



**HAL**  
open science

# Optimisation de fonction de contraste en séparation de sources

Benoît Stoll

► **To cite this version:**

Benoît Stoll. Optimisation de fonction de contraste en séparation de sources. Traitement du signal et de l'image [eess.SP]. Université de Toulon et du Var, 2000. Français. NNT : . tel-03792247

**HAL Id: tel-03792247**

**<https://hal.science/tel-03792247v1>**

Submitted on 29 Sep 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE TOULON ET DU VAR

U.F.R. Sciences et Techniques

THESE

pour obtenir le grade de

Docteur de l'Université de Toulon et du Var

Discipline : Optique, Image et Signal

présentée et soutenue publiquement

par

Benoît Stoll

le 19/01/2000

**Optimisation de  
Fonctions de Contraste  
en Séparation de Sources**

JURY :

M. P.-Y. Arquès, Président

M. G. Alengrin, Rapporteur

M. C. Jutten, Rapporteur

M. J.-C. Pesquet, Examineur

M. J.-F. Cavassilas, Directeur de thèse

M. E. Moreau, Examineur

Ne crains pas d'avancer lentement, crains seulement de t'arrêter.

Sagesse Chinoise

# Résumé

La Séparation de Sources ou Analyse en Composantes Indépendantes tente de trouver une réponse au problème suivant : Un jeu de  $M$  signaux sources sont émis dans un milieu de propagation quelconque, ils sont reçus par un réseau de  $N$  capteurs, ces capteurs nous retranscrivent un jeu de  $N$  signaux appelés observations. Ceux ci correspondent à des mélanges des  $M$  signaux originels. Comment retrouver les  $M$  signaux sources à partir de la seule connaissance des observations sans avoir de connaissance aucune sur le milieu de propagation ni sur la façon dont les signaux se sont mélangés. La seule hypothèse faite sur les sources est leur indépendance statistique mutuelle, hypothèse forte en théorie, mais très vraisemblable en pratique.

Il existe plusieurs méthodes de séparation de sources, elles sont pour la plupart basées sur les statistiques d'ordre supérieurs, qui permettent d'exploiter l'hypothèse d'indépendance des sources. Nous avons considéré le cas de la séparation de sources basée sur l'optimisation de fonctions de contrastes dans un cas de mélange linéaire spatial instantané.

Après avoir introduit le problème de la séparation de sources, les fonctions de contraste et présenté trois algorithmes classiques, le chapitre 2 suivant est consacré à une généralisation de l'algorithme **ICA**. Tout d'abord, nous proposons deux nouvelles familles de fonctions de contrastes regroupant entre autre, les contrastes à la base des algorithmes **ICA** [COM94], **JADE** [CAR93], **STOTD** [DEL96]. Ensuite nous déterminons la solution optimale dans le cas deux sources pour ces deux familles de contrastes. Cela nous permet de proposer deux algorithmes basés sur ces contrastes généralisés et utilisant une méthode de Jacobi, nous présenterons quelques simulations numériques des algorithmes ainsi obtenus.

Lorsque la plupart des techniques de séparation de sources sont basées sur un traitement en deux étapes, un blanchiment spatial des observations puis l'optimisation d'une fonctionnelle de contraste sous contrainte d'orthogonalité de la matrice de séparation, nous essayons de développer des algorithmes effectuant ces deux opérations en une seule étape : l'optimisation d'un contraste sous contrainte de blanchiment des signaux estimés. Le chapitre 3 suivant est donc consacré à des méthodes directes d'optimisation sous contrainte des fonctions de contrastes. Nous étudions deux familles de méthodes permettant d'effectuer cette opération, les méthodes duales et les méthodes directes. Cela nous permet de développer des algorithmes utilisant deux méthodes Lagrangiennes (**LAGR** et **LAGRAUG**), deux méthodes par pénalisation (**FROBR** et **KULLR**) et une méthode directe par projection sur la contrainte (**OSIBS**) que nous comparerons à l'algorithme **EASI**.

# Remerciements

J'ai effectué ma thèse au laboratoire Modélisation et Signal (MS) de l'institut des Sciences de l'Ingénieur de Toulon et du Var (ISITV) à l'Université de Toulon et du Var. Je remercie Monsieur Pierre-Yves Arquès, Directeur du laboratoire MS pour m'avoir accueilli. Je tiens particulièrement à remercier Monsieur Jean-François Cavassilas pour avoir accepté de diriger mes recherches et Monsieur Eric Moreau pour m'avoir guidé, encadré et aidé durant ces années. Je dois aussi remercier Monsieur Christian Jutten et Monsieur Gérard Alengrin de m'avoir fait l'honneur d'être rapporteurs de cette thèse, ainsi que Monsieur Jean-Christophe Pesquet pour avoir accepté de faire parti de mon jury.

Je tiens aussi à remercier tous les collègues du laboratoire qui ont bien voulu m'intégrer au sein de leur équipe. monsieur Bonnet, Jean Barrère, Bernard Xerri, Bénédicte Garbail, William Puech et Vincent Ricordel, Gilles Chabriel, Nadège Thirion-Moreau, Bruno Borloz, Philippe Courmontagne et bien sûr mes collègues thésards qui ont traversés avec moi ces années : Sébastien Paris, Christophe De Luigi. Monsieur Majastre qui a suivi mon parcours depuis l'IUT. Je voudrais citer aussi les thésards et enseignants des autres laboratoires de l'ISITV, Patrick Houbaux, Sandra Rouy, Jacques Schneider du laboratoire Modélisation Numérique et Couplage de l'ISITV, Karine Sellegri, Magalie Ferez, Laurent Massouh du Laboratoire d'Etude des Echanges Particulaires aux Interfaces de l'ISITV. Une pensée pour tous mes amis, Jean-Louis, Ariane, Erwan, Thomas, Julien, Sébastien, Buck et Martine bien sûr, sans qui cette thèse n'aurais jamais débutée. Une pensée tendre pour celle qui m'a chaviré le coeur. Merci à ceux que j'oublie ici mais qui m'ont apporté leur soutien et leur confiance. Je n'oublierais pas d'avoir une pensée émue pour ma grand mère Carmen qui aurait été très fière de moi. Je tiens enfin à associer ma famille du bout

du monde à la conclusion de cette thèse, mes deux frères Nicolas-Pierre et Daniel, ma petite sœur Griotte dont je suis très fier, et enfin ma mère Andrée-Mathilde Ethève qui m'a toujours gardé son amour malgré les distances et à qui je dédie ce document.

# Notations principales

- $M$  : nombre de signaux sources
- $N$  : nombre de capteurs
- $\theta$  : angle de rotation de Givens
- $\alpha, \mu, \gamma, \lambda$  : pas d'adaptation des algorithmes itératifs

## vecteurs et matrices utilisées

De façon générale, nous noterons les vecteurs en gras minuscule et les matrices en gras majuscule.

- $\mathbf{a}$  :  $(1, M)$  vecteur source
- $\mathbf{x}$  :  $(1, N)$  vecteur observation
- $\mathbf{b}$  :  $(1, N)$  vecteur bruit additif
- $\mathbf{z}$  :  $(1, N)$  vecteur observation blanchies
- $\mathbf{y}$  :  $(1, M)$  vecteur des sources estimées
- $\mathbf{G}$  :  $(N, M)$  matrice de mélange
- $\mathbf{B}$  : matrice de blanchiment
- $\mathbf{U}$  : matrice orthogonale de séparation
- $\mathbf{H}$  :  $(N, N)$  matrice de séparation avant itération
- $\mathbf{H}'$  :  $(N, N)$  matrice de séparation après itération
- $\mathbf{S}$  : matrice du système global mélange-séparation
- $\mathbf{D}$  : matrice diagonale inversible
- $\mathbf{P}$  : matrice de permutation



- $\mathbf{I}$  : matrice identité
- $\mathbf{R}_x$  : matrice de covariance de  $\mathbf{x}$
- $\mathbf{V}$  : matrice des vecteurs propres
- $\mathbf{\Lambda}$  : matrice diagonale des valeurs propres
- $\mathbf{L}$  : matrice triangulaire inférieure dans la factorisation de Cholesky
- $\mathbf{Q}_z$  : matrice cumulante des signaux blanchis

## opérateurs courants

- $(\cdot)^T$  opérateur de transposition
- $E[\cdot]$  opérateur d'espérance mathématique
- $\text{Trace}[\cdot]$  opérateur trace
- $\langle \cdot, \cdot \rangle$  produit scalaire Euclidien
- $\|\cdot\|_{\text{Fro}}$  la norme de Frobenius
- $\text{Kld}(\cdot)$  la divergence de Kullback-Leibler
- $\text{Cum}[\cdot]$  opérateur cumulant multivariable
- $\text{C}_R[\cdot]$  opérateur cumulant d'ordre  $R$
- $\text{sign}(\cdot)$  fonction signe
- $g(\cdot)$  fonction non linéaire

# Table des matières

<b>Résumé</b>	<b>1</b>
<b>Remerciements</b>	<b>3</b>
<b>Notations principales</b>	<b>5</b>
<b>1 Introduction</b>	<b>10</b>
1.1 Formulation du problème . . . . .	10
1.1.1 Mélange . . . . .	11
1.1.2 Séparation . . . . .	12
1.1.3 Indéterminations . . . . .	13
1.1.4 Résumé du modèle et des hypothèses . . . . .	15
1.2 Fonctions de contraste . . . . .	16
1.2.1 Vecteurs blancs . . . . .	16
1.2.2 Définitions et propriétés . . . . .	17
1.2.3 Exemples de contrastes . . . . .	18
1.3 Quelques algorithmes . . . . .	20
1.3.1 L'algorithme ICA . . . . .	21
1.3.2 L'algorithme JADE . . . . .	22
1.3.3 L'algorithme EASI . . . . .	23
Annexe 1 : Rappels sur les cumulants . . . . .	25
<b>2 Une généralisation de ICA</b>	<b>27</b>

2.1	Nouveaux contrastes . . . . .	27
2.1.1	Première famille . . . . .	28
2.1.2	Deuxième famille . . . . .	29
2.2	Solution optimale dans le cas deux sources . . . . .	31
2.2.1	Première famille . . . . .	32
2.2.2	Deuxième famille . . . . .	35
2.3	Quelques simulations numériques . . . . .	38
2.3.1	Cadre des simulations . . . . .	38
2.3.2	Les résultats . . . . .	40
2.4	Conclusion . . . . .	50
<b>3</b>	<b>Optimisation directe de contrastes</b>	<b>51</b>
3.1	Introduction . . . . .	51
3.2	Méthodes duales . . . . .	53
3.2.1	Méthode Lagrangienne classique . . . . .	53
3.2.2	Deux méthodes de pénalisation . . . . .	56
3.2.3	Lagrangien augmenté . . . . .	58
3.3	Une méthode directe . . . . .	59
3.3.1	Première étape . . . . .	59
3.3.2	Deuxième étape . . . . .	60
3.3.3	Résumé de l'algorithme . . . . .	61
3.4	Simulations numériques . . . . .	63
3.4.1	Algorithmes de Lagrangien . . . . .	63
3.4.2	Algorithmes de pénalisations . . . . .	65
3.4.3	Algorithmes directs . . . . .	68
3.5	conclusion . . . . .	74
	Annexe 2 : Détails des calculs pour OSIBS . . . . .	75
	Annexe 3 : Les algorithmes . . . . .	77

<b>Conclusion et perspectives</b>	<b>82</b>
<b>Annexes</b>	<b>84</b>
Annexe 4 : Nonlinear Constrained Optimization Using Lagrangian Approach For Blind Source Separation . . . . .	85
Annexe 5 : An Iterative Block Procedure For The Optimization Of Constrained Contrast Functions . . . . .	90
Annexe 6 : A Generalized ICA Algorithm . . . . .	97
<b>Bibliographie</b>	<b>104</b>

# Chapitre 1

## Introduction

Ce premier chapitre a pour but de poser le problème de la séparation de sources tel qu'il a été défini dans la littérature, de définir quelques concepts essentiels pour la suite de ce document et enfin de préciser quelques travaux qui serviront de base pour notre étude. Dans un premier temps, nous allons donc formuler le problème de la séparation de source, préciser les hypothèses essentielles, et le cadre de notre étude. Par la suite, nous allons définir le concept de fonctions de contraste et en donner quelques exemples dont nous auront besoin par la suite. La dernière partie de ce chapitre introductif présentera quelques algorithmes de séparation de sources classiques qui serviront de base au développement de nouveaux algorithmes.

### 1.1 Formulation du problème

Depuis une quinzaine d'années, la séparation de sources est devenue un problème important en traitement du signal [AMAR97]-[YEL94]. Ce type de problème peut intervenir lorsque plusieurs signaux sont reçus par plusieurs capteurs, l'ensemble des capteurs étant souvent appelé antenne. On retrouve cette problématique dans des domaines aussi variés que les télécommunications numériques, l'analyse de signaux géologiques, les systèmes de reconnaissance de la parole etc... Pour plus de détails, on pourra consulter les actes du premier congrès mondial spécifique à cette discipline [ICA'99].

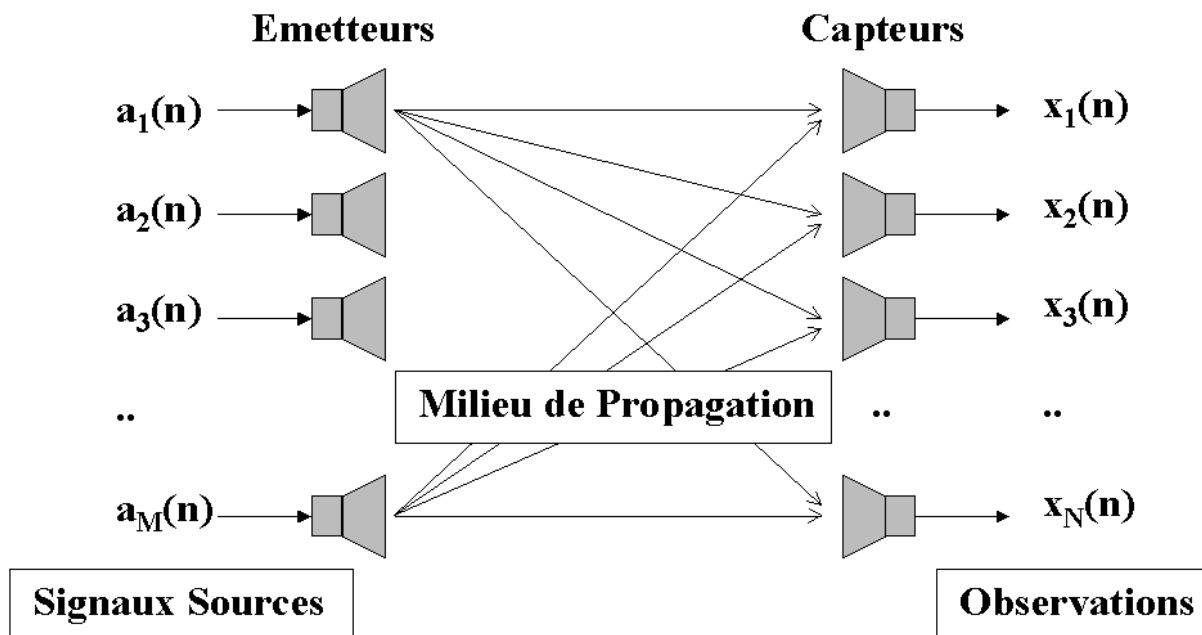


FIG. 1.1 – la propagation

### 1.1.1 Mélange

En général, les signaux reçus par les capteurs appelés observations peuvent être modélisés par la relation suivante

$$\mathbf{x}(n) = \mathcal{G}\{\mathbf{a}(n)\} + \mathbf{b}(n) \quad (1.1)$$

où  $\mathcal{G}\{\cdot\}$  correspond au système physique "mélangeant",  $\mathbf{a}(n) = [a_1(n), \dots, a_M(n)]^T$  le vecteur regroupant l'ensemble des  $M$  signaux appelés sources,  $\mathbf{x}(n) = [x_1(n), \dots, x_N(n)]^T$  le vecteur des  $N$  observations,  $\mathbf{b}(n)$  le vecteur des  $N$  bruits additifs éventuels sur les observations et  $n$  le temps discret <sup>1</sup>.

Les mélanges peuvent être regroupés en deux grandes catégories : les mélanges non-linéaires et les mélanges linéaires. Le mélange linéaire le plus général est décrit par un

---

<sup>1</sup>Dans ce document, tous les signaux considérés sont supposés à temps discret. La raison principale est l'essor très important des techniques de traitement des signaux numériques par informatique. On suppose donc qu'une première étape d'échantillonnage a été réalisé sur les observations.

produit de convolution et dans le cas le plus simple on a la relation suivante

$$\mathbf{x}(n) = \mathbf{G}\mathbf{a}(n) + \mathbf{b}(n) \quad (1.2)$$

où  $\mathbf{G}$  est une matrice  $(N, M)$  à coefficients constants appelée matrice de mélange. Ce dernier type de mélange est qualifié d'instantané dans le sens où l'observation vectorielle  $\mathbf{x}(n)$  à un instant  $n_1$  ne dépend des sources qu'au même instant  $n_1$ . Seul ce type de mélange sera considéré dans ce document.

### 1.1.2 Séparation

Dans le cadre des mélanges de type (1.2), le but de la séparation de sources est d'estimer une matrice dite de séparation afin de récupérer chacun des différents signaux sources indépendamment des autres. Les sources étant supposées non directement observables et le mélange étant inconnu, l'estimation de la matrice de séparation ne doit donc utiliser que les observations. C'est ce dernier point qui fait toute la difficulté du problème et qui lui vaut le qualificatif d'aveugle, de non supervisé ou bien encore d'autodidacte.

Le problème de la séparation de sources consiste donc en l'estimation d'une matrice  $\mathbf{H}$  appelée "*matrice de séparation*" et opérant sur l'observation  $\mathbf{x}(n)$  ainsi

$$\mathbf{y}(n) = \mathbf{H}\mathbf{x}(n) \quad (1.3)$$

de telle sorte que l'on puisse retrouver chacune des différentes sources sur chacune des composantes du vecteur  $\mathbf{y}(n)$ . Autrement dit, il s'agit d'estimer une matrice "inverse" de celle du mélange. Le système global est alors représenté à la figure 1.2.

En présence de bruit, il est clair que l'estimation seule de l'inverse du mélange ne permettra pas la récupération exacte des sources. Chacune des sources estimée sera alors bruitée. En effet, si  $\mathbf{H} = \mathbf{G}^\#$  alors en utilisant (1.2) dans (1.3), on obtient

$$\mathbf{y}(n) = \mathbf{a}(n) + \mathbf{G}^\#\mathbf{b}(n) \quad (1.4)$$

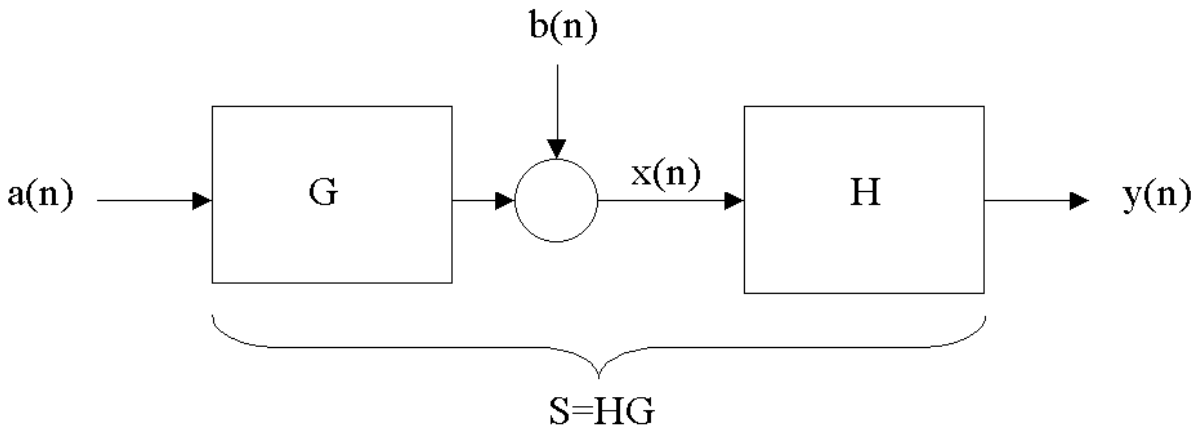


FIG. 1.2 – système global

Où  $\mathbf{G}^\#$  est la matrice pseudo-inverse de la matrice  $\mathbf{G}$

On aura alors effectivement séparé les sources sans les restituer véritablement. Pratiquement, en ce qui nous concerne, la présence du bruit viendra affecter les performances quant à l'estimation de la matrice séparante.

Les sources étant inobservables, et le mélange inconnu, il est clair que sans un minimum d'hypothèses "de travail", la séparation ne sera pas possible. Les deux hypothèses les plus importantes sont les suivantes :

- Les colonnes de  $\mathbf{G}$  sont linéairement indépendantes ;
- Les sources  $a_i(n)$ ,  $i = 1, \dots, M$ , sont statistiquement mutuellement indépendantes.

Ces deux hypothèses seront considérées dans toute la suite. La première semble naturelle dans le sens où l'on désire inverser le système. La seconde est "l'hypothèse clé" qui permettra effectivement la séparation. Il est à noter que c'est une hypothèse forte, mais qui est cependant relativement bien vérifiée dans la plupart des applications.

### 1.1.3 Indéterminations

Pour simplifier, supposons le bruit nul et  $M = N$ . Comme les sources et le mélange sont inconnus, leur caractérisation ne peut pas être unique. En effet, en considérant une



matrice diagonale  $\mathbf{D}$  inversible et une matrice de permutation  $\mathbf{P}$ , on peut toujours écrire

$$\mathbf{x}(n) = \mathbf{G}\mathbf{a}(n) = \mathbf{G}\mathbf{D}^{-1}\mathbf{D}\mathbf{a}(n) = \mathbf{G}_1\mathbf{a}_1(n)$$

où  $\mathbf{G}_1 = \mathbf{G}\mathbf{D}^{-1}$  et  $\mathbf{a}_1(n) = \mathbf{D}\mathbf{a}(n)$ . On peut aussi écrire

$$\mathbf{x}(n) = \mathbf{G}\mathbf{a}(n) = \mathbf{G}\mathbf{P}^{-1}\mathbf{P}\mathbf{a}(n) = \mathbf{G}_2\mathbf{a}_2(n)$$

où  $\mathbf{G}_2 = \mathbf{G}\mathbf{P}^{-1}$  et  $\mathbf{a}_2(n) = \mathbf{P}\mathbf{a}(n)$ .

De plus, si le vecteur  $\mathbf{a}(n)$  a des composantes indépendantes, il en est de même pour  $\mathbf{a}_1(n)$  et  $\mathbf{a}_2(n)$ .

Ainsi, sur la seule hypothèse d'indépendance, on ne pourra récupérer les sources qu'à une puissance et à un ordre près.

En définissant la matrice du système global par

$$\mathbf{S} = \mathbf{H}\mathbf{G} \tag{1.5}$$

on dira que les sources sont séparées lorsque l'on pourra factoriser  $\mathbf{S}$  ainsi :

$$\mathbf{S} = \mathbf{D}\mathbf{P} \tag{1.6}$$

où  $\mathbf{D}$  est une matrice diagonale inversible et  $\mathbf{P}$  une matrice de permutation. En pratique, l'indétermination sur la puissance des sources restituées permet d'imposer des contraintes supplémentaires. Généralement, on considère des signaux sources stationnaires et on impose

$$\mathbb{E}[\mathbf{a}] = \mathbf{0} \tag{1.7}$$

$$\mathbb{E}[\mathbf{a}\mathbf{a}^T] = \mathbf{I}$$

où  $E[\cdot]$  est l'opérateur d'espérance mathématique,  $T$  l'opérateur de transposition et  $\mathbf{I}$  la matrice identité.

### 1.1.4 Résumé du modèle et des hypothèses

Dans toute la suite, on considérera le modèle de mélange suivant

$$\mathbf{x}(n) = \mathbf{G}\mathbf{a}(n) + \mathbf{b}(n)$$

avec des vecteurs  $\mathbf{x}(n)$ ,  $\mathbf{b}(n)$  et  $\mathbf{a}(n)$  de même dimension, *i.e.*  $M = N$ .

Ce cas nécessiterait en principe la détermination au préalable du nombre de sources.

On fait les hypothèses suivantes :

- **H1** : "Indépendance", les sources  $a_i(n)$ ,  $i = 1, \dots, N$ , sont centrées, de puissance 1 et statistiquement indépendantes.
- **H2** : "Stationnarité", les sources  $a_i(n)$ ,  $i = 1, \dots, N$ , sont des signaux aléatoires stationnaires au moins jusqu'à l'ordre considéré par la suite, *i.e.* étant donné  $R$ , le cumulants<sup>2</sup> d'ordre  $R$ ,  $\text{Cum}[a_i(n), \dots, a_i(n)]$ ,  $i = 1, \dots, N$  est indépendant de  $n$  et sera noté  $C_R[a_i]$ .
- **H3** : Au plus un des cumulants  $C_R[a_i]$ ,  $i = 1, \dots, N$ , est nul.
- **H4** : Les bruits  $b_i(n)$  sont statistiquement mutuellement indépendants, indépendants des sources et de même puissance, *i.e.*  $E[\mathbf{b}\mathbf{b}^T] = \sigma_b^2 \mathbf{I}$ .
- **H5** : La matrice  $\mathbf{G}$  est inversible.

Il est utile pour la suite d'introduire les notations suivantes :

**Cas général** : L'ensemble des vecteurs aléatoires satisfaisant **H1**, **H2** et **H3** est noté  $\mathcal{A}$ . L'ensemble des matrices carrées inversibles est noté  $\mathcal{S}$ . Le sous-ensemble de  $\mathcal{S}$  des matrices satisfaisant la relation (1.6) est noté  $\mathcal{P}$  et le sous-ensemble de  $\mathcal{P}$  des matrices diagonales est noté  $\mathcal{D}$ . Finalement l'ensemble des vecteurs aléatoires  $\mathbf{y}(n) = \mathbf{S}\mathbf{a}(n)$  où  $\mathbf{a}(n) \in \mathcal{A}$  et  $\mathbf{S} \in \mathcal{S}$  est noté  $\mathcal{Y}$ .

---

<sup>2</sup>Un rappel sur la définition des cumulants et leurs propriétés est donné en Annexe 1

Il est utile pour la suite d'introduire l'ensemble  $\mathcal{U}$  des matrices unitaires. Le sous-ensemble de  $\mathcal{U}$  des matrices diagonales d'éléments de module 1 est noté  $\mathcal{D}_u$  et l'ensemble des vecteurs  $\mathbf{y}(n) = \mathbf{S}\mathbf{a}(n)$  où  $\mathbf{a}(n) \in \mathcal{A}$  et  $\mathbf{S} \in \mathcal{U}$  est noté  $\mathcal{Y}_u$ .

## 1.2 Fonctions de contraste

### 1.2.1 Vecteurs blancs

Dans le cadre de la séparation de sources, il est classique de considérer une première étape de normalisation du problème. Celle-ci est réalisée en considérant les seules statistiques d'ordre deux des observations. Elle a pour but de réaliser une décorrélation des signaux tout en imposant en plus que leur puissance soit unitaire. Un vecteur de signaux aléatoires dont les composantes sont décorrélées et de puissance 1 est alors qualifié de blanc.

Dans le cas sans bruit, cette normalisation peut être réalisée simplement en multipliant le vecteur des observations  $\mathbf{x}(n)$  par l'inverse d'une racine carrée de sa matrice de covariance. Cela revient à effectuer une classique décomposition de Karhunen-Loève du vecteur  $\mathbf{x}(n)$ . Cette opération réalisée, il est aisé de montrer [COM94] que le mélange peut alors se ramener à une "simple" matrice unitaire, *i.e.*  $\mathbf{G}\mathbf{G}^T = \mathbf{I}$ . Dans ce cas, la matrice de séparation peut bien évidemment, être restreinte à l'ensemble des matrices unitaires et donc la matrice du système global aussi.

Notons finalement qu'en présence de bruit et si l'on dispose de plus d'observations que de signaux sources, cette étape de normalisation peut-être avantageusement utilisée afin de réduire l'influence du bruit en effectuant une projection du vecteur d'observations sur l'espace signal dont la dimension aura été déterminée au préalable.

Dans toute la suite de ce paragraphe, seuls les vecteurs blancs seront considérés.

### 1.2.2 Définitions et propriétés

Tout d'abord rappelons un résultat de [COM94]. Soit  $\mathbf{a}$  un vecteur de composantes statistiquement indépendantes dont au plus une est gaussienne,  $\mathbf{S}$  une matrice unitaire et  $\mathbf{y} = \mathbf{S}\mathbf{a}$ . Alors les composantes de  $\mathbf{y}$  sont statistiquement indépendantes si et seulement si  $\mathbf{S} = \mathbf{D}\mathbf{P}$  (cf. paragraphe 1.1.3) où  $\mathbf{D}$  est une matrice diagonale d'éléments de module 1 et  $\mathbf{P}$  une matrice de permutation.

Ainsi après blanchiment le problème de la séparation de sources peut être résolu en cherchant par transformation unitaire un vecteur de composantes indépendantes. A cette fin, il est nécessaire de disposer d'une mesure de l'indépendance statistique.

Dans le cadre spécifique de la séparation de sources, de telles mesures d'indépendance ont été formalisées sous le nom de fonctions de contraste ou plus simplement contrastes, par P. Comon [COM94]. Ces dernières ont récemment été généralisées dans [MOR8-99].

La définition d'un contraste est

**Définition 1.** *Un contraste sur un ensemble  $\mathcal{Y}$  est une fonction multi-variable  $\mathcal{I}(\cdot)$  de l'ensemble  $\mathcal{Y}$  sur  $\mathbb{R}$  satisfaisant les trois propriétés suivantes :*

- **P1** :  $\forall \mathbf{y} \in \mathcal{Y}, \quad \forall \mathbf{D} \in \mathcal{D}, \quad \mathcal{I}(\mathbf{D}\mathbf{y}) = \mathcal{I}(\mathbf{y}) ;$
- **P2** :  $\forall \mathbf{a} \in \mathcal{A}, \quad \forall \mathbf{S} \in \mathcal{S}, \quad \mathcal{I}(\mathbf{S}\mathbf{a}) \leq \mathcal{I}(\mathbf{a}) ;$
- **P3** :  $\forall \mathbf{a} \in \mathcal{A}, \quad \forall \mathbf{S} \in \mathcal{S}, \quad \mathcal{I}(\mathbf{S}\mathbf{a}) = \mathcal{I}(\mathbf{a}) \Rightarrow \mathbf{S} \in \mathcal{P}.$

La très grande majorité des contrastes travaille sur des vecteurs blancs. Ainsi dans la définition précédente il suffit de remplacer  $\mathcal{Y}$  par  $\mathcal{Y}_u$ ,  $\mathcal{D}$  par  $\mathcal{D}_u$ ,  $\mathcal{S}$  par  $\mathcal{U}$ . Dans toute la suite ne seront considérés que des contrastes sur  $\mathcal{Y}_u$  (opérant donc sur des vecteurs blancs) que l'on nommera pour simplifier contrastes.

Les contrastes possèdent deux propriétés importantes [MOR1-99a]. La première est une propriété d'additivité.

**Propriété 1.** *Si  $\mathcal{I}_1(\cdot)$  est un contraste sur  $\mathcal{Y}$  alors  $\forall \alpha \in \mathbb{R}_+^*$  et  $\forall \beta \in \mathbb{R}$ , la fonction  $\mathcal{I}_2(\cdot) = \alpha\mathcal{I}_1(\cdot) + \beta$  est aussi un contraste sur  $\mathcal{Y}$ .*

Cela permet d'introduire la notion de contraste équivalent :

**Définition 2.** Si deux contrastes  $\mathcal{I}_1(\cdot)$  et  $\mathcal{I}_2(\cdot)$  vérifient la propriété 1, ils sont alors dits équivalents.

La seconde propriété est

**Propriété 2.** Soit  $\mathcal{I}_1(\cdot)$  une fonction de  $\mathcal{Y}$  sur  $\mathbb{R}$  et  $\mathcal{I}_2(\cdot)$  un contraste sur  $\mathcal{Y}$ . Si

$$\begin{aligned}\forall \mathbf{y} \in \mathcal{Y}, \quad & \mathcal{I}_1(\mathbf{y}) \leq \mathcal{I}_2(\mathbf{y}) \\ \forall \mathbf{a} \in \mathcal{A}, \quad & \mathcal{I}_1(\mathbf{a}) = \mathcal{I}_2(\mathbf{a})\end{aligned}$$

alors  $\mathcal{I}_1(\cdot)$  est un contraste sur  $\mathcal{Y}$ .

### 1.2.3 Exemples de contrastes

L'indépendance statistique est définie grâce à la densité de probabilité du vecteur considéré. Soit  $p_{\mathbf{y}}(\mathbf{y})$  la densité de probabilité du vecteur aléatoire  $\mathbf{y}$ , alors  $\mathbf{y}$  aura ses composantes indépendantes si et seulement si

$$p_{\mathbf{y}}(\mathbf{y}) = \prod_{i=1}^N p_{y_i}(y_i). \quad (1.8)$$

Ainsi, la façon la plus naturelle de mesurer l'indépendance est de disposer d'une mesure (de similarité) entre les densités à gauche et à droite du signe égal dans (1.8). Une des mesures la plus connue est la divergence de Kullback qui appliquée à notre cas donne l'information mutuelle (moyenne) :

$$\mathcal{I}_m(\mathbf{y}) = \int p_{\mathbf{y}}(\mathbf{y}) \log \frac{p_{\mathbf{y}}(\mathbf{y})}{\prod p_{y_i}(y_i)} d\mathbf{y}. \quad (1.9)$$

$\mathcal{I}_m(\mathbf{y})$  (qui est toujours supérieur ou égal à 0) est nulle si et seulement si le vecteur  $\mathbf{y}$  a des composantes indépendantes. Il a été montré dans [COM94] que  $-\mathcal{I}_m(\mathbf{y})$  est un contraste. D'un point de vue pratique, l'utilisation du contraste précédent est sérieusement compromise car il est nécessaire de connaître (ou d'approximer) les densités de probabilité des vecteurs considérés ce qui n'est pas en général un problème simple<sup>3</sup>. Ainsi, il serait

---

<sup>3</sup>Si l'on considère plutôt la fonctionnelle de vraisemblance à maximiser, on montre [PHAM96] [PHAM97] [CAR98] que la dérivée relative de la log-vraisemblance par rapport à la matrice de séparation

au moins nécessaire d'avoir une "bonne" approximation du contraste  $-\mathcal{I}_m(\mathbf{y})$ . Dans le meilleur des cas, il serait de plus important de montrer que cette approximation est encore un contraste afin d'en justifier l'utilisation.

D'un autre point de vue, il n'y a aucune raison qu'un contraste ne soit exclusivement qu'une approximation d'un critère plus général. Ainsi la fonction suivante [COM94]

$$\mathcal{I}_R^2(y) = \sum_{i=1}^N (\mathbf{C}_R[y_i])^2, \quad R \geq 3, \quad (1.10)$$

est aussi un contraste. Elle a, en comparaison avec  $\mathcal{I}_m(\mathbf{y})$ , l'avantage de la simplicité (au prix peut être d'une certaine optimalité statistique). De plus, il a été montré [COM94] que la maximisation de  $\mathcal{I}_R^2(\mathbf{y})$  est équivalente à la minimisation de la somme des carrés de tous les cumulants croisés d'ordre  $R$ . Ceci donne une interprétation intéressante de  $\mathcal{I}_R^2(\mathbf{y})$  en terme de "mesure d'indépendance à l'ordre  $R$ ".

Plus tard, il a été construit une famille de contraste dont  $\mathcal{I}_R^2(\mathbf{y})$  est un membre. Ainsi dans [MOR8-99], il est montré que la fonction

$$\mathcal{I}_R^f(\mathbf{y}) = \sum_{i=1}^N \alpha_i f(|\mathbf{C}_R[y_i]|), \quad R \geq 3, \quad (1.11)$$

où  $f(\cdot)$  est une fonction croissante strictement convexe et où les  $\alpha_i$ ,  $i = 1, \dots, N$ , satisfont  $\alpha_1 \geq \dots \geq \alpha_N > 0$ , est un contraste. Ainsi en considérant  $\alpha_i = 1$ , pour tout  $i$  et  $f(u) = u^2$  on obtient  $\mathcal{I}_R^2(\mathbf{y})$ . De plus, si l'on considère  $f(u) = u$  on obtient le contraste

$$\mathcal{I}_R^1(\mathbf{y}) = \sum_{i=1}^N |\mathbf{C}_R[y_i]|, \quad R \geq 3. \quad (1.12)$$

Ce dernier contraste possède une intéressante simplification lorsque  $\forall i, \text{Sign}(\mathbf{C}_R[a_i]) = \varepsilon_R$  avec  $R$  un entier pair, *i.e.* lorsque tout les cumulants d'ordre  $R$  pair des sources ont un s'exprime simplement grâce aux densités de probabilités marginales (réelles ou supposées) des signaux sources. Cela simplifie beaucoup l'optimisation d'une telle fonctionnelle par rapport à l'optimisation de l'information mutuelle.

signe constant. En effet, dans ce dernier cas, la fonction suivante est un contraste

$$\mathcal{I}_R(\mathbf{y}) = \varepsilon_R \sum_{i=1}^N \mathbf{C}_R[y_i], \quad R \geq 4 \text{ et } R \text{ pair.} \quad (1.13)$$

On peut remarquer que les contrastes de type  $\mathcal{I}_R^f(\cdot)$  ne comportent aucun cumulants croisé. Ce n'est pas le cas du contraste [CAR93]

$$\mathcal{J}(\mathbf{y}) = \sum_{i_1, i_2, i_3=1}^N (\text{Cum}[y_{i_1}, y_{i_1}, y_{i_2}, y_{i_3}])^2 \quad (1.14)$$

qui possède une intéressante interprétation en terme de diagonalisation conjointe d'un ensemble de matrices. Finalement, un dernier contraste a été proposé dans [DEL97]

$$\mathcal{Q}(\mathbf{y}) = \sum_{i_1, i_2=1}^N (\text{Cum}[y_{i_1}, y_{i_1}, y_{i_1}, y_{i_2}])^2. \quad (1.15)$$

Il possède aussi une intéressante interprétation en terme de diagonalisation conjointe d'un ensemble de tenseurs d'ordre trois.

Ces deux derniers contrastes ont été récemment généralisés pour des cumulants d'ordre quelconque supérieurs ou égal à trois dans [MOR3-99].

### 1.3 Quelques algorithmes

La première contribution majeure traitant du problème de séparation de sources est l'algorithme connu sous le nom de "Hérault-Jutten" [HER85] [JUT91]. Cet algorithme adaptatif est basé sur un réseau de neurones totalement récursif. Depuis le nombre d'algorithmes proposé n'a cessé de croître. On pourra consulter à ce propos [ICA'99]. On peut distinguer les algorithmes adaptatifs, des algorithmes utilisant des blocs de données. On peut aussi distinguer les algorithmes utilisant exclusivement des statistiques d'ordre deux et ceux utilisant (plus ou moins explicitement) des statistiques d'ordre supérieurs à deux. Nous allons maintenant présenter les principes de base de trois algorithmes particuliè-

rement importants pour la suite de cette thèse. Les deux premiers sont des algorithmes utilisant des blocs de données et le dernier est un algorithme adaptatif.

### 1.3.1 L'algorithme ICA

L'algorithme **ICA** [COM94] pour "Independent Component Analysis" est aussi l'une des premières contributions importantes en séparation de sources. C'est un algorithme utilisant des blocs de données dont le but est la maximisation du contraste

$$\mathcal{I}_4^2(\mathbf{y}) = \sum_{i=1}^N (C_4[y_i])^2 \quad (1.16)$$

après blanchiment des observations. La maximisation est réalisée sur le principe des algorithmes de type Jacobi, *i.e.* la matrice de séparation est décomposée en un produit de  $N' = \frac{N(N-1)}{2}$  rotations planes (ou de Givens) opérant sur chacune des paires du vecteur de sortie.

Ainsi, on décompose un problème de dimension  $N$  en  $N'$  problèmes de dimension deux dont on espère qu'ils soient plus simples. Cela sera tout particulièrement le cas si l'on est capable d'obtenir la solution analytique exacte du problème de dimension deux. Ce qui a été réalisé dans [COM94] concernant  $\mathcal{I}_4^2(\mathbf{y})$ . Il s'agit alors de déterminer les racines d'un polynôme de degré quatre, ce qui est analytiquement réalisable. Cependant dans le cas de mélanges complexes, les choses s'avèrent beaucoup plus compliquées. C'est l'une des principales raisons ayant conduit à proposer un algorithme de même type [COM97] mais construit sur le contraste "plus simple"

$$\mathcal{I}_4(\mathbf{y}) = \varepsilon_4 \sum_{i=1}^N C_4[y_i]. \quad (1.17)$$

Il a alors été montré [COM97] qu'une solution optimale de  $\mathcal{I}_4(\mathbf{y})$  peut être trouvée en calculant les racines d'un polynôme de degré deux dans le cas réel et d'un polynôme de degré trois dans le cas complexe.



### 1.3.2 L'algorithme JADE

Le terme JADE signifie "Joint Approximate Diagonalization of Eigen-Matrices". Cet algorithme [CAR93] est le premier algorithme de séparation de sources par diagonalisation conjointe d'un ensemble de matrices. Ces matrices sont construites à partir de cumulants d'ordre quatre des composantes du vecteur de sortie. Cette notion de diagonalisation conjointe est définie par l'intermédiaire d'un critère (dit de diagonalisation conjointe) qui est le suivant

$$\mathcal{D}(\mathbf{H}, \mathcal{M}) = \sum_{m=1}^M \left( \sum_{i=1}^N |M_{i,i}^H(m)|^2 \right) \quad (1.18)$$

où

$$M_{i,i}^H(m) = \sum_{n_1, n_2} H_{i, n_1} H_{i, n_2} M_{n_1, n_2}(m) \quad (1.19)$$

et  $\mathcal{M}$  est un ensemble de  $M$  matrices carrées  $(N, N)$   $\mathbf{M}(m)$  pour  $m$  allant de 1 à  $M$ . Une matrice diagonalisant conjointement l'ensemble  $\mathcal{M}$  est alors définie comme l'argument d'un maximum de  $\mathcal{D}(\mathbf{H}, \mathcal{M})$  sur l'ensemble des matrices unitaires.

Concernant le critère  $\mathcal{D}(\mathbf{H}, \mathcal{M})$ , il s'agit donc de déterminer une matrice unitaire maximisant la somme des carrés des éléments diagonaux de chacune des matrices de l'ensemble  $\mathcal{M}$  après changement de base. Il est facile de montrer que cela revient à minimiser la somme des carrés des éléments hors diagonal, ce qui correspond à une "mesure naturelle" de l'écart par rapport à une matrice diagonale.

Un point très intéressant concernant l'algorithme **JADE**, est qu'il existe une équivalence entre son approche par diagonalisation conjointe et la fonction de contraste donnée dans (1.14). En effet, en introduisant les matrices de cumulants  $\mathbf{C}(i_3, i_4) = (C_{i_1, i_2}(i_3 i_4))$  avec

$$C_{i_1, i_2}(i_3 i_4) = \text{Cum}[x_{i_1}, x_{i_2}, x_{i_3}, x_{i_4}] \quad (1.20)$$

et en notant  $\mathcal{C}_4^2$  leur ensemble (pour  $i_3$  et  $i_4$  variant entre 1 et  $N$ , *i.e.* il y a  $N^2$  matrices dans cet ensemble), alors si  $\mathbf{H}$  est une matrice unitaire, on montre l'équivalence

$$\mathcal{D}(\mathbf{H}, \mathcal{C}_4^2) = \mathcal{J}(\mathbf{H}\mathbf{x}). \quad (1.21)$$

### 1.3.3 L'algorithme EASI

L'idée originale à la base de l'algorithme **EASI** est la notion de gradient matriciel relatif qui généralise en un certain sens, la notion d'erreur relative. Un algorithme de gradient classique s'écrit, en général sous la forme

$$\Delta\mathbf{H} = -\mu \frac{\partial C}{\partial \mathbf{H}} \quad (1.22)$$

où  $C$  est le critère que l'on désire minimiser par rapport à son argument matriciel  $\mathbf{H}$ .

Dans [CAR96] un gradient qualifié de relatif est défini par la relation

$$\nabla_{\mathbf{H}} C = \frac{\partial C}{\partial \mathbf{H}} \mathbf{H}^T. \quad (1.23)$$

Il est alors utilisé pour l'estimation de  $\mathbf{H}$  (minimum de  $C$ ) grâce à

$$\Delta\mathbf{H} = -\mu \nabla_{\mathbf{H}} C \mathbf{H}. \quad (1.24)$$

Cela conduit généralement à des algorithmes dont l'équation d'estimation ne dépend que des composantes du vecteur de sortie  $\mathbf{y}$  et dont on montre qu'ils possèdent des performances uniformes de séparation en l'absence de bruit.

Dans [CAR96] le critère (à maximiser) suivant

$$\mathcal{J}_4(\mathbf{y}) = \varepsilon_4 \sum_{i=1}^N \mathbb{E}[y_i^4] \quad (1.25)$$

où  $\varepsilon_4$  est le signe (supposé identique) du cumulante d'ordre 4 des sources, a été utilisé.

Cependant son optimisation doit être réalisé sous contrainte de blancheur du vecteur de

sortie :  $\mathbf{E}[\mathbf{y}\mathbf{y}^T] = \mathbf{I}$ .

L'algorithme **EASI** propose une optimisation originale du critère contraint précédent à l'aide d'un algorithme de gradient relatif. Si l'on ne s'occupe pas de la contrainte, on a

$$\nabla_H \mathcal{J}_4 = \varepsilon \mathbf{E}[g(\mathbf{y})\mathbf{y}^T] \quad (1.26)$$

où la fonction non linéaire  $g(\cdot) = (\cdot)^3$  agit composantes par composantes. Considérons maintenant la contrainte de blancheur du vecteur de sortie. Une mesure de la distance entre une matrice et la matrice identité est donné par

$$D(\mathbf{R}_y) = \frac{1}{2}(\text{Trace}[\mathbf{R}_y] - \log \det(\mathbf{R}_y) - N) \quad (1.27)$$

qui est toujours non négative et nulle si et seulement si  $\mathbf{R}_y = \mathbf{I}$ . Ainsi un critère à minimiser pour respecter la contrainte de blancheur est donné par

$$J_2(\mathbf{y}) = D(\mathbf{E}[\mathbf{y}\mathbf{y}^T]) \quad (1.28)$$

dont le gradient relatif vaut

$$\nabla_H J_2 = \mathbf{E}[\mathbf{y}\mathbf{y}^T] - \mathbf{I}. \quad (1.29)$$

Pour des sources de cumulants d'ordre 4 négatif ( $\varepsilon_4 = -1$ ), les auteurs proposent la combinaison suivante des deux adaptations précédentes pour l'estimation de la matrice de séparation

$$\Delta \mathbf{H} = -\mu \left[ \nabla_H J_2 + \frac{\nabla_H \mathcal{J}_4 - (\nabla_H \mathcal{J}_4)^T}{2} \right] \mathbf{H}. \quad (1.30)$$

Pour en obtenir la version stochastique **EASI**, il suffit de remplacer chacun des termes  $\mathbf{E}[u]$  par sa valeur instantanée  $u(n)$ . Il est à noter qu'une version normalisée est aussi proposée dans [CAR96].

## Annexe 1: Rappels sur les cumulants

Parmi les outils statistiques auxquels nous avons accès, nous allons nous intéresser aux cumulants. Pour une variable aléatoire  $y$  centrée, on a

- $\text{Cum}[y, y] = \mathbf{E}[y^2]$
- $\text{Cum}[y, y, y] = \mathbf{E}[y^3]$
- $\text{Cum}[y, y, y, y] = \mathbf{E}[y^4] - 3\left(\mathbf{E}[y^2]\right)^2$

Jusqu'à l'ordre trois, les cumulants coïncident avec les moments.

Pour la suite, nous noterons  $\mathbf{C}_R[y] = \text{Cum}(y, y, \dots, y)$  le cumulante d'ordre  $R$  de la variable aléatoire  $y$ . Ces cumulants seront qualifiés d'*auto-cumulants* par opposition avec les cumulants *croisés* qui s'appliquent aux cas où l'on dispose conjointement de plusieurs variables aléatoires.

Pour la suite, nous allons préciser les cumulants croisés d'ordre 4 dans le cas où l'on considère deux variables aléatoires  $x_1$  et  $x_2$  sachant que l'on peut imaginer toutes les combinaisons possibles entre l'ordre et le nombre de variables aléatoires considérées :

- $\text{Cum}[x_1, x_1, x_1, x_2] = \mathbf{E}[x_1^3 x_2] - 3\mathbf{E}[x_1^2] \mathbf{E}[x_1 x_2]$
- $\text{Cum}[x_1, x_1, x_2, x_2] = \mathbf{E}[x_1^2 x_2^2] - \mathbf{E}[x_1^2] \mathbf{E}[x_2^2] - 2\mathbf{E}[x_1 x_2]^2$
- $\text{Cum}[x_1, x_2, x_2, x_2] = \mathbf{E}[x_1 x_2^3] - 3\mathbf{E}[x_2^2] \mathbf{E}[x_1 x_2]$

Les cumulants possèdent quelques propriétés intéressantes :

- **P1** : Multi-linéaires
  - $\text{Cum}[x + y, z_1, \dots, z_p] = \text{Cum}[x, z_1, \dots, z_p] + \text{Cum}[y, z_1, \dots, z_p]$
  - $\text{Cum}[\lambda x, z_1, \dots, z_p] = \lambda \text{Cum}[x, z_1, \dots, z_p]$
- **P2** : Symétriques
  - $\text{Cum}[x, y] = \text{Cum}[y, x]$
- **P3** : Indépendance
  - $\text{Cum}[y_1, y_2, y_3, y_4] = 0$  si  $\exists y_i$  indépendant de  $y_j$  ( $i \neq j$ )
  - si  $x_i$  indépendant de  $y_j$  alors  $\text{Cum}[x_1 + y_1, \dots, x_i + y_i] = \text{Cum}[x_1, \dots, x_i] + \text{Cum}[y_1, \dots, y_i]$
- **P4** : si  $x$  est une variable aléatoire et  $y$  une variable aléatoire gaussienne de même

moyenne et de même variance, alors  $C_R[x] = E[x] - E[y]$

Parmi ces outils statistiques d'ordre supérieur, le *Kurtosis* (cumulant d'ordre 4) apparaît comme très intéressant. Typiquement, en statistique, la valeur du kurtosis est introduite pour mesurer si une distribution est *pointue* ou pas, les valeurs positives du kurtosis étant données par des distributions pointues et étroites tandis que les valeurs négatives sont données par des distributions *plates*, le kurtosis d'une variable aléatoire à densité de probabilité gaussienne est égal à zéro. On peut donc considérer le Kurtosis comme une certaine mesure de la distance entre une variable aléatoire et une gaussienne.

Le kurtosis d'une variable aléatoire  $y$  est défini comme :

$$C_4[y] = E[y^4] - 3(E[y^2])^2 \quad (1.31)$$

- Si  $y$  est à distribution gaussienne, son kurtosis  $C_4[y] = 0$ .
- les signaux dont le kurtosis est négatif sont souvent qualifiés de *sous-gaussien*.
- les signaux dont le kurtosis est positif sont souvent qualifiés de *sur-gaussien*.

Cependant cette notion de sous(sur)-gaussien doit être maniée avec précision [MAN99], en effet, la relation entre le signe du kurtosis et la forme de la densité de probabilité n'est vraie que dans le cas de signaux à distribution uni-modale.

Le kurtosis est parfois défini en utilisant une variable aléatoire normalisée, *i. e.* de puissance un.

# Chapitre 2

## Une généralisation de ICA

Les contrastes tels que  $\mathcal{J}(\mathbf{y})$  dans (1.14) ou  $\mathcal{Q}(\mathbf{y})$  dans (1.15) contiennent des cumulants croisés dont le nombre et le type sont imposés par la fonction elle-même et par l'ordre des cumulants considérés [MOR8-99].

Nous montrons dans ce chapitre que les cumulants croisés au niveau des contrastes peuvent être choisis de manière arbitraire en ce qui concerne leur nombre et leur type. Cela nous permet de proposer dans un premier temps une généralisation des contrastes à la base des algorithmes **ICA1** [COM97] et **ICA2** [COM94] où l'on pourra "jouer" sur les cumulants croisés. Par la suite, nous allons chercher une solution analytique au problème de la séparation de deux sources pour les deux familles de contrastes précédemment proposés. Cela nous permettra de développer par la suite deux algorithmes basés sur les techniques de Jacobi et d'en proposer des simulations numériques afin d'en étudier les comportements.

### 2.1 Nouveaux contrastes

Pour des raisons de simplicité, nous n'allons considérer que les cumulants d'ordre 4 dans le cas réel. Considérons ainsi les notations suivantes

$$Y_{i,j,k,l} = \text{Cum}[y_i, y_j, y_k, y_l] \quad (2.1)$$

$$X_{i,j,k,l} = \text{Cum}[x_i, x_j, x_k, x_l] \quad (2.2)$$

### 2.1.1 Première famille

Maintenant il est intéressant de remarquer que le contraste  $\mathcal{J}(\mathbf{y})$  dans (1.14) peut toujours s'écrire

$$\mathcal{J}(\mathbf{y}) = \mathcal{I}_4^2(\mathbf{y}) + 2(\mathcal{C}_1(\mathbf{y}) + \mathcal{C}_2(\mathbf{y}) + \mathcal{C}_3(\mathbf{y})) \quad (2.3)$$

où  $\mathcal{I}_4^2(\mathbf{y})$  est le contraste dans (1.16) et

$$\mathcal{C}_1(\mathbf{y}) = \sum_{\substack{i_1, i_2 \\ i_2 \neq i_1}} (\mathbf{Y}_{i_1, i_1, i_1, i_2})^2 \quad (2.4)$$

$$\mathcal{C}_2(\mathbf{y}) = \sum_{\substack{i_1, i_2 \\ i_2 > i_1}} (\mathbf{Y}_{i_1, i_1, i_2, i_2})^2 \quad (2.5)$$

$$\mathcal{C}_3(\mathbf{y}) = \sum_{\substack{i_1, i_2, i_3 \\ i_1 \neq i_2 \neq i_3 \\ i_3 > i_2}} (\mathbf{Y}_{i_1, i_1, i_2, i_3})^2 \quad (2.6)$$

On propose alors le résultat suivant

**Proposition 2.1.1.** *La fonction*

$$\mathcal{J}_p(\mathbf{y}) = \mathcal{I}_4^2(\mathbf{y}) + 2(\alpha_1 \mathcal{C}_1(\mathbf{y}) + \alpha_2 \mathcal{C}_2(\mathbf{y}) + \alpha_3 \mathcal{C}_3(\mathbf{y})) \quad (2.7)$$

où pour  $i = 1, 2, 3$ ,  $\alpha_i \leq 1$ , est un contraste pour des vecteurs blancs.

**preuve.** La preuve est très simple en utilisant la Propriété 2 (cf. paragraphe 1.2.2) des contrastes. En effet, comme  $\forall i, \alpha_i \leq 1$  on a immédiatement

$$\mathcal{J}_p(\mathbf{y}) \leq \mathcal{J}(\mathbf{y})$$

$$\mathcal{J}_p(\mathbf{a}) = \mathcal{J}(\mathbf{a})$$

Maintenant comme  $\mathcal{J}(\mathbf{y})$  est un contraste pour des vecteurs blancs, la propriété 2 permet de conclure que  $\mathcal{J}_p(\mathbf{y})$  est aussi un contraste pour des vecteurs blancs.  $\square$

En suivant le principe de la preuve précédente, il est clair que l'on peut pondérer de la

même façon les cumulants croisés de tout contraste qui en comporte. Ainsi  $\mathcal{J}_p(\mathbf{y})$  constitue une nouvelle famille de contrastes paramétrés par les coefficients  $\alpha_1, \alpha_2, \alpha_3$ . Remarquons que si l'on fait  $\alpha_1 = \alpha_2 = \alpha_3 = 0$ , alors  $\mathcal{J}_p(\mathbf{y}) = \mathcal{I}_4^2(\mathbf{y})$ . On retrouve le contraste à la base de l'algorithme **ICA2** [COM94]. Maintenant si  $\alpha_1 = \alpha_2 = \alpha_3 = 1$ , alors  $\mathcal{J}_p(\mathbf{y}) = \mathcal{J}(\mathbf{y})$ . On retrouve le contraste à la base de l'algorithme **JADE** [CAR93].

De plus si  $\alpha_1 = \frac{1}{2}$  et  $\alpha_2 = \alpha_3 = 0$  alors,  $\mathcal{J}_p(\mathbf{y}) = \mathcal{Q}(\mathbf{y})$ . On retrouve le contraste à la base de l'algorithme **STOTD** [DEL97].

Pour finir, si  $\alpha_1 = \alpha_3 = 0$  et  $\alpha_2 = 1$  alors  $\mathcal{J}_p(\mathbf{y}) = \sum_{i,j} (\text{Cum}[y_i, y_i, y_j, y_j])^2$ . On retrouve un contraste récemment proposé dans [CAR99] [MOR3-99]. Toute autre valeur de  $\alpha_1, \alpha_2$  et  $\alpha_3$  conduit à un nouveau contraste.

### 2.1.2 Deuxième famille

Cette deuxième famille de contrastes sera construite à partir de  $\mathcal{I}_4(\mathbf{y})$  (1.17). On notera  $\varepsilon$  le signe du kurtosis des sources supposé identique *i.e.*

$$\forall i, \varepsilon = \text{sign}(\mathcal{C}_4[a_i]). \quad (2.8)$$

Cependant il est nécessaire au préalable de présenter deux résultats.

**Lemme 1.** *La fonction*

$$\mathcal{K}_\ell(\mathbf{y}) = \sum_{i,j} Y_{i,i,j,j} \quad (2.9)$$

*est invariante par transformation unitaire de son argument vectoriel.*

**preuve.** Posons  $\mathbf{y} = \mathbf{H}\mathbf{x}$  où  $\mathbf{H}$  est une matrice unitaire. On a

$$Y_{i,i,j,j} = \sum_{i_1, i_2, i_3, i_4} H_{i, i_1} H_{i, i_2} H_{j, i_3} H_{j, i_4} X_{i_1, i_2, i_3, i_4} \quad (2.10)$$



et ainsi

$$\mathcal{K}_\ell(\mathbf{y}) = \sum_{i_1, i_2, i_3, i_4} \left( \sum_i H_{i, i_1} H_{i, i_2} \right) \left( \sum_j H_{j, i_3} H_{j, i_4} \right) \mathbf{X}_{i_1, i_2, i_3, i_4}. \quad (2.11)$$

Comme  $\mathbf{H}$  est unitaire, on a  $\sum_i H_{i, i_1} H_{i, i_2} = \delta[i_1 - i_2]$  où  $\delta[k] = 1$  si  $k = 0$  et 0 sinon.

Finalement

$$\mathcal{K}_\ell(\mathbf{y}) = \sum_{i_1, i_3} \mathbf{X}_{i_1, i_1, i_3, i_3} = \mathcal{K}_\ell(\mathbf{x}). \quad (2.12)$$

Ainsi la fonction  $\mathcal{K}_\ell(\mathbf{y})$  est invariante par transformation unitaire de son argument  $\square$

Le deuxième résultat est

**Lemme 2.** *La fonction*

$$\mathcal{K}(\mathbf{y}) = -\varepsilon \sum_{\substack{i, j \\ j > i}} \mathbf{Y}_{i, i, j, j} \quad (2.13)$$

*est un contraste pour des vecteurs blancs.*

**preuve.** voir [MOR1-96]  $\square$

Remarquons maintenant que

$$\varepsilon \mathcal{K}_\ell(\mathbf{y}) = \mathcal{I}_4(\mathbf{y}) - 2\mathcal{K}(\mathbf{y})$$

comme  $\mathcal{K}_\ell(\mathbf{y})$  est constant (cf. Lemme 1), *i.e.* ne dépend pas de  $H$ , cette dernière relation prouve que les contrastes  $\mathcal{I}_4(\mathbf{y})$  et  $\mathcal{K}(\mathbf{y})$  sont équivalents.

Cependant, on peut encore écrire

$$\varepsilon \mathcal{K}_\ell(\mathbf{y}) = [\mathcal{I}_4(\mathbf{y}) - 2\alpha \mathcal{K}(\mathbf{y})] - 2(1 - \alpha) \mathcal{K}(\mathbf{y})$$

Maintenant si l'on veut garder une équivalence entre  $\mathcal{K}(\mathbf{y})$  et la fonction  $\mathcal{I}_4(\mathbf{y}) - 2\alpha \mathcal{K}(\mathbf{y})$

il est nécessaire d'imposer  $\alpha < 1$  (cf. Définition 2).

On a alors le résultat suivant

**Proposition 2.1.2.** *La fonction*

$$\mathcal{K}_p(\mathbf{y}) = \mathcal{I}_4(\mathbf{y}) - 2\alpha\mathcal{K}(\mathbf{y}) \quad (2.14)$$

où  $\alpha < 1$ , est un contraste pour des vecteurs blancs

## 2.2 Solution optimale dans le cas deux sources

En vue d'une maximisation par un algorithme de type Jacobi, il est important de déterminer une solution analytique au problème de la séparation de deux sources par optimisation d'un contraste. Ainsi, nous allons maintenant considérer les deux nouvelles familles de contrastes  $\mathcal{J}_p(\mathbf{y})$  dans (2.7) et  $\mathcal{K}_p(\mathbf{y})$  dans (2.14). Dans le cas de deux sources, ces deux fonctions peuvent s'écrire

$$\mathcal{J}_p(\mathbf{y}) = Y_{1,1,1,1}^2 + Y_{2,2,2,2}^2 + 2\alpha_1 (Y_{1,1,1,2}^2 + Y_{1,2,2,2}^2) + 2\alpha_2 Y_{1,1,2,2}^2 \quad (2.15)$$

et

$$\mathcal{K}_p(\mathbf{y}) = \varepsilon (Y_{1,1,1,1} + Y_{2,2,2,2} + 2\alpha Y_{1,1,2,2}). \quad (2.16)$$

**Remarque.** Considérons la fonction

$$\mathcal{K}'_p(\mathbf{y}) = Y_{1,1,1,1} + Y_{2,2,2,2} + 2\alpha Y_{1,1,2,2} \quad (2.17)$$

En utilisant  $\mathbf{y} = \mathbf{S}\mathbf{a}$  (cf. paragraphe 1.2.2) où la matrice unitaire  $\mathbf{S}$  est paramétrée par une rotation, *i.e.*

$$\mathbf{S} = \begin{pmatrix} \cos \theta_s & \sin \theta_s \\ -\sin \theta_s & \cos \theta_s \end{pmatrix} \quad (2.18)$$

on a alors

$$\mathcal{K}'_p(\mathbf{y}) = (\mathbf{C}_4[a_1] + \mathbf{C}_4[a_2]) (\cos^4 \theta_s + \sin^4 \theta_s + 2\alpha \cos^2 \theta_s \sin^2 \theta_s).$$

Ainsi en ce qui concerne les sources, seule l'expression  $\mathbf{C}_{a_1}^{a_2} = \mathbf{C}_4[a_1] + \mathbf{C}_4[a_2]$  influence le contraste. En particulier  $\mathbf{C}_{a_1}^{a_2}$  doit être non nul et son signe  $\varepsilon_{12}$  conditionne la maximisation ou la minimisation de  $\mathcal{K}'_p(\mathbf{y})$ . Ainsi, dans le cas de deux sources, on peut remplacer  $\varepsilon$  dans  $\mathcal{K}_p(\mathbf{y})$  par  $\varepsilon_{12}$ , ce qui est un peu plus général.

### 2.2.1 Première famille

Grâce à (1.3) et à la multilinéarité des cumulants on obtient aisément

$$Y_{i,j,k,l} = \sum_{n_1, n_2, n_3, n_4=1}^2 H_{i,n_1} H_{j,n_2} H_{k,n_3} H_{l,n_4} X_{n_1, n_2, n_3, n_4} \quad (2.19)$$

Pour la maximisation de  $\mathcal{J}_p(\mathbf{y})$  on suppose qu'un blanchiment des observations a été réalisé. Ainsi la matrice  $\mathbf{H}$  dans (1.3) peut être considérée unitaire et suivant [COM94] nous allons la paramétrer ainsi

$$H = \frac{1}{\sqrt{1+t^2}} \begin{pmatrix} 1 & t \\ -t & 1 \end{pmatrix}. \quad (2.20)$$

Il est nécessaire d'introduire la notation suivante

$$\xi = t - \frac{1}{t}. \quad (2.21)$$

Déterminons maintenant l'expression de  $\mathcal{J}_p(\mathbf{y})$  en fonction de  $\xi$ . En utilisant (2.20) dans (2.19), on obtient

$$\begin{aligned}
 \frac{(1+t^2)^2}{t^2} \mathbf{Y}_{1,1,1,1} &= \frac{1}{t^2} \mathbf{X}_{1,1,1,1} + \frac{4}{t} \mathbf{X}_{1,1,1,2} + 6\mathbf{X}_{1,1,2,2} + 4t\mathbf{X}_{1,2,2,2} + t^2\mathbf{X}_{2,2,2,2} \\
 \frac{(1+t^2)^2}{t^2} \mathbf{Y}_{2,2,2,2} &= t^2\mathbf{X}_{1,1,1,1} - 4t\mathbf{X}_{1,1,1,2} + 6\mathbf{X}_{1,1,2,2} - \frac{4}{t}\mathbf{X}_{1,2,2,2} + \frac{1}{t^2}\mathbf{X}_{2,2,2,2} \\
 \frac{(1+t^2)^2}{t^2} \mathbf{Y}_{1,1,2,2} &= \mathbf{X}_{1,1,1,1} + 2\left(t - \frac{1}{t}\right)\mathbf{X}_{1,1,1,2} + \left(t^2 + \frac{1}{t^2} - 4\right)\mathbf{X}_{1,1,2,2} \\
 &\quad + 2\left(\frac{1}{t} - t\right)\mathbf{X}_{1,2,2,2} + \mathbf{X}_{2,2,2,2} \\
 \frac{(1+t^2)^2}{t^2} \mathbf{Y}_{1,1,1,2} &= -\frac{1}{t}\mathbf{X}_{1,1,1,1} + \left(\frac{1}{t^2} - 3\right)\mathbf{X}_{1,1,1,2} + 3\left(\frac{1}{t} - t\right)\mathbf{X}_{1,1,2,2} \\
 &\quad + (3 - t^2)\mathbf{X}_{1,2,2,2} + t\mathbf{X}_{2,2,2,2} \\
 \frac{(1+t^2)^2}{t^2} \mathbf{Y}_{1,2,2,2} &= -t\mathbf{X}_{1,1,1,1} + (3 - t^2)\mathbf{X}_{1,1,1,2} + 3\left(t - \frac{1}{t}\right)\mathbf{X}_{1,1,2,2} \\
 &\quad + \left(\frac{1}{t^2} - 3\right)\mathbf{X}_{1,2,2,2} + \frac{1}{t}\mathbf{X}_{2,2,2,2}
 \end{aligned}$$

Introduisons les notations suivantes

$$\begin{aligned}
 \mathbf{A}_1 &= \mathbf{X}_{1,1,1,1}^2 + \mathbf{X}_{2,2,2,2}^2 \\
 \mathbf{A}_2 &= \mathbf{X}_{1,1,1,2}^2 + \mathbf{X}_{1,2,2,2}^2 \\
 \mathbf{A}_3 &= \mathbf{X}_{1,1,2,2}^2 \\
 \mathbf{A}_4 &= \mathbf{X}_{1,2,2,2}\mathbf{X}_{2,2,2,2} - \mathbf{X}_{1,1,1,1}\mathbf{X}_{1,1,1,2} \\
 \mathbf{A}_5 &= \mathbf{X}_{1,1,2,2}(\mathbf{X}_{1,1,1,1} + \mathbf{X}_{2,2,2,2}) \\
 \mathbf{A}_6 &= \mathbf{X}_{1,1,1,2}\mathbf{X}_{2,2,2,2} - \mathbf{X}_{1,1,1,1}\mathbf{X}_{1,2,2,2} \\
 \mathbf{A}_7 &= \mathbf{X}_{1,1,1,1}\mathbf{X}_{2,2,2,2} \\
 \mathbf{A}_8 &= \mathbf{X}_{1,1,2,2}(\mathbf{X}_{1,2,2,2} - \mathbf{X}_{1,1,1,2}) \\
 \mathbf{A}_9 &= \mathbf{X}_{1,1,1,2}\mathbf{X}_{1,2,2,2}
 \end{aligned}$$

En utilisant (2.21) et après quelques calculs le contraste  $\mathcal{J}_p(\mathbf{y})$  peut s'écrire :

$$\mathcal{J}_p(\mathbf{y}) = \frac{b_0 + b_1\xi + b_2\xi^2 + b_3\xi^3 + b_4\xi^4}{(\xi^2 + 4)^2} \quad (2.22)$$

où

$$b_0 = 2(1 + \alpha_2 + 2\alpha_1)\mathbf{A}_1 + 16(2 + \alpha_1)\mathbf{A}_2 + 8(9 + \alpha_2)\mathbf{A}_3 + 8(3 - \alpha_2)\mathbf{A}_5 \\ + 4(1 + \alpha_2 - 2\alpha_1)\mathbf{A}_7 + 32(2 - \alpha_1)\mathbf{A}_9$$

$$b_1 = 8(3 - \alpha_2)\mathbf{A}_4 + 8(1 + \alpha_2 - 2\alpha_1)\mathbf{A}_6 + 16(3 + \alpha_2 - 3\alpha_1)\mathbf{A}_8$$

$$b_2 = 2(2 + \alpha_1)\mathbf{A}_1 + 4(4 + 2\alpha_2 - \alpha_1)\mathbf{A}_2 + 4(9\alpha_1 - 2\alpha_2)\mathbf{A}_3 + 4(3 + \alpha_2 - 3\alpha_1)\mathbf{A}_5 \\ + 4(6\alpha_1 - 4\alpha_2)\mathbf{A}_9$$

$$b_3 = 4(2 - \alpha_1)\mathbf{A}_4 + 4(3\alpha_1 - 2\alpha_2)\mathbf{A}_8$$

$$b_4 = \mathbf{A}_1 + 2\alpha_1\mathbf{A}_2 + 2\alpha_2\mathbf{A}_3$$

Afin de déterminer une solution optimale, il s'agit maintenant de calculer les racines de la dérivée de  $\mathcal{J}_p$  par rapport à  $\xi$ . Cette dérivée s'obtient aisément, on a

$$\frac{\partial \mathcal{J}_p}{\partial \xi} = \frac{c_0 + c_1\xi + c_2\xi^2 + c_3\xi^3 + c_4\xi^4}{(\xi^2 + 4)^3} \quad (2.23)$$

où

$$c_0 = 4b_1$$

$$c_1 = 8b_2 - 4b_0$$

$$c_2 = 12b_3 - 3b_1$$

$$c_3 = 16b_4 - 2b_2$$

$$c_4 = -b_3$$

Il s'agit donc de déterminer les valeurs de  $\xi$  solutions de  $\sum_{i=0}^4 c_i \xi^i = 0$ . Il est à noter que cela peut être réalisé de façon analytique car il s'agit d'un polynôme de degré 4. On

retiendra alors celle des valeurs de  $\xi$  donnant la plus forte valeur du contraste. Finalement [COM94],  $t$  est obtenue en résolvant l'équation

$$t^2 - \xi t - 1 = 0$$

et en retenant la valeur de  $t$  comprise dans l'intervalle  $] -1, 1]$ .

## 2.2.2 Deuxième famille

Pour la maximisation de  $\mathcal{K}_p(\mathbf{y})$  on suppose toujours qu'un blanchiment des observations a été réalisé. Cependant, pour des raisons de simplicité, la matrice unitaire  $\mathbf{H}$  de séparation va être paramétrée ainsi

$$\mathbf{H} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \quad (2.24)$$

où  $c = \cos \theta$  et  $s = \sin \theta$ .

Maintenant on a

$$\mathbf{Y}_{1,1,1,1} = c^4 \mathbf{X}_{1,1,1,1} + 4c^3 s \mathbf{X}_{1,1,1,2} + 6c^2 s^2 \mathbf{X}_{1,1,2,2} + 4cs^3 \mathbf{X}_{1,2,2,2} + s^4 \mathbf{X}_{2,2,2,2} \quad (2.25)$$

$$\mathbf{Y}_{2,2,2,2} = s^4 \mathbf{X}_{1,1,1,1} - 4s^3 c \mathbf{X}_{1,1,1,2} + 6s^2 c^2 \mathbf{X}_{1,1,2,2} - 4sc^3 \mathbf{X}_{1,2,2,2} + c^4 \mathbf{X}_{2,2,2,2} \quad (2.26)$$

$$\begin{aligned} \mathbf{Y}_{1,1,2,2} = & c^2 s^2 \mathbf{X}_{1,1,1,1} + 2(c s^3 - c^3 s) \mathbf{X}_{1,1,1,2} + (c^4 - 4c^2 s^2 + s^4) \mathbf{X}_{1,1,2,2} \\ & + 2(c^3 s - c s^3) \mathbf{X}_{1,2,2,2} + c^2 s^2 \mathbf{X}_{2,2,2,2} \end{aligned} \quad (2.27)$$

En introduisant les notations suivantes

$$T_1 = \mathbf{X}_{1,1,1,1} + \mathbf{X}_{2,2,2,2}$$

$$T_2 = \mathbf{X}_{1,1,2,2}$$

$$T_3 = \mathbf{X}_{1,1,1,2} - \mathbf{X}_{1,2,2,2}$$

et en utilisant (2.25), (2.26), (2.27), on obtient

$$\varepsilon \mathcal{K}_p(\mathbf{y}) = (c^4 + s^4 + 2\alpha c^2 s^2) T_1 + (12c^2 s^2 + 2\alpha (c^4 + s^4 - 4c^2 s^2)) T_2 + 4cs (c^2 - s^2) (1 - \alpha) T_3. \quad (2.28)$$

Maintenant quelques calculs élémentaires de trigonométrie conduisent à

$$c^4 + s^4 = c_2^2 + \frac{1}{2} s_2^2 \quad (2.29)$$

$$c^2 s^2 = \frac{1}{4} s_2^2 \quad (2.30)$$

$$cs (c^2 - s^2) = \frac{1}{2} c_2 s_2 \quad (2.31)$$

où  $c_2 = \cos 2\theta$  et  $s_2 = \sin 2\theta$ . En introduisant (2.29), (2.30), (2.31) dans (2.28) on obtient

$$\varepsilon \mathcal{K}_p(\mathbf{y}) = c_2^2 (T_1 + 2\alpha T_2) + s_2^2 \left( \frac{1}{2} (1 + \alpha) T_1 + (3 - \alpha) T_2 \right) + 2c_2 s_2 (1 - \alpha) T_3. \quad (2.32)$$

Ainsi,  $\mathcal{K}_p(\mathbf{y})$  peut se mettre sous la forme

$$\mathcal{K}_p(\mathbf{y}) = \mathbf{u}_\theta^T \mathbf{A} \mathbf{u}_\theta \quad (2.33)$$

où  $\mathbf{u}_\theta^T = (\cos 2\theta \quad \sin 2\theta)$  et où  $\mathbf{A}$  est une matrice (2,2) réelle symétrique définie par

$$A_{1,1} = \varepsilon (T_1 + 2\alpha T_2)$$

$$A_{1,2} = \varepsilon (1 - \alpha) T_3$$

$$A_{2,2} = \varepsilon \left( \frac{1}{2} (1 + \alpha) T_1 + (3 - \alpha) T_2 \right)$$

Donc la valeur de  $\theta$  maximisant  $\mathcal{K}_p(\mathbf{y})$  peut maintenant être facilement trouvée en calculant le vecteur propre normalisé de  $\mathbf{A}$  associé à la plus grande des valeurs propres.

Le signe de ce vecteur peut être fixé en imposant  $\theta$  dans l'intervalle  $[-\frac{\pi}{4}, \frac{\pi}{4}]$ . Cependant

on peut toujours écrire

$$\mathbf{u}_\theta^T \mathbf{A} \mathbf{u}_\theta = C + D \cos(4(\theta - \gamma)) \quad (2.34)$$

où

$$C = \frac{A_{1,1} + A_{2,2}}{2}$$

est un terme constant puisque ne dépendant pas de  $\theta$  et où

$$D = \sqrt{\left(\frac{A_{1,1} + A_{2,2}}{2}\right)^2 + A_{1,2}^2}$$

est un terme positif.

L'angle  $\gamma$  dans (2.34) vaut

$$\gamma = \frac{1}{4} \arctan\left(A_{1,2}, \frac{1}{2}(A_{1,1} - A_{2,2})\right)$$

où la valeur de  $\arctan(y, x)$  est, par définition, l'unique angle  $\beta \in ]-\pi, \pi]$  pour lequel on

a

$$\cos \beta = \frac{x}{\sqrt{x^2 + y^2}}$$

$$\sin \beta = \frac{y}{\sqrt{x^2 + y^2}}.$$

Rappelons que l'on recherche une valeur de  $\theta$  notée  $\theta_{opt}$  qui maximise le contraste  $\mathcal{K}_p(\mathbf{y})$  écrit comme  $\mathbf{u}_\theta^T \mathbf{A} \mathbf{u}_\theta$  dans (2.34). Comme  $C$  et  $D$  sont constants et que  $D$  est positif, le maximum du contraste correspond à la valeur maximum du cosinus qui est obtenu lorsque

$$\theta_{opt} = \gamma.$$



## 2.3 Quelques simulations numériques

Dans ce paragraphe, nous allons illustrer le comportement de ces deux algorithmes d'analyse en composantes indépendantes grâce à quelques simulations numériques. Nous noterons **ICA1G** et **ICA2G** les algorithmes correspondants aux algorithmes de Jacobi basés respectivement sur les critères  $\mathcal{K}_p(\mathbf{y})$  (2.14) et  $\mathcal{J}_p(\mathbf{y})$  (2.7).

Nous avons vu que l'algorithme **ICA2G** regroupe les algorithmes d'analyse en composantes indépendantes déjà existants comme les algorithmes **ICA2** de Comon [COM94], **JADE** de Cardoso [CAR93], et **STOTD** de De Lathauwer [DEL97] dans un modèle plus général à deux coefficients.

Nous rappelons la valeur de ces coefficients ci-après

- $\alpha_2 = 0$  et  $\alpha_1 = 0$ , pour **ICA2**
- $\alpha_2 = 0$  et  $\alpha_1 = 0.5$ , pour **STOTD**
- $\alpha_2 = 1$  et  $\alpha_1 = 1$ , pour **JADE**

Nous allons donc comparer les performances relatives des trois critères relatifs à **ICA2**, **STOTD** et **JADE**, avec un nouveau correspondant aux coefficients  $\alpha_1 = 0$  et  $\alpha_2 = 1$  et que nous appellerons **COMP**. Nous présenterons par la suite quelques résultats relatifs à l'algorithme **ICA1G**, modèle à un coefficient.

### 2.3.1 Cadre des simulations

Pour effectuer ces simulations nous avons utilisé des sources paramétrées grâce à un coefficient qui nous servira à choisir la valeur du kurtosis, nous les appellerons sources "**CP**". Nous allons donc utiliser un signal pouvant prendre les trois valeurs suivantes  $[-1, 0, \tau]$  avec les probabilités respectives suivantes :  $\frac{1}{\tau+1}$ ,  $\frac{\tau-1}{\tau}$ ,  $\frac{1}{\tau(\tau+1)}$ , ceci avec  $\tau \geq 1$ .

- Lorsque le coefficient  $\tau$  est égal à un, le cumuland d'ordre trois est nul et le kurtosis est égal à -2;
- lorsque  $\tau$  est égal à 2, le cumuland d'ordre trois est égal à 1 et le kurtosis est nul ;
- pour  $\tau$  supérieur à 2, le kurtosis devient positif.

La figure (Fig. 2.1) représente l'évolution des cumulants d'ordre trois et quatre des sources ainsi créés en fonction de la valeur du coefficient  $\tau$ . Nous avons choisi d'utiliser des matrices

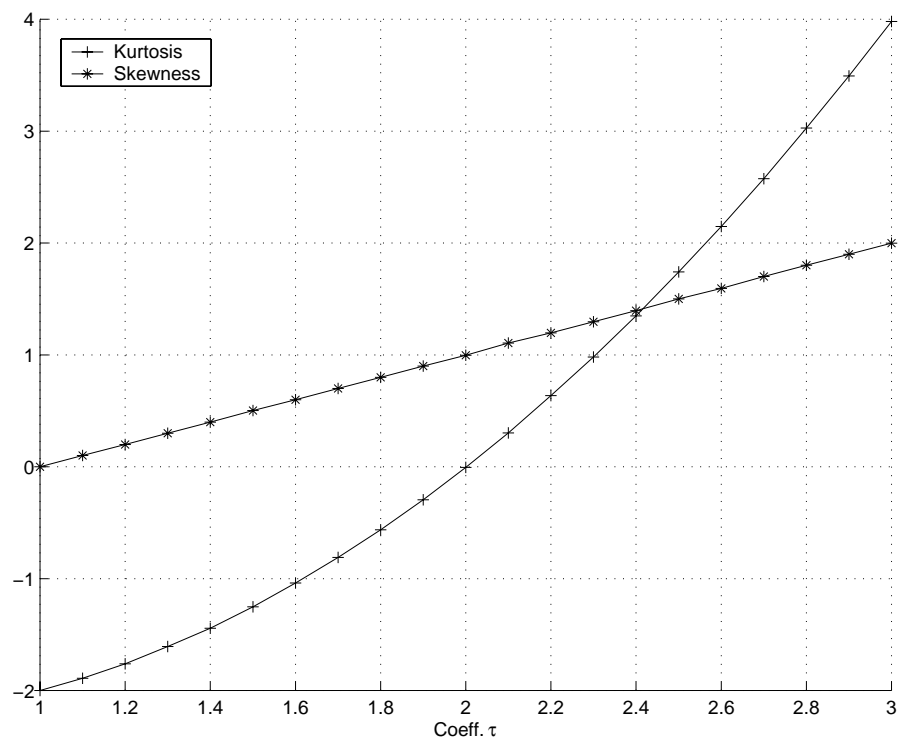


FIG. 2.1 – Valeur des kurtosis et skewness d'une source CP

de mélange dont les coefficients sont tirés de façon aléatoire entre -1 et 1. La procédure de simulation consistera à tirer une matrice de façon aléatoire puis avec cette même matrice de mélange, nous allons effectuer une procédure de Monte-Carlo consistant à effectuer un certain nombre d'expériences et à en donner la moyenne et la variance afin de se préserver des cas particuliers et de donner des courbes représentatives.

Pour pouvoir comparer les performances de plusieurs algorithmes lors de simulations numériques, nous avons besoin d'un outil nous permettant de quantifier la séparation de sources. Cet outil doit être adapté aux problèmes spécifiques du problème posé, il doit en particulier tenir compte des indéterminations inhérentes au modèle (cf. paragraphe 1.1.3). Nous devons donc utiliser une fonctionnelle mesurant la distance entre deux matrices modulo un facteur multiplicatif de la forme  $DP$ . Cette fonctionnelle appelée indice de

performances que l'on considérera est

$$\text{ind}(\mathbf{S}) = \frac{1}{2} \left[ \sum_i \left( \sum_j \frac{|S_{i,j}|^2}{\max_l |S_{i,l}|^2} - 1 \right) + \sum_j \left( \sum_i \frac{|S_{i,j}|^2}{\max_l |S_{i,l}|^2} - 1 \right) \right]. \quad (2.35)$$

Il suffit donc d'appliquer cette fonctionnelle à la matrice globale  $\mathbf{S}$  du système de mélange séparation pour en mesurer les performances. En effet, cet indice est toujours positif et est nul lorsque l'équation (1.6) est vérifiée, *i.e.* lorsque l'on a la séparation. Ainsi une petite valeur de celui ci indique la proximité d'une solution séparante.

## 2.3.2 Les résultats

### Influence de la gaussianité sur les performances

Nous allons dans un premier temps observer l'évolution de l'indice de performance des algorithmes **ICA2G** et **ICA1G** par rapport au coefficient  $\tau$  régissant la valeur du cumulants des sources CP. Pour cela nous avons effectué une procédure de Monte-Carlo sur 50 réalisations dans le cas de trois sources (Fig. 2.2, Fig. 2.3), cinq sources (Fig. 2.4, Fig. 2.5) et sept sources (Fig. 2.6, Fig. 2.7). Le nombre d'échantillons utilisé pour les sources est de 1000 points et la matrice de mélange est

$$G_{31} = \begin{bmatrix} -0.0229 & 0.9597 & -0.9656 \\ 0.3519 & -0.0585 & -0.6217 \\ -0.0131 & 0.1465 & 0.3369 \end{bmatrix}$$

pour trois sources. Dans le cas cinq sources nous avons

$$G_{51} = \begin{bmatrix} -0.0873 & 0.2093 & 0.0876 & -0.2902 & -0.6403 \\ -0.2903 & 0.0370 & -0.8352 & 0.4756 & -0.8792 \\ 0.8734 & -0.9950 & -0.6726 & -0.9751 & 0.0843 \\ 0.9911 & -0.3661 & -0.5412 & 0.3053 & -0.9717 \\ -0.5380 & -0.4770 & -0.9380 & 0.3174 & 0.4018 \end{bmatrix}$$

et pour sept sources

$$G_{71} = \begin{bmatrix} 0.9003 & -0.9630 & -0.6475 & -0.2943 & -0.4556 & -0.1627 & 0.3626 \\ -0.5377 & 0.6428 & -0.1886 & 0.6263 & -0.6024 & 0.6924 & -0.2410 \\ 0.2137 & -0.1106 & 0.8709 & -0.9803 & -0.9695 & 0.0503 & 0.6636 \\ -0.0280 & 0.2309 & 0.8338 & -0.7222 & 0.4936 & -0.5947 & 0.0056 \\ 0.7826 & 0.5839 & -0.1795 & -0.5945 & -0.1098 & 0.3443 & 0.4189 \\ 0.5242 & 0.8436 & 0.7873 & -0.6026 & 0.8636 & 0.6762 & -0.1422 \\ -0.0871 & 0.4764 & -0.8842 & 0.2076 & -0.0680 & -0.9607 & -0.3908 \end{bmatrix}$$

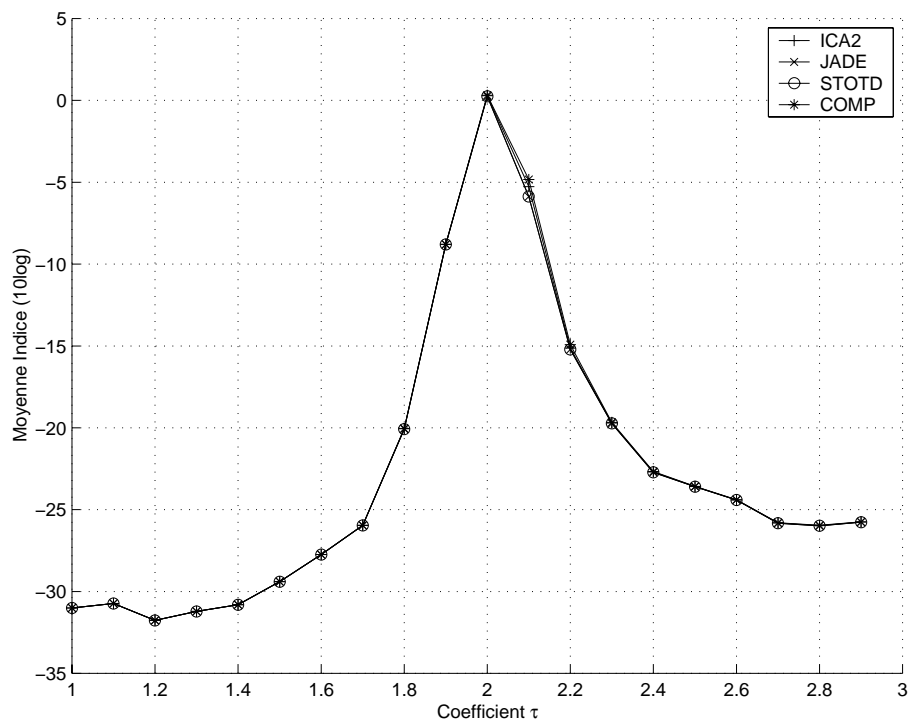


FIG. 2.2 – ICA2G : Cas de trois signaux sources CP, matrice de mélange  $G_{31}$

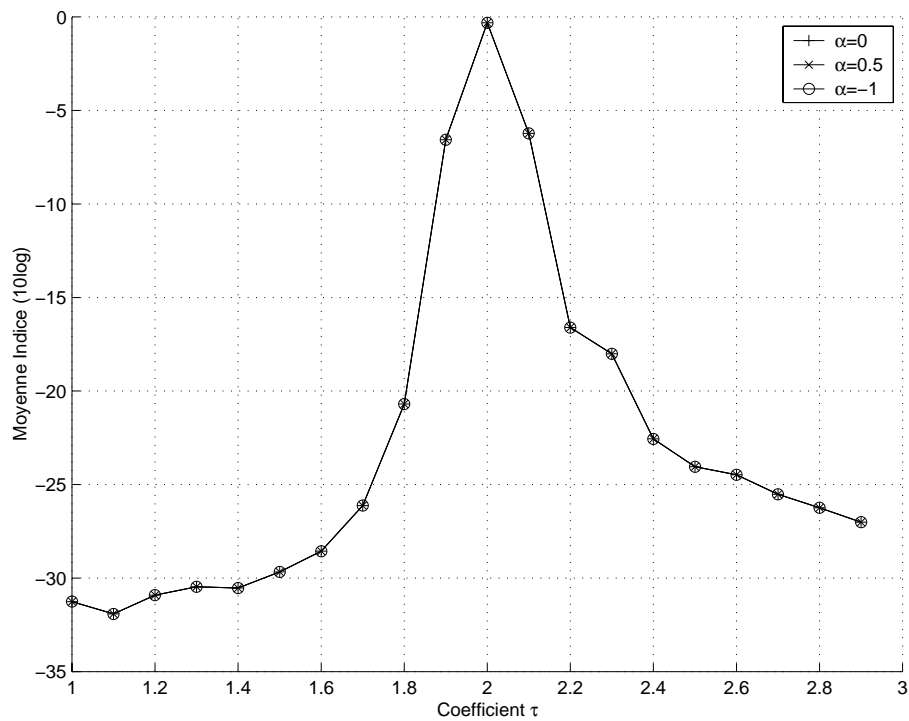


FIG. 2.3 – ICA1G : Cas de trois signaux sources CP, matrice de mélange  $G_{31}$

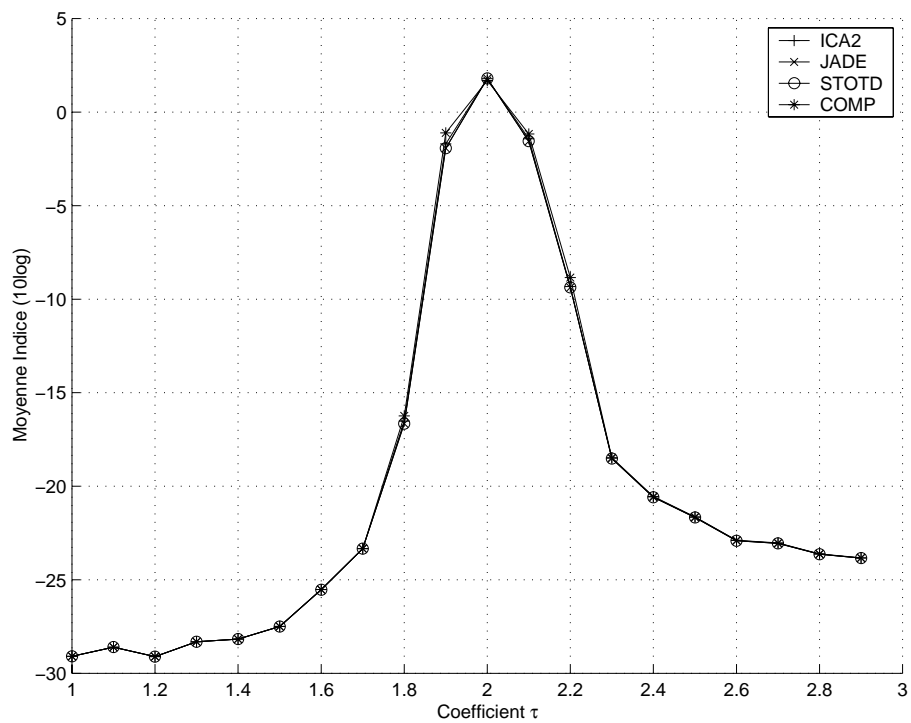


FIG. 2.4 – ICA2G : Cas de cinq signaux sources CP, matrice de mélange  $G_{51}$

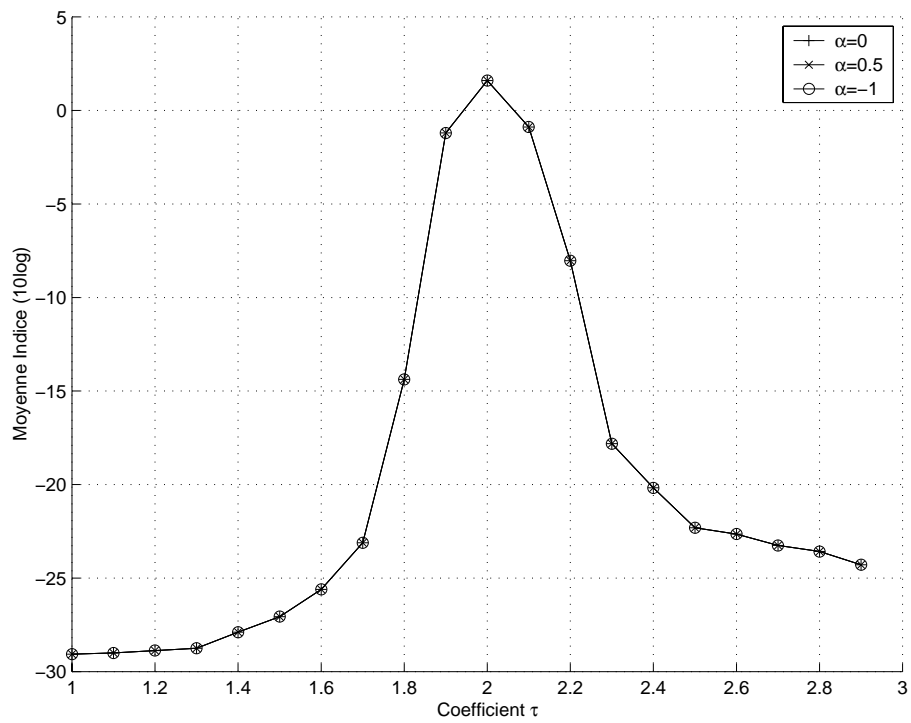


FIG. 2.5 – ICA1G : Cas de cinq signaux sources CP, matrice de mélange  $G_{51}$

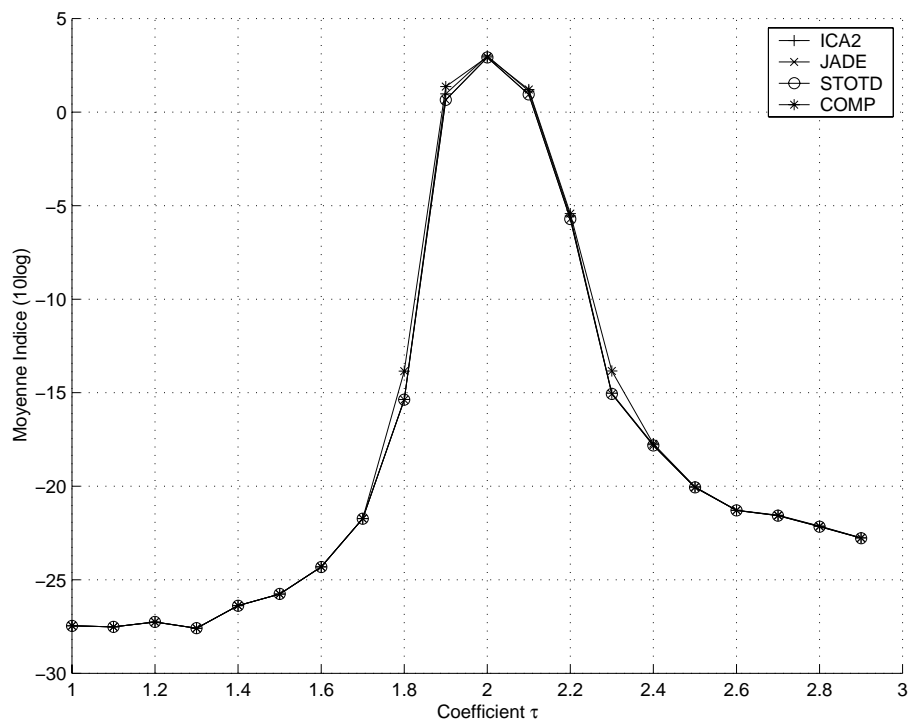


FIG. 2.6 – ICA2G : Cas de sept signaux sources CP, matrice de mélange  $G_{71}$

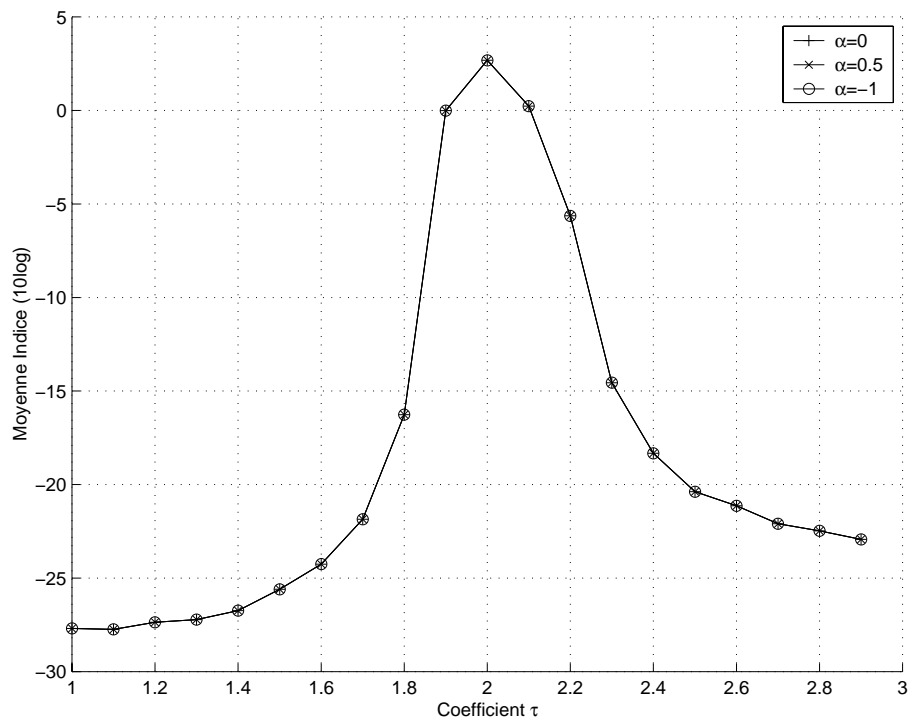


FIG. 2.7 – ICA1G : Cas de sept signaux sources CP, matrice de mélange  $G_{71}$

La première conclusion que l'on peut tirer de ces simulations numérique est que ces deux algorithmes généralisés **ICA1G** et **ICA2G** ne sont pas efficaces sur des sources de cumulants d'ordre quatre nul ( $\tau = 2$ ), ce qui n'est guère surprenant (cf. paragraphe 1.1.4). Nous pouvons remarquer la similitude de comportement entre l'ensemble des algorithmes présentés. Il est à noter principalement que les cumulants croisés semblent avoir une faible influence sur les performances des algorithmes.

### L'algorithme **ICA2G**

Les figures qui vont suivre représentent l'évolution de l'indice de performance tout au long des relaxations de l'algorithme de **ICA2G** pour les quatre variantes décrites au début de ce paragraphe. Ceci dans les cas trois, cinq et sept sources et avec des sources CP à kurtosis négatif ( $\tau = 1, 7$ ) et des sources CP à kurtosis positif ( $\tau = 2, 3$ ). Ici, nous avons effectué une procédure de Monte-Carlo sur 250 réalisations.

Dans le cas trois sources, avec  $\tau = 1, 7$  (FIG. 2.8) et  $\tau = 2, 3$  (FIG. 2.9), nous avons

$$G_{32} = \begin{bmatrix} 0.9003 & -0.0280 & -0.0871 \\ -0.5377 & 0.7826 & -0.9630 \\ 0.2137 & 0.5242 & 0.6428 \end{bmatrix} \quad (2.36)$$

Pour cinq sources, avec  $\tau = 1, 7$  (FIG. 2.10) et  $\tau = 2, 3$  (FIG. 2.11), nous avons

$$G_{52} = \begin{bmatrix} -0.7654 & 0.3508 & 0.7128 & -0.6094 & -0.5001 \\ 0.6772 & 0.0135 & 0.5697 & -0.2152 & -0.7827 \\ 0.1998 & 0.9621 & -0.5215 & -0.2022 & 0.6160 \\ 0.6845 & -0.3770 & 0.0241 & 0.3584 & -0.8136 \\ 0.0195 & 0.7672 & -0.9691 & 0.0294 & -0.2994 \end{bmatrix} \quad (2.37)$$



Pour sept sources, avec  $\tau = 1, 7$  (FIG. 2.12) et  $\tau = 2, 3$  (FIG. 2.13), nous avons

$$G_{72} = \begin{bmatrix} 0.9003 & -0.9630 & -0.6475 & -0.2943 & -0.4556 & -0.1627 & 0.3626 \\ -0.5377 & 0.6428 & -0.1886 & 0.6263 & -0.6024 & 0.6924 & -0.2410 \\ 0.2137 & -0.1106 & 0.8709 & -0.9803 & -0.9695 & 0.0503 & 0.6636 \\ -0.0280 & 0.2309 & 0.8338 & -0.7222 & 0.4936 & -0.5947 & 0.0056 \\ 0.7826 & 0.5839 & -0.1795 & -0.5945 & -0.1098 & 0.3443 & 0.4189 \\ 0.5242 & 0.8436 & 0.7873 & -0.6026 & 0.8636 & 0.6762 & -0.1422 \\ -0.0871 & 0.4764 & -0.8842 & 0.2076 & -0.0680 & -0.9607 & -0.3908 \end{bmatrix} \quad (2.38)$$

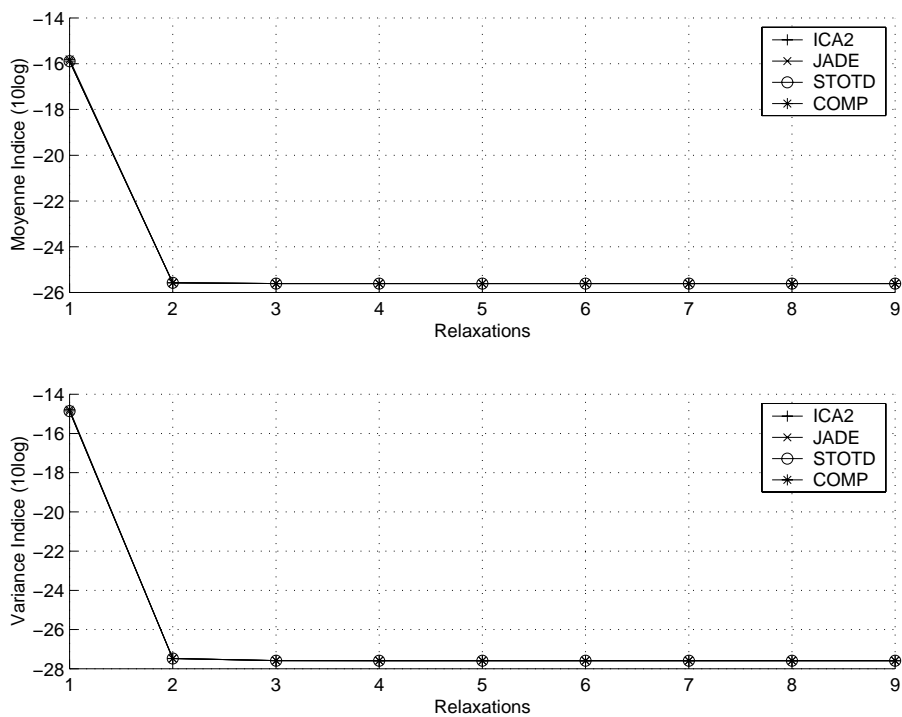


FIG. 2.8 – ICA2G : trois signaux sources CP ( $\tau = 1, 7$ ), matrice de mélange  $G_{32}$

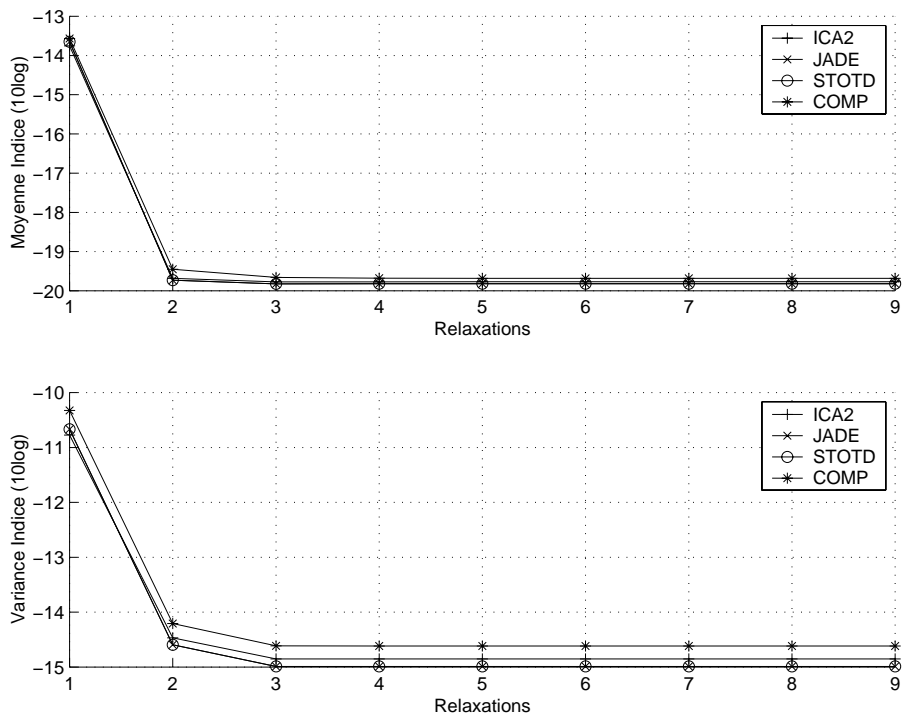


FIG. 2.9 – ICA2G : trois signaux sources CP ( $\tau = 2, 3$ ), matrice de mélange  $G_{32}$

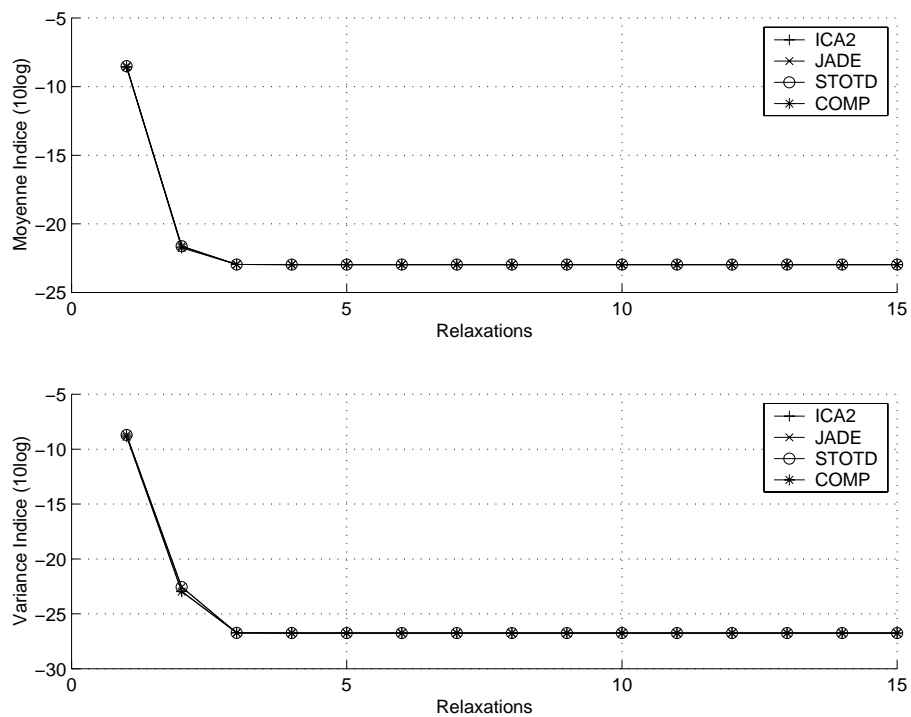


FIG. 2.10 – ICA2G : cinq signaux sources CP ( $\tau = 1, 7$ ), matrice de mélange  $G_{52}$

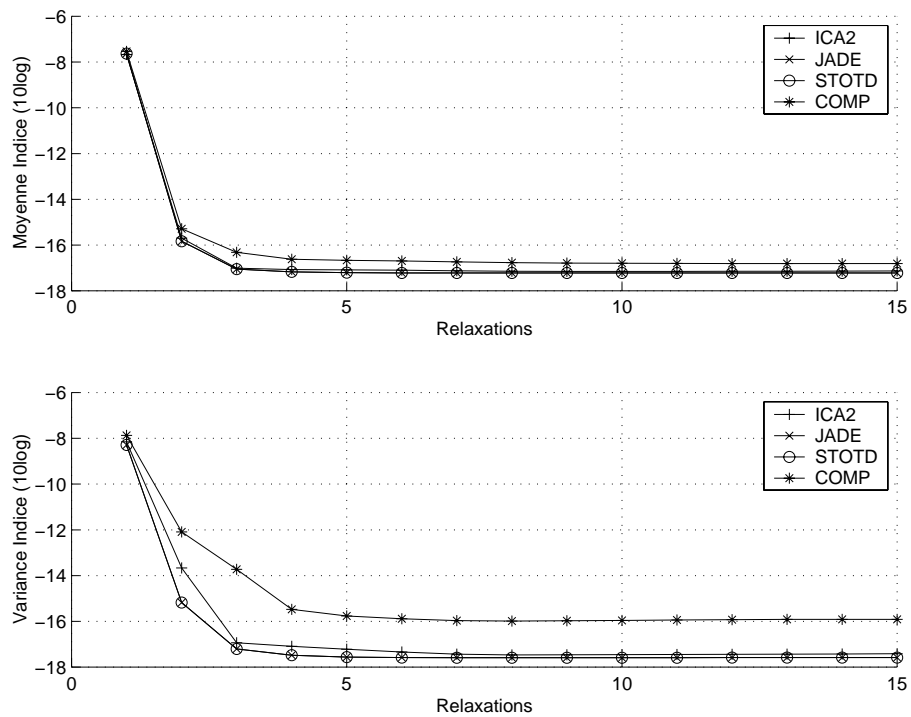


FIG. 2.11 – ICA2G : cinq signaux sources CP ( $\tau = 2, 3$ ), matrice de mélange  $G_{52}$

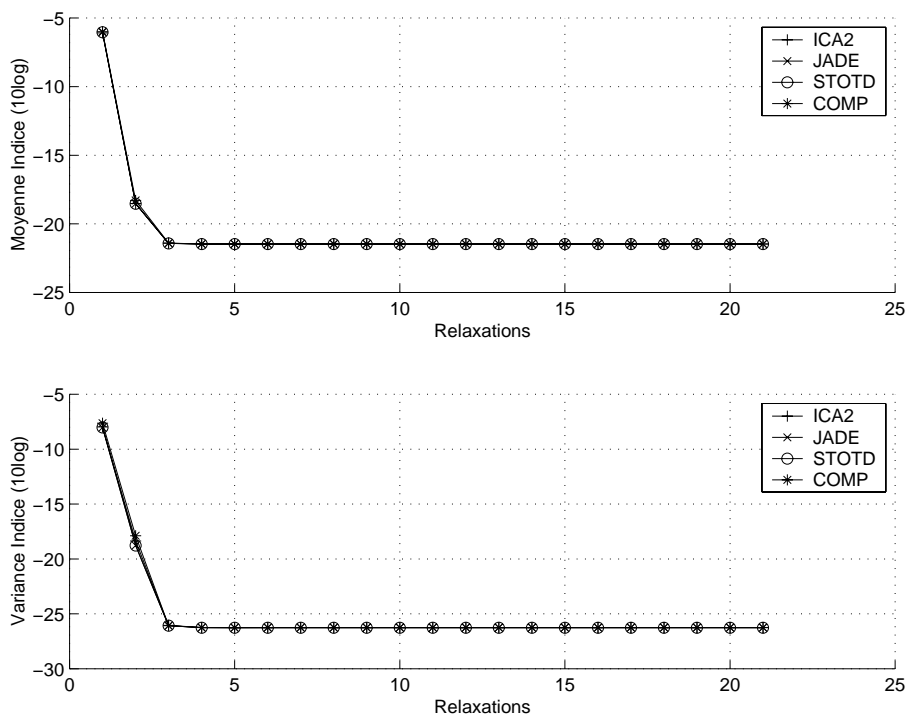


FIG. 2.12 – ICA2G : sept signaux sources CP ( $\tau = 1, 7$ ), matrice de mélange  $G_{72}$

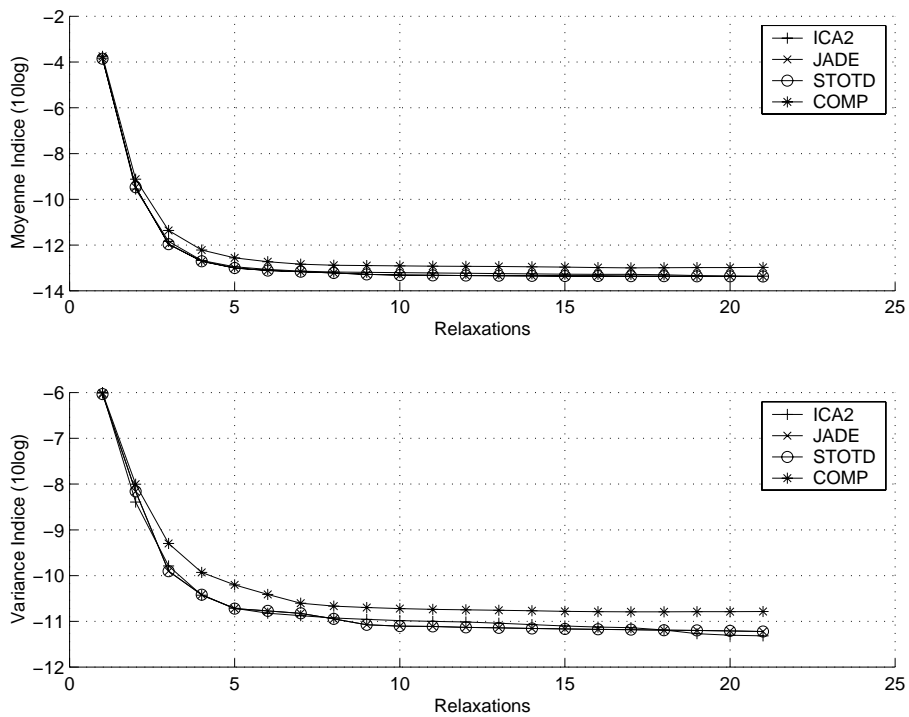


FIG. 2.13 – ICA2G : sept signaux sources CP ( $\tau = 2, 3$ ), matrice de mélange  $G_{72}$

Les figures précédentes nous montrent l'évolution de l'indice de performance au cours des relaxations pour les quatre algorithmes **ICA2**, **JADE**, **STOTD** et **COMP**. Chaque relaxation correspondant à la séparation d'un couple de sources d'entrées, nous pouvons remarquer qu'après un nombre de relaxations réduit (cinq pour le cas sept sources), l'algorithme a convergé vers une solution de séparation. Un autre point à remarquer est le fait que les courbes correspondant aux fonctions de contrastes des algorithmes **ICA2**, **JADE**, **STOTD** et **COMP** sont très proches, ce qui rejoint les conclusions décrites dans [CAP97]. La différence la plus importante intervient lorsque l'on considère des sources CP avec un kurtosis positif ( $\tau = 2, 3$ ).

## 2.4 Conclusion

L'originalité dans ce chapitre réside dans l'introduction de deux familles de contrastes généralisés, nous avons constatés que certains contrastes déjà existants contenaient des cumulants croisés, l'idée était de créer une famille de contraste prenant en compte les auto-cumulants ainsi que les cumulants croisés, ceux ci étant pondérés par un coefficient. Ainsi en jouant sur la valeur de ces coefficients, nous retrouvons d'une part certains contrastes existants et d'autre part, nous sommes à même d'en proposer de nouveaux.

Nous avons ensuite développé deux algorithmes basés sur ces deux familles de contrastes du même type que l'algorithme **ICA2** c'est à dire du type Jacobi. Les simulations informatiques effectuées montrent la convergence de ces algorithmes dans plusieurs cas de figures.

# Chapitre 3

## Optimisation directe de contrastes

Dans ce chapitre, nous allons nous intéresser à la séparation de sources par optimisation de fonctions de contraste, pour cela, nous allons adopter une philosophie quelque peu différente de celle envisagée classiquement dans la littérature. La plupart des fonctions de contrastes fonctionnent sur des vecteurs blancs, ainsi la plupart des algorithmes de séparation de sources procèdent en deux étapes avec un pré-blanchiment des observations, nous allons montrer que cette première étape n'est pas toujours nécessaire et que l'on peut envisager d'utiliser ces fonctions de contraste sans blanchiment explicite des observations. Pour cela, nous allons appliquer des méthodes d'optimisation non linéaires avec contrainte, il en existe plusieurs sortes, nous étudierons essentiellement trois méthodes duales et une méthode directe. Chaque méthode conduisant à un algorithme dont nous donnerons quelques simulations numériques.

### 3.1 Introduction

La très grande majorité des contrastes, tout au moins l'ensemble des contrastes présentés jusqu'ici, ne sont définis que par l'intermédiaire de vecteurs blancs dont on rappelle que ce sont des vecteurs dont la matrice de covariance est égale à l'identité. Ainsi la résolution du problème de la séparation de sources par l'optimisation de tels contrastes, notés

$\mathcal{I}(\cdot)$  a l'objectif suivant :

$$\max_{\mathbf{H}} \mathcal{I}(\mathbf{y}) \quad \text{sous la contrainte} \quad \mathbf{E}[\mathbf{y}\mathbf{y}^T] = \mathbf{I} \quad (3.1)$$

Classiquement l'objectif précédent est réalisé en 2 étapes :

- étape 1 : réaliser un blanchiment des observation :  $\mathbf{x}_b = \mathbf{B}\mathbf{x}$   
de telle façon que  $\mathbf{E}[\mathbf{x}_b\mathbf{x}_b^T] = \mathbf{I}$
- étape 2 : maximiser le contraste sur l'ensemble des matrices unitaires

L'étape 2 est souvent réalisée grâce à une paramétrisation par rotation plane des matrices unitaires (cf. Chapitre 2). Cependant on peut remarquer que si l'on considère  $\mathbf{y} = \mathbf{H}\mathbf{x}$  (1.3) dans la contrainte, cette dernière devient

$$\mathbf{H}\mathbf{R}_x\mathbf{H}^T = \mathbf{I} \quad (3.2)$$

Ainsi, on peut rendre implicite l'étape de blanchiment en maximisant le contraste sur l'ensemble des matrices satisfaisant (3.2). L'objectif devient alors

$$\max_{\mathbf{H}} \mathcal{I}(\mathbf{H}\mathbf{x}) \quad \text{sous la contrainte} \quad \mathbf{H}\mathbf{R}_x\mathbf{H}^T = \mathbf{I} \quad (3.3)$$

Ce type de problème d'optimisation est relativement classique dans la littérature de programmation mathématique, voir par exemple [MIN83].

En effet la programmation mathématique a pour objet l'étude des problèmes d'optimisation ainsi que la conception et la mise en œuvre des algorithmes de résolution. Il en existe deux grandes catégories : la programmation linéaire liée à l'optimisation de fonctions linéaires sous contraintes linéaires et la programmation non linéaire dans les autres cas.

Ainsi l'objectif (3.3) fait partie des problèmes d'optimisation non linéaire avec contraintes qui ne sont pas des problèmes faciles en général. En optimisation non linéaire avec contraintes, on distingue deux types de méthodes : les méthodes duales et les méthodes directes. Parmi les méthodes duales on trouve entre autres les méthodes Lagrangiennes et

les méthodes de pénalisation qui sont des méthodes assez répandues. Parmi les méthodes directes, on trouve entre autre les méthodes de changement de variables (c'est le cas par exemple lorsque après blanchiment, on paramétrise la matrice unitaire par rotation plane) et les méthodes de projection.

Dans les paragraphes qui suivent, nous présentons trois méthodes duales et une méthode de projection dont le but est d'atteindre l'objectif (3.3) afin de résoudre le problème de la séparation de sources.

## 3.2 Méthodes duales

L'idée de base des méthodes duales est de transformer le problème initial avec contraintes en un "nouveau" problème sans contraintes. Ainsi un terme dépendant des contraintes est ajouté au critère de différentes façons. La résolution du nouveau problème se fait généralement de manière itérative grâce à une procédure de type gradient et une solution respectant les contraintes n'est atteinte qu'à la fin de la convergence.

On peut maintenant remarquer que les contraintes (3.3) peuvent toujours s'écrire

$$\mathcal{H}(\mathbf{H}) = 0 \tag{3.4}$$

avec  $\mathcal{H}(\mathbf{H}) = \overline{\mathcal{H}}(\mathbf{H}) - \mathbf{I}$

et

$$\overline{\mathcal{H}}(\mathbf{H}) = \mathbf{H}\mathbf{R}_x\mathbf{H}^T \tag{3.5}$$

### 3.2.1 Méthode Lagrangienne classique

Dans la méthode Lagrangienne classique, les contraintes sont incluses additivement dans la fonction à optimiser par l'intermédiaire de paramètres appelés coefficients ou



multiplicateurs de Lagrange. Cela peut s'écrire ainsi

$$\mathcal{L}(\mathbf{H}, \mathbf{L}) = \mathcal{I}(\mathbf{H}) + \langle \mathbf{L}, \mathcal{H}(\mathbf{H}) \rangle \quad (3.6)$$

où pour simplifier, on a fait un (léger) abus de notation en écrivant  $\mathcal{I}(\cdot)$  comme directement fonction de  $\mathbf{H}$ .

La matrice symétrique  $\mathbf{L}$  regroupe l'ensemble des multiplicateurs de Lagrange et  $\langle \mathbf{A}, \mathbf{B} \rangle = \text{Trace}[\mathbf{A}\mathbf{B}^T]$  est le produit scalaire classique (Euclidien) de deux matrices.

A l'optimum, on doit avoir

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial \mathbf{H}} = 0 \\ \frac{\partial \mathcal{L}}{\partial \mathbf{L}} = 0 \end{cases} \quad (3.7)$$

Ces deux conditions constituent des conditions nécessaires d'optimalité pour le problème d'optimisation contraint. En utilisant (3.6), on obtient aisément

$$\frac{\partial \mathcal{L}}{\partial \mathbf{H}} = \frac{\partial \mathcal{I}}{\partial \mathbf{H}} + (\mathbf{L} + \mathbf{L}^T) \mathbf{H} \mathbf{R}_x = \frac{\partial \mathcal{I}}{\partial \mathbf{H}} + 2\mathbf{L} \mathbf{H} \mathbf{R}_x$$

car  $\mathbf{L}$  est symétrique et de plus

$$\frac{\partial \mathcal{L}}{\partial \mathbf{L}} = \mathbf{H} \mathbf{R}_x \mathbf{H}^T - \mathbf{I}$$

Ainsi les valeurs optimales de  $\mathbf{H}$  et  $\mathbf{L}$  sont solutions du système

$$\frac{\partial \mathcal{I}}{\partial \mathbf{H}} + 2\mathbf{L} \mathbf{H} \mathbf{R}_x = 0 \quad (3.8)$$

$$\mathbf{H} \mathbf{R}_x \mathbf{H}^T - \mathbf{I} = 0 \quad (3.9)$$

Maintenant en multipliant à droite les deux membres de (3.8) par  $\mathbf{H}^T$  et en utilisant (3.9), on obtient

$$2\mathbf{L} = -\mathcal{I}'(\mathbf{H}) \quad (3.10)$$

où

$$\mathcal{I}'(\mathbf{H}) = \frac{\partial \mathcal{I}}{\partial \mathbf{H}} \mathbf{H}^T \quad (3.11)$$

Il est à noter que  $\mathcal{I}'(\mathbf{H})$  est appelé gradient relatif dans [CAR96]. En rappelant que  $\mathbf{L}$  est une matrice symétrique et en utilisant (3.10), une valeur optimale de  $\mathbf{H}$  doit donc satisfaire

$$\mathcal{I}'(\mathbf{H}) = (\mathcal{I}'(\mathbf{H}))^T \quad (3.12)$$

Dit autrement, les matrices  $\mathbf{H}$  que l'on recherche, doivent rendre le gradient relatif de  $\mathcal{I}(\cdot)$  symétrique. Malheureusement, il semble difficile de résoudre analytiquement l'équation (3.12) en l'inconnue  $\mathbf{H}$  et donc on utilisera une procédure itérative afin de résoudre le système (3.7).

**Remarque.** Puisque l'on cherche une matrice  $\mathbf{H}$  rendant  $\mathcal{I}'(\cdot)$  symétrique, on pourrait construire une procédure qui chercherait à annuler la partie antisymétrique de  $\mathcal{I}'(\cdot)$ . Il est alors intéressant de remarquer qu'un tel objectif est recherché par la première partie de l'algorithme EASI [CAR96] et que ce fait est déduit d'une toute autre façon. L'approche Lagrangienne permet alors d'en donner une nouvelle interprétation.

Une description quelque peu formelle d'une itération de la procédure Lagrangienne de résolution des conditions nécessaires d'optimalité est la suivante

- étape 1 : Étant donné une matrice  $\mathbf{L}(k)$ , trouver une solution  $\mathbf{H}(k+1)$  au problème

$$\max_{\mathbf{H}} \mathcal{L}(\mathbf{H}(k), \mathbf{L}(k))$$

- étape 2 : Si  $\mathcal{H}(\mathbf{H}(k+1)) = 0$  alors une solution optimale est trouvée et l'on arrête, sinon

$$\mathbf{L}(k+1) = \mathbf{L}(k) - \mu_l \mathcal{H}(\mathbf{H}(k+1)) \quad (3.13)$$

où  $\mu_l > 0$  et ensuite, on repart à l'étape 1.

Il est à noter que le coefficient  $\mu_l$  dans (3.13) pondérant le terme de contrainte dans l'adaptation de  $\mathbf{L}$ , doit en général dépendre des itérations ainsi  $\mu_l(k+1) \geq \mu_l(k)$ . La séquence de paramètre  $\mu_l(k)$  peut être soit présélectionnée ou bien déterminée sur la base des résultats obtenus durant les itérations. De plus, en général, afin d'assurer la convergence,  $\mu_l(k)$  devrait tendre vers l'infini. Ceci risquera de poser des problèmes de stabilité numérique. En général, pour des problèmes d'optimisation non linéaire, la méthode du Lagrangien classique n'est pas reconnue comme une très bonne méthode de part ses nombreux problèmes de convergence [MIN83], comme on le constatera au paragraphe 3.4 présentant quelques simulations numériques. Ainsi elle est souvent associée à d'autres méthodes comme les méthodes de pénalisation présentées au paragraphe suivant, possédant elles de bonnes propriétés de convergence globale.

### 3.2.2 Deux méthodes de pénalisation

Une autre idée simple afin d'obtenir un critère sans contrainte consiste à ajouter au critère initial un terme de pénalisation quadratique qui sera d'autant plus élevé que la matrice recherchée ne satisfera pas les contraintes. Ainsi par exemple la fonction de pénalisation quadratique s'écrit  $\langle \mathcal{H}(\mathbf{H}), \mathcal{H}(\mathbf{H}) \rangle$  et le nouveau critère s'écrit

$$\mathcal{P}_q(\mathbf{H}) = \mathcal{I}(\mathbf{H}) - \frac{1}{4} c_q \langle \mathcal{H}(\mathbf{H}), \mathcal{H}(\mathbf{H}) \rangle \quad (3.14)$$

où  $c_q$  est un paramètre positif associé à cette méthode dite de pénalisation quadratique. Cependant, on peut envisager des façons "non quadratiques" pour la prise en compte des contraintes. Nous en présenterons une maintenant qui a été utilisée dans [CAR96] pour l'algorithme EASI. Pour une matrice  $(N, N)$  symétrique  $\mathbf{A}$ , la mesure suivante

$$\text{Kld}(\mathbf{A}) = \text{Trace}[\mathbf{A}] - \log \det(\mathbf{A}) - N$$

est toujours positive et nulle ssi  $\mathbf{A} = \mathbf{I}$ . Ainsi,  $\text{Kld}(\mathbf{A})$  mesure l'écart entre  $\mathbf{A}$  et la matrice identité. Pour notre problème, il s'agit de mesurer l'écart entre  $\overline{\mathcal{H}}(\mathbf{H})$  dans (3.5) et l'identité. On peut donc proposer la fonction de pénalisation  $\text{Kld}(\overline{\mathcal{H}}(\mathbf{H}))$  et le critère

$$\mathcal{P}_d(\mathbf{H}) = \mathcal{I}(\mathbf{H}) - \frac{1}{2}c_d \text{Kld}(\overline{\mathcal{H}}(\mathbf{H})) \quad (3.15)$$

où  $c_d$  est un paramètre positif associé à cette méthode. L'optimisation des deux fonctionnelles précédentes (3.14) et (3.15) peut être réalisée très simplement par exemple par un algorithme de gradient. Pour cela, il est en particulier nécessaire de calculer les gradients des deux fonctions de pénalisation. Après quelques calculs on obtient les gradients relatifs suivants

$$\frac{\partial \langle \mathcal{H}(\mathbf{H}), \mathcal{H}(\mathbf{H}) \rangle}{\partial \mathbf{H}} \mathbf{H}^T = 4(\mathbb{E}[\mathbf{y}\mathbf{y}^T] - \mathbf{I})\mathbb{E}[\mathbf{y}\mathbf{y}^T]$$

et

$$\frac{\partial \text{Kld}(\overline{\mathcal{H}}(\mathbf{H}))}{\partial \mathbf{H}} \mathbf{H}^T = 2(\mathbb{E}[\mathbf{y}\mathbf{y}^T] - \mathbf{I})$$

Cependant, la convergence d'un tel algorithme de gradient requiert d'accroître le coefficient  $c_q$  (ou  $c_d$ ) vers l'infini au cours des itérations. Cela pourra conduire encore à des problèmes de stabilité numérique.

### 3.2.3 Lagrangien augmenté

Une solution afin de palier les problèmes de convergence des deux types de méthodes précédentes est de combiner additivement le Lagrangien classique  $\mathcal{L}(\mathbf{H}, \mathbf{L})$  et une méthode de pénalisation, *i.e.*  $\mathcal{P}_q(\mathbf{H})$  ou  $\mathcal{P}_d(\mathbf{H})$ . Cela introduit les méthodes dites de Lagrangien augmenté. En correspondance avec les deux méthodes de pénalisation, nous introduisons deux méthodes de Lagrangien augmenté par les deux fonctionnelles suivantes

$$\mathcal{L}_q(\mathbf{H}, \mathbf{L}) = \mathcal{I}(\mathbf{H}) + \langle \mathbf{L}, \mathcal{H}(\mathbf{H}) \rangle - \frac{1}{4}c_q \langle \mathcal{H}(\mathbf{H}), \mathcal{H}(\mathbf{H}) \rangle \quad (3.16)$$

et

$$\mathcal{L}_d(\mathbf{H}, \mathbf{L}) = \mathcal{I}(\mathbf{H}) + \langle \mathbf{L}, \mathcal{H}(\mathbf{H}) \rangle - \frac{1}{2}c_d \text{Kld}(\overline{\mathcal{H}}(\mathbf{H})) \quad (3.17)$$

La procédure d'optimisation d'un Lagrangien augmenté *e.g.*  $\mathcal{L}_q(\mathbf{H}, \mathbf{L})$  se fait de la même façon que pour un Lagrangien classique :

- étape 1 : Étant donné une matrice  $\mathbf{L}(k)$ , trouver une solution  $\mathbf{H}(k+1)$  au problème

$$\max_{\mathbf{H}} \mathcal{L}_q(\mathbf{H}(k), \mathbf{L}(k))$$

- étape 2 : Si  $\mathcal{H}(\mathbf{H}(k+1)) = 0$  alors une solution optimale est trouvée et l'on s'arrête, sinon, on itère

$$\mathbf{L}(k+1) = \mathbf{L}(k) - c_q \mathcal{H}(\mathbf{H}(k+1))$$

et l'on repart à l'étape 1.

La différence importante avec les méthodes précédentes et que maintenant il n'est plus théoriquement nécessaire d'accroître le paramètre  $c_q$  vers l'infini afin d'assurer une convergence. Il suffit de le borner à une valeur suffisamment élevée.

### 3.3 Une méthode directe

Même si l'idée est identique, à savoir transformer le problème initial avec contraintes, en un "nouveau" sans contrainte, on n'ajoute maintenant plus de terme au critère. Les contraintes sont imposées de "façon externe", soit grâce à une bonne paramétrisation du problème (par exemple par l'utilisation de rotations planes pour des matrices unitaires) soit par une opération supplémentaire (généralement à chaque itération). Dans les deux cas précédents, une "solution" respectant les contraintes est alors obtenue à chaque itération. Notre but est de proposer une telle méthode afin d'atteindre l'objectif (3.3). Celle ci sera itérative et comportera deux étapes.

#### 3.3.1 Première étape

Une procédure classique et simple afin d'obtenir un maximum (local) d'une fonction objectif est d'utiliser une procédure de gradient conduisant à la récursion classique

$$\mathbf{H}_1 = \mathbf{H} + \lambda \dot{\mathcal{I}}(\mathbf{H}) \quad (3.18)$$

où  $\mathbf{H} = (H_{i,j})$  et  $\mathbf{H}_1$  sont respectivement les matrices de séparation estimées avant et après l'itération,  $\lambda$  un réel positif et

$$\dot{\mathcal{I}}(\mathbf{H}) = \frac{\partial \mathcal{I}(\mathbf{H})}{\partial \mathbf{H}} \quad (3.19)$$

la matrice dont l'élément  $(l, m)$  vaut  $\frac{\partial \mathcal{I}(\mathbf{H})}{\partial H_{l,m}}$ . Cependant, il est aussi possible de considérer un autre gradient dit relatif [CAR96] conduisant à la récursion

$$\mathbf{H}_1 = (\mathbf{I} + \lambda \mathcal{I}'(\mathbf{H})) \mathbf{H} \quad (3.20)$$

où  $\mathcal{I}'(\mathbf{H}) = \dot{\mathcal{I}}(\mathbf{H}) \mathbf{H}^T$  correspond au gradient relatif déjà vu au niveau de l'équation (3.10). Il est clair que si  $\mathbf{H}$  est une matrice unitaire, les deux procédures itératives (3.18) et (3.20) sont identiques. Cependant elles diffèrent si  $\mathbf{H}$  n'est pas unitaire comme dans

notre cas. Nous avons choisi d'utiliser (3.20) de par ses bonnes propriétés vis à vis des performances de séparation.

### 3.3.2 Deuxième étape

Rappelons que l'on recherche une matrice  $\mathbf{H}$  devant satisfaire la contrainte  $\mathbf{H}\mathbf{R}_x\mathbf{H}^T = \mathbf{I}$ . Supposons maintenant qu'avant une itération de la procédure de gradient, la matrice de séparation estimée vérifie cette contrainte, alors il n'y a aucune raison qu'après elle la vérifie encore. En effet, en utilisant (3.20) on obtient

$$\mathbf{H}_1\mathbf{R}_x\mathbf{H}_1^T = \mathbf{I} + \lambda\left(\mathcal{I}'(\mathbf{H}) + (\mathcal{I}'(\mathbf{H}))^T\right) + \lambda^2\mathcal{I}'(\mathbf{H})(\mathcal{I}'(\mathbf{H}))^T$$

qui est donc, en général, différent de  $\mathbf{I}$ . Ainsi une seconde étape de normalisation est nécessaire. Elle devra correspondre à une sorte de "projection" sur l'ensemble des matrices satisfaisant  $\mathbf{H}\mathbf{R}_x\mathbf{H}^T = \mathbf{I}$ . Cela de façon à maintenir la contrainte tout au long des itérations. Nous allons noter cette opération de normalisation  $\mathcal{N}(\mathbf{H}_1)$ . Ainsi, parce que  $\mathbf{H}_1\mathbf{R}_x\mathbf{H}_1^T \neq \mathbf{I}$ , le signal  $\mathbf{y}_1$  reconstruit à partir de  $\mathbf{H}_1 : \mathbf{y}_1 = \mathbf{H}_1\mathbf{x}$  ne possède pas une matrice de covariance égale à l'identité. On voudrait donc trouver un vecteur  $\mathbf{y}_n$  le plus proche possible de  $\mathbf{y}_1$  et tel que  $\mathbf{R}_{y_n} = \mathbf{I}$ . A cette fin on définit une erreur d'approximation par

$$\mathbf{e} = \mathbf{y}_1 - \mathbf{y}_n.$$

Le vecteur  $\mathbf{y}_n$  résultant de l'opération de normalisation s'écrit

$$\mathbf{y}_n = \mathcal{N}(\mathbf{H}_1)\mathbf{x}.$$

Ainsi, on a

$$\mathbf{e} = (\mathbf{H}_1 - \mathcal{N}(\mathbf{H}_1))\mathbf{x}. \tag{3.21}$$

Notre objectif est maintenant de trouver une matrice  $\mathcal{N}(\mathbf{H}_1)$  satisfaisant

$$\mathcal{N}(\mathbf{H}_1)\mathbf{R}_x(\mathcal{N}(\mathbf{H}_1))^T = \mathbf{I} \quad (3.22)$$

et telle que l'erreur  $\mathbf{e}$  soit la plus petite en moyenne quadratique. Il s'agit donc de déterminer  $\mathcal{N}(\mathbf{H}_1)$  minimisant

$$E_e = \mathbb{E}[\mathbf{e}^T \mathbf{e}]. \quad (3.23)$$

Après quelques calculs donnés en Annexe 2 on propose la normalisation décrite ci après. Considérons d'abord la décomposition suivante  $\mathbf{R}_x = \mathbf{M}\mathbf{M}^T$  de la matrice de covariance des observations. Ainsi la contrainte peut maintenant s'écrire  $\mathbf{U}_d\mathbf{U}_d^T = \mathbf{I}$  où  $\mathbf{U}_d = \mathcal{N}(\mathbf{H}_1)\mathbf{M}$ . Considérons maintenant une décomposition en valeurs singulières (SVD) de  $\mathbf{H}_1\mathbf{M}$  :

$$\mathbf{H}_1\mathbf{M} = \mathbf{U}_t\mathbf{D}_t\mathbf{V}_t^T$$

où les matrices  $\mathbf{U}_t$  et  $\mathbf{V}_t$  sont unitaires et  $\mathbf{D}_t$  est diagonale. La normalisation suivante

$$\mathcal{N}(\mathbf{H}_1) = \mathbf{U}_t\mathbf{V}_t^T\mathbf{M}^{-1}$$

correspond au meilleur choix (en moyenne quadratique) d'une matrice conduisant à un vecteur de matrice de covariance égale à l'identité.

### 3.3.3 Résumé de l'algorithme

Nous avons appelé **OSIBS** l'algorithme regroupant les deux étapes des deux paragraphes précédents. OSIBS signifie "One Stage Iterative Block Separation" et peut être résumé comme suit

algorithme OSIBS



1. Estimer la matrice de covariance des observations :  $\hat{\mathbf{R}}_x$
2. Calculer une racine carrée  $\mathbf{M}$  de  $\hat{\mathbf{R}}_x$
3. Choisir une valeur initiale  $\mathbf{H}$  pour la matrice de séparation et une constante positive  $\lambda$ . Initialiser les sorties  $\mathbf{Y} = \mathbf{H}\mathbf{X}$  et itérer les étapes (a)-(d) suivantes jusqu'à la convergence :
  - (a) Estimer la dérivée du contraste choisi  $\mathcal{I}$
  - (b) Calculer  $\mathbf{H}_1 = (\mathbf{I} + \lambda\mathcal{I}\mathbf{H}^T)\mathbf{H}$
  - (c) Calculer  $\mathbf{H} = \mathcal{N}(\mathbf{H}_1)$
  - (d) Déterminer les sources estimées :  $\mathbf{Y} = \mathbf{H}\mathbf{X}$

En notant  $N_d$  le nombre de données disponibles,  $\mathbf{X}$  et  $\mathbf{Y}$  sont les matrices  $(N, N_d)$  correspondants respectivement aux observations et aux sources estimées.

## 3.4 Simulations numériques

Nous allons présenter des comparaisons entre les algorithmes décrits dans ce chapitre. De façon générale dans ce paragraphe, nous allons considérer des signaux sources de plusieurs natures, tout d'abord des signaux sources 2-PAM (prenants leurs valeurs parmi  $\{-1, +1\}$  identiquement distribués), puis des signaux 4-PAM (prenants leurs valeurs parmi  $\{-3/\sqrt{5}, -1/\sqrt{5}, 1/\sqrt{5}, 3/\sqrt{5}\}$  identiquement distribués) et enfin des signaux CP trois états paramétrés par un coefficient  $\tau$  (cf. paragraphe 2.3.1). Le nombre d'échantillons étant fixé à  $N_d = 1000$ .

### 3.4.1 Algorithmes de Lagrangien

Nous allons d'abord comparer les deux algorithmes de Lagrangien (**LAGR** et **LAGRAUG**) correspondants aux algorithmes basés respectivement sur les techniques de Lagrangien (cf. paragraphe 3.2.1) et de Lagrangien augmenté (cf. paragraphe 3.2.3). Nous nous plaçons dans un cas de deux sources et la matrice de mélange vaut

$$G_{21} = \begin{bmatrix} 1 & 0.6 \\ 0.4 & 1 \end{bmatrix}. \quad (3.24)$$

Nous avons pris les mêmes coefficients d'optimisation pour les deux algorithmes. La figure (FIG. 3.1) est obtenue en utilisant des signaux sources 2-PAM. Pour la figure (FIG. 3.2) nous avons utilisés des signaux 4-PAM. Pour la figure (FIG. 3.3), nous avons utilisé des signaux sources CP avec un coefficient  $\tau = 1,7$  (kurtosis négatif).

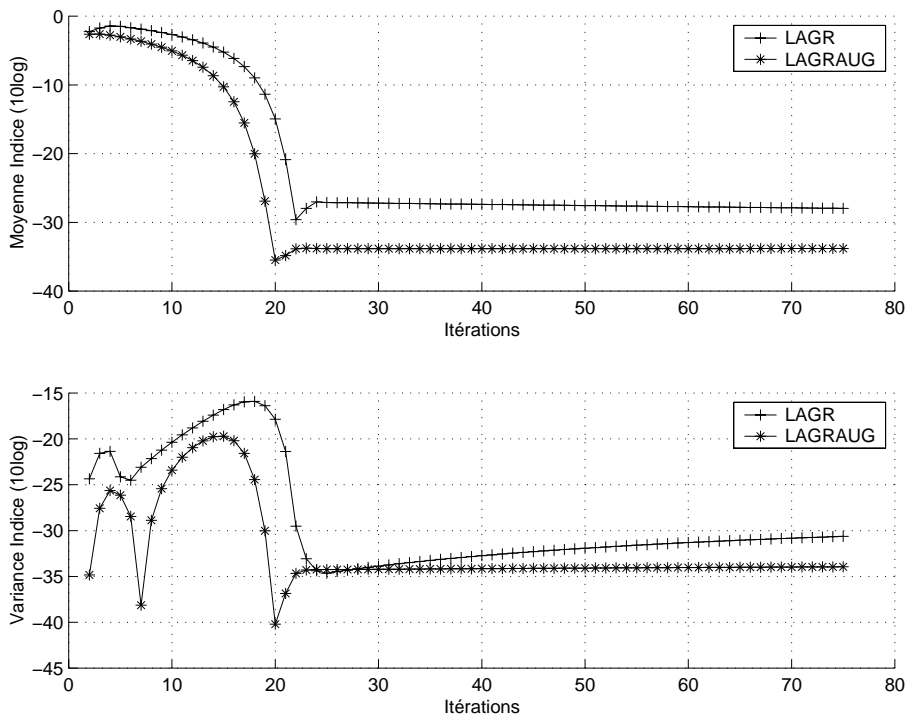


FIG. 3.1 – Cas de deux signaux sources 2-PAM.

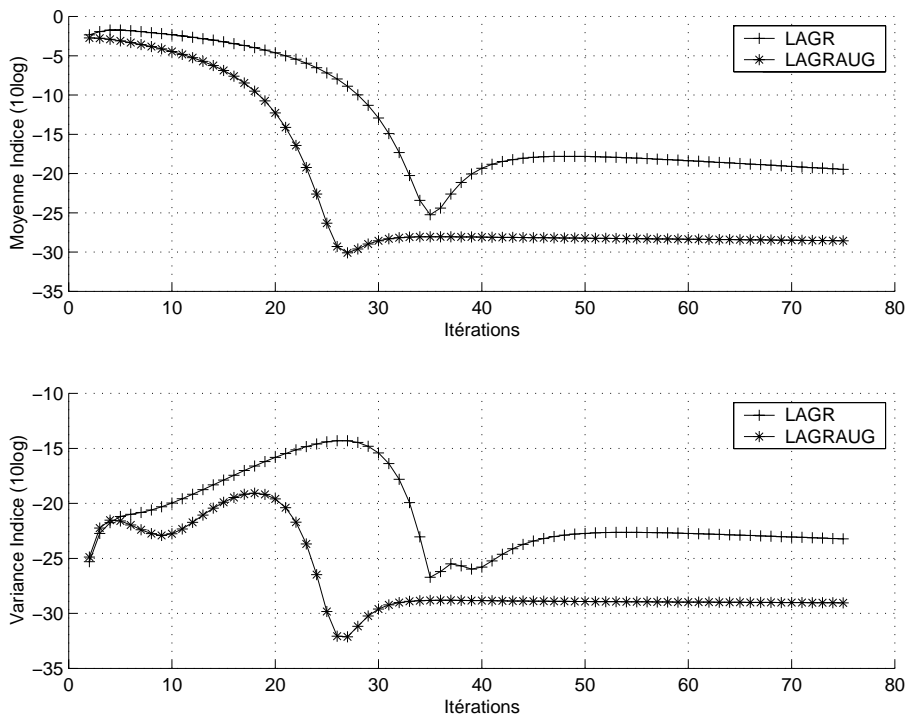
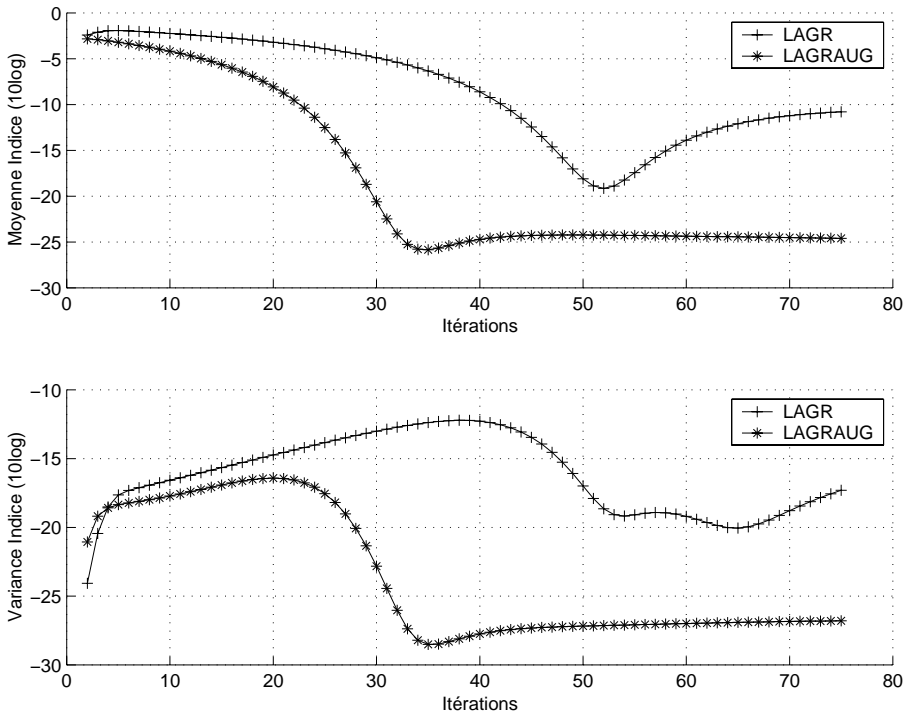


FIG. 3.2 – Cas de deux signaux sources 4-PAM.


 FIG. 3.3 – Cas de deux signaux sources CP ( $\tau = 1, 7$ ).

On constate que l'algorithme de Lagrangien augmenté a de meilleures performances que celles de l'algorithme de Lagrangien. L'algorithme basé sur le Lagrangien augmenté semble converger plus rapidement vers un indice de performance meilleur, au détriment cependant d'une complexité plus importante.

### 3.4.2 Algorithmes de pénalisations

Pour comparer les deux méthodes de séparation de source par pénalisation, nous allons de nouveau nous placer dans le cas de deux sources. Nous appellerons **FROBR** et **KULLR** les algorithmes basés sur les critères  $\mathcal{P}_q(\mathbf{H})$  (3.14) et  $\mathcal{P}_d(\mathbf{H})$  (3.15) respectivement. La matrice de mélange est  $G_{21}$  (3.24). Les mêmes signaux qu'au paragraphe précédent sont considérés. La première figure (FIG. 3.4) considère des signaux 2-PAM, la deuxième figure (FIG. 3.5) des signaux 4-PAM et la dernière (FIG. 3.6) des signaux CP avec un coefficient  $\tau = 1, 7$ . Afin de comparer ces deux algorithmes, nous prenons des coefficients d'optimisation égaux.

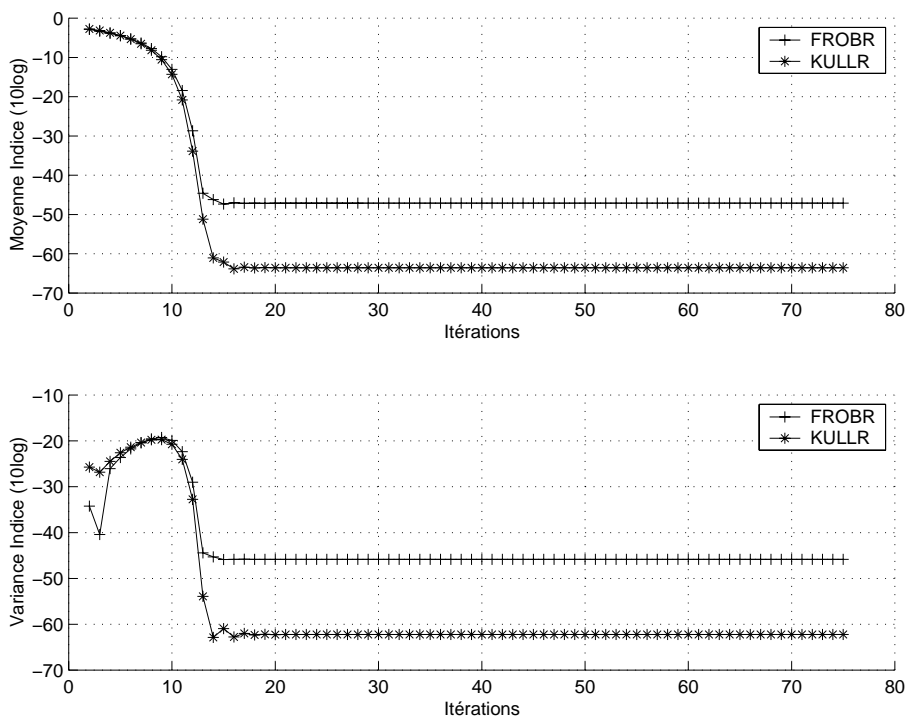


FIG. 3.4 – Cas de deux signaux sources 2-PAM.

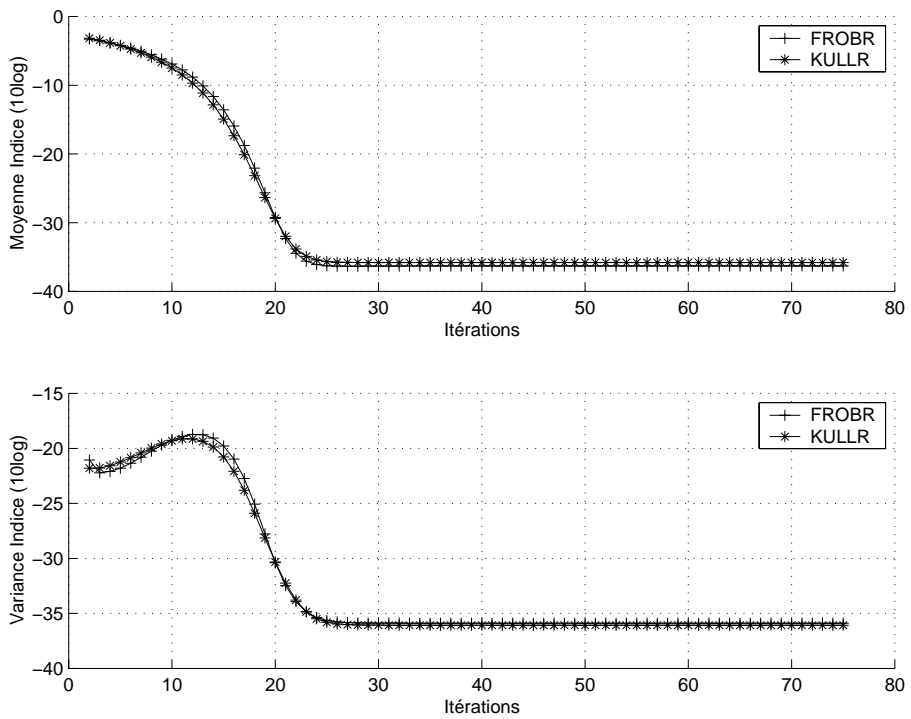
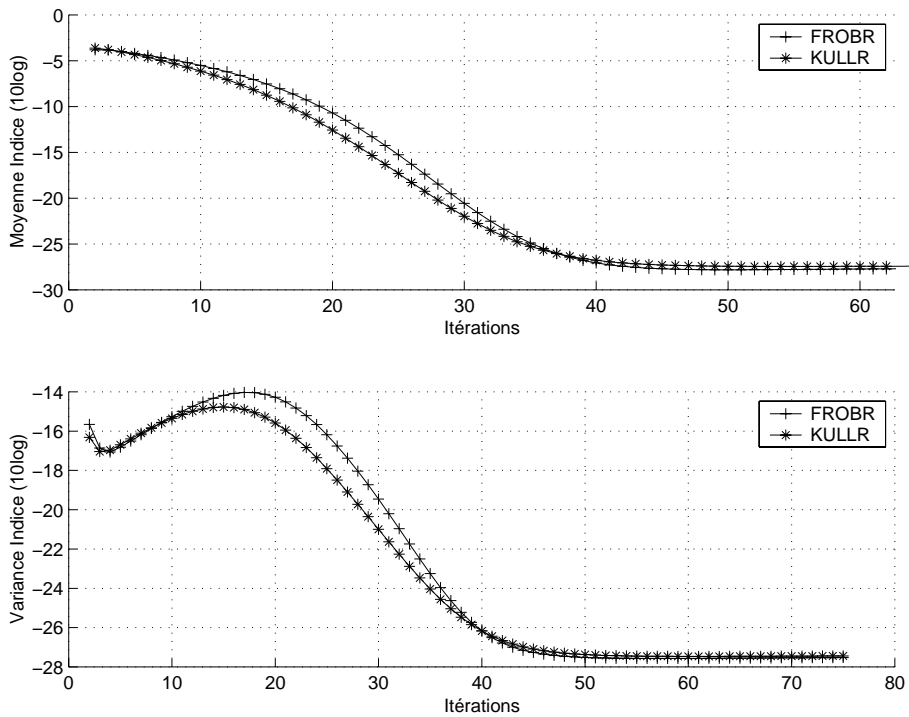


FIG. 3.5 – Cas de deux signaux sources 4-PAM.

FIG. 3.6 – Cas de deux signaux sources CP ( $\tau = 1, 7$ ).

Nous remarquons que les deux algorithmes **KULLR** et **FROBR** sont à peu près équivalents en terme de vitesse de convergence et de performances atteintes. Leur comportement est intéressant dans la mesure où des phénomènes de rebondissement n'apparaissent pas comme pour les algorithmes basés sur les Lagrangiens. Cependant pour ces quatre algorithmes, des limitations apparaissent dès que l'on considère des cas plus complexes comme par exemple si l'on augmente le nombre de sources, si l'on prend des matrices de mélanges mal conditionnées ou si l'on considère un cas bruité. Dans le cas des matrices de mélanges moins favorables, cela pourrait provenir du fait que l'on initialise la matrice de séparation comme une matrice unité, les matrices de mélanges dont l'inverse sont trop éloignées de celle ci peuvent entraîner une mauvaise convergence de ces algorithmes. De façon plus générale, le manque de robustesse pourrait provenir du fait que ces algorithmes itératifs ne vérifient la contrainte qu'une fois la convergence atteinte. Cela pourrait aussi expliquer les phénomènes de rebondissement et de faible robustesse.

### 3.4.3 Algorithmes directs

Dans ce paragraphe, nous allons effectuer quelques comparaisons entre l'algorithme **OSIBS** et l'algorithme **EASI** [CAR96], ce dernier sera utilisé dans une version bloc. Les sources utilisées sont des signaux 2-PAM, 4-PAM et des signaux CP avec un coefficient  $\tau = 1, 7$ . Les matrices de mélanges sont  $G_{21}$  (3.24),

$$G_{22} = \begin{bmatrix} 1 & 0.95 \\ 0.8 & 1 \end{bmatrix} \quad \text{et} \quad G_{33} = \begin{bmatrix} 1 & 0.2 & 0.8 \\ 0.65 & 1 & 0.7 \\ 0.6 & 0.5 & 1 \end{bmatrix}.$$

Les trois premières figures (FIG. 3.7), (FIG. 3.8) et (FIG. 3.9) utilisent les trois signaux décrits précédemment avec la matrice de mélange  $G_{21}$ . Les trois suivantes (FIG. 3.10), (FIG. 3.11) et (FIG. 3.12) utilisent les mêmes signaux mais avec la matrice de mélange  $G_{22}$ . Les figures (FIG. 3.13), (FIG. 3.14) et (FIG. 3.15) séparent trois sources des types précédemment utilisés avec la matrice de mélange  $G_{33}$ .

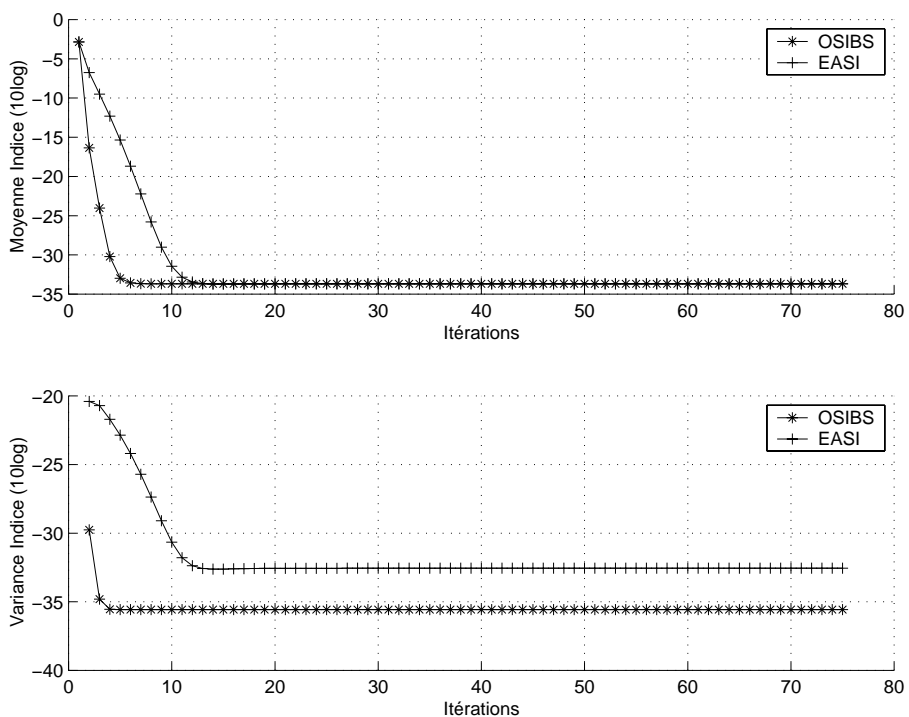


FIG. 3.7 – Cas de deux signaux sources 2-PAM, matrice de mélange  $G_{21}$ .

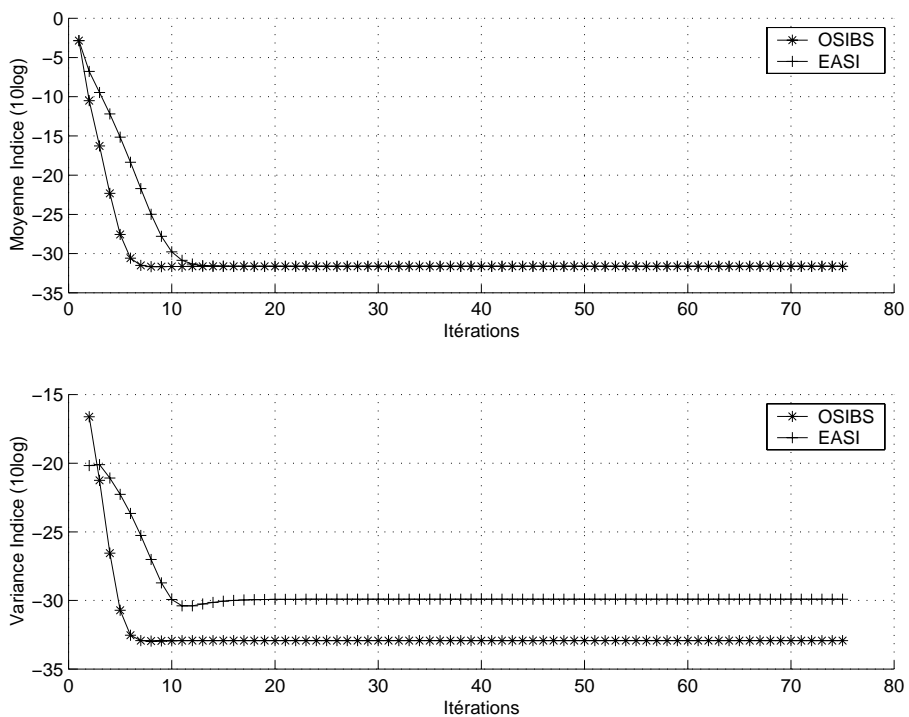


FIG. 3.8 – Cas de deux signaux sources 4-PAM, matrice de mélange  $G_{21}$ .

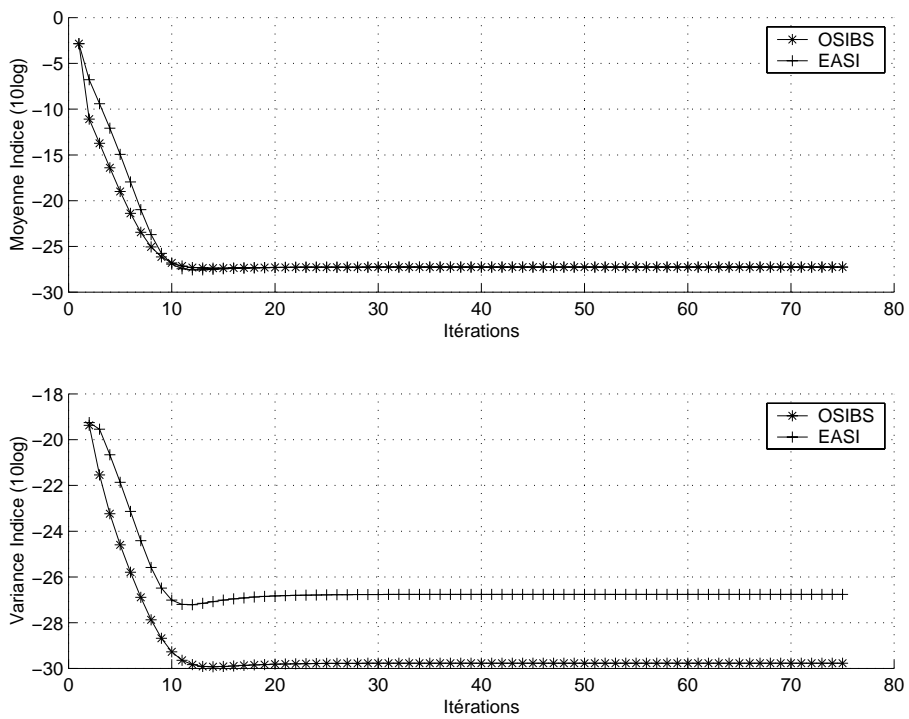


FIG. 3.9 – Cas de deux signaux sources CP ( $\tau = 1, 7$ ), matrice de mélange  $G_{21}$ .



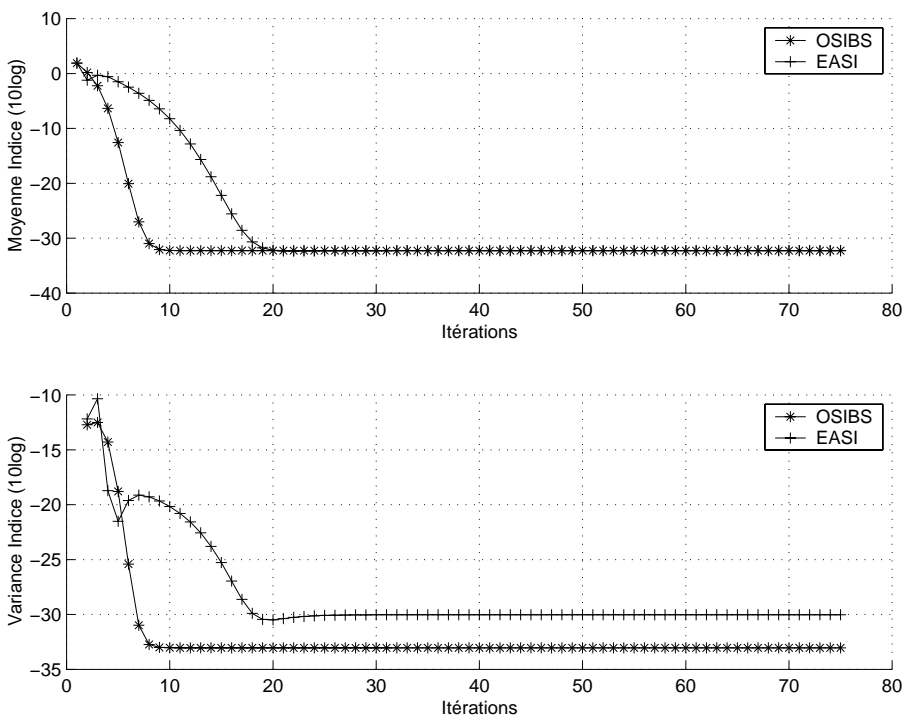


FIG. 3.10 – Cas de deux signaux sources 2-PAM, matrice de mélange  $G_{22}$ .

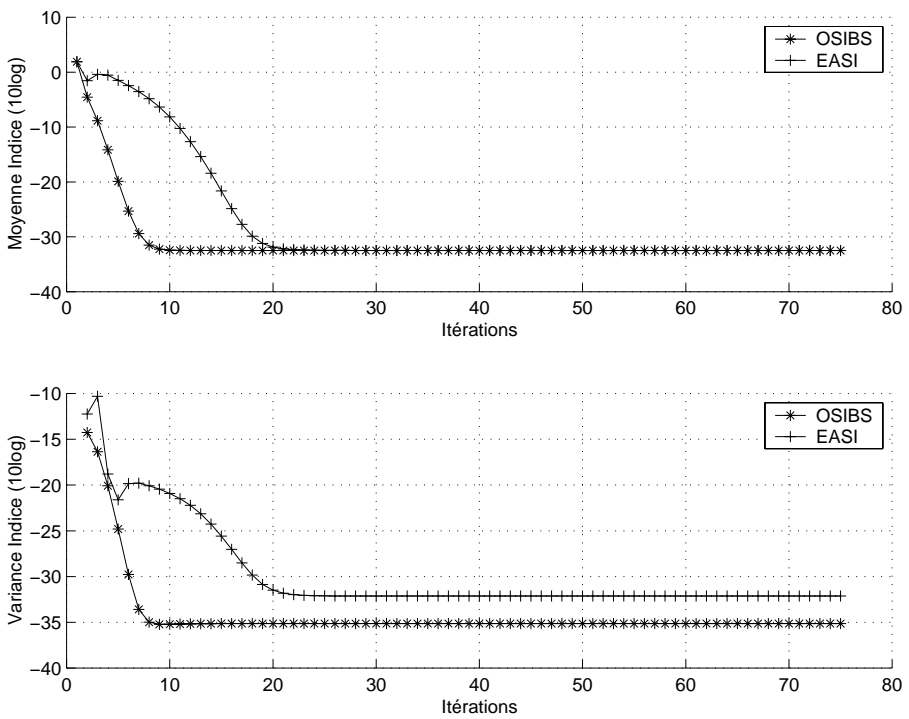


FIG. 3.11 – Cas de deux signaux sources 4-PAM, matrice de mélange  $G_{22}$ .

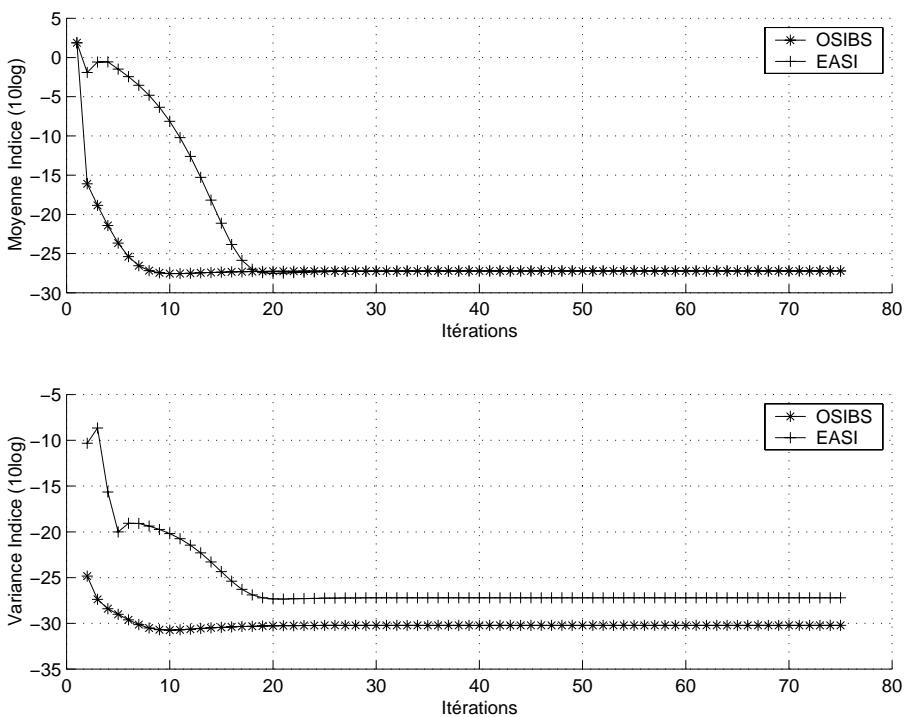


FIG. 3.12 – Cas de deux signaux sources CP ( $\tau = 1, 7$ ), matrice de mélange  $G_{22}$ .

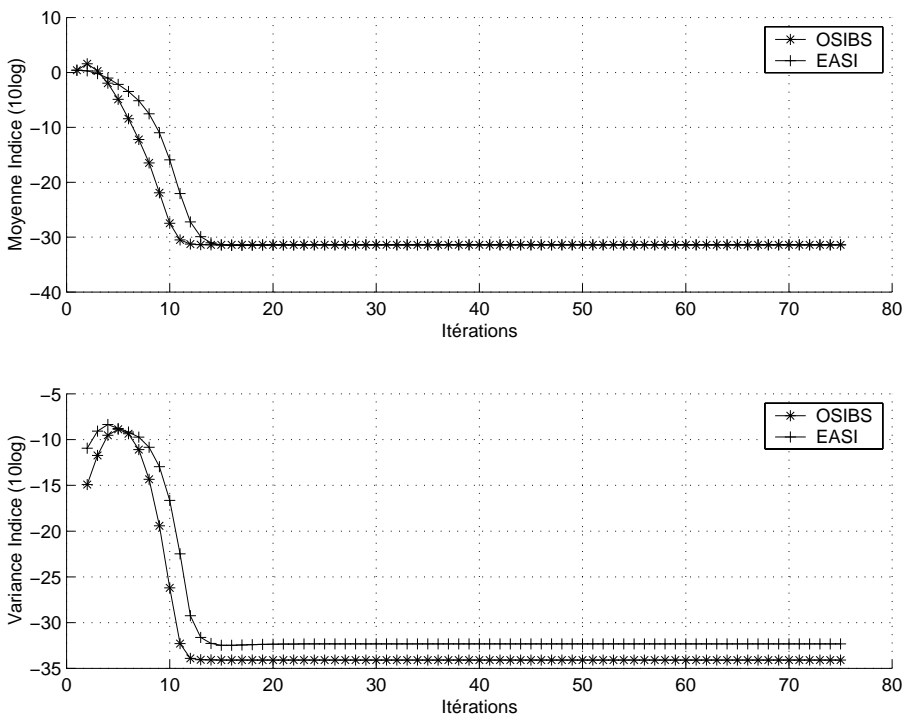


FIG. 3.13 – Cas de trois signaux sources 2-PAM, matrice de mélange  $G_{33}$ .

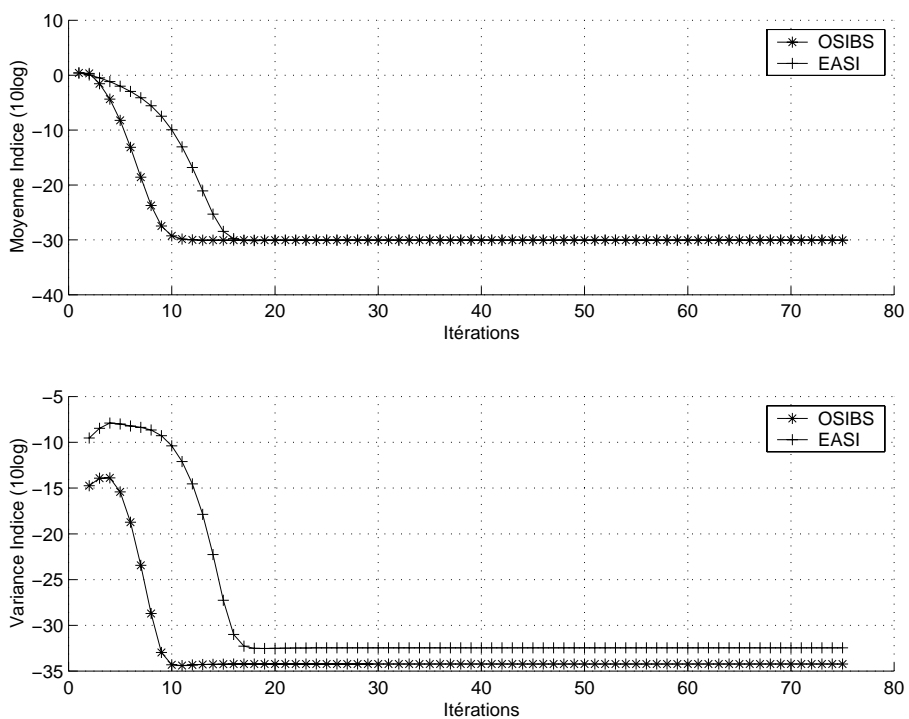


FIG. 3.14 – Cas de trois signaux sources 4-PAM, matrice de mélange  $G_{33}$ .

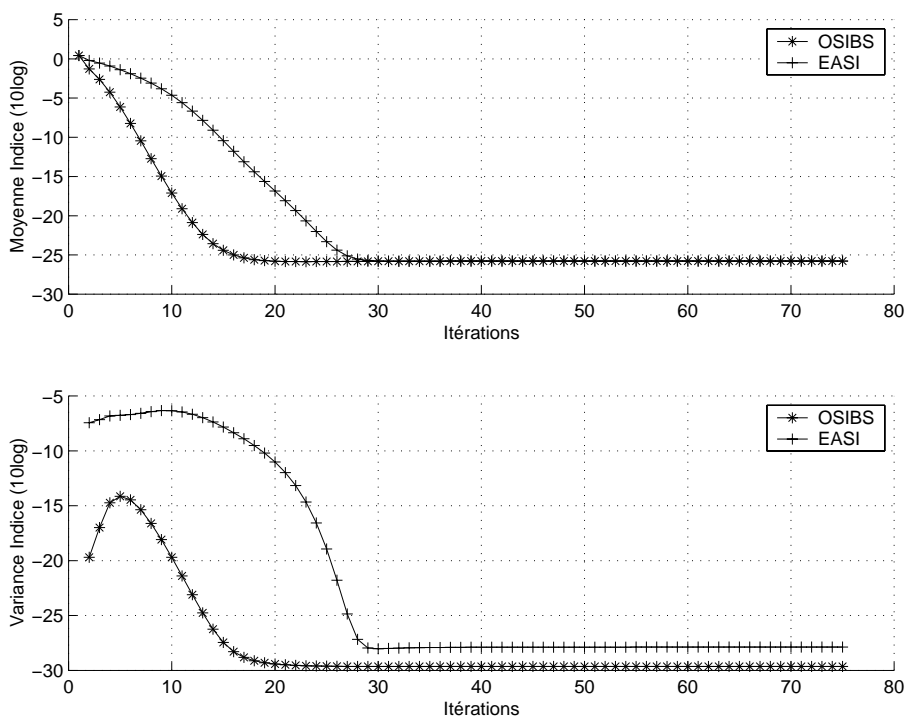


FIG. 3.15 – Cas de trois signaux sources CP ( $\tau = 1, 7$ ), matrice de mélange  $G_{33}$ .

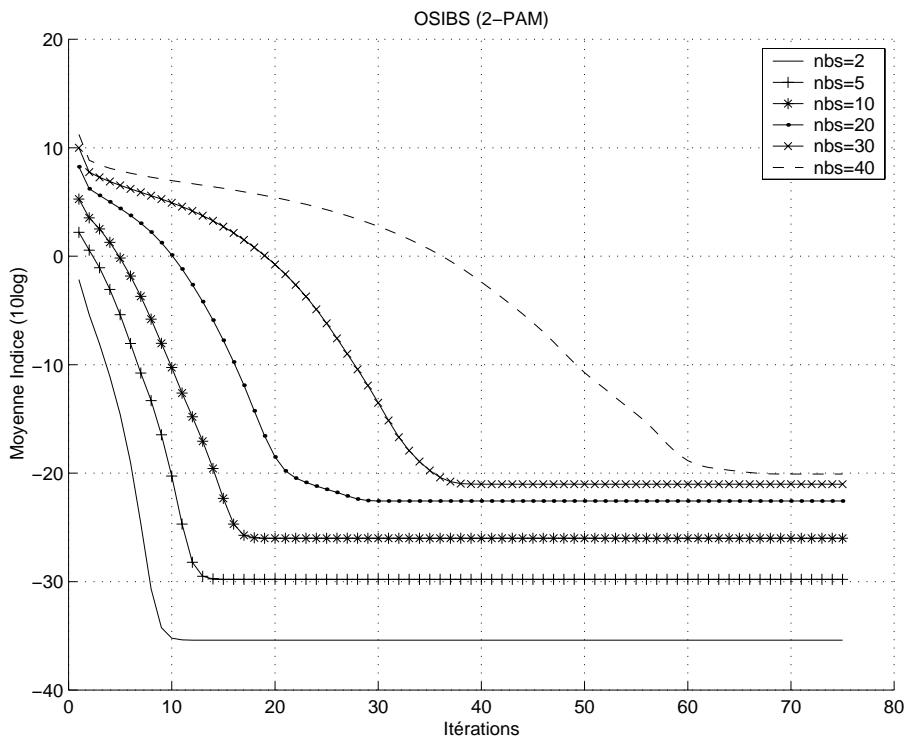


FIG. 3.16 – OSIBS : Signaux sources 2-PAM, convergences

Nous voyons sur ces simulations que dans l'ensemble l'algorithme **OSIBS** affiche des performances équivalentes à l'algorithme **EASI**. Cependant **OSIBS** semble avoir une meilleure vitesse de convergence. Au vu des différentes simulations numériques, nous avons remarqué que l'algorithme **OSIBS** semble plus robuste que la version bloc de l'algorithme **EASI**. La figure (FIG 3.16) illustre la convergence de l'algorithme **OSIBS** pour un nombre de sources allant jusqu'à 40 et pour une matrice de mélange dont les éléments sont tirés de façon aléatoire entre  $-1$  et  $+1$ , l'algorithme **EASI** semblant avoir des difficultés à converger pour un nombre de sources supérieur à 3. Sur la figure (FIG. 3.16) nous remarquons que lorsque le nombre de sources augmente, la vitesse de convergence et les performances moyennes se dégradent naturellement mais l'algorithme continue de converger relativement facilement.

## 3.5 conclusion

Pour conclure sur cette étude, il a été montré dans ce chapitre que l'approche de la séparation de sources par optimisation de fonction de contraste sous contrainte pouvait donner des résultats intéressants, les méthodes d'optimisation avec contraintes donnant des résultats relativement satisfaisants. Lors de cette étude, nous avons mis en évidence que les méthodes duales de Lagrangien et de pénalisation souffrent d'un problème de stabilité numérique et de manque de robustesse, défaut que l'on a pu minimiser dans une moindre mesure par la méthode de Lagrangien augmenté, cependant c'est avec les méthodes directes que l'on a pu obtenir les résultats les plus probants.

## Annexe 2: Détails des calculs pour OSIBS

Calculons  $E_e$  en utilisant (3.23) et (3.21), on obtient

$$\begin{aligned}
E_e &= \mathbb{E} \left[ \mathbf{x}^T (\mathbf{H}_1 - \mathcal{N}(\mathbf{H}_1))^T (\mathbf{H}_1 - \mathcal{N}(\mathbf{H}_1)) \mathbf{x} \right] \\
&= \mathbb{E} \left[ \text{Trace} \left[ (\mathbf{H}_1 - \mathcal{N}(\mathbf{H}_1)) \mathbf{x} \mathbf{x}^T (\mathbf{H}_1 - \mathcal{N}(\mathbf{H}_1))^T \right] \right] \\
&= \text{Trace} \left[ (\mathbf{H}_1 - \mathcal{N}(\mathbf{H}_1)) \mathbf{R}_x (\mathbf{H}_1 - \mathcal{N}(\mathbf{H}_1))^T \right] \\
&= \text{Trace} \left[ \mathbf{H}_1 \mathbf{R}_x \mathbf{H}_1^T + \mathcal{N}(\mathbf{H}_1) \mathbf{R}_x (\mathcal{N}(\mathbf{H}_1))^T - \mathbf{H}_1 \mathbf{R}_x (\mathcal{N}(\mathbf{H}_1))^T - \mathcal{N}(\mathbf{H}_1) \mathbf{R}_x \mathbf{H}_1^T \right] \\
&= C - E'_e
\end{aligned}$$

où en utilisant (3.22)

$$\begin{aligned}
C &= \text{Trace} \left[ \mathbf{H}_1 \mathbf{R}_x \mathbf{H}_1^T \right] + \text{Trace} \left[ \mathcal{N}(\mathbf{H}_1) \mathbf{R}_x (\mathcal{N}(\mathbf{H}_1))^T \right] \\
&= \text{Trace} \left[ \mathbf{H}_1 \mathbf{R}_x \mathbf{H}_1^T \right] + \text{Trace}[\mathbf{I}]
\end{aligned}$$

Ce terme  $C$  est une constante par rapport à notre problème car il ne depend pas de  $\mathcal{N}(\mathbf{H}_1)$ .

Maintenant pour  $E'_e$ , on a

$$\begin{aligned}
E'_e &= \text{Trace} \left[ \mathbf{H}_1 \mathbf{R}_x (\mathcal{N}(\mathbf{H}_1))^T + \mathcal{N}(\mathbf{H}_1) \mathbf{R}_x \mathbf{H}_1^T \right] \\
&= \text{Trace} \left[ \mathbf{H}_1 \mathbf{R}_x (\mathcal{N}(\mathbf{H}_1))^T + \left( \mathbf{H}_1 \mathbf{R}_x (\mathcal{N}(\mathbf{H}_1))^T \right)^T \right] \\
&= 2 \text{Trace} \left[ \mathbf{H}_1 \mathbf{R}_x (\mathcal{N}(\mathbf{H}_1))^T \right].
\end{aligned}$$

Ainsi, parce que  $C$  est constant, la minimisation de  $E_e$  est donc équivalente à la maximisation de  $E'_e$ . Soit  $\mathbf{M}$  une racine carrée de  $\mathbf{R}_x$ , *i.e.*  $\mathbf{R}_x = \mathbf{M} \mathbf{M}^T$ , nous devons trouver un

maximum à l'expression

$$E'_e = 2\text{Trace} \left[ \mathbf{H}_1 \mathbf{M} \mathbf{M}^T (\mathcal{N}(\mathbf{H}_1))^T \right] = 2\text{Trace} [\mathbf{B} \mathbf{A}^T]$$

où

$$\mathbf{B} = \mathbf{H}_1 \mathbf{M}$$

$$\mathbf{A} = \mathcal{N}(\mathbf{H}_1) \mathbf{M}.$$

La contrainte (3.22) peut maintenant s'écrire grâce à  $\mathbf{A}$  ainsi :  $\mathbf{A} \mathbf{A}^T = \mathbf{I}$ , *i.e.*  $\mathbf{A}$  est une matrice unitaire. Le problème de maximisation précédent devient maintenant un problème d'algèbre "classique". Rappelons sa solution en utilisant nos notations. Si  $\mathbf{B} = \mathbf{U}_t \mathbf{D}_t \mathbf{V}_t^T$  est une décomposition en valeur singulières (SVD) de  $\mathbf{B}$ , alors

$$\text{Trace} [\mathbf{B} \mathbf{A}^T] = \text{Trace} [\mathbf{U}_t \mathbf{D}_t \mathbf{V}_t^T \mathbf{A}^T] = \text{Trace} [\mathbf{D}_t \mathbf{V}_t^T \mathbf{A}^T \mathbf{U}_t].$$

Donc

$$\text{Trace} [\mathbf{B} \mathbf{A}^T] = \sum_{i=1}^N \sigma_i(\mathbf{B}) T_{i,i}$$

où  $\sigma_i(\mathbf{B})$  sont les valeurs singulières de  $\mathbf{B}$  et  $\mathbf{T} = (T_{i,j}) = \mathbf{V}_t^T \mathbf{A}^T \mathbf{U}_t$  est une matrice unitaire.

Finalement, parce que  $\mathcal{N}(\mathbf{H}_1) = \mathbf{A} \mathbf{M}^{-1}$ , une matrice  $\mathcal{N}(\mathbf{H}_1)$  maximisant  $E'_e$  est donc

$$\mathcal{N}(\mathbf{H}_1) = \mathbf{U}_t \mathbf{V}_t^T \mathbf{M}^{-1}.$$

## Annexe 3: Les algorithmes

### Contraste utilisé

Par soucis de simplicité en ce qui concerne les algorithmes d'optimisation directe (cf. chapitre 3), nous allons utiliser la fonction de contraste suivante :

$$\mathcal{J}_4(\mathbf{y}) = \varepsilon \sum_{i=1}^N \mathbb{E} [y_i^4]$$

où  $\varepsilon = -1$  (resp.  $+1$ ) quand les sources ont un kurtosis négatif (resp. positif). Dans ce cas nous trouvons facilement :

$$\frac{\partial \mathcal{J}_4(\mathbf{y})}{\partial \mathbf{H}} = 4\varepsilon \mathbb{E} [g(\mathbf{y}) \mathbf{x}^T]$$

où  $g(\cdot)$  est la fonction cubique non-linéaire introduisant les statistiques d'ordres supérieurs. De façon classique, nous pouvons prendre  $g(\cdot) = (\cdot)^3$  ou  $g(\cdot) = \tanh(\cdot)$  pour les sources à kurtosis négatif (avec  $\varepsilon = +1$ )

et en utilisant (1.3)

$$\nabla_H \mathcal{J}_4(\mathbf{y}) = 4\varepsilon \mathbb{E} [g(\mathbf{y}) \mathbf{y}^T]$$

Enfin, nous remplaçons l'opérateur espérance mathématique par une moyenne sur les  $N_d$  échantillons.  $\mathbb{E}[\mathbf{y}\mathbf{y}^T]$  sera ainsi remplacé par  $\frac{1}{N_d} \mathbf{Y}\mathbf{Y}^T$ ,  $N_d$  étant le nombre d'échantillons disponibles. Dans la mesure où cela ne prête pas à confusion, nous ne tiendrons plus compte dans la suite de la constante  $N_d$ . Les versions empiriques des contrastes sont donc :

$$\frac{\partial \mathcal{J}_4}{\partial \mathbf{H}} = 4\varepsilon g(\mathbf{Y}) \mathbf{X}^T$$



$$\nabla_{\mathbf{H}} \mathcal{J}_4 = 4\varepsilon g(\mathbf{Y}) \mathbf{Y}^T$$

## L'algorithme LAGR

Nous avons vu au paragraphe 3.2.1 que la maximisation de la fonction de Lagrange  $\mathcal{L}(\mathbf{H}, \mathbf{L})$  dans (3.6) permet de résoudre le problème de la séparation de sources. L'optimum sera obtenu lorsque les deux dérivées partielles (3.7) seront nulles. L'algorithme comportera donc deux parties, la première se chargera de maximiser la fonction de Lagrange par rapport à la matrice de séparation  $\mathbf{H}$ , la seconde adaptera la matrice des coefficients de Lagrange afin de satisfaire la contrainte (3.13). Nous allons utiliser la méthode des gradients relatifs afin d'effectuer ces optimisations.

Ces deux étapes sont représentées par les équations suivantes

$$\mathbf{H}_{k+1} = (\mathbf{I} + \mu_n \nabla_{\mathbf{H}} \mathcal{L}(\mathbf{H}, \mathbf{L})) \mathbf{H}_k \quad (3.25)$$

$$\mathbf{L}_{k+1} = \mathbf{L}_k - \mu_l \frac{\partial \mathcal{L}(\mathbf{H}, \mathbf{L})}{\partial \mathbf{L}} \quad (3.26)$$

La première équation correspond à la maximisation de  $\mathcal{L}(\mathbf{H}, \mathbf{L})$  par rapport à  $\mathbf{H}$  ( $\mathbf{L}$  constant) tandis que la seconde équation cherche un minimum de  $\mathcal{L}(\mathbf{H}, \mathbf{L})$  par rapport à  $\mathbf{L}$  ( $\mathbf{H}$  constant). L'algorithme basé sur (3.25) et (3.26) sera appelé **LAGR**. Les expressions des dérivées partielles sont les suivantes (cf. paragraphe 3.2.1)

$$\nabla_{\mathbf{H}} \mathcal{L}(\mathbf{H}, \mathbf{L}) = \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{y}) + 2\mathbf{L}\mathbf{H}\mathbf{R}_x\mathbf{H}^T = \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{y}) + 2\mathbf{L}\mathbf{E}[\mathbf{y}\mathbf{y}^T] \quad (3.27)$$

$$\frac{\partial \mathcal{L}(\mathbf{H}, \mathbf{L})}{\partial \mathbf{L}} = \mathbf{H}\mathbf{R}_x\mathbf{H}^T - \mathbf{I} = \mathbf{E}[\mathbf{y}\mathbf{y}^T] - \mathbf{I} \quad (3.28)$$

où l'on a  $\nabla_{\mathbf{H}} \mathcal{I}(\mathbf{y})$  le gradient relatif de la fonction de contraste appropriée  $\mathcal{I}(\mathbf{y})$ .

L'algorithme empirique de Lagrange **LAGR** est donc donné par :

$$\begin{cases} \mathbf{H}_{k+1} = (\mathbf{I} + \mu_n [\nabla_{\mathbf{H}} \mathcal{I}(\mathbf{Y}_k) + 2\mathbf{L}_{k-1} \mathbf{Y}_k \mathbf{Y}_k^T]) \mathbf{H}_k \\ \mathbf{L}_{k+1} = \mathbf{L}_k - \mu_l [\mathbf{Y}_k \mathbf{Y}_k^T - \mathbf{I}] \end{cases}$$

## L'algorithme LAGRAUG

Le principe de l'algorithme de Lagrangien augmenté est le même que pour le Lagrangien précédent, il suffit de remplacer l'expression de la dérivée partielle dans (3.25) par :

$$\begin{aligned} \nabla_{\mathbf{H}} \mathcal{L}_q(\mathbf{H}, \mathbf{L}) &= \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{y}) + 2\mathbf{L} \mathbf{H} \mathbf{R}_x \mathbf{H}^T - (\mathbf{H} \mathbf{R}_x \mathbf{H}^T - \mathbf{I}) \mathbf{H} \mathbf{R}_x \mathbf{H}^T \\ &= \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{y}) + 2\mathbf{L} \mathbf{E}[\mathbf{y} \mathbf{y}^T] - (\mathbf{E}[\mathbf{y} \mathbf{y}^T] - \mathbf{I}) \mathbf{E}[\mathbf{y} \mathbf{y}^T] \end{aligned}$$

La version empirique est donc :

$$\nabla_{\mathbf{H}} \mathcal{L}_q(\mathbf{H}, \mathbf{L}) = \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{Y}_k) + 2\mathbf{L}_k \mathbf{Y}_k \mathbf{Y}_k^T - (\mathbf{Y}_k \mathbf{Y}_k^T - \mathbf{I}) \mathbf{Y}_k \mathbf{Y}_k^T$$

L'algorithme empirique de Lagrangien augmenté **LAGRAUG** est donc donné par :

$$\begin{cases} \mathbf{H}_{k+1} = (\mathbf{I} + \mu_n [\nabla_{\mathbf{H}} \mathcal{I}(\mathbf{Y}_k) + 2\mathbf{L}_k \mathbf{Y}_k \mathbf{Y}_k^T - (\mathbf{Y}_k \mathbf{Y}_k^T - \mathbf{I}) \mathbf{Y}_k \mathbf{Y}_k^T]) \mathbf{H}_k \\ \mathbf{L}_{k+1} = \mathbf{L}_k - \mu_l [\mathbf{Y}_k \mathbf{Y}_k^T - \mathbf{I}] \end{cases}$$

## L'algorithme FROBR

Les algorithmes de séparation de sources par pénalisation sont assez simples à programmer, en effet, nous pouvons mettre en œuvre une méthode d'optimisation par gradient

relatif dont l'itération s'écrit

$$\mathbf{H} = (\mathbf{I} + \mu \nabla_{\mathbf{H}} \mathcal{P}_q(\mathbf{H})) \mathbf{H}$$

avec

$$\nabla_{\mathbf{H}} \mathcal{P}_q(\mathbf{H}) = \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{y}) - c_q (\mathbf{H} \mathbf{R}_x \mathbf{H}^T - \mathbf{I}) \mathbf{H} \mathbf{R}_x \mathbf{H}^T$$

La version empirique de cet algorithme s'écrit donc

$$\mathbf{H}_{k+1} = (\mathbf{I} + \mu \nabla_{\mathbf{H}} \mathcal{P}_q(\mathbf{H}_k)) \mathbf{H}_k$$

avec

$$\nabla_{\mathbf{H}} \mathcal{P}_q(\mathbf{H}_k) = \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{Y}_k) - c_q (\mathbf{Y}_k \mathbf{Y}_k^T - \mathbf{I}) \mathbf{Y}_k \mathbf{Y}_k^T$$

## KULLR

De la même façon, on écrit l'algorithme de pénalisation par la divergence de Kullback comme :

$$\mathbf{H} = (\mathbf{I} + \mu \nabla_{\mathbf{H}} \mathcal{P}_d(\mathbf{H})) \mathbf{H}$$

avec

$$\nabla_{\mathbf{H}} \mathcal{P}_d(\mathbf{H}) = \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{y}) - c_d (\mathbf{H} \mathbf{R}_x \mathbf{H}^T - \mathbf{I})$$

et la version empirique est

$$\mathbf{H}_{k+1} = (\mathbf{I} + \mu \nabla_{\mathbf{H}} \mathcal{P}_d(\mathbf{H}_k)) \mathbf{H}_k$$

avec

$$\nabla_{\mathbf{H}} \mathcal{P}_d(\mathbf{H}_k) = \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{Y}_k) - c_d (\mathbf{Y}_k \mathbf{Y}_k^T - \mathbf{I})$$

## OSIBS

La mise en œuvre de l'algorithme **OSIBS** consiste en la maximisation de la fonction de contraste grâce à une technique de gradient relatif soit :

$$\mathbf{H}'_{k+1} = (\mathbf{I} + \mu \nabla_{\mathbf{H}} \mathcal{I}(\mathbf{Y})) \mathbf{H}_k$$

puis, on opère une opération de normalisation telle que décrite au paragraphe 3.3.

On cherche les matrices  $\mathbf{U}_k$ ,  $\mathbf{D}_k$ ,  $\mathbf{V}_k$  tels que  $\mathbf{H}'_{k+1}\mathbf{M} = \mathbf{U}_k\mathbf{D}_k\mathbf{V}_k^T$ , puis on obtient la matrice de séparation normalisée par

$$\mathbf{H}_{k+1} = \mathbf{U}_k\mathbf{V}_k^T\mathbf{M}^{-1}$$

avec  $\mathbf{M} = \sqrt{\mathbf{R}_x}$ .

## EASI

L'algorithme empirique de séparation de source **EASI** tel qu'il a été donné dans [CAR96] est écrit ci-après :

$$\mathbf{H}_{k+1} = \mathbf{H}_k - \lambda_k [\mathbf{Y}_k\mathbf{Y}_k^T - \mathbf{I} + g(\mathbf{Y}_k)\mathbf{Y}_k^T - \mathbf{Y}_k g(\mathbf{Y}_k)^T] \mathbf{H}_k$$

où  $g(\cdot)$  étant une fonction cubique non-linéaire conduisant à la séparation. Nous rappelons que cet algorithme nécessite un blanchiment préalable des données.

# Conclusion et perspectives

Dans cette thèse, nous avons considéré le problème de la séparation de sources par optimisation de fonctions de contrastes dans un cas de mélange linéaire instantané. Pour cela, nous avons étudié plusieurs approches. Pour la première nous avons d'abord proposé une généralisation de certaines fonctions de contrastes existantes. Puis nous avons proposé deux algorithmes (**ICA2G** et **ICA1G**) basés sur ces deux nouveaux contrastes paramétrés par un jeu de coefficients. Suivant la valeur donnée à ces coefficients, nous retrouvons plusieurs critères connus et pouvons en proposer de nouveaux. Cela pourrait être particulièrement intéressant dans la mesure où l'on pourrait adapter ces coefficients afin de trouver une fonction de contraste la plus adaptée à un problème donné.

La deuxième approche est une approche basée sur l'optimisation de fonctions de contraste sous contrainte. En effet, la plupart des algorithmes de séparation de sources effectuent la maximisation d'une fonction de contraste après une opération de blanchiment des signaux d'observation. Notre approche a pour but d'optimiser la fonction de contraste en tenant compte directement de la contrainte de blancheur des signaux, cela nous permet d'obtenir des algorithmes de séparation en une seule étape, sans blanchiment explicite préalable des signaux. Pour ce faire, nous avons considéré deux familles de méthodes mathématiques d'optimisation non-linéaires avec contraintes, les méthodes duales qui introduisent la contrainte au niveau du critère et les méthodes directes qui forcent la contrainte après une étape d'optimisation du contraste par une méthode de gradient. Parmi les méthodes duales, nous avons considéré les méthodes Lagrangiennes qui ont aboutis aux algorithmes **LAGR** et **LAGRAUG**. Puis les algorithmes **KULLR** et **FROBR** s'appuient sur une mesure de la contrainte au moyen de la divergence de Kullback-Leibler et de la norme

de Frobenius. Les algorithmes basés sur les méthodes duales souffrent de problèmes de robustesse, ce qui pourrait être expliqué par le fait que la contrainte ne sera vérifiée qu'une fois la convergence atteinte. C'est pourquoi nous avons considéré les méthodes directes et proposé un dernier algorithme appelé **OSIBS**, basé sur une opération de "projection" sur la contrainte. A chaque itération, l'étape de maximisation de la fonction de contraste donne une estimation de la matrice de séparation, nous déterminons alors une matrice respectant les contraintes qui est la plus proche (en norme quadratique) de la matrice de séparation estimée préalablement. Cette méthode semble beaucoup plus robuste et converge même pour un grand nombre de sources. Des simulations informatiques illustrent le comportement des algorithmes présentés.

Les perspectives de recherche sont multiples : Dans un premier temps, il serait intéressant d'appliquer d'autres méthodes algorithmiques d'optimisation des familles de contrastes généralisées que nous avons proposé et de comparer les performances des algorithmes obtenus avec les deux algorithmes développés dans la première partie de cette étude. Par la suite et dans la mesure où nous disposons de familles de contrastes dont nous pouvons choisir le nombre et le type de cumulants croisés, il serait intéressant d'étudier l'influence de ceux ci sur la séparation, et en particulier chercher s'il est possible sur un problème donné de trouver un jeu de coefficients optimaux qui améliorerait les performances. De façon plus générale, est il possible, grâce à des connaissances à priori sur les sources de choisir une fonction de contraste plutôt qu'une autre parmi le grand nombre de contrastes existants ? Concernant les méthodes directes d'optimisation sous contraintes dont nous avons vu qu'elles conduisent à des performances relativement intéressantes, il serait bon d'envisager d'autres méthodes pour forcer la contrainte dans l'étape de normalisation de l'algorithme **OSIBS** et d'en comparer les résultats. A moyen terme, il serait intéressant de développer les algorithmes les plus robustes pour des signaux multi-dimensionnels et envisager des applications en analyse d'images par exemple. Enfin, à plus long terme, nous pourrions envisager de développer les algorithmes les plus performants dans le cas de signaux complexes et le cas de mélanges convolutifs.

# Annexes

## **Annexe 4: Nonlinear Constrained Optimization Using Lagrangian Approach For Blind Source Separation**

*Proc. IX European Signal Processing Conference (EUSIPCO'98),  
Island of Rhodes, Greece, pp 2397-2400, Sept. 1998.*



# NONLINEAR CONSTRAINED OPTIMIZATION USING LAGRANGIAN APPROACH FOR BLIND SOURCE SEPARATION

*Benoit STOLL and Eric MOREAU*

MS-GESSY, ISITV, Université de Toulon et du Var,  
Av. G. Pompidou, BP 56, F-83162 La Valette du Var, FRANCE  
e-mail: [stoll@isitiv.univ-tln.fr](mailto:stoll@isitiv.univ-tln.fr); [moreau@isitiv.univ-tln.fr](mailto:moreau@isitiv.univ-tln.fr)

## ABSTRACT

The paper deals with the blind source separation problem. We introduce two new adaptive algorithms based on the minimization of constrained contrast functions using a Lagrangian approach. The algorithms “only” require one stage for separation and the approach is general in the sense that it can be used with any contrasts working with normalized vectors. The computer simulation shows good performances in comparison to the EASI algorithm.

## 1 INTRODUCTION

We consider the source separation problem [3] which find numerous applications in diverse fields of engineering and applied sciences, e.g. data communications, seismic exploration, antenna processing, speech processing etc... It can be simply formulated as follows. Several linear mixture of different signals called sources are observed. We want to recover the unknown original sources without knowing the mixing system. Hence this must be realized from the only observations and this is the reason why this problem is often qualified as “blind” or “unsupervised”.

Among the great number of approaches that have been proposed in the recent literature, we are primary concerned with high-order statistics inverse criteria based approaches. In this field, contrast functions constitute separation criteria in the sense that their maximization solve the source separation problem. Such contrasts have been first introduced in [2] and recently generalized in [7] and [8]. Some contrasts work with normalized vectors (or “white” vectors), e.g. [2], [7], [8], while in [5] one can find contrasts which do not require the normalization. Unfortunately, the optimization procedure of this last class of contrasts is very complex. Now the classical way for the maximization of the first class of contrast require two stages. The first stage consists in a normalization (whitening) of the observations and the second stage maximize the contrast over the set of unitary matrices.

The main purpose of this communication is to propose a one stage algorithm for the maximization of contrasts

requiring normalization. For this task, the normalization is viewed as a constraint which is considered in the criterion thanks to a Lagrangian approach. It has to be noticed that in [4] a Lagrangian approach has also been considered but restricted to the estimation of an unitary matrix. In this latter case a first whitening stage is again required.

Because the parameter we are looking for is a matrix, the methods are derived using both a classical gradient and a relative gradient [1]. This leads to two algorithms whose performances are illustrated thanks to computer simulations in comparison to the EASI algorithm [1].

## 2 PROBLEM FORMULATION

The source separation problem consist in estimating a set of  $M$  independent signals from  $N \geq M$  observed instantaneous mixtures of these signals.

We can express the classical linear memoriless mixture model as

$$\mathbf{x} = \mathbf{G}\mathbf{a} + \mathbf{b} \quad (1)$$

where  $\mathbf{x}$  is the  $(N, 1)$  vector of observations,  $\mathbf{a}$  the  $(M, 1)$  vector of sources,  $\mathbf{b}$  the  $(N, 1)$  vector of additive noise and  $\mathbf{G}$  the  $(N, M)$  invertible mixing matrix. In the following, we consider  $M = N$  and  $\mathbf{b} = 0$ .

The separation problem consist in estimating a separating matrix  $\mathbf{H}$  such that the vector

$$\mathbf{y} = \mathbf{H}\mathbf{x} \quad (2)$$

restore the  $N$  input sources  $a_i$ . In the noiseless case, this is to identify an inverse  $\mathbf{G}^{-1}$  of the mixing matrix. Ideally, considering the global system, the global matrix should be a unit matrix

$$\mathbf{S} = \mathbf{H}\mathbf{G} = \mathbf{I} \quad (3)$$

As sources are unobservable, there are some inherent indeterminacy in their estimation. That is, in general, we can not identify the order and the power of each components of the source vector  $\mathbf{a}$ . Hence  $\mathbf{H}$  is called a separating matrix when the global matrix  $\mathbf{S}$  reads

$$\mathbf{S} = \mathbf{D}\mathbf{P} \quad (4)$$

where  $\mathbf{D}$  is an invertible constant diagonal matrix and  $\mathbf{P}$  a permutation matrix.

For separation, the key hypothesis is the joint independence of the  $N$  sources and the non-zero character of some of their cumulants. Thus Gaussian sources are excluded. Without loss of generality, the sources can be assumed zero-mean with unit variance, *i.e.*

$$\mathbf{E}[\mathbf{a}\mathbf{a}^T] = \mathbf{I} \quad (5)$$

where  $\mathbf{E}$  stands for the mathematical expectation operator, the superscript  $T$  the transpose operator and  $\mathbf{I}$  the  $(N, N)$  identity matrix.

### 3 RECALLS ON CONTRAST FUNCTION

Contrast functions constitute separation criteria in the sense that their maximization solve the separation problem, *i.e.* they are maximum if and only if the relation (4) holds for  $\mathbf{S}$ .

As originally defined [2], a contrast function has to be a symmetrical and scale invariant function to be maximized. According to this definition a first useful contrast was proposed in [2] for normalized vectors.

For real sources, it reads

$$I_2(\mathbf{y}) = \sum_{i=1}^N (\mathbf{C}_p[y_i])^2 \quad (6)$$

where  $\mathbf{C}_p[y_i]$  is the  $p$ -th order joint cumulant of  $y_i$  and  $p$  an integer greater or equal to 3.

It has also been shown that the minimization of  $I_2(\mathbf{y})$  is equivalent to the minimization of the sum of squares of all cross-cumulants of the same order  $p$ .

Later in [5],[6], it was shown that squaring the cumulants in (6) is not necessary, hence, the proposed contrast reads

$$I'_1(\mathbf{y}) = \sum_{i=1}^N |\mathbf{C}_p[y_i]|. \quad (7)$$

Moreover, when  $p = 4$ , one can still "simplify" to obtain the simplest contrast

$$I''_1(\mathbf{y}) = \varepsilon_4 \sum_{i=1}^N \mathbf{E}[y_i^4] \quad (8)$$

under the assumption that the fourth-order cumulants of all the sources  $a_i$  have the same sign denoted by  $\varepsilon_4$ .

Finally, in [7] extended forms of contrast functions are introduced based on some convex functions of the absolute values of joint cumulants, and in [8] non symmetrical contrasts are defined which allow to exhibit a novel wide class of contrast function whose maximization is proved to be a sufficient condition for separation. Two examples of non symmetrical contrasts are

$$J'_1(\mathbf{y}) = \sum_{i=1}^N \gamma_i |\mathbf{C}_p[y_i]| \quad (9)$$

and

$$J''_1(\mathbf{y}) = \varepsilon_4 \sum_{i=1}^N \gamma_i \mathbf{E}[y_i^4] \quad (10)$$

where  $\gamma_1 \geq \dots \geq \gamma_N > 0$  and assuming that the  $p$ -th order ( $p = 4$  for  $J''_1(\mathbf{y})$ ) cumulants of sources satisfy one of the two following (non restrictive) conditions:

- c1.  $|\mathbf{C}_p[a_1]| \geq \dots \geq |\mathbf{C}_p[a_N]| > 0$ ;
- c2.  $|\mathbf{C}_p[a_1]| \geq \dots \geq |\mathbf{C}_p[a_{N-1}]| > |\mathbf{C}_p[a_N]| = 0$ .

In particular this means that at most one of the cumulants  $\mathbf{C}_p[a_i]$ ,  $i \in \{1, \dots, N\}$ , is null.

It has to be noticed that the following derivation can be used for any contrast using white vectors and thus, in particular, the five hereabove contrasts.

### 4 SOURCE SEPARATION USING A LAGRANGIAN APPROACH

In order to achieve the separation, we intend to maximize a contrast, say  $C(\mathbf{y})$ , which works with normalized (white) vectors  $\mathbf{y}$ , *i.e.* vectors such that  $\mathbf{E}[\mathbf{y}\mathbf{y}^T] = \mathbf{I}$ . We propose to realize the separation in one stage. Hence we consider the normalization of the output vector  $\mathbf{y}$  as a constraint and the contrast based criterion reads

$$\max_{\mathbf{H}} C(\mathbf{y}) \quad \text{subject to} \quad \mathbf{E}[\mathbf{y}\mathbf{y}^T] = \mathbf{I}.$$

It is now possible to write an unconstrained criterion as the maximization of the so-called Lagrangian function defined as

$$L(\mathbf{y}) = C(\mathbf{y}) + \text{trace}[\mathbf{\Lambda}(\mathbf{E}[\mathbf{y}\mathbf{y}^T] - \mathbf{I})] \quad (11)$$

where  $\mathbf{\Lambda}$  is a symmetrical matrix referred to as the Lagrange multiplier.

At the optimum, we have

$$\frac{\partial L}{\partial \mathbf{H}} = 0 \quad \text{and} \quad \frac{\partial L}{\partial \mathbf{\Lambda}} = 0.$$

The first condition is a (first order) necessary condition for  $\mathbf{H}$  to be a maximum of  $C$  under the constraint.

Using (2),  $L(\cdot)$  can be written

$$L(\mathbf{H}) = C(\mathbf{H}) + \text{trace}[\mathbf{\Lambda}(\mathbf{H}\mathbf{E}[\mathbf{x}\mathbf{x}^T]\mathbf{H}^T - \mathbf{I})] \quad (12)$$

where we do, for simplicity, some abuse of notation writing  $L(\cdot)$  as a function of  $\mathbf{H}$ . Hence

$$\frac{\partial L}{\partial \mathbf{H}} = \frac{\partial C}{\partial \mathbf{H}} + (\mathbf{\Lambda} + \mathbf{\Lambda}^T)\mathbf{H}\mathbf{E}[\mathbf{x}\mathbf{x}^T]$$

and

$$\frac{\partial L}{\partial \mathbf{\Lambda}} = \mathbf{H}\mathbf{E}[\mathbf{x}\mathbf{x}^T]\mathbf{H}^T - \mathbf{I}.$$

The optimum value of  $\mathbf{H}$  and  $\mathbf{\Lambda}$  satisfy

$$\frac{\partial C}{\partial \mathbf{H}} + (\mathbf{\Lambda} + \mathbf{\Lambda}^T)\mathbf{H}\mathbf{E}[\mathbf{x}\mathbf{x}^T] = 0 \quad (13)$$

and

$$\mathbf{H} \mathbf{E}[x x^T] \mathbf{H}^T = \mathbf{I} . \quad (14)$$

Right multiplying (13) by  $\mathbf{H}^T$  and using (14) leads to

$$(\mathbf{\Lambda} + \mathbf{\Lambda}^T) = -\frac{\partial C}{\partial \mathbf{H}} \mathbf{H}^T = -\nabla_{\mathbf{H}} C \quad (15)$$

where  $\nabla_{\mathbf{H}} C$  is referred to as a “relative gradient” [1].

Transposing (15), we have

$$\mathbf{\Lambda} + \mathbf{\Lambda}^T = -(\nabla_{\mathbf{H}} C)^T$$

and then

$$\nabla_{\mathbf{H}} C = (\nabla_{\mathbf{H}} C)^T \quad (16)$$

Thus an optimal solution of  $L(\cdot)$  is solution to (16). Unfortunately (16) seems to be difficult to solve. Let us now consider two algorithms searching for a local minimum of  $L(\cdot)$ .

## 5 TWO LAGRANGIAN ALGORITHMS

The optimal values of  $\mathbf{H}$  and  $\mathbf{\Lambda}$  can not be derived from the above development. They have thus to be estimated.

Classically, the two equations of the Lagrange programming are given by

$$\Delta \mathbf{H} = \mu_n \frac{\partial L}{\partial \mathbf{H}} \quad (17)$$

$$\Delta \mathbf{\Lambda} = -\mu_l \frac{\partial L}{\partial \mathbf{\Lambda}} \quad (18)$$

The first equation corresponds to the maximization of  $L$  with respect to  $\mathbf{H}$  ( $\mathbf{\Lambda}$  “constant”) while the second equation reach for a minimum of  $L$  with respect to  $\mathbf{\Lambda}$  ( $\mathbf{H}$  “constant”) according to the “convex duality” theory. The algorithm based on (17) and (18) will be denoted by LAGC.

Now equation (17) corresponds to a classic gradient optimization scheme. Following some ideas of [1], we can also propose the use of a relative gradient as

$$\Delta \mathbf{H} = \mu'_n \nabla_{\mathbf{H}} L \quad (19)$$

and the algorithm based on (19) and (18) will be denoted by LAGR.

For simplicity, we only consider the use of the “simplest” contrast in (8) writing now

$$C(\mathbf{y}) = \varepsilon_4 \sum_{i=1}^N \mathbf{E}[y_i^4] \quad (20)$$

recalling  $\varepsilon_4 = -1$  (resp.  $+1$ ) when sources have negative (resp. positive) fourth-order cumulant. In that case, we easily have

$$\frac{\partial C}{\partial \mathbf{H}} = 4\varepsilon_4 \mathbf{E}[f(\mathbf{y}) \mathbf{x}^T]$$

and using (2)

$$\nabla_{\mathbf{H}} C = 4\varepsilon_4 \mathbf{E}[f'(\mathbf{y}) \mathbf{y}^T]$$

where  $f(\cdot)$  is the component wise non linear cubic function.

In practice, a stochastic gradient algorithm is used by dropping one expectation operator in (17), (18), (19). Hence for the specific contrast in (20), the stochastic Lagrangian algorithm LAGC reads

$$\begin{aligned} \Delta \mathbf{H}_k &= \mu_n \left[ 4\varepsilon_4 f(\mathbf{y}_k) + (\mathbf{\Lambda}_{k-1} + \mathbf{\Lambda}_{k-1}^T) \mathbf{y}_k \right] \mathbf{x}_k^T \\ \Delta \mathbf{\Lambda}_k &= -\mu_l \left[ \mathbf{y}_k \mathbf{y}_k^T - \mathbf{I} \right] \end{aligned} \quad (21)$$

where  $\Delta \mathbf{H}_k = \mathbf{H}_k - \mathbf{H}_{k-1}$  and  $\Delta \mathbf{\Lambda}_k = \mathbf{\Lambda}_k - \mathbf{\Lambda}_{k-1}$  and the stochastic algorithm LAGR reads

$$\Delta \mathbf{H}_k = \mu'_n \left[ 4\varepsilon_4 f(\mathbf{y}_k) + (\mathbf{\Lambda}_{k-1} + \mathbf{\Lambda}_{k-1}^T) \mathbf{y}_k \right] \mathbf{y}_k^T \mathbf{H}_{k-1}$$

together with (21).

## 6 COMPUTER SIMULATIONS

In order to illustrate the two hereabove algorithms we present computer simulations using the following mixing matrix

$$\mathbf{G} = \begin{pmatrix} 1 & 0.6 \\ 0.4 & 1 \end{pmatrix} . \quad (22)$$

The algorithm performance is evaluated thanks to the positive index [5]

$$\begin{aligned} i(\mathbf{S}) &= \frac{1}{2} \left[ \sum_i \left( \sum_j \frac{|S_{ij}|^2}{\max_{\ell} |S_{i\ell}|^2} - 1 \right) \right. \\ &\quad \left. + \sum_j \left( \sum_i \frac{|S_{ij}|^2}{\max_{\ell} |S_{\ell j}|^2} - 1 \right) \right] \end{aligned} \quad (23)$$

which equals zero when perfect separation is realized.

Two cases of sources are considered:

*Case 1:* The sources are binary. Using Monte Carlo runs, fig. 1 shows the mean of the index over 100 independent realizations and with respect to discrete time. Given the same convergence speed (defined here as the time for the algorithms to reach an index of -15dB), the figure clearly shows that the two novel algorithms work well in comparison to the EASI algorithm (the complexity being approximately the same).

*Case 2:* The sources take the four values  $\pm 1/\sqrt{5}, \pm 3/\sqrt{5}$  with equal probability. Using Monte Carlo runs, fig. 2 shows the mean of the index over 100 independent realizations and with respect to discrete time. The figure shows that the LAGR algorithm leads to a better mean index after convergence in comparison to the EASI algorithm.

In conclusion, two novel adaptive stochastic gradient based algorithms are derived using a Lagrangian approach. The algorithms only require one stage for separation and the approach is general in the sense that

it can be used with any contrasts working with normalized vectors. The computer simulation shows good performances in comparison to the EASI algorithm.

Acknowledgment: The second author would like to thank Dr. J.-F. Cardoso for fruitful discussions related to this work.

## References

- [1] J.F. Cardoso and B.H. Laheld, "Equivariant Adaptive Source Separation", *IEEE Transactions on Signal Processing*, Vol. 44, no. 12, pp 3017-3030, December 1996.
- [2] P. Comon, "Independent component analysis, a new concept?", *Signal Processing*, Vol. 36, pp 287-314, 1994.
- [3] C. Jutten and J. Herault, "Blind Separation of Sources: An adaptive algorithm based on neuromimetic architecture" *Signal Processing*, Vol.24, pp 1-10, 1991.
- [4] J. Karhunen and J. Joutsensalo, "Representation and separation of signal using nonlinear PCA Type learning", *Neural Networks*, Vol.7, no.1, pp 113-127, 1994.
- [5] E. Moreau and O. Macchi, "High order contrasts for self-adaptive source separation", *International Journal of Adaptive Control and Signal Processing*, Vol. 10, no.1, pp 19-46, January 1996.
- [6] E. Moreau, "Criteria for complex sources separation", in *proc. EUSIPCO'96*, Trieste, Italy, pp 931-934, September 1996.
- [7] E. Moreau and J.-C. Pesquet, "Generalized contrasts for multichannel blind deconvolution of linear systems", *IEEE Signal Processing Letters*, Vol.4, no. 6, pp 182-183, June 1997.
- [8] E. Moreau and N. Thirion-Moreau, "Non symmetrical contrasts for source separation", submitted to *IEEE Trans. Signal Processing*, December 1997.

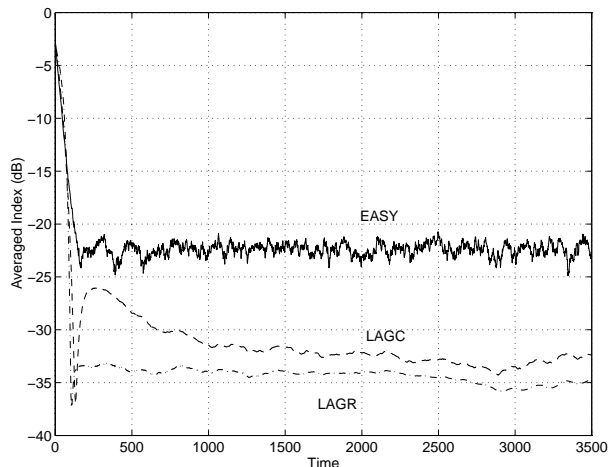


Figure 1: Performance of the two novel algorithms in comparison with EASI in case 1.

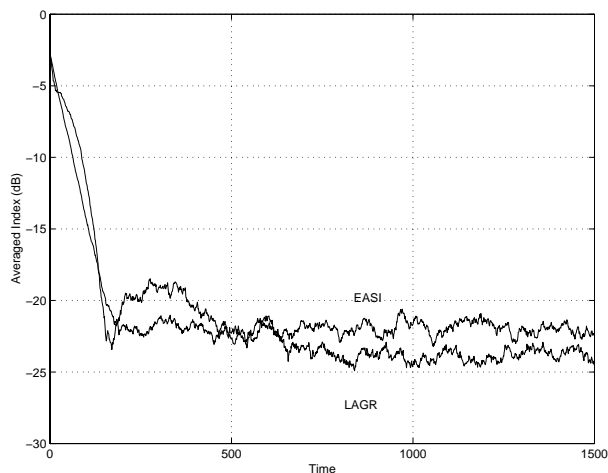


Figure 2: Performance of the LAGR algorithms in comparison with EASI in case 2.

## **Annexe 5: An Iterative Block Procedure For The Optimization Of Constrained Contrast Functions**

*Int. Conf. on Independent Component Analysis (ICA'99),*

pp 59-64, Aussois, Janvier 1999.

# AN ITERATIVE BLOCK PROCEDURE FOR THE OPTIMIZATION OF CONSTRAINED CONTRAST FUNCTIONS

*Eric MOREAU and Benoit STOLL*

MS-GESSY, ISITV, av. G. Pompidou, BP 56,  
F-83162 La Valette du Var Cedex, France  
e-mail: `moreau@isitiv.univ-tln.fr`

## ABSTRACT

We consider the source separation problem based on the maximization of contrast functions. A general iterative procedure for the optimization of any contrasts working on white vectors is proposed. It does not require any parametrization of the separating matrix. Comparisons with the ICA and JADE algorithms by computer simulations illustrate the good performances of this new algorithm and the potential interest of such an approach.

## 1. INTRODUCTION

We consider the source separation problem [5] which can be simply formulated as follows. Several inobservable linear spatial mixtures of some independent signals called “sources” are observed. The purpose is to recover the original sources assuming that the mixing system is unknown.

In the past ten years, numerous solutions to this problem have been proposed. In this paper we are concerned with block (off-line) algorithms based on high-order (higher than two) statistics. Related algorithms are JADE [1] and ICA [3]. The JADE algorithm is based on an algebraical diagonalization of a fourth-order cumulant matrix. The ICA algorithm is based on the optimization of a specific contrast function which is the sum of the squares of outputs auto-cumulants. Moreover this contrast requires white vectors, *i.e.* vectors whose covariance matrix is equal to the identity.

Since [3], numerous contrasts working on white vectors have been proposed, e.g. [6]-[9]. Our main interest in this paper is to propose a general simple iterative procedure for the optimization of any contrasts working on white vectors. Unlike ICA, the proposed algorithm does not require a parametrization of the separating matrix thanks to Givens (planar) rotations. Thus this leads to a “simplified” optimization procedure. The basic idea is to take into account the whitening constraint using a kind of “projected gradient” procedure.

## 2. PROBLEM FORMULATION

We consider the classical linear memoryless mixture model

$$\mathbf{x}(n) = \mathbf{G}\mathbf{a}(n) \quad (1)$$

where  $n \in \mathbb{Z}$  is the discrete time,  $\mathbf{a}(n)$  the  $(N, 1)$  vector of  $N \in \mathbb{N}^* \setminus \{1\}$  inobservable *real* input signals  $a_i(n)$ ,  $i \in \{1, \dots, N\}$ , called sources,  $\mathbf{x}(n)$  the  $(N, 1)$  vector of observed signals  $x_i(n)$ ,  $i \in \{1, \dots, N\}$ , and  $\mathbf{G}$  the  $(N, N)$  square mixing matrix, assumed *invertible*. For clarity, in this article we restrict our attention to the real case although the following derivations may be easily extended to the complex one.

The sources are assumed to satisfy the following three assumptions

**A1a.** “*Independence*” The sources  $a_i(n)$ ,  $i \in \{1, \dots, N\}$ , are zero-mean, unit power and statistically mutually independent;

**A1b.** “*Stationarity*”  $\mathbf{a}(n)$  is a random vector stationary up to order under consideration, *i.e.*  $\forall i \in \{1, \dots, N\}$ , the cumulant  $\text{Cum} \underbrace{[a_i(n), \dots, a_i(n)]}_{p \times}$  is an independent function of  $n$ , denoted by  $C_p[a_i]$ ;

**A1c.** “*At most one null cumulant*” Given the order  $p \in \mathbb{N}^*$ , the cumulants of sources are assumed to satisfy one of the two following conditions:

- c1.  $|C_p[a_1]| \geq \dots \geq |C_p[a_N]| > 0$ ;
- c2.  $|C_p[a_1]| \geq \dots \geq |C_p[a_{N-1}]| > |C_p[a_N]| = 0$ .

In particular this means that at most one of the cumulants  $C_p[a_i]$ ,  $i \in \{1, \dots, N\}$ , is null.

Notice that the unit power assumption and **A1c** can be satisfied without loss of generality because sources are inobservable and the mixture is unknown.

The separation task is realized by computing a vector

$$\mathbf{y}(n) = \mathbf{H}\mathbf{x}(n) \quad (2)$$

in such a way that the different components of the  $(N, 1)$  vector  $\mathbf{y}(n)$  restore the different sources. Hence the separation problem is solved if the global matrix defined by

$$\mathbf{S} \stackrel{\text{def}}{=} \mathbf{H}\mathbf{G} \quad (3)$$

reads

$$\mathbf{S} = \mathbf{D}\mathbf{P} \quad (4)$$

where  $\mathbf{D}$  is an invertible diagonal matrix corresponding to arbitrary attenuations for restored sources and  $\mathbf{P}$  a permutation matrix corresponding to an arbitrary order of restitution. According to (2), (1) and (3) the output vector reads

$$\mathbf{y}(n) = \mathbf{S}\mathbf{a}(n). \quad (5)$$

### 3. ABOUT CONTRAST FUNCTIONS

In the following the explicit dependence of sources, observations and outputs vectors with the discrete time  $n$  will be omitted whenever no confusion is possible.

Before recalling the definition of a contrast, it is useful to define some notations. Let  $\mathcal{A}$  denotes the set of random vectors satisfying assumptions A1a, A1b and A1c. Let  $\mathcal{S}$  denotes the set of invertible matrices. The subset of  $\mathcal{S}$  of matrices  $\mathbf{S}$  of the form (4) is denoted by  $\mathcal{P}$  and the subset of  $\mathcal{P}$  of invertible diagonal matrices is denoted by  $\mathcal{D}$ . The set of random vectors  $\mathbf{y}(n)$  built from (5) where  $\mathbf{a}(n) \in \mathcal{A}$  and  $\mathbf{S} \in \mathcal{S}$  is denoted by  $\mathcal{Y}$ .

As initially defined [3] a contrast has to be a symmetrical and scale invariant function which must be maximized to get separation. Examples of such contrasts can be found e.g. in [3],[6]-[9].

In [9] it is shown that the symmetrical property of contrast is rather restrictive and not really of prime necessity. It is the reason why the following generalized definition of contrast was proposed.

**Definition.** “Non symmetrical contrast” A contrast on  $\mathcal{Y}$  is a multivariate mapping  $\mathcal{I}(\cdot)$  from the set  $\mathcal{Y}$  to  $\mathbb{R}$  which satisfies the following three requirements:

- R1.  $\forall \mathbf{y} \in \mathcal{Y}, \forall \mathbf{D} \in \mathcal{D}, \mathcal{I}(\mathbf{D}\mathbf{y}) = \mathcal{I}(\mathbf{y});$
- R2.  $\forall \mathbf{a} \in \mathcal{A}, \forall \mathbf{S} \in \mathcal{S}, \mathcal{I}(\mathbf{S}\mathbf{a}) \leq \mathcal{I}(\mathbf{a});$
- R3.  $\exists \mathcal{P}_d \subset \mathcal{P}, \mathcal{P}_d \neq \emptyset / \forall \mathbf{a} \in \mathcal{A}, \forall \mathbf{S} \in \mathcal{S}, \mathcal{I}(\mathbf{S}\mathbf{a}) = \mathcal{I}(\mathbf{a}) \Leftrightarrow \mathbf{S} \in \mathcal{P}_d.$

According to this definition, the maximization of a contrast is a sufficient condition for source separation. That is, if  $\mathcal{I}(\cdot)$  is such a contrast, then a sufficient condition of separation is  $\mathcal{I}(\mathbf{y}) = \mathcal{I}(\mathbf{a})$ .

Classical contrasts rely on the important notion of white vectors, *i.e.* vectors  $\mathbf{y}$  such that their covariance matrix  $\mathbf{R}_y \stackrel{\text{def}}{=} \mathbf{E}[\mathbf{y}\mathbf{y}^T]$  equals the identity. Because of

assumption A1a on sources then  $\mathbf{R}_a = \mathbf{I}$ . Hence  $\mathbf{R}_y = \mathbf{I}$  is equivalent to  $\mathbf{S}\mathbf{S}^T = \mathbf{I}$ , *i.e.*  $\mathbf{S}$  is an unitary matrix. Hence from now, we only consider such vectors. For this let  $\mathcal{U}$  denotes the set of unitary matrices and  $\mathcal{Y}_u$  the set of random vectors  $\mathbf{y}(n)$  built from (5) where  $\mathbf{a}(n) \in \mathcal{A}$  and  $\mathbf{S} \in \mathcal{U}$ .

Among the new contrasts (on  $\mathcal{Y}_u$ ) proposed in [9], three of them will be of particular interest in the following. The first one is

$$\mathcal{I}_1(\mathbf{y}) = \sum_{i=1}^N \gamma_i |\mathbf{C}_p[y_i]| \quad (6)$$

where the real numbers  $\gamma_i, i \in \{1, \dots, N\}$ , are assumed to satisfy

$$\gamma_1 \geq \dots \geq \gamma_N > 0. \quad (7)$$

The second contrast

$$\mathcal{I}_2(\mathbf{y}) = \varepsilon_p \sum_{i=1}^N \gamma_i \mathbf{C}_p[y_i] \quad (8)$$

requires an additional assumption on sources, that is

A1d. “Identical sign” When cumulants of sources satisfy condition c1 of A1c, we assume that  $\forall i \in \{1, \dots, N\}$ ,  $\text{sign}(\mathbf{C}_p[a_i]) = \varepsilon_p$  while when they satisfy condition c2 of A1c, we assume that  $\forall i \in \{1, \dots, N-1\}$ ,  $\text{sign}(\mathbf{C}_p[a_i]) = \varepsilon_p$  while by convention  $\text{sign}(\mathbf{C}_p[a_N]) = \varepsilon_p$ .

The third contrast which is perhaps the simplest one, is

$$\mathcal{I}_3(\mathbf{y}) = \varepsilon_4 \sum_{i=1}^N \gamma_i \mathbf{E}[y_i^4]. \quad (9)$$

It also requires assumption A1d.

Finally it has to be noticed that if source  $a_N$  is such that  $\mathbf{C}_p[a_N] = 0$  then in the three hereabove contrasts one can take  $\gamma_N = 0$  while the  $N-1$  other ones satisfy  $\gamma_1 \geq \dots \geq \gamma_{N-1} > 0$ .

### 4. OPTIMIZATION OF A CONSTRAINED CONTRAST

Contrasts on  $\mathcal{Y}_u$ , say  $\mathcal{I}(\cdot)$  for generality (e.g.  $\mathcal{I}_1(\cdot)$ ,  $\mathcal{I}_2(\cdot)$  and  $\mathcal{I}_3(\cdot)$  hereabove defined), assume that vectors  $\mathbf{y}$  are white. Two strategies can be considered for the derivation of a practical maximization algorithm.

The first (classical) one consists in a two stages separation:

- s1. Realize a whitening of the observations:  $\mathbf{x}_b = \mathbf{B}\mathbf{x}$  such that  $\mathbf{R}_{x_b} = \mathbf{I}$ .

**s2.** find a matrix  $\mathbf{H}$  maximizing  $\mathcal{I}(\mathbf{H}\mathbf{x}_b)$  under the constraint  $\mathbf{H}\mathbf{H}^T = \mathbf{I}$ .

The second strategy consists in a one stage separation:

**p1.** find a matrix  $\mathbf{H}$  maximizing  $\mathcal{I}(\mathbf{H}\mathbf{x})$  under the constraint  $\mathbf{R}_y = \mathbf{H}\mathbf{R}_x\mathbf{H}^T = \mathbf{I}$ .

Some algorithms have already been designed to realize the first strategy [3][7]. However, it seems that the second strategy have not yet been considered. Hence following the ideas of [7], we derive hereafter an algorithm corresponding to this second strategy.

A classical way in order to obtain a (local) maximum of a contrast is to use a gradient based algorithm leading to the following recursive scheme

$$\mathbf{H}' = \mathbf{H} + \lambda \dot{\mathbf{I}} \quad (10)$$

where  $\mathbf{H} = (H_{i,j})$  and  $\mathbf{H}'$  are respectively the separating matrices before and after the iteration,  $\lambda$  a positive constant and  $\dot{\mathbf{I}} = \partial\mathcal{I}/\partial\mathbf{H}$  the matrix whose  $(\ell, m)$  component is  $\partial\mathcal{I}/\partial H_{\ell,m}$ . This is referred to as a classical gradient approach.

However, and because parameters we are looking for are matrices, one can also consider a relative or natural gradient approach [2]. This leads to the following recursive scheme

$$\mathbf{H}' = \left( \mathbf{I} + \lambda \dot{\mathbf{I}} \mathbf{H}^T \right) \mathbf{H} . \quad (11)$$

Clearly if  $\mathbf{H}$  is an unitary matrix (as in the first strategy), (10) and (11) are identical. However if not (as in the second strategy) this leads to a different algorithm. In the following, we only consider the estimation in (11).

Let us now consider specifically contrasts  $\mathcal{I}_1(\cdot)$ ,  $\mathcal{I}_2(\cdot)$  and  $\mathcal{I}_3(\cdot)$ . Differentiating  $\mathcal{I}_1(\cdot)$  and  $\mathcal{I}_2(\cdot)$  with respect to  $H_{\ell,m}$  leads to

$$\frac{\partial\mathcal{I}_k}{\partial H_{\ell,m}} = \sum_{i=1}^N \gamma_i \beta_{k,i} \frac{\partial\mathcal{C}_p[y_i]}{\partial H_{\ell,m}}, \quad k = 1, 2 \quad (12)$$

where  $\beta_{1,i} = \text{sgn}(\mathcal{C}_p[y_i])$  is assumed constant and  $\beta_{2,i} = \varepsilon_p$ . According to (2), we have

$$\frac{\partial\mathcal{C}_p[y_i]}{\partial H_{\ell,m}} = p \delta[i, \ell] \text{Cum} \underbrace{[y_i, \dots, y_i, x_m]}_{(p-1) \times} \quad (13)$$

where  $\delta[i, \ell] = 1$  if  $\ell = i$  and 0 otherwise. Using (13) in (12), we finally have

$$\frac{\partial\mathcal{I}_k}{\partial H_{\ell,m}} = p \gamma_\ell \beta_{k,\ell} \text{Cum} \underbrace{[y_\ell, \dots, y_\ell, x_m]}_{(p-1) \times} . \quad (14)$$

Now differentiating  $\mathcal{I}_3(\cdot)$  with respect to  $H_{\ell,m}$  one has

$$\frac{\partial\mathcal{I}_3}{\partial H_{\ell,m}} = 4\varepsilon_2 \gamma_\ell \mathbb{E}[y_\ell^3 x_m^*] . \quad (15)$$

Recalling that  $\mathbf{H}$  has to respect certain constraints throughout the iterations ( $\mathbf{H}\mathbf{R}_x\mathbf{H}^T = \mathbf{I}$  for the second strategy considered), it is easily seen that the iterative procedures in (10) (or (11) in other respects) do not keep these constraints, *i.e.*  $\mathbf{H}'\mathbf{R}_x\mathbf{H}'^T \neq \mathbf{I}$ .

That is why a second step of “projection” is needed in the algorithm. It corresponds to a normalization of  $\mathbf{H}'$  in order to maintain the constraint throughout the iterations. This normalization will be denoted by  $\mathcal{N}(\mathbf{H}')$  in the following.

Because  $\mathbf{H}'\mathbf{R}_x\mathbf{H}'^T \neq \mathbf{I}$ , the reconstructed signals  $\mathbf{y}' = \mathbf{H}'\mathbf{x}$  built from  $\mathbf{H}'$  does not have a covariance matrix equal to the identity. Thus one would like to find a vector  $\mathbf{y}$  which is the closest to  $\mathbf{y}'$  while having an identity covariance matrix. For this task let us define an error vector as

$$\mathbf{e} = \mathbf{y}' - \mathbf{y} = (\mathbf{H}' - \mathbf{H}) \mathbf{x} . \quad (16)$$

Our objective is now to determine a matrix  $\mathbf{H}$  satisfying  $\mathbf{H}\mathbf{R}_x\mathbf{H}^T = \mathbf{I}$  and such that the energy of  $\mathbf{e}$  is minimum. We have the following result

**Proposition 1.** *A matrix  $\mathbf{H}$  satisfying  $\mathbf{H}\mathbf{R}_x\mathbf{H}^T = \mathbf{I}$  which minimizes the error energy  $E_e = \mathbb{E}[\mathbf{e}^T \mathbf{e}]$  where  $\mathbf{e} = (\mathbf{H}' - \mathbf{H}) \mathbf{x}$  for a given matrix  $\mathbf{H}'$ , is given by  $\mathbf{H} = \mathbf{U}_t \mathbf{V}_t^T \mathbf{M}^{-1}$  where  $\mathbf{M}$  is a square root of  $\mathbf{R}_x$  and  $\mathbf{U}_t$  and  $\mathbf{V}_t$  are the unitary matrices derived from the SVD of  $\mathbf{H}'\mathbf{M}$  as  $\mathbf{H}'\mathbf{M} = \mathbf{U}_t \mathbf{D}_t \mathbf{V}_t^T$ . •*

*Proof.* We have from (16)

$$\begin{aligned} E_e &= \mathbb{E}[\mathbf{x}^T (\mathbf{H}' - \mathbf{H})^T (\mathbf{H}' - \mathbf{H}) \mathbf{x}] \\ &= \mathbb{E}[\text{Tr}\{(\mathbf{H}' - \mathbf{H}) \mathbf{x} \mathbf{x}^T (\mathbf{H}' - \mathbf{H})^T\}] \\ &= \text{Tr}\{(\mathbf{H}' - \mathbf{H}) \mathbf{R}_x (\mathbf{H}' - \mathbf{H})^T\} \\ &= \text{Tr}\{\mathbf{H}'\mathbf{R}_x\mathbf{H}'^T + \mathbf{H}\mathbf{R}_x\mathbf{H}^T - \mathbf{H}'\mathbf{R}_x\mathbf{H}^T - \mathbf{H}\mathbf{R}_x\mathbf{H}'^T\} \\ &= C - E'_e \end{aligned} \quad (17)$$

where  $C = \text{Tr}\{\mathbf{H}'\mathbf{R}_x\mathbf{H}'^T\} + \text{Tr}\{\mathbf{H}\mathbf{R}_x\mathbf{H}^T\}$  (which is also equal to  $\text{Tr}\{\mathbf{H}'\mathbf{R}_x\mathbf{H}'^T\} + \text{Tr}\{\mathbf{I}\}$ ) is a constant term since it does not depend on  $\mathbf{H}$  and where

$$\begin{aligned} E'_e &= \text{Tr}\{\mathbf{H}'\mathbf{R}_x\mathbf{H}^T + \mathbf{H}\mathbf{R}_x\mathbf{H}'^T\} \\ &= \text{Tr}\{\mathbf{H}'\mathbf{R}_x\mathbf{H}^T + (\mathbf{H}'\mathbf{R}_x\mathbf{H}^T)^T\} \\ &= 2 \text{Tr}\{\mathbf{H}'\mathbf{R}_x\mathbf{H}^T\} . \end{aligned} \quad (18)$$



Because  $C$  is constant, the minimization of  $E_e$  is equivalent to the maximization of  $E'_e$  in (18). Let us now reformulate the problem. Let  $\mathbf{M}$  be a square root of  $\mathbf{R}_x$  then we have to find a maximum of

$$E'_e = 2 \operatorname{Tr}\{\mathbf{H}'\mathbf{M}\mathbf{M}^T\mathbf{H}^T\} = 2 \operatorname{Tr}\{\mathbf{B}\mathbf{A}^T\}$$

where  $\mathbf{B} = \mathbf{H}'\mathbf{M}$  and  $\mathbf{A} = \mathbf{H}\mathbf{M}$  has to satisfy  $\mathbf{A}\mathbf{A}^T = \mathbf{I}$ , i.e.  $\mathbf{A}$  is unitary.

This latter problem is a classical algebraical one [4]. Let us briefly recall the solution according to our notation. If  $\mathbf{B} = \mathbf{U}_t\mathbf{D}_t\mathbf{V}_t^T$  is a singular value decomposition of  $\mathbf{B}$ , then

$$\operatorname{Tr}\{\mathbf{B}\mathbf{A}^T\} = \operatorname{Tr}\{\mathbf{U}_t\mathbf{D}_t\mathbf{V}_t^T\mathbf{A}^T\} = \operatorname{Tr}\{\mathbf{D}_t\mathbf{V}_t^T\mathbf{A}^T\mathbf{U}_t\}.$$

Thus  $\operatorname{Tr}\{\mathbf{B}\mathbf{A}^T\} = \sum_{i=1}^N \sigma_i(\mathbf{B})T_{i,i}$  where  $\sigma_i(\mathbf{B})$  are the singular values of  $\mathbf{B}$  and  $\mathbf{T} = (T_{i,j}) = \mathbf{V}_t^T\mathbf{A}^T\mathbf{U}_t$  is a unitary matrix. This sum is clearly maximized when  $T_{i,i} = 1$  for all  $i$ , that is, when  $\mathbf{A} = \mathbf{U}_t\mathbf{V}_t^T$ . Finally, because  $\mathbf{H} = \mathbf{A}\mathbf{M}^{-1}$ , the matrix  $\mathbf{H}$  maximizing  $E'_e$  is thus  $\mathbf{H} = \mathbf{U}_t\mathbf{V}_t^T\mathbf{M}^{-1}$ . •

Hence according to the hereabove Proposition, if we consider the following decomposition of the covariance matrix  $\mathbf{R}_x = \mathbf{M}\mathbf{M}^T$ , then the constraint can be written as  $\mathbf{U}_d\mathbf{U}_d^T = \mathbf{I}$  where  $\mathbf{U}_d = \mathbf{H}\mathbf{M}$ . Further, let us consider the SVD of  $\mathbf{H}'\mathbf{M}$  as  $\mathbf{H}'\mathbf{M} = \mathbf{U}_t\mathbf{D}_t\mathbf{V}_t^T$  where matrices  $\mathbf{U}_t$  and  $\mathbf{V}_t$  are unitary and  $\mathbf{D}_t$  is diagonal, then we can impose

$$\mathcal{N}(\mathbf{H}') = \mathbf{U}_t\mathbf{V}_t^T\mathbf{M}^{-1} \quad (19)$$

which corresponds to the best (in the mean square error sense) choice of a matrix leading an output vector  $\mathbf{y}$  with the required constraint.

Let us now summarize the One Stage Iterative Block Separation algorithm (OSIBS):

#### OSIBS algorithm

1. Estimate the covariance matrix of the observations:  $\widehat{\mathbf{R}}_x$ .
2. Compute the SVD of  $\widehat{\mathbf{R}}_x$ :  $\widehat{\mathbf{R}}_x = \mathbf{U}_r\mathbf{D}_r\mathbf{U}_r^T$ .
3. Compute a square root of  $\widehat{\mathbf{R}}_x$ : for example  $\mathbf{M} = \mathbf{U}_r\mathbf{D}_r^{1/2}\mathbf{U}_r^T$ .
4. Depending on *a priori* information about sources, choose a contrast:  $\mathcal{I}_1(\cdot)$ ,  $\mathcal{I}_2(\cdot)$  or  $\mathcal{I}_3(\cdot)$ , say  $\mathcal{I}(\cdot)$ .
5. Choose an initial value for  $\mathbf{H}$  (e.g.  $\mathbf{H} = \mathbf{I}$ ) and choose a real constant  $\lambda$ . Initialize the outputs:  $\mathbf{y} = \mathbf{H}\mathbf{x}$  and then iterate (a)-(d) until convergence:

- (a) Estimate the derivative of a contrast  $\widehat{\mathcal{I}}$ .
- (b) Compute  $\mathbf{H}' = \left(\mathbf{I} + \lambda \widehat{\mathcal{I}} \mathbf{H}^T\right) \mathbf{H}$ .
- (c) Compute  $\mathbf{H} = \mathcal{N}(\mathbf{H}')$  as defined in (19).
- (d) Compute the estimated source signals  $\mathbf{y} = \mathbf{H}\mathbf{x}$ .

## 5. COMPUTER SIMULATIONS

In order to illustrate the performances of the OSIBS algorithm, some computer simulations are now presented. Contrast  $\mathcal{I}_3(\cdot)$  is used. We consider the case of two sources ( $N = 2$ ) and the two following mixing matrices

$$\mathbf{G}_1 = \begin{pmatrix} 1 & 0.5 \\ 0.4 & 1 \end{pmatrix}, \quad \mathbf{G}_2 = \begin{pmatrix} 1 & 0.95 \\ 0.9 & 1 \end{pmatrix}. \quad (20)$$

A noisy model is considered where the additive noise on each channel is assumed to be a zero-mean, i.i.d. Gaussian random signal with equal power.

The performances of the algorithms are associated to an index/measure defined on the global matrix  $\mathbf{S} = (S_{i,j})$  according to

$$\operatorname{ind}(\mathbf{S}) = \frac{1}{2} \left[ \sum_i \left( \sum_j \frac{|S_{i,j}|^2}{\max_\ell |S_{i,\ell}|^2} - 1 \right) + \sum_j \left( \sum_i \frac{|S_{i,j}|^2}{\max_\ell |S_{\ell,j}|^2} - 1 \right) \right]. \quad (21)$$

Indeed, this non-negative index is zero if  $\mathbf{S}$  satisfies (4) and a small value indicates the proximity to the desired solutions.

For the input sources, two cases are considered:

- i*) the two sources are 2-PAM signals (taking their values in  $\{-1, +1\}$  with equal probability)
- ii*) the two sources are 4-PAM signals (taking their values in  $\{-3/\sqrt{5}, -1/\sqrt{5}, 1/\sqrt{5}, 3/\sqrt{5}\}$  with equal probability).

Concerning the algorithms, we compare 1) the JADE algorithm (Version 1.5 Dec. 1997. for the real case) represented by a dashdot line with plus points; 2) the ICA algorithm represented by a dotted line with x-mark points and 3) the OSIBS algorithm represented by a solid line with circle points.

#### Case 1: Index versus data number

First we plot the mean and the standard deviation (STD) of the estimated index over 500 independent runs as a function of the data number. In Fig.1 for case *i*) of sources and Fig.2 for case *ii*) (resp. Fig.3

and Fig.4 for the same respective case of sources) the mixture  $\mathbf{G}_1$  (resp.  $\mathbf{G}_2$ ) is used and the noise power is kept constant to 0dB (resp.  $-20$ dB). The figures show that performances of the OSIBS algorithm are a bit better in the presented cases.

### Case 2: Index versus noise power

Now the data number is held constant to  $N_d = 500$  and we plot the mean and STD of the estimated index over 500 independent runs as a function of the noise power. In Fig.5 for case *i*) of sources and Fig.6 for case *ii*) (resp. Fig.7 and Fig.8 for the same respective case of sources) the mixture  $\mathbf{G}_1$  (resp.  $\mathbf{G}_2$ ) is used. The figures show that the three algorithms have relatively similar performances with a little advantage in favour of OSIBS.

## 6. CONCLUSION

We have proposed an iterative procedure for the maximization of any kind of contrasts involving white vectors. It does not require a parametrization of the separating matrix which thus leads to a “simplest” algorithm. A comparison based on computer simulations illustrates the potential interest of such an approach at least in using the contrast  $\mathcal{I}_3(\cdot)$ .

## 7. REFERENCES

- [1] J.F. Cardoso and A. Soudoumiac, “Blind Beamforming for non Gaussian Signals”, *IEE Proceedings-F*, Vol. 40, pp 362-370, 1993.
- [2] J.F. Cardoso and B.H. Laheld, “Equivariant adaptive source separation”, *IEEE Trans. Signal Processing*, Vol. 44, No. 12, pp 3017-3030, Dec. 1996.
- [3] P. Comon, “Independent component analysis, a new concept?”, *Signal Processing*, Vol. 36, pp 287-314, 1994.
- [4] R.A. Horn and C.R. Johnson, *Matrix analysis*, Cambridge University Press, 1985.
- [5] C. Jutten and J. Herault, “Blind separation of sources, Part I: An adaptive algorithm based on neuromimetic architecture”, *Signal Processing*, Vol. 24, pp 1-10, 1991.
- [6] E. Moreau and O. Macchi, “High order contrasts for self-adaptive source separation”, *International Journal of Adaptive Control and Signal Processing*, Vol. 10, No. 1, pp 19-46, Jan. 1996.

- [7] E. Moreau, “Criteria for complex sources separation”, in Proc. *EUSIPCO'96, European Signal Processing Conference*, Trieste, Italy, Vol. II, pp 931-934, September 1996.
- [8] E. Moreau and J.-C. Pesquet, “Generalized contrasts for multichannel blind deconvolution of linear systems”, *IEEE Signal Processing Letters*, Vol. 4, No. 6, pp 182-183, June 1997.
- [9] E. Moreau and N. Thirion-Moreau, “Non symmetrical contrasts for source separation”, *IEEE Trans. Signal Processing*, submitted, December 1997.

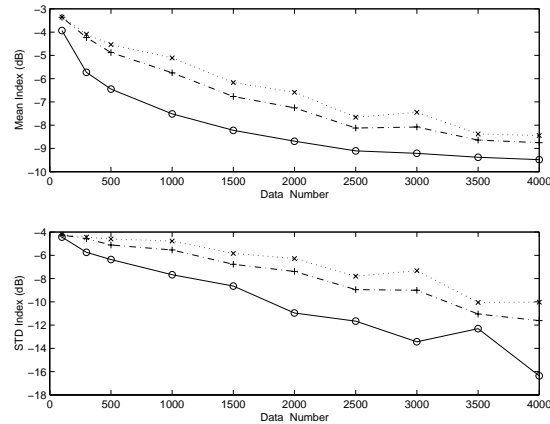


Figure 1: For case *i*) of sources and matrix  $\mathbf{G}_1$ , comparison of the experimental index w.r.t. data number.

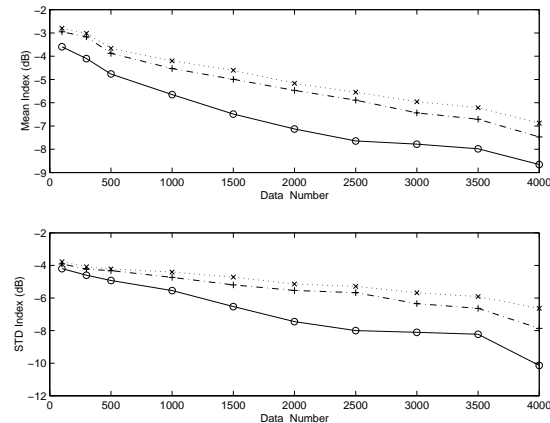


Figure 2: For case *ii*) of sources and matrix  $\mathbf{G}_1$ , comparison of the experimental index w.r.t. data number.

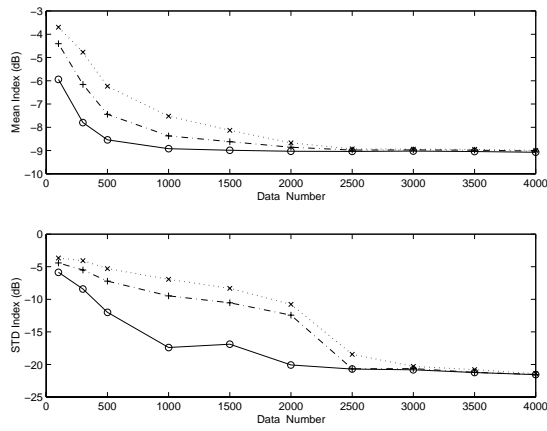


Figure 3: For case *i)* of sources and matrix  $\mathbf{G}_2$ , comparison of the experimental index w.r.t. data number.

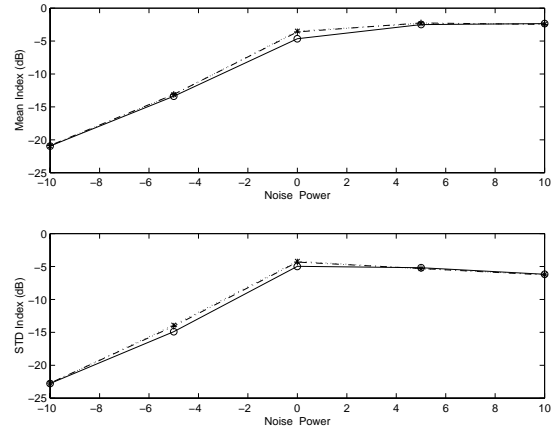


Figure 6: For case *ii)* of sources and matrix  $\mathbf{G}_1$ , comparison of the experimental index w.r.t. noise power.

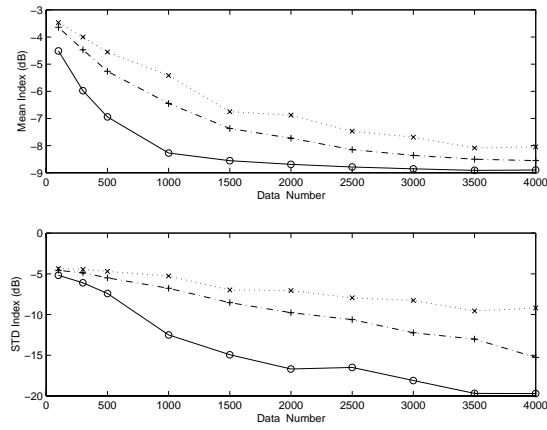


Figure 4: For case *ii)* of sources and matrix  $\mathbf{G}_2$ , comparison of the experimental index w.r.t. data number.

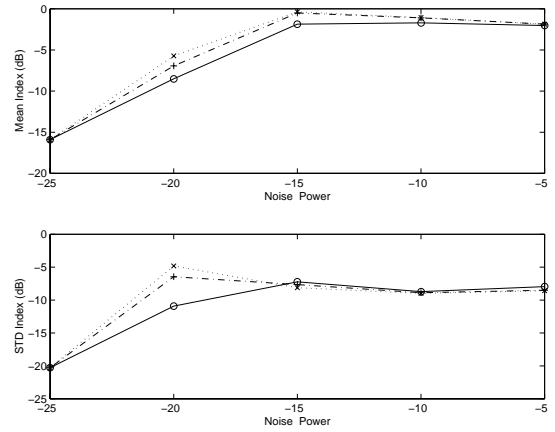


Figure 7: For case *i)* of sources and matrix  $\mathbf{G}_2$ , comparison of the experimental index w.r.t. noise power.

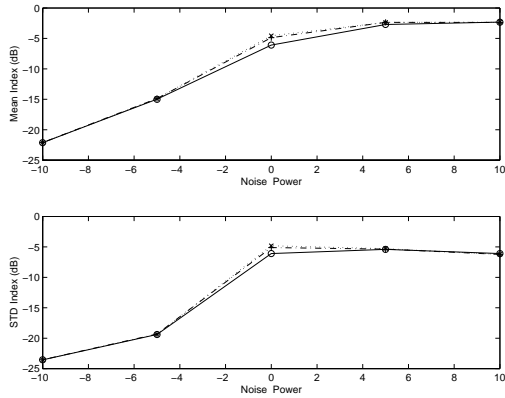


Figure 5: For case *i)* of sources and matrix  $\mathbf{G}_1$ , comparison of the experimental index w.r.t. noise power.

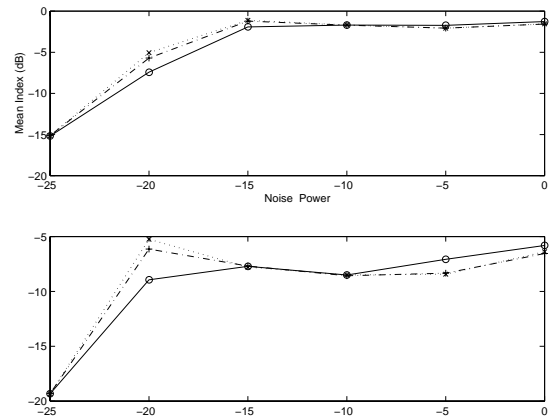


Figure 8: For case *ii)* of sources and matrix  $\mathbf{G}_2$ , comparison of the experimental index w.r.t. noise power.

## Annexe 6: A Generalized ICA Algorithm

accepted for publication *IEEE Signal Processing Letters*,  
January 2000.

# A Generalized ICA Algorithm

Benoit STOLL and Eric MOREAU,<sup>1</sup>

MS-GESSY, ISITV, av. G. Pompidou, BP56,

F-83162 La Valette du Var Cedex, France

moreau@isitv.univ-tln.fr

fax: +33 4 94 14 24 48

Submitted to IEEE Signal Processing Letters, November 1999

**EDICS number:** SPL.SP.3.2

## Abstract

In this paper, we consider the source separation problem through a block algorithm based on the maximization of contrast functions. We propose a new contrast with parameterized cross-cumulants. It allows us to put three classical contrast in a common framework. Following the same spirit of the ICA algorithm, we derive the analytical solution for the case of two sources. Finally, a computer simulation is performed to illustrate the behaviour of a Jacobi-like algorithm for the maximization of the new contrast.

<sup>1</sup>Eric Moreau is the corresponding author.

## I. INTRODUCTION

We consider the source separation problem where several linear spatial mixtures of some independent signals called “sources” are observed as  $\mathbf{x}(n) = \mathbf{G}\mathbf{a}(n)$  where  $n \in \mathbb{Z}$  is the discrete time,  $\mathbf{a}(n)$  the  $(N, 1)$  vector of  $N \in \mathbb{N}^* \setminus \{1\}$  inobservable *real* input sources  $a_i(n)$ ,  $\mathbf{x}(n)$  the  $(N, 1)$  vector of observations  $x_i(n)$  and  $\mathbf{G}$  the  $(N, N)$  square mixing matrix assumed *invertible*. For clarity, we restrict our attention to the real case even though the following derivations could easily be extended to the complex one. The sources  $a_i(n)$  are assumed zero-mean, unit power, statistically mutually independent and stationary, *i.e.* the  $R$ -th order cumulant  $\text{Cum} \underbrace{[a_i(n), \dots, a_i(n)]}_{R \times}$  is an independent function of  $n$ , denoted by  $C_R[a_i]$ . Moreover, we assume that at most one of the cumulants  $C_R[a_i]$  is zero. The separation problem consists in estimating a matrix  $\mathbf{H}$  in such a way that the vector  $\mathbf{y}(n) = \mathbf{H}\mathbf{x}(n)$  restores one of the different input sources on each of its components. An interesting way to get a solution in the source separation problem, is to perform the optimization of so-called “contrast functions” [1],[2]. They have to be maximized to get a separating solution.

In this paper, we consider a block approach based on high-order (higher than two) statistics. In this field, the ICA algorithm [1], the STOTD algorithm [3] and the JADE algorithm [4] constitute three important contributions. These algorithms rely on the optimization of a specific contrast function. They are Jacobi-like algorithms where, after a whitening of the observations, the final unitary matrix to estimate is decomposed in a product of Givens (or plane) rotations. Hence they process all different possible pairs of outputs. For this task, it has required the analytical derivation of the plane rotation component for the case of two sources.

Our main objective in this paper is to show that the three contrasts optimized by ICA, STOTD and JADE belong in a same family of contrast where the cross-cumulants are parameterized. Furthermore we derive the analytical maximization argument of the generalized contrast for the case of two sources and illustrate the behavior of what we call the “generalized ICA” by a computer simulation.

## II. A GENERALIZED CONTRAST

Let us recall that contrast functions was introduced for the source separation problem in [1] and recently generalized in [2]. Basically a contrast is a multivariate function defined on a certain set  $\mathcal{Y}$  of random vectors  $\mathbf{y}$  which only depends on the probability density of  $\mathbf{y}$  and whose global

maxima only correspond to some separation solutions. Hence the (global) maximization of a contrast solves the separation problem. In this paper the set  $\mathcal{Y}$  is the set of white vectors, *i.e.* vectors  $\mathbf{y}(n)$  such that  $\mathbb{E}[\mathbf{y}\mathbf{y}^T] = \mathbf{I}$ . As a consequence [1] the maximization of a given contrast has to be realized onto the set of unitary matrices.

Let us define the following four functions:

$$\begin{aligned} \mathcal{I}(\mathbf{y}) &= \sum_{i_1=1}^N (\mathcal{C}_4[y_{i_1}])^2 ; & \mathcal{C}_1(\mathbf{y}) &= \sum_{\substack{i_1, i_2=1 \\ i_2 \neq i_1}}^N (\text{Cum}[y_{i_1}, y_{i_1}, y_{i_1}, y_{i_2}])^2 ; \\ \mathcal{C}_2(\mathbf{y}) &= \sum_{\substack{i_1, i_2=1 \\ i_2 > i_1}}^N (\text{Cum}[y_{i_1}, y_{i_1}, y_{i_2}, y_{i_2}])^2 ; & \mathcal{C}_3(\mathbf{y}) &= \sum_{\substack{i_1, i_2, i_3=1 \\ i_3 \neq i_2 \neq i_1 \\ i_3 > i_2}}^N (\text{Cum}[y_{i_1}, y_{i_1}, y_{i_2}, y_{i_3}])^2 . \end{aligned}$$

We propose the following result:

**Proposition 1.** *If  $(\alpha_1, \alpha_2, \alpha_3) \in (-\infty, 1]^3$ , the function*

$$\mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{y}) = \mathcal{I}(\mathbf{y}) + 2(\alpha_1 \mathcal{C}_1(\mathbf{y}) + \alpha_2 \mathcal{C}_2(\mathbf{y}) + \alpha_3 \mathcal{C}_3(\mathbf{y})) \quad (1)$$

*is a contrast for white vectors  $\mathbf{y}$ .*

*Proof.* Because  $\alpha_k \leq 1$  for  $k = 1, 2, 3$ , we have  $\mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{y}) \leq \mathcal{J}_{1,1,1}(\mathbf{y})$ . Now recalling that  $\mathcal{J}_{1,1,1}(\mathbf{y})$  is a contrast (see e.g. [4] or [5]) then  $\mathcal{J}_{1,1,1}(\mathbf{y}) \leq \mathcal{J}_{1,1,1}(\mathbf{a})$ . Because the sources are statistically independent their cross-cumulants are zero and thus  $\mathcal{C}_1(\mathbf{a}) = \mathcal{C}_2(\mathbf{a}) = \mathcal{C}_3(\mathbf{a}) = 0$ . Then  $\mathcal{J}_{1,1,1}(\mathbf{a}) = \mathcal{I}(\mathbf{a}) = \mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{a})$  and considering altogether the above results  $\mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{y}) \leq \mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{a})$ . Then the function  $\mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{y})$  has to be maximized. Finally, because  $\mathcal{J}_{1,1,1}(\mathbf{y})$  is a contrast, it is not difficult to see that the equality holds in  $\mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{y}) \leq \mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{a})$  only for separating states. Thus  $\mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{y})$  is a contrast.  $\square$

Let us make three remarks.

**R1:** Let us notice that if  $\alpha_1 = \alpha_2 = \alpha_3 = 0$  then  $\mathcal{J}_{0,0,0}(\mathbf{y}) = \mathcal{I}(\mathbf{y})$  which is the ICA based contrast. Also if  $\alpha_1 = \alpha_2 = \alpha_3 = 1$  then  $\mathcal{J}_{1,1,1}(\mathbf{y})$  is the JADE based contrast. Finally if  $\alpha_1 = 0.5$  and  $\alpha_2 = \alpha_3 = 0$  then  $\mathcal{J}_{\frac{1}{2},0,0}(\mathbf{y})$  is the STOTD based contrast. All other values of  $\alpha_k$ ,  $k = 1, 2, 3$ , yields a new contrast.

**R2:** For simplicity we have only considered fourth-order cumulants. But, using the results in [5], the proposition 1 can be easily generalized to any order of cumulants if it is greater or equal to three.

**R3:** If a given contrast have cross-cumulants then following the same principle of the above proof,

one can parameterized alike its cross-cumulants. This is done here in Proposition 1 using the contrast  $\mathcal{J}_{1,1,1}(\mathbf{y})$ .

### III. THE CASE OF TWO SOURCES

Let us now consider the specific case of  $N = 2$  for which we would like to determine an argument of the maximization of  $\mathcal{J}_{\alpha_1, \alpha_2, \alpha_3}(\mathbf{y})$ . For this task we follow now the same derivations as in [1]. Using the following notation  $Y_{i_1, i_2, i_3, i_4} = \text{Cum}[y_{i_1}, y_{i_2}, y_{i_3}, y_{i_4}]$ , the generalized contrast in (1) can be written as

$$\mathcal{J}_{\alpha_1, \alpha_2}(\mathbf{y}) = Y_{1,1,1,1}^2 + Y_{2,2,2,2}^2 + 2\alpha_1(Y_{1,1,1,2}^2 + Y_{1,2,2,2}^2) + 2\alpha_2 Y_{1,1,2,2}^2 \quad (2)$$

where we have drop the dependency of the contrast w.r.t.  $\alpha_3$  because the term  $\mathcal{C}_3(\mathbf{y})$  does not exist. Assuming that a first whitening stage have been realized on the observations, we have then to determine a unitary matrix  $\mathbf{H}$  which we parameterized as  $\mathbf{H} = \frac{1}{\sqrt{1+\theta^2}} \begin{pmatrix} 1 & \theta \\ -\theta & 1 \end{pmatrix}$ . Introducing the notation  $\xi = \theta - \frac{1}{\theta}$ , the contrast  $\mathcal{J}_{\alpha_1, \alpha_2}(\mathbf{y})$  in (2) can be written as  $\mathcal{J}_{\alpha_1, \alpha_2}(\mathbf{y}) = (\xi^2 + 4)^{-2} \sum_{k=0}^4 b_k \xi^k$  where

$$\begin{aligned} b_4 &= A_1 + 2\alpha_1 A_2 + 2\alpha_2 A_3 \\ b_3 &= 4(2 - \alpha_1)A_4 + 4(3\alpha_1 - 2\alpha_2)A_8 \\ b_2 &= 2(2 + \alpha_1)A_1 + 4(4 - \alpha_1 + 2\alpha_2)A_2 + 4(9\alpha_1 - 2\alpha_2)A_3 + 4(3 - 3\alpha_1 + \alpha_2)A_5 + \\ &\quad 4(6\alpha_1 - 4\alpha_2)A_9 \\ b_1 &= 8(3 - \alpha_2)A_4 + 8(1 - 2\alpha_1 + \alpha_2)A_6 + 16(3 - 3\alpha_1 + \alpha_2)A_8 \\ b_0 &= 2(1 + 2\alpha_1 + \alpha_2)A_1 + 16(2 + \alpha_1)A_2 + 8(9 + \alpha_2)A_3 + 8(3 - \alpha_2)A_5 + \\ &\quad 4(1 - 2\alpha_1 + \alpha_2)A_7 + 32(2 - \alpha_1)A_9 \end{aligned} \quad (3)$$

where, using  $X_{i_1, i_2, i_3, i_4} = \text{Cum}[x_{i_1}, x_{i_2}, x_{i_3}, x_{i_4}]$ , the values of  $A_k$ ,  $k = 1, \dots, 9$ , are

$$\begin{aligned} A_1 &= X_{1,1,1,1}^2 + X_{2,2,2,2}^2 & A_6 &= X_{1,1,1,2}X_{2,2,2,2} - X_{1,1,1,1}X_{1,2,2,2} \\ A_2 &= X_{1,1,1,2}^2 + X_{1,2,2,2}^2 & A_7 &= X_{1,1,1,1}X_{2,2,2,2} \\ A_3 &= X_{1,1,2,2}^2 & A_8 &= X_{1,1,2,2}(X_{1,2,2,2} - X_{1,1,1,2}) \\ A_4 &= X_{1,2,2,2}X_{2,2,2,2} - X_{1,1,1,1}X_{1,1,1,2} & A_9 &= X_{1,1,1,2}X_{1,2,2,2} \\ A_5 &= X_{1,1,2,2}(X_{2,2,2,2} + X_{1,1,1,1}) \end{aligned}$$



Now the derivative of  $\mathcal{J}_{\alpha_1, \alpha_2}$  w.r.t.  $\xi$  is  $\frac{d\mathcal{J}_{\alpha_1, \alpha_2}}{d\xi} = (\xi^2 + 4)^{-3} \sum_{k=0}^4 c_k \xi^k$  where

$$c_4 = -b_3 ; \quad c_3 = 16b_4 - 2b_2 ; \quad c_2 = 12b_3 - 3b_1 ; \quad c_1 = 8b_2 - 4b_0 ; \quad c_0 = 4b_1 .$$

Now one has to calculate the values of  $\xi$  such that  $\sum_{k=0}^4 c_k \xi^k = 0$ . This can be realized in closed-form since it is the zeros of a polynomial of degree 4. Then we keep the  $\xi$  leading to the maximal value of the contrast. Finally, the value of  $\theta$  is determined from the zeros of  $\theta^2 - \xi\theta - 1$  in keeping the one in  $(-1, 1]$ . The Jacobi-like algorithm based on the above developments is denoted gICA( $\alpha_1, \alpha_2$ ) for “generalized ICA” of parameters ( $\alpha_1, \alpha_2$ ). The implementation follows the same lines as the original ICA [1].

#### IV. COMPUTER SIMULATION

In order to illustrate the behaviour of the gICA algorithm for different values of  $\alpha_1$  and  $\alpha_2$  a computer simulation is presented in the case of five sources. We consider discrete i.i.d. signal called PS( $\tau$ ) [5] which takes its values in the set  $\{-1, 0, \tau\}$  with the respective probability  $\{\frac{1}{1+\tau}, \frac{\tau-1}{\tau}, \frac{1}{\tau(1+\tau)}\}$ . This gives a simple way to parameterize the fourth-order cumulant of the source which is  $C_4[a] = \tau^2 - \tau - 2$ . The first two sources are PS(1.5) (negative fourth-order cumulant), the following two are PS(2.5) (positive fourth-order cumulant) and the last one is Gaussian (i.i.d.). We use blocks of  $N_d = 200$  data. The components of the mixing matrix  $\mathbf{G}$  are chosen randomly with an uniform law between  $-1$  and  $+1$  and is kept constant. The condition number of the considered matrix is 5.02. The performances of the algorithm are associated to a non-negative index  $\text{ind}(\cdot)$  [5] defined on the global matrix  $\mathbf{S} = \mathbf{H}\mathbf{G}$ . A small value of  $\text{ind}(\cdot)$  indicates the proximity to one separating solution. The mean and standard deviation (STD) of the estimated index  $\text{ind}(\cdot)$  over 100 Monte-Carlo runs are plotted with a solid line for gICA(0, 0), a dotted line for gICA(0, 1), a dashdot line for gICA(1, 0) and a dashed line for gICA(1, 1) as a function of sweeps<sup>2</sup>. The figure shows that the performance of the gICA algorithm can depend on the parameters ( $\alpha_1, \alpha_2$ ). In the presented case, gICA(1, 0) has the best performances.

#### V. CONCLUSION AND DISCUSSION

We have proposed a new contrast function in parameterizing some cross-cumulants and derived the analytical solution in the case of two sources. A computer simulation illustrate the behavior of the algorithm for four different values of the parameters. Clearly works remain to be done to

<sup>2</sup>A sweep is one “iteration” consisting in processing the outputs through all the possible pairs.

validate the usefulness of the proposed “generalized” algorithm. The simulation show (onto one example) that one could take advantage of the parameters e.g. to derive an optimal contrast. On the other hand, as exemplified in [6] for the ICA and JADE algorithms, performances of this kind of approach can significantly differ for sources with different fourth-order cumulant and with a non-Gaussian background noise. This can be a direction of future works.

## REFERENCES

- [1] P. Comon, “Independent component analysis, a new concept?”, *Signal Processing*, Vol. 36, pp 287-314, 1994.
- [2] E. Moreau and N. Thirion-Moreau, “Nonsymmetrical contrasts for source separation”, *IEEE Trans. Signal Processing*, Vol. 47, No. 8, pp 2241-2252, August 1999.
- [3] L. De Lathauwer, B. De Moor and J. Vandewalle, “Blind source separation by simultaneous third-order tensor diagonalization”, in Proc. *EUSIPCO'96*, Trieste, Italy, pp 2089-2092, Sept. 1996.
- [4] J.F. Cardoso and A. Souloumiac, “Blind Beamforming for non Gaussian Signals”, *IEE Proceedings-F*, Vol. 40, pp 362-370, 1993.
- [5] E. Moreau, “A generalization of joint diagonalization criteria for source separation”, *IEEE Trans. Signal Processing*, submitted, March 1999.
- [6] P. Comon, P. Chevalier and V. Capdevielle, “Performance of contrast-based blind source separation”, in Proc. *SPAWC'97*, Paris, France, pp 345-348, April 1997.

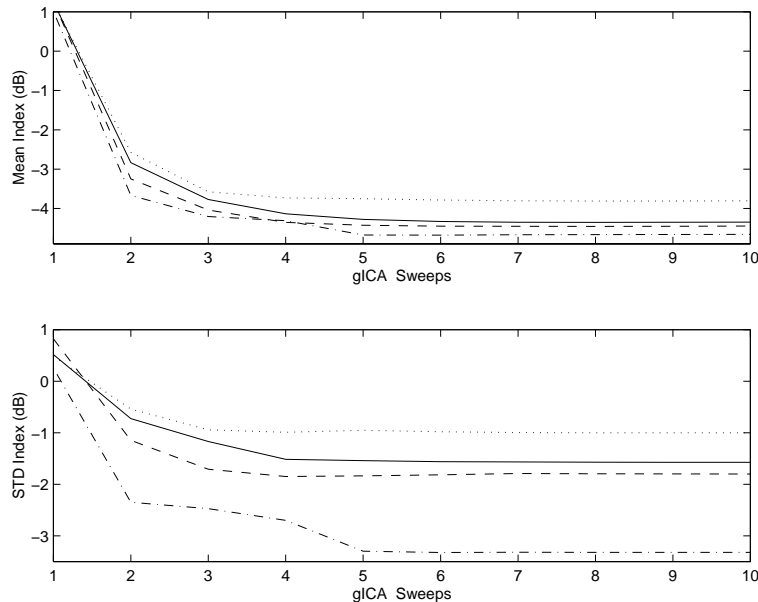


Fig. 1. For  $N = 5$ , the mean and STD of the index over 100 independent runs is plotted for gICA(0,0) (solid line), gICA(0,1) (dotted line), gICA(1,0) (dashdot line) and gICA(1,1) (dashed line) as a function of sweeps.

# Bibliographie

- [AMAR97] S. Amari and J.-F. Cardoso, “ Blind Source Separation. Semiparametric Statistical Approach ”, *IEEE Trans. Signal Processing*, Vol. 45, No. 11, pp 2692-2700, Nov. 1997.
- [BELL98] A. Bell and T. Sejnowsky, “ An Information Maximization Approach to Blind Separation and Blind Deconvolution ”, *Neural Computation*, pp 1129-1159, 1995.
- [BEL97] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso and E. Moulines, “ A Blind Source Separation Technique Using Second-Order statistics ”, *IEEE Trans. Signal Processing*, Vol. 45, No. 2, pp 434-444, February 1997.
- [CAP97] V. Capdevielle, P. Chevalier, P. Calvary et P. Comon, “ Comparaison des Performances de Plusieurs Méthodes de Séparation Aveugle de Sources aux Ordres Supérieurs ”, *GRETSI '97*, Grenoble, Vol. 1, pp 115-118, Sept. 1997.
- [CAR93] J.-F. Cardoso and A. Souloumiac, “ Blind Beamforming for Non Gaussian Signals ”, *IEEE-Proceedings-F*, Vol.140, No. 6, pp 362-370, December 1993.
- [CAR96] J.-F. Cardoso and B.H. Laheld, “ Equivariant Adaptive Source Separation ”, *IEEE Transactions on Signal Processing*, Vol. 44, No. 12, pp 3017-3030, December 1996.
- [CAR98] J.F. Cardoso, “ Blind Signal Separation : Statistical Principles ”, *Proc. IEEE*, Vol. 90, no 8, pp 2009-2026, Oct. 1998.
- [CAR99] J.-F. Cardoso, “ High-Order Contrasts for Independent Component Analysis ”, *Neural Computation*, Vol. 11, No. 1, pp 157-192, Jan. 1999.

- [CAS97] L. Castedo, C.J. Escudero and A. Dapena, “ A Blind Signal Separation Method for Multiuser Communications ”, *IEEE Transactions on Signal Processing*, Vol. 45, No. 5, pp 1343-1348, May 1997.
- [CAVA97] J.-F. Cavassilas, B. Xerri et G. Chabriel, “ Séparation Autodidacte de Sources Temporellement Corrélées ”, *GRETSI '97*, Grenoble, Vol. 1, pp 107-110, Sept. 1997.
- [CAVA98] J.-F. Cavassilas, B. Xerri and G. Chabriel, “ Blind Source Separation with Two-Delayed Mixtures ”, in Proc. *Computational Engineering in Systems Applications (CESA '98) IMACS Multiconferences*, Nabeul-Hammamet, Tunisie, April 1998.
- [CHA97] G. Chabriel, J.-F. Cavassilas and B. Xerri, “ Second Order Blind Identification of Slightly Delayed Mixtures ”, in PROC. *ICA '99*, Aussois, France, pp 75-79, Jan. 1999.
- [CICH96] A. Cichocki and R. Unbehauen, “ Robust Neural Networks with On-Line Learning for Blind Identification and Blind Separation of Sources ”, *IEEE Transactions on circuits and systems*, Vol. 43, pp 894-906, Nov. 1996, .
- [COM91] P. Comon, C. Jutten and J. Herault “ Blind Separation of Sources. Part II : Problems Statement ” *Signal Processing*, Vol. 24, pp 11-20, 1991.
- [COM94] P. Comon, “ Independent Component Analysis, a New Concept ? ”, *Signal Processing*, Vol. 36, pp 287-314, 1994.
- [COM96] P. Comon , “ Contrasts for Multichannel Blind Deconvolution ”. *IEEE Signal Processing Letters*, Vol. 3(7), pp 209-211, July 1996.
- [COM97] P. Comon and E. Moreau, “ Improved Contrast Dedicated to Blind Separation in Communications ”, in Proc. *ICASSP'97*, Munich, Germany, Vol. 5, pp 3453-3456, April 1997.
- [COM4-97] P. Comon, P. Chevalier and V. Capdevielle, “ Performance of Contrast Based Blind Source Separation ”, in Proc. *SPAWC'97*, Paris, France, pp 345-348, April 1997.
- [DEL96] L. DeLathauwer, B. De Moor and J. Vandewalle, “ Blind Source Separation by Simultaneous Third-Order Tensor Diagonalization ”, *European Signal Processing Conference (EUSIPCO'96)*, Trieste, Italy, pp 2089-2092, Sept. 1996.

- [DEL97] L. DeLathauwer, “ Signal Processing Based on Multilinear Algebra ”, *PhD Thesis*, K.U. Leuven, Sept. 1997.
- [DELF95] N. Delfosse and P. Loubaton, “ Adaptive Blind Separation of Independent Sources : A Deflation Approach ”, *Signal Processing*, Vol. 45, pp 59-83, 1995.
- [GAET90] M. Gaeta and J.-L. Lacoume, “ Source Separation Without A-Priori Knowledge : The Maximum Likelihood Solution ”, in *Proc. EUSIPCO'90, European Signal Processing Conference*, Barcelona, Spain, pp. 621-624, 1990.
- [HER85] J. Hérault, C. Jutten et B. Ans, “ Détection de Grandeurs Primitives dans un Message Composite par une Architecture de Calcul Neuromimétique en Apprentissage non Supervisé ”, in *Proc. GRETSI'85*, Nice, France, pp 1017-1022, Mai 1985.
- [HYV97] A. Hyvärinen, “ A Family of Fixed-Point Algorithms for Independent Component Analysis ”, in *Proc. International Conferences on Acoustics, Speech and Signal Processing (ICASP'97)*, Munich, Germany, 1997.
- [HYV99] A. Hyvärinen, “ Gaussian Moments for Noisy Independent Component Analysis ”, *IEEE Signal Processing Letters*, Vol. 6, No. 6, pp 145-147, June 1999.
- [ICA'99] *ICA '99, First International Workshop on Independent Component analysis and signal separation*, Aussois, France, January 1999.
- [JUT91] C. Jutten and J. Herault, “ Blind Separation of Sources. Part I : An Adaptive Algorithm Based on Neuromimetic Architecture ” *Signal Processing*, Vol. 24, No 1, pp 1-10, 1991.
- [KAR94] J. Karhunen and J. Joutsensalo, “ Representation and Separation of Signal Using Nonlinear PCA Type Learning ”, *Neural Networks*, Vol. 7, No.1, pp 113-127, 1994.
- [KAR5-95] J. Karhunen, E. Oja, L. Wang, R. Vigàrio, and J. Joutsensalo, “ A Class of Neural Networks for Independent Component Analysis ”, *IEEE Trans. Neural Networks*, Vol. 8, No. 3, pp 486-504, May 1997.

- [LAC92] J.-L. Lacoume and P. Ruiz, “ Separation of Independent Sources From Correlated Inputs ’, *IEEE Trans. on Signal Processing*, Vol. 40, No. 12, pp 3074-3078, Dec. 1992.
- [LAC97] J.-L. Lacoume, P.-O. Amblard et P. Comon, *Statistiques d’Ordres Supérieurs pour le Traitement de Signal*, Masson, France, Juillet 1997.
- [MAN95] A. Mansour and C. Jutten, “ Fourth Order Criteria for Blind Source Separation ”, *IEEE Trans. Signal Processing*, Vol. 43, pp 2022-2025, Aug. 1995.
- [MAN99] A. Mansour and C. Jutten, “ What Should We Say About the Kurtosis? ”, *IEEE Signal Processing Letters*, Vol. 6, No.12 , Dec. 1999.
- [MEN91] J. Mendel, “ Tutorial on Higher Order Statistics (spectra) in Signal Processing and system Theory : Theoretical Results and some Applications ”, in Proc. *IEEE*, Vol. 3, pp 278-305, March 1991.
- [MIN83] M. Minoux, *Programmation Mathématique. Théorie et Algorithmes Tome 1*, Collection Technique et Scientifique des Télécommunications, Dunod, Paris, 1983.
- [MOR1-96] E. Moreau and O. Macchi, “ High Order Contrasts for Self-Adaptive Source Separation ”, *International Journal of Adaptive Control and Signal Processing*, Vol. 10, No. 1, pp 19-46, January 1996.
- [MOR9-96] E. Moreau, “ Criteria for Complex Sources Separation ”, in *proc. EUSIP-CO’96, European Signal Processing Conference*, Trieste, Italy, Vol. II, pp 931-934, September 1996.
- [MOR6-97] E. Moreau and J.-C. Pesquet, “ Generalized Contrasts for Multichannel Blind Deconvolution of Linear Systems ”, *IEEE Signal Processing Letters*, Vol. 4, No. 6, pp 182-183, June 1997.
- [MOR1-99a] E. Moreau, J.-C. Pesquet and N. Thirion-Moreau, “ An Equivalence Between Non Symmetrical Contrasts and Cumulant Matching For Blind Signal Separation ”, in Proc. *Int. Conf. on Independent Component Analysis (ICA’99)*, Aussois, France, pp 301-306, January 1999.

- [MOR1-99b] Eric Moreau and Benoît Stoll, “ An Iterative Block Procedure For The Optimization Of Constrained Contrast Functions ”, *Int. Conf. on Independent Component Analysis (ICA'99)*, pp 59-64, Aussois, Janvier 1999.
- [MOR3-99] E. Moreau, “ A Generalization of Joint Diagonalization Criteria for Source Separation ”, *IEEE Trans. Signal Processing*, Submitted, March 1999.
- [MOR8-99] E. Moreau and N. Thirion-Moreau, “ Non Symmetrical Contrasts for Source Separation ”, *IEEE Trans. Signal Processing*, Vol.47, No. 8, pp 2241-2252, August 1999.
- [PHAM96] D.T. Pham, “ Blind Separation of Instantaneous Mixture of Sources Via an Independent Component Analysis ”, *IEEE Trans. Signal Processing*, Vol. 44, No. 11, pp 2768-2779, Nov. 1996.
- [PHAM97] D.T. Pham and P. Garat, “ Blind Separation of Mixture of Independent Sources Through a Quasi-Maximum Likelihood Approach ”, *IEEE Trans. Signal Processing*, Vol. 45, No. 7, pp 1712-1725, July 1997.
- [STO98] Benoît Stoll and Eric Moreau, “ Nonlinear Constrained Optimization Using Lagrangian Approach For Blind Source Separation ”, in Proc. *IX European Signal Processing Conference (EUSIPCO'98)*, Island of Rhodes, Greece, pp 2397-2400, Sept. 1998.
- [STO00] Benoît Stoll and Eric Moreau, “ A Generalized ICA Algorithm ”, accepted for publication *IEEE Signal Processing Letters*, January 11th 2000.
- [TAL99] A. Taleb et C. Jutten, “ Source Separation in Post-Nonlinear Mixtures ”, *IEEE Trans Signal Processing*, Vol. 47, No. 10, pp 2807-2820, Oct. 1999.
- [THI96] N. Thirion, J. Mars and J.-L. Boelle, “ Separation of Seismic Signals : A New Concept Based on a Blind Algorithm ”, in Proc. *EUSIPCO'96*, Trieste, Italy, pp 85-88, Sept. 1996.
- [TON91] L. Tong, R.W. Liu, V.C. Sonn and Y.F. Huang, “ Indeterminacy and Identifiability of Blind Identification ”, *IEEE Trans. Circuit and Systems*, Vol. 38, No. 5, pp 499-509, May 1991.

- [YEL94] D. Yellin and E. Weinstein, “ Criteria for Multichannel Signal Separation ”, *IEEE Trans. Signal Processing*, Vol. 42, No. 8, pp 2158-2168, August 1994.



---

## Titre

# Optimisation de Fonctions de Contraste en Séparation de Sources

---

## Résumé

La Séparation de Sources consiste à retrouver un jeu de  $M$  signaux indépendants appelés sources à partir de l'observation de  $N$  de leurs mélanges. Il existe de nombreuses méthodes de séparation qui sont pour la plupart basées sur les statistiques d'ordre supérieures. Ces méthodes permettent d'exploiter l'hypothèse d'indépendance des sources. Dans ce cadre nous avons considéré le cas de la séparation de sources basée sur l'optimisation de fonctions de contraste dans un cas de mélange linéaire purement spatial.

Nous proposons d'abord deux nouvelles familles de contrastes regroupant comme cas particulier des contrastes existants. Puis nous déterminons la solution optimale dans le cas deux sources pour ces deux familles de contrastes. Cela nous permet finalement de proposer deux algorithmes, qui constituent alors deux généralisations d'algorithmes classiques.

Ensuite, nous étudions l'optimisation d'un contraste sous contrainte, afin de proposer des algorithmes ne nécessitant pas, comme dans le cas précédent de blanchiment préalable des données. Deux familles de méthodes directes d'optimisation sous contrainte sont considérées : les méthodes duales et les méthodes directes. Cela nous permet en outre de développer des algorithmes utilisant les notions de Lagrangien, de pénalisation et de projection sur la contrainte. Des simulations informatiques illustrent le comportement des algorithmes proposés.

---

## Title

# Contrasts Functions Optimization for Blind Source Separation

---

## Abstract

Blind Source Separation aim to recover a set of  $M$  independent signals called sources from the observation of  $N$  mixtures. Several Source Separation methods exist, most of them are based on Higher Order Statistics. Those methods exploit the source independence hypothesis. Among them we consider the case of the source separation based on contrast function optimization in a spatial linear mixture case.

We first propose two contrast families including as a particular case some existing contrasts. Then we determine the optimal solution in a two sources case for this couple of contrast families, thus proposing two algorithms which constitute two classic algorithm generalizations.

Then, we study constrained contrast optimization in order to propose algorithms which don't need, as before, data pre-whitening. Two direct constrained optimization method families are considered : the dual methods and the direct methods. Thus we can develop algorithms using Lagrangian concept, penalization concept and a concept of projecting onto the constraint. Computer simulations illustrate the behaviour of the algorithms.

---

## Discipline

Optique, Image et Signal

---

## Mots-Clés

Séparation de Sources, Analyse en Composantes Indépendantes, Statistiques d'Ordre Supérieur, Fonctions de Contrastes, Méthodes Directes, Méthodes Lagrangiennes, Méthodes par pénalisation.

---