EUSIPCO 2013 1569744591

A NEW JACOBI-LIKE NONNEGATIVE JOINT DIAGONALIZATION BY CONGRUENCE

Lu Wang[1,2,4], *Laurent Albera*[1,2,4,5], *Hua Zhong Shu*[3,4], *Lotfi Senhadji*[1,2,4]

[1]INSERM, U 1099, 35042 Rennes Cedex, France
[2]LTSI, Université de Rennes 1, Campus de Beaulieu, 35042 Rennes Cedex, France
[3]LIST, Southeast University, 2 Sipailou, 210096, Nanjing, China
[4]Centre de Recherche en Information Biomédicale Sino-Français (CRIBs), Rennes, France
[5]INRIA, Centre Inria Rennes - Bretagne Atlantique, 35042 Rennes Cedex, France.
Email: wangLyu1986@hotmail.com, {laurent.albera, lotfi.senhadji}@univ-rennes1.fr, shu.list@seu.edu.cn

ABSTRACT

A new joint diagonalization by congruence algorithm is presented, which allows the computation of a nonnegative joint diagonalizer. The nonnegativity constraint is ensured by means of a square change of variable. Then we propose a Jacobi-like approach using LU matrix factorization, which consists of formulating a high-dimensional optimization problem into several sequential one-dimensional subproblems. Numerical experiments emphasize the advantages of the proposed method, especially in the presence of bottlenecks such as for low SNR values and a small number of available matrices. An illustration of blind source separation shows the interest of the proposed algorithm.

Index Terms— Nonnegative joint diagonalization by congruence, LU factorization, blind source separation, semi-nonnegative independent component analysis.

1. INTRODUCTION

Consider a set $\boldsymbol{\mathcal{C}} = \{\boldsymbol{C}^{(k)}\}_{k=1}^{K}$ of K real symmetric matrices sharing the following joint congruent structure:

$$\boldsymbol{C}^{(k)} = \boldsymbol{A}\boldsymbol{D}^{(k)}\boldsymbol{A}^{\mathsf{T}} \tag{1}$$

where $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ denotes an unknown transformation matrix, and $\boldsymbol{\mathcal{D}} = \{\boldsymbol{D}^{(k)}\}_{k=1}^{K}$ is a set of $(N \times N)$ unknown diagonal matrices. The Joint Diagonalization by Congruence (JDC) of such matrices consists of identifying the matrix $\boldsymbol{A}$ up to a diagonal matrix and a permutation matrix. JDC is a fundamental tool for Blind Source Separation (BSS) and Independent Component Analysis (ICA). In such problems, JDC identifies the mixing matrix $\boldsymbol{A}$ or its inverse from the observation vector which obeys the linear instantaneous mixing model. The K matrices $\boldsymbol{C}^{(k)}$ can be chosen as time-shifted covariance matrices, or higher order cumulant matrix slices.

The JDC problem has been mostly handled as a particular optimization problem. The algorithms computing $\boldsymbol{A}$ mainly depend on the criterion chosen to perform the optimization [1]. A great number of algorithms, such as JAD [2], FFDIAG [3], LUJ1D [4] and J-DI [5], compute $\boldsymbol{A}$ by minimizing the following indirect least square criterion:

$$J(\boldsymbol{A}) = \sum_{k=1}^{K} \| \operatorname{off}(\boldsymbol{A}^{-1}\boldsymbol{C}^{(k)}\boldsymbol{A}^{-\mathsf{T}}) \|_F^2 \tag{2}$$

where $\operatorname{off}(.)$ nullifies the diagonal parts of the input matrix, the superscript $^{-\mathsf{T}}$ means the inverse of the transposed matrix, and $\|.\|_F$ denotes the Frobenius norm. Other cost functions are also used. ACDC [6] and the sub-spaces fitting algorithm [7] estimate $\boldsymbol{A}$ and the set $\boldsymbol{\mathcal{D}}$ by using a direct least square criterion. Pham proposed an information theoretic criterion [8], which requires the target set $\boldsymbol{\mathcal{C}}$ to be positive definite.

Recently BSS involving nonnegativity constraint has shown an interest in numerous applications such as image processing, data mining and biomedical engineering (see chapter 13 of [9]). Some BSS applications only involve a nonnegative mixing matrix $\boldsymbol{A}$, such as magnetic resonance spectroscopy [10]. In this paper, we present a nonnegative JDC algorithm based on criterion (2), which is committed to seek a nonnegative joint diagonalizer $\boldsymbol{A}$. The nonnegativity constraint is imposed by means of a square change of variable. Then a Jacobi-like approach based on LU matrix factorization is presented, which formulates a high-dimensional optimization problem into several sequential one-dimensional subproblems. Numerical experiments emphasize the advantages of the proposed method, especially in the presence of bottlenecks such as for low Signal to Noise Ratio (SNR) values and a small number of matrices to be jointly diagonalized. An illustration of BSS of infrared spectra shows the interest of the proposed algorithm.

2. NONNEGATIVE JOINT DIAGONALIZATION

A way of imposing nonnegativity of any matrix belonging to $\mathbb{R}_{+}^{N\times N}$ is through a square change of variable:

$$\boldsymbol{A} = \boldsymbol{B} \boxdot \boldsymbol{B} \overset{\text{def}}{=} \boldsymbol{B}^{\boxdot 2} \tag{3}$$

with $\boldsymbol{B} \in \mathbb{R}^{N\times N}$, as originally proposed in [11] for Nonnegative Matrix Factorization (NMF), where $\boxdot$ stands for

Hadamard product. By assuming that $\boldsymbol{A}$ is non singular, the minimization of (2) such that $\boldsymbol{A}$ is nonnegative can be reformulated as an unconstrained problem by minimizing the following criterion:

$$J_+(\boldsymbol{B}) = \sum_{k=1}^{K} \| \operatorname{off}((\boldsymbol{B}^{\boxdot 2})^{-1}\boldsymbol{C}^{(k)}(\boldsymbol{B}^{\boxdot 2})^{-\intercal})\|_F^2 \qquad (4)$$

Minimizing (4) with respect to (w.r.t) $\boldsymbol{B}$ is the main goal of this paper. For this purpose, based on LU matrix factorization, the high dimensional optimization problem is then reduced to the search of a sequence of sparse triangular matrices. Hence we obtain a NonNegative extension of Afsari's LUJ1D algorithm [4], namely NNLUJ1D. Let's recall the following definition:

Definition 1 *A unit triangular matrix is a triangular matrix whose main diagonal elements are equal to* 1.

Then any non singular matrix $\boldsymbol{B} \in \mathbb{R}^{N\times N}$ can be factorized as $\boldsymbol{B} = \boldsymbol{L}\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{\Pi}$, where $\boldsymbol{L} \in \mathbb{R}^{N\times N}$ is a unit lower triangular matrix, $\boldsymbol{U} \in \mathbb{R}^{N\times N}$ is a unit upper triangular matrix, $\boldsymbol{\Lambda} \in \mathbb{R}^{N\times N}$ is a diagonal matrix and $\boldsymbol{\Pi} \in \mathbb{R}^{N\times N}$ is a permutation matrix. Consequently, due to the fact that $(\boldsymbol{L}\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{\Pi})^{\boxdot 2} = (\boldsymbol{L}\boldsymbol{U})^{\boxdot 2}\boldsymbol{\Lambda}^{\boxdot 2}\boldsymbol{\Pi}$ and the indeterminacies of the JDC problem, the matrix $\boldsymbol{B}$ solving (4) can be chosen as $\boldsymbol{B} = \boldsymbol{L}\boldsymbol{U}$ without loss of generality. Now, let's consider the following definition and lemma:

Definition 2 *An elementary triangular matrix $\boldsymbol{T}^{(i,j)}(a)$ is a unit triangular matrix whose non-diagonal elements are zeros except the (i,j)-th entry, which is equal to a.*

Lemma 1 *Any $(N\times N)$ unit lower (or upper) triangular matrix can be factorized as a product of $N(N-1)/2$ elementary lower (or upper, respectively) triangular matrices.*

The proof of lemma 1 is straightforward by reducing $\boldsymbol{L}$ (or $\boldsymbol{U}$) into identity matrices using Gaussian elimination. Lemma 1 yields that $\boldsymbol{B}$ can be written as a product of elementary triangular matrices:

$$\boldsymbol{B} = \prod_{j=1}^{N}\prod_{i=j+1}^{N} \boldsymbol{L}^{(i,j)}(\ell_{i,j}) \prod_{i=1}^{N}\prod_{j=i+1}^{N} \boldsymbol{U}^{(i,j)}(u_{i,j}) \stackrel{\text{def}}{=} \prod_{(i\neq j)}^{N(N-1)} \boldsymbol{T}^{(i,j)}(t_{i,j}) \qquad (5)$$

where $\boldsymbol{T}^{(i,j)}(t_{i,j})$ is defined for the sake of convenience as follows:

$$\boldsymbol{T}^{(i,j)}(t_{i,j}) = \begin{cases} \boldsymbol{L}^{(i,j)}(\ell_{i,j}), & \text{if } i > j \\ \boldsymbol{U}^{(i,j)}(u_{i,j}), & \text{if } i < j \end{cases} \qquad (6)$$

and $N(N-1)$ is the total number of elementary lower and upper triangular matrices. As a consequence, the minimization of (4) is reduced to the sequential search of the $N(N-1)$ parameters $t_{i,j}$. Indeed, instead of simultaneously identifying these $N(N-1)$ parameters, a Jacobi-like procedure will repeat several sweeps of $N(N-1)$ sequential optimizations until convergence. Each optimization w.r.t only one parameter $t_{i,j}$ with a selected (i,j) index. Let $\tilde{\boldsymbol{A}}$ and $\tilde{\boldsymbol{B}}$ denote the current estimate of $\boldsymbol{A}$ and $\boldsymbol{B}$ before estimating the parameter $t_{i,j}$, respectively. Then $\tilde{\boldsymbol{A}}^{(\text{new})}$ and $\tilde{\boldsymbol{B}}^{(\text{new})}$ stand for $\tilde{\boldsymbol{A}}$ and $\tilde{\boldsymbol{B}}$ updated by $\boldsymbol{T}^{(i,j)}(t_{i,j})$, respectively.

Proposition 1 *If we have $\tilde{\boldsymbol{B}}^{(new)} = \tilde{\boldsymbol{B}}\boldsymbol{T}^{(i,j)}(t_{i,j})$, then $\tilde{\boldsymbol{A}}^{(new)} = (\tilde{\boldsymbol{B}}^{(new)})^{\boxdot 2}$ can be written as a function of $t_{i,j}$ as follows:*

$$\tilde{\boldsymbol{A}}^{(new)} = (\tilde{\boldsymbol{B}}^{\boxdot 2})\boldsymbol{T}^{(i,j)}(t_{i,j}^2) + 2\, t_{i,j}(\tilde{\boldsymbol{b}}_i \boxdot \tilde{\boldsymbol{b}}_j)\boldsymbol{e}_j^{\intercal} \qquad (7)$$

where $\tilde{\boldsymbol{b}}_i$ denotes the i-th column of $\tilde{\boldsymbol{B}}$, and $\boldsymbol{e}_j$ is the j-th column of the identity matrix $\boldsymbol{I} \in \mathbb{R}^{N\times N}$.

The proof of proposition 1 is straightforward and therefore omitted. A natural way to compute the parameter $t_{i,j}$ is to minimize the criterion (4) w.r.t $t_{i,j}$ by replacing matrix $\boldsymbol{B}$ by $\tilde{\boldsymbol{B}}\boldsymbol{T}^{(i,j)}(t_{i,j})$. We use $J_+(t_{i,j})$ instead of $J_+(\tilde{\boldsymbol{B}}\boldsymbol{T}^{(i,j)}(t_{i,j}))$ for the sake of convenience.

According to (4), the minimization of $J_+(t_{i,j})$ requires to express the following update of the K matrices $\boldsymbol{C}^{(k)}$:

$$\boldsymbol{C}^{(k,\text{new})} = (\tilde{\boldsymbol{A}}^{(\text{new})})^{-1}\boldsymbol{C}^{(k)}(\tilde{\boldsymbol{A}}^{(\text{new})})^{-\intercal} \qquad (8)$$

as an explicit function of $t_{i,j}$. This can be done by means of proposition 1. From equation (7), the first term of the sum is a non singular matrix and the second term is a rank-1 matrix. The sum of such two matrices can be inverted by Sherman-Morrison formula [12]:

Theorem 1 *Suppose that $\boldsymbol{R}$ is a non singular square matrix and $\boldsymbol{u}$, $\boldsymbol{v}$ are two vectors satisfying $1+\boldsymbol{v}^{\intercal}\boldsymbol{R}^{-1}\boldsymbol{u} \neq 0$, then:*

$$(\boldsymbol{R} + \boldsymbol{u}\boldsymbol{v}^{\intercal})^{-1} = \boldsymbol{R}^{-1} - \frac{\boldsymbol{R}^{-1}\boldsymbol{u}\boldsymbol{v}^{\intercal}\boldsymbol{R}^{-1}}{1+\boldsymbol{v}^{\intercal}\boldsymbol{R}^{-1}\boldsymbol{u}} \qquad (9)$$

Suppose that $\tilde{\boldsymbol{B}}$, and the two vectors $2\,t_{i,j}(\tilde{\boldsymbol{b}}_i \boxdot \tilde{\boldsymbol{b}}_j)$ and $\boldsymbol{e}_j$ satisfy the conditions of the Sherman-Morrison formula, the expression of $(\tilde{\boldsymbol{A}}^{(\text{new})})^{-1}$ has the following form:

$$(\tilde{\boldsymbol{A}}^{(\text{new})})^{-1} = \boldsymbol{T}^{(i,j)}(-t_{i,j}^2)\boldsymbol{Q}(\tilde{\boldsymbol{B}}^{\boxdot 2})^{-1} \qquad (10)$$

with:

$$\boldsymbol{Q} = \boldsymbol{I} - \frac{2t_{i,j}}{1+2t_{i,j}\beta_j}\,\boldsymbol{\beta}\boldsymbol{e}_i^{\intercal} \qquad (11)$$

where $\boldsymbol{I} \in \mathbb{R}^{N\times N}$ is the identity matrix, $\boldsymbol{\beta} = (\tilde{\boldsymbol{B}}^{\boxdot 2})^{-1}(\tilde{\boldsymbol{b}}_i \boxdot \tilde{\boldsymbol{b}}_j)$ is a column vector and β_j is the j-th element of $\boldsymbol{\beta}$. Inserting (10) into (8), $\boldsymbol{C}^{(k,\text{new})}$ can be rewritten by:

$$\boldsymbol{C}^{(k,\text{new})} = \boldsymbol{T}^{(i,j)}(-t_{i,j}^2)\,\boldsymbol{Q}\,\tilde{\boldsymbol{C}}^{(k)}\boldsymbol{Q}^{\intercal}\,\boldsymbol{T}^{(i,j)}(-t_{i,j}^2)^{\intercal} \qquad (12)$$

where $\tilde{\boldsymbol{C}}^{(k)} = \tilde{\boldsymbol{A}}^{-1}\boldsymbol{C}^{(k)}\tilde{\boldsymbol{A}}^{-\intercal}$ is a constant matrix. Then through a straightforward computation of (12), each element of $\boldsymbol{C}^{(k,\text{new})}$ can be expressed as a function of $t_{i,j}$ as described in the following proposition:

Proposition 2 *Each non-diagonal element of* $\boldsymbol{C}^{(k,new)}$ *is a rational function in* $t_{i,j}$*:*

$$C_{m,n}^{(k,new)} = E_{m,n}^{(k,3)} t_{i,j}^3 + E_{m,n}^{(k,2)} t_{i,j}^2 + E_{m,n}^{(k,1)} t_{i,j} + E_{m,n}^{(k,0)} \quad (13)$$

where $E_{m,n}^{(k,3)}$, $E_{m,n}^{(k,2)}$, $E_{m,n}^{(k,1)}$ *and* $E_{m,n}^{(k,0)}$ *are the* (m,n)*-th elements of the* $(N \times N)$ *symmetric coefficient matrices* $\boldsymbol{E}^{(k,3)}$, $\boldsymbol{E}^{(k,2)}$, $\boldsymbol{E}^{(k,1)}$ *and* $\boldsymbol{E}^{(k,0)}$*, respectively, with* $1 \leq m \neq n \leq N$*. These coefficients are defined as follows:*

$$E_{m,n}^{(k,3)} = \begin{cases} \dfrac{2(\tilde{C}_{j,j}^{(k)} \beta_m - \tilde{C}_{m,j}^{(k)} \beta_j)}{(1 + 2t_{i,j}\beta_j)^2} & \text{if } n = i, 1 \leq m \neq i \leq N \\ E_{n,m}^{(k,3)} & \text{if } m = i, 1 \leq n \neq i \leq N \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

$$E_{m,n}^{(k,2)} = \begin{cases} \dfrac{4(\tilde{C}_{j,j}^{(k)} \beta_m \beta_n + \tilde{C}_{m,n}^{(k)} \beta_j^2 - (\tilde{C}_{m,j}^{(k)} \beta_n + \tilde{C}_{j,n}^{(k)} \beta_m)\beta_j)}{(1 + 2t_{i,j}\beta_j)^2} & \text{if } 1 \leq m < n \leq N, m \neq i, n \neq i \\ \dfrac{4(\tilde{C}_{j,j}^{(k)} \beta_m \beta_n + \tilde{C}_{m,n}^{(k)} \beta_j^2 - (\tilde{C}_{m,j}^{(k)} \beta_n + \tilde{C}_{j,n}^{(k)} \beta_m)\beta_j) - \tilde{C}_{m,j}^{(k)}}{(1 + 2t_{i,j}\beta_j)^2} & \text{if } n = i, 1 \leq m < i \\ E_{n,m}^{(k,2)} & \text{if } 1 \leq n < m \leq N \\ 0 & \text{otherwise} \end{cases} \quad (15)$$

$$E_{m,n}^{(k,1)} = \begin{cases} \dfrac{4\tilde{C}_{m,n}^{(k)} \beta_j - 2(\tilde{C}_{m,j}^{(k)} \beta_n + \tilde{C}_{j,n}^{(k)} \beta_m)}{(1 + 2t_{i,j}\beta_j)^2} & \text{if } 1 \leq m < n \leq N \\ E_{n,m}^{(k,1)} & \text{if } 1 \leq n < m \leq N \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

$$E_{m,n}^{(k,0)} = \begin{cases} \dfrac{\tilde{C}_{m,n}^{(k)}}{(1 + 2t_{i,j}\beta_j)^2} & \text{if } 1 \leq m \neq n \leq N \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

where $\tilde{C}_{m,n}^{(k)}$ *is the* (m,n)*-th element of the matrix* $\tilde{\boldsymbol{C}}^{(k)}$*.*

The proof of proposition 2 is omitted due to lack of space. Then the total sum of the squares of these non-diagonal elements can be expressed in a compact matrix form as follows:

$$J_+(t_{i,j}) = \sum_{k=1}^{K} \|\boldsymbol{E}^{(k)} \boldsymbol{\tau}\|_F^2 = \boldsymbol{\tau}^\intercal \boldsymbol{Q}_E \boldsymbol{\tau} \quad (18)$$

with:

$$\boldsymbol{Q}_E = \textstyle\sum_{k=1}^{K} \boldsymbol{E}^{(k)\intercal} \boldsymbol{E}^{(k)} \quad (19)$$

where $\boldsymbol{E}^{(k)} = [\mathrm{vec}\boldsymbol{E}^{(k,3)}, \mathrm{vec}\boldsymbol{E}^{(k,2)}, \mathrm{vec}\boldsymbol{E}^{(k,1)}, \mathrm{vec}\boldsymbol{E}^{(k,0)}]$ is a $(N^2 \times 4)$ matrix, $\boldsymbol{\tau} = [t_{i,j}^3, t_{i,j}^2, t_{i,j}, 1]^\intercal$ is a 4-dimensional parameter vector, and $\mathrm{vec}(.)$ reshapes a matrix into a column vector. Matrix $\boldsymbol{Q}_E$ is a (4×4) symmetric coefficient matrix.

Equation (18) shows that $J_+(t_{i,j})$ is a rational function, where the degrees of the numerator and the denominator are 6 and 4, respectively. The global minimum $t_{i,j}$ can be obtained by computing the roots of its derivative and selecting the one yielding the smallest value of (18). Once the optimal $t_{i,j}$ is computed, the matrix $\tilde{\boldsymbol{A}}$ is updated by computing $(\tilde{\boldsymbol{B}}\boldsymbol{T}^{(i,j)}(t_{i,j}))^{\boxdot 2}$. Then the Jacobi-like procedure is repeated to compute $t_{i,j}$ with the next (i,j) index. The processing of all the $N(N-1)$ parameters is called a sweep. The algorithm requires several sweeps to converge and can be stopped when the value of (18) falls below a fixed small positive threshold.

3. SIMULATION RESULTS

In this section, the performance of the proposed NNLUJ1D algorithm is evaluated with synthetic data, in order to show the influence of SNR and the number K of matrices to be jointly diagonalized. The proposed algorithm is compared with two classical non-orthogonal JDC methods, namely FFDIAG [3] and LUJ1D [4]. Then an illustration of BSS application is given to show the interest of the proposed algorithm. The performance is measured in terms of the error between the true diagonalizer $\boldsymbol{A}$ and its estimate $\tilde{\boldsymbol{A}}$. So the scale and permutation invariant distance defined in [13] is used:

$$\Delta(\boldsymbol{A}, \tilde{\boldsymbol{A}}) = \min_{\boldsymbol{\Pi}} \Psi(\boldsymbol{A}, \tilde{\boldsymbol{A}}\boldsymbol{\Pi}) \quad (20)$$

where the distance (20) requires to sweep all the $(N \times N)$ permutation matrices $\boldsymbol{\Pi}$, and:

$$\Psi(\boldsymbol{M}, \tilde{\boldsymbol{M}}) = \frac{1}{N} \sum_{n=1}^{N} \|\boldsymbol{m}_n - \frac{\tilde{\boldsymbol{m}}_n^\intercal \boldsymbol{m}_n}{\tilde{\boldsymbol{m}}_n^\intercal \tilde{\boldsymbol{m}}_n} \tilde{\boldsymbol{m}}_n\|_F^2 \quad (21)$$

with $\boldsymbol{m}_n$ and $\tilde{\boldsymbol{m}}_n$ the n-th columns of $\boldsymbol{M}$ and $\tilde{\boldsymbol{M}}$, respectively. A small $\Delta(\boldsymbol{A}, \tilde{\boldsymbol{A}})$ value means a better JDC performance in the sense that $\tilde{\boldsymbol{A}}$ is closer to $\boldsymbol{A}$.

The synthetic matrix set $\boldsymbol{\mathcal{C}}$ is generated randomly according to the JDC model (1). In the following tests, $\boldsymbol{A} \in \mathbb{R}_+^{3\times3}$ is generated according to (3) where $\boldsymbol{B} \in \mathbb{R}^{3\times3}$ is a random matrix with elements independently drawn from a real zero-mean unit-variance Gaussian distribution. The diagonal elements of $\boldsymbol{D}^{(k)} \in \mathbb{R}^{3\times3}$ are similarly generated. The resulting target set $\boldsymbol{\mathcal{C}}_N$ is perturbed by a white Gaussian noise array $\boldsymbol{\mathcal{N}}$:

$$\boldsymbol{\mathcal{C}}_N = \frac{\boldsymbol{\mathcal{C}}}{\|\boldsymbol{\mathcal{C}}\|_F} + \sigma_N \frac{\boldsymbol{\mathcal{N}}}{\|\boldsymbol{\mathcal{N}}\|_F} \quad (22)$$

where σ_N is a scalar controlling the noise level. Then the SNR is defined as $\mathrm{SNR} = -20\log_{10}(\sigma_N)$. Moreover, we repeat all the experiments with 500 independent Monte Carlo trials. All the algorithms stop either when the relative error of the corresponding criterion between two successive sweeps is less than 10^{-5} or when the number of sweeps exceeds 500. All the algorithms are initialized by identity matrices.

3.1. Effect of SNR

In this test, the compared algorithms are applied to noisy data for different SNR levels. The number of matrices is set to $K = 3$. We repeat the experiment with SNR ranging from 0 dB to 30 dB. Figure 1 shows the averaged error $\Delta(\boldsymbol{A}, \tilde{\boldsymbol{A}})$ of compared algorithms as a function of SNR.

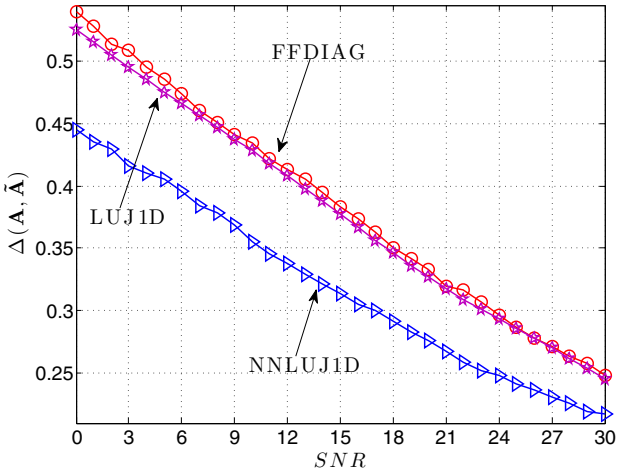


Fig. 1. The average error $\Delta(\boldsymbol{A}, \tilde{\boldsymbol{A}})$ evolution of compared algorithms versus SNR. The dimension of $\boldsymbol{A}$ and the number of matrices are set to $N = 3$ and $K = 3$, respectively.

One can notice that, as SNR grows, the performance of all the JDC methods increases quasi-linearly. The difference between FFDIAG and LUJ1D is small. It appears that the proposed NNLUJ1D algorithm achieves better estimations than FFDIAG and LUJ1D, especially when the SNR values are lower than 15 dB.

3.2. Effect of the number of matrices

In this test, the effect of the number K of matrices on the performance of the compared algorithms is evaluated. The SNR value is fixed to 5 dB. We repeat the experiment with K ranging from 3 to 30. Figure 2 shows the averaged error $\Delta(\boldsymbol{A}, \tilde{\boldsymbol{A}})$ of compared algorithms as a function of K. For all the JDC algorithms, the increase of K induces better estimation performance. The classical algorithms FFDIAG and LUJ1D behave similarly. NNLUJ1D maintains a competitive advantage over FFDIAG and LUJ1D through all K values.

3.3. Performance of BSS of infrared spectra

In this section, we demonstrate the potential usefulness of the proposed algorithm through a BSS application carried out on infrared spectra. NNLUJ1D is compared with three well-known BSS algorithms, namely the ICA methods CoM2 [14] and SOBI [15], and a NMF method based on alternating NonNegative Least Squares (NNLS) [16]. Three gas phase Fourier-transform infrared spectra of three materials, namely Toluene, Dichloromethane and Methanol (see figure 3(a)), with 0.125 cm^{-1} resolution [17], serve as source signals $\boldsymbol{s}$. Three linear observations (see figure 3(b)) are created by the linear mixed model $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s}$ with $\boldsymbol{A}$ similarly generated as in the previous section. The matrix set $\boldsymbol{\mathcal{C}}$ contains 12 matrix slices chosen from the third and fourth order cumulants.

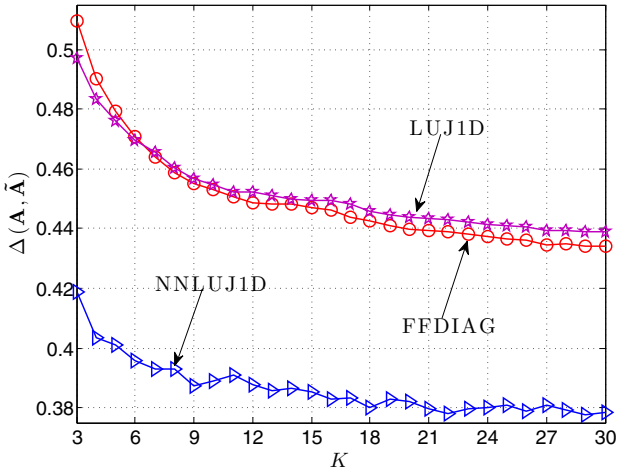


Fig. 2. The average error $\Delta(\boldsymbol{A}, \tilde{\boldsymbol{A}})$ evolution of compared algorithms versus the number of matrices K. The dimension of $\boldsymbol{A}$ and the SNR are set to $N = 3$ and SNR $= 5$ dB, respectively.

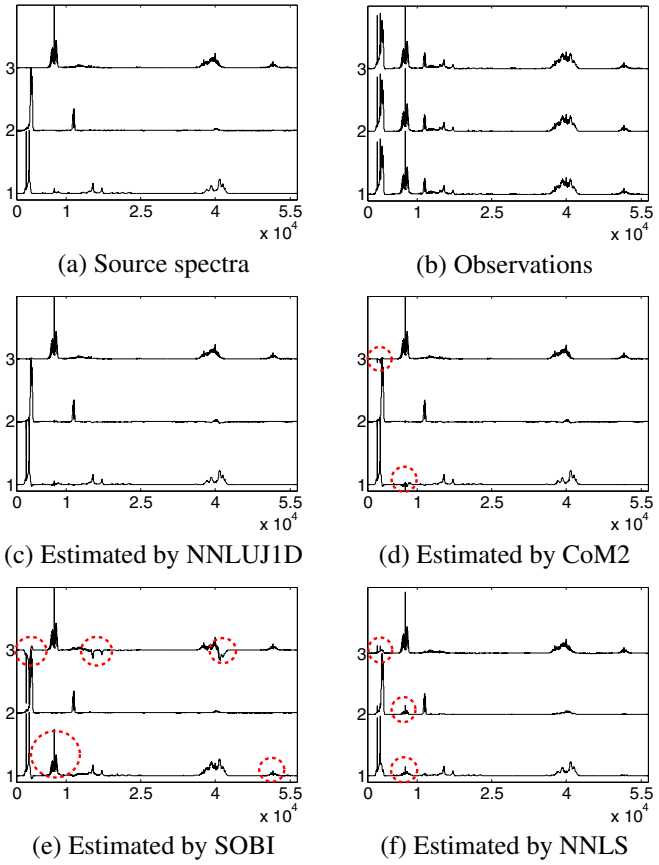
Table 1. Average estimating errors of the mixing matrix $\Delta(\boldsymbol{A}, \tilde{\boldsymbol{A}})$ and the sources $\Delta(\boldsymbol{s}^\top, \tilde{\boldsymbol{s}}^\top)$ of four methods for BSS of infrared spectra

	NNLUJ1D	CoM2	SOBI	NNLS
$\Delta(\boldsymbol{A}, \tilde{\boldsymbol{A}})$	0.0401	0.0494	0.2845	0.1144
$\Delta(\boldsymbol{s}^\top, \tilde{\boldsymbol{s}}^\top)$	0.0690	0.0837	0.4165	0.1863

The average estimation errors $\Delta(\boldsymbol{A}, \tilde{\boldsymbol{A}})$ of the mixing matrices and that $\Delta(\boldsymbol{s}^\top, \tilde{\boldsymbol{s}}^\top)$ of the source spectra of the compared methods are shown in table 1. It can be seen that the proposed NNLUJ1D method gives the smallest estimating errors both for $\boldsymbol{A}$ and $\boldsymbol{s}$. The estimated spectra of the compared methods are displayed in figures 3(c) to 3(f). It shows that the proposed NNLUJ1D algorithm gives a better result than classic methods. The results infer that when the source spectra are partially correlated, using only the independency or the non-negativity may not yield a perfect decomposition. In such a case, the complementary nonnegative information along with the independence could improve the separation result.

4. CONCLUSION

In this paper, we address the problem of nonnegative JDC. We exploit the elementary triangular parameterization of the Hadamard square root of the nonnegative joint diagonalizer. Thus we propose a Jacobi-like approach. In each Jacobi-like iteration, the optimization is formulated into the minimization of a rational function w.r.t only one parameter. Numerical experiments on simulated data spotlight the advantages of the proposed method in the presence of bottlenecks, such as for low SNR values and a small number of available matrices to be jointly diagonalized. Furthermore, the potential interest of the proposed algorithm is demonstrated through a BSS exper-



(a) Source spectra (b) Observations

(c) Estimated by NNLUJ1D (d) Estimated by CoM2

(e) Estimated by SOBI (f) Estimated by NNLS

Fig. 3. Infrared source spectra, mixtures and estimated spectra by NNLUJ1D, CoM2, SOBI and NNLS.

imental context.

5. REFERENCES

[1] G. Chabriel and J. Barrère, "A direct algorithm for nonorthogonal approximate joint diagonalization," *IEEE Trans. Signal Process.*, vol. 60, no. 1, pp. 39–47, 2012.

[2] J. F. Cardoso and A. Souloumiac, "Jacobi angles for simultaneous diagonalization," *SIAM J. Matrix Anal. Appl.*, vol. 17, pp. 161–164, 1996.

[3] A. Ziehe, P. Laskov, G. Nolte, and K. R. Muller, "A fast algorithm for joint diagonalization with non-orthogonal transformations and its application to blind source separation," *J. Mach. Learning Res.*, vol. 5, pp. 777–800, 2004.

[4] B. Afsari, "Simple LU and QR based non-orthogonal matrix joint diagonalization," in *ICA 2006, Springer LNCS series*, 2006.

[5] A. Souloumiac, "Nonorthogonal joint diagonalization by combining Givens and hyperbolic rotations," *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2222–2231, 2009.

[6] A. Yeredor, "Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation," *IEEE Trans. Signal Process.*, vol. 50, no. 7, pp. 1545–1553, 2002.

[7] A. J. van der Veen, "Joint diagonalization via subspace fitting techniques," in *Proc. ICASSP '01*, 2001, vol. 5, pp. 2773–2776.

[8] D. T. Pham, "Joint approximate diagonalization of positive definite hermitian matrices," *SIAM J. Matrix Anal. Appl.*, vol. 22, pp. 1837–1848, 2001.

[9] P. Comon and C. Jutten, *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, Academic Press, 2010.

[10] P. Sajda, S. Du, T. R. Brown, R. Stoyanova, D. C. Shungu, X. L. Mao, and L. C. Parra, "Nonnegative matrix factorization for rapid recovery of constituent spectra in magnetic resonance chemical shift imaging of the brain," *IEEE Trans. Med. Imag.*, vol. 23, no. 12, pp. 1453–1465, 2004.

[11] M. Chu, F. Diele, R. Plemmons, and S. Ragni, "Optimality computation and interpretation of non negative matrix factorizations," Tech. Rep., Wake Forest University, 2004.

[12] M. S. Bartlett, "An inverse matrix adjustment arising in discriminant analysis," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 107–111, 1951.

[13] P. Comon, X. Luciani, and A. L. F. de Almeida, "Tensor decompositions, alternating least squares and other tales," *J. Chemometr.*, vol. 23, pp. 393–405, 2009.

[14] P. Comon, "Independent component analysis, a new concept?," *Signal Process.*, vol. 36, no. 3, pp. 287–314, 1994.

[15] A. Belouchrani, K. Abed-Meraim, J. F. Cardoso, and E. Moulines, "A blind source separation technique using second-order statistics," *IEEE Trans. Signal Process.*, vol. 45, no. 2, pp. 434–444, 1997.

[16] H. Kim and H. Park, "Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 2, pp. 713–730, 2008.

[17] P. M. Chu, F. R. Guenther, G. C. Rhoderick, and W. J. Lafferty, "The NIST quantitative infrared database," *J. Res. Natl. Inst. Stand. Technol*, vol. 104, no. 59, pp. 59–81, 1999.