

# Nouvel algorithme de zéro-bloc diagonalisation conjointe par approche de Levenberg-Marquardt

Omar CHERRAK<sup>1,2</sup>, Hicham GHENNIoui<sup>1</sup>, Nadège THIRION-MOREAU<sup>2</sup>, El Hossain ABARKAN<sup>1</sup>

<sup>1</sup>Université Sidi Mohamed Ben Abdellah

Faculté des Sciences et Techniques de Fès, B.P. 2202, Route d'Immouzzar, Fès, Maroc

<sup>2</sup>Aix-Marseille Université, CNRS, ENSAM, LSIS, UMR 7296, F-13397 Marseille, France

Université de Toulon, CNRS, LSIS, UMR 7296, F-83957 La Garde, France, Tel: +33 4 94 14 24 56/26 71

{omar.cherrak,hicham.ghennioui}@usmba.ac.ma, thirion@univ-tln.fr, habarkan@yahoo.fr

**Résumé** – Dans cette communication, nous introduisons un nouveau problème de décomposition matricielle conjointe d'un ensemble de matrices complexes donné appelé Zéro-Bloc Diagonalisation Conjointe Non-Unitaire. Ce problème peut se rencontrer dans différents domaines d'applications dont la séparation aveugle de mélanges convolutifs de sources (ou déconvolution multi-dimensionnelle) notamment dans les méthodes du domaine temps-fréquence spatial quadratique. Nous présentons une nouvelle approche basée sur un schéma d'optimisation numérique itératif de type Levenberg-Marquardt, lequel nécessite les matrices Hessiennes complexes ou des approximations de celles-ci. Nous proposons donc, ici, le calcul exact des matrices Hessiennes puis nous en dérivons l'algorithme suggéré. Les principaux avantages de cette méthode découlent des propriétés de l'algorithme Levenberg-Marquardt lequel est stable, robuste et rapide. Des simulations numériques illustrent le comportement et l'efficacité de l'approche proposée dans différents contextes.

**Abstract** – In this communication, we address a new problem of joint decomposition of a given set of complex matrices, namely the Non-Unitary Joint Zero-Block Diagonalization. This problem might be encountered in various fields of applications, among which is the blind separation of convolutive mixtures of sources (also known as multidimensional deconvolution) especially when spatial quadratic time-frequency domain methods are considered. We present a new approach based on the Levenberg-Marquardt algorithm which is an iterative numerical optimization scheme that requires the calculation of the complex Hessian matrices or approximations of these latter. We propose, here, the exact calculation of complex Hessian matrices, then the suggested algorithm can be derived. The main advantages of the proposed method stem from the properties of Levenberg-Marquardt algorithm which is stable, more robust and exhibits fast convergence. Computer simulations are provided in order to illustrate the behavior and efficiency of the proposed method in different contexts.

## 1 Introduction

Durant les dernières années, le problème de la décomposition conjointe d'ensembles de matrices (voire de tenseurs) a suscité de nombreux travaux en traitement du signal, essentiellement dans le cadre d'applications en séparation aveugle de sources (SAS) et en traitement d'antenne (voir [6] pour un panorama assez exhaustif). L'un des tout premiers problèmes à avoir été considéré est celui de la diagonalisation conjointe (DC) d'un ensemble de matrices sous la contrainte d'unitarité du diagonaliseur conjoint recherché, conduisant à de nombreux algorithmes en SAS pour des mélanges instantanés, les plus connus étant JADE [3] et SOBI [2]. Les travaux suivants ont traité du problème de la DC de matrices mais en levant la contrainte d'unitarité sur la matrice recherchée [4][19][20] ou de la DC de tenseurs [9][16]. Un second type de décomposition matricielle, dénommé Bloc-Diagonalisation Conjointe (BDC), a été appliqué dans des problèmes de localisation de sources large bande en présence de bruit corrélé et de SAS pour des mélanges convolutifs (ou déconvolution multidimensionnelle). Plusieurs algorithmes ont été suggérés sous différentes

hypothèses qu'elles concernent l'ensemble des matrices à bloc-diagonaliser (définies positives ou hermitiennes) ou le bloc-diagonaliseur (supposé unitaire [10] ou non [7][13][17]). Enfin, un troisième type de décomposition matricielle, dénommé Zéro Diagonalisation Conjointe (ZDC), s'est avéré utile en SAS, télécommunications numériques et cryptographie. Si le premier algorithme suggéré opérait sous contrainte d'unitarité [1], parce qu'il était appliqué après une étape classique de blanchiment spatial, le fait que cette étape borne les performances a très rapidement conduit au développement d'autres solutions sans contrainte [5][11][12][21].

Nous considérons, ici, le problème de la zéro bloc-diagonalisation conjointe non unitaire (ZBDC<sub>NU</sub>) d'un ensemble de matrices complexes. Il généralise celui de la ZDC non unitaire. Nous proposons un nouvel algorithme basé sur un schéma d'optimisation itératif de Levenberg-Marquardt lequel nécessite les matrices Hessiennes complexes ou des approximations de celles-ci. L'algorithme que nous dérivons ici est fondé sur un calcul exact des matrices Hessiennes. Des simulations numériques illustrent le comportement de l'approche proposée dans diffé-

rentes situations. Les matrices sont d'abord considérées comme exactement zéro bloc-diagonales, puis elles sont ensuite progressivement perturbées par un bruit additif Gaussien. Enfin, ce nouvel algorithme de ZBDC<sub>NU</sub> est comparé à l'algorithme proposé dans [8] reposant sur le gradient conjugué non linéaire lequel ne nécessite pas le calcul des matrices Hessiennes. Cette comparaison illustre l'intérêt de la méthode suggérée.

## 2 Énoncé du problème & hypothèses

Le problème de la zéro-bloc diagonalisation conjointe non unitaire (ZBDC<sub>NU</sub>) s'énonce comme suit. Considérons un ensemble  $\mathcal{N}$  de  $N_m$  ( $N_m \in \mathbb{N}^*$ ) matrices carrées  $\mathbf{X}_i \in \mathbb{C}^{M \times M}$ ,  $i \in \{1, \dots, N_m\}$  admettant toutes une factorisation de la forme  $\mathbf{X}_i = \mathbf{A}\mathbf{Z}_i\mathbf{A}^H$  où  $(\cdot)^H$  désigne l'opérateur de transposition conjuguaison et où les matrices

$$\mathbf{Z}_i = \begin{pmatrix} \mathbf{0}_{11} & \mathbf{Z}_{i,12} & \dots & \mathbf{Z}_{i,1r} \\ \mathbf{Z}_{i,21} & \mathbf{0}_{22} & \ddots & \mathbf{Z}_{i,2r} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{Z}_{i,r1} & \dots & \vdots & \mathbf{0}_{rr} \end{pmatrix} \forall i \in \{1, \dots, N_m\},$$

sont des matrices zéro-bloc diagonales de taille  $N \times N$ , où  $r \in \mathbb{N}^*$  est le nombre de blocs considérés. Les matrices  $\mathbf{Z}_{i,jk}$ ,  $i \in \{1, \dots, N_m\}$ ,  $j, k \in \{1, \dots, r\}$  sont des matrices rectangulaires de taille  $n_j \times n_k$  de telle sorte que  $n_1 + \dots + n_r = N$  alors que  $\mathbf{0}_{jj}$  désigne la matrice nulle de taille  $n_j \times n_j$ .  $\mathbf{A}$  est une matrice de rang plein de taille  $M \times N$  ( $M \geq N$ ) tandis que sa pseudo-inverse (ou inverse généralisée de Moore-Penrose) de taille  $N \times M$  est notée  $\mathbf{B}$ . L'ensemble des  $N_m$  matrices  $\mathbf{Z}_i \in \mathbb{C}^{N \times N}$  zéro-bloc diagonales est noté  $\mathcal{Z}$ . La taille des blocs  $n_j$  pour tout  $j = 1, \dots, r$  est supposée connue.

Le problème de la ZBDC<sub>NU</sub> consiste à estimer soit l'ensemble des matrices zéro-bloc diagonales  $\mathcal{Z}$  et la matrice  $\mathbf{A}$  soit la matrice  $\mathbf{B}$ , ce uniquement à partir des matrices de l'ensemble  $\mathcal{N}$ . Pour résoudre ce problème, nous suggérons, ici, de considérer la fonction de coût suivante :

$$\mathcal{C}_{ZBD}(\mathbf{B}) = \sum_{i=1}^{N_m} \|\text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i\mathbf{B}^H\}\|_F^2, \quad (1)$$

où  $\|\cdot\|_F$  désigne la norme de Frobenius et où l'opérateur matriciel  $\text{Bdiag}_{(n)}\{\cdot\}$  est défini de la manière suivante :

$$\text{Bdiag}_{(n)}\{\mathbf{X}\} = \begin{pmatrix} \mathbf{X}_{11} & \mathbf{0}_{12} & \dots & \mathbf{0}_{1r} \\ \mathbf{0}_{21} & \mathbf{X}_{22} & \ddots & \mathbf{0}_{2r} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0}_{r1} & \mathbf{0}_{r2} & \dots & \mathbf{X}_{rr} \end{pmatrix}, \quad (2)$$

avec  $\mathbf{X}$  matrice carrée  $N \times N$  composée de blocs  $\mathbf{X}_{ij}$  de taille  $n_i \times n_j$ , pour tout  $i$  et  $j$  allant de 1 à  $r$  (avec  $n_1 + \dots + n_r = N$ ). On note enfin  $\mathbf{n} = (n_1, n_2, \dots, n_r)$ .

## 3 Nouvel algorithme de ZBDC<sub>NU</sub>

La fonction de coût donnée au niveau de l'Eq. (1) doit être minimisée afin d'estimer le zéro-bloc diagonaliseur conjoint  $\mathbf{B} \in \mathbb{C}^{N \times M}$ . Pour ce faire, nous proposons un nouvel algorithme basé sur une approche itérative de type Levenberg-Marquardt (LMA) [15]. Cette approche connue pour sa robustesse

et son efficacité a été largement utilisée dans divers domaines dont les réseaux de neurones, le traitement d'antenne, la séparation aveugle de source, etc...

### 3.1 Principe & calculs

Partant d'une matrice  $\mathbf{B}$  initiale, notée  $\mathbf{B}^{(0)}$ , à chaque itération  $m$ , nous devons ré-estimer cette matrice que nous notons alors  $\mathbf{B}^{(m)}$ . Cette matrice peut aussi être stockée dans un vecteur  $\mathbf{b}^{(m)} = \text{vec}\{\mathbf{B}^{(m)}\}$ ,  $\text{vec}\{\cdot\}$  étant l'opérateur de vectorisation qui permet de ranger les colonnes d'une matrice  $N \times M$  dans un vecteur de taille  $(N \times M) \times 1$ . Dans un schéma de type LMA, ce vecteur est mis à jour selon la règle d'adaptation suivante, pour tout  $m = 1, 2, \dots$  :

$$\mathbf{b}^{(m)} = \mathbf{b}^{(m-1)} - \left[ \mathbf{H}^{(m-1)} + \lambda \mathbf{I}_{MN} \right]^{-1} \mathbf{g}^{(m-1)}, \quad (3)$$

où  $[\cdot]^{-1}$  désigne l'inverse d'une matrice,  $\lambda$  est un petit coefficient (positif) d'amortissement,  $\mathbf{I}_{MN}$  est la matrice identité de taille  $MN \times MN$ ,  $\mathbf{H}$  désigne la matrice Hessienne et le vecteur  $\mathbf{g}$  résulte de la vectorisation du gradient matriciel complexe noté  $\mathbf{G}^{(m-1)} = \nabla_{\mathbf{a}} \mathcal{C}_{ZBD}(\mathbf{B}^{(m-1)})$  i.e.  $\mathbf{g}^{(m-1)} = \text{vec}\{\mathbf{G}^{(m-1)}\}$ . Le gradient matriciel a été calculé dans [8] où l'on démontre que cette matrice vaut :

$$2 \sum_{i=1}^{N_m} \left[ \left( \text{Bdiag}\{\mathbf{B}\mathbf{X}_i\mathbf{B}^H\} \right) \mathbf{B}\mathbf{X}_i^H + \left( \text{Bdiag}\{\mathbf{B}\mathbf{X}_i\mathbf{B}^H\} \right)^H \mathbf{B}\mathbf{X}_i \right], \quad (4)$$

$(\cdot)^T$  étant l'opérateur de transposition et  $(\cdot)^*$  celui de conjuguaison. Il nous reste donc à calculer la matrice Hessienne  $\mathbf{H} =$

$\begin{pmatrix} \mathbf{H}_{\mathbf{B},\mathbf{B}^*} & \mathbf{H}_{\mathbf{B}^*,\mathbf{B}^*} \\ \mathbf{H}_{\mathbf{B},\mathbf{B}} & \mathbf{H}_{\mathbf{B}^*,\mathbf{B}} \end{pmatrix}$  elle-même composée de quatre matrices complexes définies comme dans [14], par :

$$\mathbf{H}_{\mathbf{B},\mathbf{B}^*} = \frac{\mathbf{A}_{00} + \mathbf{A}_{11}^T}{2} = [\mathbf{H}_{\mathbf{B}^*,\mathbf{B}}]^T, \quad (5)$$

$$\mathbf{H}_{\mathbf{B}^*,\mathbf{B}^*} = \frac{\mathbf{A}_{01} + \mathbf{A}_{01}^T}{2}, \quad \mathbf{H}_{\mathbf{B},\mathbf{B}} = \frac{\mathbf{A}_{10} + \mathbf{A}_{10}^T}{2}, \quad (6)$$

où les quatre matrices complexes  $\mathbf{A}_{00}$ ,  $\mathbf{A}_{01}$ ,  $\mathbf{A}_{10}$  et  $\mathbf{A}_{11}$  impliquées sont déterminées au moyen du calcul de la dérivée seconde de la fonction de coût notée  $d^2 \mathcal{C}_{ZBD}(\mathbf{B})$  que l'on commence par mettre sous la forme :

$$d^2 \mathcal{C}_{ZBD}(\mathbf{B}) = \left[ (\text{dvec}\{\mathbf{B}^*\})^T, (\text{dvec}\{\mathbf{B}\})^T \right] \begin{bmatrix} \mathbf{A}_{00} & \mathbf{A}_{01} \\ \mathbf{A}_{10} & \mathbf{A}_{11} \end{bmatrix} \begin{bmatrix} \text{dvec}\{\mathbf{B}\} \\ \text{dvec}\{\mathbf{B}^*\} \end{bmatrix}. \quad (7)$$

En annexe, nous établissons qu'elles valent :

$$\mathbf{A}_{00} = \mathbf{A}_{11}^* = \left( \mathbf{X}_i^T \mathbf{B}^T \otimes \mathbf{I}_N \right) \mathbf{T}_{\text{Bdiag}}^T \left( \mathbf{B}^* \mathbf{X}_i^* \otimes \mathbf{I}_N \right) + \left( \mathbf{X}_i^* \mathbf{B}^T \otimes \mathbf{I}_N \right) \mathbf{T}_{\text{Bdiag}}^T \left( \mathbf{B}^* \mathbf{X}_i^T \otimes \mathbf{I}_N \right) \quad (8)$$

$$+ \mathbf{X}_i^* \otimes \left( \text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i\mathbf{B}^H\} \right) + \mathbf{X}_i^T \otimes \text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i^H\mathbf{B}^H\},$$

$$\mathbf{A}_{10} = \mathbf{A}_{01}^* = \mathbf{T}_{\text{vec}}^T \left( \mathbf{I}_N \otimes \mathbf{X}_i \mathbf{B}^H \right) \mathbf{T}_{\text{Bdiag}}^T \left( \mathbf{B}^* \mathbf{X}_i^* \otimes \mathbf{I}_N \right) + \mathbf{T}_{\text{vec}}^T \left( \mathbf{I}_N \otimes \mathbf{X}_i^H \mathbf{B}^H \right) \mathbf{T}_{\text{Bdiag}}^T \left( \mathbf{B}^* \mathbf{X}_i^T \otimes \mathbf{I}_N \right), \quad (9)$$

où l'opérateur  $\otimes$  désigne le produit de Kronecker,  $\mathbf{T}_{\text{vec}}$  est une matrice carrée de permutation de taille  $(MN \times MN)$  et  $\mathbf{T}_{\text{Bdiag}} = \text{diag}\{\text{vec}\{\text{Bdiag}\{\mathbf{1}_N\}\}\}$ ,  $\mathbf{1}_N$  est la matrice  $N \times N$  dont tous les éléments sont égaux à un et  $\text{diag}\{\mathbf{a}\}$  est une matrice diagonale carrée qui contient les éléments du vecteur  $\mathbf{a}$  sur sa diagonale.

Selon [18], il est suggéré de ne considérer que la diagonale (partie réelle) de la matrice Hessienne (i.e. Eq. (5)) afin d'avoir facilement son inverse et assurer la minimisation de la fonction de coût .

### 3.2 L'algorithme en résumé

Le principe de l'algorithme proposé, noté  $\text{ZBDC}_{\text{LM}}$ , est récapitulé ci-dessous :

**Data:**  $N_m$  matrices carrées  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_{N_m}$ , critère d'arrêt  $\epsilon$  (nbre positif petit), nombre max. d'itérations  $M_{\text{max}}$

**Result:** Estimation du zéro bloc-diagonaliseur conjoint  $\mathbf{B}$   
Initialisations :  $\mathbf{B}^{(0)}$ ;  $m = 0$ ; facteur d'amortissement  $\lambda^{(0)}$

**repeat**

  Calcul du gradient matriciel complexe

$\mathbf{g}^{(m)} = \text{vec}\{\nabla_a \mathcal{C}_{\text{ZBD}}(\mathbf{B}^{(m)})\}$  (Eq. (4))

  Calcul des deux blocs de la diagonale de la Hessienne (Eq. (5))

  Calcul de  $\mathbf{b}^{(m+1)} = \text{vec}\{\mathbf{B}^{(m+1)}\}$  (Eq. (3))

  Calcul de l'erreur  $e^{(m)} = \frac{1}{N_m} \mathcal{C}_{\text{ZBD}}(\mathbf{B}^{(m+1)})$

$m = m + 1$ ;

**if**  $e^{(m)} \geq e^{(m-1)}$  **then**

$\lambda^{(m)} = \frac{\lambda^{(m-1)}}{10}$ ,  $e^{(m)} = e^{(m-1)}$

**else**

$\lambda^{(m)} = 10\lambda^{(m-1)}$

**end**

**until** ( $\|\mathbf{B}^{(m+1)} - \mathbf{B}^{(m)}\|_F^2 \leq \epsilon$ ) ou ( $m \geq M_{\text{max}}$ );

## 4 Simulations numériques

Le comportement de l'algorithme est maintenant évalué grâce à des simulations numériques. Nous commençons par considérer des ensembles de  $N_m$  matrices exactement zéro-bloc diagonales générées aléatoirement (*i.e.* les éléments des matrices que nous construisons suivent tous une loi Gaussienne de moyenne 0 et de variance 1). Ensuite, nous les perturbons afin d'évaluer la robustesse de l'algorithme dans le cas d'erreurs vis à vis de l'hypothèse de zéro bloc-diagonalité. Dans ce second cas, une matrice dont les termes aléatoires sont tirés selon une loi Gaussienne de moyenne 0 et de variance  $\sigma_b^2$  est ajoutée à la précédente. Le rapport signal à bruit (RSB) est alors défini en dB comme  $\text{RSB} = 10 \log(\frac{1}{\sigma_b^2})$ . Pour juger de la qualité de l'estimation, nous utilisons l'indice d'erreur suivant  $I_{\text{conv}}(\mathbf{G})$  - généralisant celui introduit dans [16] :

$$\frac{1}{r(r-1)} \left[ \sum_{i=1}^r \left( \sum_{j=1}^r \frac{\|(\mathbf{G})_{i,j}\|_F^2}{\max_{\ell} \|(\mathbf{G})_{i,\ell}\|_F^2} - 1 \right) + \sum_{j=1}^r \left( \sum_{i=1}^r \frac{\|(\mathbf{G})_{i,j}\|_F^2}{\max_{\ell} \|(\mathbf{G})_{\ell,j}\|_F^2} - 1 \right) \right] \quad (10)$$

où  $(\mathbf{G})_{i,j} \forall i, j \in \{1, \dots, r\}$  est la  $(i, j)$ -ème matrice zéro-bloc de  $\mathbf{G} = \hat{\mathbf{B}}\mathbf{A}$ . Quand l'estimation est parfaite, l'indice d'erreur doit tendre vers  $-\infty$  en échelle logarithmique (0 en échelle linéaire). Les résultats affichés résultent du moyennage par méthode de Monte-Carlo sur 100 tirages, en initialisant toujours à partir de la décomposition en valeurs propres généralisées [17]. Au niveau de la Fig. 1, partie gauche (resp. partie droite), nous comparons les performances de l'algorithme proposé ( $\text{ZBDC}_{\text{LM}}$ ) à celles de l'algorithme  $\text{ZBDC}_{\text{GC}}$  [8] (basé sur la méthode du Gradient Conjugué qui ne nécessite pas le calcul des matrices Hessiennes) en fonction du nombre d'itérations pour une taille de bloc  $r = 4$  en considérant différents nombres de matrices ( $N_m = 5, 20$  et  $100$ ) dans le cas non bruité c'est-à-dire pour un RSB de 100 dB (resp. dans le cas bruité avec un RSB de 25 dB).

Nous montrons ensuite au niveau de la courbe de gauche de la Fig. 2 l'influence de la taille  $N_m$  de l'ensemble à zéro-bloc dia-

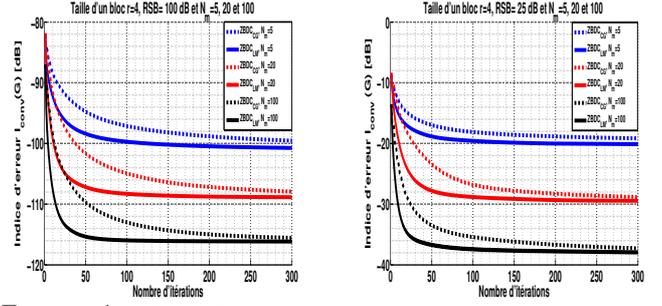


FIGURE 1 –  $I_{\text{conv}}(\mathbf{G})$  en dB en fonction du nombre d'itérations pour  $N_m = 5, 20$  et  $100$  matrices, en considérant la taille d'un bloc  $r = 4$ . Gauche : RSB = 100 dB. Droite : RSB = 25 dB.

gonaliser conjointement sur les résultats. Nous traçons l'évolution des performances en fonction de  $N_m$  pour un nombre de blocs  $r = 4$  et pour différentes valeurs de RSB = 20, 40 et 100 dB. Sur la courbe de droite de la Fig. 2, nous illustrons l'évolution des performances par rapport à différentes valeurs de RSB pour un nombre de blocs  $r = 4$  et pour différents nombres de matrices à zéro bloc-diagonaliser conjointement :  $N_m = 5, 20$  et  $100$ .

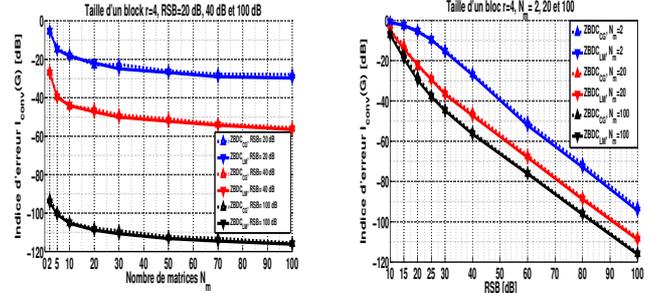


FIGURE 2 – Comportement de l'algorithme  $\text{ZBDC}_{\text{LM}}$ . Gauche :  $I_{\text{conv}}(\mathbf{G})$  en dB en fonction du nombre  $N_m$  de matrices considérées pour différentes valeurs de RSB = 20 dB, 40 dB et 100 dB. Droite :  $I_{\text{conv}}(\mathbf{G})$  en dB en fonction du RSB pour des ensembles de taille variable ( $N_m = 2, 20$  et  $100$  matrices).

Ces simulations illustrent le bon comportement de l'algorithme proposé. Nous constatons que plus le RSB est favorable et/ou plus le nombre de matrices  $N_m$  à zéro bloc-diagonaliser est grand, meilleurs sont les résultats.

## 5 Conclusion

Dans cette communication, nous avons proposé un nouvel algorithme pour la zéro bloc diagonalisation conjointe non unitaire, noté  $\text{ZBDC}_{\text{LM}}$ , car il repose sur un schéma d'optimisation de type Levenberg-Marquardt. Cet algorithme repose sur un calcul exact des matrices Hessiennes complexes. Le principal avantage de cet algorithme est qu'il généralise le problème de la zéro-diagonalisation conjointe tout en témoignant de bonnes performances dans des situations complexes (cas bruité et/ou faible nombre de matrices dans l'ensemble à zéro bloc-diagonaliser conjointement). Les extensions futures de ces travaux consisteront à illustrer l'intérêt de cet algorithme dans un véritable contexte de séparation aveugle de mélanges convo-

lutifs de sources (plus particulièrement pour les méthodes du domaine temps-fréquence spatial quadratique).

## ANNEXE

Considérant quatre matrices carrées  $\mathbf{D}_1$ ,  $\mathbf{D}_2$ ,  $\mathbf{D}_3$  de taille  $M \times M$  et  $\mathbf{D}_4$  de taille  $M \times N$ .  $\text{tr}\{\cdot\}$  désigne l'opérateur de trace. Nos développements sont basés sur les propriétés suivantes [14] :

- P<sub>1</sub>.  $\|\text{Bdiag}\{\mathbf{D}_1\}\|_F^2 = \text{tr}\{\mathbf{D}_1^H \text{Bdiag}\{\mathbf{D}_1\}\}$ .
- P<sub>2</sub>.  $\text{tr}\{\mathbf{D}_1\} = \text{tr}\{\mathbf{D}_1^T\}$ .
- P<sub>3</sub>.  $\text{tr}\{\mathbf{D}_1 + \mathbf{D}_2\} = \text{tr}\{\mathbf{D}_1\} + \text{tr}\{\mathbf{D}_2\}$ .
- P<sub>4</sub>.  $\text{tr}\{\mathbf{D}_1 \mathbf{D}_2 \mathbf{D}_3\} = \text{tr}\{\mathbf{D}_3 \mathbf{D}_1 \mathbf{D}_2\} = \text{tr}\{\mathbf{D}_2 \mathbf{D}_3 \mathbf{D}_1\}$ .
- P<sub>5</sub>.  $\text{tr}\{\mathbf{D}_1^H \mathbf{D}_2\} = (\text{vec}\{\mathbf{D}_1\})^H \text{vec}\{\mathbf{D}_2\}$ .
- P<sub>6</sub>.  $d(\text{tr}\{\mathbf{D}_1\}) = \text{tr}\{d(\mathbf{D}_1)\}$ .
- P<sub>7</sub>.  $\text{vec}\{\mathbf{D}_1 + \mathbf{D}_2\} = \text{vec}\{\mathbf{D}_1\} + \text{vec}\{\mathbf{D}_2\}$ .
- P<sub>8</sub>.  $\text{vec}\{\mathbf{D}_1 \mathbf{D}_2 \mathbf{D}_3\} = (\mathbf{D}_3^T \otimes \mathbf{D}_1) \text{vec}\{\mathbf{D}_2\}$ .
- P<sub>9</sub>.  $\text{vec}\{\mathbf{D}_1^T\} = \mathbf{T}_{vec} \text{vec}\{\mathbf{D}_1\}$ .
- P<sub>10</sub>.  $\text{vec}\{\text{Bdiag}\{\mathbf{D}_1\}\} = \mathbf{T}_{Bdiag} \text{vec}\{\mathbf{D}_1\}$ .
- P<sub>11</sub>.  $d(\text{vec}\{\mathbf{D}_1\}) = \text{vec}\{d(\mathbf{D}_1)\}$ .
- P<sub>12</sub>.  $(\mathbf{D}_1 \otimes \mathbf{D}_2)^T = \mathbf{D}_1^T \otimes \mathbf{D}_2^T$ .
- P<sub>13</sub>.  $(\mathbf{D}_1 \otimes \mathbf{D}_2)^H = \mathbf{D}_1^H \otimes \mathbf{D}_2^H$ .
- P<sub>14</sub>.  $(\mathbf{D}_1 \otimes \mathbf{D}_2)(\mathbf{D}_3 \otimes \mathbf{D}_4) = (\mathbf{D}_1 \mathbf{D}_3 \otimes \mathbf{D}_2 \mathbf{D}_4)$ .

La dérivée seconde de la fonction de coût (Eq. (1)) est définie comme dans [14] par :

$$d^2 C_{ZBD}(\mathbf{B}) = d\mathcal{D}_B(C_{ZBD}(\mathbf{B})) \text{dvec}\{\mathbf{B}\} + d\mathcal{D}_B^*(C_{ZBD}(\mathbf{B})) \text{dvec}\{\mathbf{B}^*\}, \quad (11)$$

où  $\mathcal{D}_B(C_{ZBD}(\mathbf{B}))$  et  $\mathcal{D}_B^*(C_{ZBD}(\mathbf{B}^*))$  sont les matrices jacobiennes de  $C_{ZBD}(\mathbf{B})$ , définies comme dans [14],

$$\begin{aligned} \mathcal{D}_B(C_{ZBD}(\mathbf{B})) &= \left( \text{vec} \left\{ \frac{\partial C_{ZBD}(\mathbf{B})}{\partial \mathbf{B}} \right\} \right)^T, \\ \mathcal{D}_B^*(C_{ZBD}(\mathbf{B})) &= \left( \text{vec} \left\{ \frac{\partial C_{ZBD}(\mathbf{B})}{\partial \mathbf{B}^*} \right\} \right)^T. \end{aligned} \quad (12)$$

Les dérivées partielles  $\frac{\partial C_{ZBD}(\mathbf{B})}{\partial \mathbf{B}}$  and  $\frac{\partial C_{ZBD}(\mathbf{B})}{\partial \mathbf{B}^*}$  de la fonction de coût (Eq. (1)) ont été calculées dans [8] :

$$\frac{\partial C_{ZBD}(\mathbf{B})}{\partial \mathbf{B}} = \sum_{i=1}^{N_m} (\text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i \mathbf{B}^H\})^T \mathbf{B}^* \mathbf{X}_i^* + \sum_{i=1}^{N_m} (\text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i \mathbf{B}^H\})^* \mathbf{B}^* \mathbf{X}_i^T, \quad (13)$$

$$\frac{\partial C_{ZBD}(\mathbf{B})}{\partial \mathbf{B}^*} = \sum_{i=1}^{N_m} (\text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i \mathbf{B}^H\}) \mathbf{B}\mathbf{X}_i^H + \sum_{i=1}^{N_m} (\text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i \mathbf{B}^H\})^H \mathbf{B}\mathbf{X}_i. \quad (14)$$

En utilisant les propriétés ci-dessus (P<sub>1</sub>, P<sub>2</sub>, ..., P<sub>14</sub>) et l'expression des dérivées partielles  $\frac{\partial C_{ZBD}(\mathbf{B})}{\partial \mathbf{B}}$  et  $\frac{\partial C_{ZBD}(\mathbf{B})}{\partial \mathbf{B}^*}$ , la dérivée des matrices jacobiennes vaut alors :

$$\begin{aligned} d\mathcal{D}_B(C_{ZBD}(\mathbf{B})) &= \sum_{i=1}^{N_m} (\text{vec}\{d\mathbf{B}\})^T [(\mathbf{T}_{vec})^T (\mathbf{I}_N \otimes \mathbf{X}_i \mathbf{B}^H) \mathbf{T}_{Bdiag}^T (\mathbf{B}^* \mathbf{X}_i^* \otimes \mathbf{I}_N) \\ &\quad + (\mathbf{T}_{vec})^T (\mathbf{I}_N \otimes \mathbf{X}_i^H \mathbf{B}^H) \mathbf{T}_{Bdiag}^T (\mathbf{B}^* \mathbf{X}_i^T \otimes \mathbf{I}_N)] \\ &\quad + (\text{vec}\{d\mathbf{B}^*\})^T [(\mathbf{X}_i^T \mathbf{B}^T \otimes \mathbf{I}_N) \mathbf{T}_{Bdiag}^T (\mathbf{B}^* \mathbf{X}_i^* \otimes \mathbf{I}_N) \\ &\quad + (\mathbf{X}_i^* \mathbf{B}^T \otimes \mathbf{I}_N) \mathbf{T}_{Bdiag}^T (\mathbf{B}^* \mathbf{X}_i^T \otimes \mathbf{I}_N) \\ &\quad + (\mathbf{X}_i^* \otimes (\text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i \mathbf{B}^H\})) \\ &\quad + (\mathbf{X}_i^T \otimes (\text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i^H \mathbf{B}^H\}))]. \end{aligned} \quad (15)$$

$$\begin{aligned} d\mathcal{D}_B^*(C_{ZBD}(\mathbf{B})) &= \sum_{i=1}^{N_m} [(\mathbf{X}_i^* \mathbf{B}^T \otimes \mathbf{I}_N) \mathbf{T}_{Bdiag} (\mathbf{B}^* \mathbf{X}_i^T \otimes \mathbf{I}_N) \text{vec}\{d\mathbf{B}\} \\ &\quad + (\mathbf{X}_i^* \mathbf{B}^T \otimes \mathbf{I}_N) \mathbf{T}_{Bdiag} (\mathbf{I}_N \otimes \mathbf{B}\mathbf{X}_i) (\mathbf{T}_{vec}) \text{vec}\{d\mathbf{B}^*\} \\ &\quad + (\mathbf{X}_i^T \mathbf{B}^T \otimes \mathbf{I}_N) \mathbf{T}_{Bdiag} (\mathbf{B}^* \mathbf{X}_i^* \otimes \mathbf{I}_N) \text{vec}\{d\mathbf{B}\} \\ &\quad + (\mathbf{X}_i^T \mathbf{B}^T \otimes \mathbf{I}_N) \mathbf{T}_{Bdiag} (\mathbf{I}_N \otimes \mathbf{B}\mathbf{X}_i^H) (\mathbf{T}_{vec}) \text{vec}\{d\mathbf{B}^*\} \\ &\quad + (\mathbf{I}_N \otimes (\text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i \mathbf{B}^H\})) (\mathbf{X}_i^* \otimes \mathbf{I}_N) \text{vec}\{d\mathbf{B}\} \\ &\quad + (\mathbf{I}_N \otimes (\text{Bdiag}_{(n)}\{\mathbf{B}\mathbf{X}_i \mathbf{B}^H\})^H) (\mathbf{X}_i^T \otimes \mathbf{I}_N) \text{vec}\{d\mathbf{B}\}]. \end{aligned} \quad (16)$$

Les Eqs. (7) et (11) conduisent aux résultats présentés au niveau des Eqs. (8) et (9). Enfin, nous pouvons en déduire les quatre matrices Hessiennes complexes sous la forme des Eqs. (5) et (6).

## Références

- [1] A. Belouchrani, K. Abed-Meraïm, M. G. Amin, and A. Zoubir. Joint anti-diagonalization for blind source separation. In *Proc. IEEE ICASSP*, pages 2196–2199, Salt Lake City, UT, May 2001.
- [2] A. Belouchrani, K. Abed-Meraïm, J.-F. Cardoso, and E. Moulines. A blind source separation technique using second order statistics. *IEEE Trans. Signal Proc.*, 45 :434–444, Feb. 1997.
- [3] J.-F. Cardoso and A. Souloumiac. Blind beamforming for non-gaussian signals. *IEE Proc.-F*, 40(6) :362–370, 1993.
- [4] G. Chabriel and J. Barrère. A direct algorithm for nonorthogonal approximate joint diagonalization. *IEEE Trans. Signal Proc.*, 60(1) :39–47, 2012.
- [5] G. Chabriel, J. Barrère, N. Thirion-Moreau, and E. Moreau. Algebraic joint zero-diagonalization and blind source separation. *IEEE Trans. Signal Proc.*, 56(3) :980–989, March 2008.
- [6] G. Chabriel, M. Kleinstueber, E. Moreau, H. Shen, P. Tichavsky, and A. Yeredor. Joint matrices decompositions and blind source separation : A survey of methods, identification, and applications. *IEEE Signal Proc. Mag.*, 31(3) :34–43, 2014.
- [7] O. Cherrak, H. Ghennioui, E.-H. Abarakan, and N. Thirion-Moreau. Non-unitary joint block diagonalization of matrices using a levenberg-marquardt algorithm. In *Proc. EUSIPCO*, pages 1–5, Sept. 2013.
- [8] O. Cherrak, H. Ghennioui, E.-H. Abarakan, N. Thirion-Moreau, and E. Moreau. Non-unitary joint zero-block diagonalization of matrices using a conjugate gradient algorithm. In *Proc. EUSIPCO*, Sept. 2015.
- [9] P. Comon. Independent component analysis, a new concept? *Signal Proc.*, 36 :287–314, 1994.
- [10] L. De Lathauwer, C. Févotte, B. De Moor, and J. Vandewalle. Jacobi algorithm for block diagonalization in blind identification. In *Proc. Symp. Inform. Theory in the Benelux*, pages 155–162, Louvain-la-Neuve, Belgium, May 2002.
- [11] E. M. Fadaïli, N. Thirion-Moreau, and E. Moreau. Non orthogonal zero-diagonalization for sources separation based on time-frequency representation. In *Proc. IEEE ICASSP*, volume V, pages 297–300, Philadelphia, USA, March 2005.
- [12] E. M. Fadaïli, N. Thirion-Moreau, and E. Moreau. Non-orthogonal joint diagonalization/zero-diagonalization for source separation based on time-frequency distributions. *IEEE Trans. Signal Proc.*, 55(5), May 2007.
- [13] H. Ghennioui, N. Thirion-Moreau, E. Moreau, and D. Aboutajdine. Gradient-based joint block diagonalization algorithms : Application to blind separation of fir convolutive mixtures. *Signal Proc.*, 90(6) :1836–1849, June 2010.
- [14] A. Hjørungnes. *Complex-Valued Matrix Derivatives With Applications in Signal Processing and Communications*. Cambridge, University Press, Feb. 2011.
- [15] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Math.*, 11 :431–441, 1963.
- [16] E. Moreau. A generalization of joint-diagonalization criteria for source separation. *IEEE Trans. Signal Proc.*, 49(3) :530–541, March 2001.
- [17] D. Nion. A tensor framework for non-unitary joint block diagonalization. *IEEE Trans. Signal Proc.*, 59(10) :4585–4594, Oct. 2011.
- [18] J. R. Shewchuk. Introduction to the conjugate gradient method without the agonizing pain. Technical report, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, 1994.
- [19] A. Souloumiac. Nonorthogonal joint diagonalization by combining givens and hyperbolic rotations. *IEEE Trans. Signal Proc.*, 57(6) :2222–2231, June 2009.
- [20] A. Yeredor, A. Ziehe, and K.-R. Müller. Approximate joint diagonalization using natural gradient approach. In *Lecture Notes in Computer Science, in Proceedings ICA*, pages 89–96, Granada, Spain, May 2004.
- [21] W. T. Zhang and S. T. Lou. A low complexity iterative algorithm for joint zero-diagonalization. *IEEE Signal Proc. Lett.*, 19(2) :115–118, Jan. 2012.