

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

Ministère de L'Enseignement Supérieur et de la Recherche Scientifique



UNIVERSITÉ FERHAT ABBAS - SETIF1

FACULTÉ DE TECHNOLOGIE

THÈSE

Présentée au Département d'Electronique

Pour l'obtention du diplôme de

DOCTORAT EN SCIENCES

Option : Electronique

Par

KHALFA Ali

THÈME

**Méthodes avancées pour l'analyse paramétrique
non-stationnaire des signaux de la parole et des
signaux audio.**

Application à la séparation de source informée

Soutenue le 13/02/2020 devant le Jury:

ZIET Lahcene	Professeur	Univ. Ferhat Abbas - Sétif 1	Président
AMARDJIA Nourredine	Professeur	Univ. Ferhat Abbas - Sétif 1	Directeur de thèse
HOCINI Abdessalam	Professeur	Univ. Md Boudiaf - M'sila	Examineur
KHEDROUCHE Djamel	Professeur	Univ. Md Boudiaf - M'sila	Examineur

Remerciements

Je tiens tout d'abord à remercier vivement mon directeur de thèse, Mr. AMARDJIA Nourredine, professeur à l'université Ferhat Abbas de Sétif 1, pour avoir dirigé et encadré cette thèse, ainsi que pour le temps passé à me conseiller. Je le remercie aussi pour sa confiance, sa souplesse et son ouverture d'esprit qui m'ont permis de mener à bien ce travail de recherche.

Je tiens aussi à remercier mon co-encadreur, Mr. CHIKOUCHE Djamel, professeur à l'université Mohamed Boudiaf de M'sila pour ses conseils et l'aide qu'il m'a apportée.

Je tiens à exprimer ma profonde gratitude à Mr. Ziet Lahcène, professeur à l'université Ferhat Abbas de Sétif 1, pour m'avoir fait l'honneur de présider le jury de thèse de mon doctorat et pour l'intérêt et le soutien dont il m'a fait preuve.

Mes remerciements vont aussi à Mr **HOCINI Abdessalam** et Mr **KHEDROUCHE Djamel**, tous les deux professeurs à l'université Mohamed Boudiaf de M'sila, pour avoir accepté d'évaluer et de juger mon travail en tant que membres de jury de ma thèse de doctorat.

Je souhaite aussi remercier mes collègues enseignants chercheurs, Dr. Kenane El Haddi de l'université de M'sila et Dr. Attia Abdelouhab de l'université de Borj-Bou-Arreridj, avec qui j'ai eu la chance de travailler, et toutes les personnes qui ont contribué de près ou de loin à l'aboutissement de mes travaux de recherche.

Enfin, je remercie ma famille, et tout particulièrement ma mère et ma femme, ainsi que mes frères et sœurs. Vous m'avez fait confiance et vous m'avez motivé pour aller toujours plus loin. La réussite de cette thèse est d'ailleurs le fruit de votre soutien, aussi je la partage avec vous.

Résumé

La Séparation Aveugle de Sources (SAS) est un sujet de grand intérêt dans le domaine du traitement du signal qui a été très investi au cours de la dernière décennie. Il est utilisé pour résoudre des problèmes pratiques d'analyse de données dans divers domaines de la science. La SAS est le processus qui vise à séparer un nombre de sources de signaux à partir des mélanges observés de ces sources. L'hypothèse la plus importante est que les composants des mélanges observés sont mutuellement indépendants ou qu'ils sont mutuellement décorrélés. La SAS extrait les sources en exploitant cette hypothèse.

L'objectif de cette thèse est de présenter mais aussi de développer des algorithmes de la SAS. Ces algorithmes seront ensuite évalués en comparant leurs performances en s'appuyant sur un ensemble d'expériences de séparation aveugle de mélanges instantanés de sources de signaux audio. La première partie est consacrée particulièrement aux méthodes Infomax et RobustICA appliquées aux signaux audio, où nous présentons les résultats de simulation ainsi que les critères de performance de chaque méthode. Dans la deuxième partie de cette thèse, l'algorithme HEPHO (High Exploration Particle Swarm Optimization) qui est une amélioration de l'algorithme PSO (Particle Swarm Optimization) est développé puis utilisé pour séparer un ensemble de signaux audio source. Par rapport à l'algorithme PSO, l'algorithme HEPHO dépend de deux opérateurs supplémentaires. Le premier opérateur est basé sur le mécanisme de croisements multiples de l'algorithme génétique, tandis que le second s'appuie sur le mécanisme de la colonie d'abeilles. Les deux opérateurs ont été utilisés pour mettre à jour la vitesse et la position des particules. Ainsi, ils sont utilisés pour trouver la matrice de séparation optimale. La méthode proposée améliore l'efficacité globale de la méthode PSO standard en termes de bonne exploration et de bonnes performances. Les résultats des expériences ont été réalisés en utilisant les fichiers de BSS demo.

Mots clés: Séparation Aveugle de Sources; Signaux de parole et de musique; Infomax, RobustICA; Artificial Bee Colony; Multi-crossover; Particle Swarm Optimization; BSS demo.

Abstract

Blind Source Separation (BSS) is a topic of great interest in the field of signal processing that has been greatly invested over the last decade. It's used to solve practical problems of data analysis in various fields of science. BSS is the process of separating a number of signal sources from the mixtures observed from these sources. The most important hypothesis is that the components of the mixtures observed are mutually independent or mutually decorrelated. The BSS extracts the sources by exploiting this hypothesis.

The objective of this thesis is to present but also to develop BSS algorithms. These algorithms will then be evaluated by comparing their performances based on a set of blind separation experiments of instantaneous mixtures of audio signal sources. The first part focuses on the Infomax and RobustICA methods applied to audio signals, where we present the simulation results as well as the performance criteria of each method. In the second part of this thesis, the High Particle Swarm Optimization (HEPSO) algorithm which is an improvement of the Particle Swarm Optimization (PSO) algorithm is developed and then used to separate a set of audio signals sources. Compared to the PSO algorithm, the HEPSO algorithm depends on two additional operators. The first operator is based on the multiple crossing mechanism of the genetic algorithm, while the second is based on the mechanism of the bee colony. Both operators were used to update the speed and position of the particles. Thus, they are used to find the optimal separation matrix. The proposed method improves the overall efficiency of the standard PSO method in terms of good exploration and performance. The results of the experiments were done using the BSS demo files.

Keywords: Blind Source Separation; Speech and music signals; Infomax; RobustICA; Artificial Bee Colony; Multi-crossover; Particle Swarm Optimization; BSSdemo.

ملخص

يُعد الفصل الأعمى للمصدر (BSS) موضوعًا ذا أهمية كبيرة في مجال معالجة الإشارات التي تم استثمارها بشكل كبير على مدار العقد الماضي. يتم استخدامه لحل المشكلات العملية لتحليل البيانات في مختلف مجالات العلوم. BSS هي عملية فصل عدد من مصادر الإشارة عن المخاليط الملاحظة من هذه المصادر. الفرضية الأكثر أهمية هي أن مكونات الخلائط التي تمت ملاحظتها مستقلة بشكل متبادل أو غير مرتبطة بعضها البعض. تستخرج BSS المصادر من خلال استغلال هذه الفرضية. الهدف من هذه الرسالة هو تقديم خوارزميات BSS ولكن أيضا تطويرها. سيتم بعد ذلك تقييم هذه الخوارزميات بمقارنة أدائها استنادا إلى مجموعة من تجارب الفصل الأعمى للمخاليط الفورية لمصادر الإشارات الصوتية. يركز الجزء الأول على طرق Infomax وrobustica المطبقة على الإشارات الصوتية، حيث نقدم نتائج المحاكاة بالإضافة إلى معايير الأداء لكل طريقة. في الجزء الثاني من هذه الأطروحة، تم تطوير الخوارزمية (HEPSO) والتي يعد تحسينًا لخوارزمية (PSO) و من ثم تستخدم لفصل مجموعة من الإشارات صوتية المصدر. مقارنة بخوارزمية PSO ، تعتمد خوارزمية HEPSO على عاملين إضافيين يعتمد المشغل الأول على آلية تقاطع متعددة للخوارزمية الجينية ، بينما يعتمد المشغل الثاني على آلية مستعمرة النحل. تم استخدام كلا المشغلين لتحديث سرعة وموقع الجزيئات. وبالتالي، يتم استخدامها للعثور على مصفوفة الفصل الأمثل. تعمل الطريقة المقترحة على تحسين الكفاءة العامة لطريقة PSO القياسية من حيث الاستكشاف والأداء الجيد. تم إجراء نتائج التجارب باستخدام ملفات BSSdemo التجريبية.

الكلمات المفتاحية: فصل المصدر العمياء؛ إشارات الكلام والموسيقى؛ RobustICA ; INFOMAX ; مستعمرة النحل الاصطناعية ; متعددة العبور ; تحسين سرب الجسيمات ; BSSdemo.

Table des figures

I.1. Principe de la Séparation Aveugle de sources.....	5
I.2. Séparation aveugle du signal parole.....	9
I.3.Modèle Mélange Linéaire	11
I.4.Modèle des mélanges convolutifs	14
I.5. L'émission, la propagation et la réception du signal Audio.....	17
I.6. L'évolution de l'amplitude sonore dans le temps.....	17
I.7. Les types de fréquences.....	18
I.8. Exemple d'une chaîne numérique pour le signal Audio.....	19
I.9. Echantillonnage d'un signal audio	20
I.10. Signal audio échantillonné avant et après quantification	21
II.1.Stratégie de la méthode Informations Maximisation.....	35
II.2. Transformation de y en Y	36
II.3. Fonction tangente hyperbolique.....	45
II.4. Dérivée tangente hyperbolique.....	45
II.5. Sources original"speech1.wav, speech2.wav".....	48
II.6. Mélanges	49
II.7. Sources Estimés.....	50
II.8. Entropie $H(Y)$, ∇h , η pour $=2$	51
II.9. Sources originals speech1.wav, speech2.wav, speech1.wav.....	55
II.10. Mélanges.....	55
II. 11. Sources Estimés,	56
II. 12. Sources originals music1.wav, music2.wav, music3.wav.	57
II. 13. Mélanges	58
II.14. Sources estimées.....	59
II. 15. Sources originals Speech2.wav, music2.wav, music3.wav).....	60
II.16. Mélanges.....	61
II.17. Sources estimées.....	61
III.1. Exemple de recherche d'une fonction d'une seule variable, avec un seul extremum global(rouge) et plusieurs extrema locaux vert).....	67
III.2. Schéma général d'un algorithme évolutionnaire [III.9]	69
III.3. Migration de la solution de l'espèce	72
III.4. Processus d'immigration et d'émigration dans un habitat [III.16] ...	73
III.5. Schéma de principe de base de l'algorithme Hill Climbing(HC).....	77
III.6. Principe des algorithmes génétiques.....	81
III.7. Les phases de définition d'un codage.....	82
III.8. Croisement en un point de deux chromosomes.....	85
III.9. Croisement en 2-points de deux chromosomes.....	85
III.10. Comportement d'une fourmi naturelle lors de la recherche de nourriture	88

Table des figures

III.11. Configuration expérimentale pour l'expérience à double pont	89
III.12. Une fourmi en ville i choisit la prochaine ville à visiter via un mécanisme stochastique.....	84
IV.1. Le Modèle de base du processus BSS.....	100
IV.2. Concept de base d'une particule de PSO.....	103
IV.3. Schéma de la méthode proposée	105
IV.4. signaux originaux et estimés poème-mâle_30s.wav et sentence_ femelle _28s.wav	108
IV.5. Signaux originaux et estimés de music_Loophass_30s.wav et henry_theater_male_30s.wav.....	109
IV.6. Signaux originaux et estimés de music_mandolin_30s.wav et music_vivaldi_30s.wav	110
IV.7. Signaux poème-mâle_30s.wav et entence_female_28s.wav.....	113
IV.8. Signaux originaux et estimés de la musique loophass.wav et du théâtre henry.male.wav.....	114
IV.9. Signaux originaux et estimés de music_mandolin_30s.wav et music_vivaldi_30s.wav.....	115

Liste des tableaux

II.1.Valeurs des variables de l'algorithme Infomax pour $M=2$	48
II.2. Paramètres de performance de RobustICA.....	62
III.1.Algorithme de la recherche tabou (TS).....	71
III.2. Algorithme optimisation à base de biogéographie.....	74
III.3. L'Algorithme du Hill Climbing (HC).....	76
III. 4. L'algorithme d'optimisation Recuit simulé.....	79
III. 5. Algorithme de base de l'algorithme génétique.....	86
IV.1. SDR (dB) pour l'algorithmes PSO et HEPSO.....	111
IV.2. AVCC pour le PSO et l'algorithmes proposé.....	111
IV.3. MSE pour le PSO et l'algorithmes proposé.....	112
IV.4. SDR (dB) pour la méthode GA + kurtosis et la méthode proposée.....	112
IV5. AVCC pour le GA+ Kurtosis et algorithme.....	112
IV6. SDR (dB) pour l'algorithme HEPSO en présence de bruit.....	116
IV.7. AVCC pour HPSO+Bruit.....	116
IV8. MSE pour HEPSO+Bruit.....	117

Liste des abréviations

AAC	Advanced Audio Coding
ABC	Artificial Bee Colony
ACO	Ant Colony Optimisation
ACS	Ant Colony System
ACI	Analyse Composantes Indépendantes
AE	Algorithmes Evolutionnaires
ALAC	Apple Lossless Audio
AS	Ant system
A-LAW	Système De Quantification logarithmique
AVCC	Valeurs Absolue Des coefficients De corrélation
BBO	Biogeography-Based-Optimization
BSS	Blind Source Separation
CD	Compact Disque
CDMA	Code Division Multiple Access
CSA	Classical Simulated Annealing
DVD	Digital Versatile Disc
EEG	Electrocardiogrammes
EMG	Electroencéphalographie
EMG	Electromyogrammes
FastICA	Fast Independent Component Analysis
FSA	Fast Simulated Annealing
GA	Genetic Algorithm
GTP	Generative Topographique Mappings
GSA	Generalized Simulated Annealing
HC	Hill Climbing
HEPSO	High Exploration Particle Swarm Optimisation
HSI	Habitat Suitability Index
ICA	Independent Component Analysis
INFOMAX	Information Maximization
IRM	Imagerie Par Résonance Magnétique
JADE	Joint Approximate Diagonalization of Eigenmatrices
ML	Maximum de Vraisemblance
MLP	Multi-layer Perceptrons
MP3	MPEG-1 Audio Layer3
MSE	Mean Square Error / Erreur Quadratique Moyenne
NMF	Non Negative Matrix Factorization
OEP	Optimisation Par Essaims Particulaires
PCA	Principal Component Analysis
PDF	Fonction De Densité De Probabilité
PCM	Pulse Code Modulation
PMF	Positive Matrix Factorization
PNN	Polynomial Neural Network
PNL	Post-Non-linéaire
PSO	Particle Swarm Optimization
RBF	Réseaux Fonction Radial
SAS	Séparation Aveugle De source
SA	Simulated Annealing

Liste des abréviations

SCA	Sparse Component Analysis
SOBI	Second Order Blind Identification
SOM	Self Organizing Maps
SIV	Suitability Index Variables
TS	Tabu Search

Table des matières

Remerciements	ii
Résumé	iii
Abstract	iv
ملخص.....	v
Table des figures.....	vi
Liste des tableaux.....	viii
Liste des abréviations	ix
Introduction générale	1
CHAPITRE I : Etat de l’art sur la Séparation Aveugle de Sources	
I.1 Introduction	4
I.2 Description générale de la Séparation Aveugle de Sources (SAS)	4
I.3 Historique de la Séparation Aveugle de Sources	5
I.4 Les applications la SAS.....	8
I.4.1 Le Signal Parole	8
I.4.2 Les systèmes de communication voix/données et Traitement de signal	9
I.4.3 Le traitement et la reconnaissance d'images	9
I.4.4 Le traitement de l'information géologique spatiale	10
I.4.5 Le traitement du signal biomédical	10
I.4.6 L’analyse de documents texte	11
I.5 Les types de mélanges	11
I.5.1 Les mélanges linéaires	11
I.5.1.1 La séparabilité et l’indétermination	12
I.5.1.2 Les critères d’indépendance	13
I.5.1.3 L’information mutuelle.....	13
I.5.1.4 La non-gaussianité	13
I.5.2 Les mélanges convolutifs.....	14
I.5.3 Les mélanges non linéaires	15
I.6 Le Signal Audio.....	16
I.6.1 Emission, propagation et réception d’un signal audio	17
I.6.2 Les caractéristiques d’un signal audio.....	17
I.6.2.3 Le Signal Audio en numérique.....	18
I.6.2.4 La fréquence d’échantillonnage d’un signal audio	19
I.6.2.5 La quantification d’un signal audio	20
I.6.2.6 La compression du signal audio numérisé	21

Table des matières

I.6.2.7 Les algorithmes de compression	22
• La compression sans pertes	22
• La compression avec pertes	23
I.7 Les format des fichiers audio	23
I.7 .1 Les caractéristique des formats audio	23
I.7.2 Les types de formats	24
I.7.2.1 Les formats audio avec compression sans pertes	24
I.7 .2.2 Les formats audio avec compression avec pertes	25
I. 7.2.3 Les formats audio non compressés	25
I.8 Conclusion	26
Bibliographie	27
CHAPITRE II: Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux	
Audio.	
II.1 Introduction	34
II.2 Méthode Informations Maximisation (Infomax)	34
II.2.1 Entropie d'une fonction de densité de probabilité univariée	35
II.2.2 Entropie d'une fonction de densité de probabilité multivariée	37
II.2.2.a. Le Jacobien	38
II.2.2.b. Le Jacobien et la matrice de séparation	38
II.2.3 Le Gradient Ascent	41
II.2.3. a. Le Gradient d'entropie	41
II.2.3. b. L'Algorithme Gradient Ascent Infomax	44
II.3.1 Modélisation et Simulation	44
II.3.2 Adaptation de l'algorithme Infomax	45
II.3.3 Organigramme de la méthode Infomax	47
II.3.4 Paramètres de la méthode Infomax	47
II.4 Audio Séparation utilisant la méthode Robust ICA	51
II.4.1 Modèle du signal	51
II. 4.2 La méthode Fast ICA modifiée	52
II.4.3 Optimisation du contraste	52
II. 4.4 La méthode Robust ICA	53
II.4 .5 L'Algorithme de la Méthode Robust ICA	54
II.4.6 Simulations	54
Simulation 1	54
Simulation 2	56

Table des matières

Simulation 3	59
II.5 Conclusion	62
Bibliographies	63

CHAPITRE III : Algorithmes Evolutionnaires et Optimisation

III.1 Introduction	66
III.2 Terminologie et notations	68
III.2.1 Schéma général d'un algorithme évolutionnaire	68
III.3 La recherche tabou.....	69
III.3.1 Concepts de base de la recherche tabou	70
III.4 Algorithme d'optimisation à base de la Biogéographie (BBO).....	71
III.4.1 Principe de fonctionnement de la BBO.....	72
III.4. 2 La Migration.....	72
III.4. 3 La mutation	73
III.4. 4 L'Élitisme	74
III.4. 5 L'algorithme de la BBO.....	74
L'algorithme de base de BBO est comme suit [III.17] :	74
Tableau III.2. Algorithme optimisation à base de biogéographie.....	74
III.5 L'algorithme d'ascension de collines (Hill Climbing).....	75
III.6 L'algorithme du Recuit Simulé	78
III.6.1 Interprétation.....	79
III.7 Les algorithmes génétiques.....	80
III.7.1 Implantation de l'algorithme génétique	83
III.7.1 .1 La population initiale	83
III.7.1 .2 La fonction d'objectif.....	83
III.7.1.3 Le chromosome.....	84
III.7.1 .4 La sélection.....	84
III.7.1.5 Le croisement	84
III.7.1 .5.1 Le croisement en 1-point.....	85
III.7.1.5.2 Le croisement en 2 points	85
III.7.1 .5 La mutation	86
III.7.2 L'hybridation.....	86
III.8 Optimisation par colonie de fourmis.....	87
III.8.1 Introduction.....	87
III.8.2 Intelligence Commune des Fourmis.....	88

Table des matières

III.8.2.1 Technique d'optimisation	88
III.8.2.2 L'algorithme ACO pour le problème du voyageur commercial.	90
III.8.2.3 Les algorithmes ACO.....	91
III.8.2.4 Applications de l'algorithme de l'optimisation de la colonie de fourmis	93
III.9 Conclusion	94
Bibliographie	95

CHAPITRE IV : Séparation aveugle de sources audio basée sur l'algorithme HEPSO

IV. Introduction.....	99
IV.1 Problème de la SAS	100
IV.2 Optimisation par Essaims Particulaires (OPE).....	101
IV.3 Principe de l'algorithme PSO.....	102
IV.4 Évaluation de la fonction objective	104
IV.5 Méthode Proposée [IV.19]	104
IV.5.1 SAS basée sur l'algorithme HEPSO	105
IV.5.2 Les Opérateurs	106
IV.5.2.1 L'Opérateur 1	106
IV.5.2.2 L'Opérateur 2	106
IV. 6 Résultats de simulation	107
IV.6.1 Les signaux source	107
IV.6.2 Les paramètres de simulation	107
IV.6.3 Les mesures de performance	107
V.7 Interprétations	110
IV.7.1 Séparation des signaux en présence de bruit.....	113
IV.7.2 Interprétations	117
IV.8 Conclusion	117
Bibliographie	118
Conclusion générale et perspectives	120

Introduction générale

La séparation aveugle de sources (SAS ou BSS pour Blind Source Separation en anglais) est un sujet relativement nouveau dans le traitement des signaux et qui a été largement considéré depuis le milieu des années 80. Il consiste à récupérer des sources indépendantes non observées provenant de mélanges de ces sources, en supposant que ces informations observées ne concernent ni les sources originales, ni le système de mélange. La SAS peut être appliquée à une variété de situations telles que l'analyse des signaux biomédicaux obtenus par EEG ou dans les télécommunications sans fil pour séparer plusieurs signaux reçus. Elle est aussi appliquée dans la séparation de sons en diaphonie et dans la suppression du bruit additif des signaux et des images. Actuellement, au niveau mondial, les applications se concentrent sur trois domaines : le biomédical, l'audio et les télécommunications. Un des avantages de la séparation de sources est de ne pas nécessiter la connaissance d'un signal de référence, comme c'est usuel dans les méthodes classiques du traitement du signal. De plus, il est possible de séparer des sources de distributions identiques. Une des branches de la SAS concerne le problème d'un certain nombre de personnes qui parlent simultanément dans une salle et où on essaie de suivre une des discussions. Il s'avère alors qu'il est très difficile d'écouter ceux qui parlent et de discerner ou séparer les différentes voix. Un système de traitement de signal numérique peut alors être développé pour extraire les voix des personnes qui discutent. Ce système permettra de faire la séparation des différentes voix sans en connaître leurs originales d'où l'appellation aveugle.

Le but de cette thèse est d'exploiter et aussi de développer des algorithmes qui simulent les éléments de la matrice de séparation pour avoir une matrice optimale qui rend les sources indépendantes.

Ce manuscrit est organisé en quatre chapitres. Le premier chapitre présente l'état de l'art de la SAS. Nous commençons par donner une description générale de la SAS, les types de mélanges et leurs modèles mathématiques, ainsi que les applications de la SAS. Les caractéristiques du signal audio sont ensuite introduites et discutées. Les signaux audio seront utilisés dans les simulations réalisées et présentées dans le dernier chapitre. Le chapitre II est entièrement consacré à deux méthodes de séparation aveugle. Nous commençons par la méthode Informations Maximisation (Infomax) qui est une méthode de recherche de signaux mutuellement indépendants par la maximisation du flux d'information ou de l'entropie. Elle est basée sur la théorie de l'information qui vise à trouver des signaux sources indépendants en

maximisant l'entropie. Une deuxième méthode, appelée RobustICA, est aussi étudiée. Cette technique simple consiste à optimiser la fonction de contraste de Kurtosis. Il faut néanmoins noter que ces deux méthodes se basent sur l'algorithme du gradient (gradient descent algorithm) qui présente des inconvénients, tels que la présence de minima locaux qui empêchent de trouver la solution optimale : le minimum global est alors masqué par un minimum local et certains paramètres ne sont valides que dans un certain intervalle. Cela nécessite une fonction objective dérivable pour l'optimisation. Mais dans le cas réel, il est intéressant d'optimiser des fonctions dont la formule analytique est inconnue. Cependant, la dérivée de la fonction que l'on cherche à optimiser peut-être difficile à calculer, ou même ne pas exister du tout. Pour pallier à ce problème, on pourra faire appel aux algorithmes évolutionnaires. Ainsi, le chapitre III introduit les différents algorithmes évolutionnaires existant dans la littérature et abordés dans cet axe de recherche. Nous les présentons en détails en insistant sur leur rôle d'éviter de tomber dans les minimas locaux. Le chapitre IV décrit une nouvelle méthode d'optimisation évolutionnaire, notée HEPSO (High Exploration Particle Swarm Optimization), que nous avons développée. Des simulations de cette nouvelle technique sur des signaux audio montrent son efficacité par comparaison avec les deux méthodes décrites au chapitre II ainsi qu'avec l'algorithme PSO (Particle Swarm Optimization). Enfin, une conclusion générale est présentée, laquelle inclut le bilan du travail réalisé ainsi que des perspectives de recherche future.

CHAPITRE I

Etat de l'art sur la Séparation Aveugle de Sources

I.1 Introduction

La séparation aveugle de sources (SAS) est un problème qui consiste à retrouver des signaux à partir de leurs mélanges. On peut la comprendre ou l'expliquer par les exemples suivants :

- Dans une fête animée, on a tendance à entendre un mélange de voix, de bruits et de la musique. Si ces sons sont captés par un microphone, comment peut-on identifier une seule voix d'intérêt à partir du son mélangé ?
- Lors de la réception de signaux radio provenant d'un champ de bataille, comment peut-on capturer, séparer et distinguer une communication amie de celles des troupes ennemies ?
- Dans le domaine de la télédétection par l'analyse d'images, les informations de sol, celles de minéral et d'autres informations d'interférence sont mélangées dans le signal reçu. Comment déterminer les informations de sol et de minéral à partir de ce signal mélangé ?

I.2 Description générale de la Séparation Aveugle de Sources (SAS)

La séparation aveugle de sources (SAS) (en anglais BSS pour Blind Source Séparation) est une puissante méthode en traitement du signal qui a été proposée à la fin des années 80 [I.1]. Dérivée à partir des réseaux de neurones artificiels, du traitement statistique du signal et de la théorie de l'information, la SAS est devenue un sujet important de la recherche et de développement dans de nombreux domaines, notamment les sciences biomédicales, les communications, le traitement de la parole, le traitement d'images, les sciences de la Terre, etc.

À l'heure actuelle, les travaux de recherche liés à la SAS ont acquis un grand intérêt dans le domaine du traitement du signal et sont donc devenus d'une grande valeur pratique.

La SAS, qui est un problème classique et complexe dans le traitement des signaux, consiste à extraire et à récupérer les signaux sources à partir de données statistiques à variables multiples. Le mot « Aveugle » signifie que le signal source est inconnu (non observé) et que, soit les caractéristiques du système hybride ne sont pas connues à l'avance, soit qu'il n'y a qu'une petite quantité d'information a priori sur le signal source (comme la non Gaussienne, le cycle-stabilité et l'indépendance statistique).

Dans les applications scientifiques et techniques, de nombreux signaux observés peuvent être perçus comme un mélange d'une pluralité de signaux sources, c'est-à-dire que le signal mixte

Observé est une série de sorties de capteurs, chacun recevant différentes combinaisons de signaux sources. La principale tâche de la SAS est donc de récupérer le signal source qui nous intéresse à partir des données observées Figure I.1.

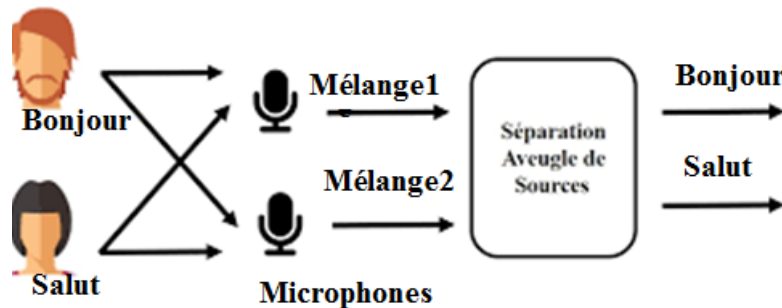


Figure I.1. Principe de la Séparation Aveugle de Sources.

I.3 Historique de la Séparation Aveugle de Sources

Avec le développement rapide de la technologie de traitement des signaux numériques, un grand nombre d'algorithmes SAS efficaces sont continuellement proposés. Le problème de la SAS devient le sujet le plus populaire aujourd'hui dans le domaine du traitement de l'information. Depuis le milieu des années 1990, il y a eu une augmentation du nombre de communications consacrées à la SAS. La première conférence internationale sur l'ACI (Analyse en Composantes Indépendantes, en anglais ICA pour Independent Component analysis) qui est une méthode d'analyse des données issue des statistiques, des réseaux de neurones et du traitement du signal, a eu lieu à Aussois, en France, en janvier 1999, et le deuxième atelier a suivi en juin 2000 à Helsinki, en Finlande avec plus de 100 chercheurs travaillant sur l'ACI et la SAS. Les travaux présentés dans ces nombreuses conférences ont tous contribué à la transformation de la SAS en un domaine de recherche développé.

Dans [I.1], Herault et Jutten présentent un modèle de réseau neuronal de rétroaction pour séparer deux signaux sources indépendants mélangés. En fait, résoudre les problèmes de la SAS est une tâche très difficile parce que nous n'avons pas d'informations sur les signaux sources. Dans l'algorithme proposé dans [I.1], deux hypothèses sont formulées : on suppose que les signaux sources sont statistiquement indépendants et que la distribution statistique du signal source est connue. Si les signaux source ont une distribution gaussienne, il est facile de voir qu'il n'y a pas de solution générale au problème de la SAS, car toute distribution gaussienne à mélange linéaire est toujours gaussienne. Le modèle de réseau Herault et Jutten

[I.2] suppose que les signaux source sont des signaux sous-gaussiens, ce qui signifie que le kurtosis de chaque signal doit être inférieur à zéro (le kurtosis d'un signal gaussien est zéro). Comme les conditions de solvabilité du problème de la SAS n'ont pas été résolues par l'algorithme proposé par Herault et Jutten, Linsker [I.3] a proposé le critère du maximum d'information mutuelle qui convient le mieux à l'établissement d'un modèle auto-organisateur.

Dans [I.4], Giannakis et Swami introduisent le cumulante d'ordre trois basé sur une recherche exhaustive pour résoudre le problème d'identification de la SAS. En 1989, la première conférence internationale sur l'analyse spectrale d'ordre supérieur a été effectuée. Lors de cette conférence, Cardoso et Comon ont présenté dans [I.5] des exposés sur l'ACI. Ces travaux ont fourni un cadre généralement clair pour l'ACI. Si les signaux originaux sont statistiquement indépendants, des signaux sources mutuellement indépendants peuvent être obtenus statistiquement. Depuis lors, la théorie de l'ACI s'est progressivement perfectionnée. Dans [I.6], Herault et Jutten ont publié un article sur le problème de la SAS dans le traitement du signal, introduisant un algorithme de réseau de neurones artificiel pour la SAS. Cette étude a marqué le début d'un nouveau domaine de recherche. Leur algorithme d'apprentissage était heuristique (approximatif mais rapide) et n'indiquait pas clairement la nécessité d'utiliser les statistiques d'ordre supérieur des signaux observés. Au cours des vingt dernières années depuis ces études préliminaires, le problème de la SAS est devenu un point chaud dans le domaine du traitement du signal. Des recherches plus approfondies ont amélioré à la fois la théorie et son application pratique. Dans [I.7] Comon a proposé une méthode populaire de l'ACI basée sur l'information mutuelle minimale. Bell et Sejnowski [I.8] ont proposé l'approche par entropie maximale basée sur le principe Infomax. L'algorithme Infomax a été développé par les Amari et al. [I.9] en utilisant le gradient naturel et ses liens fondamentaux avec une estimation maximum de vraisemblance [I.10]. Quelques années plus tard, les chercheurs Hyvärinen et Oja présentèrent l'algorithme FastICA [I.11], [I.12], [I.13] et ont donc contribué à l'application de l'ACI aux problèmes à grande échelle en raison de son efficacité computationnelle. Jusqu'à présent, les méthodes ACI standard, comme l'algorithme rapide FastICA, l'algorithme Infomax et l'algorithme d'extension maximisation de l'information [I.14], entre autres, ont été plus complets. L'ACI standard utilise un modèle mathématique idéalisé mais les utilisateurs se concentrent maintenant sur les extensions de l'ACI, comme l'ACI bruitée [I.15], les représentations parcimonieuses [I.16] et l'ACI non-linéaire [I.17] [I.18].

Les algorithmes SAS peuvent être divisés en deux catégories : les algorithmes adaptatifs et ceux par clusters ou par lots. D'autre part, du point de vue des hypothèses, les algorithmes SAS

peuvent être divisés en trois groupes : ceux basés sur l'indépendance statistique, ceux basés sur les caractéristiques de parcimonie et ceux basés sur les contraintes non-négatives du signal source. Du point de vue de la fonction de coût, les algorithmes SAS peuvent être catégorisés en méthodes indépendantes fondées sur des statistiques de deuxième ordre, en approches basées sur des statistiques d'ordre supérieur, en méthodes basées sur des réseaux neuronaux et en méthodes basées sur des fonctions non linéaires. La plupart des algorithmes SAS existants sont utilisés pour résoudre des problèmes de mélange instantanés linéaires et ont réussi à obtenir de bonnes performances. Par exemple, l'algorithme ACI a été appliqué au domaine du traitement de signal comme le traitement des signaux biomédicaux (par exemple les signaux de l'électroencéphalographie (EEG) et les signaux de la magnétoencéphalographie (MEG)) ainsi que les systèmes de reconnaissance vocale. En plus, ces dernières années, des approches non linéaires ont été développées pour résoudre le problème de la SAS [I.19]. Jutten et Babaie-Zadeh ont proposé plusieurs algorithmes pour résoudre le problème de la SAS dont le modèle post-non linéaire (PNL) [I.20]. Ces algorithmes ont atteint une valeur fonctionnelle dans le traitement des signaux de réseaux de capteurs, les communications micro-ondes, les communications par satellites et de nombreux systèmes biologiques [I.21]. Ils ont intégré l'apprentissage de réseau bayésien dans la SAS non linéaire et ont obtenu de meilleurs résultats. La méthode bayésienne non linéaire de séparation aveugle est devenue un axe de recherche important ces dernières années. L'algorithme SAS non linéaire dans le domaine nucléaire développé par Taleb, Jutten, et Olympieff [I.22] et la méthode SAS localement linéaire ont également attiré beaucoup d'attention. Cependant, en raison de la complexité inhérente des problèmes non linéaires, il n'y a pas un algorithme communément applicable qui peut être adéquat à toutes sortes de problèmes pratiques. Par conséquent, une série de méthodes de la SAS non linéaires basées sur différents modèles ont été proposées [I.23], [I.24]. Avec le développement des réseaux neuronaux, les réseaux à fonction radiale de base (RBF), les réseaux de perceptrons multicouches (MLP : Multi Layer Perceptron) [I.25], les réseaux neuronaux polynomiaux (PNN) et les algorithmes génétiques (GA) [I.26] se sont révélés d'un grand intérêt pour la SAS en raison de leur souplesse non linéaire. Par conséquent, les réseaux neuronaux peuvent être exploités pour résoudre des problèmes non linéaires. Dans des conditions non linéaires, la méthode de décomposition par réseau RBF a la vitesse de convergence la plus rapide, mais est de faible précision lors de la récupération du signal. Le réseau MLP reproduit la source la plus précise du signal récupéré mais a une grande complexité de calcul. Un réseau PNN muni d'une fonction flexible d'activation de neurones cachés peut empêcher un réseau « sur optimisé » et produit les solutions les plus régulières. La question se pose donc : Comment choisir le réseau neural le plus adapté ? Nous devons estimer la vitesse, la précision et la

complexité en tenant compte des exigences de l'application pratique réelle. Le réseau MLP est plus généralisé que les autres. Il est plus adéquat s'il n'y a pas de connaissances apriori sur les systèmes hybrides. Le réseau PNN peut fournir des résultats de séparation clairs mais nécessite une connaissance a priori du signal source [I.26]. Actuellement, les algorithmes SAS sont divisés en trois groupes principaux : l'ACI, la NMF (Non negative Matrix Factorization) [I.27] et la SCA (Sparse Component Analysis, i.e. Analyse en Composantes Parcimonieuses) [1.28]. L'ACI est une méthode de SAS qui trouve des composants statistiquement indépendants et non gaussiens à partir de données multi-variées. L'évolution de la théorie de l'ACI remonte au début des années 1990 lorsque les chercheurs français [I.29] ont d'abord proposé le concept d'ACI. Jusqu'au milieu des années 1990, la théorie de l'ACI et plusieurs algorithmes ont été développés et la technique s'est rapidement étendue à d'autres domaines de recherche. Paatero et Tapper [I.30] ont d'abord introduit une matrice de factorisation non négative (NMF), parfois appelée PMF (Positive Matrix Factorization, i.e. Factorisation Matricielle Positive). Lorsqu'elle est appliquée aux problèmes de la SAS, elle ne tient pas compte de la parcimonie et de l'indépendance. La MF (matrice de factorisation) est largement utilisée dans le traitement de matrices lorsque les signaux ont une signification physique. La SCA est basée sur une description de parcimonie. Elle possède une grande résistance au bruit et est utilisée pour la séparation de sources lorsqu'il y a moins de capteurs que de sources présentes.

I.4 Les applications la SAS

Comme énoncé dans l'introduction et la description générale de la SAS, cette dernière permet de nombreuses applications dans de nombreux domaines tels que l'acoustique, le traitement audio, les télécommunications, le génie biomédical, l'astrophysique, etc. On peut donc citer les exemples concrets suivants :

I.4.1 Le Signal Parole

L'une des motivations initiales de la recherche de la SAS était le problème de séparation d'un signal parole (Figure I.2). L'idée de base est qu'il y a plusieurs signaux observés, qui sont des mélanges de nombreux signaux source. Le but de la SAS est de séparer ces signaux source. Dans un système de reconnaissance vocale, le signal observé est la convolution des réponses d'impulsion produites par l'interaction du signal de la parole source et du capteur (environnement). Dans les situations réelles, les positions des microphones par rapport aux sources peuvent être arbitraires et donc le processus de mélange n'est pas connu. Dans cette situation, les algorithmes SAS sont importants. En outre, les algorithmes SAS convolutifs peuvent être appliqués de préférence dans le domaine de la musique. Douglas [1.31]

a séparé deux voix d'une musique en utilisant un algorithme SAS convolutif, tandis que Vincent [1.32] a appliqué la technologie ACI pour obtenir les sons de différents instruments dans une bande sonore audio.

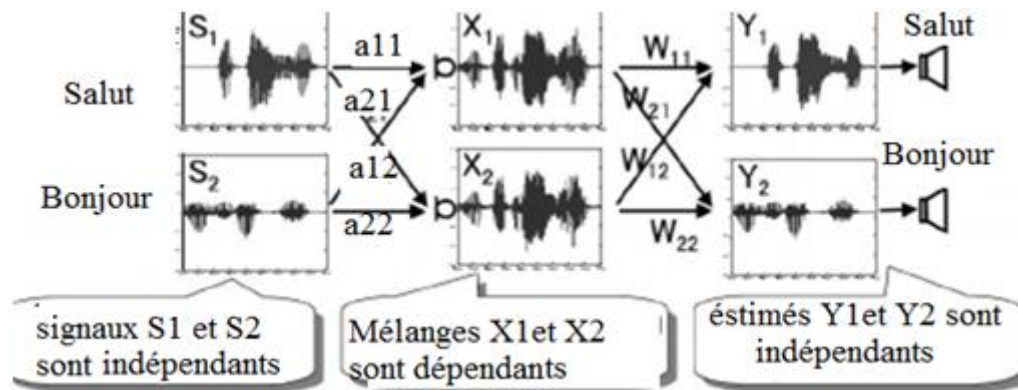


Figure I.2. Séparation aveugle du signal parole.

I.4.2 Les systèmes de communication voix/données et Traitement de signal

La SAS a un large éventail d'applications dans les télécommunications. Zhang Hu et al., dans [I.33] ont proposé un système de réception de réseaux d'antennes et des techniques de filtrage spatial ainsi qu'un algorithme SAS pour réaliser la séparation d'un signal multi-utilisateur dans le même canal, de sorte que plusieurs signaux peuvent être transmis sur le même canal en même temps, améliorant ainsi considérablement la capacité du canal. Dans la technique CDMA (Code Division Multiple Access ou Accès Multiple par Répartition en Code : AMRC) plusieurs utilisateurs partagent la même largeur de bande du canal et transmettent des signaux simultanément. Les utilisateurs peuvent distinguer leurs propres signaux par codes. En raison de l'évanouissement dû à la propagation par trajets multiples, les signaux à la réception ne sont plus orthogonaux et par conséquent, l'utilisation de la méthode ACI peut contribuer à améliorer le rapport signal sur bruit (SNR) du côté récepteur. Au niveau des capteurs, chaque capteur reçoit un signal mélangé des signaux source et comme les caractéristiques du mélange sont complètement inconnues, cela devient un problème de la SAS. De nombreuses applications telles que la détection du sonar en mer, la détection radar passive et même la détection d'aéronefs furtifs ont été explorées à l'aide des techniques SAS pour résoudre le problème [1.34].

I.4.3 Le traitement et la reconnaissance d'images

Actuellement, la SAS est largement utilisée dans le domaine du traitement d'images. On la retrouve dans l'extraction de caractéristiques d'images, la reconnaissance de visages, la

détection d'objets en mouvement, le tatouage d'images numériques, le débruitage d'images, etc. [I.35] [I.36]. Par exemple, les méthodes SAS peuvent aider à récupérer le contenu d'origine d'images de celles contaminées et éliminer les mauvais effets [I.37].

I.4.4 Le traitement de l'information géologique spatiale

Dans le traitement des informations spatiales géo-scientifiques, les techniques SAS ont été appliquées à de nombreux aspects du traitement des données de télédétection, la fusion d'images, la décomposition de pixels et la classification et la reconnaissance d'images [I.38] [I.39]. La SAS peut également être utilisée en exploration sismique [1.40]. En effet l'utilisation des techniques SAS lors du dynamitage par explosion dans le sol peut aider à distinguer différentes caractéristiques lithologiques (la lithologie est la nature des roches formant une couche géologique) des signaux réfléchis ou diffractés reçus [1.40].

I.4.5 Le traitement du signal biomédical

La SAS est devenue un sujet important dans la recherche et le développement dans le domaine des sciences biomédicales. C'est un domaine d'application très encourageant pour les techniques SAS, non seulement parce que c'est un domaine de grande importance, mais aussi parce que certaines des données d'imagerie cérébrale sont bien décrites par le modèle SAS. C'est particulièrement vrai pour les électrocardiogrammes (ECG), les électromyogrammes (EMG), ainsi que des signaux IRM (Imagerie par Résonance Magnétique fonctionnelle), qui sont des enregistrements de champs électriques et magnétiques de signaux provenant de courants neuronaux dans le cerveau [I.41]. Dans le traitement du signal ECG fœtal, le signal ECG mesuré chez une femme enceinte comprend en réalité les signaux de la femme enceinte et du bébé. Comme les paramètres du support de transmission sont inconnus, cela induit que c'est une application typique de la technique de la SAS. Les méthodes SAS peuvent ainsi séparer le signal ECG de la femme enceinte de celui du bébé, fournissant ainsi des données précises sur un diagnostic complémentaire. De plus, l'ACI a été appliquée avec succès au signal IRM dans le traitement du cerveau pour obtenir des informations sur les caractéristiques cachées de celui-ci, fournissant des données utiles pour la recherche scientifique sur le comportement humain [I.42].

Nombreux chercheurs ont appliqué l'algorithme NMF à la génétique et à la découverte de médicaments. Les résultats expérimentaux ont montré que les résultats des séquences de molécules d'ADN analysées par NMF sont plus fiables. L'algorithme NMF aide à sélectionner les ingrédients les plus efficaces dans l'industrie pharmaceutique ayant les effets les moins négatifs [I.43].

I.4.6 L'analyse de documents texte

L'important dans le traitement de texte intelligent est de capturer la sémantique ou les informations représentant le document texte entier. Cependant, les méthodes traditionnelles considèrent seulement les statistiques de mots, en ignorant les autres informations. Avec la vaste application de la technique SAS, les chercheurs ont appliqué l'algorithme NMF à l'analyse et au traitement de texte et les résultats s'avèrent très positifs [I.44].

I.5 Les types de mélanges

Il existe plusieurs types de mélanges divisés en deux classes : les mélanges linéaires et les mélanges non-linéaires.

I.5.1 Les mélanges linéaires

On dit que les mélanges sont linéaires instantanés quand les observations s'écrivent comme des combinaisons linéaires des sources. Dans ce cas, le nombre de sources est supposé égal au nombre d'observations. Supposons qu'il existe N sources statistiquement indépendantes passant par un système de mélange linéaire instantané et reçues par M capteurs. Ensuite, chaque signal observé est une combinaison linéaire de ces sources. Le modèle d'un mélange linéaire est donné sur la figure I.3.

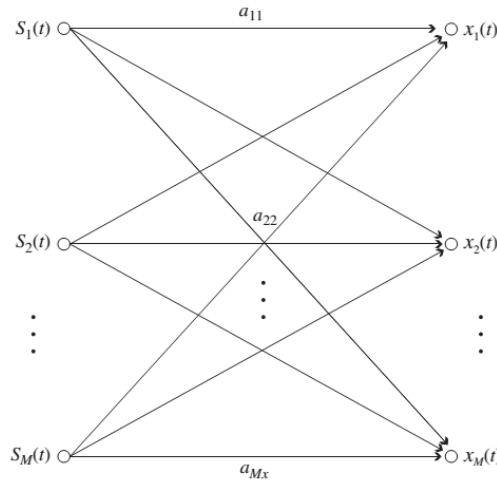


Figure I.3. Modèle du mélange linéaire

L'équation suivante régit la fonction de mélange instantané linéaire invariant dans le temps :

$$x_j(t) = \sum_{i=1}^N a_{ji} s_i(t) \quad (\text{I.1})$$

avec :

$a_{ji} (i \in \{1, 2, \dots, N\}, j \in \{1, 2, \dots, M\})$ sont les paramètres du mélange, $s_i(t) (i \in \{1, 2, \dots, N\})$ sont les signaux de source et $x_i(t) (i \in \{1, 2, \dots, M\})$ sont les signaux observés.

Il est plus facile d'utiliser la forme vectorielle pour exprimer le processus de mélange :

$$s(t) = [s_1(t) \dots \dots s_N(t)]^T, x(t) = [x_1(t) \dots \dots x_M(t)]^T.$$

Donc l'équation (I.1) peut être écrite comme suit :

$$x(t) = A.s(t) \tag{I.2}$$

où $A \in R^{M \times N}$ est la matrice de mélange et $s \in R^{M \times N}$ est le vecteur source dont les éléments sont les coefficients de la matrice A .

Comme le canal de transmission et le réseau de capteurs peuvent inclure du bruit additif qui est à prendre en compte, l'équation (1.2) devient :

$$x(t) = A.s(t) + n(t) \tag{I.3}$$

où $n(t) = [n_1(t) \dots \dots n_M(t)]^T$ est le vecteur de bruits additifs.

Si le nombre d'observations est inférieur au nombre de sources, on dit que le mélange est sous déterminé, et dans ce cas, il est impossible de résoudre le problème sans hypothèses supplémentaires [1.45].

Dans le cas où le nombre d'observations est supérieur aux nombres de sources, on dit que le mélange est sur-déterminé. Dans ce cas, Bermon et al. Utilisent l'ACI pour résoudre le problème de séparation aveugle de sources à condition de supposer qu'il y ait au plus une source gaussienne [I.46].

I.5.1.1 La séparabilité et l'indétermination

La seule information sur les signaux source est leur indépendance. On peut facilement trouver des transformations linéaires qui préservent l'indépendance. Par exemple, permuter les sources ou changer leurs énergies ne change pas leur indépendance. D'autre part, si les sources sont tous à variance unitaire moyenne nulle et la matrice de mélange est une matrice unitaire (transformation de rotation), il est facile de vérifier que leurs estimations seront indépendantes. Cependant, il a été démontré dans [I.47] que ces cas sont les seuls cas pour lequel une transformation linéaire préserve l'indépendance. En d'autres termes, les mélanges instantanés linéaires sont séparables, jusqu'à des indéterminations triviales (une permutation et un changement d'échelle), à condition qu'il existe au plus une source gaussienne.

I.5.1.2 Les critères d'indépendance

Il est bien connu que l'indépendance est bien plus que la dé-corrélation. En fait, la transformation du vecteur d'observation en sorties dé-corrélées, qu'on appelle généralement Analyse de Composantes Principales (PCA : Principal Component Analysis), ne suffit pas pour séparer les sources. La PCA ne représente que "la moitié du travail ICA". Ce fait montre que pour trouver les autres équations requises dans un contexte totalement aveugle, des contraintes d'indépendance d'ordre supérieur doivent être utilisées. Cela montre également pourquoi les sources gaussiennes ne peuvent pas être séparées : elles n'ont pas de statistiques d'ordre supérieur (à deux). Certains algorithmes [I.48, I.49, I.50, I.51, I.52, I.53] ont utilisé des méthodes de statistiques d'ordre supérieur.

I.5.1.3 L'information mutuelle

L'information mutuelle dépend des densités des variables aléatoires, qui doivent être estimées à partir des données. Une méthode pour l'estimation de la fonction de densité de probabilité (PDF) d'une variable aléatoire consiste à faire une expansion à un certain ordre de précision par une série Edgeworth [I.54]. Une autre approche basée sur l'information mutuelle consiste à calculer son gradient par rapport à la matrice de séparation et à estimer ce gradient à partir des données [I.55]. Cette approche caractérise la pertinence de la fonction de score (le log-dérivé de la densité) d'une variable aléatoire.

I.5.1.4 La non-gaussianité

La non-gaussianité pourrait être utilisée pour estimer l'indépendance. Les mesures courantes de la non-gaussianité sont le kurtosis (le cumulants d'ordre quatre) et la néguentropie. Le kurtosis en tant que mesure de l'indépendance statistique est largement utilisé dans l'ACI. Son avantage exceptionnel peut être facilement réalisé. Cependant, le kurtosis est sensible aux valeurs singulières et n'est pas une mesure robuste de la non-gaussianité. Ainsi, divers facteurs tels que la maximisation d'informations mutuelles, la maximisation de néguentropie et le maximum de vraisemblance sont pris en compte. La néguentropie est le meilleur critère pour la non-gaussianité. Cependant, elle doit calculer la PDF qui présente une complexité de calcul élevée et donc un calcul approximatif de la néguentropie est nécessaire. La séparation de sources basée sur la minimisation de l'information mutuelle des sorties possède une autre propriété intéressante qui rend cette approche très attrayante : l'estimation du maximum de vraisemblance (ML) des sources [I.56, I.57]. Par conséquent, de plus en plus les travaux récents sont basés sur ce critère [I.54, I.57, I.58].

I.5.2 Les mélanges convolutifs

Le modèle des mélanges convolutifs des sources est plus proche d'un modèle de mélanges réel.

Supposons qu'il existe des sources statistiquement indépendantes $s_i(t)$, $i = 1, 2, \dots, N$ reçues par M capteurs après le processus de mélange par convolution (figure I.4).

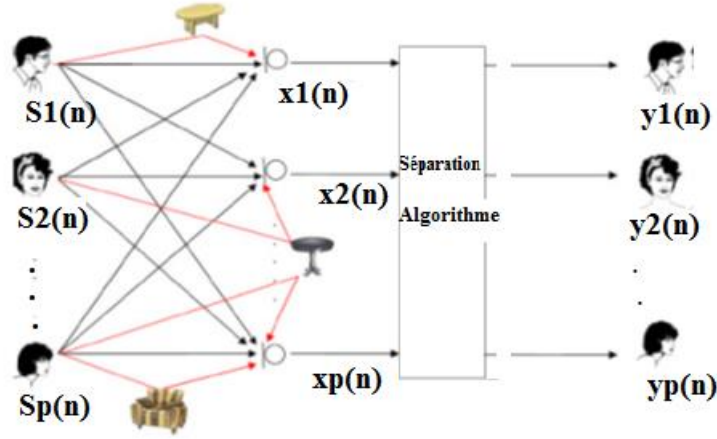


Figure I.4. Modèle des mélanges convolutifs

Les signaux mélangés sont notés par $x_i(t)$, $i = 1, 2, \dots, M$, et le modèle de mélanges par convolution peut être exprimé par :

$$x_j(t) = \sum_{i=1}^N a_{ji}(t) * s_i(t) = \sum_{i=1}^N \sum_{\tau=0}^{L-1} a_{ji}(\tau) s_i(t - \tau) \quad (I.4)$$

où $*$ désigne l'opération de convolution et $a_{ji}(\tau)$ est la réponse impulsionnelle du capteur j sur la source i . Étant donné que chaque canal peut être désigné par un filtre à réponse impulsionnelle finie (FIR) du nième ordre, le système de mélange par convolution peut être représenté par la matrice FIR via l'équation (I.5).

$$x = As \quad (I.5)$$

où A est une matrice FIR de la forme :

$$A = \begin{bmatrix} a_{11}^T & \cdots & a_{1n}^T \\ \vdots & \ddots & \vdots \\ a_{1n}^T & \cdots & a_{nn}^T \end{bmatrix} \quad (I.6)$$

où a_{ji} est une colonne de dimension L qui désigne l'ordre du filtre FIR.

La forme vectorielle de l'équation (I.5) est :

$$x(t) = \sum_{\tau=0}^{L-1} A(\tau)s(t - \tau) \quad (I.7)$$

où $s(t) = [s_1(t) \dots \dots s_N(t)]^T$, $x(t) = [x_1(t) \dots \dots x_M(t)]^T$, et A la matrice de mélange.

Pour $L=1$, le mélange convolutif devient un mélange instantané.

On parle de mélanges convolutifs lorsque les sources s'écrivent comme la somme de sources convoluées par un filtre de propagation, c.à.d. les signaux reçus au niveau des capteurs dépendent à la fois des sources et de leurs versions retardées. Les sources peuvent contribuer au mélange avec plusieurs délais différents. Cela peut se produire dans plusieurs domaines d'applications tels que les télécommunications où les modèles de canal incluent souvent la propagation par trajets multiples puisque la réverbération en traitement audio peut introduire des retards [I.59]. Le modèle de mélange convolutif est plus compliqué que le modèle de mélange instantané linéaire. Cependant, il a été prouvé dans [I.60] que les mélanges convolutifs sont séparables, c'est-à-dire que dans un système de mélange-séparation convolutif l'indépendance des sorties assure la séparation des sources. Mais dans ce cas, l'indétermination n'est pas aussi triviale que dans le cas instantané puisque l'indétermination de la permutation existe toujours. Cette indétermination peut être inacceptable, car elle peut fortement altérer les sources. Cependant, comme l'a proposé Simon [I.61], après la séparation des sources, l'effet de chaque source sur chaque capteur peut être estimé.

I.5.3 Les mélanges non linéaires

En réalité, les signaux observés sont principalement mélangés de manière non linéaire. La résolution du problème de la SAS pour un mélange non linéaire est généralement difficile car elle nécessite des informations préalables supplémentaires ou bien des contraintes appropriées. Le modèle du mélange non linéaire est une extension du modèle linéaire. Il est donné par :

$$x(t) = f(s(t)) + n(t) \quad (I.8)$$

où $x(t)$ représente les vecteurs du signal observé de dimension M , $s(t)$ les vecteurs du signal source inconnus de dimension N , et $n(t)$ des vecteurs de bruit additifs de dimension M qui sont indépendants des signaux de source.

$f: R^N \rightarrow R^M$ est la fonction de mélange non linéaire. Le problème des mélanges non linéaires a été utilisé par C. Jutten [I.62] qui a utilisé des mélanges non linéaires afin d'évaluer la robustesse et les performances de l'algorithme Herault-Jutten. Plus tard, Burel [I.63] a proposé une

solution basée sur un réseau de neurones. Dans son algorithme, il est supposé que la structure de mélange est connue mais dépend de paramètres inconnus. Pajunen et al. [I.64] ont utilisé la méthode dite Kohonen's Self –Organizing Maps (SOM) pour séparer les mélanges non linéaires [I.65]. La SOM est une méthode non supervisée et est basée sur la quantification des données avec un réseau de neurones. Elle peut être utilisée pour balayer les données afin qu'elles soient uniformément réparties sur le quadrillage rectangulaire. Ensuite, ils utilisent cette approche pour trouver une transformation qui transforme les observations en un système uniformément distribué. Une des difficultés de cette approche est la tendance à créer des sorties uniformément réparties. Pour résoudre ce problème, la méthode Générative Topographique Mappings (GTP) a été utilisée comme alternative à la méthode SOM [I.66, I.67]. Dans cette approche, les sources peuvent avoir n'importe quelle distribution, mais celle-ci doit être préalablement connue. Deco et Brauer [I.68] ont également abordé le problème, en considérant une condition de conservation du volume sur les transformations non linéaires. Yang Et. al. [I.69] ont également étudié le problème des mélanges de types particuliers, en partant du principe que les non-linéarités inverses peuvent être approximées avec un perceptron à deux couches. Valpola et al. [I.70, I.71] ont proposé une approche d'apprentissage d'ensemble. Dans leur approche, la transformation est modélisée par un réseau de neurones à plusieurs couches de perceptron (MLP) et la procédure d'apprentissage est basée sur l'apprentissage d'ensemble bayésien non supervisé. Une autre méthode utilisée pour le mélange non linéaire et qui est basée sur l'information mutuelle des sorties a été proposée par Almeida [I.72]. Cette approche peut être vue comme une extension de la méthode INFOMAX de Bell et Sejnowski [I.73]. Dans cette approche, le critère d'indépendance est l'information mutuelle des sorties.

La plupart des travaux ci-dessus, même s'ils sont basés sur une bonne intuition, manquaient de justification théorique. En fait, dans la plupart des cas, l'objectif a été de trouver des sorties en composants indépendants. Dans certains travaux, cela est justifié en forçant la transformation à être modélisée par des réseaux de neurones par exemple, avec l'espoir qu'une transformation efficace ne peut à la fois préserver l'indépendance et mélanger les sources.

I.6 Le Signal Audio

Dans les travaux de recherche qu'on a effectués on a exclusivement travaillé avec le signal audio (parole + musique). Il est donc utile de bien définir ce qu'est un signal audio qu'on appelle communément son. Un son est une énergie qui se propage sous forme de vibrations mécaniques dans un support gazeux, fluide ou solide. Il est propagé grâce à l'élasticité du milieu environnant sous forme d'ondes longitudinales. Par extension physiologique, le son désigne la sensation auditive à laquelle cette vibration est susceptible de donner naissance. Le traitement

du son est la branche du traitement du signal qui s'applique aux signaux audio, dans le but notamment de les numériser, les coder, les compresser, d'améliorer leur qualité ou d'en extraire l'information.

I.6.1 Emission, propagation et réception d'un signal audio

Quand on veut transmettre un signal audio, une énergie doit avant tout mettre en mouvement un corps pour créer une vibration. Ainsi, le muscle du larynx ou la tension électrique dans un haut-parleur, provoqueront l'énergie nécessaire pour produire cette vibration. Ensuite, pour que ce son puisse se propager, il est nécessaire de disposer d'un milieu élastique propice à la transmission des vibrations. En créant des pressions ou des dépressions, l'air permet la propagation de l'onde. Pour être perçue, il doit y avoir un récepteur sensible. Chez l'homme, l'oreille possède une membrane (le tympan) capable de transmettre les informations de vibration en signaux nerveux jusqu'au cerveau, grâce au nerf auditif (figure I.5).



Figure I.5. L'émission, la propagation et la réception du signal Audio

I.6.2 Les caractéristiques d'un signal audio

Le son est un phénomène vibratoire qui peut être analysé comme un signal qui varie dans le temps. Les caractéristiques utiles sont l'amplitude et la fréquence.

I.6.2.1 L'amplitude

Le son est caractérisé par son amplitude, appelée aussi intensité sonore. C'est l'expression de la pression de l'air qui se mesure en décibels (dB). 0 dB correspond au minimum que l'oreille humaine puisse percevoir (seuil d'audibilité) (figure I.6).

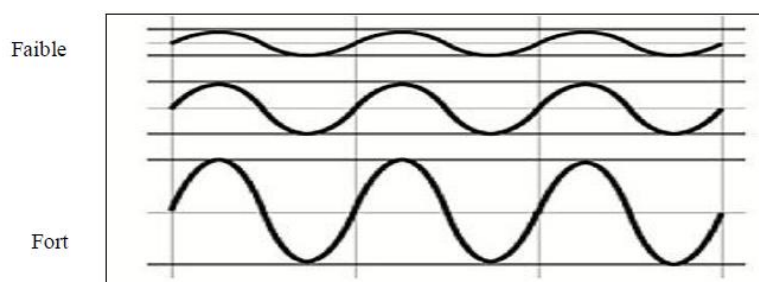


Figure I.6. L'évolution de l'amplitude sonore dans le temps

Ci-dessous sont donnés des exemples concrets de l'amplitude sonore :

- De 0 à 10 dB : Seuil d'audibilité
- De 60 à 70 dB : sonnerie de téléphone.
- 120 dB : seuil de la douleur.

I.6.2.2 La fréquence

Il y a une variété de sons dans notre environnement, allant de faibles sons comme le chant des oiseaux et le bruissement des feuilles à plus fort comme la musique, les cris et le bruit industriel. Chaque son que nous entendons possède certaines fréquences. Lorsque nous disons que quelqu'un a une voix profonde, nous décrivons une voix produisant de basses fréquences. De même, si nous disons que quelqu'un a une voix aigüe, nous décrivons une voix produisant des fréquences plus élevées. Alors, c'est quoi la fréquence d'un son ?

Techniquement la fréquence, est le nombre de formes d'onde générées en une seconde, exprimée en Hertz (Hz). L'oreille humaine perçoit des fréquences entre 20 Hz (pas le plus bas) et 20 kHz (pas le plus haut). Tous les sons de moins de 20 Hz sont qualifiés d'infrasons, mais certains animaux (comme les éléphants) les entendent. De même, tous les sons supérieurs à 20 kHz sont qualifiés d'ultrasons, mais ce sont des sons pour un chat ou un chien (jusqu'à 40 kHz) ou pour un dauphin ou une chauve-souris (jusqu'à 160 kHz).

En musique, la fréquence définit donc la hauteur d'un son, appelée aussi la note. Par exemple, la note « LA » correspond à 440 Hz, soit 440 vibrations en une seconde (figure I.7).



Figure I.7. Les types de fréquences.

I.6.2.3 Le Signal Audio en numérique

Le stockage et la reproduction de la musique audio analogique reposent sur les mêmes principes que l'audition humaine. Dans un système audio analogique, les sons commencent comme des formes d'onde physiques dans l'air, sont transformés en une

représentation électrique de la forme d'onde via un transducteur (par exemple, un microphone) et sont stockés ou transmis. Pour être recréé en son, le processus est inversé, par amplification puis conversion en formes d'ondes physiques via un haut-parleur. Bien que sa nature puisse changer, ses caractéristiques fondamentales semblables à des vagues demeurent inchangées pendant son stockage, sa transformation, sa duplication et son amplification. Tous les signaux audio analogiques sont sensibles au bruit et à la distorsion, en raison du bruit inhérent aux circuits électroniques.

L'audio numérique est une technologie qui utilise les signaux numériques pour la reproduction du son. Il comprend la conversion analogique-numérique, le stockage, la transmission et la conversion numérique-analogique (figure I.8) [I.74].

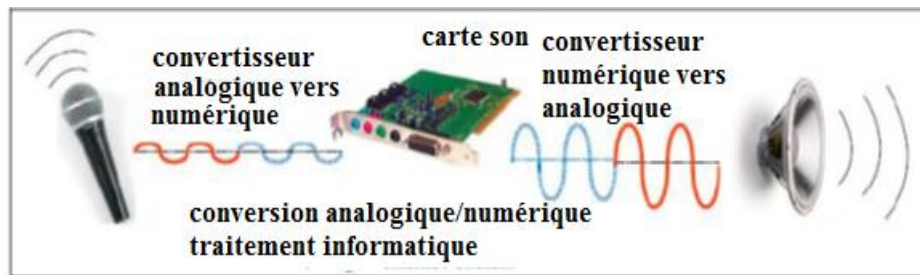


Figure I.8. Exemple d'une chaîne numérique pour le signal Audio

Un signal audio numérique commence par un convertisseur analogique-numérique qui convertit un signal analogique en signal numérique. Cela consiste donc à passer d'un signal continu (une variation de tension électrique) en une suite de valeurs mesurées à intervalles réguliers, donc un signal discontinu [I.75]. L'avantage d'un signal audio numérique est la résistance à la corruption et la possibilité de lire et de reproduire autant de fois ce signal sans aucune dégradation puisqu'il a été réduit en une suite de nombres stockés dans un fichier informatique. Cela dit, la compression audio comme le MP3 peut provoquer une perte volontaire du signal afin d'économiser de l'espace de stockage.

I.6.2.4 La fréquence d'échantillonnage d'un signal audio

La fréquence d'échantillonnage est définie comme le nombre d'échantillons par seconde d'un signal continu pour créer un signal discret ou numérique. Dans le domaine temporel tel que les formes d'onde du son (et d'autres types de contenu audio), les fréquences sont mesurées en hertz (Hz) ou en cycles par seconde. Le théorème d'échantillonnage de Nyquist – Shannon (principe de Nyquist) stipule qu'une reconstruction parfaite d'un signal est possible lorsque la fréquence d'échantillonnage est supérieure à deux fois la fréquence maximale du signal échantillonné. Par exemple, si un signal audio a une limite supérieure de

20 kHz (la limite supérieure approximative de l'audition humaine), une fréquence d'échantillonnage supérieure à 40 kHz évitera le repliement du spectre et permettra une reconstruction théoriquement parfaite.

Les fréquences d'échantillonnage couramment utilisées en audio sont 44100Hz et 48000Hz. Elles sont souvent exigées par des contraintes technologiques. Par exemple, la norme du disque compact audio (CD audio) impose une fréquence d'échantillonnage de 44100Hz. L'échantillonnage est réalisé par découpage temporel du signal audio analogique. Ce découpage temporel permet de reconstruire en données chiffrées la forme d'onde du signal numérisé. La numérisation n'établit que des séries de 0 et de 1 : il s'agit d'un codage binaire.

La recherche montre que les instruments de musique peuvent produire de l'énergie au-dessus de 20 KHz, mais l'énergie sonore au-dessus de 40 KHz est faible. La plupart des microphones ne choisissent pas les sons à plus de 20KHz. L'audition humaine dépasse rarement 20KHz et n'atteint certainement pas 40KHz (figure I.9) [I.76].

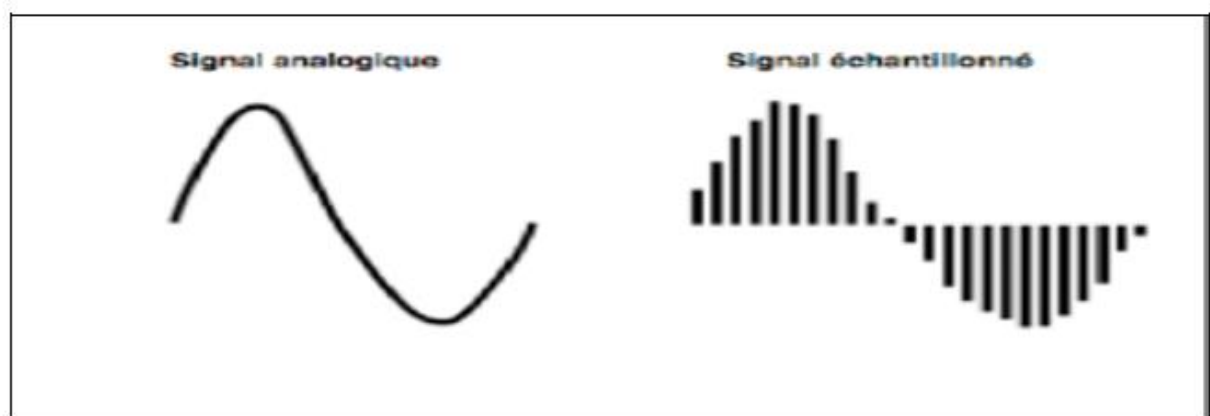


Figure I.9. Echantillonnage d'un signal audio

I.6.2.5 La quantification d'un signal audio

La conversion analogique-numérique est habituellement décomposée en deux processus distincts : l'échantillonnage et la quantification. L'échantillonnage convertit un signal de tension variant dans le temps en un signal à temps discret, c.à.d. une séquence de nombres réels. La quantification remplace chaque nombre réel par une approximation à partir d'un ensemble fini de valeurs discrètes. Le plus souvent, ces valeurs discrètes sont représentées sous forme de mots à virgule fixe. Bien que tous les niveaux de quantification soient possibles, les longueurs de mot courantes sont les suivantes : 8 bits (256 niveaux), 16 bits (65 536 niveaux) et 24 bits (16,8 millions de niveaux). La quantification d'une séquence de nombres produit une séquence d'erreurs de quantification qui est parfois modélisée comme un signal aléatoire additif

appelé bruit de quantification en raison de son comportement stochastique. Plus un quantificateur utilise de niveaux, plus sa puissance de bruit de quantification est faible [I.77]. Lorsqu'un échantillon est quantifié, son amplitude analogique doit être arrondie à la valeur numérique disponible la plus proche. Ce processus d'arrondi s'appelle approximation. Plus le nombre de bits utilisés par échantillon est petit, plus les valeurs analogiques auxquelles il faut arrondir les valeurs sont grandes. La différence entre la valeur analogique et la valeur numérique s'appelle l'erreur d'approximation ou de quantification, comme indiqué sur la figure (I.10). Si la valeur d'amplitude de l'échantillon se situe entre deux niveaux de l'échelle de quantification, elle est approximée au niveau le plus proche. Cette approximation amène une erreur que l'on nomme « erreur de quantification ». Par la suite, plus le nombre de bits est élevé, plus le nombre de niveaux est important et l'erreur de quantification faible. Autrement dit, les petites variations d'amplitude du signal échantillonné sont d'autant mieux approximées que la résolution de la quantification est élevée. La fiabilité de la forme d'onde numérisée à la forme d'onde du signal analogique dépend donc de la résolution (exprimée en bits) et de la fréquence d'échantillonnage (exprimée en kHz). De même que pour la fréquence d'échantillonnage, le choix de la résolution de la quantification est soumis à des contraintes techniques (figure I.10).

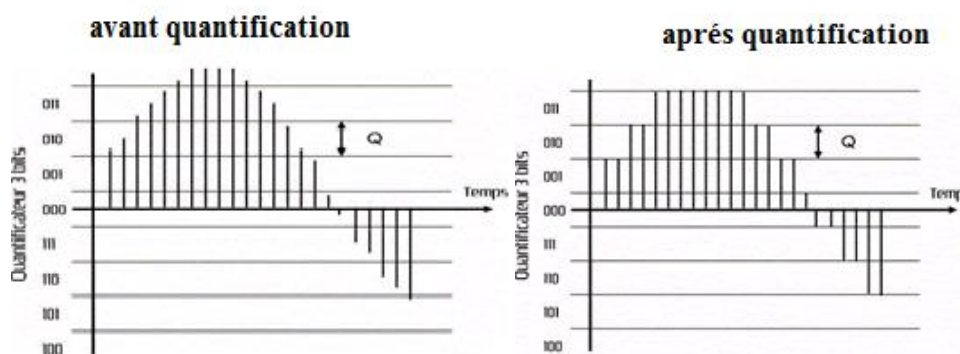


Figure I.10. Signal audio échantillonné avant et après quantification

I.6.2.6 La compression du signal audio numérisé

La compression audio numérique permet un stockage et une transmission efficaces des données audio. Les différentes techniques de compression audio offrent différents niveaux de complexité, qualité audio compressée et quantité de compression de données. Les différents types de techniques de compression audio diffèrent dans les compromis entre la complexité de l'encodeur et du décodeur, la qualité audio compressée, et la quantité de compression de données [I.78]. La compression numérique a pour objectif de produire une représentation d'un signal audio qui, une fois décodé et reproduit, a la même qualité que le signal d'origine, tout en utilisant un minimum d'informations numériques (débit binaire) pour la représentation

comprimée. Les applications audio numériques émergentes dans les réseaux, les ordinateurs sans fil et multimédia font face à de graves lacunes telles que des limitations de bande passante et une capacité de stockage limitée. Ces technologies ont créé une demande de qualité audio pouvant être transférée et stockée à des débits binaires faibles. Cela crée un besoin de compression dont le rôle est de minimiser le nombre de bits nécessaires pour conserver une qualité acceptable du signal source d'origine.

I.6.2.7 Les algorithmes de compression

Les algorithmes de compression sont généralement utilisés pour réduire la taille d'un fichier sans supprimer d'informations et analyser les caractéristiques des données à compresser, afin d'obtenir une représentation compacte. Il existe deux grandes familles d'algorithmes de compression en termes de possibilité de reconstruction de la source originale. Ils sont appelés compression sans pertes et compression avec pertes.

- **La compression sans pertes**

La compression de données sans pertes est la réduction de la taille d'un fichier, de sorte qu'une fonction de décompression puisse restaurer le fichier d'origine exactement sans perte de données.

Les techniques de compression sans pertes permettent une reconstruction exacte du signal d'origine. La compression sans pertes supprime les redondances dans les données et le codage du signal résultant avec un schéma de codage efficace [I.79]. Bien que la compression sans pertes permette une reconstruction parfaite de la forme d'onde d'origine, elle n'est pas généralement préférée comme méthode de compression unique pour l'audio numérique en raison de sa limite en taux de compression. La plupart des signaux audio d'intérêt ont des propriétés temporelles et spectrales variables selon le temps. On s'attend alors à ce que la technique de compression sans pertes puisse s'adapter aux caractéristiques du signal à court terme, ce qui implique que le débit binaire instantané nécessaire pour représenter le signal compressé varie également dans le temps. Dans certaines applications, telles que le stockage d'un fichier de données audio sur un disque dur, le problème majeur est le débit moyen, car la taille du fichier résultant doit être minimisée [I.80]. Ce type de compression est utilisée en Audio : Real Player (Windows, Mac OS X, Linux, Unix, Palm OS, Windows Mobile) et WMA (Windows Media Audio).

- **La compression avec pertes**

La compression avec pertes peut permettre de réduire la taille du fichier d'origine à environ 5 à 20% de la taille du fichier d'origine tout en maintenant une bonne qualité sonore, de sorte que la (parole +musique) ne se distingue pas du fichier d'origine pour l'auditeur moyen. Les principales applications concernent principalement le taux de transfert des données numériques. La télévision numérique, les émissions radio, les DVD et les satellites utilisent tous la compression avec pertes pour accélérer les taux de transfert et rendre la donnée plus accessible aux utilisateurs. L'innovation de la compression audio avec pertes consistait à utiliser la psycho-acoustique parce que toutes les données audio ne peuvent pas être perçues par le système auditif humain. Dans la compression audio avec pertes, les méthodes de psycho-acoustique sont utilisées pour éliminer les composantes non audibles (ou moins audibles) du signal audio, de sorte que la compression de la parole ou le codage de la voix est parfois distingué en tant que discipline distincte de la compression audio. Différentes normes de compression audio et vocale sont classées dans les codecs audio. La compression vocale est utilisée par exemple dans la téléphonie Internet, tandis que la compression audio est utilisée dans les CD et est décodée par des lecteurs audio [I.81]. La compression avec pertes est parfois appelée compression irréversible ou non conservative. Ce type de compression est utilisée en MP3 (MPEG-1 Audio layer 3).

I.7 Les format des fichiers audio

Un format de fichier audio est un format de fichier permettant de stocker des données audio numériques sur un système informatique, La structure de bits des données audio s'appelle le format de codage audio et peut être compressée pour réduire la taille du fichier, souvent à l'aide d'une compression avec pertes. Les formats audio sont généralement divisés en trois catégories : Le format non compressé, le format compressé avec pertes et le format compressé sans pertes. Les formats le plus utilisés sont les formats MP3 et WMA.

I.7 .1 Les caractéristique des formats audio

Les formats audios délimitent la quantité d'informations dans le signal audio afin de réduire le débit (et donc le poids du fichier), en tenant compte de la perception humaine des sons et de la qualité de reproduction qu'ils tolèrent. Ils varient selon le nombre d'échantillons par seconde.

De nombreux formats de fichiers audio permettent un nombre variable de canaux. Ainsi, un fichier pourrait être mono, stéréo ou n'importe quel nombre de canaux discrets selon

l'algorithme de codage, le type de compression utilisé (le cas échéant), et éventuellement des commandes et / ou informations utiles au système d'exploitation pour lequel le format du fichier a été développé. Les fichiers audio peuvent être encodés de différentes manières. La plupart des schémas de codage sont linéaires (comme PCM), alors que certains sont logarithmiques (comme A-Law). Les systèmes de codage varient également dans leur utilisation d'entiers signés ou non signés. En outre, certains formats de fichier (tels que MP3) utilisent un schéma de compression pour réduire considérablement la taille d'un fichier audio [I.82].

I.7.2 Les types de formats

Un format de fichier audio est un format de fichier permettant de stocker des données audio numériques sur un système informatique. La structure de bits des données audio s'appelle le format de codage audio et peut être décompressée ou compressée pour réduire la taille du fichier, souvent à l'aide d'une compression avec pertes. Les données peuvent être binaires dans un format de codage audio, mais elles sont généralement incorporées dans un format dont la spécification décrit la manière des différents éléments de données qui coexistent dans un fichier informatique [I.83]. Il est important de faire la distinction entre un format de fichier et un codec audio. Un codec effectue le codage et le décodage des données audio brutes tandis que les données elles-mêmes sont stockées dans un fichier avec un format de fichier audio spécifique. La plupart des formats de fichiers audio documentés publiquement peuvent être créés avec un ou plusieurs codeurs ou codecs. La plupart des formats de fichiers audio ne prennent en charge qu'un seul type de données audio créées avec un codeur audio.

I.7.2.1 Les formats audio avec compression sans pertes

Il s'agit d'une technique de compression qui décompresse le fichier audio conformément à son origine. Les méthodes sans pertes peuvent fournir un niveau élevé de compression numérique. Il n'y a aucune perte de taille ou de qualité audio. La compression sans pertes est idéale pour les fichiers audio professionnels. Les formats de musique de compression sans pertes incluent les formats FLAC, ALAC et WMA.

Le format FLAC, ou Free Lossless Audio Codec, délivre un son de qualité CD dans un fichier dont la taille est inférieure à celle d'un CD réel. C'est également un codec non propriétaire à code source ouvert, idéal pour les développeurs de logiciels qui souhaitent utiliser de l'audio sans pertes. Bien que les fichiers FLAC soient facilement accessibles, ils peuvent être six fois plus volumineux que les fichiers MP3. Leur téléchargement est donc plus long.

Le format ALAC, ou Apple Lossless Audio Codec, a un son identique à celui de l'enregistrement audio original, mais il est compressé à une taille plus petite sans supprimer les

bits. ALAC fonctionne principalement avec les produits Apple. Il prend également plus de temps pour télécharger.

Le format WMA, ou Windows Media Audio, offre un choix aux auditeurs. Généralement, les fichiers WMA sont plus petits que leurs équivalents non compressés et ont des fonctionnalités similaires à celles des fichiers MP3 et FLAC.

I.7 .2.2 Les formats audio avec compression avec pertes

C'est une forme de compression qui perd des données pendant le processus de compression, mais la différence de qualité n'est pas perceptible à entendre. Les formats de compression avec pertes incluent les formats MP3 et AAC. Le format MP3 correspond à MPEG-1 Audio Layer 3. Il a été publié en 1993 et est devenu populaire. C'est le format audio le plus populaire pour les fichiers de musique. Le but principal du MP3 est d'éliminer tous les sons insensibles ou moins perceptibles par les oreilles de l'homme. Par conséquent, la taille du fichier de musique est réduite. Le MP3 est un format universel compatible avec presque tous les appareils.

Le format AAC correspond à Advanced Audio Coding (codage audio avancé). Il a été développé en 1997 après le format MP3. L'algorithme de compression utilisé par AAC est beaucoup plus complexe et avancé que le format MP3. Par conséquent, comparé à un fichier MP3 de même débit, le fichier AAC aura généralement une meilleure qualité sonore. Il s'agit de la méthode de compression audio standard utilisée par YouTube, et PlayStation.

I. 7.2.3 Les formats audio non compressés

Le format PCM (Pulse Code Modulation) représente la modulation par code d'impulsion. Il représente les signaux audio analogiques bruts sous forme numérique. Pour convertir un signal analogique en un signal numérique, il doit être enregistré à un intervalle particulier. Par conséquent, il possède un taux d'échantillonnage et un débit (bits utilisés pour représenter chaque échantillon). C'est une représentation exacte du son analogique et n'implique pas de compression. C'est le format audio le plus courant utilisé sur les CD et DVD.

Le format WAV, pour Waveform Audio File Format, a été développé par Microsoft et IBM en 1991. Il s'agit simplement d'un conteneur Windows pour les formats audio. Cela signifie qu'un fichier WAV peut contenir de l'audio compressé. La plupart des fichiers WAV contiennent de l'audio non compressé au format PCM. C'est juste un emballage. Il est compatible avec Windows et Mac.

Dans notre thèse, nous sommes intéressés aux fichiers audio non compressés représentés dans le format WAV.

I.8 Conclusion

La notion de séparation aveugle de source a été adoptée depuis les années 1980. Son développement n'a pas arrêté de grandir en raison du fait qu'elle concerne plusieurs disciplines dans le domaine du traitement de l'information. Dans ce chapitre, nous avons passé en revue les méthodes typiques de séparation des sources aveugles pour les mélanges linéaires (instantanés et convolutifs) et non linéaires instantanés. Nous avons proposé une formulation et les éléments théoriques de la séparation aveugle de sources. Nous avons aussi introduit les notions des signaux audio. Dans les chapitres suivants nous présenterons notre travail de recherche qui comprend les algorithmes développés dans le but de faire la séparation aveugle de sources audio (parole et musique).

Bibliographie

- [1.1] Herault, J., and Jutten, J. Space or time adaptive signal processing by neural network models, in AIP Conference Proceedings 151 on Neural Networks for Computing, American Institute of Physics Inc., 1986, pp. 206–211.
- [1.2] COHEN, Marc H. et ANDREOU, Andreas G. Current-mode subthreshold MOS implementation of the Herault-Jutten autoadaptive network. IEEE Journal of Solid-State Circuits, 1992, vol. 27, no 5, p. 714-727.
- [1.3] LINSKER, Ralph. An application of the principle of maximum information preservation to linear systems. In: Advances in neural information processing systems. 1989. p. 186-194.
- [1.4] Giannakis, B., and Swami, A. New results on state-space and input-output identification of on-Gaussian processing using cumulants. Proceedings of the SPIE, 1987, 826, 199–205.
- [1.5] CARDOSO, J.-F. Source separation using higher order moments. In: Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on. IEEE, 1989. p. 2109-2112.
- [1.6] JUTTEN, Christian et HERAULT, Jeanny. Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. Signal processing, 1991, vol. 24, no 1, p. 1-10.
- [1.7] COMON, Pierre. Independent component analysis, a new concept? Signal processing, 1994, vol. 36, no 3, p. 287-314.
- [1.8] BELL, Anthony J. et SEJNOWSKI, Terrence J. An information-maximization approach to blind separation and blind deconvolution. Neural computation, 1995, vol. 7, no 6, p. 1129-1159.
- [1.9] AMARI, Shun-ichi, CICHOCKI, Andrzej, et YANG, Howard Hua. A new learning algorithm for blind signal separation. In: Advances in neural information processing systems. 1996. p. 757-763.
- [1.10] CHO, Hyun-Cheol et LEE, Kwon-Soon. Blind Image Separation with Neural Learning Based on Information Theory and Higher-order Statistics. The Transactions of the Korean Institute of Electrical Engineers, 2008, vol. 57, no 8, p. 1454-1463.
- [1.11] Hyvärinen, A. and Oja, E. A fast fixed-point algorithm for independent component analysis. Neural Computation, 9(7), 1997, 1483–1492.
- [1.12] Hyvärinen, A. Fast and robust fixed-point algorithm for independent component analysis. IEEE Transactions on Neural Networks, 10(3), 1999, 626–634.
- [1.13] HYVÄRINEN, Aapo et OJA, Erkki. Independent component analysis: algorithms and applications. Neural networks, 2000, vol. 13, no 4-5, p. 411-430.
- [1.14] LEE, Te-Won, GIROLAMI, Mark, et SEJNOWSKI, Terrence J. Independent component analysis using an extended Infomax algorithm for mixed sub-Gaussian and super Gaussian sources. Neural computation, 1999, vol. 11, no 2, p. 417-441.
- [1.15] ALBERA, Laurent, FERRÉOL, Anne, CHEVALIER, Pascal, et al. ICAR, un algorithme d'ICA à convergence rapide, robuste au bruit. In : 19^e Colloque sur le traitement du

signal et des images, FRA. GRETSI, Groupe d'Etudes du Traitement du Signal et des Images, 2003.

[1.16] Zhong, M., Tang, H., Chen, H. et al. (2004a) An EM algorithm for learning sparse and over complete representations. *Neural Computing*, 57, 469–476.

[1.17] Hyvärinen, A. and Pajunen, P. Nonlinear independent component analysis: existence and uniqueness. *Neural Networks*, 12(3), 1999, 429–439.

[1.18] TALEB, Anisse et JUTTEN, Christian. Nonlinear source separation: The post-nonlinear mixtures, *In: Proceedings of the ESANN'97*. 1997.

[1.19] BABAIE-ZADEH, Massoud, JUTTEN, Christian, et NAYEBI, Kambiz. Differential of the mutual information. *IEEE Signal Processing Letters*, 2004, vol. 11, no 1, p. 48-51.

[1.20] VALPOLA, Harri et KARHUNEN, Juha. An unsupervised ensemble learning method for nonlinear dynamic state-space models. *Neural computation*, 2002, vol. 14, no 11, p. 2647-2692.

[1.21] MARTINEZ, Dominique et BRAY, Alistair. Nonlinear blind source separation using kernels. *IEEE Transactions on Neural networks*, 2003, vol. 14, no 1, p. 228-235.

[1.22] TALEB, Anisse, JUTTEN, Christian, et OLYMPIEFF, Serge. Source separation in post nonlinear mixtures: An entropy-based algorithm. In: *Acoustics, Speech and Signal Processing*, 1998. Proceedings of the 1998 IEEE International Conference on. IEEE, 1998. p. 2089-2092.

[1.23] WOO, Wai Lok et SALI, Sadettin. General multilayer perceptron demixer scheme for nonlinear blind signal separation. *IEE Proceedings-Vision, Image and Signal Processing*, 2002, vol. 149, no 5, p. 253-262.

[1.24] WOO, W. L. et KHOR, L. C. Blind restoration of nonlinearly mixed signals using multilayer polynomial neural network. *IEE Proceedings-Vision, Image and Signal Processing*, 2004, vol. 151, no 1, p. 51-61.

[1.25] TAN, Ying, WANG, Jun, et ZURADA, Jacek M. Nonlinear blind source separation using a radial basis function network. *IEEE transactions on neural networks*, 2001, vol. 12, no 1, p. 124-134.

[1.26] Rojas, F., Rojas, I., Clemente, R. *et al.* Nonlinear blind source separation using genetic algorithm. *Proceedings of Independent Component Analysis and Signal Separation*, San Diego, CA, 2001, pp. 400–405.

[1.27] PAATERO, Pentti et TAPPER, Unto. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 1994, vol. 5, no 2, p. 111-126.

[1.28] ZIBULEVSKY, Michael et PEARLMUTTER, Barak A. Blind source separation by sparse decomposition in a signal dictionary. *Neural computation*, 2001, vol. 13, no 4, p. 863-882.

[1.29] JUTTEN, Christian et HERAULT, Jeanny. Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal processing*, 1991, vol. 24, no 1, p. 1-10.

- [1.30] Paatero, P. and Tapper, U. Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, **5**(2), 1994,111–126.
- [1.31] Douglas, S.C. *Blind Signal Separation and Blind Deconvolution*, CRC Press, 2002,New York.
- [1.32] Vincent, E. Musical source separation using time-frequency source priors. *IEEE Transactions on Speech and Audio Processing*, **14**(1), 2005, 91–98.
- [1.33] Zhang, X., and Hu, B.An application of blind signal separation in wireless digital communication. *Journal of China Institute of Communications*, **21**(2), 2000,73–77 .
- [1.34] Zhang, A.Q., Qiu, T.S., and Zhang, X.H. Blind separation technique for multipath under water acoustic signals based on extended joint diagonalization method. *Systems Engineering and Electronics*, **25**(9),2003, 1058–1083.
- [1.35] Tukr, M., and Pentland, A. Face recognition using eigen faces. *Proceedings of IEEE Conferences on Computer Vision and Pattern Recognition*, Maui, HI, 1991, pp. 586–591.
- [1.36] Bartlett, M.S., Movellan, J.R., and Sejnowski, T.J.Face recognition by independent component analysis. *IEEE Transactions on Neural Networks*, **13**(6), 2002,1450–1464.
- [1.37] Ma, J.C., Niu, Y.L., and Chen, H.Y. *Blind Signal Processing*, National Defense Industry Press,2006, Beijing .
- [1.38] Si, H.L., and Zhang, L.M. *Intelligent Visual Image Processing*, Shanghai Science and Technology Education Press, 2002, Shanghai .
- [1.39] Yu, X.C., Ren, J.M., and Zhang, N. An optimal ICA algorithm applied to fMRI data. *Proceedings of 2007 International Conference on Machine Learning and Cybernetics*, 2007, pp. 3565–3570.
- [1.40] Liu, X.W., Liu, H., and Zheng, T.Y. Separation of converted-wave from seismic multiples by independent component analysis. *Journal of Seismology*, **23**(1),2003, 11–19
- [1.41] Cichocki, A. and Amari, S.Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications, John Wiley & Sons, Ltd, 2002, Chichester.
- [1.42] Long, Z., Yao, L., Zhao, X. *et al.* (2003) Spatial independent component analysis of multitask-related activation in fMRI data. *International Conference on Artificial Neural Networks*, 2003, pp. 515–522
- [1.43] Gao, Y., and Church, G. Improving molecular cancer class discovery through sparse non-negative matrix factorization. *Bioinformatics*, **21**(21),2005, 3970–3975.
- [1.44] Wei, X., Xin, L., and Yihong, G. (2003) Document clustering based on non-negative matrix factorization. *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Toronto, 2003,Canada, pp. 267–273.
- [1.45] BERMOND, Olivier et CARDOSO, Jean-François. Méthodes de séparation de sources dans le cas sous-déterminé. In : 17^e Colloque sur le traitement du signal et des images, FRA, 1999. GRETSI, Groupe d'Etudes du Traitement du Signal et des Images, 1999.

- [1.46] ZHANG, L.-Q., CICHOCKI, Andrzej, et AMARI, Shun-ichi. Natural gradient algorithm for blind separation of overdetermined mixture with additive noise. *IEEE Signal Processing Letters*, 1999, vol. 6, no 11, p. 293-295.
- [1.47] COMON, Pierre. Independent component analysis, a new concept? *Signal processing*, 1994, vol. 36, no 3, p. 287-314.
- [1.48] C. Jutten, L. Nguyen Thi, E. Dijkstra, E. Vittoz, and J. Caelen, "Blind separation of sources: an algorithm for separation of convolutive mixtures," in *International Signal Processing Workshop on Higher Order Statistics*, Chamrousse, France, July 1991, pp. 273–276.
- [1.49] J.-L. Lacoume and P. Ruiz, "Sources identification: a solution based on cumulants," in *IEEE ASSP Workshop*, Mineapolis, USA, August 1988.
- [1.50] J.-F. Cardoso and A. Souloumiac, "Blind beamforming for non-Gaussian signals," *IEE Proceedings-F*, vol. 140, pp. 362–370, December 1993.
- [1.51] E. Moreau, Apprentissage et adaptative, séparation auto-adaptative de sources indépendantes, Ph. D. thesis, Université de Paris-Sud, 1995.
- [1.52] A. Mansour and C. Jutten, "A direct solution for blind separation of sources," *IEEE Trans. on Signal Processing*, vol. 44, pp. 746–748, 1996.
- [1.53] J.-F. Cardoso, "High order contrasts for independent component analysis," *Neural Computation*, vol. 11, pp. 157–192, 1999.
- [1.54] M. Kendall and A. Stuart, *The Advanced Theory of Statistics, Distribution Theory* vol. 1, Griffin, 1977.
- [1.55] A. Taleb and C. Jutten, "Entropy optimization, application to blind source separation, in *ICANN*, Lausanne, Switzerland, October 1997, pp. 529–534.
- [1.56] J.-F. Cardoso, "Blind signal separation: statistical principles," *Proceedings IEEE*, vol. 9, pp. 2009–2025, 1998.
- [1.57] A. Taleb, Séparation de sources dans des mélanges post non-linéaires, Thèse de l'INP de Grenoble, 1999.
- [1.58] D. T. Pham, "Mutual information approach to blind separation of stationary sources," in *Proceedings of ICA'99*, Aussois, France, January 1999, pp. 215–220.
- [1.59] A. Gorokhov and Ph. Loubaton, "Second-order blind identification of convolutive mixtures with temporally correlated sources: a subspace based approach," in *Proc. European Signal Processing Conf. EUSIPCO 96*, Trieste, Italia, September 1996, pp. 2093–2096.
- [1.60] D. Yellin and E. Weinstein, "Criteria for multichannel signal separation," *IEEE Trans. Signal Processing*, pp. 2158–2168, August 1994.
- [1.61] C. Simon, Séparation aveugle des sources en mélange convolutif, Ph. D.thesis, l'université de Marnela Vallee, Novembre 1999.
- [1.62] C. Jutten, Calcul neuromimetique et traitement du signal : analyse en composantes indépendantes, Thèse d'état des sciences physiques, UJF-INP Grenoble, 1987.
- [1.63] G. Burel, "Blind separation of sources: a nonlinear neural algorithm," *Neural Networks*, vol. 5, no. 6, pp. 937–947, 1992.

- [1.64] K. Matsuoka, M. Ohya, and M. Kawamoto, "A neural net for blind separation of nonstationary signals," *Neural Networks*, vol. 8, no. 3, pp. 411–419, 1995.
- [1.65] S. Haykin, *Neural Networks - A Comprehensive Foundation*, Prentice Hall, 1998, Second edition.
- [1.66] P. Pajunen and J. Karhunen, "A maximum likelihood approach to nonlinear blind source separation," in *ICANN97*, Lausanne (Switzerland), October 1997, pp. 541–546.
- [1.67] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley & Sons, 2001.
- [1.68] G. Deco and W. Brauer, "Nonlinear higher-order statistical decorrelation by volume-conserving architectures," *Neural Networks*, vol. 8, pp. 525–535, 1995.
- [1.69] H.-H. Yang, S.I. Amari, and A. Cichocki, "Information-theoretic approach to blind separation of sources in non-linear mixtures," *Signal Processing*, pp. 291–300, February 1998.
- [1.70] H. Valpola, "Nonlinear independent component analysis using ensemble learning: Theory," in *ICA2000*, Helsinki, Finland, 2000, pp. 251–256.
- [1.71] H. Valpola, X. Giannakopoulos, A. Honkela, and J. Karhunen, "Nonlinear independent component analysis using ensemble learning: Experiments and discussion," in *ICA2000*, Helsinki, Finland, 2000, pp. 351–356.
- [1.72] L. B. Almeida, "ICA of linear and nonlinear mixtures based on mutual information, in *International Joint Conference on Neural Networks*, Washington, DC, USA, July 2001.
- [1.73] T. Bell and T. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1004–1034, 1995.
- [1.74] PROAKIS, John G. *Digital signal processing: principles algorithms and applications*. Pearson Education India, 2001.
- [1.75] Borwick, John. 1996. *Sound Recording Practice*, 4th edition. Oxford: Oxford University Press. ISBN 0198166087.
- [1.76] LAVRY, Dan. *Sampling Theory for Digital Audio*. Lavry Engineering, Inc. Available online: http://www.Lavryengineering.com/documents/Sampling_Theory.pdf (checked 24.5.2010), 2004.
- [1.77] LIPSHITZ, Stanley P., WANNAMAKER, Robert A., et VANDERKOOY, John. Quantization and dither: A theoretical survey. *Journal of the audio engineering society*, 1992, vol. 40, no 5, p. 355-375.
- [1.78] PAN, Davis Yen. Digital audio compression. *Digital Technical Journal*, 1993, vol. 5, no 2, p. 28-40.
- [1.79] Mat Hans, Ronald W. Shafer, "Lossless Compression of Digital Audio", *IEEE Signal Processing Magazine*, July 2001.
- [1.80] MAHER, ROBERT C. *Lossless Compression of Audio Data*. Lossless Compression Handbook, 2003.
- [1.81] MAHMUD, Salauddin. An improved data compression method for general data. *International Journal of Scientific & Engineering Research*, 2012, vol. 3, no 3, p. 2.

[1.82] BIJSTERVELD, Karin et VAN DIJCK, José. Sound souvenirs: audio technologies, memory and cultural practices. Amsterdam University Press, 2009.

[1.83] HO, Anthony TS et LI, Shujun (ed.). Handbook of digital forensics of multimedia data and devices. John Wiley & Sons, 2001.

CHAPITRE II

Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

II.1 Introduction

Depuis environ deux décennies, de nombreuses méthodes de séparation aveugle de sources ont été développées pour séparer des mélanges surdéterminés ($P \leq N$) de P sources statistiquement indépendantes reçues par un réseau de N capteurs. Parmi ces méthodes, on peut citer la méthode SOBI (Second Order Blind Identification : Identification Aveugle de second ordre) [II.1] qui exploite uniquement les statistiques du second ordre. D'autres méthodes telles que les méthodes JADE (Joint Approximate Diagonalization of Eigen-matrices) [II.2] et FastICA [II.3] (Fast Independent Component Analysis) exploitent à la fois les statistiques du second ordre et les statistiques d'ordre quatre. Très récemment, de nouvelles méthodes SAS exploitant les statistiques d'ordre quatre uniquement, telles que la méthode ICA [II.4] ont été exploitées. Dans ce qui suit, nous présentons deux méthodes SAS que nous utilisons pour la séparation de signaux audio. Ces deux méthodes sont appelées Infomax et Robust ICA. Nous les décrivons en détails dans ce qui suit.

II.2 Méthode Informations Maximisation (Infomax)

En 1995, A.J. Bell et T.J. Sejnowski ont développé une approche de maximisation des informations pour la séparation et la déconvolution aveugle de sources, qui est maintenant appelée Infomax [II.5]. Infomax est une méthode de recherche de signaux mutuellement indépendants par la maximisation du flux d'information ou de l'entropie. Elle est basée sur la théorie de l'information qui vise à trouver des signaux sources indépendants en maximisant l'entropie. L'analyse en composantes indépendantes permet d'extraire M signaux y à partir de M signaux mixtures x en optimisant une matrice de séparation W . La relation liant y à x est donnée par la relation suivante :

$$y = Wx \quad (\text{II.1})$$

Sous forme matricielle la relation s'écrit :

$$\begin{bmatrix} y_1^1 & y_2^1 & \dots & \dots & y_1^N \\ y_2^1 & y_2^2 & \dots & \dots & y_2^N \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ y_M^1 & y_M^2 & y_M^3 & \dots & y_M^N \end{bmatrix} = \begin{bmatrix} W_{11} & W_{12} & \dots & W_{1M} \\ W_{21} & W_{22} & \dots & W_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ W_{M1} & W_{M2} & \dots & W_{MM} \end{bmatrix} \begin{bmatrix} x_1^1 & x_2^1 & \dots & \dots & x_1^N \\ x_2^1 & x_2^2 & \dots & \dots & x_2^N \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_M^1 & x_M^2 & x_M^3 & \dots & x_M^N \end{bmatrix} \quad (\text{II.2})$$

où les indices de x et y indiquent le numéro du signal et les indices supérieurs sont les indices de temps.

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

Dans Infomax les signaux extraits sont des signaux de source s'ils sont mutuellement indépendants. Cependant, l'indépendance des signaux ne peut pas être mesurée mais l'entropie le peut. L'entropie est liée à l'indépendance dans la mesure où l'entropie maximale implique des signaux indépendants. Par conséquent l'ICA a pour objectif de trouver la matrice de séparation qui maximise l'entropie dans les signaux extraits y . L'entropie des mélanges de signaux x est constante, mais le changement d'entropie peut être maximisé en transformant les signaux estimés y à un autre ensemble de signaux $Y = g(y) = g(Wx)$. Cette transformation indique également que le changement d'entropie de $x \rightarrow Y$ peut être maximisé en optimisant la matrice de séparation W , et lorsque l'entropie est maximisée, les signaux résultants sont indépendants. On prend ensuite l'inverse $y = g^{-1}(Y)$ qui entraîne l'extraction des signaux estimés y qui sont aussi indépendants, et donc ces signaux doivent être identiques aux signaux originaux sources. La stratégie Infomax est représentée sur la figure II .1.

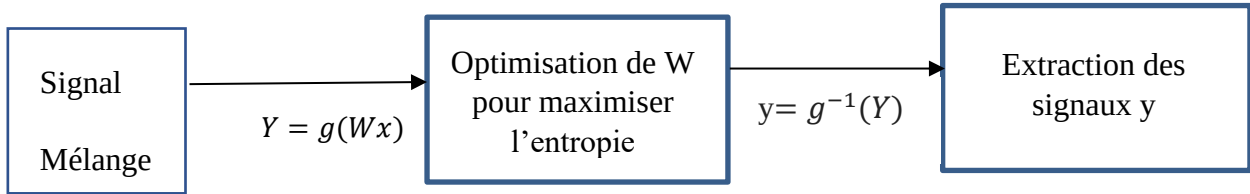


Figure II. 1. Stratégie de la méthode Informations Maximisation.

II.2.1 Entropie d'une fonction de densité de probabilité univariée

Puisque la méthode Infomax obtient des signaux mutuellement indépendants en maximisant l'entropie et que l'entropie des mélanges de signaux x est constante, une expression pour l'entropie d'un signal transformé Y est nécessaire pour maximiser le changement d'entropie. Une expression simplifiée de l'entropie peut être obtenue en considérant le cas à une variable où le signal contient une seule variable dépendante. Dans ce cas, x est un vecteur aléatoire et chaque élément de x est un signal différent échantillonné au même instant t .

$$H(Y) = -\frac{1}{N} \sum_t^N \ln P_Y(Y^t) \quad (\text{II.3})$$

où $Y = g(y)$, et y est une fonction scalaire du temps ($y = y(t) = y^t$) qui indique la valeur scalaire de y à l'instant t , et $g(y)$ est la fonction de distribution cumulative (cdf) du signal souhaité y et fait souvent référence au « modèle cdf » des signaux source qui est choisi pour extraire un type de signal source souhaité. Le cas univarié est exploité en modifiant l'équation (II.1) en $y = W^T x$, où W^T représente une ligne de la matrice de séparation W , et x est un

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

vecteur représentant M signaux dans le temps comme indiqué dans la notation matricielle suivante :

$$y = W^T x \quad (II.4)$$

où

$$x = \begin{bmatrix} x_1^t \\ x_2^t \\ \vdots \\ x_M^t \end{bmatrix} \quad \text{et} \quad W = \begin{bmatrix} W_{11} & W_{12} & \dots & W_{1M} \\ W_{21} & W_{22} & \dots & W_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ W_{M1} & W_{M2} & \dots & W_{MM} \end{bmatrix} \quad (II.5)$$

$\swarrow W^T$

Donc

$$y^t = W^T x = [W_{21} \ W_{22} \ \dots \ W_{2M}] \begin{bmatrix} x_1^t \\ x_2^t \\ \vdots \\ x_M^t \end{bmatrix} \quad (II.6)$$

$$y^t = W_{21}x_1^t + W_{22}x_2^t + \dots + W_{2M}x_M^t \quad (II.7)$$

où y^t est une valeur scalaire du signal échantillonné à l'instant t .

La transformation de y^t dans le modèle cdf $g(y^t)$ donne la valeur Y , où Y est une variable aléatoire comprise entre zéro et un, c'est à dire

$$Y = g(y^t) = g(W_{21}x_1^t + W_{22}x_2^t + \dots + W_{2M}x_M^t) \quad (II.8)$$

A partir de l'équation (II.3), $p_Y(Y^t)$ est le pdf du signal $Y = g(y)$ et est lié au pdf du signal extrait y , $p_y(y^t)$, comme le montre la figure II.2.

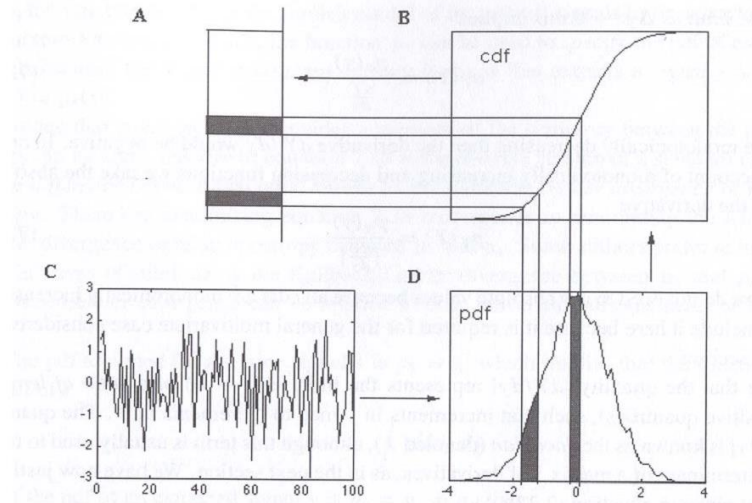


Figure II.2. Transformation de y en Y [II.6].

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

De la figure II.2 on a :

$$p_Y(Y^t)\Delta Y = p_y(y^t)\Delta y \quad (\text{II.9})$$

En réarrangeant l'équation (II.9), on obtient

$$p_Y(Y^t) = p_y(y^t) \frac{\Delta y}{\Delta Y} = \frac{p_y(y^t)}{\frac{\Delta Y}{\Delta y}} \quad (\text{II.10})$$

$$\text{Puisque } \frac{\Delta Y}{\Delta y} \rightarrow \frac{dY}{dy} \text{ comme } \Delta y \rightarrow 0 \quad (\text{II.11})$$

L'équation (II.10) devient

$$p_Y(Y^t) = \frac{p_y(y^t)}{\frac{dY}{dy}} \quad (\text{II.12})$$

Le module du dénominateur de l'équation (II.12) est pris en compte pour les fonctions monotones croissantes et décroissantes, entraînant :

$$p_Y(Y^t) = \frac{p_y(y^t)}{\left| \frac{dY}{dy} \right|} \quad (\text{II.13})$$

Puisque $Y = g(y)$ où $g(y)$ est le modèle cdf du signal source, $\frac{dY}{dy} = g'(y)$ et $g'(y)$ est le pdf du signal source $p_s(y)$. En substituant ce résultat à l'équation (II.13), on obtient

$$p_Y(Y^t) = \frac{p_y(y^t)}{p_s(y)} \quad (\text{II.14})$$

En remplaçant $p_Y(Y^t)$ trouvé dans l'équation (II.14) dans l'équation (II.3), nous obtenons une expression univariée pour l'entropie en termes de pdf de la source et des signaux extraits :

$$H(Y) = -\frac{1}{N} \sum_t \ln \frac{p_y(y^t)}{p_s(y^t)} \quad (\text{II.15})$$

Pour résoudre l'équation (II.15), une expression pour la pdf du signal extrait $p_y(y)$ est nécessaire.

II.2.2 Entropie d'une fonction de densité de probabilité multivariée

Le modèle univarié peut être étendu à un cas général dans lequel il y a plus d'une variable aléatoire. Pour le cas multi-variable l'équation (II.13) devient :

$$p_Y(Y^t) = \frac{p_y(y^t)}{\left| \frac{\partial Y}{\partial y} \right|} \quad (\text{II.16})$$

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

Le dénominateur de l'équation (II.16) est le jacobien, qui est examiné de manière plus détaillée dans la section suivante.

On a vu dans le cas univarié que $Y = g(y)$, où $g(y)$ est le modèle cdf des signaux source.

$\frac{\partial Y}{\partial y}$ est le pdf des signaux source, qui peut également être exprimé par $p_s(y)$.

L'équation (16) peut être alors écrite sous la forme suivante :

$$p_Y(Y^t) = \frac{p_y(y^t)}{p_s(y)} \quad (\text{II.17})$$

La substitution de l'équation (II.16) à l'équation (II.17) donne une expression à plusieurs variables pour l'entropie en termes de signal source pdf $p_s(y)$ et de signal extrait pdf $p_y(y^t)$.

$$H(Y) = -\frac{1}{N} \sum_t \ln \frac{p_y(y^t)}{p_s(y)} \quad (\text{II.18})$$

La pdf de $p_y(y)$ du signal reconstruit $y = Wx$ est donnée par l'équation ci-dessous

$$p_y(y) = \frac{p(x)}{\left| \frac{\partial Y}{\partial y} \right|} \quad (\text{II.19})$$

II.2.2.a. Le Jacobien

Le jacobien J est une valeur scalaire qui est le déterminant d'une matrice ($M \times M$) jacobienne J de dérivées partielles [II.7]. Si $M = 2$, alors :

$$J = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} \end{bmatrix} \quad (\text{II.20})$$

Puisque le jacobien J est le déterminant de la matrice jacobienne [II.7]

$$Jac = |J| = \begin{vmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} \end{vmatrix} = \frac{\partial y_1}{\partial x_1} \frac{\partial y_2}{\partial x_2} - \frac{\partial y_1}{\partial x_2} \frac{\partial y_2}{\partial x_1}. \quad (\text{II.21})$$

II.2.2.b. Le Jacobien et la matrice de séparation

Un exemple avec $M = 2$ est utilisé pour illustrer la relation entre la matrice jacobienne J et la matrice de séparation W . A partir de l'équation (II.1) $y = Wx$, où

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \quad W = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (\text{II.22})$$

et en substituant ces valeurs dans l'équation (II.1), on obtient :

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (\text{II.23})$$

L'évaluation de cette équation montre que :

$$y_1 = W_{11}x_1 + W_{12}x_2 \quad (2.24)$$

et

$$y_2 = W_{21}x_1 + W_{22}x_2 \quad (\text{II.25})$$

A partir de l'équation (II.20), la matrice jacobienne requiert des expressions pour

$$\frac{\partial y_1}{\partial x_1}, \frac{\partial y_1}{\partial x_2}, \frac{\partial y_2}{\partial x_1}, \text{ et } \frac{\partial y_2}{\partial x_2}.$$

Les deux premières dérivées partielles sont obtenues à partir de l'équation (II.24), tandis que les deux autres sont obtenues à partir de l'équation (II.25).

$$\frac{\partial y_1}{\partial x_1} = W_{11} \quad (\text{II.26})$$

$$\frac{\partial y_1}{\partial x_2} = W_{12} \quad (\text{II.27})$$

$$\frac{\partial y_2}{\partial x_1} = W_{21} \quad (\text{II.28})$$

$$\frac{\partial y_2}{\partial x_2} = W_{22} \quad (\text{II.29})$$

La substitution des équations (II.26) à (II.29) dans l'équation (II.20) donne la matrice jacobienne.

$$J = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \quad (\text{II.30})$$

L'équation (II.30) est identique à l'expression utilisée dans l'équation (II.22). Donc

$$J = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} = W \quad (\text{II.31})$$

Le déterminant de J est alors égal au déterminant de W , et donc à partir de l'équation (II.21)

$$Jac = |J| = |W| \quad (\text{II.32})$$

L'expression multivariable de l'entropie dans l'équation (II.18) nécessite une expression pour

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

$p_y(y)$ où $p_y(y) = \frac{p_x(x)}{\left| \frac{\partial y}{\partial x} \right|}$ comme montré dans l'équation (II.19), $\left| \frac{\partial y}{\partial x} \right|$ est le Jacobien, à partir de l'équation (II.32)

$$\left| \frac{\partial y}{\partial x} \right| = J = |W| \quad (II.33)$$

et la pdf du signal reconstruit est donnée par :

$$p_y(y) = \frac{p_x(x)}{|W|} \quad (II.34)$$

Ce résultat conduit à l'expression de l'entropie utilisée dans l'algorithme Infomax. En substituant l'expression de la pdf des signaux reconstruits trouvés dans l'équation (II.34) dans l'expression de l'entropie de l'équation (II.18), nous obtenons :

$$H(Y) = -\frac{1}{N} \sum_t^N \ln \frac{p_x(x^t)}{W/p_s(y^t)} \quad (II.35)$$

A partir des propriétés des logarithmes, cette équation peut être écrite comme suit :

$$H(Y) = -\frac{1}{N} \sum_t^N (\ln p_x(x^t) - \ln |W| - \ln p_s(y^t)) \quad (II.36)$$

On peut écrire l'équation (II.36) comme suit :

$$H(Y) = -\frac{1}{N} \sum_t^N (\ln p_x(x^t) + \ln |W| + \frac{1}{N} \sum_t^N (\ln p_s(y^t)) \quad (II.37)$$

D'après l'équation (II.3), le premier terme de l'équation (II.37) représente l'entropie de X :

$$H(X) = -\frac{1}{N} \sum_t^N (\ln p_x(x^t)) \quad (II.38)$$

L'équation (II.37) devient comme suit :

$$H(Y) = H(X) + \ln |W| + \frac{1}{N} \sum_t^N (\ln p_s(y^t)) \quad (II.39)$$

La matrice de séparation qui maximise l'entropie $H(Y)$ n'affecte pas l'entropie $H(X)$, donc $H(X)$ peut être ignoré, ce qui signifie que la matrice de séparation maximise aussi l'équation (II.39).

$$h(Y) = \ln |W| + \frac{1}{N} \sum_t^N (\ln p_s(y^t)) \quad (II.40)$$

où t est l'index de temps.

L'équation (II.40) peut être modifiée en ignorant l'ordre des signaux M ,

$$h(Y) = \ln |W| + \frac{1}{N} \sum_{i=1}^M \sum_{t=1}^N (\ln p_s(y_i^t)) \quad (II.41)$$

La matrice de séparation W qui maximise l'équation (II.41) maximise l'entropie en Y , ce qui induit que les lignes de la matrice de Y sont indépendantes. Puisque y est l'inverse de Y , cela

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

implique que les lignes de la matrice de y sont indépendantes, et donc que W est la matrice de séparation produisant les signaux d'origine. L'équation (II.41) est l'équation fondamentale utilisée dans l'algorithme Infomax.

II.2.3 Le Gradient Ascent

Le Gradient Ascent est un processus itératif pour faire un pas dans la direction du gradient maximal jusqu'à ce qu'un maximum local soit atteint.

L'équation (II.41) donne l'entropie des signaux transformés Y dans un intervalle constant. L'objectif d'Infomax est de trouver une matrice de séparation W qui maximise l'entropie de Y , c.-à-d. maximise $h(Y)$ où $Y = g(y) = g(Wx)$. Le Gradient accent est la méthode utilisée pour optimiser la matrice de séparation W .

II.2.3. a. Le Gradient d'entropie

Dans l'équation (II.41), en prenant l'espérance mathématique réécrite comme suit :

$$h(Y) = \ln|W| + E\{\sum_{i=1}^M \ln p_s(y_i)\} \quad (\text{II.42})$$

Le gradient est trouvé en prenant la dérivée partielle de h par rapport W , c.-à-d. $\frac{\partial h}{\partial W}$.

Le gradient est d'abord trouvé par rapport à un élément de W , $\frac{\partial h}{\partial W_{ij}}$ et ensuite étendu à chaque élément. On a par conséquent :

$$\frac{\partial h}{\partial W_{ij}} = E\left\{\sum_{i=1}^M \frac{\partial \ln g'(y_i)}{\partial W_{ij}}\right\} + \frac{\partial \ln|W|}{\partial W_{ij}} \quad (\text{II.43})$$

La simplification de cette dérivée partielle a lieu en deux parties, en traitant d'abord le premier terme et ensuite le second terme.

Simplification du premier terme de l'équation (II.43), c.-à-d.

$$\frac{\partial \ln g'(y_i)}{\partial W_{ij}} \quad (\text{II.44})$$

On pose

$$u = g'(y_i) \quad (\text{II.45})$$

L'équation (II.44) peut être exprimée par :

$$\frac{\partial \ln u}{\partial W_{ij}} \quad (\text{II.46})$$

En utilisant la règle de Chain Rule [II.7], nous obtenons :

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

$$\frac{\partial \ln u}{\partial w_{ij}} = \frac{u'}{u} \quad (\text{II.47})$$

L'équation (II.45) donne une expression pour u , et la dérivée de u est :

$$u' = \frac{\partial u}{\partial w_{ij}} = \frac{\partial \ln g'(y_i)}{\partial w_{ij}} \quad (\text{II.48})$$

En remplaçant l'expression de l'équation (II.47) par les équations (II.45) et (II.48), nous obtenons :

$$\frac{\partial \ln u}{\partial w_{ij}} = \frac{1}{g'(y_i)} \frac{\partial g'(y_i)}{\partial w_{ij}} \quad (\text{II.49})$$

L'expression $\frac{\partial g'(y_i)}{\partial w_{ij}}$ dans l'équation (II.49) peut être simplifiée à l'aide de la « Chain Rule» [II.7],

$$\frac{\partial g}{\partial w} = \frac{\partial g}{\partial y} \frac{\partial y}{\partial w} \quad (\text{II.50})$$

À partir de l'équation (II.50), on a :

$$\frac{\partial g'(y_i)}{\partial w_{ij}} = \frac{\partial g'(y_i)}{\partial (y_i)} \frac{\partial (y_i)}{\partial w_{ij}} \quad (\text{II.51})$$

L'équation (II.51) est simplifiée encore

$$\frac{\partial g'(y_i)}{\partial w_{ij}} = g''(y_i) \frac{\partial (y_i)}{\partial w_{ij}} \quad (\text{II.52})$$

L'expression $\frac{\partial (y_i)}{\partial w_{ij}}$ est un élément d'un mélange x . D'après les équations (II.25) et (II.26) on obtient :

$$\frac{\partial (y_i)}{\partial w_{ij}} = x_j \quad (\text{II.53})$$

En substituant l'équation (II.53) dans l'équation (II.52), on obtient :

$$\frac{\partial g'(y_i)}{\partial w_{ij}} = g''(y_i) x_j \quad (\text{II.54})$$

En utilisant l'équation (II.54), l'équation (II.49) devient :

$$\frac{\partial \ln u}{\partial w_{ij}} = \frac{1}{g'(y_i)} g''(y_i) x_j \quad (\text{II.55})$$

L'équation (II.55) peut maintenant remplacer l'expression $\frac{\partial \ln g'(y_i)}{\partial w_{ij}}$ dans l'espérance mathématique de l'équation (II.43). Cela donne :

$$\frac{\partial h}{\partial w_{ij}} = E \left\{ \sum_{i=1}^M \frac{g''(y_i) x_j}{g'(y_i)} \right\} + \frac{\partial \ln |W|}{\partial w_{ij}} \quad (\text{II.56})$$

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

Nous posons :

$$\Psi(y_i) = \frac{g''(y_i)}{g'(y_i)} \quad (\text{II.57})$$

L'équation (II.56) peut alors être exprimée par :

$$\frac{\partial h}{\partial w_{ij}} = E\{\sum_{i=1}^M \Psi(y_i) x_j\} + \frac{\partial \ln|W|}{\partial w_{ij}} \quad (\text{II.58})$$

L'équation (II.58) représente la dérivée partielle de l'entropie dans l'équation (II.43) avec le premier terme totalement simplifié.

Pour simplifier le second terme $\frac{\partial \ln|W|}{\partial w_{ij}}$ on prend un exemple pour montrer cette simplification.

$$\frac{\partial \ln|W|}{\partial w_{ij}} = [W^{-T}]_{ij} \quad (\text{II.59})$$

où $[W^{-T}]_{ij}$ est un élément de l'inverse de la matrice de séparation transposée :

$$W^{-T} = [W^T]^{-1} \quad (\text{II.60})$$

Lorsque $M = 2$, la matrice de séparation W est :

$$W = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \quad (\text{II.61})$$

La transposée W^T est :

$$W^T = \begin{bmatrix} W_{11} & W_{21} \\ W_{12} & W_{22} \end{bmatrix} \quad (\text{II.62})$$

et le déterminant de la matrice de séparation $|W|$ est égal au déterminant de la matrice de séparation transposée $|W^T|$

$$|W| = W_{11}W_{22} - W_{21}W_{12} = |W^T| \quad (\text{II.63})$$

On prend $i=j=1$ comme exemple

$$\frac{\partial \ln|W|}{\partial w_{11}} = \frac{\partial \ln(W_{11}W_{22} - W_{21}W_{12})}{\partial w_{11}} = \frac{W_{22}}{W_{11}W_{22} - W_{21}W_{12}} = \frac{W_{22}}{|W|} \quad (\text{II.64})$$

En utilisant l'équation (II.62) dans l'équation (II.60), on obtient la transposée inverse W^{-T} :

$$W^{-T} = [W^T]^{-1} = \frac{1}{W_{11}W_{22} - W_{21}W_{12}} \begin{bmatrix} W_{22} & -W_{12} \\ W_{21} & W_{11} \end{bmatrix} = \frac{W_{22}}{W_{11}W_{22} - W_{21}W_{12}} = \frac{W_{22}}{|W|} \quad (\text{II.65})$$

L'équation (II.59) peut être substituée dans l'expression du gradient d'entropie dans l'équation (II.58), entraînant :

$$\frac{\partial h}{\partial w_{ij}} = E\{\sum_{i=1}^M \Psi(y_i) x_j\} + [W^{-T}]_{ij} \quad (\text{II.66})$$

Si on considère tous les éléments de la matrice de séparation, elle génère une expression complète du gradient d'entropie ∇h , défini comme suit :

$$\nabla h = \begin{bmatrix} \frac{\partial h}{\partial w_{11}} & \frac{\partial h}{\partial w_{12}} & \dots & \frac{\partial h}{\partial w_{1M}} \\ \frac{\partial h}{\partial w_{21}} & \frac{\partial h}{\partial w_{22}} & \ddots & \frac{\partial h}{\partial w_{2M}} \\ \frac{\partial h}{\partial w_{M1}} & \frac{\partial h}{\partial w_{M2}} & \dots & \frac{\partial h}{\partial w_{MM}} \end{bmatrix} \quad (\text{II.67})$$

Le gradient d'entropie ∇h pour tous les éléments de la matrice de séparation W est alors :

$$\nabla h = W^{-T} + E\{\sum_{i=1}^M \Psi(y) x^T\} \quad (\text{II.68})$$

En supposant que les signaux soient ergodiques [II.8], l'équation (II.68) devient alors :

$$\nabla h = W^{-T} + \frac{1}{N} (\sum_{t=1}^M \Psi(y^t) [x^t]^T) \quad (\text{II.69})$$

II.2.3. b. L'Algorithme Gradient Ascent Infomax

La matrice de séparation optimale W est trouvée en maximisant l'entropie. Cela est fait itérativement jusqu'à atteindre un maximum local.

$$W_{new} = W_{old} + \eta \nabla h \quad (\text{II.70})$$

où η est une petite constante. En insérant l'expression de ∇h dans l'équation (II.69) dans l'équation (II.70), nous obtenons l'expression qui optimise la matrice de séparation W pour maximiser l'entropie :

$$W_{new} = W_{old} + \eta (W_{old}^{-T} + \frac{1}{N} (\sum_{t=1}^M \Psi(y^t) [x^t]^T)) \quad (\text{II.71})$$

L'équation (II.71) est la forme générale de l'algorithme Infomax utilisant l'algorithme du Gradient Ascent pour optimiser la matrice de séparation W . Cependant, il est important de noter que l'algorithme du gradient Ascent ne trouve qu'un maximum local, ce qui n'est pas nécessairement le maximum global de la fonction. Cet algorithme est initié à partir de différents points de départ.

II.3.1 Modélisation et Simulation

Il est clair que l'entropie et son gradient dépendent tous les deux d'un modèle pdf des signaux sources, et il convient de rappeler que les signaux extraits sont transformés par leur modèle. Un autre paramètre de modélisation pdf est le kurtosis K . Le kurtosis caractérise la

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

forme de pic ou d'aplatissement relatifs d'une distribution comparée à une distribution normale. Un kurtosis positif indique une distribution relativement pointue, tandis qu'un kurtosis négatif signale une distribution relativement aplatie. Le kurtosis K est défini par :

$$K = \frac{E\{x^4\}}{E\{x^2\}^2} - 3 \quad (\text{II.72})$$

où $E\{x^4\}$ est le moment d'ordre quatre et $E\{x^2\}$ est le moment d'ordre deux. Les signaux gaussiens ont zéro kurtosis. Lorsque le kurtosis est négatif, le signal est sous-gaussien. Un signal super-gaussien a un kurtosis positif.

II.3.2 Adaptation de l'algorithme Infomax

Infomax utilise un pdf super-gaussien pour modéliser les signaux audios [II.8]. Un cdf typique utilisé pour modéliser les signaux qui ont un kurtosis positif est la tangente hyperbolique, tracée sur la figure II.3 et utilisée dans notre simulation.

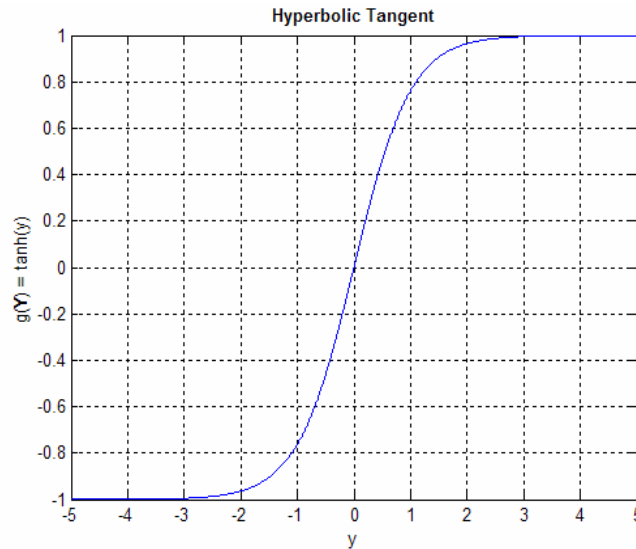


Figure II. 3. Fonction tangente hyperbolique

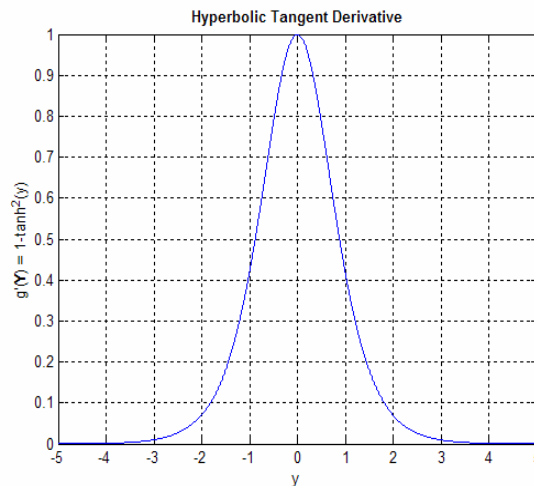


Figure II.4. Dérivée tangente hyperbolique.

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

Par Conséquent,

$$Y = g(y) = \tanh(y) \quad (\text{II.73})$$

La dérivée $g'(y)$, ou la pdf, est illustré sur la figure II.4. Cette pdf n'est pas idéale car sa surface est 2, mais a également été utilisé avec succès dans [II.8].

$$g'(y) = \frac{d}{dy} \tanh(y) = 1 - \tanh^2(y) \quad (\text{II.74})$$

Avec l'adaptation de l'équation (II.41) pour un signal de kurtosis positif, l'expression de l'entropie devient :

$$h(Y) = \ln|W| + \frac{1}{N} \sum_{t=1}^M \sum_{t=1}^N \ln(1 - \tanh^2 y_i^t) \quad (\text{II.75})$$

Le gradient d'entropie ∇h contient le rapport $\Psi(y) = \frac{g''(y)}{g'(y)}$, alors que la pdf $g'(y)$ est donnée dans l'équation (2.74). La dérivée de la pdf, $g''(y)$ est aussi nécessaire.

$$g''(y) = \frac{d}{dy} g'(y) = \frac{d}{dy} (1 - \tanh^2(y)) = -\frac{d}{dy} (\tanh^2(y)) = -2 \tanh(y) \frac{d}{dy} \tanh(y) \quad (\text{II.76})$$

Puisque $\frac{d}{dy} \tanh(y)$ est égal à $g'(y)$, l'équation (II.76) devient :

$$g''(y) = -2 \tanh(y) g'(y) \quad (\text{II.77})$$

et l'expression de la dérivée du modèle pdf est :

$$g''(y) = -2 \tanh(y) g'(y) \quad (\text{II.78})$$

Le rapport $\Psi(y)$ dans le gradient ∇h est :

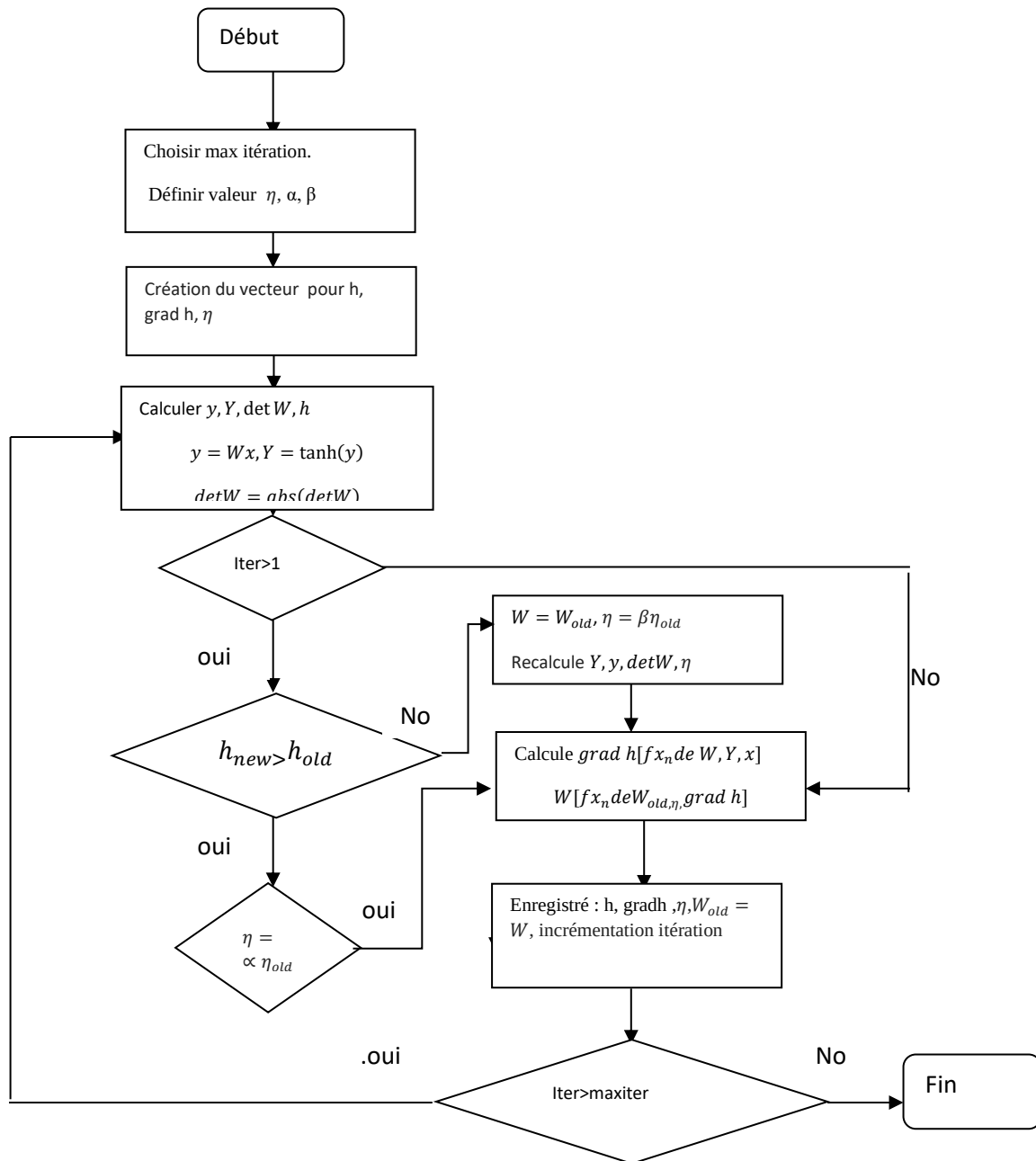
$$\Psi(y) = \frac{g''(y)}{g'(y)} = \frac{-2 \tanh(y) g'(y)}{g'(y)} = -2 \tanh(y) \quad (\text{II.79})$$

En adaptant l'équation (2.69) pour un signal de kurtosis positif, on obtient le gradient ∇h

$$\nabla h = W^{-T} + \frac{1}{N} (\sum_{t=1}^M -2 \tanh(y^t) [x^t]^T) \quad (\text{II.80})$$

Des expressions pour la mise en œuvre d'Infomax kurtosis positif c.-à-d. pour un signal audio ont été obtenues. Maintenant, un outil informatique est nécessaire pour exécuter l'algorithme de manière efficace.

II.3.3 Organigramme de la méthode Infomax



II.3.4 Paramètres de la méthode Infomax

$M = 2$ signaux sources indiqués par Signal 1 (speech1.wav) et Signal 2 (speech2.wav) sur respectivement les figures II. 5 Les mélanges aléatoires des deux signaux source sont montrés sur respectivement les figures II. 6. L'algorithme Infomax a été simulé avec les valeurs du tableau (II. 1).

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

Tableau II.1. Valeurs des variables de l'algorithme Infomax pour $M = 2$.

Maximum nombre d'itérations pour Infomax	100
Pas initial η	0.1
Pas croissant α	1.2
Pas décroissant β	0.1
Gradient Ascent répétitions	05

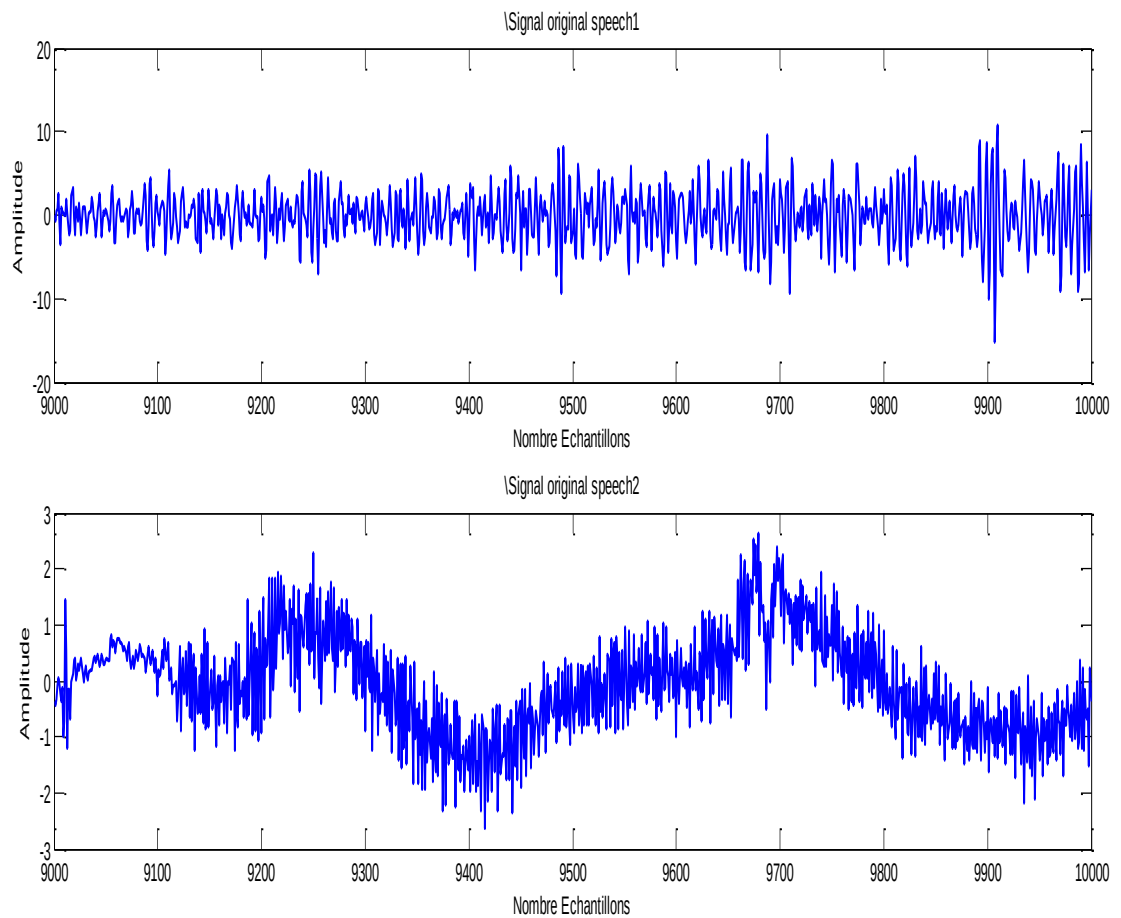


Figure II.5. Sources originales "speech1.wav, speech2.wav"

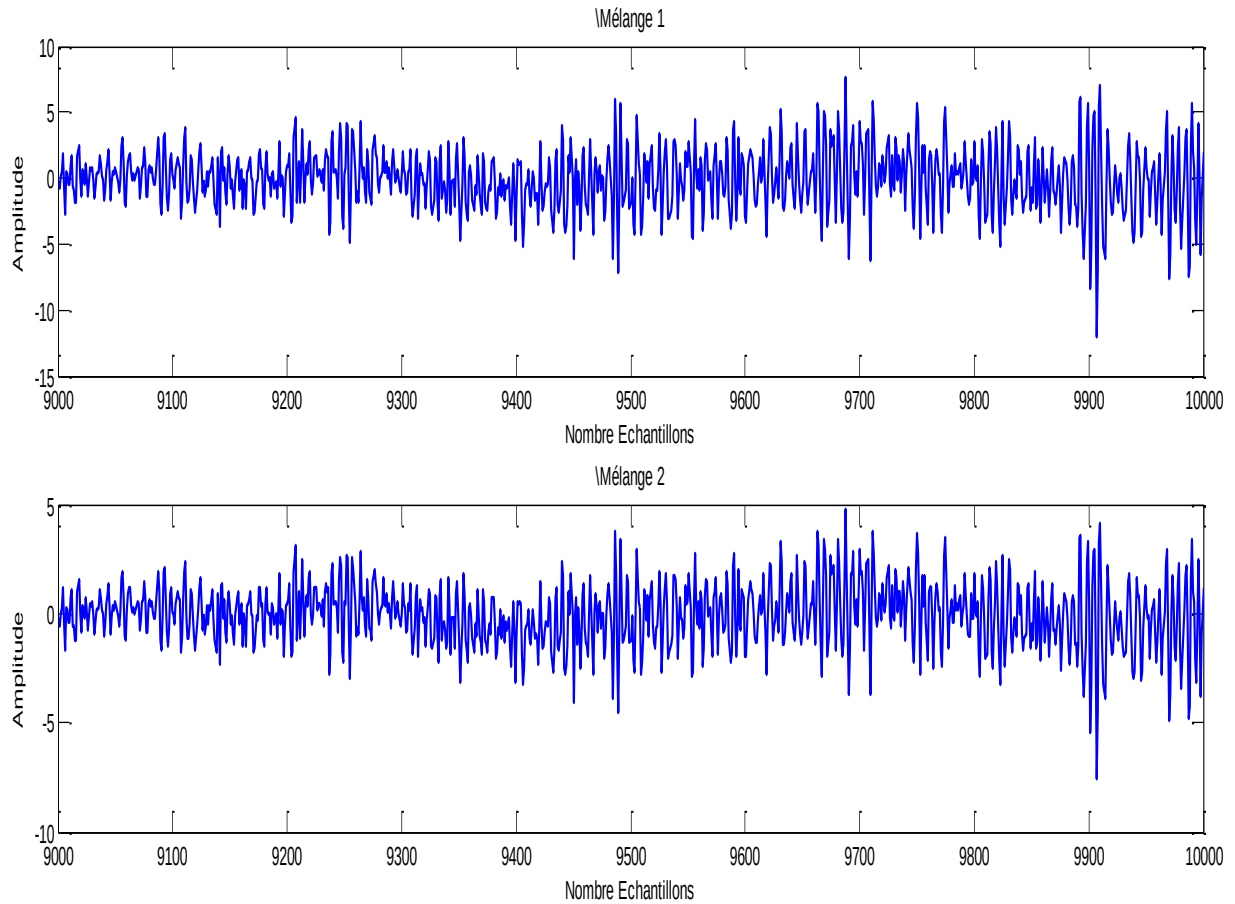


Figure II. 6. Mélanges

Les signaux extraits par l'algorithme Infomax sont illustrés sur les figures II.7. La similitude des signaux extraits avec les signaux source originaux est évidente. Infomax ne conserve pas l'ordre des signaux. La figure II.8 montre les valeurs de la fonction pour l'entropie $h(Y)$, l'amplitude du gradient d'entropie ∇h , et η . Sur la figure II.8, les résultats ont convergé après 15 itérations, bien que généralement, les résultats convergent après environ 35 à 50 itérations. Le dessin du milieu représente l'amplitude du gradient d'entropie $|\nabla h|$. Quand l'entropie est maximisée, l'amplitude du gradient diminue, indiquant que le maximum est très proche de la valeur actuelle. Le dessin en bas montre comment le pas η change avec l'algorithme "apprentissage du Gradient Ascent".

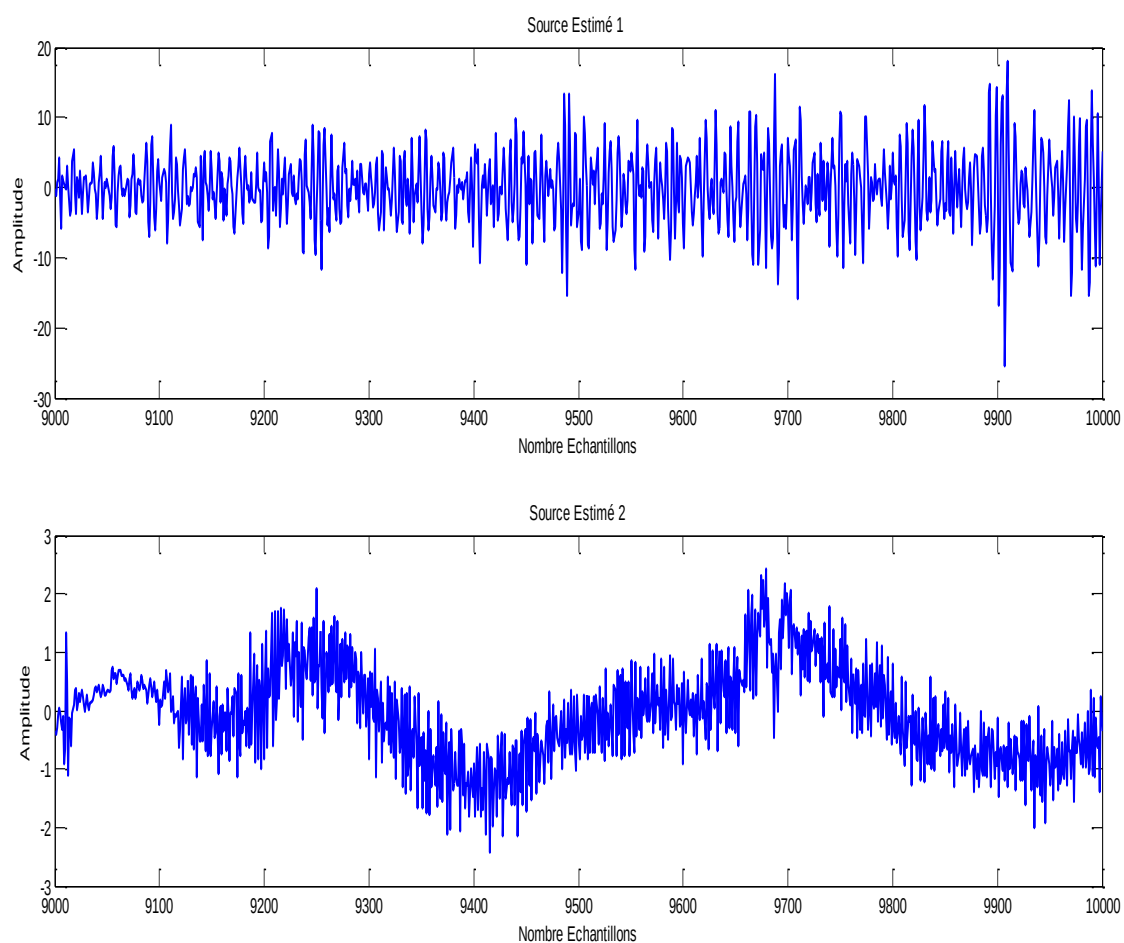


Figure II.7. Sources Estimées.

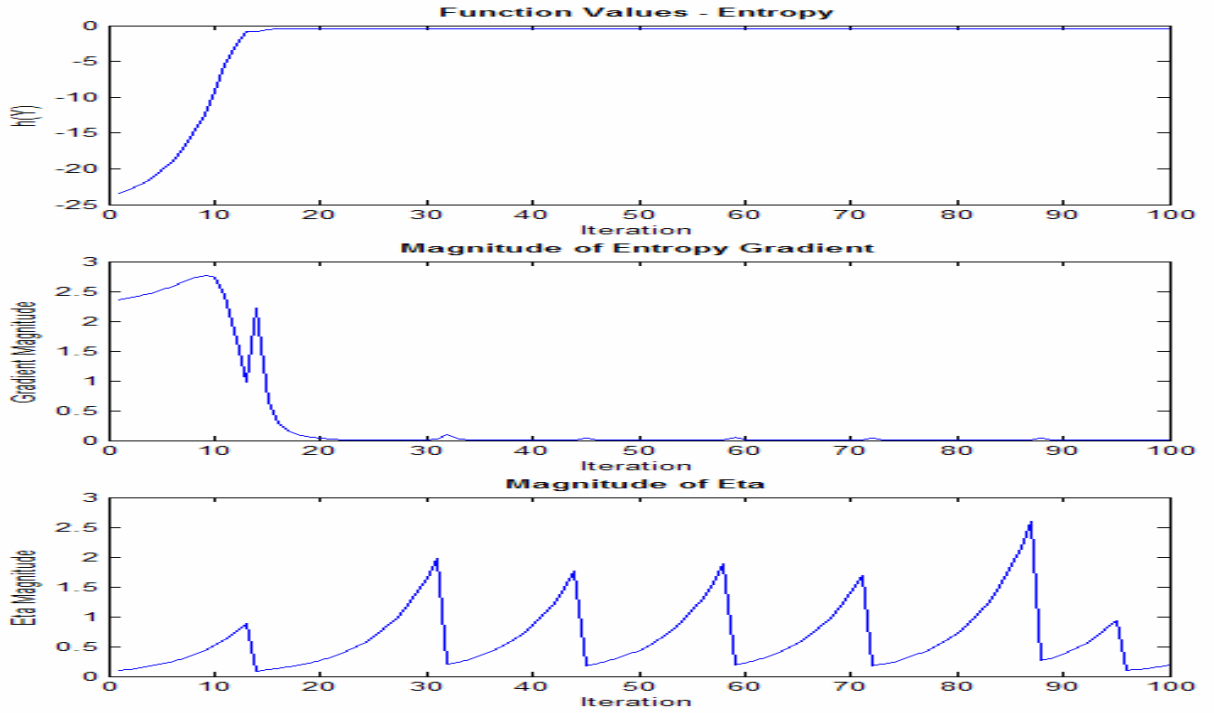


Figure II.8. Entropie $H(Y)$, ∇h , η pour $M=2$.

II.4 Audio Séparation utilisant la méthode Robust ICA

L'analyse en composantes indépendantes (ICA) vise à décomposer un vecteur aléatoire observé en variables statistiquement indépendantes. Deux approches principales de l'ICA ont été proposées à ce jour. Dans la définition d'origine de l'ICA réalisée dans les premiers travaux tels que [II.9] et [II.10], les composants indépendants sont extraits conjointement ou simultanément par une approche parfois appelée symétrique. L'approche par déflation estime les sources l'une après l'autre [2.11]. Une nouvelle méthode pour les ICA déflationnistes est appelée RobustICA. Cette technique simple consiste à optimiser la fonction de contraste de Kurtosis. L'étape conduisant au maximum global du contraste le long de la recherche se trouve parmi les racines d'un polynôme.

II.4.1 Modèle du signal

Le vecteur aléatoire observé $x \in \mathbb{C}^L$ est supposé être généré à partir du modèle de mélange linéaire instantané :

$$x = Hs + n \quad (\text{II.81})$$

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

où le vecteur source $= [s_1, s_2, s_3, \dots, s_k]^T \in \mathbb{C}^K$ est constitué de $K \leq L$ composants inconnus mutuellement indépendants. Les éléments de la matrice de mélange $H \in \mathbb{C}^{K \times L}$ sont également inconnus, de même que le vecteur de bruit n qui est supposé être uniquement indépendant des sources. Notre étude est concentrée sur implémentation en bloc, utilisant les informations contenues dans le bloc du signal observé.

Étant donné un signal de sortie du capteur bloc composé de T échantillons, ICA vise à estimer la valeur correspondante à la réalisation du vecteur source.

II.4.2 La méthode Fast ICA modifiée

Dans l'approche de déflation en ICA, on recherche un vecteur d'extraction W afin que l'estimation :

$$y = w^H \cdot x \quad (\text{II.82})$$

où $(.)^H$ désigne l'opérateur conjugué-transposé et x le vecteur aléatoire observé, qui maximise un critère d'optimalité ou une fonction de contraste, et devrait donc être une composante indépendante des autres. Le contraste largement utilisé est le kurtosis, défini comme le cumulants normalisés d'ordre quatre :

$$k(w) = \frac{E\{|y|^4\} - 2E^2\{|y|^2\} - |E\{y^2\}|^2}{E^2\{|y|^2\}} \quad (\text{II.83})$$

où $E\{.\}$ dénote l'espérance mathématique.

Nous pouvons imposer la normalisation $\|w\| = 1$. Le critère de maximisation du kurtosis (KM) basé sur le contraste (II.83) est assez général en ce sens qu'il n'exige pas que les observations soient pré-blanchies et qu'il peut être appliqué à des sources à valeurs réelles ou complexes sans aucune modification.

II.4.3 Optimisation du contraste

Dans le cas réel, le contraste (II.83) devient alors équivalent au critère du moment d'ordre quatre

$$M(W) = E\{|z|^4\} \quad (\text{II.84})$$

qui doit être optimisé sous une contrainte, $\|w\| = 1$, pour éviter des valeurs arbitrairement grandes de z . Sous la même contrainte, les critères (II.83) et (II.84) sont également équivalents si les sources ont une valeur complexe mais sont du second ordre circulaire, c'est-à-dire que la matrice du deuxième moment non circulaire $C_x = E\{xx^T\}$ est nulle. Par conséquent, le

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

contraste (II.84) est moins général que le critère (II.83) dans la mesure où il oblige les observations qu'il y ait pré-blanchiment et que les sources soient réelles ou complexes, mais circulaires. En effet, l'extension de l'algorithme FastICA aux signaux complexes n'est valable que pour les sources circulaires du second ordre [II.12], [II.13].

Sous la contrainte $\|w\|=1$, les points stationnaires de $M(W)$ sont obtenus en tant que condition de colinéarité

$$E\{|w^H x|^2 x x^H\} W = \lambda W \quad (\text{II.85})$$

où λ est un multiplicateur lagrangien.

L'équation (II.85) est une équation à point- fixe uniquement si λ est connu, ce qui n'est pas le cas ici. λ doit être déterminé de manière à satisfaire la contrainte, puis cela dépend de W_0 , la valeur optimale de W : $\lambda = M\{|W_0^H x|^4\}$.

λ est arbitrairement défini sur une valeur fixe déterministe de sorte que FastICA devient un algorithme de Newton standard approximatif [II.14], [II.15]. Dans [II.16], dans le cas de la valeur réelle, la matrice hessienne de $M(W)$ est approchée comme suit :

$$E\{W^T x x^T W\} x x^T \simeq E\{W^T x x^T W\} E\{x x^T\} = W^T W = I \quad (\text{II.86})$$

En conséquence, FastICA basé sur le kurtosis conduit l'algorithme du « Gradient Descent », avec un pas fixe judicieusement choisi, à la convergence cubique.

$$W^+ = W - \frac{1}{3} E\{x(W^T x)^3\} \quad (\text{II.87})$$

$$W^+ \leftarrow \frac{W^+}{\|w\|} \quad (\text{II.88})$$

II. 4.4 La méthode Robust ICA

Robust ICA optimise la forme généralisée du kurtosis suivante :

$$k(W) = \frac{E\{|y|^4\} - 2E^2\{|y|^2\} - |E\{y^2\}|^2}{E^2\{|y|^2\}} \quad (\text{II.89})$$

où W désigne un vecteur de séparation et y une source séparée. La définition ci-dessus du kurtosis peut être appliquée à des données réelles et complexes. De plus, le pré-blanchiment n'est pas nécessaire pour RobustICA. Elle utilise l'optimisation de la fonction de contraste absolu du kurtosis, au lieu de l'optimisation en point fixe, utilisée par FastICA :

$$\mu_{\text{opt}} = \text{argmax}(K(W + \mu g)) \quad (\text{II.90})$$

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

La recherche d'une direction peut être donné par le gradient du kurtosis $g = \nabla_w k(w)$, μ_{opt} est calculé algébriquement avec un coût de calcul minimum. Il peut être calculé à partir des racines d'un polynôme de faible degré qui maximise la valeur absolue de la fonction de contraste.

RobustICA peut traiter des problèmes réels et des sources complexes. Le blanchiment préalable n'est pas obligatoire avant RobustICA. Plusieurs composants indépendants (IC) dans ce cas peuvent être extraits avec la méthode de régression linéaire contrairement à l'orthogonalisation symétrique qui est utilisée par FastICA. La méthode est robuste à la présence de points de selle et extrema locaux parasites de la fonction de contraste [2.17]. RobustICA peut atteindre d'excellentes performances de séparation avec un coût de calcul faible, par rapport à d'autres implémentations ICA.

II.4 .5 L'Algorithme de la Méthode Robust ICA

À chaque itération, l'optimisation optimale effectue les opérations suivantes :

- 1) Calcul des coefficients polynomiaux. Pour le contraste du kurtosis, le polynôme est donné par :

$$p(\mu) = \sum_{k=0}^4 a_k \mu^k \quad (\text{II.91})$$

Les coefficients $\{a_k\}_{k=0}^4$ peuvent facilement être obtenus à chaque itération du bloc de signal observé et les valeurs actuelles de W et g . Le conditionnement numérique dans la détermination de μ_{opt} peut être amélioré en normalisant le vecteur du gradient.

- 2) Extraire les racines polynomiales $\{\mu_k\}_{k=1}^4$.

Les racines du polynôme du 4ème degré peuvent être trouvées pratiquement par l'utilisation de procédures algébriques classiques connues, telles que la formule de Ferrari [II.18].

- 3) Sélectionner la racine menant au maximum absolu : $\mu_{opt} = \text{argmax}(K(W + \mu_k g))$
- 4) Mettre à Jour : $W^+ = W + \mu_{opt} g$.
- 5) Normaliser : $W^+ \leftarrow W^+ / \|W^+\|$

II.4.6 Simulations

Tous les résultats expérimentaux sont implémentés dans des scripts Matlab.

Simulation 1

Dans cette simulation, nous utilisons trois signaux sources : signal 1 (speech1.wav), Signal 2 (speech2.wav) et Signal 3 (speech3.wav) montrés sur la figure II. 9.

Les mélanges observés des trois signaux source sont illustrés sur la figure II.10. Les sources estimées sont obtenues à l'aide de l'algorithme RobustICA comme illustré sur la figure II.11.

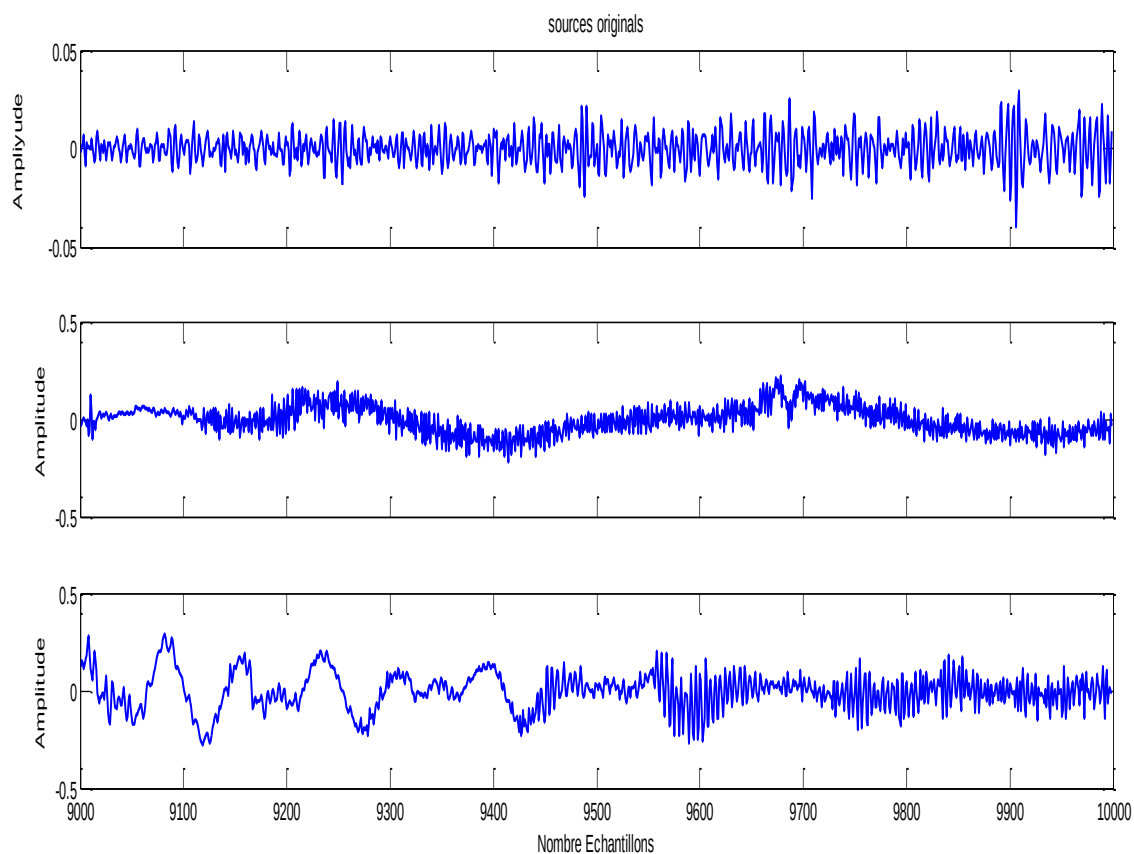


Figure II. 9. Sources originales (speech1.wav, speech2.wav, speech3.wav).

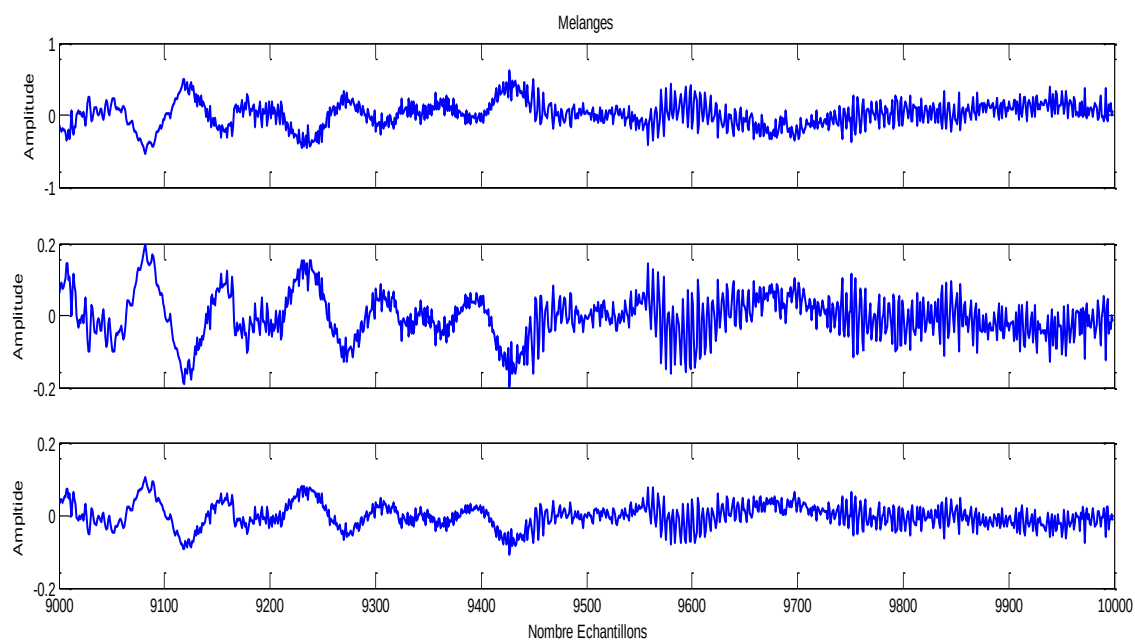


Figure II. 10. Mélanges.

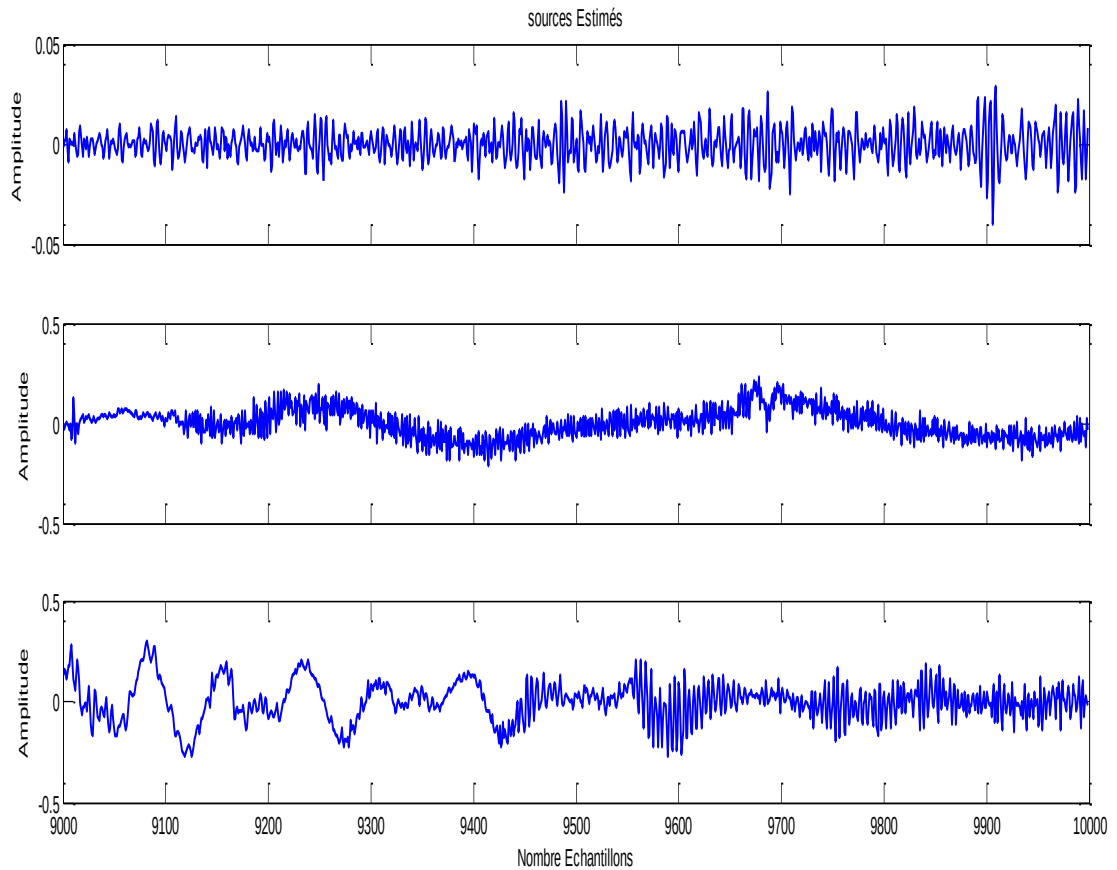


Figure II.11. Sources estimées.

Simulation 2

Dans cette simulation, nous utilisons à nouveau trois signaux sources, notés par Signal 1 (music1.wav) et Signal 2 (music2.wav) et Signal 3 (music3.wav) donnés sur la figure II.12. Les mélanges observés des trois signaux source sont illustrés sur la figure II.13 et les sources estimées trouvées par l'algorithme RobustICA sont illustrés sur la figure II.14.

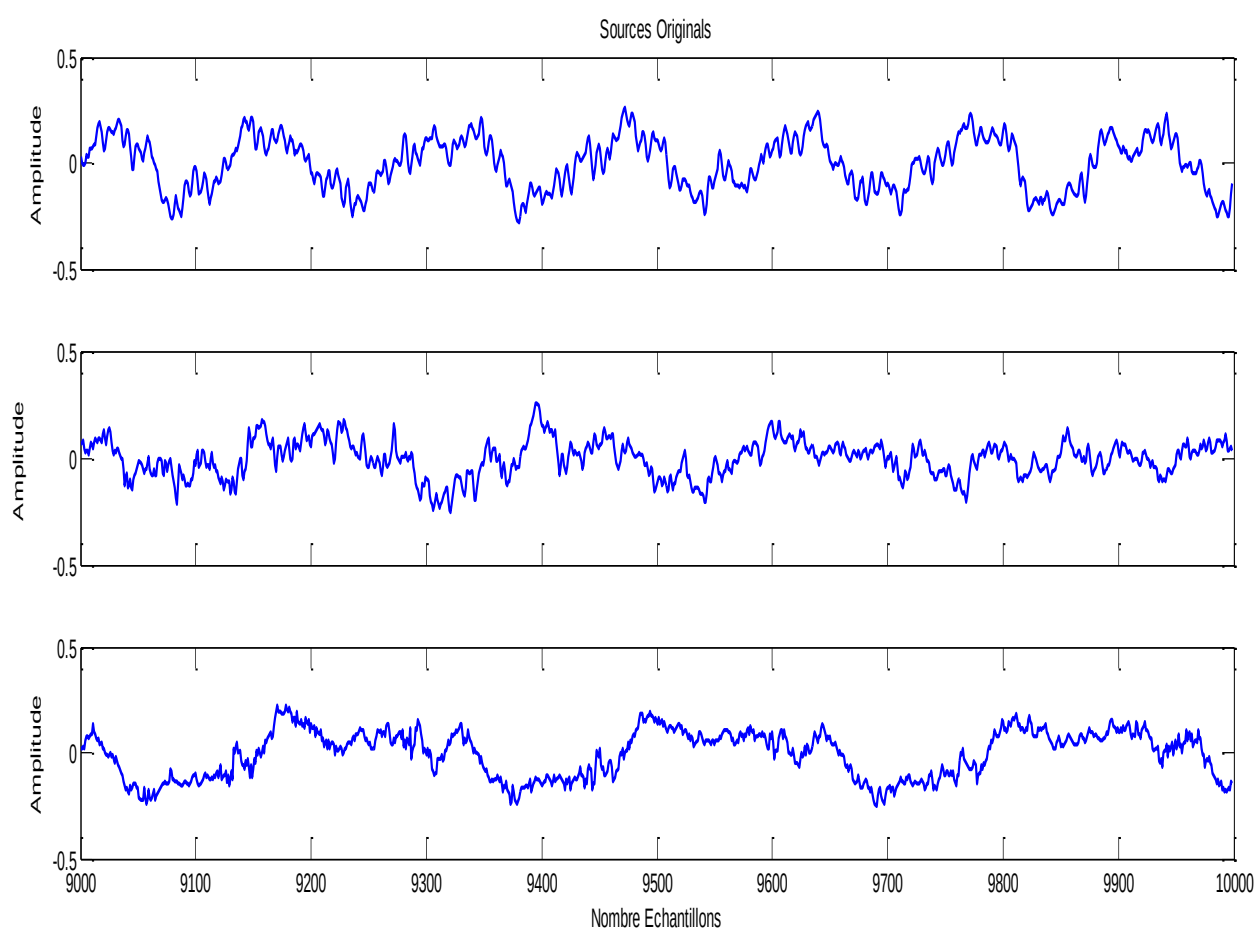


Figure II. 12. Sources originales (music1.wav, music2.wav, music3.wav).

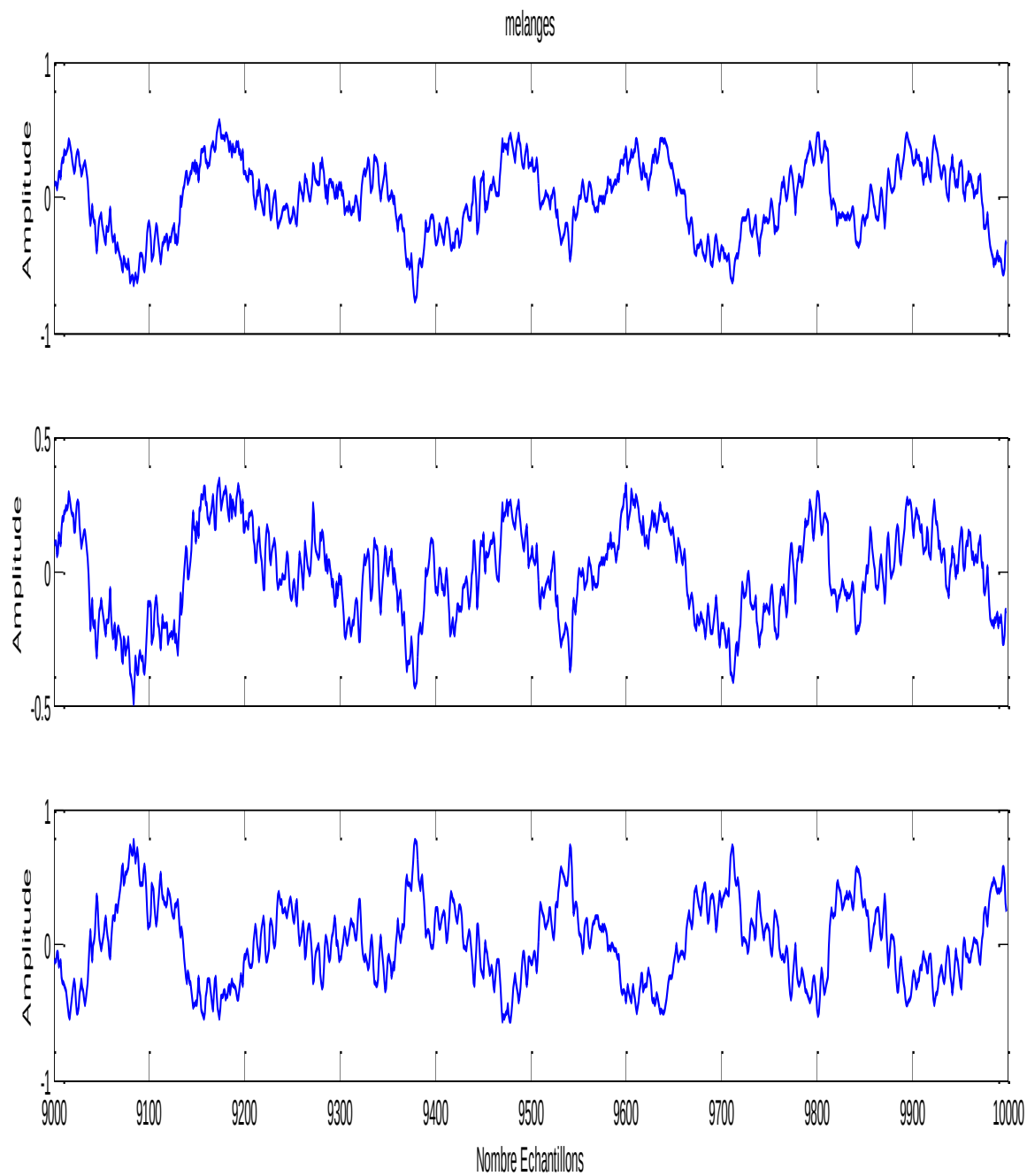


Figure II. 13. Mélanges.

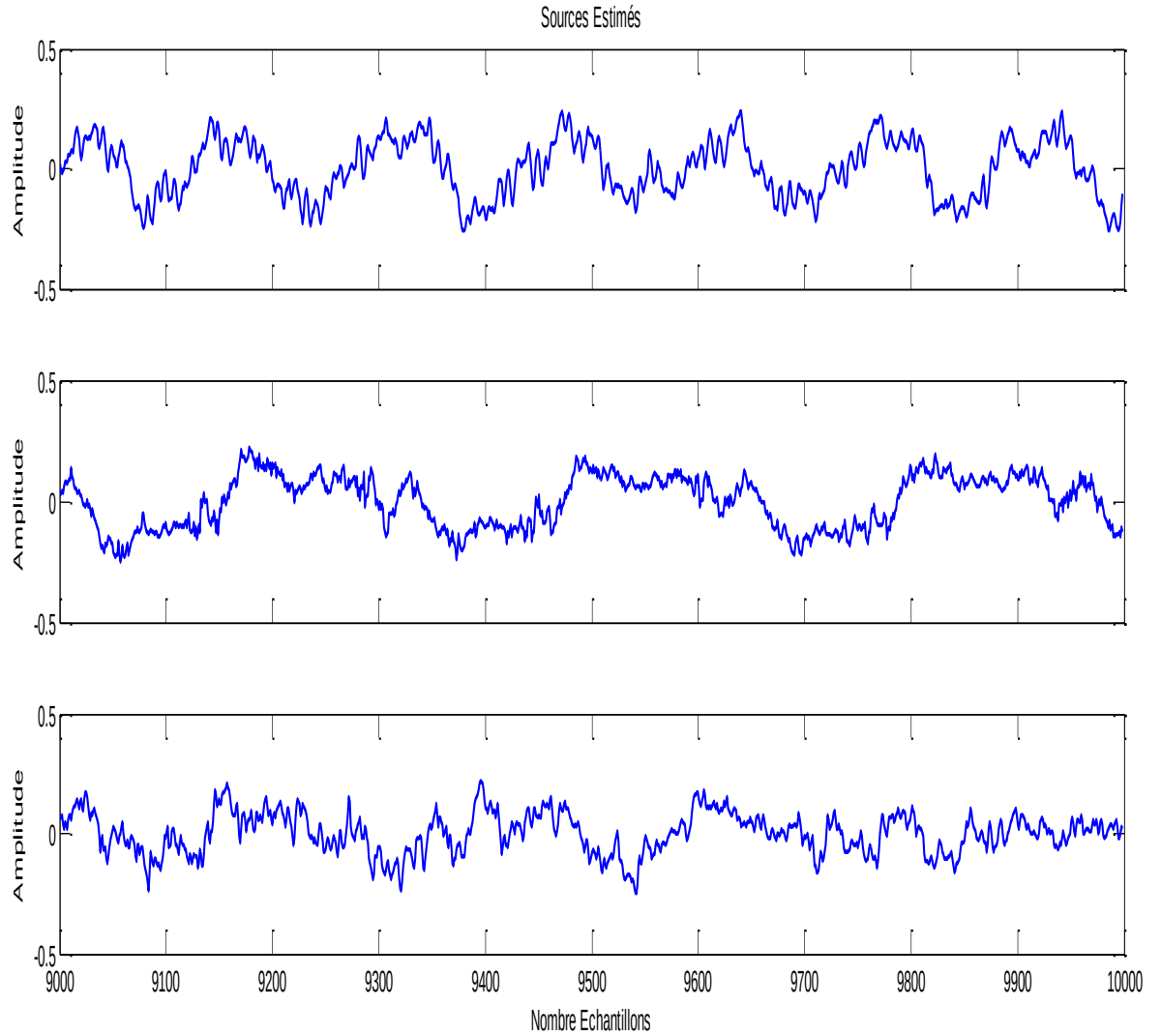


Figure II.14. Sources estimées.

Simulation 3

Dans cette dernière simulation, nous utilisons également trois signaux sources, Signal1 (speech2.wav), Signal 2 (music2.wav) et Signal 3 (music3.wav) illustrés sur la figure II.15. Les mélanges observés des trois signaux source sont illustrés sur la figure II.16 et les sources estimées trouvées par l'algorithme RobustICA sont données sur la figure II.17.

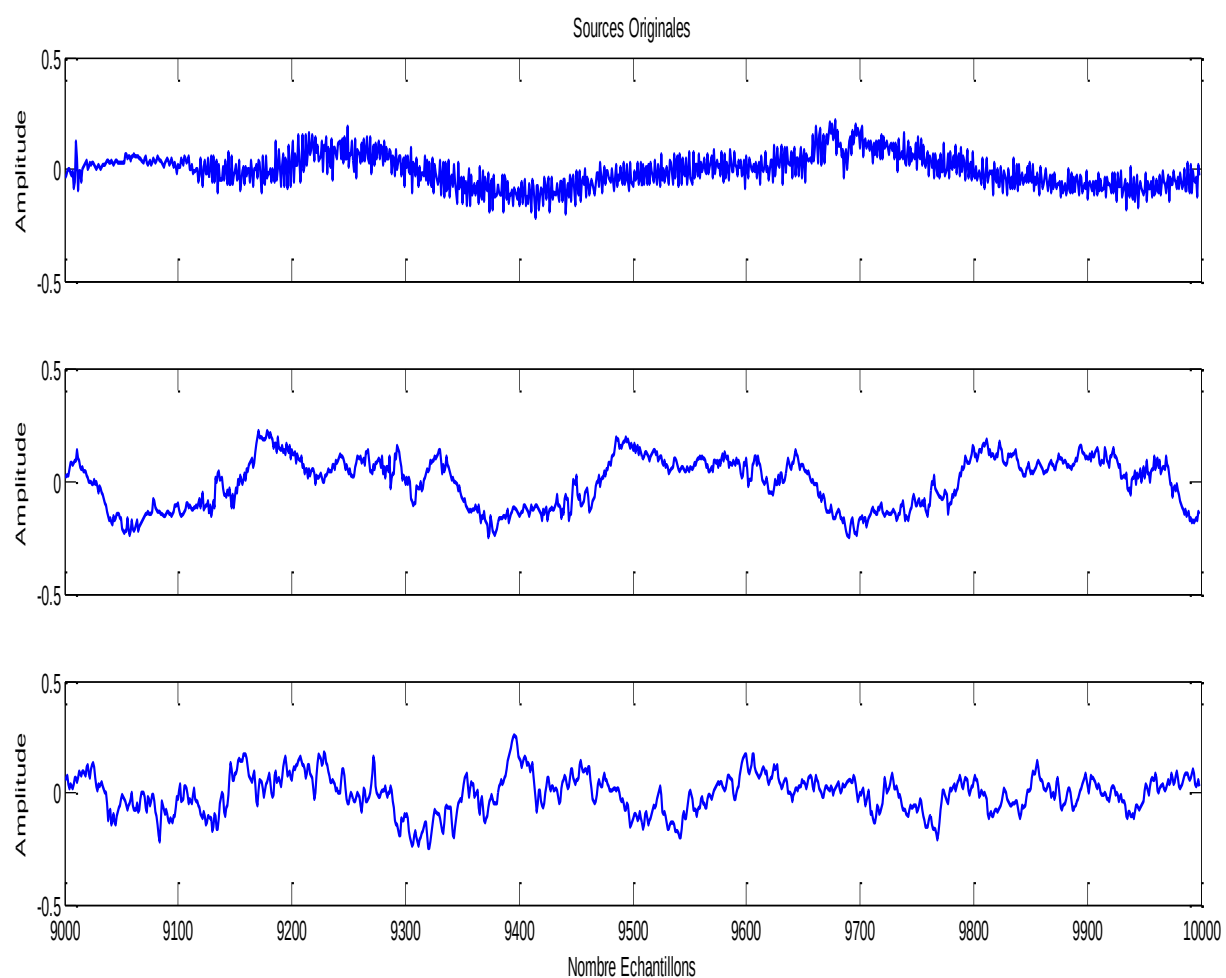


Figure II.15. Sources originales (speech2.wav, music2.wav, music3.wav).

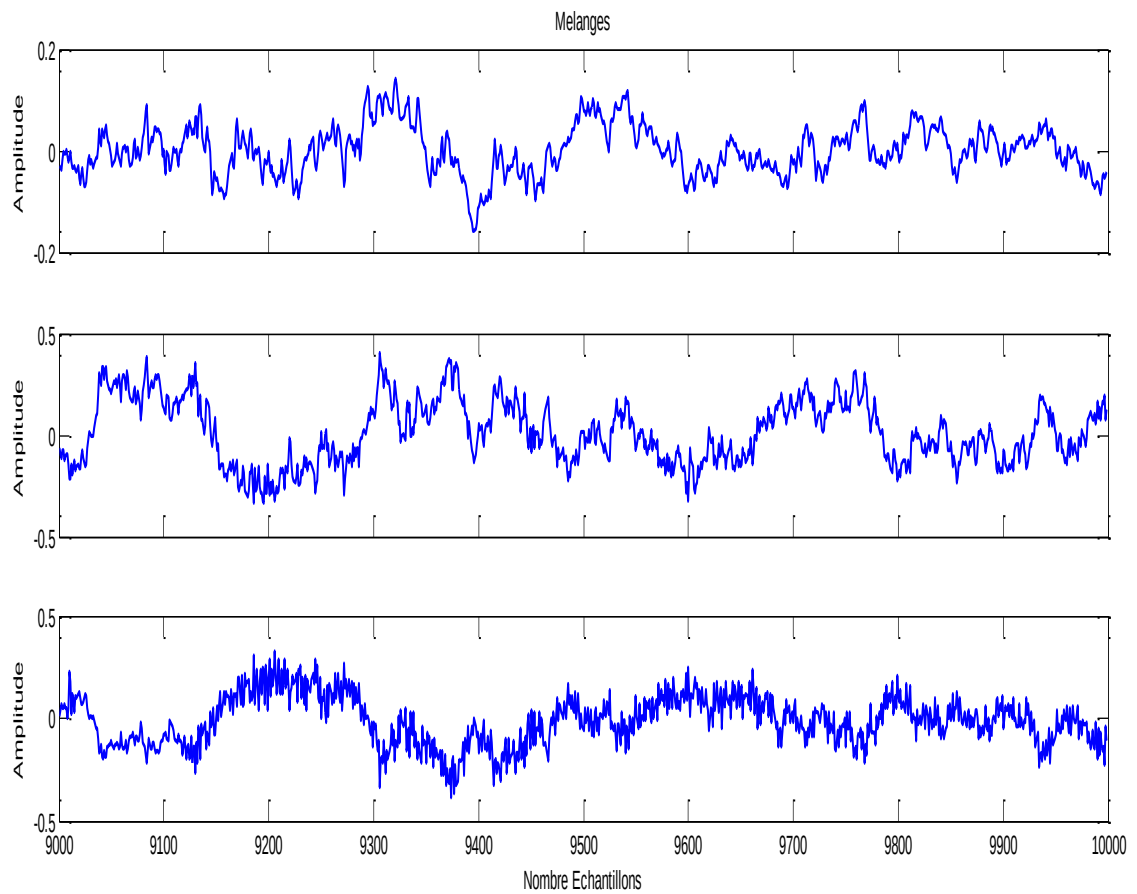


Figure II. 16. Mélanges.

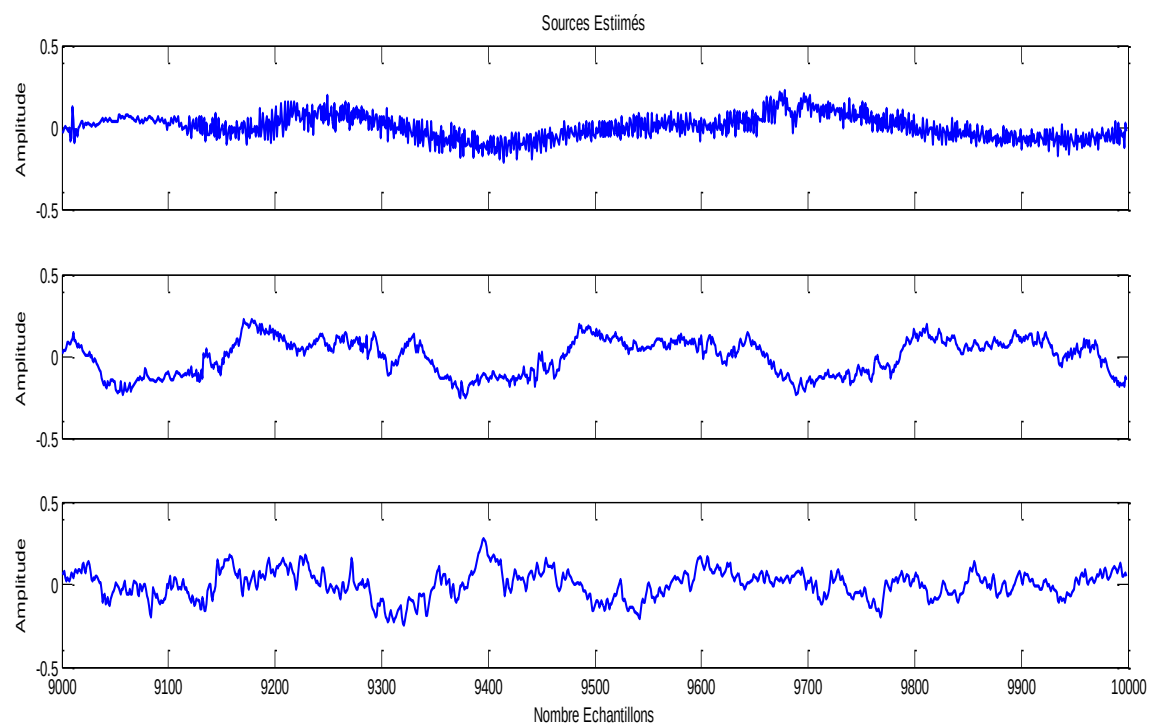


Figure II.17. Sources estimées.

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

Nous avons évalué les caractéristiques de convergence de RobustICA et la qualité d'extraction des sources tableau II.I. Dans le cas des sources de la parole (Simulation 1), une extraction estimée de la qualité des signaux est obtenue après 9 itérations. Dans le cas des sources de musique (Simulation 2), l'algorithme RobustICA effectue des extractions de qualité de performance plus robustes, et les signaux estimés sont obtenus après 10 itérations.

Dans le cas des sources de musique et de parole (Simulation 3), la performance est inférieure à celle des cas précédents.

Tableau II.2. Paramètres de performance de RobustICA

	Nombre d'itérations	% Signal Mean Square Error (SMSE)
Simulation 1	9	0.0054
Simulation 2	10	0.0051
Simulation 3	11	0.0056

II.5 Conclusion

Dans ce chapitre nous avons présenté le principe de deux méthodes appliquées à la séparation de sources. Ces méthodes sont : infomax, et Robust ICA. Nous avons tout d'abord, présenté le principe de la méthode Infomax de maximisation de l'entropie qui s'est avérée une solution performante au problème de séparation de source aveugle. La méthode Infomax est mise en œuvre avec plusieurs itérations d'un algorithme du Gradient Ascent et qui tel qu'implémenté était assez réussi pour extraire des signaux à partir de leurs mélanges.

Robust ICA peut traiter des sources audio et ne nécessite pas de pré-blanchiment. En conséquence, la méthode est plus tolérante que les techniques basées sur le blanchiment aux corrélations de sources résiduelles susceptibles d'apparaître dans de courts enregistrements de données. En outre, l'approche optimale confère à la méthode une robustesse accrue à l'initialisation et aux points de selle, en particulier dans les petites fenêtres d'observation. La complexité des calculs requise pour atteindre une qualité d'extraction de source donnée est faible et la vitesse de convergence est élevée.

Bibliographies

- [II.1] Belouchrani, Adel, et al. A blind source separation technique using second-order statistics. *IEEE Transactions on signal processing*, 1997, 45.2: 434-444.
- [II.2] Cardoso, Jean-François; Souloumlac, Antoine. Jacobi angles for simultaneous diagonalization. *SIAM journal on matrix analysis and applications*, 1996, 17.1: 161-164.
- [II.3] Hyvarinen, Aapo. Fast ICA for noisy data using Gaussian moments. In: *ISCAS'99. Proceedings of the 1999 IEEE International Symposium on Circuits and Systems VLSI* (Cat. No. 99CH36349). IEEE, 1999. p. 57-61.
- [II.4] Hyvarinen, Aapo; Oja, Erkki. Independent component analysis: algorithms and applications. *Neural networks*, 2000, 13.4-5: 411-430.
- [II.5] A.J. Bell and T.J. Sejnowski, "An information maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, pp. 1129 – 1159, 1995.
- [II.6] James V. Stone, *Independent Component Analysis: A Tutorial Introduction*, Massachusetts Institute of Technology, Massachusetts, 2004.
- [II.7] James Stewart, *Calculus: Early Transcendentals*, 5th Edition, pp. 218, 250-252, Thomson Learning Inc., California, 2003.
- [II.8] James V. Stone, *Independent Component Analysis: A Tutorial Introduction*, Massachusetts Institute of Technology, Massachusetts, 2004.
- [II.9] Comon, P.: Independent component analysis, a new concept? *Signal Processing* 36(3), 287–314 (1994).
- [II.10] Cardoso, J.F., Souloumiac, A.: Blind beamforming for non-Gaussian signals. *IEE Proceedings-F* 140(6), 362–370 (1993).
- [II.11] Delfosse, N., Loubaton, P.: Adaptive blind separation of independent sources: a deflation approach. *Signal Processing* 45(1), 59–83 (1995).
- [II.12] Bingham, E., Hyvärinen, A.: A fast fixed-point algorithm for independent component analysis of complex valued signals. *International Journal of Neural Systems* 10(1), 1–8 (2000).
- [II.13] Ristaniemi, T., Joutsensalo, J.: Advanced ICA-based receivers for block fading DS-CDMA channels. *Signal Processing* 82(3), 417–431 (2002).
- [II.14] Hyvarinen, A., Oja, E.: A fast fixed-point algorithm for independent component analysis. *Neural Computation* 9(7), 1483–1492 (1997).
- [II.15] Hyvarinen, A., Karhunen, J., Oja, E.: *Independent Component Analysis*. John Wiley & Sons, New York (2001).
- [II.16] Hyvarinen, A.: Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks* 10(3), 626–634 (1999).
- [II.17] Zarzoso V, Comon P (2010) Robust independent component analysis by iterative maximization of the kurtosis contrast with algebraic optimal step size. *IEEE Trans Neural Network* 21(2) :248–261.

Chapitre II Méthodes de Séparation Aveugle de Sources Appliquées aux Signaux Audio

[II.18] Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes in C. The Art of Scientific Computing, 2nd end. Cambridge University Press, Cambridge

CHAPITRE III

Algorithmes Evolutionnaires et Optimisation

III.1 Introduction

L'approche fondamentale d'optimisation consiste à formuler une fonction de coût qui résume la performance ou la valeur d'une décision et l'améliore de manière itérative en sélectionnant parmi les alternatives disponibles. La plupart des méthodes d'optimisation classiques génèrent une séquence déterministe de solutions d'essai basée sur le gradient ou les statistiques d'ordre supérieur de la fonction de coût [III.1]. Dans les conditions de régularité de cette fonction, il est possible de montrer que ces techniques génèrent des séquences qui convergent asymptotiquement vers des solutions localement optimales, et dans certains cas, elles convergent de manière exponentielle rapide [III.2]. Des variations de ces procédures sont souvent appliquées à la formation de réseaux de neurones (rétro propagation) [III.3], [III.4] ou à l'estimation de paramètres dans des applications d'identification de systèmes et de commandes adaptatives (méthodes d'erreur de prédiction récursive, Newton-Gauss) [III.2], [III.5]. Mais ces méthodes ne fonctionnent souvent pas correctement lorsque des perturbations aléatoires sont imposées à la fonction de coût. De plus, les solutions localement optimales s'avèrent souvent insuffisantes pour résoudre les problèmes d'ingénierie du monde réel.

Le développement et l'application de techniques de recherche basés sur les principes de l'évolution naturelle constituent un domaine important de la recherche actuelle. Il est basé sur la notion de la sélection naturelle développé par 'Charles Darwin' dans 'the Origin of Species'. Les algorithmes évolutifs sont caractérisés par l'existence d'une population constituée d'individus exposés à la pression de l'environnement, ce qui conduit à la sélection naturelle, c'est-à-dire à la survie du plus apte, et donc à l'accroissement de l'aptitude moyenne de la population. La condition physique est la mesure du degré d'adaptation d'un organisme à son environnement. Plus la forme physique est grande, plus l'organisme est en forme et adapté à l'environnement. En général, les algorithmes évolutifs se concentrent uniquement sur un sous-ensemble de mécanismes définis sur le processus évolutif biologique [III.6]. Dans les années 1950 et 1960, plusieurs informaticiens ont indépendamment étudié les systèmes évolutifs dans l'idée que l'évolution pourrait être utilisée comme un outil d'optimisation des problèmes d'ingénierie. L'idée dans tous ces systèmes était de développer une population de solutions candidates à un problème donné, en utilisant des opérateurs inspirés par la variation génétique naturelle et la sélection naturelle. Enchenberg a introduit "les stratégies d'évolution", méthode qu'il utilisait pour optimiser les paramètres à valeurs réelles pour des dispositifs tels que les pales aérodynamiques [III.7].

Les algorithmes évolutionnaires qu'on appelle aussi les algorithmes d'optimisation stochastiques se basent sur les principes de l'évolution naturelle [III.8]. Quelques problèmes d'optimisation ne peuvent être résolus ni de façon exacte (on n'en connaît pas de solution mathématique), ni de manière optimale. Ce sont des problèmes « difficiles », pour lesquels on va concevoir des heuristiques de résolution.

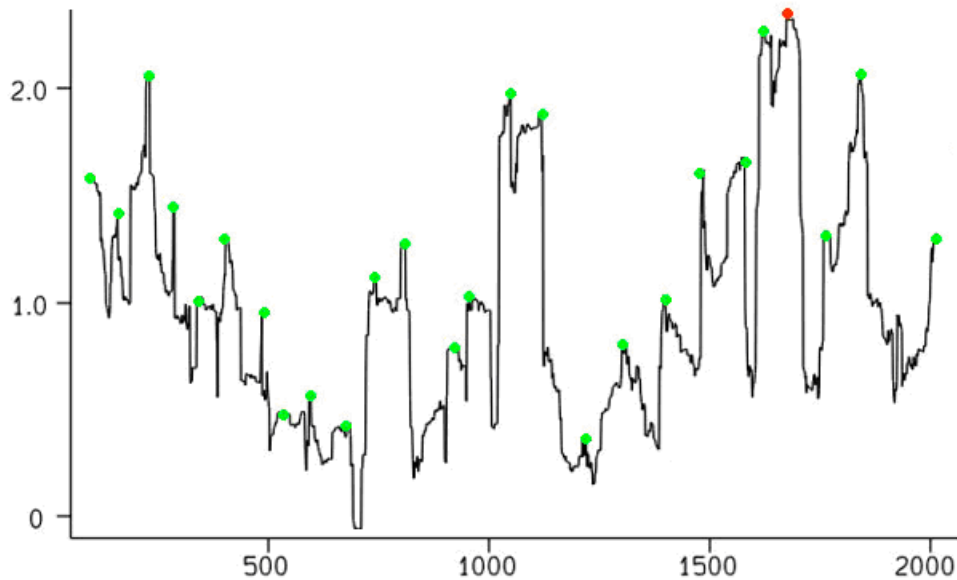


Figure III.1. Exemple de recherche d'une fonction d'une seule variable, avec un seul extremum global (rouge) et plusieurs extrema locaux (vert)

Dans la figure III.1 représente la répartition des extrema de f parmi l'ensemble des valeurs possibles, pour évaluer la difficulté du problème. Il peut exister un seul extremum global, ou plusieurs, ou encore plusieurs extrema locaux proches ou éloignés, répartis uniformément ou pas, denses ou pas. On appelle intensification toute démarche visant à diriger l'effort de recherche vers les meilleures solutions, et diversification toute démarche permettant de découvrir de nouvelles zones contenant éventuellement de meilleures solutions. Le bon équilibre entre ces deux notions dépend du problème à résoudre, et des réglages de l'algorithme utilisé. On ne connaît toujours pas de méthode universelle permettant de trouver les extrema globaux d'une fonction arbitraire. Mais, parmi les heuristiques, certaines peuvent optimiser une large gamme de problèmes différents pour lesquels on ne connaît pas de méthode classique plus efficace, sans nécessiter de changements profonds dans l'algorithme employé : ce sont les méta-heuristiques. Elles utilisent un haut niveau de concept, dont l'intérêt majeur est la facilité d'utilisation dans des problèmes concrets. De plus, elles sont efficaces pour atteindre un optimum avec une précision acceptable, dans un temps raisonnable.

La majorité des problèmes du monde réel nécessitent l'optimisation simultanée de plusieurs objectifs dépendants les uns des autres. Alors qu'en optimisation à objectif unique, la solution optimale est clairement définie. Nombreux papiers abordent les limites des méthodes classiques et mettent en œuvre également plusieurs outils heuristiques modernes qui ont évolué au cours des deux dernières décennies et visant à résoudre le problème d'optimisation des coefficients de matrice de séparation W , tels que le calcul évolutif, l'algorithme génétique (GA : Génétique algorithm), la recherche tabou (TS : Tabu Search), l'optimisation (BBO : Biogeography-based -optimization), l'optimisation par essaim de particules (PSO : Particle Swarm Optimization), la méthode d'ascension de collines (HC : Hill Climbing), le recuit simulé (SA : Simulated Annealing), etc. Ces outils facilitent la résolution des problèmes d'optimisation qui étaient auparavant difficiles ou impossibles à résoudre.

III.2 Terminologie et notations

On cherche à optimiser une fonction f à valeurs réelles définie sur un espace de recherche Ω (Figure III.1).

- La fonction objectif f est appelée fonction fitness.
- Les points de l'espace de recherche Ω sont appelés des individus.
- Les tuples d'individus sont appelés des populations.
- On parle d'une génération pour la boucle principale de l'algorithme.
- On notera Π_i la population de taille fixe P à la génération i .

III.2.1 Schéma général d'un algorithme évolutionnaire

La majorité des algorithmes évolutionnaires (AE) reposent sur une représentation darwinienne relativement simpliste et une optimisation stochastique résumées dans le diagramme de la figure III.2. L'algorithme fait évoluer une population de solutions Π . Cette évolution résulte d'une part d'un darwinisme artificiel, qui se manifeste par la sélection et le remplacement et ne dépend que de la fonction de fitness f , d'autre part de l'effet du hasard, qui s'exprime dans l'initialisation et les opérateurs de variation, et ne dépend que de la représentation de l'espace de recherche.

L'idée est que la sélection favorise les individus qui optimisent la fonction d'objectif et que les variations font apparaître dans la population sélectionnée des individus que l'on peut souhaiter

meilleurs vis-à-vis de la fonction d'objectif. Dans cette évolution, les générations successives de la population restent à une taille constante et la forme stochastique ne dépend que de la génération précédente. Les principes darwiniens sont implémentés dans l'algorithme de la manière suivante :

- Initialisation de la population Π_0 en choisissant P individus dans Ω , généralement par tirage aléatoire avec une probabilité uniforme sur Ω .
- Évaluation des individus de Π_0 (calcul des valeurs de f pour tous les individus).
- La génération i construit la population Π_i à partir de la population Π_{i-1} .
- Sélection des individus les plus performants de Π_{i-1} au sens de f (les plus adaptés se reproduisent).
- Application (avec une probabilité donnée) des opérateurs de variation aux parents sélectionnés, ce qui génère de nouveaux individus, les enfants. On parle de mutation pour les opérateurs unaires, et de croisement pour les opérateurs binaires (ou n-aires). A noter que cette étape est toujours stochastique.
- Evaluation des enfants.
- Remplacement de la population Π_{i-1} par une nouvelle population créée à partir des enfants et/ou des vieux parents de la population Π_{i-1} au moyen d'une sélection darwinienne (les plus adaptés survivent).
- L'évolution stoppe quand le niveau de performance souhaité est atteint.

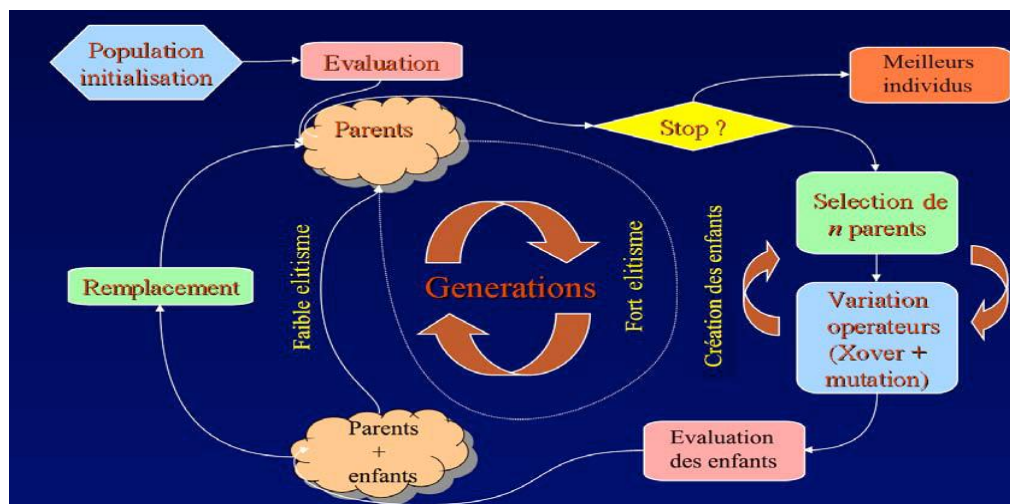


Figure III. 2. Schéma général d'un algorithme évolutionnaire [III.9]

III.3 La recherche tabou

Dans la vie actuelle, la technologie et l'information augmentent rapidement, ce qui complique la recherche d'une solution optimale. Pour trouver une solution à un problème

discret avec la présence de contraintes ou de variables de décision on opte pour une optimisation combinatoire. Dans le but de résoudre ce problème, le concept méta-heuristique, appelé algorithme tabou (Tabu Search :TS), a été introduit initialement par Glover [III.10]. C'est l'une des heuristiques les plus efficaces pour trouver des solutions de qualité dans des délais relativement courts. La principale caractéristique de la recherche tabou est basée sur l'utilisation d'un mécanisme qui s'inspire de la mémoire humaine [III.11]. Au cours des dernières années, plus d'une centaine d'articles présentant les applications de la recherche tabou ont été faites. Dans plusieurs cas, la TS apporte des solutions très proches de l'optimalité et permet de résoudre les problèmes difficiles qui se posent. Ces succès ont rendu la TS extrêmement populaire à ceux qui souhaitent trouver de bonnes solutions aux grands problèmes de combinaison rencontrés dans du pratique. Ainsi la TS est une méthode d'optimisation mathématique, appartenant à la catégorie des techniques recherche locale utilisée pour résoudre des problèmes complexes de très grande taille [III.12], [III.13], [III.14]. La TS améliore les performances d'une méthode de recherche locale. Cette technique utilise une mémoire ou plusieurs qui sont mises à jour et exploitées au cours de la recherche. Une fois qu'une solution est déterminée, elle est marquée comme taboue. Pour examiner les régions de l'espace de recherche qui seraient laissées inexploité par la procédure de recherche locale, la TS modifie la structure du voisinage de chaque solution au fur et à mesure que la recherche continue. Les solutions obtenues sont déterminées par l'utilisation de structure de mémoire. Au cours de la recherche, ces solutions visitées sont mises dans une liste appelée liste taboue qui est une mémoire à court terme. Cette liste de longueur finie k mémorise les k dernières solutions rencontrées.

III.3.1 Concepts de base de la recherche tabou

- Utilisation d'une structure de mémoires souples pour mémoriser les régions visitées et inclure des mécanismes permettant d'interdire à l'algorithme certains mouvements pour ne pas retourner trop rapidement vers les régions visitées.
- Critère d'aspiration (souplesse du mécanisme de la liste taboue) : autorisation et acceptation de certains mouvements tabous.
- Stratégie d'intensification : exploitation plus approfondie des régions prometteuses visitées récemment et renforcement de la recherche dans ces régions.
- Stratégie de transformation : exploration des régions qui n'ont pas encore été visitées.

Algorithme de base de la recherche tabou

Tableau III.1 Algorithme de la recherche tabou (TS).

Etape 1	Initialisation : générer une configuration initiale x_0 dans S . Poser $x = x_0$ et $f = f(x_0)$
Etape 2	$k = 0$. Liste taboue $T = \emptyset$.
Etape 3	Répéter - Incrémenter $k = k + 1$ et générer un sous-ensemble V de solutions de telle sorte que la condition taboue ne soit pas violée - Choisir (en tenant compte des restrictions tabou et du critère d'aspiration) x le meilleur élément dans V ($\text{best } V(x)$, par rapport à la fonction f) - $x_{k+1} = \text{best}(V(x))$ - Si $(f(x_{k+1}) < f(x))$ alors $x = x_{k+1}$ - Mettre à jour la liste taboue T - Stop, si la condition d'arrêt est atteinte (éventuellement appliquer les stratégies d'intensification et de diversification). Sinon, aller à l'étape 3.
Etape 4	- Retourner x la meilleure solution trouvée.

III.4 Algorithme d'optimisation à base de la Biogéographie (BBO)

L'algorithme de la BBO (Biogeography-based Optimization) a été développé en 2008 [III.15]. La stratégie de migration de BBO est similaire à l'approche de recombinaison globale de l'algorithme génétique [III.16] et les stratégies évolutives dans lesquelles les parents peuvent contribuer à une nouvelle génération. La BBO est la nouvelle approche de la résolution de problèmes et partage certaines caractéristiques avec d'autres algorithmes tout comme les algorithmes GA et PSO, la BBO dispose d'un moyen de partager des informations entre ses solutions. La théorie de la biogéographie est l'étude de la migration, de la spéciation et de l'extinction d'espèces. Les modèles de la BBO décrivent comment une espèce migre d'une île (habitat) à une autre et comment naissent de nouvelles espèces et s'éteignent. L'algorithme d'optimisation BBO a été généralisé pour résoudre des problèmes d'optimisation [III.15].

III.4.1 Principe de fonctionnement de la BBO

La BBO, est un algorithme basé sur une population d'individus appelés île (ou habitats). Chaque île représente une solution possible au problème à résoudre. La fonction d'objectif de chaque île est déterminée par son HSI (Habitat Suitability Index) qui est une mesure de la qualité d'une solution candidate. Chaque île est aussi représentée par des variables d'indice de qualité SIVs (Suitability Index Variables). Un HSI élevé d'une île signifie une bonne performance sur le problème d'optimisation, et un faible HSI signifie une mauvaise performance pour un problème d'optimisation [III.15].

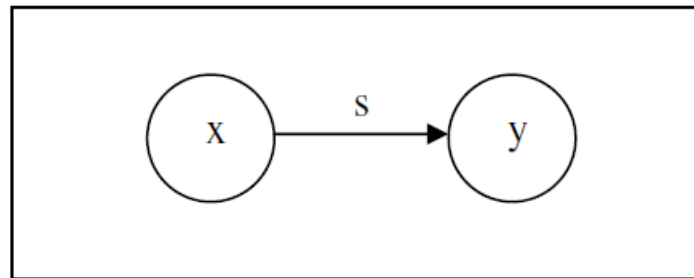


Figure III.3. Migration de la solution de l'espèce

D'après la figure III.3, on peut dire qu'un individu (solution) de l'espèce a émigré de x et immigré à y . Quand s a immigré à y , il partagera des informations probabilistes de x basées sur des valeurs de fitness avec y . Le fonctionnement de la BBO est fondé sur la migration et la mutation. La population initiale représente l'espace de recherche, elle est générée par tirage aléatoire. L'évaluation de la population initiale engendre la migration de certains individus et les descendants vont être mutés. La migration permet de créer un nouvel ensemble d'individus et la mutation permet de fixer la proportion de la population qui sera renouvelée à chaque génération. Les meilleurs individus trouvés sont conservés par l'élitisme (sélection). Les nouveaux descendants remplacent les parents afin de former une nouvelle population. La caractéristique de solution d'immigration reste dans l'individu émigré [III.15].

III.4. 2 La Migration

Le principe de la migration consiste à remplacer une ancienne caractéristique d'une solution chez un individu par une nouvelle caractéristique d'un autre individu. L'opérateur de migration, inclut l'immigration et l'émigration. L'immigration consiste au déplacement des vecteurs de solutions vers d'autres îles (habitats) et l'émigration est la réception des vecteurs de solutions par d'autres îles (habitats).

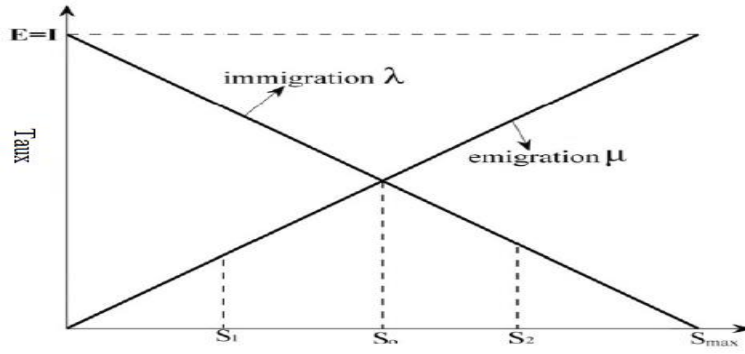


Figure III. 4. Processus d'immigration et d'émigration dans un habitat [III.16].

La figure III.4 montre le modèle de migration linéaire simple pour le cas $E = I$. Selon la figure 1, les habitats à valeur HSI élevée ont tendance à comporter un grand nombre d'espèces, tandis que ceux à faible indice HSI comptent un petit nombre d'espèces [III.15]. D'après la figure 1, on peut conclure que l'habitat comptant peu d'espèces (solution médiocre, HSI faible), comme S_1 , présente un faible taux d'émigration et un taux d'immigration élevé. Cela signifie que l'habitat avec un HSI faible a tendance à prendre des informations sur les bons habitats avec une probabilité élevée, alors que la probabilité de partager ses informations pour d'autres habitats est relativement faible. D'autre part, l'habitat qui contient plus d'espèces (bonne solution, HSI élevé) comme S_2 , présente un faible taux d'immigration et un taux d'émigration élevé [III.16]. Quand le nombre d'espèces dans l'île est maximal $S = S_{max}$ (capacité maximale de l'île), on remarque aussi que le taux d'immigration et les taux d'émigration sont égaux à un point qui est le nombre d'espèces d'équilibre (S_0). Le taux d'immigration et le taux d'émigration quand il y a S espèces sur l'île (λ_S, μ_S . Respectivement) sont donnés par les équations suivantes

$$\lambda_S = I \left(\frac{1-S}{S_{max}} \right) \quad (III.1)$$

$$\mu_S = E \left(\frac{S}{S_{max}} \right) \quad (III.2)$$

III.4. 3 La mutation

L'opérateur de mutation est fréquemment utilisé dans les Algorithmes Evolutionnaires. La mutation est le changement brusque et radical fait au HSI de tout habitat due à certains événements catastrophiques. Elle permet de maintenir la diversité au sein de la population. Le taux de mutation $m(S)$ est inversement proportionnel à P_S qui est la probabilité que l'habitat contient exactement S espèces. Le nombre des espèces change pendant l'intervalle de temps $[t; t + \Delta_t + t[$ [3.15].

$$m(S) = m_{max} \left(1 - \frac{P_S}{P_{max}}\right) \quad (\text{III.3})$$

où m_{max} est un paramètre défini par l'utilisateur qui représente le taux de mutation maximale et P_{max} est la probabilité maximale. Si une île est sélectionnée pour la mutation, alors un SIV choisi au hasard dans l'île est simplement remplacé par une variable aléatoire générée dans son intervalle de définition.

III.4. 4 L'Élitisme

A chaque génération, les meilleurs individus sont conservés par l'élitisme (sélection). Cette sélection empêche la disparition de l'individu le plus performant au cours du remplacement. Ces individus sont réintroduits dans la population.

III.4. 5 L'algorithme de la BBO

L'algorithme de base de BBO est comme suit [III.17] :

Tableau III.2. Algorithme optimisation à base de biogéographie

- 1 : Générer aléatoirement un ensemble de solutions initiales (îles).
- 2 : Tant que le critère d'arrêt n'est pas atteint faire
- 3 : Évaluer la fonction d'objectif (HSI) de chaque solution
- 4 : Calculer le nombre d'espèce S , le taux d'immigration λ et d'émigration μ pour chaque solution
- 5 : Migration :
- 6 : Pour $i = 1$ à N faire
- 7 : Utiliser λ_i pour décider, de manière probabiliste, d'immigrer à X_i
- 8 : Si $\text{rand}(0, 1) < \lambda_i$ alors
- 9 : Pour $j = 1$ à N faire
- 10 : Sélectionner l'île d'émigration X_j avec une probabilité μ_j
- 11 : Si $\text{rand}(0, 1) < \mu_j$ alors
- 12 : Remplacer une variable de décision (SIV) choisie aléatoirement dans X_i par la variable correspondante dans X_j
- 13 : Fin Si
- 14 : Fin Pour
- 15 : Fin Si
- 16 : Fin Pour

17 : Fin Tant que

18 : Mutation : muter les individus au taux de mutation donné par l'équation (III.3)

19 : Remplacement de la population par les descendants

20 : Implémenter l'élitisme

Retourner la meilleure solution trouvée

Le concept général de la migration est l'échange de caractéristiques entre les îles. Les taux d'immigration (λ_S) et d'émigration (μ_S) de chaque île sont utilisés pour transmettre de façon probabiliste, $\text{rand}(0,1)$ est un nombre aléatoire uniformément distribué dans l'intervalle $[0, 1]$ et $X_{i,j}$ est le j – ème SIV de la solution X_j .

Le HSI d'une île peut changer soudainement, en raison d'événements aléatoires : désastres naturels (tempêtes, ouragans, incendies, ...) ou des épidémies, etc. La BBO modélise ce phénomène comme une mutation des SIVs.

III.5 L'algorithme d'ascension de collines (Hill Climbing)

Ascension de collines (Hill Climbing) tire son nom du fait qu'on suppose que l'espace de fonction est "en forme de colline". C'est une technique d'intelligence artificielle qui a pour but de trouver numériquement des solutions des équations diophantiennes (une équation diophantienne est une équation polynomiale à une ou plusieurs inconnues dont les solutions sont cherchées parmi les nombres entiers, éventuellement rationnels, les coefficients étant eux-mêmes également entiers). De telles équations sont importantes car elles ont beaucoup d'applications dans des domaines tels que la cryptographie à clé publique, la factorisation d'entiers, les courbes algébriques, les courbes projectives et la dépendance des données dans les super-ordinateurs. Un fait important : il a été prouvé qu'il n'y a pas une méthode générale pour trouver des solutions de telles équations. La méthode qui utilise une représentation arborescente pour décrire les solutions possibles des équations diophantiennes adopte une nouvelle méthodologie pour générer des successeurs. La fonction heuristique utilisée aide à rendre le processus de recherche de solutions un processus de minimisation [III.18].

L'algorithme du Hill-Climbing est une implémentation de base de la recherche locale. D'un candidat initial généré aléatoirement, l'algorithme remplace de manière itérative le candidat actuel par un meilleur, en recherchant parmi les alternatives locales un voisin. Dans l'algorithme, la solution s garde la trace du meilleur candidat, à tout moment. Si le score de s atteint 0, ce qui signifie qu'il s'agit d'un optima global, l'algorithme s'arrête. Par contre, si

chaque voisin de x a été évalué sans remplacer s en tant que candidat amélioré, l'algorithme s'arrête sans indication globale de l'optima. Lorsqu'il atteint un candidat dont le voisinage ne contient aucune solution améliorée, il renvoie la solution actuelle. Le voisinage d'un candidat actuel est généralement défini comme un ensemble contenant tous les candidats possibles qui diffèrent du candidat actuel avec une petite modification. La modification dépend de la structure des candidats. Pour les vecteurs, le voisinage peut être défini comme tous les vecteurs où une valeur est incrémentée ou décrémentée d'un par rapport au candidat. Cette technique se différencie des méthodes de descente traditionnelles en acceptant également les solutions de qualité égale. Ainsi, cette méthode choisit le premier voisin qui a une qualité meilleure ou équivalente.

L'algorithme de base du Hill Climbing est comme suit :

Tableau III.3. L'Algorithme du Hill Climbing (HC).

Entrée : Une solution candidate x

Sortie : Solution s ou nul si impossible de trouver une solution valide

1 : $s \leftarrow x$

2 : Tant que $\text{score}(s) > 0$ faire

3 : pour chaque $x \in N(x)$ faire

4 : $s \leftarrow \min(\text{score}(x), \text{score}(s))$

5 : Fin pour

6 : Si $s = x$ Alors

7: Return Null

8: Sinon

9 : $x \leftarrow s$

10: End if

11 : Fin Tant que

12 : Résultat $\leftarrow s$

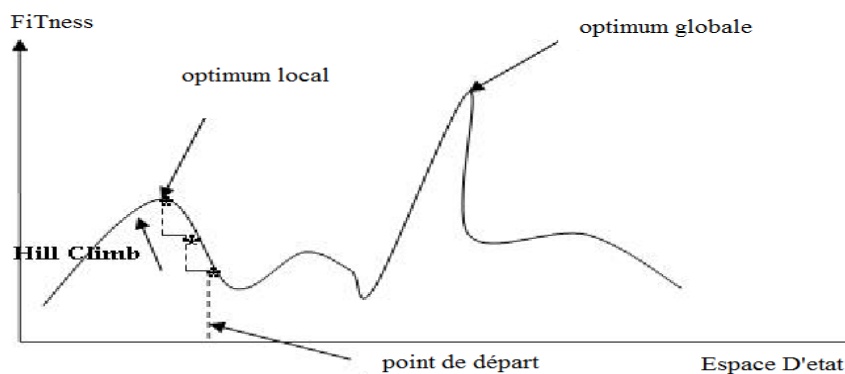


Figure III. 5. Schéma de principe de base de l'algorithme Hill Climbing (HC).

Il y a quelques inconvénients avec cet algorithme métaheuristique. L'algorithme Hill Climbing est idéal pour trouver rapidement les optima locaux, mais échoue souvent à trouver la meilleure solution globale [III.20]. C'est parce qu'il est très vulnérable aux solutions et aux plateaux des optima locaux. Comme son nom l'indique, la procédure monte continuellement vers le sommet d'une colline en faisant un petit pas à la fois. Quand elle n'a pas d'étape suivante qui puisse la mener vers le haut, elle se considère elle-même être sur le dessus. Le problème est qu'il ne voit qu'une étape à la fois en termes d'amélioration du candidat. La figure III.4 illustre ces défis. Comme l'optima global est le sommet le plus élevé, un optima local est un candidat avec un score inférieur à l'optima global, mais tous les candidats dans le score de voisinage des optima locaux sont encore moins élevés, ce qui entraîne l'arrêt de l'algorithme Hill Climbing. Un plateau est une situation où, pour un certain nombre de pas des quartiers des candidats, on choisit le meilleur candidat dans chaque étape qui mène au même résultat. Quand l'algorithme pense avoir atteint le sommet, il pourrait en fait être un moyen de continuer à rechercher une meilleure solution s'il est disposé à poursuivre certaines étapes intermédiaires sans progresser. Compter plus d'étapes, à chaque itération, est une possibilité, mais cela augmente la complexité de la procédure. Le seul avantage du Hill Climbing est que la recherche d'une solution peut prendre plusieurs étapes et rendre ce temps métaheuristique plus efficace. On peut donc améliorer la solution sans trop alourdir la complexité, en exécutant plusieurs fois l'algorithme Hill Climbing avec différents candidats initiaux. Dans cette approche, l'algorithme Hill Climbing se déroule un nombre de fois ou jusqu'à atteindre une limite de temps. En générant au hasard un candidat initial à chaque fois, ceci fournit différents candidats optimaux locaux parmi lesquels l'algorithme choisit la meilleure solution, à moins que l'une des exécutions n'aboutisse à un optima global, qui est immédiatement retourné.

III.6 L'algorithme du Recuit Simulé

Le recuit simulé en anglais « simulated annealing (SA) » a été inventé par S. Kirkpatrick, CD. Gelatt et MP Vecchi en 1983[III.21]. Ce nom vient du recuit en métallurgie, une technique impliquant le chauffage et le refroidissement contrôlé d'un matériau afin d'améliorer ses propriétés structurelles en éliminant les défauts du cristal. Le recuit simulé est une méthode permettant de résoudre des problèmes d'optimisation sans contraintes ni contraintes liées. La méthode modélise le processus physique de chauffage d'un matériau, puis d'abaissement lent de la température pour réduire les défauts, minimisant ainsi l'énergie du système. Le recuit simulé appartient à une classe d'algorithmes de recherche locale qui suivent une stratégie stochastique. Ces algorithmes jouent un rôle particulier dans la recherche locale pour deux raisons. Premièrement, ils semblent avoir beaucoup de succès lorsqu'ils sont appliqués à une large gamme de problèmes pratiques [III.21]. La méthode d'optimisation SA provient d'un schéma d'échantillonnage conçu par Metropolis et al. [III.22] pour effectuer une intégration Monte Carlo de plusieurs centaines d'intégrales dimensionnelles nécessaires pour évaluer les propriétés d'équilibre d'une collection de molécules à une température donnée T . La méthode SA peut être classée en trois catégories, recuit simulé classique CSA (classical SA), recuit simulé rapide FSA (Fast SA), et le recuit simulé généralisée GSA (Generalized SA). Le recuit simulé FSA est une recherche semi-locale et consiste en de longs sauts occasionnels. L'algorithme FSA est inversement linéaire dans le temps et est rapide par rapport au recuit simulé classique (CSA) dont les fonctions de coût de processus possèdent des degrés tout à fait arbitraires de non-linéarités, de discontinuités et aléatoire. Il traite des conditions aux limites et des contraintes assez arbitraires imposées à ces fonctions de coût qui sont implémentés assez facilement avec un degré de codage assez minime par rapport aux autres algorithmes d'optimisation non linéaire, ce qui garantit statistiquement la recherche d'une solution optimale [III.23]. L'algorithme GSA ou machine de Tsallis est un processus stochastique markovien, basé sur les statistiques de Tsallis qui a été utilisé dans de nombreuses classes de problèmes, comme la physique et la chimie et en particulier dans l'optimisation des systèmes moléculaires et les problèmes de repliement des protéines [III.24] [III.25]. La famille des algorithmes de recuit simulé dépend d'une fonction de visite qui détermine comment le domaine de la fonction est recherché, et d'une fonction d'acceptation qui dit si un résultat d'énergie plus élevée devait être accepté ou rejeté. Dans l'algorithme de Métropolis, on part d'une forme donnée et on fait subir au système une modification élémentaire. Si cette perturbation a pour effet de diminuer la fonction objectif (ou énergie) du système, elle est acceptée. Sinon, elle est acceptée avec une

probabilité P . En répétant ce processus, on génère une séquence de formes qui tendent vers l'équilibre thermodynamique. L'algorithme de recuit simulé est différent de l'algorithme Hill Climbing dans la décision de remplacer la solution S , par une solution meilleure. Plus précisément dans l'algorithme de recuit simulé, si R est meilleure que S , nous remplaçons la solution S par R comme d'habitude. Mais si R est pire que S , nous pouvons encore remplacer S avec R avec une probabilité, $P(t, R, S)$, avec

$$P(t, R, S) = e^{(f(R)-f(S))/t} \quad (\text{III.4})$$

L'algorithme du Recuit Simulé est comme suit :

Tableau III. 4. L'algorithme d'optimisation Recuit Simulé

```

1 :  $t \leftarrow t_0$  ;  $t_0$  est la température initialisée à une grande valeur
2 :  $s \leftarrow s_0$  ;  $s_0$  une solution initiale
3 :  $best \leftarrow s$ 
4 : Tant que ( $best$  est la meilleure solution) & ( $Number\text{-}Iter = \max$ ) & ( $t \leq 0$ ) Faire
5 :  $r \leftarrow s'$  où  $s' \in N(s)$ 
6 :  $\rho \leftarrow random(0,1)$  ;
7 : Si ( $f(s) < f(best)$ ) ou ( $\rho < e^{(f(r)-f(s))/t}$ ) alors
8 :  $s \leftarrow r$  ;
9 : Si ( $f(s) - f(best) > 0$ )
10 :  $best \leftarrow s$  ;
11 : décrémenter  $t$  ;
12 : Fin Tant que
```

III.6.1 Interprétation

Par exemple, nous avons un paramètre température t , que l'on veut ajuster. Si la température t est proche de 0, la fraction $f(r) - f(s)/t$ devient un grand nombre, alors la probabilité est proche de 0. Si t est très élevé, la probabilité est proche de 1. L'idée est d'initialiser d'abord la température à un nombre élevé, ce qui donne à l'algorithme une exploration de l'espace de solutions très aléatoire. Ensuite, diminuer la température t d'une manière lente, jusqu'à ce que la température converge à zéro. Dans ce cas l'algorithme de recuit simulé rejoint l'algorithme Hill Climbing de base.

III.7 Les algorithmes génétiques

Les algorithmes génétiques (AG) ont été inventés par John Holland dans les années 1960 et développés par Holland et ses étudiants et collègues à l'Université du Michigan dans les années 1960 et 1970. En contraste avec les stratégies d'évolution et de programmation évolutive, l'objectif initial de Holland n'était pas de concevoir des algorithmes pour résoudre des problèmes spécifiques, mais plutôt d'étudier formellement le phénomène d'adaptation tel qu'il se produit dans la nature et développer des moyens permettant d'importer les mécanismes d'adaptation naturelle dans les systèmes informatiques.

Holland a présenté l'algorithme génétique comme une abstraction de l'évolution biologique. L'AG est une méthode permettant de passer d'une population de "chromosomes" (par exemple, des chaînes d'uns et de zéros, ou "bits") à une nouvelle population en utilisant une sorte de "sélection naturelle" en association avec les opérateurs de croisement, mutation et inversion. Chaque chromosome est constitué de "gènes" (par exemple, des bits), chaque gène étant un exemple d'un "allèle" particulier (par exemple, 0 ou 1). L'opérateur de sélection choisit ces chromosomes dans la population qui sera autorisée à se reproduire et, en moyenne, les chromosomes en ajustement produisent plus de descendance que les moins en forme [III.26]. Au cours des dernières années, il y a eu une interaction généralisée entre les chercheurs étudiant diverses méthodes de calcul évolutionnaire, les frontières entre les AG, les stratégies d'évolution, la programmation évolutionnaire et d'autres approches évolutives. Aujourd'hui, les chercheurs utilisent souvent le terme "algorithme génétique" pour décrire quelque chose de très loin de la conception originale de Hollande. Dans les algorithmes génétiques, le terme chromosome fait généralement référence à une solution potentielle à un problème, souvent codée sous forme de chaîne de bits. Les "gènes" sont soit des bits simples, soit des blocs courts de bits adjacents qui codent un élément particulier de la solution candidate (par exemple, dans le contexte de l'optimisation de la fonction multi-paramètres, les bits codant un paramètre particulier peuvent être considérés comme un gène). Un allèle dans une chaîne de bits vaut 0 ou 1. Pour les grands alphabets, il est possible d'avoir plus d'allèles à chaque locus. Le croisement consiste généralement en un échange de données génétiques entre deux parents simples de chromosomes. La mutation consiste à retourner le bit de manière aléatoire. De plus, l'AG apporte de légères modifications à ses solutions jusqu'à obtenir la meilleure solution. Il se compose de plusieurs solutions appelées chromosomes ou individus. Cet algorithme est un puissant outil de recherche, en particulier lorsqu'il est appliqué à des problèmes d'optimisation combinatoire. Toutefois, la mise en œuvre d'un AG efficace est souvent confrontée à deux

problèmes majeurs, d'un côté, la convergence prématurée vers les optima locaux et de l'autre côté, les exigences relatives à la recherche de longues périodes afin de parvenir à un optimum ou à une bonne solution sous-optimale. La figure III.6 donne le principe des algorithmes génétiques.

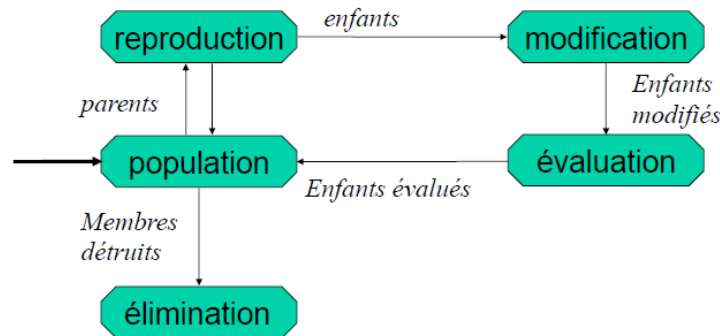


Figure III.6. Principe des algorithmes génétiques

Les algorithmes génétiques fournissent des solutions aux problèmes n'ayant pas de solutions calculables en temps raisonnable de façon analytique ou algorithmique. L'algorithme génétique fait évoluer sur plusieurs générations, une grande population en appliquant trois opérateurs, qui se composent de sélection, de croisement et de mutation. Ceci est essentiel pour produire plus de descendants qui tendent à homogénéiser la population et à améliorer le résultat moyen au fur et à mesure que l'algorithme progresse ou évolue. L'algorithme commence avec une population initiale représentant des solutions candidates aléatoires. Des milliers de solutions plus ou moins bonnes sont créées au hasard puis sont soumises à un procédé d'évaluation de la pertinence de la solution mimant l'évolution des espèces : les plus "adaptées", c'est-à-dire les solutions au problème qui sont les plus optimales survivent davantage que celles qui le sont moins et la population évolue par générations successives en croisant les meilleures solutions entre elles et en les faisant muter, puis en relançant ce procédé un certain nombre de fois afin d'essayer de tendre vers la solution optimale. Les chromosomes sélectionnés sont utilisés pour créer une nouvelle population sur la base de deux principaux opérateurs génétiques, croisement et mutation. Le croisement permet la création de nouveaux individus selon un processus fort simple : Il autorise l'échange d'information entre les chromosomes (individus). La mutation aléatoire est également appliquée pour ajouter un peu de distinction à la population. Les descendants ayant les meilleures performances sont susceptibles d'être les parents de la prochaine génération. Le processus est répété jusqu'à atteindre un critère d'arrêt. Dans la mise en œuvre d'un AG, les paramètres à optimiser (gènes) doivent être codés dans une

chaîne de sorte que le croisement et la mutation peuvent être appliqués. Le codage est un processus de représentation des gènes. Le processus peut être effectué par utilisation de : bits, nombres, arbres, tableaux, listes ou tous autres objets. La littérature définit deux types de codage : binaire et réel.

- **Le codage binaire**

Soit f une fonction de paramètre x à optimiser. La variable x représente un individu de la population et est codée sous forme d'une chaîne de n bits. Soit $x \in [x_{min}, x_{max}]$ avec $x \in R$ et x a un nombre de décimales noté d .

Dans une représentation binaire, la taille de l'individu n vérifie l'inéquation suivante :

$$|x_{max}, x_{min}| * 10^d \leq 2^n \quad (\text{III.5})$$

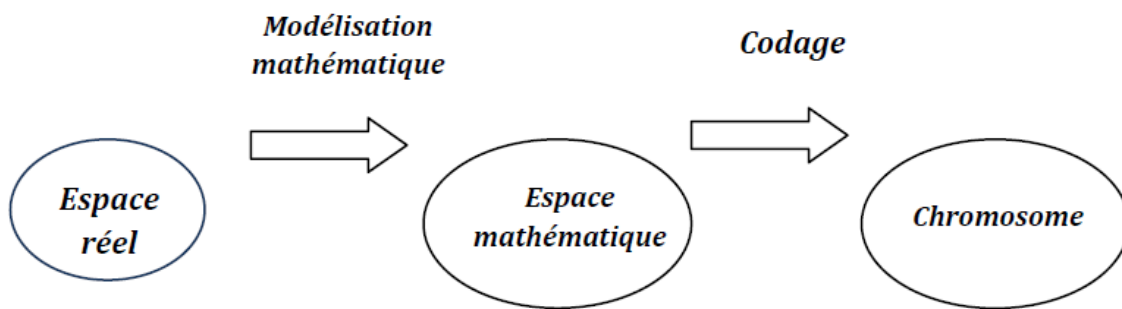


Figure III. 7. Les phases de définition d'un codage

- **Le codage réel**

Dans les algorithmes génétiques, la représentation des solutions n'est pas obligatoirement restreinte à un alphabet (0,1). Il existe une représentation plus efficace qui s'appuie sur des nombres réels. Un individu est alors représenté sous la forme d'un vecteur à N dimensions de nombres réels [III.27] :

. Ce type de codage présente certains avantages par rapport au codage binaire :

- Le codage réel est puissant pour les problèmes considérés comme difficiles pour le codage binaire.
- Ce codage requiert un ajustement des opérateurs de croisement et mutation.

Chaque cycle de l'algorithme génétique est conduit par le processus d'évolution et à chaque itération, appelée génération, est créée une nouvelle population avec le même nombre

d'individus à partir de la population précédente (parents). Cette nouvelle génération consiste généralement en des individus mieux adaptés à l'environnement tel qu'il est représenté par la fonction fitness. Au fur et à mesure des générations, les individus vont tendre, en général, vers l'optimum de la fonction d'objectif.

III.7.1 Implantation de l'algorithme génétique

L'algorithme génétique de base, bien qu'il puisse être implanté en premier lieu comme dans tous les problèmes d'optimisation, il faut bien définir le problème à optimiser. Il faut bien définir la quantité à optimiser, et aussi savoir quels paramètres on veut déterminer, puis définir le nombre de paramètres à déterminer et les bornes de ces paramètres.

III.7.1 .1 La population initiale

La première étape dans le fonctionnement d'un AG est donc la génération d'une population initiale. Cette population code une solution possible à un problème. Après avoir créé la population initiale, chaque individu est évalué et affecté à une valeur de fitness en fonction de la fonction d'objectif. Le problème principal est celui de choix de la taille initiale optimale de la population [III.28] qui ne peut pas en général être appliquée à chaque type de problème ou de fonction à évaluer [III.29]. Il faut remarquer qu'une taille de population trop petite conduit à l'obtention d'un optimum local.

III.7.1 .2 La fonction d'objectif

Dans les algorithmes génétiques, chaque solution est généralement représentée par une chaîne de nombres binaires, appelée chromosomes. Nous devons tester ces solutions et trouver le meilleur ensemble de solutions pour résoudre un problème donné [III.30].

La fonction d'objectif, ou fitness, associe une valeur pour chaque individu. Cette valeur a pour but d'estimer si un individu est mieux adapté qu'un autre à son environnement. Ce qui signifie qu'elle quantifie la réponse fournie au problème pour une solution potentielle donnée. Ainsi les individus peuvent être comparés entre eux. Cette fonction, propre au problème, est fréquemment simple à formuler lorsqu'il y a peu de paramètres. Au contraire, lorsqu'il y a beaucoup de paramètres ou lorsqu'ils sont corrélés, elle est plus difficile à définir. Dans ce cas, la fonction devient une somme pondérée de plusieurs fonctions. Une adaptation des coefficients est nécessaire.

III.7.1.3 Le chromosome

Un chromosome (également parfois appelé génome) est un ensemble de paramètres définissant une solution proposée au problème que l'algorithme génétique tente de résoudre.

Les chaînes des systèmes génétiques artificiels sont analogues aux chromosomes des systèmes biologiques. Ils portent les informations génétiques d'un individu. Chaque chromosome binaire dans un AG comprend plusieurs gènes ayant des valeurs binaires 0 et 1, ce qui détermine les attributs pour chaque individu [III.31].

III.7.1.4 La sélection

La sélection est le processus de sélection de deux parents ou plus de la population pour le croisement. Après avoir choisi un codage, l'étape suivante consiste à décider comment effectuer la sélection, c.-à-d. comment choisir des individus dans la population qui créeront une descendance pour la prochaine génération et comment beaucoup de progéniture chacun créera. En biologie, la sélection naturelle est l'un des outils qui cause l'évolution des espèces. Ce mécanisme est particulièrement important du fait qu'il explique l'adaptation des espèces aux milieux des générations. La théorie de la sélection naturelle permet d'expliquer et de comprendre comment l'environnement influe sur l'évolution des espèces et des populations en sélectionnant les individus les plus adaptés. Elle constitue donc un aspect fondamental de la théorie de l'évolution.

La sélection est réalisée sur un ensemble d'individus appelé reproducteur qui est composé :

- Soit de l'ensemble de la population.
- Soit d'un sous ensemble construit par sélection : Seuls les individus les mieux adaptés (dont les évaluations sont les plus fortes) vont « rester », les autres sont éliminés. On fixe généralement $\frac{1}{2}$ la part de la population qui survit. Dans ce cas, le tri de la population selon la fonction objectif est obligatoire.

III.7.1.5 Le croisement

Cet opérateur sélectionne les gènes des chromosomes parents et crée une nouvelle progéniture. La manière la plus simple de procéder consiste à choisir de manière aléatoire un point de croisement et que tout ce qui précède soit copié à partir d'un premier parent, puis tout ce qui suit une copie du point de croisement à partir du deuxième parent [III.32]. Il s'agit d'un processus essentiel pour explorer l'espace des solutions possibles. Le croisement a pour but d'enrichir la diversité de la population en manipulant la structure des chromosomes et l'échange

de certains gènes d'un parent avec ceux de l'autre. Certains opérateurs de croisement bien connus sont détaillés dans [III.33].

III.7.1 .5.1 Le croisement en 1-point

L'un des opérateurs de croisement les plus simples est un croisement en un point. Dans cette méthode, un point de coupure est sélectionné de manière aléatoire. Il consiste à choisir au hasard un point de croisement pour chaque couple de chromosomes. Les sous-chaînes situées après ce point sont par la suite inter-changées pour former les fils, Figure III. 7 [III.34].

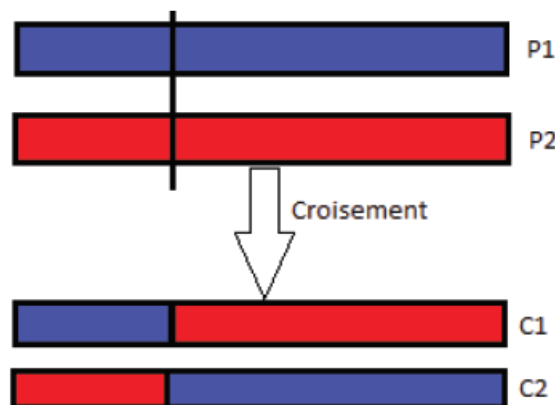


Figure III.8. Croisement en un point de deux chromosomes

III.7.1.5.2 Le croisement en 2 points

Le croisement en deux points est similaire à celui d'un croisement à un point. As ce cas, au lieu d'un point de coupure, deux points de coupure sont choisis au hasard aux mêmes positions chez deux parents et deux enfants sont générés à partir d'eux (Figure III. 9) [III.35].

Parent1	1 1 0 1 0 0 0 1 1 0 1 0
Parent2	1 0 0 1 1 0 0 1 1 0 1 1
Fils 1	1 1 0 1 1 0 0 1 1 0 1 0
Fils 2	1 0 0 1 0 0 0 1 1 0 1 1

Figure III 9. Croisement en 2-points de deux chromosomes.

III.7.1.5 La mutation

La mutation empêche l'algorithme à être piégé dans un minimum local. La mutation est considérée comme un simple opérateur de recherche. Si le croisement est censé exploiter la solution en introduisant de nouvelles structures génétiques dans la population en modifiant au hasard certaines de ses composantes, la mutation aide à sortir du piège des minimas locaux et à maintenir la diversité de la population. Il existe de nombreuses formes de mutation pour les différents types de représentation. Pour une représentation binaire, une simple mutation peut consister à inverser la valeur de chaque gène avec une petite probabilité. Cet opérateur nous garantit que l'algorithme génétique sera susceptible d'atteindre tous les points de l'espace d'état, sans pour autant les parcourir tous dans le processus de la résolution du problème [III.36].

III.7.2 L'hybridation

Les méthodes hybrides permettent de combiner les avantages des deux types de méthodes (déterministes et stochastiques). Elles peuvent être vues comme la solution parfaite vis-à-vis des désavantages des méthodes locales et des méthodes globales. Au fait, les méthodes locales mènent à une solution locale. Tandis que les méthodes globales consomment énormément de temps. Il paraît clair qu'un compromis entre exploitation et exploration doit être trouvé, d'où l'utilité des méthodes hybrides. Pour remédier aux faiblesses de l'algorithme génétique, la solution est de proposer une approche génétique hybride basée sur les meilleurs mécanismes génétiques et aussi d'améliorer la mise en œuvre des algorithmes génétiques. Plusieurs types d'hybridation sont possibles. Par exemple, la méthode hybride des algorithmes génétiques avec la méthode des essaims particulaires (PSO) permet d'accélérer la recherche pour localiser l'optimum global [III.37].

L'algorithme génétique peut être résumé comme suit :

Tableau III. 5.Algorithme de base de l'algorithme génétique

Etape 1	Initialisation : Définir la taille de la population, le taux de mutation et de croisement, le type de sélection ; choisir le critère d'arrêt et le codage à utiliser ; générer aléatoirement la population initiale dans l'espace de recherche.
Etape 2	Evaluer la fonction fitness de chaque chromosome.
Etape 3	Appliquer l'opérateur de croisement.
Etape 4	Appliquer l'opérateur de mutation.

Etape 5	Sélectionner les chromosomes.
Etape 6	Mettre à jour la nouvelle population.
Etape 7	Répéter les étapes 2-6 jusqu'à atteindre le critère d'erreur et retourner la meilleure solution.

III.8 Optimisation par colonie de fourmis

III.8.1 Introduction

L'optimisation des colonies de fourmis est une technique d'optimisation introduite au début des années 90. La source d'inspiration de l'optimisation de la colonie de fourmis est le comportement de recherche de véritables colonies de fourmis. Ce comportement est exploité dans les colonies artificielles de fourmis pour la recherche de solutions approximatives à des problèmes d'optimisation discrets, à des problèmes d'optimisation continus et à des problèmes importants en télécommunications [III.38].

Les colonies de fourmis, et plus généralement les sociétés d'insectes sociales, sont des systèmes distribués qui, en dépit de la simplicité de leurs individus, présentent une organisation sociale très structurée. Grâce à cette organisation, les colonies de fourmis peuvent accomplir des tâches complexes. Le domaine des “ algorithmes de fourmis ” étudie les modèles dérivés de l'observation du comportement des fourmis réelles et utilise ces modèles comme source d'inspiration pour la conception de nouveaux algorithmes pour la solution des problèmes d'optimisation et de contrôle distribué. Les fourmis ne sont pas comme les autres insectes sociaux, que ce soit dans leur comportement social ou dans leur organisation. Elles sont très avancées dans le développement du partage de leurs activités de production "recherche de nourriture (Figure III.10), défense du nid, entretien, resserrement du nid et entretien des larves et de leur approvisionnement en nourriture". Ces activités sont importantes pour le développement et la survie de la colonie et ont un rapport direct avec le temps et l'environnement. Kennedy et d'autres chercheurs dans leur ouvrage "Swarm Intelligence" [III.39] avaient mis en cause cela, sur la base d'une définition antérieure donnée pour décrire les systèmes de simulation d'essaims. Le terme "essaims" est utilisé dans un sens général pour faire référence à toute structure de collection d'agents d'interaction pour des raisons pratiques et évidentes d'étude en laboratoire. Cette diversité exubérante est une face d'inspiration intéressante pour les systèmes informatiques. C'est ainsi que les capacités des fourmis en matière de coopération, de communication, de compétition et d'apprentissage, peuvent être mises à profit pour la conception de robots ou d'algorithmes de résolution de problèmes.

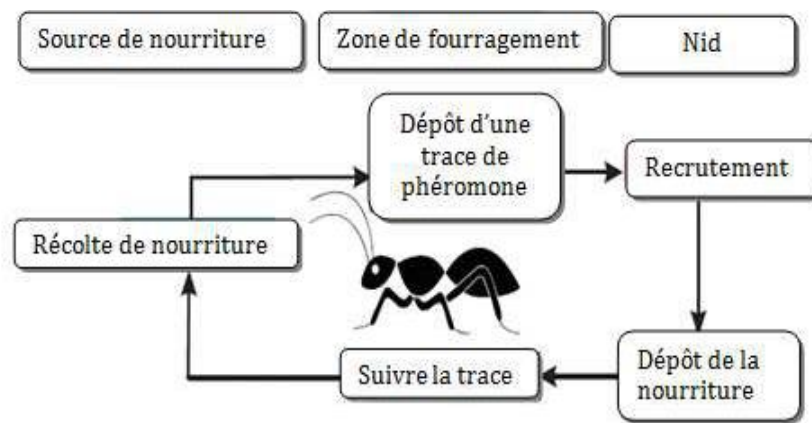


Figure III.10. Comportement d'une fourmi naturelle lors de la recherche de nourriture.

III.8.2 Intelligence Commune des Fourmis

Les fourmis offrent une grande diversité de comportements et de morphologies. L'étude précise de leurs comportements est souvent limitée aux espèces les moins peuplées pour des raisons pratiques. Cette diversité est une face d'inspiration intéressante pour les systèmes informatiques. C'est ainsi que les capacités des fourmis en matière de coopération, de communication, de compétition et d'apprentissage peuvent être mises à profit pour la conception des algorithmes de résolution des problèmes que l'on pourra retrouver dans les systèmes informatiques.

III.8.2.1 Technique d'optimisation

Deneubourg et al. [III.40] ont observé le comportement des fourmis sur la pose de phéromone. Dans une expérience connue sous le nom de "expérience de double pont", le nid d'une colonie de fourmis argentines était relié à une source par deux ponts de longueurs égales (Figure III.10.a). Dans un tel cadre, les fourmis commencent à explorer les environs du nid et finissent par atteindre la source de la nourriture. Le long de leur chemin entre nourriture source et nid, les fourmis argentines procèdent au dépôt de phéromone. Au départ, chaque fourmi choisit au hasard l'un des deux ponts. Cependant, en raison de fluctuations aléatoires, il arrive que l'un des deux ponts présente une concentration de phéromone plus élevée que l'autre et attire donc davantage de fourmis. Cela apporte une quantité supplémentaire de phéromone sur ce pont le rendant plus attrayant avec le résultat qu'après quelque temps, toute la colonie converge vers l'utilisation du même pont. Ce comportement au niveau des colonies, basé sur une autocatalyse, c'est-à-dire sur l'exploitation de la rétroaction positive, peut être utilisé par les fourmis pour trouver le chemin le plus court entre une source de nourriture et leur nid. Goss et

al. [III.41] ont considéré une variante de l'expérience à double pont dans laquelle un pont est significativement plus long que l'autre (figure III.11.b). Dans ce cas, les fluctuations stochastiques du choix initial d'un pont sont beaucoup réduites et un deuxième mécanisme joue un rôle important : les fourmis choisissant par hasard le pont court sont les premières à atteindre le nid. Le pont court reçoit donc la phéromone plus tôt que le long et de ce fait augmente la probabilité que d'autres fourmis le choisissent plutôt que le long. Goss et al. [III.41] ont développé un modèle de comportement observé : en supposant qu'à un moment donné m_1 fourmis ont utilisé le premier pont et m_2 le second, la probabilité p_1 pour une fourmi de choisir le premier pont est :

$$p_1 = \frac{(m_1+k)^h}{(m_1+k)^h + (m_2+k)^h} \quad (\text{III.6})$$

où les paramètres k et h doivent être adaptés aux données, donc $p_2 = 1 - p_1$. Les simulations de Monte Carlo ont montré un très bon ajustement pour $k = 20$ et $h = 2$ [III.41].

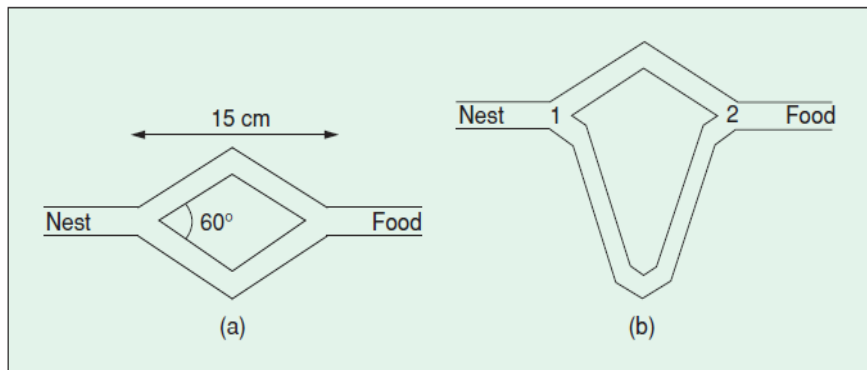


Figure III. 11. Configuration expérimentale pour l'expérience à double pont
(a) Les branches ont des longueurs égales [III.40].
(b) Les branches ont des longueurs différentes [III.41].

Le modèle proposé par Deneubourg et al. pour expliquer le comportement alimentaire des fourmis était la source principale d'inspiration pour le développement de l'optimisation de la colonie de fourmis. Dans l'algorithme ACO (Ant Colony Optimization), un certain nombre de fourmis artificielles élaborent des solutions pour le problème d'optimisation considéré et l'échange des informations sur la qualité de ces solutions via un schéma de communication qui rappelle celui adopté par les vraies fourmis. Différents algorithmes d'optimisation de colonies de fourmis ont été proposés. L'algorithme d'optimisation de colonie de fourmis d'origine est connu sous le nom d'Ant System [III.42] [III.43] et a été proposé au début des années 90. Depuis lors, plusieurs autres algorithmes ACO ont été introduits. Tous les algorithmes d'optimisation des colonies de fourmis partagent la même idée, qui est mieux illustrée par un exemple de la façon dont les algorithmes ACO peuvent être appliqués.

III.8.2.2 L'algorithme ACO pour le problème du voyageur commercial.

Dans le problème du voyageur de commerce, un ensemble de villes est donné et la distance entre chacun d'eux est connue. Le but est de trouver le circuit le plus court permettant de visiter chaque ville une fois et une seule fois. L'objectif est de trouver une longueur minimale sur un graphique entièrement connecté. Dans l'optimisation des colonies de fourmis, le problème est résolu en simulant un certain nombre de fourmis artificielles se déplaçant sur un graphique qui code le problème lui-même : chaque sommet représente une ville et chaque bord représente une connexion entre deux villes. Une variable appelée phéromone est associée à chaque bord et peut être lue et modifiée par les fourmis. L'optimisation des colonies de fourmis est un algorithme itératif. A chaque itération, un certain nombre de fourmis artificielles sont considérées. Chacun d'eux construit une solution en marchant d'un sommet à l'autre sur le graphique avec la contrainte de ne visiter aucun sommet qu'elle a déjà visité au cours de sa promenade. A chaque étape de la construction de la solution, une fourmi sélectionne le sommet suivant à visiter selon un mécanisme stochastique qui est biaisé par la phéromone : du sommet i , le sommet suivant est sélectionné stochastiquement parmi ceux qui n'avaient pas encore été visités (Figure III.12). En particulier, si le sommet j n'a pas été visité précédemment, il peut être sélectionné avec une probabilité proportionnelle à la phéromone associée au bord (i, j) . A la fin d'une itération, sur la base de la qualité de solutions construites par les fourmis, les valeurs de phéromone sont modifiées afin de biaiser les fourmis dans les itérations futures pour construire des solutions similaires aux meilleures déjà construites.

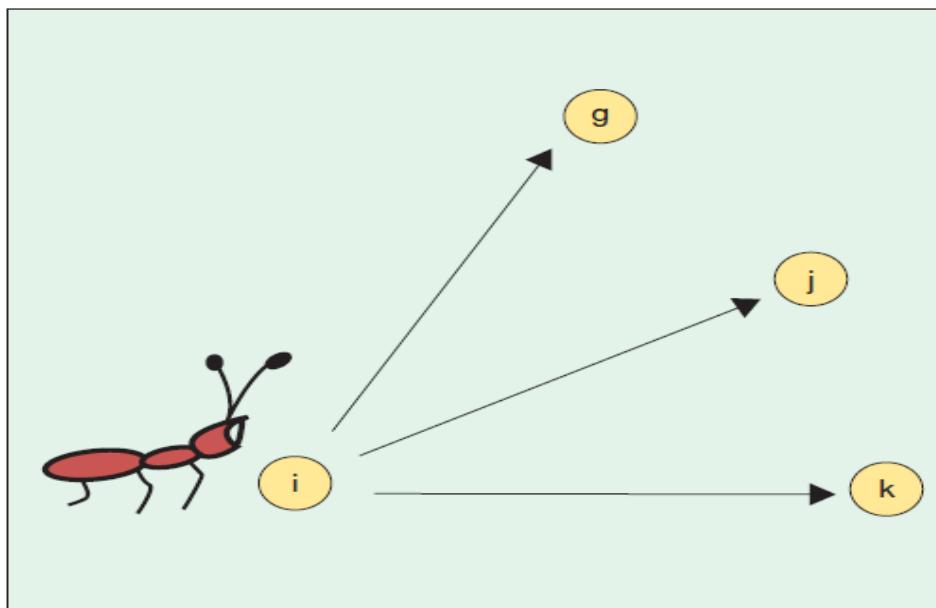


Figure III.12. Une fourmi en ville i choisit la prochaine ville à visiter via un mécanisme stochastique

III.8.2.3 Les algorithmes ACO

Plusieurs algorithmes ACO ont été proposés dans la littérature. Nous présentons ici le système Ant original, et ses deux variantes les plus réussies : MAX-MIN Ant System et Ant Système de colonies. Pour illustrer les différences entre ces trois algorithmes, nous utilisons le problème du voyageur de commerce comme exemple concret.

- **Ant System (AS)**

Ant System est le premier algorithme ACO proposé dans la littérature [III.42], [III.43]. Sa caractéristique principale est que, à chaque itération, les valeurs de phéromone sont mises à jour par toutes les m fourmis qui ont construit une solution dans l'itération elle-même. La variable phéromone τ_{ij} , associée avec le bord joignant les villes i et j , est mise à jour comme suit :

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (\text{III.7})$$

où ρ est le taux d'évaporation, m le nombre de fourmis et τ_{ij} est la quantité de phéromone déposée sur le bord (i, j) par la fourmi k .

$$\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{si la fourmi } k \text{ utilisait le bord } (i, j) \text{ dans son circuit} \\ 0 & \text{autrement} \end{cases} \quad (\text{III.8})$$

où Q est une constante et L_k est la longueur du tour construit par la fourmi k . Dans la construction d'une solution, les fourmis choisissent la ville suivante à visiter par un mécanisme stochastique. Lorsque la fourmi est en ville i et a jusqu'à présent construit la solution partielle s^p , la probabilité d'aller à la ville j est donnée par :

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{l \in N(s^p)} \tau_{il}^\alpha \eta_{il}^\beta} & \text{si } C_{il} \in N(s^p) \\ 0 & \text{autrement} \end{cases} \quad (\text{III.9})$$

où $N(s^p)$ est l'ensemble des composants réalisables ; c'est-à-dire, les bords (i, l) , l est une ville pas encore visitée par la fourmi k . Les paramètres α et β contrôlent l'importance relative de la phéromone par rapport à l'information heuristique η_{ij} , qui est donnée par :

$$\eta_{ij} = \frac{1}{d_{ij}} \quad (\text{III.10})$$

où d_{ij} est la distance entre les villes i et j .

- **MAX –MIN Ant System (MMAS)**

Cet algorithme [III.45] est une amélioration par rapport au système Ant d'origine. Ses éléments caractéristiques sont que seule la meilleure des fourmis met à jour les traces de la phéromone. La mise à jour de la phéromone est implémentée comme suit :

$$\tau_{ij} \leftarrow [(1 - \rho) \cdot \tau_{ij} + \Delta\tau_{ij}^{best}]_{\tau_{min}}^{\tau_{max}} \quad (\text{III.11})$$

où τ_{max} et τ_{min} sont respectivement les bornes supérieure et inférieure, limites imposées à la phéromone; l'opérateur $[x]_b^a$ est défini comme suit:

$$[x]_b^a = \begin{cases} a & \text{si } x > a \\ b & \text{si } x < b \\ x & \text{autrement} \end{cases} \quad (\text{III.12})$$

et $\Delta\tau_{ij}^{best}$ est défini comme ceci :

$$\Delta\tau_{ij}^{best} = \begin{cases} \frac{1}{L_{best}} & \text{si } (i,j) \text{ appartient au meilleur tour} \\ 0 & \text{autrement} \end{cases} \quad (\text{III.13})$$

où L_{best} est la longueur du tour de la meilleure fourmi. Ceci peut être (sous réserve de la décision du concepteur de l'algorithme) soit le meilleur trouvé dans l'itération actuelle - itération-best, L_{ib} la meilleure solution trouvée depuis le début de l'algorithme ou L_{bs} une combinaison des deux. Concernant les limites inférieure et supérieure des valeurs de phéromone, τ_{min} et τ_{max} , elles sont généralement obtenues de manière empirique et à l'écoute du problème spécifique considéré [III.44]. Néanmoins, certaines directives ont été fournies pour définir τ_{min} et τ_{max} sur la base de considérations analytiques [III.45].

- **Ant Colony System (ACS)**

La contribution la plus intéressante d'ACS [III.46] [III.47] est la mise à jour locale de la phéromone en plus de la mise à jour de la phéromone effectuée à la fin du processus de construction (appelée mise à jour de phéromone hors ligne). La mise à jour locale des phéromones est effectuée par toutes les fourmis après chaque étape de la construction. Chaque fourmi ne l'applique qu'au dernier bord traversé :

$$\tau_{ij} = (1 - \varphi) \cdot \tau_{ij} + \varphi \cdot \tau_0 \quad (\text{III.14})$$

où $\varphi \in [0,1]$ est le coefficient de décomposition de la phéromone et τ_0 est la valeur initiale de la phéromone. L'objectif principal de la mise à jour locale est de diversifier la recherche effectuée par les fourmis suivantes au cours d'une itération : en diminuant la concentration de la phéromone sur les bords traversés, les fourmis encouragent les fourmis subséquentes à choisir

d'autres bords et, par conséquent, à produire des solutions différentes. Il est donc moins probable que plusieurs fourmis produisent des solutions identiques au cours d'une itération.

La mise à jour des phéromones hors ligne, semblable à l'algorithme MMAS, est appliquée à la fin de chaque itération par une seule fourmi, qui peut être soit la meilleure itération, soit la meilleure jusqu'ici. Cependant, la formule de mise à jour est légèrement différente :

$$\tau_{ij} \leftarrow \begin{cases} (1 - \rho)\tau_{ij} + \rho \cdot \Delta\tau_{ij} & \text{si } (i, j) \text{ appartient au meilleur tour} \\ \tau_{ij} & \text{autrement} \end{cases} \quad (\text{III.15})$$

Comme dans MMAS, $\Delta\tau_{ij} = \frac{1}{L_{best}}$, où L_{best} peut-être soit L_{ib} ou L_{bs} . Une autre différence importante entre ACS et AS réside dans la règle de décision utilisée par les fourmis pendant le processus de construction. Dans ACS, on utilise la règle dite proportionnelle pseudo-aléatoire : la probabilité qu'une fourmi se déplace de la ville i à la ville j dépend d'une variable aléatoire q uniformément répartie sur $[0, 1]$, et d'un paramètre q_0 :

si $q \leq q_0$ alors $j = \operatorname{argmax}_{C_{ij} \in N(S^P)} \{\tau_{ij} \eta_{ij}^\beta\}$.

III.8.2.4 Applications de l'algorithme de l'optimisation de la colonie de fourmis

Ces dernières années, l'intérêt de la communauté scientifique pour l'ACO a fortement augmenté. En fait, plusieurs applications de l'ACO à une large gamme de différents problèmes d'optimisation discrets sont réussies. La grande majorité de ces applications sont à NP problèmes difficiles, c'est-à-dire aux problèmes pour lesquels les algorithmes les plus connus garantissant l'identification optimale d'une solution ont une complexité exponentielle et dans le pire des cas de tels algorithmes sont souvent irréalisables en pratique. Les algorithmes de l'ACO peuvent être utiles pour trouver rapidement des solutions de haute qualité. D'autres applications populaires de l'ACO concernent aussi les problèmes dynamiques du plus court chemin apparus dans les problèmes de réseaux de télécommunication. Le nombre d'applications à des problèmes académiques réussies a poussé les gens à adopter l'ACO pour la solution des problèmes industriels. Cette technique est également utile dans les applications au temps réel.

III.9 Conclusion

Les algorithmes évolutifs (AE) sont généralement utilisés pour fournir de bonnes solutions approximatives à des problèmes qui ne peuvent pas être résolus facilement en utilisant d'autres techniques. Beaucoup de problèmes d'optimisation entrent dans cette catégorie. Trouver une solution exacte nécessite peut-être trop de temps de calcul, mais une solution presque optimale suffit parfois. Dans ces situations, les techniques évolutives peuvent être efficaces. En raison de leur nature aléatoire, les algorithmes évolutifs ne sont jamais assurés de trouver une solution optimale à tout problème, mais ils trouveront souvent une bonne solution, le cas échéant. Les AE se sont avérés très efficaces dans les applications pratiques, en particulier celles nécessitant des solutions aux problèmes de l'optimisation combinatoire. Les AE sont très flexibles et peuvent être configurés pour traiter toute tâche d'optimisation. Les AE sont aussi utilisés pour traiter les problèmes de séparation aveugle de sources, c.à.d. trouver les coefficients de la matrice de séparation optimale. C'est le sujet du prochain chapitre.

Bibliographie

- [III.1] M. S. Bazaraa and C. M. Shetty, *Nonlinear Programming*, New York: John Wiley, 1979.
- [III.2] B. D. O. Anderson, R. R. Bitmead, C. R. Johnson, P. V. Kokotovic, R. L. Kosut, I. M. Y. Mareels, L. Praly, and B. D. Riedle, *Stability of Adaptive Systems: Passivity And Averaging Analysis*. Cambridge, MA: MIT Press, 1986
- [III.3] P. Werbos, "Beyond regression: new tools for prediction and analysis in the behavioral sciences," Doctoral dissertation, Harvard University, 1974.
- [III.4] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*. vol. 1, Cambridge, MA: MIT Press, 1986.
- [III.5] L. Ljung, *System Identification: Theory for the User*, Englewood Cliffs, NJ: Prentice-Hall. 1987.
- [III.6] Camara, Daniel. *Bio-inspired networking*. Elsevier, 2015.
- [III.7] Back, Thomas, Fogel, D. B., et Michalewicz, Z. *Introduction to evolutionary algorithms*. *Evolutionary computation*, 2000, vol. 1, p. 59-63.
- [III.8] Kursawe, Frank. A variant of evolution strategies for vector optimization. In: *International Conference on Parallel Problem Solving from Nature*. Springer, Berlin, Heidelberg, 1990. p. 193-197.
- [III.9] Marsaul T, Xavier. 1.6 Les Methodes Informatiques Evolutionnaires.
- [III.10] GLOVER, Fred. Future paths for integer programming and links to artificial intelligence. *Computers & operations research*, 1986, vol. 13, no 5, p. 533-549.
- [III.11] Zbigniew Michalewicz, David B. Fogel, *how to Solve It: Modern Heuristics*, Second Edition, Chapter 1, 2, 5, 6, Springer, 2004.
- [III.12] Glover F. Tabu search—part I, *ORSA Journal on Computing*, 1(3) (1989) pp. 190–206
- [III.13] Glover F., Tabu search—part II, *ORSA Journal on Computing*, 2(1) (1990) pp. 4–32.
- [III.14] Tai-Hsi Wu, Jinn-Yi Yeh, Chin-Chih Chang, A hybrid Tabu Search Algorithm to Cell Formation Problem and its Variants, *World Academy of Science, Engineering and technology* 53 (2009) pp. 1090–1094.
- [III.15] Simon, D., Biogeography-based optimization. *Evolutionary Computation*, IEEE Transactions on, 2008. **12**(6): p. 702-713.
- [III.16] Taheri, Seyed Heja Seyed; JALILI, Shahin. Enhanced biogeography-based optimization: a new method for size and shape optimization of truss structures with Natural frequency constraints. *Latin American Journal of Solids and Structures*, 2016, 13.7: 1406-1430.
- [III.17] J.P. Briaut et F. Stéphan, _ *Cloud computing, évolution technologique, révolution des usagers*, 2e édition Lavoisier, Paris, France, Décembre 2013.

- [III.18] Abraham, Siby, KISS, Imre, SANYAL, Sugata, et al. Steepest Ascent Hill Climbing for A Mathematical Problem. ArXiv preprint arXiv:1010.0298, 2010.
- [III.20] Skiena, Steven S. Combinatorial search and heuristic methods. In: The Algorithm Design Manual. Springer, London, 2012. p. 230-272.
- [III.21] Arts, Emile, Korst, Jan, et Michiels, Wil. Simulated annealing. In: Search methodologies. Springer, Boston, MA, 2005. p. 187-210.
- [III.22] METROPOLIS, Nicholas, ROSENBLUTH, Arianna W., ROSENBLUTH, Marshall N., et al. Equation of state calculations by fast computing machines. The journal of chemical physics, 1953, vol. 21, no 6, p. 1087-1092.
- [III.23] Szu, Harold et Hartely, Ralph. Fast simulated annealing. Physics letters A, 1987, vol.122, no 3-4, p. 157-162.
- [III.24] CURILEF, Sergio. Generalized statistical mechanics for the N-body quantum problem ideal gases. Zeitschrift für Physik B Condensed Matter, 1997, vol. 100, no 3, p. 433-440.
- [III.25] Dalligna Junior, Alcino, Silva, Renato S., Mundim, Kleber C., et al. Performance and parameterization of the algorithm Simplified Generalized Simulated Annealing. Genetics and Molecular Biology, 2004, vol. 27, no 4, p. 616-622.
- [III.26] Goldberg, David E. et Holland, John H. Genetic algorithms and machine learning. Machine learning, 1988, vol. 3, no 2, p. 95-99.
- [III.27] Eshelman, L. And Schaffer, J. Real-coded genetic algorithms and interval-schemata. In Foundations of Genetic Algorithms 2, L. D. Whitley, Ed. Morgan Kaufmann, 1993, pp. 187–202.
- [III.28] K. Deb and S. Jain, “Running performance metrics for evolutionary multi-objective optimization,” Indian Institute of Technology Kanpur, Tech. Rep., 2004.
- [III.29] M. Lunacek and D. Whitley, “The dispersion metric and the CMA evolution strategy,” In Proceeding soft Genetic and Evolutionary Computation Conference, 2006, pp. 447–484.
- [III.30] Lakshmi, K. Lohitha; Bhargavi, P.; Jyoth, S. An Analysis of Breast Cancer Gene Sequences using Differential Evaluation. International Journal of Simulation—Systems, Science & Technology, 2019, 20.1.
- [III.31] Datta, Subrata. Efficient genetic algorithm on linear programming problem for fittest chromosomes. Journal of Global Research in Computer Science, 2012, 3.6: 1-7.
- [III.32] Bajpai, Pratibha; Kumar, Manoj. Genetic algorithm—an approach to solve global Optimization problems. Indian Journal of computer science and engineering, 2010, 1.3: 199-206.
- [III.33] S. M. Lim, “Crossover and mutation operators of real coded genetic algorithms for

Global optimization problems,” unpublished PhD thesis, University Putra Malaysia, Malaysia, 2016.

[III.34] Public, K and Manger, R. (2013) Comparison of eight evolutionary crossover operators for the vehicle routing problem. *Journal of Mathematical Communications*, 18, 359-375.

[III.35] VARUN KUMAR, S. G. et PANNEERSELVAM, R. A study of crossover operators for genetic algorithms to solve VRP and its variants and new sinusoidal motion crossover operator. *Int. J. Comput. Intell. Res*, 2017, vol. 13, no 7, p. 1717-1733.

[III.36] ABDOUN, Otman, ABOUCHABAKA, Jaafar, et TAJANI, Chakir. Analyzing the performance of mutation operators to solve the travelling salesman problem. *arXiv preprint arXiv:1203.3099*, 2012.

[III.37] Y. Kao, E. Zahara. A hybrid genetic algorithm and particle swarm optimization for Multimodal functions. *Applied Soft Computing*, 8 :849–857, 2008.

[III.38] Blum, Christian. Ant colony optimization: Introduction and recent trends. *Physics of Life reviews*, 2005, vol. 2, no 4, p. 353-373.

[III.39] Olivier Rouxles." algorithmes a colonie de fourmis : applications a l'optimisation et a la programmation automatique" ; le 13 décembre 2005. *PolyJ.DREO*. Adaptation de la méthode des colonies de fourmis pour l'optimisation en variables continues. PhD thesis, Université Paris 12, 2005.

[III.40] DENEUBOURG, J.-L., ARON, Serge, GOSS, Simon, et al. The self-organizing exploratory pattern of the argentine ant. *Journal of insect behavior*, 1990, vol. 3, no 2, p. 159-168.

[III.41] S. Goss, S. Aron, J.-L. Deneubourg, and J.-M. Pasteels, “Self-organized shortcuts in the Argentine ant,” *Natur wissen schaften*, vol. 76, pp. 579–581, 1989.

[III.42] M. Dorigo, V. Maniezzo, and A. Colorni, “Positive feedback as a search strategy,” *Dipartimento di Elettronica, Politecnico di Milano, Italy, Tech. Rep.* 91-016, 1991.

[III.43] M. Dorigo, V. Maniezzo, and A. Colorni, “Ant System: Optimization by a colony of cooperating agents,” *IEEE Transactions on Systems, Man, and Cybernetics—Part B*, vol. 26, no. 1, pp. 29–41, 1996.

[III.44] K. Socha, J. Knowles, and M. Sampels, “A MAX–MIN ant system for the university Timetabling problem,” in *Proc. ANTS 2002*, ser. LNCS, M. Dorigo et al., Eds., vol. 2463, p. 1, Berlin, Germany: Springer Verlag, 2002.

[III.45] T. Stutzle and H.H. Hoos, “MAX–MIN Ant System,” *Future Generation Computer Systems*, vol. 16, no. 8, pp. 889–914, 2000.

[III.46] M. Dorigo and L.M. Gambardella, “Ant colonies for the traveling salesman problem,” *Bio Systems*, vol. 43, no. 2, pp. 73–81, 1997.

[III.47] L.M. Gambardella and M. Dorigo, “Solving symmetric and asymmetric TSPs by ant colonies,” in *Proc. 1996 IEEE International Conference on Evolutionary Computation (ICEC’96)*, T.Baeck et al., Eds. IEEE Press, Piscataway, NJ, pp. 622– 627, 1996.

CHAPITRE IV

**Séparation aveugle de sources audio basée
sur l'algorithme HEPSO**

IV. Introduction

Comme il a été dit dans les chapitres précédents la séparation de source aveugle (SAS) ou en anglais Blind Source Séparation (BSS) est un outil puissant qui est utilisé pour restaurer les signaux source de plusieurs observations sans aucune connaissance préalable de la source des signaux ainsi que de leur processus de mélange [IV.1] [IV.2]. Récemment, plusieurs approches ont été proposées dans la littérature pour résoudre les problèmes de la SAS. Peng Cheng XU et al. ont conçu une nouvelle fonction de contraste basée sur des approximations de négentropie pour l'extraction de sources complexes et ont également introduit une nouvelle famille d'algorithmes, notée Alg. 1 et Alg. 2, pour la maximiser [IV.3]. Jiangong Wang et al. ont proposé un algorithme SAS basé sur des autocorrélations linéaires et une diagonalisation conjointe (LAJD), réalisé par le biais d'un diagonaliseur fusionnant plusieurs matrices de covariance dans un délai optimal [IV.4]. Yongjian Zhao et al. ont proposé une méthode séquentielle sur la base de statistiques de second ordre [IV.5]. Les auteurs évaluent d'abord le signal source, puis un processus de déflation permet de l'extraire à partir de ses mélanges. Cela peut être utilisé pour récupérer un signal source ou tous les signaux dans un ordre spécifique. Les auteurs ont validé les performances de cette méthode via des simulations informatiques. Jehad Abadneh et al. ont utilisé la technique d'optimisation basée sur la biogéographie (BBO) afin d'optimiser la fonction de contraste qui mesure l'indépendance statistique des composants améliorés [IV.6]. Dans [IV.7] et [IV.8], Po-Sen Huang et al., ont utilisé un modèle d'apprentissage en profondeur pour la séparation monaurale de la parole telles que la séparation de la parole, la séparation de la voix de chant et le débruitage de la parole. Les auteurs ont proposé l'optimisation conjointe d'une fonction de masquage et de modèles d'apprentissage en profondeur, à savoir les réseaux de neurones profonds et les réseaux de neurones récurrents (DNN et RNN). Le critère d'apprentissage proposé permet d'améliorer le rapport signal sur interférence (SIR). Ce modèle atteint un gain SIR de 3,8 à 4,9 dB par rapport au niveau de référence de la factorisation matricielle non négative (NMF), tout en maintenant de bons SDR (Source to Distortion Ratio) et SAR (Sources to Artifacts Ratio). Les performances sont optimisées en combinant une couche de masque temps-fréquence souple avec les réseaux de manière globale, tout en recherchant un critère d'apprentissage discriminant. Liangli ZHEN et al. ont exploité un codage clair dans la SAS, avec n sources non identifiées, pour séparer ces sources de leurs m signaux de mélange observés, où $m < n$ [IV.9]. L'approche consiste tout d'abord à découvrir quelques sous-espaces 1-D de l'ensemble constitué de tous les vecteurs de représentation temps-fréquence (TF) des signaux de mélange observés, puis à détecter les points sources uniques dans le plan TF, ensuite l'estimation de la matrice de mélange est réalisée en

regroupant les vecteurs de mélange TF au niveau de ces points source. Enfin, la récupération de la source est obtenue en résolvant un ensemble de problèmes de moindres carrés. De plus, des algorithmes évolutifs sont appliqués avec succès pour résoudre plusieurs problèmes d'optimisation dans divers domaines. Les algorithmes génétiques (GA) [IV.10] et [IV.11], l'algorithme PSO (Particle Swarm Optimization) [IV.12] et [IV.13] et l'algorithme Artificial Bee Colony (ABC) [IV.14] ont été appliqués pour trouver la matrice de séparation optimale.

Dans ce chapitre, une PSO modifiée nommée HEPPO (High Exploration Particle Swarm Optimization) [IV.15] a été utilisée. Cependant, HEPPO repose sur deux opérateurs supplémentaires. Le premier est basé sur le « multi-crossover » largement utilisé dans les algorithmes génétiques. Tandis que le second repose sur le mécanisme de la colonie d'abeilles. Les deux opérateurs ont été utilisés pour mettre à jour la vitesse et la position des particules, respectivement. Ainsi, ces opérateurs sont utilisés afin de trouver la matrice de séparation optimale. La méthode proposée améliore l'efficacité globale de la PSO standard en termes de bonne exploration et de bonnes performances. Les résultats expérimentaux confirment que la technique SAS introduite illustre la robustesse et la précision de nombreux tests de signaux de parole et de musique fournis dans la plateforme « BSSdemo ».

IV.1 Problème de la SAS

La SAS est un processus utilisé pour séparer et récupérer les signaux source à partir d'un signal observé. Ce processus est traité par les propriétés statistiques des signaux source. Le modèle de base de la SAS est présenté sur la figure IV.1 [IV.16].

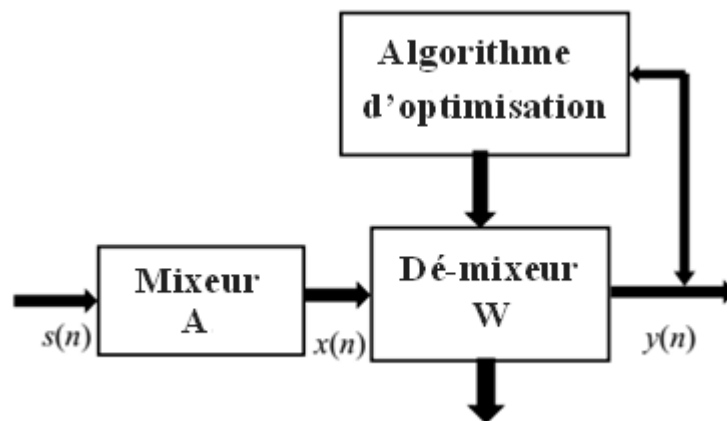


Figure IV.1. Le modèle de base du processus SAS

Dans notre étude, nous nous intéressons au processus de l'algorithme d'optimisation illustré dans le schéma de la Figure IV.1.

L'algorithme choisi (HEPPO) permet d'optimiser la fonction objective afin de rendre les signaux de sortie indépendants.

Le modèle SAS est défini par :

$$X = A.S \quad (IV.1)$$

où $X = [x_1, x_2, x_3 \dots x_m]^T$ est un vecteur contenant le signal observé, $S = [s_1, s_2, s_3 \dots s_n]^T$ est un vecteur contenant les sources originales et A constitue la matrice de mélange avec rang de colonne complet. Le modèle linéaire peut être défini comme suit :

$$X_i(t) = a_{k1}s_1 + a_{k2}s_2 + \dots a_{kn}s_n, \quad k = 1, n \quad (IV.2)$$

Nous supposons que le nombre de sources n est égal au nombre de mélanges $m (m = n)$. En fait, lors de la résolution des problèmes de SAS, deux hypothèses sont formulées. La première hypothèse est que les signaux source sont statistiquement indépendants et la seconde est qu'une seule source est autorisée à avoir une distribution gaussienne car une combinaison linéaire de signaux gaussiens est gaussienne et qu'il est dans ce cas impossible de les séparer.

Avec les hypothèses ci-dessus, le résultat de la SAS exhibe deux ambiguïtés inhérentes [IV.17]:

1. Nous ne pouvons pas déterminer la variance des composants indépendants.
2. Nous ne pouvons pas déterminer l'ordre des composants indépendants.

Dans cette étude, notre objectif principal est de séparer les signaux les uns des autres. L'ordre des signaux estimés n'est pas important. La matrice de séparation (W_i) peut estimer la source originale donnée par :

$$y = W_i x \quad (IV.3)$$

En outre, la méthode ICA (Independent Component Analysis) est très précise lorsqu'elle est connectée à la SAS. Elle trouve des composants statistiquement indépendants à partir de données multivariées. Il existe de nombreuses façons de réaliser une ACI, y compris la minimisation des informations mutuelles ou la maximisation de la non-gaussianité.

IV.2 Optimisation par Essais Particulaires (OPE)

L'intelligence de l'essaim est une branche de l'intelligence artificielle qui étudie le comportement collectif et les propriétés émergentes de systèmes complexes, auto-organisés, décentralisés avec une structure sociale. L'inspiration principale derrière le développement de l'intelligence de l'essaim provient directement de la nature. Écoles de poissons, troupeaux d'oiseaux, colonies de fourmis et troupeaux d'animaux. Le concept d'optimisation de l'essaim de particules consiste, à chaque pas de temps, à changer de direction de recherche en fonction

de deux facteurs pour découvrir la solution optimale. Le premier est sa meilleure expérience antérieure et l'autre est la meilleure expérience de tous les autres membres.

L'Optimisation par Essaims Particulaires (OEP) (en anglais : Particle Swarm Optimisation PSO) [IV.18] a été appliquée avec succès dans de nombreux domaines de recherche et d'application. Il est démontré que l'algorithme PSO obtient de meilleurs résultats d'une manière plus rapide et moins coûteuse par rapport à d'autres méthodes [IV.13]. Le processus de base de l'algorithme PSO est l'initialisation, la mise en forme, la mise à jour, la construction et la terminaison. Le processus du PSO est terminé si la condition de terminaison est satisfaite. Les détails sont donnés dans [IV.13]. Les étapes principales de cet algorithme sont les suivantes :

1. Générer des particules initiales de manière aléatoire.
2. Mesurer l'aptitude de chaque particule dans la population.
3. Calculer la vitesse de chaque particule.
4. Passer à la position suivante pour chaque particule.
5. Arrêter l'algorithme si le critère de terminaison est satisfait.

Sinon, revenir à l'étape 2.

IV.3 Principe de l'algorithme PSO

L'algorithme PSO de base consiste en un essaim de "n-pop" particules [IV.18]. La position de chaque particule représente une solution possible de la fonction de fitness dans l'espace de recherche de dimension D. La particule change son état sous l'influence de trois facteurs :

1. Sa propre inertie.
2. La position personnelle la plus optimale (personal best).
3. La position la plus optimale de l'Essaim (global best).

Ces facteurs déterminent une nouvelle position de chaque particule. Chaque particule a un vecteur de vitesse (velocity) et un vecteur de position (figure 2). À chaque itération, le fitness de chaque particule est évalué avec la fonction de coût.

Mathématiquement, les particules sont exploitées selon les équations suivantes :

$$v_i(t+1) = wv_i(t) + C_1 \cdot r_1 (x_{pbest_i}(t) - x_i(t)) + C_2 r_2 (x_{gbest}(t) - x_i(t)) \quad (IV.4)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (IV.5)$$

Où $x_i(t)$ et $v_i(t)$ indique respectivement la position et la vitesse de la particule i à l'itération t , $r_1, r_2 \in [0, 1]$, w le poids d'inertie est ajusté dynamiquement comme proposé dans [IV.15] et est donné dans (IV.8).

$$C_1 = C_{11} - (C_{11} - C_{1f}) \left(\frac{t}{\max \text{ itération}} \right) \quad (\text{IV.6})$$

$$C_2 = C_{12} - (C_{12} - C_{2f}) \left(\frac{t}{\max \text{ itération}} \right) \quad (\text{IV.7})$$

C_{11} et C_{12} sont les valeurs initiales des facteurs d'apprentissage de C_1 et C_2 respectivement, tandis que C_{1f} et C_{2f} sont respectivement leurs valeurs finales.

$$w(f) = \frac{1}{(1 + 1.5e^{-2.6f})} \quad (\text{IV.8})$$

$$f = \frac{d_g - d_{\min}}{d_{\max} - d_{\min}} \in [0, 1] \quad (\text{IV.9})$$

$$d_i = \frac{1}{N-1} \sum_{j=1, j \neq i}^N \sqrt{\sum_{k=1}^D (x_i^k - x_j^k)^2} \quad (\text{IV.10})$$

Afin de calculer $d_{\min}$ et $d_{\max}$, on peut utiliser une métrique euclidienne via l'équation (IV.10) où d_i est la distance moyenne de particule i à toutes les autres particules, N et D sont la taille de la population et le nombre de dimensions respectivement. d_i pour la meilleure particule globale indiquée en d_g . $x_{p_{\text{besi}}}(t)$, représente la meilleure position personnelle et $x_{g_{\text{besi}}}(t)$ représente la meilleure position globale de la i^{ieme} particule (sur les n -pop particules) à la k^{ieme} itération respectivement (sur les n -iter itérations).

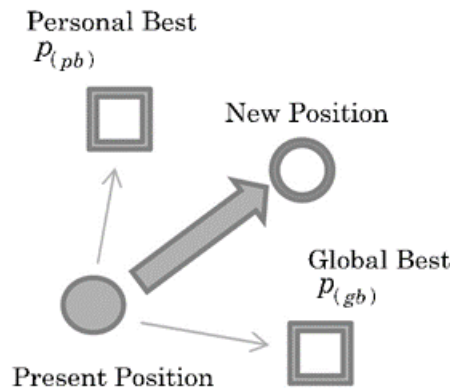


Figure IV.2. Concept de base d'une particule de PSO.

La mise à jour de la vitesse dans l'algorithme PSO se compose de trois parties :

1. Le Momentum : Il représente la tendance de la particule à se déplacer dans la même direction que lors de son déplacement dans l'itération précédente. Il incorpore l'effet de la vitesse précédente sur la vitesse actuelle de la particule.
2. La partie cognitive : elle représente la vitesse d'attraction de la particule vers son meilleur potentiel personnel (pbest), désigné comme "mémoire", "connaissance de soi" ou "souvenir".
3. La partie sociale : Elle représente la vitesse d'attraction de la particule vers le meilleur de l'essaim (gbest), désigné comme "coopération", "connaissance sociale" ou "information partagée".

IV.4 Évaluation de la fonction objective

La fonction de fitness proposée est basée sur le kurtosis et l'information mutuelle. Kurtosis joue un rôle important dans la SAS. Il est utilisé pour mesurer la non gaussianité du signal et trier les composants indépendants. Les informations mutuelles mesurent la dépendance entre les composants. Pour parvenir à l'indépendance des composants, l'information mutuelle doit être minimale. L'information mutuelle est définie comme suit :

$$I(y_1 y_2, \dots, y_n) = \sum_{i=1}^n H(y_i) - H(y) \quad (\text{IV.11})$$

Où $H(y_i) = -E \log p_{y_i}(y_i)$ et $H(y) = -E \log p(y)$, sont l'entropie des signaux de source audio estimés, E est l'espérance mathématique et p_y la densité de y . Le kurtosis des signaux de source audio estimés peut être calculé à l'aide de la formule suivante :

$$\text{kurtosis}(y) = \sum_{i=1}^n |E(y_i^4) - 3E^2(y_i^2)| \quad (\text{IV.12})$$

La fonction fitness peut être exprimée par :

$$J(y) = I(y_1 y_2, \dots, y_n) + \frac{1}{\sum_{i=1}^n |E(y_i^4) - 3E^2(y_i^2)|} \quad (\text{IV.13})$$

Lorsque $J(y)$ est maximisé, les signaux de source audio estimés sont mutuellement indépendants.

IV.5 Méthode Proposée [IV.19]

Dans la partie suivante, décrit le système de la SAS introduit et utilisé dans les problèmes de séparation de plusieurs sources audio. Fondamentalement, le système SAS a été réalisé en quatre étapes principales :

(i) Le processus de mélange des sources audio d'entrée, (ii) puis, la séparation des signaux de sortie du premier étage à l'aide des méthodes PSO et HEP SO (iii) par la suite, on a estimé les

sources de produits en utilisant l'équation (IV.3) (iiii) et enfin la performance de l'évaluation du système de BSS. Le schéma de la méthode introduite est clairement illustré à la figure IV.3.

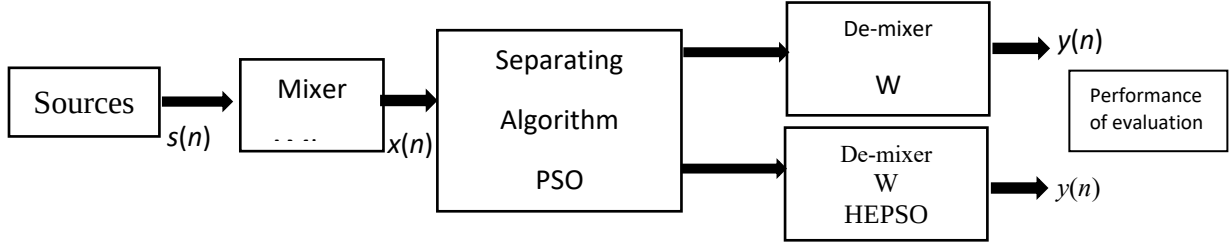


Figure IV. 3. Schéma de la méthode proposée

IV.5.1 SAS basée sur l'algorithme HEP SO

Dans cette partie, nous présentons la technique proposée pour la SAS. Nous commençons par la 1ère étape, dans laquelle la population initiale est produite. Dans la deuxième étape, le poids d'inertie (w) et les facteurs d'apprentissage (C_1 , C_2) sont calculés à chaque itération. A la troisième étape, nous évaluons la fonction objective de toutes les particules. A la quatrième étape, l'opérateur de colonie d'abeilles (équation (IV.17)) n'est utilisé que si $p_1 < \text{déviati on standard}$ ou $p_2 < \frac{P_B \times t}{\text{max iteration}}$, où p_1 et $p_2 \in [0,1]$, p_B étant la probabilité de colonie d'abeilles, t est l'itération en cours et max itération est le nombre maximum d'itérations. Lors de la 5ème étape, le multi-crossover sera généré pour chaque particule non sélectionnée dans l'opération précédente. Dans le cas d'une particule qui a $v < p_c$, l'opérateur multi-crossover génèrera une velocité ou une nouvelle particule via l'équation (IV.16), p_c est la probabilité du multi-crossover. PSO va générer d'autres particules qui ne sont sélectionnées ni pour les opérations d'abeilles ni pour les opérations de multi-crossover (équations (IV.4) et (IV.5)). À la 6ème étape, les valeurs de transformation sont obtenues à partir de x_{best} dans le vecteur de séparation.

Dans la 7ème étape, le processus d'orthogonalisation est effectué via l'équation (IV.14).

$$W_i = W_i \times \text{real}(\text{inv}(W_i \times W_i^T))^{-\frac{1}{2}} \quad (\text{IV.14})$$

Lors de la 8ème étape, nous revenons à la 3ème étape et répétons le processus jusqu'à ce que le critère d'arrêt soit satisfait. Enfin, à la 9ème étape, nous calculons la source séparée en utilisant l'équation (IV.3).

IV.5.2 Les Opérateurs

Cette sous-section présente deux opérateurs. Le premier est basé sur trois chromosomes parents [IV.22] et le second est un processus de colonie d'abeilles [IV.23]. Ces opérateurs visent à améliorer le processus et à s'échapper les minimas locaux.

IV.5.2.1 L'Opérateur 1

Une nouvelle méthode de croisement consistant en trois chromosomes parents $[\theta_1, \theta_2, \theta_3]$ est suggérée dans cette étude, mais elle n'est pas similaire à l'algorithme génétique classique à croisements multiples proposé par Chang [IV.22], qui ne repose que sur deux chromosomes. Nous supposons que les chromosomes sont choisis de manière aléatoire dans la population. Si le chromosome a la valeur d'adaptation la plus petite, le croisement multiple est utilisé pour générer un nouveau chromosome.

$$\theta'_1 = \theta_1 + r(2\theta_1 - \theta_2 - \theta_3) \quad (IV.15)$$

Où $r \in [0,1]$ est une valeur aléatoire. Le premier parent caractérise la meilleure position du groupe ($x_{gbest}(t)$), le deuxième parent représente la meilleure position personnelle (x_{pbest}), tandis que le troisième parent est utilisé pour générer la nouvelle velocity de la particule sélectionnée x_i via l'équation (IV.16).

$$v_i(t+1) = r \left[\frac{c_2}{2} x_{gbest}(t) - x_{pbest}(t) - x_i(t) \right] \quad (IV.16)$$

IV.5.2.2 L'Opérateur 2

L'algorithme de colonie d'abeilles artificielles (ABC) a été développé par Karaboga et Basturk en 2005, en inspectant le comportement d'abeilles réelles qui trouvent des sources de nourriture, appelé nectar, et en partageant les informations avec les autres abeilles du nid. Dans cet algorithme, les abeilles artificielles sont définies et classées en trois groupes : les abeilles employeurs (les abeilles qui cherchent la nourriture), les spectateurs (les abeilles d'observation), et les éclaireurs chargés de trouver de nouveaux aliments [IV.23] [IV.24]. L'algorithme ABC a cinq étapes. Dans un premier temps, les abeilles commencent à chercher de l'espace au hasard pour trouver une source de nourriture. Dans la deuxième étape et après avoir trouvé une source de nourriture, les abeilles deviennent un dossier de l'employeur et commencent à bénéficier de la source inventive. Une abeille employeur remplace son emplacement par des informations locales et trouve un aliment voisin. En fait, l'aptitude du poste est estimée et si celui-ci est meilleur, il sera remplacé par l'ancien poste. La troisième étape concerne le calcul des valeurs

de probabilité qui sont impliquées dans la sélection probabiliste en fonction des valeurs de condition physique de la source de nourriture. L'opérateur de recherche est utilisé dans la technique d'optimisation d'essaims de particules via l'équation (IV.17).

$$x_i^d(t+1) = x_i^d + (2r - 1) (x_i^d(t) - x_j^d(t)) \quad (\text{IV.17})$$

Ou $r \in [0,1]$ est une valeur aléatoire, d est un nombre aléatoire de longueur [1, dimension], et j est un nombre aléatoire de longueur [1, nombre de particules].

IV. 6 Résultats Simulations

Pour montrer les performances de la technique proposée sur la séparation des signaux de parole et de musique, des tests expérimentaux sont effectués et comparés aux résultats de la méthode PSO [IV.21].

IV.6.1 Les signaux source

Tous les signaux de parole et de musique sont sélectionnés dans la démonstration SAS [IV.24] et échantillonnés à 16 KHz. Dans la première expérience, nous prenons deux signaux de parole, le premier (poem_male_30s.wav) est prononcé par un homme, tandis que le second (sentence_female_28s.wav) est prononcé par une femme.

Dans la deuxième expérience, nous prenons un signal musical (music_Loopass_30s.wav) et un signal vocal (henry_theater_male_30s.wav).

Dans la troisième expérience, nous prenons des signaux musicaux: (music_mandolin_30s.wav) et (music_vivaldi_30s.wav).

Lors de l'initialisation, les capteurs et les sources sont réglés sur 2 éléments chacun. La matrice de mélange 2×2 A est choisie aléatoirement comme suit :

$$A = \begin{bmatrix} 0.4701 & 0.6705 \\ 0.2742 & 0.5310 \end{bmatrix}$$

IV. 6.2 Les paramètres de simulation

Les valeurs d'initialisation des paramètres pour la méthode proposée par HEPSO sont les suivantes : La taille de la population est fixée à 100, le problème a une dimension de 9 (dim = 9), la probabilité de mutation p_B est égale à 0,02, la probabilité de croisement p_c est fixée à 0,9 et les autres valeurs de combinaison sont fixés à $C_{1i} = 2.5, C_{2i} = 0.5, C_{1f} = 0.5, C_{2f} = 2.5$.

IV.6.3 Les mesures de performance

Afin d'examiner les performances de la technique proposée, nous l'évaluons à l'aide de trois indices de performance : source to distortion ratio (SDR) [IV.10], valeur absolue des

coefficients de corrélation (AVCC)) [IV.10] et erreur quadratique moyenne (MSE) [IV.10]. Ils sont, respectivement définis comme suit :

$$SDR(y_i, s_i) = 10 \log \left[\frac{\sum_{t=1}^M [y_i(t)]^2}{\sum_{t=1}^M [y_i(t) - s_i(t)]^2} \right] \quad (IV.18)$$

où y_i sont les signaux estimés, s_i sources originales avec M échantillons.

$$AVCC = \left| \frac{\sum_{t=1}^M y_i(t) s_k(t)}{\sqrt{\sum_{t=1}^M y_i(t)^2 \sum_{t=1}^M s_k(t)^2}} \right| \quad (IV.19)$$

Simulation 1 : Les signaux poème-mâle_30s.wav et sentence_female_28s.wav et leurs signaux estimés sont montrés dans figure IV.4.

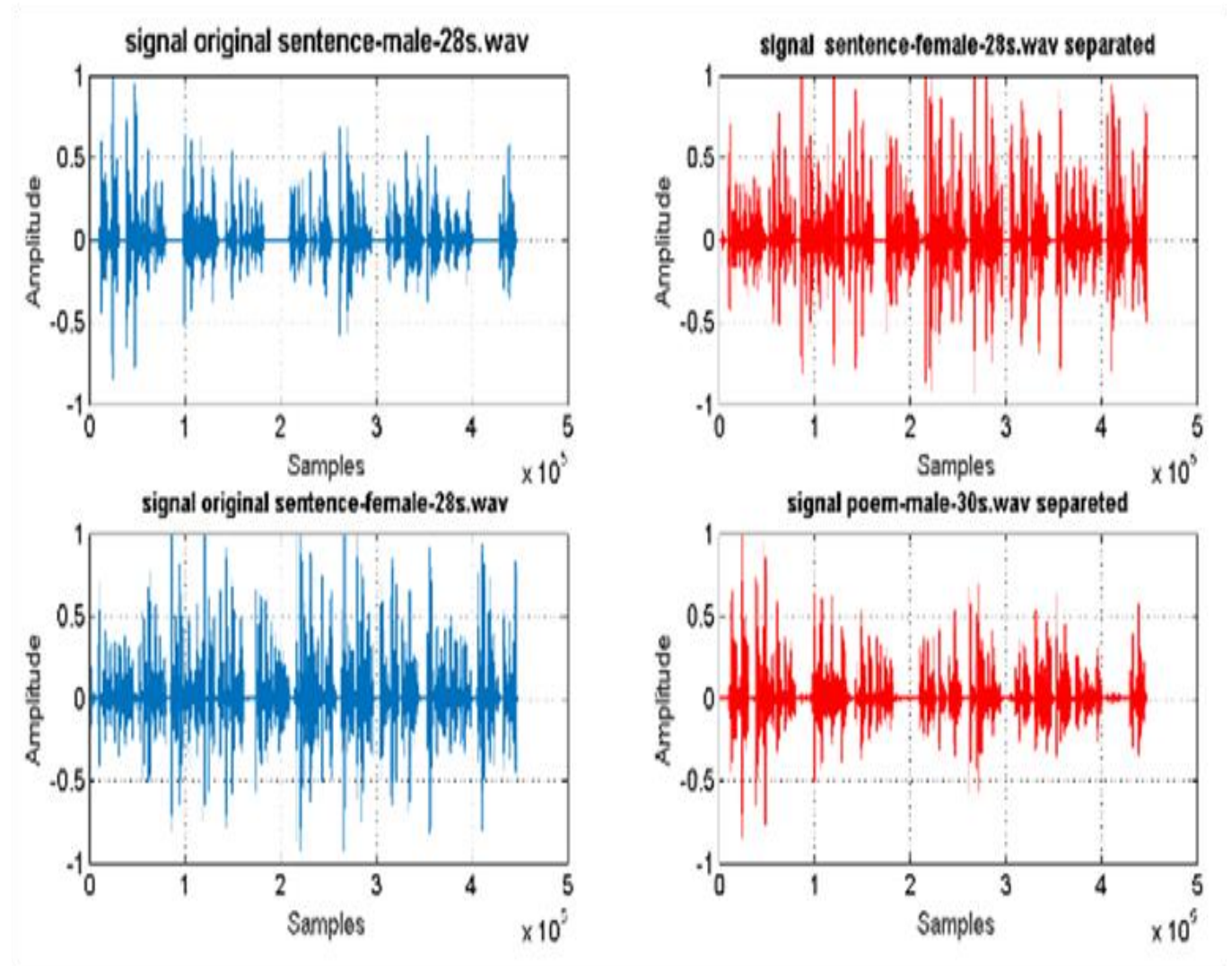


Figure IV.4. Signaux originaux et estimés poème-mâle_30s.wav et sentence_female_28s.wav

Simulation 2 : music_Loophass_30s.wav et henry_theater_male_30s.wav et leurs estimés sont montrés sur la figure IV.5.

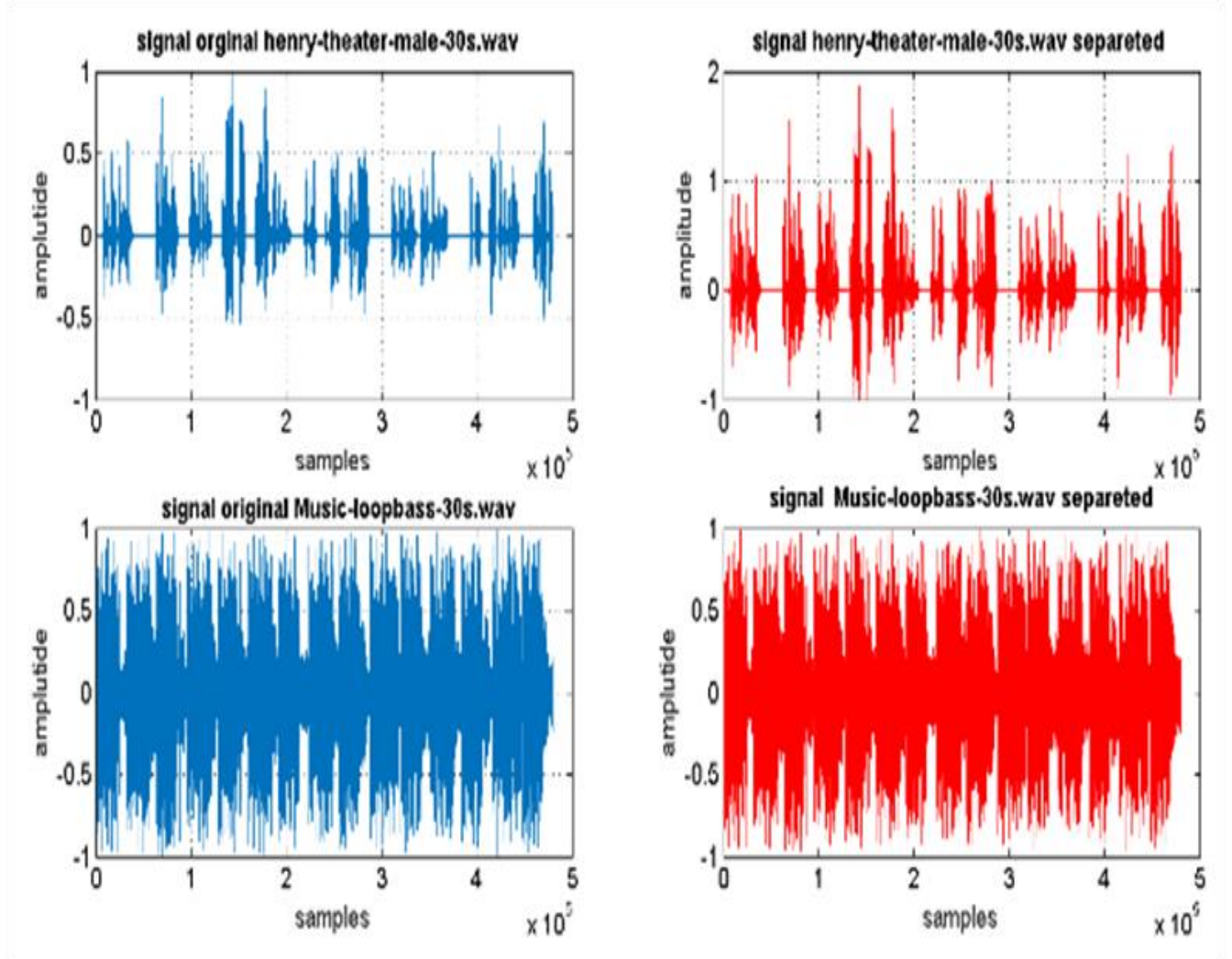


Figure IV.5. Signaux originaux et estimés de music_Loophass_30s.wav et henry_theater_male_30s.wav

Simulation 3 : music_mandolin_30s.wav et music_vivaldi_30s.wav ainsi que leurs fichiers estimés sont illustrés dans figure IV.6.

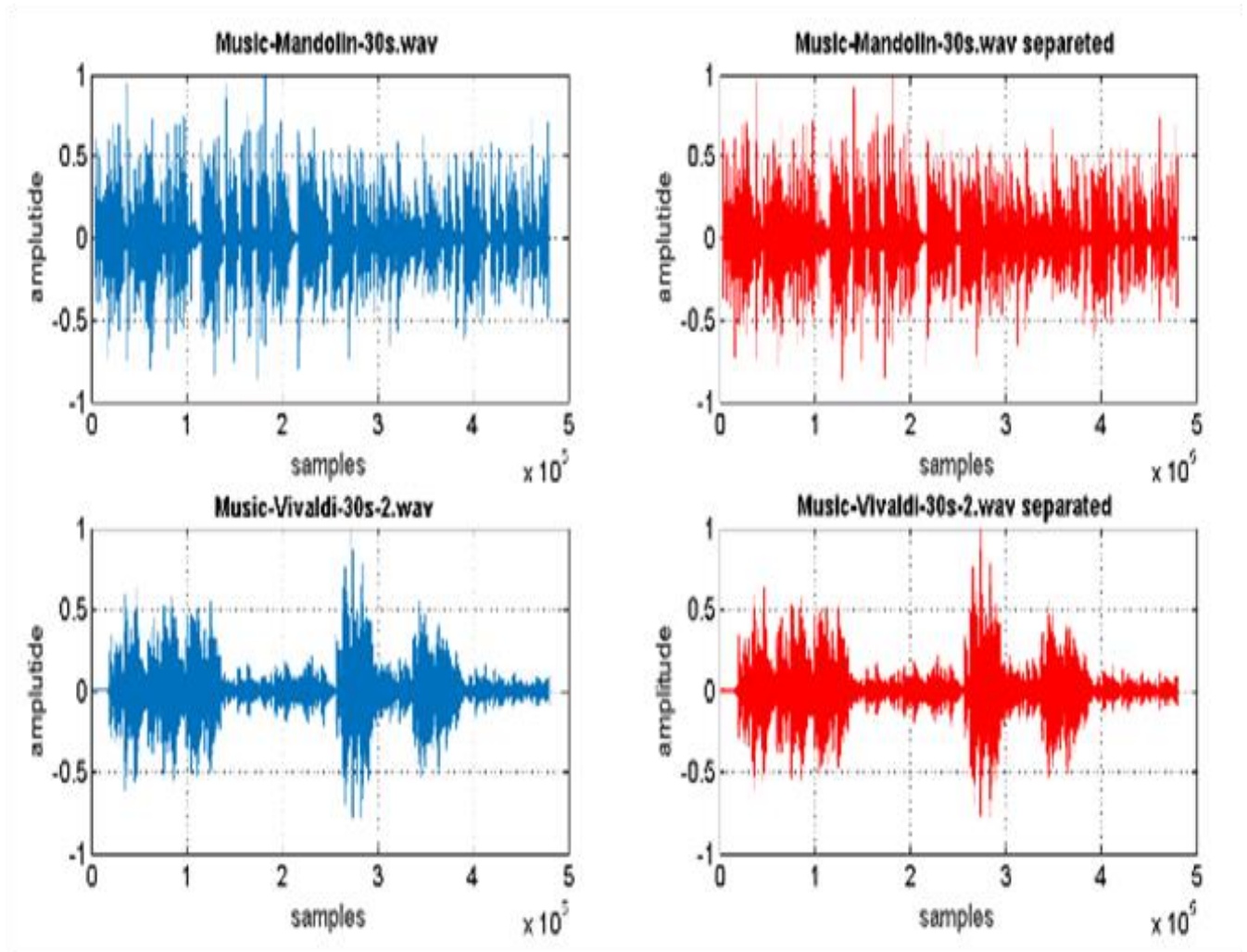


Figure IV.6. Signaux originaux et estimés de music_mandolin_30s.wav et music_vivaldi_30s.wav

IV.7 Interprétations

Des résultats expérimentaux, on peut déduire : Premièrement, sur les figures IV.4 à IV. 6, nous voyons que les formes d'onde des signaux de parole récupérés et des signaux de musique sont similaires à celles d'origine. Cependant, les signaux séparés par la musique Figure IV. 6 sont un peu moins précis. Deuxièmement, nous prouvons l'efficacité de l'algorithme proposé en utilisant des indices de performance SDR, AVCC et MSE. Les résultats expérimentaux des trois expériences sont résumés dans les tableaux 1, 2 et 3.

Tableau IV.1. SDR (dB) pour l'algorithmes PSO et HEPsO

Simulations	PSO	HEPsO
Simulation 1		
Speech s1	24.9632	48.7775
Speech s2	25.2020	39.4966
Simulation 2		
Speech s3	19.2221	74.0782
Music s4	19.4236	23.6904
Simulation 3		
Music s5	25.0062	26.1812
Music s6	26.0567	54.7541

Tableau IV.2. AVCC pour le PSO et l'algorithmes proposé

Simulations	PSO (Sources estimés)		Algorithme proposé (Sources estimés)	
	y1	y2	y1	y2
Simulation 1				
Speech s1	0.0435	0.9987	0.0071	1.0000
Speech s2	0.9989	0.0474	0.9999	0.9999
Simulation 2				
Speech1 s3	0.9941	0.1038	1.0000	0.0699
Music s4	0.1063	0.9943	0.0027	0.9977
Simulation 3				
Music s5	0.0549	0.9988	0.9984	0.0031
Music s6	0.9984	0.0485	0.0549	1.00000

Tableau IV.3. MSE pour le PSO et l'algorithmes proposé

Simulations	PSO	Algorithme proposé
Simulation 1		
Speech s1	1.7695e-04	6.6862e-08
Speech s2	8.9424e-05	1.1463e-06
Simulation 2		
Speech1 s3	6.8208e-05	2.2881e-10
Music s4	9.7202e-04	3.9061e-04
Simulation 3		
Music s5	6.1479e-04	4.7101e-05
Music s6	5.6638e-04	5.3490e-04

Ce travail a été comparé aux études précédentes traitant du problème de la SAS audio. Il a été clairement constaté que la méthode proposée, HEP SO, est supérieure à la méthode GA + kurtosis [IV.21] en termes de SDR (voir le tableau IV.4. En termes de mesure AVCC, la méthode introduite HEP SO réalise une excellente performance (100%). Le tableau IV.5, présente plus de détails sur la précision de la comparaison des travaux proposés avec GA + kurtosis [IV.21].

Tableau IV.4. SDR (dB) pour la méthode GA + kurtosis et la méthode proposée

	GA+ kurtosis[IV.21]	méthode proposée
Speech s1	30.9869	48.7775
Speech s2	34.6346	39.4966

Tableau IV.5. AVCC pour le GA+ Kurtosis et algorithme

	GA+ kurtosis(Sources estimés) [IV.21]		Proposed algorithm (Sources estimés)	
	y1	y2	y1	y2
Speech s1	0.9996	0.0185	1.0000	0.0071
Speech s2	0.0282	0.9998	0.0071	0.9999

IV.7.1 Séparation des signaux en présence de bruit

Le but de cette section est d'évaluer la robustesse de notre méthode face au bruit. Pour cela on va introduire dans le mélange des sources, avec un bruit aléatoire dans un intervalle $[-1,1]$.

Simulation 4 : Les signaux poème-mâle_30s.wav et sentence_female_28s.wav et leurs signaux estimés sont montrés dans figure IV.7.

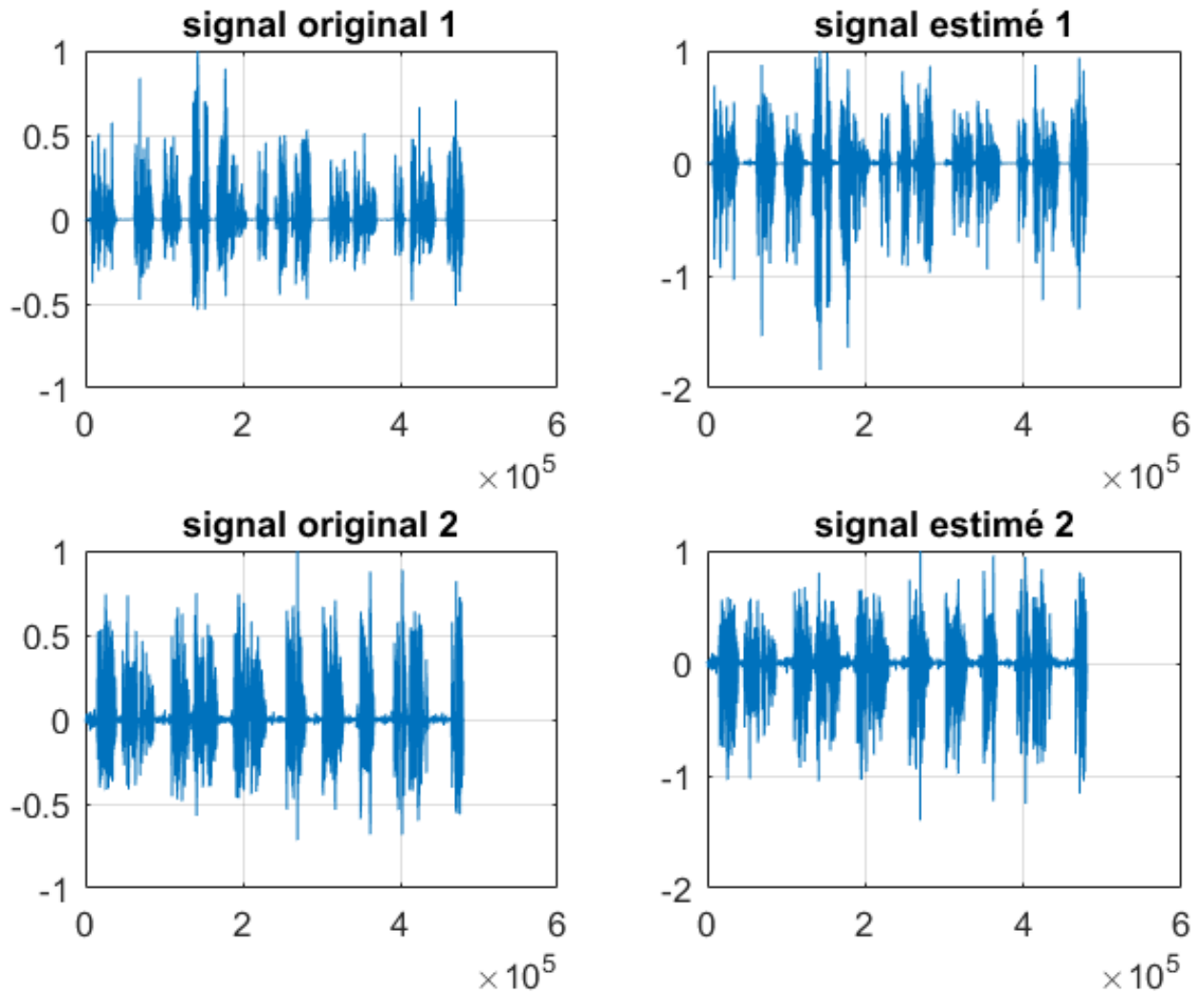


Figure IV.7. Signaux poème-mâle_30s.wav et sentence_female_28s.wav

Simulation 5 : Les signaux music_Loophass_30s.wav et henry_theater_male_30s.wav et leurs estimés sont montrés sur figure IV.8.

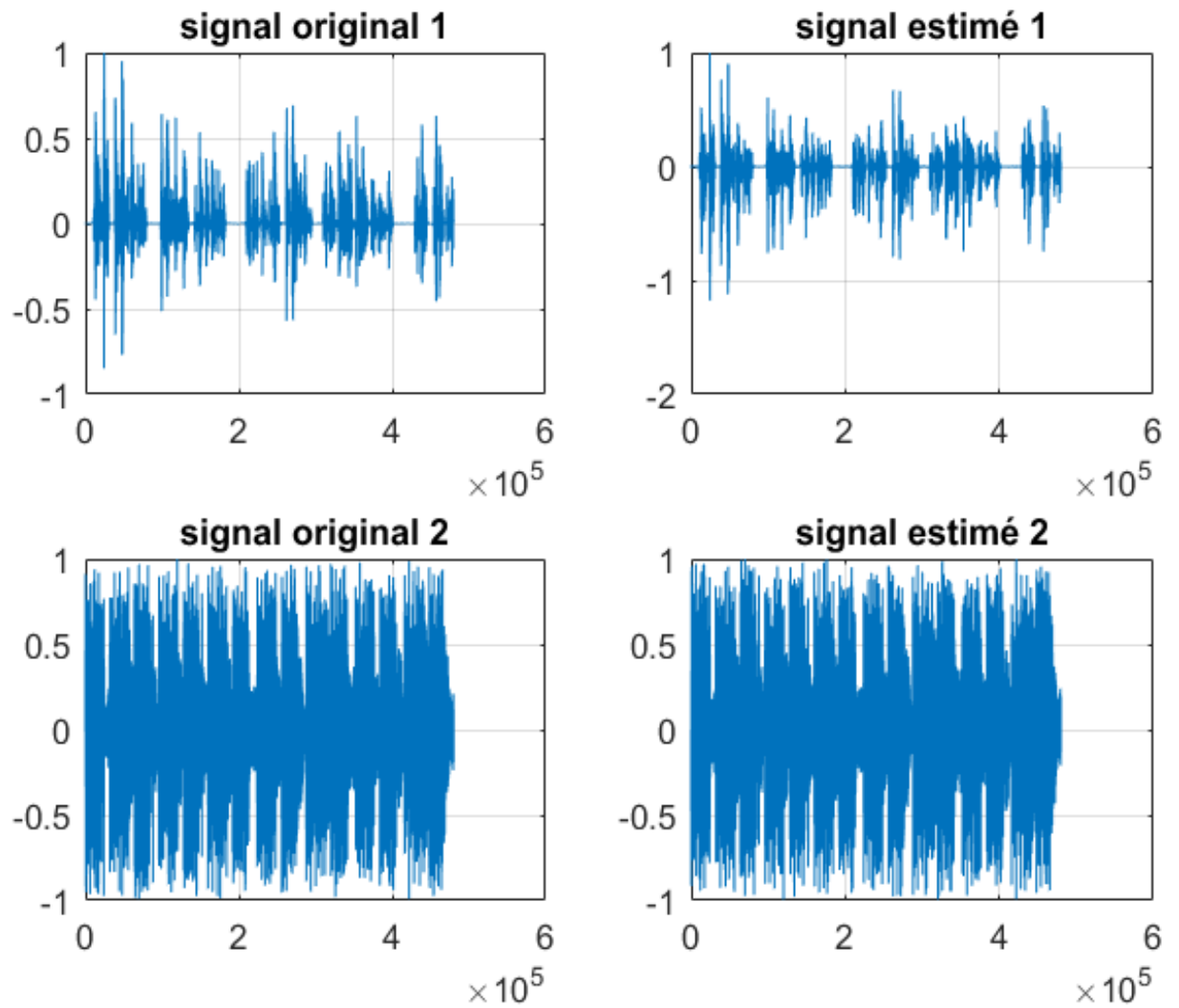


Figure IV.8. Signaux originaux et estimés de la musique loophass.wav et du théâtre henry.male.wav

Simulation 6 : Deux signaux de music_mandolin_30s.wav et music_vivaldi_30s.wav ainsi que leurs fichiers estimés sont illustrés sur la figure IV.9.

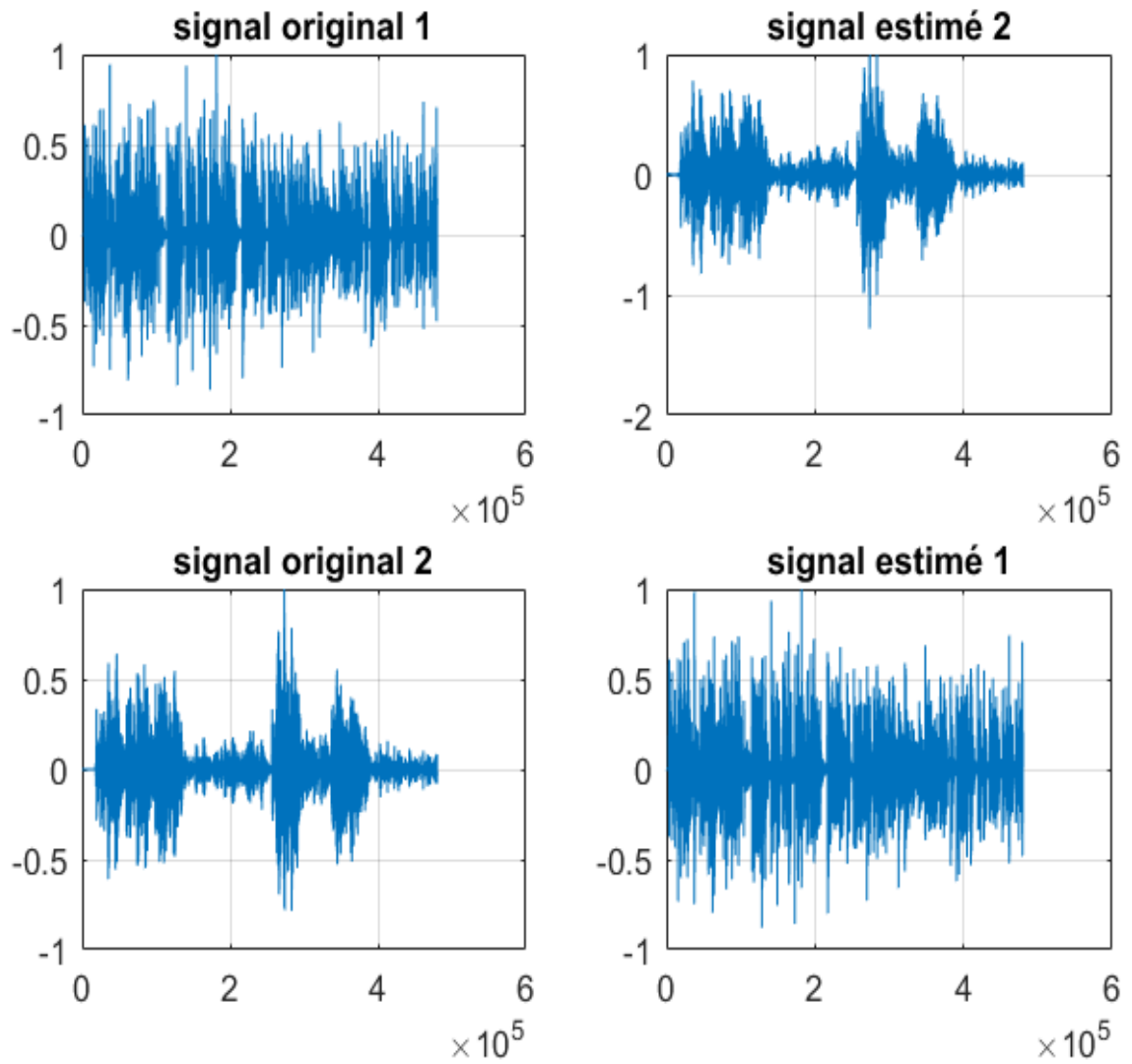


Figure IV.9. Signaux originaux et estimés de music_mandolin_30s.wav et music_vivaldi_30s.wav.

Tableau IV.6. SDR (dB) pour l'algorithme HEPSO en présence de bruit

Simulations	HEPSO en présence du bruit
Simulation 4	
Speech s1	26.2187
Speech s2	27.5926
Simulation 5	
Speech s3	31.8114
Music s4	28.5151
Simulation 6	
Music s5	29.5965
Music s6	27.4676

Tableau IV.7. AVCC pour HPSO+Bruit

Simulations	HEPSO en présence du bruit (Sources estimées)	
	y1	y2
Simulation 4		
Speech s1	0.9988	0.0454
Speech s2	0.0525	0.9991
Simulation 5		
Speech1 s3	0.9997	0.0375
Music s4	0.0258	0.9993
Simulation 3		
Music s5	0.0326	0.9989
Music s6	0.9994	0.0491

Tableau IV8. MSE pour HEPSO+Bruit

Simulations	HEPSO en présence du bruit
Simulation 4	
Speech s1	0.0041
Speech s2	0.0021
Simulation 5	
Speech1 s3	1.6153e-04
Music s4	1.2494e-04
Simulation 6	
Music s5	2.6582e-04
Music s6	5.8321e-04

IV.7.2 Interprétations

Les résultats de la séparation montrent que les signaux estimés sont similaires aux signaux originaux. Les résultats simulation ont montré l'efficacité de cette méthode qui permet de réduire le bruit aléatoire. Les performances restent acceptables.

IV.8 Conclusion

Dans ce chapitre, une séparation aveugle de sources audio utilisant la méthode HEPSO a été mise en œuvre et étudiée expérimentalement. En utilisant HEPSO, les sources distinctes ont été estimées à partir des mélanges observés. Lors de l'évaluation des performances du système HEPSO, la démonstration SAS, qui contient plusieurs exemples de signaux de parole et de musique, a été mise en œuvre. Les résultats de la simulation montrent que l'algorithme proposé HEPSO résout le problème des minima locaux et offre de meilleures performances en termes d'optimalité globale par rapport à l'algorithme PSO standard. Les résultats de performance démontrent clairement que l'algorithme HEPSO surpasse GA + kurtosis et permet également une bonne séparation pour identifier les signaux de source même en présence de bruit.

Dans nos travaux futurs, nous prévoyons d'examiner d'autres méthodes d'optimisation telles que l'algorithme Firefly (FA) afin d'améliorer la précision de leurs performances

Bibliographie

- [IV.1] Zhao, W., Wei, Y., Shen, Y., CAO, Y., Yuan, Z., Xu, P., and Jian, W. "An efficient algorithm by kurtosis maximization in reference-based framework". *Radio engineering*, 2015, vol. 24, no. 2, p. 544-551.
- [IV.2] ZHAO, W., WEI, Y. M., SHEN, Y. H., YUAN, Z. G., XU, P. C., JIAN, W. "A novel Fast ICA method for the reference-based contrast functions". *Radio engineering*, 2014, vol. 23, no. 4, p. 1221–1225.
- [IV.3] XU, Peng-cheng, SHEN, Yue-hong, et HUI, L. I. "New Negentropy Optimization Schemes for Blind Signal Extraction of Complex Valued Sources". *Radio engineering*, 2015, vol. 24, no 1. 3, SEPTEMBER (2013) pp 861-865.
- [IV.4] Jiangong Wang · Yuehong Shen · Hui Li · Hongyan Wei · Shiming Xu, "A Separation Algorithm for Sources with Temporal Structure Only Using Second-order Statistics," September 2013, *Radioengineering* 22(3):861-865.
- [IV.5] Zhao, Yongjian, JIANG, Bin, JIANG, Haining, et al. "Separating Signals with Specific Temporal Structure". In: 2017 International Conference on Computer Network, Electronic and Automation (ICCNEA). IEEE, 2017. p. 355-359.
- [IV.6] Ababneh, Jehadet IGUAL, Jorge. "Biogeography-based Optimization Algorithm for Independent Component Analysis". In: The International Conference on Computing Technology and Information Management (ICCTIM). Society of Digital Information and Wireless Communication, 2014. p. 348.
- [IV.7] Huang, Po-Sen, Kim, Minje, Hasegawa-Johnson, Mark, et al. "Deep learning for monaural speech separation". In: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE, 2014. p. 1562-1566.
- [IV.8] Huang, Po-Sen, KIM, Minje, Hasegawa-Johnson, Mark, et al. "Joint optimization of masks and deep recurrent neural networks for monaural source separation". *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2015, vol. 23, no 12, p. 2136-2147.
- [IV.9] ZHEN, Liangli, PENG, Dezhong, YI, Zhang, et al. "Underdetermined blind source separation using sparse coding". *IEEE transactions on neural networks and learning systems*, 2017, vol. 28, no 12, p. 3102-3108.
- [IV.10] Y. Yang, X. Wang, D. Zhang, "Blind Source Separation Research Based on the Feature Distance Using Evolutionary Algorithms", *International Journal of Acoustics and Vibration*, 2014, Vol. 19, No 4. 1 DOI:0.20855/ijav.2014.19.4360.
- [IV.11] P. Zheng, Y. Liu, L. Tian, Y. Cao, "A Blind Source separation Method Based on Diagonalization of correlation Matrices and Genetic Algorithm", *Fifth world Congress*, June 2004 Vol.3, p.2127-2131.
- [IV.12] H. Zhou, C. Z. Chen, X. M. Sun, H. Liu, "Research on Blind Source Separation Algorithm Based on Particle Swarm Optimization", *advanced Materials Research*, 2014, Vols.989-994, p. 1566-1569.
- [IV.13] P. K. Tripathi, et al. "Multi-Objective Particle Swarm optimization with time variant inertia and acceleration coefficients", *Information Sciences*, 2007 vol. 177, p. 5033-5049.

- [IV.14] A.Ebrahimzadeh n, S.Mavaddati, "A novel technique for blind source separation using bees colony algorithm", Elsevier Swarm and Evolutionary Computation ,2014,15–20.
- [IV.15] M.J. Mahmoodabadi, Z. SalahshoorMottaghi, A. Bagheri. "HEPSO: High exploration particle swarm Optimization", information science 2014, 101- 111.
- [IV.16] R.Niva Das, Aurobinda Routray, Pradipta K. Dash. "ICA Methods for Blind Source Separation of Instantaneous Mixtures: A Case Study", Neural Information Processing–Letters and Reviews, November 2007Vol. 11, No.11.
- [IV.17] Mazur and A. Mertins, "A sparsity-based the criterion for Solving the permutation ambiguity in convolutive blind source separation", in Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing, Prague, Czech Republic, May 2011,p. 1996– 1999.
- [IV.18] Clerc, Maurice. L'optimisation par essaim particulaire. Technique et Science Informatiques, 2002, vol. 21, no 7, p. 941-964.
- [IV.19] Khalfa, Ali, et al. Blind Audio Source Separation Based On High Exploration Particle Swarm Optimization. KSII Transactions on Internet & Information Systems, 2019.
- [IV.20] Mavrovouniotis, Michalis, LI, Changhe, et YANG, Shengxiang. "A survey of swarm intelligence for dynamic optimization: Algorithms and applications". Swarm and Evolutionary Computation, 2017, vol. 33, p. 1-17.
- [IV.21] Chien-Ching Chiu, Chien-Hung Chen & Yu-Sheng Fan, "Image Reconstruction of a Buried Conductor by Modified Particle Swarm Optimization", IETE Journal of Research, 2012, 58:4, 284-291.
- [IV.22] W.D. Chang, "A multi-crossover genetic approach to multivariable PID controllers tuning", Expert Syst. Appl. 2007; 620–626.
- [IV.23] B. Akay, D. Karaboga, "A modified artificial bee colony algorithm for real-parameter optimization", Inf. Sci. 2012, 120–142.
- [IV.24] T.D. Seeley, "The Wisdom of the Hive: The Social Physiology of Honey Bee Colonies",Harvard University Press,

Conclusion générale et perspectives

Dans cette thèse nous avons fait l'étude de la séparation aveugle de sources audio. Tout d'abord nous avons relaté l'histoire de la SAS et étudié la théorie existante. Nous avons dévoilé ce que c'est que la séparation de source à partir de certains types de mélanges, puis présenté les caractéristiques des signaux audio puisque toutes nos simulations ont été conduites sur ces derniers et qui ont donc été nécessaires pour exploiter les méthodes SAS étudiées et une développée dans cette thèse. Deux techniques ont été proposées au chapitre II et qui sont les techniques Infomax et RobustICA. Nous avons montré que ces méthodes donnent des résultats de simulation acceptables. Nous avons ensuite présenté dans le troisième chapitre la théorie détaillée des types d'algorithmes évolutionnaires qui peuvent améliorer considérablement les techniques SAS. Dans le dernier chapitre, nous avons effectué une étude sur la technique proposée HEPSO. Cette technique traite le problème rencontré dans l'algorithme du gradient descent. Elle permet d'éviter les minima locaux, lors de l'application de la séparation des sources. Les résultats de simulation ont montré une considérable amélioration des performances par rapport à l'algorithme PSO standard même en présence de bruit.

Des perspectives futures de ce travail seront pris en considération et développées. On peut citer les points suivants :

- Faire une étude approfondie visant à explorer l'utilisation des réseaux de neurones dans le contexte de la séparation multicanale de sources en mouvement.
- Appliquer la technique SAS proposée sur les modèles de mélanges non linéaires.
- Etendre la méthode développée à d'autres signaux, tels que les signaux et images ECG, à spectre étalé.
- Appliquer la technique proposée avec un nombre de sources supérieur à deux.
- Appliquer la technique proposée au problème de la SAS sous-déterminé, i.e. le nombre de signaux mixtes est inférieur à celui des sources.