

# Séparation aveugle de mélange instantanée de sources à l'aide de fonctions séparatrices ajustées

Dinh-Tuan PHAM

Laboratoire de Modélisation et Calcul, IMAG - C.N.R.S.

B.P. 53X, 38041 Grenoble cédex, France

## RÉSUMÉ

Nous proposons un algorithme de séparation aveugle des mélanges instantanés de sources basé sur des fonctions séparatrices qui sont elle-même ajustées à partir des sources venant d'être reconstruites. Nous montrons la stabilité de l'algorithme au voisinage de la vraie solution et, dans quelque cas simples, son instabilité au voisinage de la solution parasite. Cette bonne propriété de l'algorithme est confirmé dans nos simulations pour des situations plus complexes.

## ABSTRACT

We propose an algorithm for the blind separation of instantaneous mixture of independent sources, based on the use of separating functions which are themselves adapted to the just reconstructed sources. We prove the stability of our algorithm in the neighborhood of the true solution and, in some simple cases, its instability in the neighborhood of the spurious solution. This good property of the algorithm is confirmed in simulations in more complex situations.

## 1 Introduction

Le problème de séparation aveugle de sources a attiré beaucoup d'attention récemment. Nous nous intéressons ici au cas simple de mélanges instantanées. Parmi de nombreuses méthodes proposées dans la littérature ([1], [3] – [8], ...), une classe particulièrement intéressante est basée sur l'utilisation de fonctions séparatrices, introduite d'abord dans [4] et puis modifiée et analysée dans [7, 8]. L'intérêt de ce type de méthode est sa simplicité et son flexibilité qui permet d'adapter les fonctions séparatrices à la distribution des sources. Toutefois, comme cette dernière est inconnue, on est souvent amené à choisir les fonctions séparatrices a priori, plus ou moins arbitrairement.

Mais un problème plus grave de la méthode des fonctions séparatrices est la non unicité possible de la solution des équations de séparation. On a montré [4] que dans le cas de 2 sources avec fonction séparatrice cubique, ces équations admettent toujours deux solutions distinctes *modulo une permutation et un changement d'échelle*. De plus, l'algorithme adaptatif de Héroult-Jutten [4] dans ce cas convergerait vers la bonne solution si les sources sont sous-gaussienne mais vers une solution "parasite" dans le cas contraire.

Dans ce papier, nous proposons d'ajuster les fonctions séparatrices selon la distribution des sources. Cela permet, non seulement d'améliorer la performance de la méthode mais aussi, comme on verra plus tard, d'éviter le problème de non unicité mentionnée plus haut. Comme la méthode est optimale si chaque fonction séparatrice est prise proportionnelle à la dérivée logarithmique de la densité d'une source ([7, 8]), l'idée naturelle est d'estimer ces fonctions. Mais, ne disposant a priori d'aucun modèle de source, l'estimation doit être non paramétrique, ce qui demanderait beaucoup de données et une charge de calcul très lourde. Pour cette raison,

nous abandonnons cette idée et nous contentons à ajuster les fonctions séparatrices dans une classe de fonctions générées linéairement à partir des fonctions de base données a priori. Cette méthode a été mentionnée dans [7, 8] sans que sa mise en œuvre soit précisée et étudiée.

## 2 Description de l'algorithme

On dispose d'une suite d'observations vectorielles  $\mathbf{X}(t)$ , chaque composante correspond à la sortie d'un capteur, qui enregistre une combinaison linéaire de  $K$  sources  $S_1(t), \dots, S_K(t)$ , soit  $\mathbf{X}(t) = \mathbf{A}\mathbf{S}(t)$  où  $\mathbf{S}(t)$  est le vecteur des sources. Pour simplifier, on suppose que le nombre de capteurs est aussi  $K$  et donc la matrice de mélange  $\mathbf{A}$  est carrée. Le problème est de reconstruire les sources à partir des observations, c'est à dire de chercher une matrice  $\mathbf{B}$  telle que les composantes de  $\mathbf{B}\mathbf{X}$  redonnent les sources à *une permutation et une facteur d'échelle près*.

La méthode des fonctions séparatrices est basée sur les équations d'estimation (dite "de séparation") suivantes :

$$E\psi_i[(\mathbf{B}\mathbf{X})_i](\mathbf{B}\mathbf{X})_j] = 0, \quad i \neq j = 1, \dots, K, \quad (1)$$

où  $E$  est l'opérateur d'espérance,  $(\mathbf{B}\mathbf{X})_i$  désigne la  $i$ -ème composante de  $\mathbf{B}\mathbf{X}$  et  $\psi_i$  sont des fonctions "séparatrices". Dans un algorithme de traitement par lot, on remplacera les équations (1) par leur analogues où  $E$  est remplacé par l'opérateur de moyenne empirique  $\hat{E}$ . Dans un algorithme adaptatif, les équations (1) seront résolues par un algorithme stochastique.

Ici, les fonctions séparatrices sont ajustées en même temps que la séparation. La méthode de projection [7, 8] permet de les choisir de façon optimale dans la classe des combinaisons linéaires des fonctions de base données  $\phi_1, \dots, \phi_m$ . En écrivant  $\psi_i = \sum_{k=1}^m c_{ik}\phi_k$ , le choix optimal des coefficients  $c_{ik}$  est

donné par la solution de

$$[c_{i1} \ \cdots \ c_{im}] \mathbf{G}_i = \mathbf{g}_i, \quad i = 1, \dots, K, \quad (2)$$

où  $\mathbf{G}_i = E\phi[(\mathbf{B}\mathbf{X})_i] \phi[(\mathbf{B}\mathbf{X})_i]^T$ ,  $\mathbf{g}_i = E\phi'[(\mathbf{B}\mathbf{X})_i]$ ,  $\phi$  étant le vecteur  $[\phi_1 \ \cdots \ \phi_m]$ ,  $\phi'$  sa dérivée et  $^T$  désignant la transposition.

Un choix possible de fonction de base est la fonction puissance  $s \mapsto \text{signe}(s)|s|^\alpha$ ,  $\alpha > 0$ . Dans tous les cas, une des fonctions de base doit être la fonction identité.

En général, on doit coupler (1) et (2) avec une normalisation pour que les sources ont une variance compatible avec les fonctions  $\phi_i$  choisies, car ces dernières peuvent se comporter de façon très différente suivant que la source est de faible ou grande amplitude. (C'est la cas, par ex., de la fonction arctan.) Le seul cas où la normalisation n'est pas nécessaire est celui où les fonctions séparatrices sont des fonctions puissance, car alors l'effet d'échelle sera absorbé par les coefficients  $c_{ik}$ .

## 2.1 Algorithme en traitement par lot

On résout les équations (1) et (2) avec  $\hat{\mathbf{E}}$  à la place de  $\mathbf{E}$ . On note que (2) a une solution explicite pour tout  $\mathbf{B}$  fixé. Par contre (1) ne peut être résolu à  $\psi_i$  fixée, que par une méthode itérative. Dans le cas où les  $\psi_i$  sont choisies a priori, une méthode de type Newton Rhapsod est la suivante. On part d'une estimation courante  $\hat{\mathbf{B}}$  de  $\mathbf{B}$ , ce qui permet de reconstruire les sources  $\hat{S}_i = (\hat{\mathbf{B}}\mathbf{X})_i$ . On calcule la nouvelle estimation de  $\mathbf{B}$  par  $(\mathbf{I} - \Delta)\hat{\mathbf{B}}$  où  $\Delta$  est la matrice de terme diagonal nul et terme hors diagonal  $\delta_{ij}$ ,  $i \neq j$  solution de

$$\begin{bmatrix} \hat{\omega}_i \hat{\sigma}_j^2 & \hat{\gamma}_i \\ \hat{\gamma}_j & \hat{\omega}_j \hat{\sigma}_i^2 \end{bmatrix} \begin{bmatrix} \delta_{ij} \\ \delta_{ji} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{E}}\psi_i(\hat{S}_i)\hat{S}_j \\ \hat{\mathbf{E}}\psi_j(\hat{S}_j)\hat{S}_i \end{bmatrix}$$

avec  $\hat{\gamma}_i = \hat{\mathbf{E}}\psi_i(\hat{S}_i)\hat{S}_i$ ,  $\hat{\sigma}_i^2 = \hat{\mathbf{E}}\hat{S}_i^2$  et  $\hat{\omega}_i = \hat{\mathbf{E}}\psi_i'(\hat{S}_i)$ .

Souvent, on normalise les fonctions  $\psi_i$  de sorte que  $\hat{\gamma}_i = 1$ . Si les sources sont aussi normalisées pour que  $\hat{\sigma}_i = 1$ , le système précédent a pour solution explicite

$$\delta_{ij} = \frac{\hat{\omega}_j \hat{\mathbf{E}}\psi_i(\hat{S}_i)\hat{S}_j - \hat{\mathbf{E}}\psi_j(\hat{S}_j)\hat{S}_i}{\hat{\omega}_i \hat{\omega}_j - 1} \quad (3)$$

Dans le cas où les  $\psi_i$  sont ajustées, avant d'effectuer le calcul (3), on calcule les coefficients  $c_{ik}$  en résolvant les systèmes linéaires analogues à (2)

$$[c_{i1} \ \cdots \ c_{im}] \hat{\mathbf{G}}_i = \hat{\mathbf{g}}_i, \quad i = 1, \dots, K, \quad (4)$$

où  $\hat{\mathbf{G}}_i = \hat{\mathbf{E}}\phi(\hat{S}_i)\phi(\hat{S}_i)^T$ ,  $\hat{\mathbf{g}}_i = \hat{\mathbf{E}}\phi'(\hat{S}_i)$ . On calcule ensuite  $\psi_i(\hat{S}_i)$  par  $\sum_{k=1}^m c_{ik}\phi_k(\hat{S}_i)$ . On note que (4) implique alors  $\hat{\mathbf{E}}\psi_i(\hat{S}_i)\hat{S}_i = 1$  et  $\hat{\mathbf{E}}\psi_i'(\hat{S}_i) = \hat{\mathbf{E}}\psi_i^2(\hat{S}_i)$ . Ce qui donne  $\hat{\gamma}_i = 1$  et  $\hat{\omega}_i = \hat{\mathbf{E}}\psi_i^2(\hat{S}_i) > 0$  (d'après l'inégalité de Schawrtz).

À l'étape suivant, les sources seront reconstituées à partir de la nouvelle matrice  $(\mathbf{I} - \Delta)\hat{\mathbf{B}}$ . Ce qui revient simplement à remplacer  $\hat{S}_i$  par  $\hat{S}_i - \sum_{j \neq i} \delta_{ij}\hat{S}_j$ . Ces dernières ne sont plus normalisées, il faut les renormaliser avant d'effectuer les calculs (3) et (4).

Il est intéressant de noter que l'algorithme peut être réalisé uniquement à partir des sources reconstituées, sans aucune référence à la matrice de reconstruction  $\hat{\mathbf{B}}$ .

## 2.2 Algorithme en ligne (adaptatif)

Comme (2) a une solution explicite, une première approche est de faire la mise à jour directement les  $\mathbf{G}_i$  et  $\mathbf{g}_i$ . On suppose disposant, à l'instant  $t$ , d'une estimation  $\mathbf{B}(t)$  de  $\mathbf{B}$ , ce qui permet de reconstruire les sources (en cet instant) par  $\hat{S}_i(t) = i$ -ème composante de  $\mathbf{B}(t)\mathbf{X}(t)$ . On met à jour les matrices  $\mathbf{G}_i(t)$  et vecteurs  $\mathbf{g}_i(t)$  par

$$\begin{aligned} \mathbf{G}_i(t) &= \rho \mathbf{G}_i(t-1) + (1-\rho)\phi[\hat{S}_i(t)]\phi[\hat{S}_i(t)]^T \\ \mathbf{g}_i(t) &= \rho \mathbf{g}_i(t-1) + (1-\rho)\phi'[\hat{S}_i(t)] \end{aligned} \quad (5)$$

où  $\rho$  est un facteur oubli. On résout alors les systèmes  $[c_{i1}(t) \ \cdots \ c_{im}(t)]\mathbf{G}_i(t) = \mathbf{g}_i(t)$ . On calcule ensuite  $\psi_i[\hat{S}_i(t)] = \sum_{k=1}^m c_{ik}(t)\phi_k[\hat{S}_i(t)]$  et on détermine les coefficients  $\delta_{ij}(t)$  de façon semblable à (3). En supposant que les sources sont déjà normalisées pour avoir la variance unité, on a

$$\delta_{ij}(t) = (1-\rho) \frac{\hat{\omega}_j(t)\psi_i[\hat{S}_i(t)]\hat{S}_j(t) - \psi_j[\hat{S}_j(t)]}{\hat{\omega}_i(t)\hat{\omega}_j(t) - 1}$$

où  $\hat{\omega}_i(t) = \mathbf{g}_i(t)[c_{i1}(t) \ \cdots \ c_{im}(t)]^T$ . Le nouveau estimateur pour  $\mathbf{B}$  sera comme auparavant obtenu en pré-multipliant  $\mathbf{B}(t)$  par la matrice de terme diagonal 1 et hors diagonal  $-\delta_{ij}(t)$ . Mais celui-ci, noté  $\tilde{\mathbf{B}}(t+1)$ , produira des sources non normalisées. Si la  $i$ -ème source reconstituée par  $\mathbf{B}(t)$  a pour variance 1, celle reconstituée par  $\tilde{\mathbf{B}}(t+1)$  aura pour variance estimée en  $t+1$

$$d_i^2(t+1) = \rho + (1-\rho)\tilde{S}_i(t+1)^2$$

où  $\tilde{S}_i(t+1)$  est la  $i$ -ème composante de  $\tilde{\mathbf{B}}(t+1)\mathbf{X}(t+1)$ . Il faut donc diviser les lignes de  $\tilde{\mathbf{B}}(t+1)$  par les  $d_i(t+1)$  pour obtenir le nouveau estimateur "normalisé"  $\mathbf{B}(t+1)$ . On recommence alors les calculs à l'instant  $t+1$  avec celui-ci, ce qui revient à prendre  $\hat{S}_i(t+1) = \tilde{S}_i(t+1)/d_i(t+1)$ .

Comme une des fonctions  $\phi_k$ , le premier par exemple, est la fonction identité, on peut s'attendre à ce que les matrices  $\mathbf{G}_i$  ont pour l'élément (1,1) égal à 1 si les sources sont normalisées. Mais ce n'est pas le cas si on s'en tient à (5). Par suite, pour la mise à jour de la 1ère colonne et ligne de  $\mathbf{G}_i$ , on adopte une formule légèrement différente :

$$\begin{aligned} \mathbf{G}_{i,11}(t+1) &= [\rho \mathbf{G}_{i,11}(t) + (1-\rho)\tilde{S}_i(t)^2]/d_i^2(t+1), \\ \mathbf{G}_{i,k1}(t+1) &= \{\rho \mathbf{G}_{i,k1}(t) + (1-\rho)\phi_k[\hat{S}_i(t+1)]\tilde{S}_i(t)\}/d_i(t+1) \end{aligned}$$

et  $\mathbf{G}_{i,k1}(t+1) = \mathbf{G}_{i,1k}(t+1)$ ,  $\mathbf{G}_{i,kl}$  désignant l'élément  $(k,l)$  de  $\mathbf{G}_i$ . De cette façon,  $\mathbf{G}_{i,11}(t+1) = 1$  si  $\mathbf{G}_{i,11}(t) = 1$  et la matrice  $\mathbf{G}_i$  reste définie positive. Ce qui assure que  $\hat{\omega}_i(t) > 1$ .

L'algorithme précédent s'apparente à l'algorithme RLS (recursive least mean square). Il est assez coûteux en terme de calcul. Une version plus simple de type RMS (least mean squares) peut être envisagée. On mets à jour directement les  $c_{ik}(t)$  par

$$\begin{aligned} [c_{i1}(t) \ \cdots \ c_{im}(t)] &= [c_{i1}(t-1) \ \cdots \ c_{im}(t-1)] \\ &+ (1-\rho) \left\{ \phi'[\hat{S}_i(t)] - \phi[\hat{S}_i(t)] \sum_{k=1}^m c_{ik}(t-1)\phi_k[\hat{S}_i(t)] \right\} \Gamma \end{aligned}$$

où  $\Gamma$  est une matrice donnée a priori. L'intérêt de cette matrice est d'accélérer la convergence de l'algorithme.

L'idéal est d'avoir  $\Gamma = \mathbf{G}_i^{-1}$  mais comme on ne connaît pas la loi de source, on se propose de prendre  $\Gamma = [\int \phi(s)\phi(s)^T \exp(-s^2/2)ds/\sqrt{2\pi}]^{-1}$  (les sources étant supposée centrée et normalisée pour avoir la variance unité). Notons que les  $\hat{\omega}_i(t)$  doivent maintenant être mis à jour (de façon semblable qu'un élément des  $\mathbf{G}_i$ ).

### 3 Non unicité et stabilité

Nous montrons, dans une situation simple, la non unicité de solution de (1) et (2). Soit  $K = 2$  et supposons que les sources  $S_1$  et  $S_2$  sont de même loi contrée et que les fonctions  $\phi_k$  sont impaires. Alors les matrices  $\mathbf{A}^{-1}$  et  $\mathbf{JA}^{-1}$ , où  $\mathbf{J} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$ , sont toutes solutions de (1) et (2). Pour la première matrice, c'est évident (et elle est la "bonne"). Pour la deuxième, on note que elle reconstitue  $S_1^* = S_1 + S_2$  et  $S_2^* = S_1 - S_2$ . Or, la loi du couple  $(S_1, S_2)$  étant la même que celle de  $(S_2, S_1)$ , on a

$$\begin{aligned} E[\psi(S_1 + S_2)S_1] &= E[\psi(S_2 + S_1)S_2], \\ E[\psi(S_2 - S_1)S_1] &= E[\psi(S_1 - S_2)S_2] \end{aligned}$$

quelque soit la fonction  $\psi$ . Si cette dernière est impaire, la dernière membre de droite vaut  $-E[\psi(S_2 - S_1)S_2]$  et alors  $E[\psi(S_1^*)S_2^*] = 0 = E[\psi(S_2^*)S_1^*]$ . Comme les  $\phi_k$  sont impaires, quelque soit le choix de  $c_{1k}, c_{2k}$ , que l'on peut toujours faire pour satisfaire (2), les fonctions  $\psi_1$  et  $\psi_2$  sont aussi impaires et donc (1) est satisfait.

Ce qui précède illustre le problème de non unicité de solutions dans les méthodes des fonctions séparatrices. Toutefois, un algorithme de recherche des solutions de (1) et (2) ne peut converger que vers des points limites stables. Donc ce problème disparaît si ces dernières se réduisent à la seule "bonne" solution.

Notre algorithme en traitement par lot peut être vu asymptotiquement comme une itération du point fixe. Celle-ci est  $\mathbf{B} \mapsto (\mathbf{I} - \Delta)\mathbf{B}$  où  $\Delta$  est la matrice de terme diagonal nul et terme hors diagonal  $\delta_{ij}$ , solution de

$$\begin{bmatrix} \omega_i \sigma_j^2 & \gamma_i \\ \gamma_j & \omega_j \sigma_i^2 \end{bmatrix} \begin{bmatrix} \delta_{ij} \\ \delta_{ji} \end{bmatrix} = \begin{bmatrix} E\psi_i(\hat{S}_i)\hat{S}_j \\ E\psi_j(\hat{S}_j)\hat{S}_i \end{bmatrix}$$

avec  $\hat{S}_i = (\mathbf{B}\mathbf{X})_i$ ,  $\psi_i = \sum_{k=1}^m c_{ik}\phi_k$ ,  $\gamma_i = E[\psi_i(\hat{S}_i)\hat{S}_i]$ ,  $\omega_i = E[\psi_i'(\hat{S}_i)]$ ,  $\sigma_i^2 = E(\hat{S}_i^2)$ , les  $c_{ik}$  étant solution de (2). La stabilité locale du point fixe dépend du gradient relatif (voir [2]) de l'itération. Elle est réalisée si et seulement si cette dernière a toutes ses valeurs propres de module inférieur à 1.

**Lemme 3.1** — *Le gradient relatif de l'itération précédente en un point fixe  $\mathbf{B}$  est la matrice diagonale par bloc, de blocs diagonaux  $2 \times 2$  :  $\mathbf{I} - \mathbf{H}_{ij}$ ,  $i \neq j$ , où*

$$\mathbf{H}_{ij} = \begin{bmatrix} \omega_i \sigma_j^2 & \gamma_i \\ \gamma_j & \omega_j \sigma_i^2 \end{bmatrix}^{-1} \left\{ \begin{bmatrix} E[\psi_i'(\hat{S}_i)\hat{S}_j^2] & E[\psi_i(\hat{S}_i)\hat{S}_j] \\ E[\psi_j(\hat{S}_j)\hat{S}_i] & E[\psi_j'(\hat{S}_j)\hat{S}_i^2] \end{bmatrix} + \begin{bmatrix} E\frac{\partial \psi_i}{\partial \delta_{ij}}(\hat{S}_i)\hat{S}_j & E\frac{\partial \psi_i}{\partial \delta_{ji}}(\hat{S}_i)\hat{S}_j \\ E\frac{\partial \psi_j}{\partial \delta_{ij}}(\hat{S}_j)\hat{S}_i & E\frac{\partial \psi_j}{\partial \delta_{ji}}(\hat{S}_j)\hat{S}_i \end{bmatrix} \right\}.$$

*Note* : Les  $\gamma_i, \sigma_i^2, \omega_i$  et ainsi  $\mathbf{H}_{ij}$  dépendent implicitement du point considéré  $\mathbf{B}$ , via les  $\hat{S}_i$ , mais nous ne l'explicitons pas cette dépendance, pour ne pas alourdir la notation.

Au point fixe  $\mathbf{A}^{-1}$ ,  $\hat{S}_i = S_i$  (la vraie source) et donc la dernière matrice dans le Lemme est nulle dès que les sources sont centrées. Par conséquent, le gradient relatif se réduit à  $\mathbf{0}$  dans ce cas.

Dans le cas de deux sources de même loi et où les fonctions  $\phi_k$  sont impaires, nous savons qu'il y ait une autre point fixe "parasite"  $\mathbf{JA}^{-1}$ . Pour ce point fixe,  $\hat{S}_1 = S_1^*$ ,  $\hat{S}_2 = S_2^*$  et la dernière matrice dans le Lemme est encore nulle, d'après les mêmes calculs que ceux du début de ce paragraphe. Mais la matrice du gradient relatif ne se réduit pas à  $\mathbf{0}$  mais à

$$-\begin{bmatrix} \omega_1 \sigma_2^2 & \gamma_1 \\ \gamma_2 & \omega_2 \sigma_1^2 \end{bmatrix}^{-1} \begin{bmatrix} \text{cov}\{\psi_1'(S_1^*), S_2^{*2}\} & 0 \\ 0 & \text{cov}\{\psi_1'(S_1^*), S_2^{*2}\} \end{bmatrix}$$

Rappelons qu'ici  $\gamma_i = E[\psi_i(S_i^*)S_i^*]$ ,  $\omega_i = E\psi_i'(S_i^*)$ ,  $\sigma_i^2 = E(\hat{S}_i^*)^2$ ,  $i = 1, 2$ . Si en plus les sources sont de loi symétrique, de sorte que  $S_1^*$  et  $S_2^*$  ont la même loi,  $\psi_1$  sera la même que  $\psi_2$  (car les  $c_{1k} = c_{2,k}$ ,  $\forall k$ ), alors la matrice précédente se réduit à

$$-\begin{bmatrix} 1 & \lambda \\ \lambda & 1 \end{bmatrix}^{-1} \tilde{\lambda} \text{ où}$$

$$\lambda = \frac{E[\psi(S_1^*)S_2^*]}{E\psi'(S_1^*)ES_1^{*2}} = \frac{E[\psi(S_2^*)S_1^*]}{E\psi'(S_2^*)ES_2^{*2}}$$

$$\tilde{\lambda} = \frac{\text{cov}\{\psi'(S_1^*), S_2^{*2}\}}{E\psi'(S_1^*)ES_2^{*2}} = \frac{\text{cov}\{\psi'(S_2^*), S_1^{*2}\}}{E\psi'(S_2^*)ES_1^{*2}},$$

$\psi$  étant la fonction commune  $\psi_1$  ou  $\psi_2$ . Les valeurs propres de cette matrices sont  $-\tilde{\lambda}/(1 \pm \lambda)$  et donc l'itération du point fixe en le point  $\mathbf{JA}^{-1}$  est instable ces valeurs propres ne sont pas toutes strictement inférieures à 1.

Considérons le cas simple où  $\psi(s)$  est choisi de la forme  $c_1 + c_2 s^3$ . Un calcul explicite est alors possible, menant à

$$-\frac{\tilde{\lambda}}{1 - \lambda} = 3, \quad -\frac{\tilde{\lambda}}{1 + \lambda} = \frac{3(\mu_4 - 3)^2}{2\mu_6 + 6\mu_4 - (\mu_4 - 3)^2}$$

où  $\mu_4 = ES_i^{*4}/\sigma_i^{*4} - 3$ ,  $\mu_6 = ES_i^{*6}/\sigma_i^{*6} - 15\mu_4$ ,  $\sigma_i^*$  étant l'écart-type de  $S_i^*$ . Donc le point fixe  $\mathbf{JA}^{-1}$  est instable *quelque soit la loi de source*.

Considérons maintenant le cas de l'algorithme adaptatif. Celui-ci, dans sa version RLS, peut être vu comme un algorithme stochastique dont les équations d'équilibre sont (1) avec les  $c_{ik}$  exprimés en fonction de  $\mathbf{B}$  à l'aide de (2). On a

**Lemme 3.2** — *Le gradient relatif associé aux équations d'équilibre précédentes en un point équilibre  $\mathbf{B}$  est la matrice diagonale par bloc, de bloc diagonaux  $2 \times 2$  :  $\mathbf{H}_{ij}$ ,  $i \neq j$ .*

Comme précédemment, en le point d'équilibre  $\mathbf{A}^{-1}$ , la matrice de gradient relatif se réduit à la matrice identité. Ce point est donc stable. Par contre, dans les cas de 2 sources de même loi symétrique et où les fonction séparatrices sont choisies dans la classe des fonctions  $\{s \mapsto c_1 s + c_2 s^3\}$ , ce gradient en le point  $\mathbf{JA}^{-1}$ , a pour valeurs propres  $-2$  et  $2[\mu_6 + 3\mu_4 + (\mu_4 - 3)^2]/[2\mu_6 + 6\mu_4 - (\mu_4 - 3)^2]$ . Ce point est donc instable, quelque soit la loi de source.

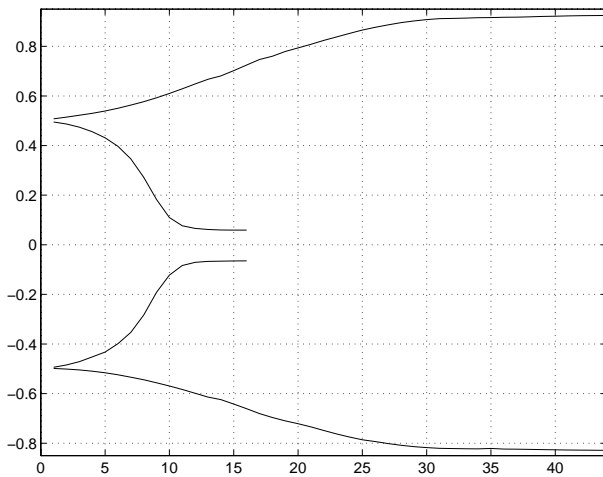


FIG. 1 — Évolution de  $B_{12}/(|B_{11}| + |B_{12}|)$  et  $B_{21}/(|B_{22}| + |B_{21}|)$  en l'algorithme en traitement par lot dans le cas de deux sources de même loi uniforme (courbes au milieu) et exponentielle (courbes extrêmes) en fonction du nombre d'itérations.

Pour ce qui est de la version LMS de l'algorithme adaptatif, on peut voir qu'il y ait un découplage entre la mise à jour de  $\mathbf{B}$  et de  $c_{ik}$  et par suite on a encore les mêmes résultats de stabilité.

## 4 Simulation

Pour vérifier si notre algorithme permet d'éviter la solution parasite, on se place précisément dans le cas de non unicité de solution décrit dans le paragraphe 3 : 2 sources de même loi avec fonctions séparatrices impaires. Comme on peut voir facilement, le comportement de l'algorithme reste inchaogé en si on l'applique aux sources pures et change en même temps le point d'initialisation de  $\mathbf{B}$  en  $\mathbf{BA}$ . Nous simulons donc des sources pures et nous initialisons l'algorithme de façon la plus défavorable, en partant de la solution parasite  $\mathbf{J}$ . Comme loi des sources, nous considérons deux cas extrêmes : la loi uniforme sur  $[-.5, .5]$  qui est fortement sous-gaussienne et la loi exponentielle bilatérale qui est fortement sur-gaussienne. Les fonctions de base sont  $\phi_1(s) = s$ ,  $\phi_2(s) = s^3$  et  $\phi_3(s) = \sqrt{s}$ .

La figure 1 montre bien que l'algorithme converge dans les 2 cas vers la bonne solution quoiqu'il est initialisé à la solution parasite  $\mathbf{J}$  (dans la légende des figures  $B_{ij}$  désignant le terme général de  $\mathbf{B}$ ). On voit que la convergence est beaucoup plus rapide dans le cas uniforme que dans le cas exponentielle. L'estimateur final est aussi plus précis dans le premier cas que dans le second (où les sources sont permutées) :  $B_{12}/(|B_{11}| + |B_{12}|) = .0593$  contre  $.9245$  et  $B_{21}/(|B_{22}| + |B_{21}|) = -.065$  contre  $-.8286$ . Le nombre de donnée est ici 200.

La figure 2 montre la convergence de l'algorithme adaptatif vers la bonne solution même en partant de la solution parasite  $\mathbf{J}$ . Ici encore, la convergence est plus rapide dans le cas uniforme que dans le cas exponentielle. Le facteur oubli est ici .99.

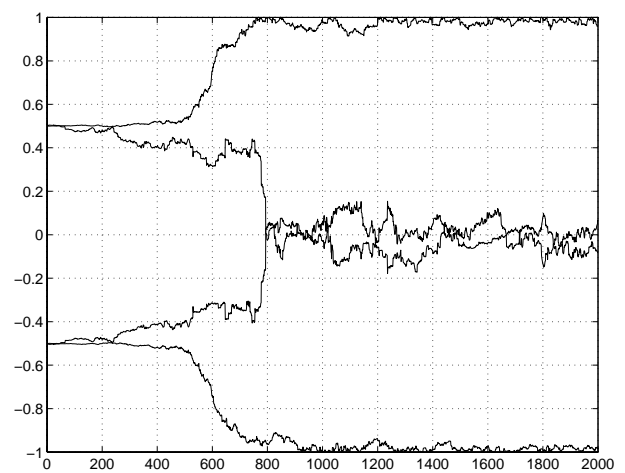


FIG. 2 — Évolution de  $B_{12}/(|B_{11}| + |B_{12}|)$  et  $B_{21}/(|B_{22}| + |B_{21}|)$  en l'algorithme adaptatif dans le cas de deux sources de même loi uniforme (courbes extrêmes) et exponentielle (courbes au milieu) en fonction du temps.

## 5 Conclusion

L'algorithme présenté ici a deux avantages par rapport à un algorithme basé sur des fonctions séparatrices fixées a priori. D'une part il s'adapte à la loi inconnue des sources. D'autre part, il peut éviter le problème de solution parasite.

## Références

- [1] Cardoso, J.-F. "Iterative technique for blind source separation using only fourth order cumulants", *Proc. EUSIPCO 92*, Brussels, Aug. 1992, **2**, pp. 739-742.
- [2] Cardoso, J.-F., Laheld, B. "Equivariant adaptive source separation", *IEEE Trans. on S.P.*, 1996, **44**, 12, 3017-3030.
- [3] Gaéta, M., Lacoume, J.-L. "Source separation without a priori knowledge : the maximum likelihood solution", *Proceedings of EUSIPCO 90*, Barcelona, Spain, 1990, pp. 621-624.
- [4] Jutten, C. Héroult, J. "Blind separation of sources, Part I : an adaptative algorithm based on neuromimetic structure", *Signal Processing* **24**, 1991, pp.1-10.
- [5] Loubaton, Ph., Delfosse, N. "Séparation adaptative de sources indépendantes par une approche de déflation", *Proc. Colloque GRETSI 93*, Juan-les-Pins, France, Sept. 1993, **1**, pp. 325-328.
- [6] Moreau, E., Macchi, O. "New self-adaptive algorithm for source separation based on contrast functions", *Proc. IEEE SP Workshop on Higher-Order Statistics*, Lake Tahoe, U.S.A., 1993, pp. 215-219.
- [7] Pham, D. T., Garat, Ph., Jutten, C. "Separation of a mixture of independent sources through a maximum likelihood approach", *Proc. EUSIPCO 92*, Brussels, Aug. 1992, **2**, pp. 771-774.
- [8] Pham, D. T., Garat, Ph. "Blind Separation of Mixture of Independent Sources Through a Quasi-Maximum Likelihood Approach". À paraître dans *IEEE Trans. S.P.*, 1997, **45**.