H$

$\otimes$ is the Kronecker product, $\mathsf{vec}\{\cdot\}$ includes all the columns of matrix given in argument into a column vector and $\mathbf{T_{off}}$ is a matrix defined as:

$$\mathbf{T_{off}} = \mathbf{I}_{N^2} - \mathsf{diag}\{\mathsf{vec}\{\mathbf{I}_N\}\} = \mathbf{T_{off}}^H, \qquad (13)$$

where $\mathbf{I}_{N^2}$ is the $N^2 \times N^2$ identity matrix, $\mathbf{I}_N$ is the $N \times N$ identity matrix matrix, $\mathsf{diag}\{\cdot\}$ is a square matrix containing the elements of the vector given in argument on its diagonal. The derivative of (10) with respect to μ leads to:

$$\frac{\partial \mathscr{J}(\mathbf{B} - \mu \nabla_r \mathscr{J}(\mathbf{B})\mathbf{B})}{\partial \mu} = 4c_4\mu^3 + 3c_3\mu^2 + 2c_2\mu + c_1. \qquad (14)$$

The optimal step size is obtained in two steps: after finding all the roots of (14), they are inserted into (10). The root that provides the minimum value is the optimal step size.

4. COMPUTER SIMULATIONS

In this section, computer simulations are presented to illustrate the performance and robustness of the proposed algorithm in comparison with the (adapted) one given in [8].

We consider an $M \times N$ complex matrix $\mathbf{A}$ built from a normal distribution with zero mean and unit variance. The diagonals of the $N \times N$ matrices $\mathbf{D}_i^{(1)}$ and $\mathbf{D}_i^{(2)}$ are generated with a complex Gaussian process with zero mean and variance σ_s^2. Now the matrices $\mathbf{M}_i^{(1)}$ and $\mathbf{M}_i^{(2)}$ are then built as defined in (1) and (2).

To test the robustness of the algorithm, different levels of additive noise is considered. Then, (1) and (2) have to be modified, taking into account perturbations:

$$\tilde{\mathbf{M}}_i^{(1)} = \mathbf{A}\mathbf{D}_i^{(1)}\mathbf{A}^H + \mathbf{B}_i^{(1)}, \qquad (15)$$

and

$$\tilde{\mathbf{M}}_i^{(2)} = \mathbf{A}\mathbf{D}_i^{(2)}\mathbf{A}^T + \mathbf{B}_i^{(2)}, \qquad (16)$$

where $\mathbf{B}_i^{(j)} \forall j = 1,2$ are matrices $\in \mathbb{C}^{M\times M}$ generated with a Gaussian process with zero mean and variance σ_b^2. The Signal-to-Noise Ratio (SNR) is then defined as: SNR $= 10\log_{10}(\frac{\sigma_s^2}{\sigma_b^2})$ (in dB).

We use the Performance Index defined in [4] as:

$$I(\mathbf{G}) = \frac{1}{r(r-1)} \sum_{i=1}^{r} \left(\sum_{j=1}^{r} \frac{\|\mathbf{G}_{i,j}\|_F^2}{\max_l \|\mathbf{G}_{i,l}\|_F^2} - 1 \right) + \frac{1}{r(r-1)} \sum_{j=1}^{r} \left(\sum_{i=1}^{r} \frac{\|\mathbf{G}_{i,j}\|_F^2}{\max_l \|\mathbf{G}_{l,j}\|_F^2} - 1 \right), \qquad (17)$$

where $\mathbf{G} = \mathbf{B}\mathbf{A}$ is the so-called global matrix. Next, the proposed algorithm $\mathsf{JMD_{RG}}$ is illustrated and compared to the extendedFAJD one which was introduced in [8] in the square case. In order to consider the rectangular case for this method, a dimension reduction by projection is first realized in order to come back to the square case. For the two algorithms, the initial guess is fixed in using the above projection that corresponds to the range space of matrix $\mathbf{A}$.

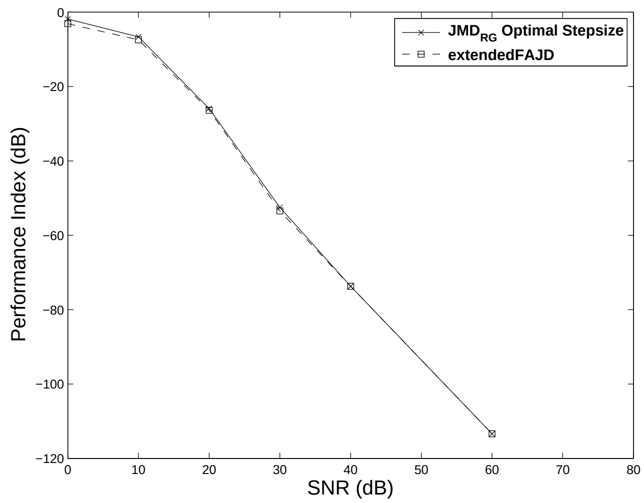

Figure 1: Influence of noise level on the Performance Index for square case

In Figure 1, the square case is considered with $M = N = 3$ and $N_1 = N_2 = 20$. The mean performance index over 100 Monte Carlo trials is plotted w.r.t. the SNR in the square case. We can see that the $\mathsf{JMD_{RG}}$ algorithm nearly reaches the same performances as extendedFAJD in the square case.

In Figure 2, the rectangular case is considered with $M = 5$, $N = 2$ and $N_1 = N_2 = 20$. 100 Monte Carlo trials are performed. We notice that for low SNR values (between 0 and 20 dB), algorithm extendedFAJD has better performances while for higher SNR it is the converse. This is certainly due to the projection stage which reduces the noise influence onto considered matrices. Remark for SNR greater than 40dB, $\mathsf{JMD_{RG}}$ the performance index becomes infinite implying that the sought matrix is perfectly estimated.

5. DISCUSSION

In this article, we present a new approach for joint decomposition of matrices. Using potentially different matrix decompositions, the approach allows the use of more statistical information in the complex domain. Relative gradient

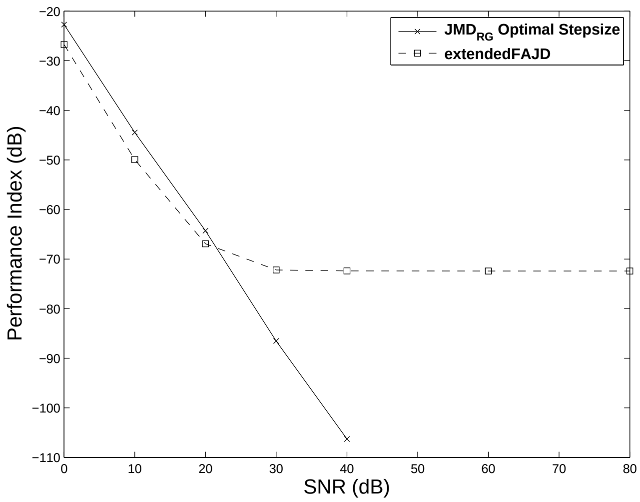

Figure 2: Influence of noise level on the Performance Index for rectangular case

search procedure helps improve the conditioning of the estimated matrix and a search procedure for the optimal step size (instead of fixing the step size) increases the overall convergence speed. Computer simulations highlights potential advantage of the proposed algorithm for high SNR in the rectangular case while preserving the performance gain in the square case.

REFERENCES

[1] T. Adalı and H. Li, “Complex-valued Signal Processing,” in Adaptive Signal Processing: Next Generation Solutions, Wiley Interscience, 2010 (T. Adalı and S, Haykin, editors).

[2] J.-F. Cardoso and A. Souloumiac, “Blind beamforming for non Gaussian signal,” *IEE Proc.-F*, Vol. 40, pp. 362–370, 1993.

[3] E.-M. Fadaili, N. Thirion-Moreau and E. Moreau, “Non orthogonal joint diagonalization/zero-diagonalization for source separation based on time-frequency distributions,” *IEEE Trans. Signal Process.*, Vol. 55, No. 5, pp. 1673–1687, May 2007.

[4] 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.

[5] H. Ghennioui, N. Thirion-Moreau, E. Moreau, D. Aboutajdine, “Gradient-based joint block diagonalization algorithms: application to blind separation of FIR convolutive mixtures,” *Signal Process. (2010)*, doi:10.1016/j.sigpro.2009.12.002

[6] A. Hjϕrungnes and D. Gesbert, “Complex-valued matrix differentiation: techniques and key results,” *IEEE Trans. Signal Process.*, Vol. 55, No. 6, pp. 2740–2746, June 2007.

[7] M. Joho and H. Mathis, “Joint diagonalization of correlation matrices by using gradient methods with application to blind signal separation,” in *Proc. IEEE Sensor Array and Multichannel Signal Processing Workshop SAM*, pp. 273–277, 2002.

[8] X.-L. Li and X.-D. Zhang, “Nonorthogonal joint diagonalization free of degenerate solution,” *IEEE Trans. Signal Process.*, Vol. 55, No. 5, pp. 1803–1814, 2007.

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

[10] B. Picinbono, “On circularity,” *IEEE Trans. Signal Process.*, Vol. 42, No. 12, pp 3473–3482, December 1994.

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

[12] A. Yeredor, A. Ziehe and K.-R. Müller, “Approximate joint diagonalization using a natural gradient approach,” in Lecture Notes in Computer Science (LNCS 3195), Springer - Verlag 2004, *5th International Conference on ICA*, Granada, Spain, pp. 89–96, September 2004.

[13] W.-J. Zeng, X.-L. Li, X.-D. Zhang and X. Jiang, “An improved signal-selective direction finding algorithm using second-order cyclic statistics,” in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 2141–2144, 2009