N° d'ordre: 3504

THÈSE

Présentée devant

devant l'Université de Rennes 1

pour obtenir

le grade de : DOCTEUR DE L'UNIVERSITÉ DE RENNES 1
Mention TRAITEMENT DU SIGNAL ET TÉLÉCOMMUNICATIONS

par

Alexey OZEROV

Équipe d'accueil : METISS/IRISA
École Doctorale : MATISSE
Composante universitaire : SPM

Titre de la thèse :

Adaptation de modèles statistiques pour la séparation de sources mono-capteur
Application à la séparation voix / musique dans les chansons

soutenue le 15 décembre 2006 devant la commission d'examen

Mme.	:	Régine	ANDRÉ-OBRECHT	Université Toulouse 3	Rapporteurs
M.	:	Eric	MOULINES	ENST Paris	
M.	:	Bernard	DELYON	Université Rennes 1	Président
M.	:	Pierrick	PHILIPPE	France Télécom R&D	Examinateurs
M.	:	Rémi	GRIBONVAL	IRISA	
M.	:	Frédéric	BIMBOT	IRISA	Directeur de thèse
M.	:	Gaël	RICHARD	ENST Paris	Invité

Remerciements

Je voudrais tous d'abord remercier les trois personnes qui m'ont encadré pendant ces trois ans à France Télécom R&D et à l'IRISA. Je remercie Frédéric Bimbot, chargé de recherche à l'IRISA et responsable scientifique de l'équipe METISS, pour m'avoir bien encadré au niveau scientifique, ainsi qu'au niveau administratif. En particulier, je lui suis reconnaissant de m'avoir encouragé à généraliser certains concepts développés dans ma thèse. Je tiens à remercier également Pierrick Philippe, ingénieur de recherche à France Télécom R&D, pour m'avoir proposé de travailler sur ce sujet passionnant et m'avoir guidé durant ces travaux. Ses conseils m'ont permis de garder les pieds sur terre et de prendre l'habitude d'essayer toujours d'exprimer mon travail de manière simple, ce qui m'a beaucoup aidé pour la rédaction de cette thèse. Enfin, je remercie bien évidemment Rémi Gribonval, chargé de recherche à l'IRISA, avec qui j'ai pu discuter plus en détails les aspects théoriques et pointus de ma thèse. De plus, je lui suis reconnaissant de m'avoir invité à quelques rencontres scientifiques où j'ai fait connaissance avec d'autres chercheurs et enrichi mes connaissances.

Je suis tout particulièrement reconnaissant aux rapporteurs de cette thèse, Régine André-Obrecht et Eric Moulines, pour l'attention qu'ils ont porté à ces travaux. Je remercie Bernard Delyon, qui a bien voulu présider le jury de soutenance de cette thèse, ainsi que Gaël Richard, d'avoir bien voulu examiner à ce travail.

Dans cette thèse, je me suis beaucoup basé sur les thèses de Laurent Benaroya et Emmanuel Vincent, et je les remercie tous les deux pour quelques discussions scientifiques qui m'ont permis d'éclaircir leurs travaux. Merci beaucoup à mes amis musiciens, Olivier LeBlouch et Ewen Camberlein, pour m'avoir aidé à collecter les données d'évaluation utilisées tout le long de ce travail, et à Michaël Betser pour m'avoir donné plusieurs fois un coup de main pour mes articles. Merci à Gilles Gonon d'avoir créé une belle interface de démonstration de remixage de sources, que j'ai pu utiliser pour ma soutenance. Enfin, je remercie Elodie pour la relecture de ma thèse et de mes articles, ainsi que pour son soutien au quotidien.

Je conclus en remerciant toutes les personnes de France Télécom R&D à Rennes et de l'équipe METISS de l'IRISA avec qui j'ai travaillé ou bien juste passé des moments agréables. Merci à Jean-Bernard, Patrice, Sébastian, Alexandra, Julie, Typhaine, Chen, Jingqiang, Oxana, Iryna, Julien, Guillaume, Sylvain, Mathieu, Boris, Benjamin, Simon, Amadou, Mikael, Sacha, Ewa, Lorcan et à tous les autres que je n'ai pas mentionnés car il ne reste plus la place sur cette page.

Résumé

La séparation de sources avec un seul capteur est un problème très récent, qui attire de plus en plus d'attention dans le monde scientifique. Cependant, il est loin d'être résolu et, même plus, il ne peut pas être résolu en toute généralité. La difficulté principale est que, ce problème étant extrêmement sous déterminé, il faut disposer de fortes connaissances sur les sources pour pouvoir les séparer. Pour une grande partie des méthodes de séparation, ces connaissances sont représentées par des modèles statistiques des sources, notamment par des Modèles de Mélange de Gaussiennes (MMG), qui sont appris auparavant à partir d'exemples.

L'objet de cette thèse est d'étudier les méthodes de séparation basées sur des modèles statistiques en général, puis de les appliquer à un problème concret, tel que la séparation de la voix par rapport à la musique dans des enregistrements monophoniques de chansons. Apporter des solutions à ce problème, qui est assez difficile et peu étudié pour l'instant, peut être très utile pour faciliter l'analyse du contenu des chansons, par exemple dans le contexte de l'indexation audio.

Les méthodes de séparation existantes donnent de bonnes performances à condition que les caractéristiques des modèles statistiques utilisés soient proches de celles des sources à séparer. Malheureusement, il n'est pas toujours possible de construire et d'utiliser en pratique de tels modèles, à cause de l'insuffisance des exemples d'apprentissage représentatifs et des ressources calculatoires.

Pour remédier à ce problème, il est proposé dans cette thèse d'adapter *a posteriori* les modèles aux sources à séparer. Ainsi, un formalisme général d'adaptation est développé. En s'inspirant de techniques similaires utilisées en reconnaissance de la parole, ce formalisme est introduit sous la forme d'un critère d'adaptation Maximum *A Posteriori* (MAP). De plus, il est montré comment optimiser ce critère à l'aide de l'algorithme EM à différents niveaux de généralité.

Ce formalisme d'adaptation est ensuite appliqué dans certaines formes particulières pour la séparation voix / musique. Les résultats obtenus montrent que pour cette tâche, l'utilisation des modèles adaptés permet d'augmenter significativement (au moins de 5 dB) les performances de séparation par rapport aux modèles non adaptés. Par ailleurs, il est observé que la séparation de la voix chantée facilite l'estimation de sa fréquence fondamentale (pitch), et que l'adaptation des modèles ne fait qu'améliorer ce résultat.

Abstract

Single channel source separation is a quite recent problem of constantly growing interest in the scientific world. However, this problem is still very far to be solved, and even more, it cannot be solved in all its generality. Indeed, since this problem is highly underdetermined, the main difficulty is that a very strong knowledge about the sources is required to be able to separate them. For a grand class of existing separation methods, this knowledge is expressed by statistical source models, notably Gaussian Mixture Models (GMM), which are learned from some training examples.

The subject of this work is to study the separation methods based on statistical models in general, and then to apply them to the particular problem of separating singing voice from background music in mono recordings of songs. It can be very useful to propose some satisfactory solutions to this problem, which is quite difficult and has not been much studied yet, in order to simplify an automatic analysis of songs contents, for example in the context of audio indexing.

The existing model-based methods give satisfactory separation performances, provided that the models of the sources match accurately the statistical properties of the mixed signals. However, because of the shortage of representative training data and of calculation resources, it is not always possible to construct and use such models in practice.

To overcome this problem, it is proposed in this work to resort to an adaptation scheme which, for each recording, adjusts the source models to the properties of the signals observed in the mix. A general formalism for source model adaptation is developed. In a similar way as it is done for instance in speaker (or channel) adaptation for speech recognition, this formalism is introduced in terms of a Maximum *A Posteriori* (MAP) adaptation criterion. It is then shown how to optimize this criterion using the EM algorithm at different levels of generality.

This adaptation formalism is then applied in some particular forms to the voice / music separation task. The obtained results show that for this task an adaptation scheme can significantly improve (at least by 5 dB) the separation performance in comparison with non-adapted models. In addition, it is observed that the singing voice separation simplifies its fundamental frequency (pitch) estimation, and that the model adaptation leads to a further improvement of this result.

Table des matières

Table des matières 1

Liste des notations 7

Liste des abréviations 13

Table des figures 15

Liste des tableaux 17

Liste des algorithmes 19

Introduction 21

I Cadre du travail 29

1 Séparation de sources audio 31

1.1 Introduction au niveau acoustique 31

1.2 Formulation pour des signaux numériques 32

1.2.1 Modèles de mélange 33

1.2.2 Formulation du problème de la SSA 34

1.3 Pourquoi sépare-t-on ? 34

1.4 Classification des problèmes de la SSA par niveau de difficulté 36

1.5 Conclusion 37

2 Séparation de sources avec un seul capteur 39

2.1 Présentation intuitive 40

2.1.1 Hypothèse de travail : faible recouvrement dans le domaine de Fourier . . 40

2.1.2 Masquage temps - fréquence 42

2.1.3 Masquage oracle 42

2.1.4 Exemple de construction d'un masque 43
2.1.5 Exemple d'algorithme de séparation 44
2.2 Méthodes basées sur des modèles *a priori* : état de l'art 44
2.2.1 Réseaux bayésiens (modèles graphiques orientés) 47
2.2.2 Méthodes basées sur les MMG / MMC 49
2.2.2.1 Quelques remarques sur les MMG 49
2.2.2.2 Quelques remarques sur les MMC 50
2.2.3 Méthodes similaires pour le débruitage de la parole avec un seul capteur . 50
2.2.4 Extensions des méthodes basées sur les MMG et les MMC 51
2.2.5 Autres modèles 53
2.2.6 Modèles utilisés dans cette thèse 53
2.3 Méthodes basées sur des modèles probabilistes *a priori* : présentation technique générale 54
2.3.1 Domaine du traitement 54
2.3.2 Apprentissage de modèles 56
2.3.3 Estimation de sources 56
2.4 Méthodes de séparation basées sur des Modèles de Mélange de Gaussiennes (MMG) 57
2.4.1 Modélisation des spectres par des MMG 58
2.4.1.1 Apprentissage des MMG spectraux 59
2.4.1.2 Estimateur minimisant l'EQM spectrale 59
2.4.1.3 Estimateur dur vs. estimateur doux 60
2.4.1.4 Estimateur minimisant l'EQM log spectrale 61
2.4.1.5 Une remarque sur la phase 61
2.4.2 Modélisation des log spectres par des MMG 62
2.4.2.1 Apprentissage des MMG log spectraux 63
2.4.2.2 Distribution approchée du log spectre de mélange 63
2.4.2.3 Estimateur minimisant l'EQM log spectrale 65
2.5 Conclusion 67

3 Evaluation et diagnostic **69**
3.1 Evaluation de la qualité de séparation 69
3.2 Mesures de performance de séparation 71
3.2.1 Mesures héritées du débruitage de la parole 71
3.2.2 Mesures pour la séparation de sources 72
3.2.3 Mesures normalisées 72
3.3 Estimateurs oracles et limites de performance 74

3.4 Résumé 75

4 Expérimentations préliminaires dans le cadre de la séparation voix / musique 77

4.1 Problème de séparation voix / musique 77

4.2 Objectifs des expérimentations préliminaires 78

4.3 Mesures de performance 78

4.4 Description des données expérimentales pour la séparation 78

4.5 Expérimentations et résultats 79

4.5.1 Choix de la fenêtre d'analyse 79

4.5.2 Effet de l'hétérogénéité entre données d'apprentissage et de test et effet du dimensionnement des modèles 79

4.5.3 Effets du domaine de modélisation et de la mesure de distorsion 81

4.5.4 Précision des estimateurs durs par rapport aux estimateurs doux 85

4.6 Conclusion 86

5 Problématique 89

5.1 Limites des modèles probabilistes *a priori* 89

5.2 L'adaptation comme solution 91

II Adaptation des modèles : développement d'un formalisme général 93

6 Formalisme d'adaptation 95

6.1 Cahier des charges pour l'adaptation 95

6.2 Formalisme d'adaptation basé sur le critère MAP 96

6.2.1 Représentation à l'aide des réseaux bayésiens 98

6.2.2 Rôle des lois *a priori* 100

6.2.3 Positionnement par rapport à l'état de l'art 102

6.3 Conclusion 103

7 Algorithme d'adaptation 105

7.1 Algorithme d'adaptation sous sa forme générale 105

7.2 Algorithme d'adaptation pour les familles exponentielles 106

7.3 Statistiques naturelles des MMG et leurs espérances conditionnelles 108

7.4 Conclusion 109

8 Extensions du formalisme d'adaptation 111

8.1 Adaptation contrainte 111

8.2 Utilisation d'informations auxiliaires 113

8.3 Conclusion 115

III Application d'adaptation à la séparation voix / musique 117

9 Système de séparation voix / musique 119

9.1 Système de séparation 120

9.2 Description du module d'adaptation 120

9.3 Segmentation en parties vocales et non vocales 121

9.4 Adaptation acoustique du modèle de musique 122

9.4.1 Illustration expérimentale 124

9.4.2 Explication du réapprentissage sur les parties non vocales à l'aide du formalisme d'adaptation 127

9.5 Adaptation des filtres et des gains de DSP 130

9.5.1 Adaptation d'un filtre 130

9.5.2 Adaptation des gains de DSP 131

9.5.3 Adaptation conjointe des filtres et des gains de DSP 133

9.6 Conclusion 134

10 Intégration de l'adaptation des filtres et des gains lors de l'apprentissage du modèle général 137

10.1 Apprentissage du modèle général à filtres adaptés 137

10.2 Illustration expérimentale 138

10.3 Apprentissage prenant en compte l'adaptation contrainte 140

10.4 Conclusion 142

IV Evaluation du système de séparation voix / musique 143

11 Segmentation en parties vocales et non vocales 145

11.1 Description des données expérimentales pour la segmentation 145

11.2 Protocole expérimental 145

11.3 Mesure de performance 146

11.4 Paramètres acoustiques 146

11.5 Simulations 146

11.5.1 Décision par trame vs. décision par bloc, taille du bloc 147

11.5.2 Nombre d'états des modèles 147

11.6 Conclusion 147

12 Séparation voix / musique **151**
12.1 Protocole expérimental 151
12.2 Simulations 151
12.2.1 Seuil de décision de la segmentation automatique 152
12.2.2 Apport des différentes adaptations 152
12.2.3 Effet du nombre d'états des modèles 154
12.3 Conclusion 156

13 Apport de la séparation pour l'estimation du pitch de la voix **159**
13.1 Estimateur de pitch 159
13.2 Description des données expérimentales 159
13.3 Protocole expérimental 160
13.4 Pitch de référence 160
13.5 Mesures de performance 160
13.5.1 Mesures de performance : Option 1 160
13.5.2 Mesures de performance : Option 2 161
13.6 Simulations 161
13.6.1 Expérimentations avec les données utilisées pour la séparation 161
13.6.2 Expérimentations avec les données de l'ISMIR 2004 163
13.7 Conclusion 165

V Conclusion et perspectives **169**

14 Conclusion **171**

15 Perspectives **173**

Annexes **176**

A Rappels de probabilités et de statistiques **179**
A.1 Densité d'un vecteur aléatoire gaussien réel / complexe 179
A.2 Familles exponentielles et statistiques naturelles 180
A.3 Algorithme EM pour l'estimation MAP 181
A.3.1 Cas particulier des familles exponentielles 182

B Démonstration de certains résultats **183**
B.1 Familles exponentielles des MMG 183

B.2 Calcul des espérances conditionnelles des statistiques naturelles des MMG 184
B.3 Formules de réestimation pour l'adaptation des filtres et des gains de DSP 185
B.3.1 Adaptation d'un filtre 185
B.3.2 Adaptation des gains de DSP 186
B.3.3 Adaptation conjointe des filtres et des gains de DSP 186

Bibliographie **194**

Liste des notations

Quelques conventions

Les accolades et les crochets. Les accolades $\{\cdot\}$ sont utilisées pour noter des ensembles, par exemple $\omega = \{\omega_i\}_{i=1}^{Q}$ est un ensemble. Les crochets $[\cdot]$ sont utilisés pour noter des vecteurs et des matrices, par exemple $s = [s(n)]_{n=1}^{N}$ est un vecteur et $S = [S(t,f)]_{t,f=1}^{T,F}$ est une matrice. Parfois, dans les cas où il est clair, d'après le contexte, quelles valeurs peuvent prendre les indices (par ex. $i = 1, 2, \ldots, Q$), on ne précise pas explicitement leurs ensembles de variation, et on écrit plus simplement $\omega = \{\omega_i\}_i$ ou $S = [S(t,f)]_{t,f}$.

Les signaux temporels et leurs TFCT. Les signaux temporels sont des vecteurs réels et sont notés par des lettres minuscules, par ex. $s = [s(n)]_n$. Les Transformées de Fourier à Court Terme (TFCT) des signaux temporels sont des matrices complexes et sont notées par les lettres majuscules correspondantes, par ex. $S = [S(t,f)]_{t,f}$ est la TFCT du signal s. De plus, quand la TFCT est écrite avec un seul indice temporel t (par ex. $S(t)$), ceci signifie le spectre à court terme, c'est-à-dire le vecteur $S(t) = [S(t,f)]_f$.

Les indices des sources ($k = 1, 2$ ou $k = v, m$). Quand le problème de séparation de sources avec un seul capteur est traité de manière générale, les indices 1 et 2 sont utilisés pour étiqueter les sources ($k = 1, 2$). Cependant, dès qu'il s'agit de la séparation voix / musique, ces indices sont remplacés par v et m ($k = v, m$).

Les densités de probabilité des variables aléatoires. Nous utilisons des notations génériques pour les densités ou densités conditionnelles des variables aléatoires. Notamment, toutes les densités conditionnelles sont notées par $p(\cdot|\cdot)$ et il est clair en fonction des arguments avant et après la barre conditionnelle de quelles variables il s'agit. Par exemple, soit $\mathcal{A}$ et $\mathcal{B}$ des variables aléatoires et A et B leurs réalisations, alors $p(A|B) = p_{\mathcal{A}|\mathcal{B}=B}(A)$ est la densité de $\mathcal{A}$ calculée au point A, conditionnellement au fait que $\mathcal{B} = B$. Notons également que pour des raisons de compacité de la présentation, nous confondons souvent dans cette thèse, comme beaucoup d'auteurs, les notations des variables aléatoires avec celles de leurs réalisations (par ex. $\mathcal{A}$ avec A ou $\mathcal{B}$ avec B).

Les espérances conditionnelles sont notées par $\mathbb{E}_A\left[\mathbf{f}(A)|B\right]$, où A est une variable aléatoire,

$\mathbf{f}(\cdot)$ est une fonction, et B est la réalisation d'une autre variable aléatoire $\mathcal{B}$. A en indice signifie que cette espérance est calculée par rapport à la variable aléatoire A pour éviter d'éventuelles confusions liées au fait que les notations des variables aléatoires sont confondues avec celles de leurs réalisations. Autrement dit, $\mathbb{E}_A\left[\mathbf{f}(A)|B\right] \triangleq \int_A \mathbf{f}(A)p(A|B)dA$.

Notations mathématiques usuelles

$\propto$	Symbole de la proportionnalité
$\triangleq$	Egalité par définition
$\bar{0}$	Vecteur réel ou complexe dont toutes les composantes sont nulles
Id	Transformation unitaire (c'est-à-dire $\mathrm{Id}(a) = a$)
$\mathrm{diag}[a]$	Matrice diagonale dont la diagonale est le vecteur a
a^T	Transposée du vecteur ou de la matrice a
a^H	Transposée-conjuguée du vecteur ou de la matrice complexe a
$\bar{a}$	Conjugué du nombre complexe a
$\Re a$	Partie réelle du nombre complexe a
$\Im a$	Partie imaginaire du nombre complexe a
$\|a\|$	Module du nombre complexe a ou valeur absolue du nombre réel
$\langle a, b\rangle$	Produit scalaire des vecteurs a et b
$a \times b$	Multiplication élément par élément des matrices ou des vecteurs a et b
$\\|a\\|_2$	Norme euclidienne (l_2) du vecteur a ou norme de Frobenius de la matrice a
$\angle a$	Argument (phase) du nombre complexe a
$\nabla \mathbf{f}(a)$	Gradient de la fonction vectorielle $\mathbf{f}(a)$
$\delta(\cdot)$	Distribution de Dirac
$\delta(\cdot, \cdot)$	Symbole de Kronecker
$\exp(a)$	Exponentielle du nombre a ou du vecteur a appliquée élément par élément
$\log(a)$	Logarithme du nombre a ou du vecteur a appliqué élément par élément
$P(a)$	Probabilité de l'événement a
$P(a\|b)$	Probabilité conditionnelle de l'événement a sachant l'événement b
$p(a)$	Densité de probabilité de la variable a
$p(a\|b)$	Densité de probabilité conditionnelle de la variable a sachant l'événement b
$\mathbb{E}_a\left[\mathbf{f}(a)\|b\right]$	Espérance conditionnelle de $\mathbf{f}(a)$ sachant b, calculée par rapport à la variable a
$\mathcal{U}(a, b)$	Loi uniforme d'une variable aléatoire réelle sur l'interval $[a, b]$
$\mathcal{N}(\mu, r^2)$	Loi normale (gaussienne) d'une variable aléatoire réelle (moyenne μ, variance r^2)
$N(a; \mu, R)$	Densité de probabilité du vecteur aléatoire gaussien réel a
$N_C(a; \mu, R)$	Densité de probabilité du vecteur aléatoire gaussien complexe circulaire a

$\Phi(\cdot)$	Fonction de répartition de la loi normale centrée de variance unitaire
$\#(a)$	Nombre d'éléments de l'ensemble a

Notations particulières utilisées

$a_{l,k}$	Gain de mixage (k-ème source, l-ème mélange) 33
$a_{l,k}(n)$	Réponse impulsionnelle du filtre de mixage (k-ème source, l-ème mélange) .. 33
A	Matrice de mélange 33
$\mathcal{A}$	Modèle de mélange 32
B	Paramètres acoustiques pour la segmentation 121
C_k	Paramètres libres 112
$\mathbf{C}_k$	Ensemble des paramètres libres 141
$d(\widehat{S}_k, S_k)$	Mesure de distorsion minimisée 56
$\mathcal{D}$	Transformée du domaine défini par $\mathcal{F}$ dans un domaine de minim. de l'EQM 54
$E_{v,z}$	TFCT du z-ème enregistrement de la base d'entraînement Y_v 137
η	Seuil de décision pour la segmentation en parties vocales et non vocales 122
f	Indice fréquentiel de la TFCT 40
F	Indice de la fréquence de Nyquist de la TFCT 40
$\mathcal{F}$	Transformée du domaine temporel dans un domaine de traitement 54
g_v ou g_m	Vecteur des gains de DSP adaptables 131
$g_v \bullet \Lambda_v$	Opération non standard d'application des gains de DSP g_v au MMG Λ_v ... 131
Γ_N	MMG des trames non vocales 121
Γ_V	MMG des trames vocales 121
$\mathcal{H}_v$ ou $\mathcal{H}_m$	Matrice diagonale d'un filtre adaptable 130
$\mathbf{H}_v$	Ensemble des filtres adaptables 138
i	Indice des états (DSP) d'un MMG de la 1-ère source ($k = 1$) 44
I	Informations auxiliaires 113
j	Indice des états (DSP) d'un MMG de la 2-ème source ($k = 2$) 44
k	Indice des sources 32
K	Nombre de sources 32
l	Indice des mélanges 32
L	Nombre de mélanges 32
$\mathcal{L}$	Transformée du domaine défini par $\mathcal{F}$ dans un domaine de modélisation 54
λ_k	Modèle (MMG) adapté de la k-ème source 95
λ_k^{Idl}	Modèle (MMG) idéal de la k-ème source (appris sur S_k) 79

$\tilde{\lambda}_m$	Modèle (MMG) de musique adapté acoustiquement	120
Λ_k	Modèle (MMG) général (ou *a priori*) de la k-ème source	46
Λ_v^C	MMG général de voix à paramètres libres adaptés	141
$\Lambda_v^{\mathcal{H}}$	MMG général de voix à filtres adaptés	138
m	Indice de la source de musique	77
$\mathcal{M}_k$	Masque temps - fréquence pour la k-ème source	42
$\mathbb{M}$	Ensemble de masques admissibles	74
$\mu_{k,i}$	Vecteur moyen d'un MMG général (k-ème source, i-ème état)	62
n	Indice du temps discret	32
$\omega_{k,i}$	Poids d'une gaussienne d'un MMG adapté (k-ème source, i-ème état)	96
Ω	Transformée reliant $\mathcal{Z}$ avec $\mathcal{X}$ (terminologie de EM)	181
Ψ_k	Déformation paramétrique	112
$q_k(t)$	Indice de l'état émis pour la trame numero t (k-ème source)	48
Q_k	Nombre de DSP (états) d'un modèle (MMG) de la k-ème source	44
$r_{k,i}^2(f)$	Variance d'un MMG général (élément diagonal de $R_{k,i}$)	44
$R_{k,i}$	Matrice de covariance diagonale d'un MMG général	58
ρ	Estimation du pitch de la voix chantée	159
s_k	k-ème source dans le domaine temporel	32
$\hat{s}_k$	Estimation de la k-ème source dans le domaine temporel	39
$s_{l,k}^{\mathrm{img}}$	Contribution de la k-ème source dans le l-ème mélange	32
S_k	TFCT de la k-ème source	40
$\widehat{S}_k$	Estimation de la TFCT de la k-ème source	42
$\widehat{\mathbb{S}}_k$	TFCT de l'estimation de la k-ème source	72
$\mathbf{S}_k$	Logarithme de la TFCT de la k-ème source	62
$\sigma_{k,i}^2(f)$	Variance d'un MMG adapté (élément diagonal de $\Sigma_{k,i}$)	96
$\Sigma_{k,i}$	Matrice de covariance diagonale d'un MMG adapté (source k, état i)	96
t	Indice temporel de la TFCT	40
$\mathbf{t}_{k,i}^0$	Statistique comptant le nombre d'observations d'un état	108
$\mathbf{t}_{k,i}^2(f)$	Statistique représentant l'énergie de la TFCT	108
T	Nombre total de trames de la TFCT d'un signal	40
$\mathbf{T}(\cdot)$	Statistique (naturelle)	181
τ	Facteur de confiance	123
θ	Paramètres estimés (terminologie de EM)	181
$u_{k,i}$	Poids d'une gaussienne d'un MMG général (k-ème source, i-ème état)	58
v	Indice de la source de voix chantée	77
voc	Ensemble des indices des trames vocales	120

x	Mélange monophonique dans le domaine temporel	39
x_l	l-ème mélange dans le domaine temporel	32
X	TFCT du mélange	40
$\mathcal{X}$	Données observées (terminologie de EM)	181
$\mathbf{X}$	Logarithme de la TFCT du mélange	63
y_k	Données d'entraînement pour la k-ème source dans le domaine temporel	43
Y_k	TFCT des données d'entraînement pour la k-ème source	44
$\mathbf{Y}_k$	Logarithme de la TFCT des données d'entraînement pour la k-ème source	64
$\mathcal{Z}$	Données complètes (terminologie de EM)	181

Liste des abréviations

ACI	Analyse en Composantes Indépendantes	36
AR	AutoRegressif (modèle)	51
ASI	Analyse en Sous-espaces Indépendants	46
CASA	Analyse Computationnelle de Scènes Auditives (*Comput. Audit. Scene Anal.*)	46
CMS	*Cepstral Mean Subtraction*	146
DET	*Detection Error Tradeoff*	146
DLS	Distorsion du Log Spectre	71
DLSN	DLS Normalisée	73
DSP	Densités Spectrales de Puissance	43
EER	*Equal Error Rate*	147
EM	*Expectation - Maximization*	23
EMLLR	*Eigenspace-Based MLLR*	102
EQM	Erreur Quadratique Moyenne	54
MAP	Maximum *A Posteriori*	23
MeanMAX	*Mean Maximum* (approximation)	65
MFCC	*Mel Frequency Cepstral Coefficients*	146
MIXMAX	*Mixture Maximum* (approximation)	65
MLLR	*Maximum Likelihood Linear Regression*	102
MMC	Modèle de Markov Caché	48
MMG	Modèle de Mélange de Gaussiennes	44
MV	Maximum de Vraisemblance	56
NMF	Factorisation en Matrices Non Négatives (*Non negative Matrix Factorisation*)	46
OLA	*OverLap and Add*	46
QV	Quantification Vectorielle	53
RSA	Rapport Source à Artefacts	72
RSB	Rapport Signal à Bruit	71
RSD	Rapport Source à Distorsion	72
RSI	Rapport Source à Interférences	72

RSDN RSD Normalisé 73
SAGE *Space-Alternating Generalized EM* 134
SMAP *Structural MAP* 102
SSA Séparation de Sources Audio 31
TER *Total Error Rate* (mesure d'erreur de la segmentation) 155
TFCT Transformée de Fourier à Court Terme 40
VFAR *Vocal False Alarm Rate* (mesure d'erreur de la segmentation) 146
VMER *Vocal Miss Error Rate* (mesure d'erreur de la segmentation) 146
VN *Variance Normalization* 146
VTS *Vector Taylor Series* (approximation) 65
WDO *W-Disjoint Orthogonality* 40

Table des figures

1.1 Scène sonore enregistrée par deux microphones. 32

2.1 Exemple de séparation de la voix chantée et du violon. 41
2.2 Schéma d'un algorithme de séparation utilisant des ensembles de DSP comme connaissances *a priori* sur les sources. 45
2.3 Méthodes basées sur des modèles *a priori* des sources. 47
2.4 Exemple d'un réseau bayésien. 47
2.5 Réseaux bayésiens de modèles de sources et des modèles de mélanges correspondants. 49
2.6 Illustration du débruitage de la parole et de la séparation de parole femme / homme en utilisant deux modèles : grossier (AR d'ordre 10) et fin (DSP). 52
2.7 Schéma général de la séparation de sources basée sur des modèles probabilistes *a priori*. 55
2.8 Séparation avec des méthodes basées sur des MMG. 58
2.9 MMG spectral à 16 états. 59
2.10 MMG log spectral à 16 états. 63
2.11 Comparaison des approximations pour des MMG log spectraux. 66

3.1 Deux types de procédures d'évaluation d'un système de séparation de sources. . . 70
3.2 Interprétation du RSDN pour la séparation de 5 enregistrements différents. . . . 73

4.1 Le comportement du RSDN moyen pour l'estimateur oracle en fonction de la taille et du type de fenêtre d'analyse de la TFCT. 80
4.2 RSDN moyen pour six chansons de test en fonction du nombre d'états des modèles et pour différents types de modèles. 81
4.3 RSDN détaillé pour six chansons de test en fonction du nombre d'états des modèles et pour différents types de modèles. 82
4.4 MMG généraux à 1 état et filtre de Wiener correspondant. 82
4.5 Quelques exemples de MMG à 16 états. 83

6.1 Séparation avec des modèles adaptés *a posteriori*. 99
6.2 Réseaux bayésiens correspondants aux processus aléatoires impliqués dans les procédures d'apprentissage des modèles, d'estimation des sources et d'adaptation *a posteriori* des modèles. 101

7.1 Algorithme EM pour l'optimisation du critère MAP (6.3) dans le cas des familles exponentielles. 108

8.1 Réseau bayésien représentant l'adaptation contrainte. 112
8.2 Réseau bayésien représentant la prise en compte des informations auxiliaires I dans la procédure d'adaptation des modèles. 115

9.1 Module d'adaptation pour la séparation voix / musique. 121
9.2 Segmentation en parties vocales et non vocales avec la décision par blocs. 123
9.3 RSDN moyen en fonction du logarithme à base 2 du facteur de confiance τ pour l'adaptation acoustique du modèle de musique. 126
9.4 Modèle bimodal et son approximation par un MMG à deux états. 128
9.5 Réseau bayésien correspondant à l'apprentissage du modèle de musique sur les parties non vocales. 128
9.6 Interprétation matricielle de l'adaptation d'un filtre et des gains de DSP. 132

10.1 Réseau bayésien correspondant à l'apprentissage prenant en compte l'adaptation contrainte. 141

11.1 Influence de la taille du bloc de décision sur les performances de segmentation en parties vocales et non vocales. 148
11.2 Influence du nombre d'états des modèles sur les performances de segmentation en parties vocales et non vocales. 149

12.1 Performances de séparation en fonction du seuil de la segmentation automatique. 153
12.2 RSDN moyen pour six chansons de test en fonction du nombre d'états des modèles et pour différents types de modèles. 156
12.3 RSDN détaillé pour six chansons de test en fonction du nombre d'états des modèles et pour différents types de modèles. 157
12.4 Performances de segmentation en parties vocales et non vocales pour chaque chanson de test. 157

A.1 Distribution d'une variable aléatoire gaussienne complexe circulaire. 180

Liste des tableaux

4.1 Performances des méthodes en fonction du modèle MMG et de l'EQM minimisée en utilisant les estimateurs **doux**. 85

4.2 Performances des méthodes en fonction du modèle MMG et de l'EQM minimisée en utilisant les estimateurs **durs**. 86

6.1 Cahier des charges pour l'adaptation des modèles. 96

10.1 Apport de la procédure d'apprentissage à filtres adaptés. 140

12.1 Importance d'adaptation des différentes combinaisons des paramètres. 154

13.1 Résultats moyens de l'estimation de pitch en utilisant les données pour la séparation et l'option 1 des mesures de performance. 163

13.2 Résultats moyens de l'estimation de pitch en utilisant les données pour la séparation et l'option 2 des mesures de performance. 163

13.3 Résultats de l'estimation de pitch détaillés pour chaque chanson de la base de test pour la séparation en utilisant l'option 1 des mesures de performance. 164

13.4 Résultats moyens de l'estimation de pitch en utilisant les données de l'ISMIR 2004 et l'option 1 des mesures de performance. 166

13.5 Résultats de l'estimation de pitch détaillés pour chaque extrait de la base de test de l'ISMIR 2004 en utilisant l'option 1 des mesures de performance. 166

Liste des algorithmes

1 Séparation de sources avec un seul capteur. 46
2 Algorithme EM pour l'apprentissage d'un MMG spectral. 59
3 Algorithme EM pour l'apprentissage d'un MMG log spectral. 64
4 Calcul des espérances conditionnelles des statistiques naturelles. 109
5 Adaptation conjointe des filtres et des gains de DSP. 135
6 Algorithme SAGE pour l'apprentissage du modèle général de voix à filtres adaptés.139

Introduction

Les scènes sonores sont souvent composées du mélange de sons (appelés ici *sources*) émis par plusieurs émetteurs sonores tels que des instruments musicaux, des personnes qui parlent, des bruits ambiants etc. Une scène sonore peut être enregistrée en utilisant un ou plusieurs microphones, ce qui correspond aux enregistrements audio monophoniques (1 microphone) ou stéréo (2 microphones). Ayant à sa disposition un enregistrement audio, qui est une sorte d'image de la scène sonore correspondante, on n'a plus accès à toutes les particularités de cette scène, notamment aux sources. Cependant, pour de nombreuses applications, il peut être très utile de pouvoir reconstituer dans la mesure du possible la scène sonore initiale pour pouvoir la modifier ou l'analyser.

Le problème de la séparation de sources audio consiste à séparer les sources, c'est-à-dire à les estimer à partir d'un enregistrement, à l'aide de l'ordinateur. D'un côté, ceci permettra de modifier la scène sonore correspondante, par exemple en modifiant les positions des émetteurs, en intervenant sur les intensités des sources ou en rajoutant de nouveaux effets spéciaux. De l'autre côté, ceci facilitera l'extraction automatique des informations sémantiques portées par chaque source, comme par exemple la partition musicale ou la parole. Notons cependant que les objectifs de la séparation de sources audio semblent être plus ambitieux que les capacités humaines concernant l'analyse d'une scène sonore à partir d'un enregistrement. En effet, un être humain qui a des connaissances en musique est capable en écoutant un enregistrement musical de prêter attention à un instrument particulier et d'en extraire certaines informations, par exemple la mélodie jouée ou la parole chantée, mais il n'est pas capable de vraiment les séparer.

Dans cette thèse, nous nous intéressons au problème de la séparation de sources audio en utilisant un seul capteur (microphone). Pour l'évaluation des techniques proposées, nous avons choisi une tâche particulière, la séparation de la voix chantée par rapport à la musique ambiante dans des chansons populaires. Bien séparer la voix de la musique peut être très utile pour l'indexation audio. En effet, il est plus facile d'extraire certaines métadonnées utilisées pour l'indexation (par exemple la mélodie chantée ou l'identité du chanteur) à partir de la voix seule qu'à partir de la voix noyée dans la musique.

Le problème de la séparation de sources avec un seul capteur est plus difficile que le problème de la séparation avec plusieurs capteurs, car avec un seul capteur la diversité spatiale, c'est-à-dire le fait que les sources proviennent de directions différentes, n'est pas exploitable. Pour expliquer l'utilisation de la diversité spatiale, considérons comme exemple la technique la plus basique d'élimination de la voix chantée dans des enregistrement stéréo (avec deux capteurs) de chansons. Cette technique est basée sur l'hypothèse que la voix est mixée au milieu, c'est-à-dire qu'elle intervient avec les mêmes intensités dans le canal gauche et le canal droit, et que les autres instruments interviennent avec des intensités différentes. Dans ce cas, une simple soustraction des deux canaux élimine parfaitement la voix grâce au fait que les sources viennent de directions différentes. Cependant, cette technique fonctionne rarement, car pour la plupart des chansons l'hypothèse utilisée n'est pas vérifiée.

A partir d'un enregistrement mono (avec un seul capteur), il n'est plus possible de déterminer les directions de provenance des sources. Même un être humain écoutant un enregistrement mono n'est pas capable de le faire. Ainsi, la diversité spatiale n'est plus utilisable pour séparer les sources et il faut avoir d'autres connaissances pour parvenir à les séparer. Plusieurs approches pour la séparation de sources avec un seul capteur ont été proposées récemment. Une grande partie de ces approches utilisent des modèles *a priori* de sources comme connaissance permettant de les séparer. Les modèles spectraux ou plus formellement les modèles probabilistes appelés Modèles de Mélange de Gaussiennes (MMG) sont souvent utilisés pour modéliser les sources.

L'idée de ces techniques est de représenter chaque source par un modèle spectral qui est un ensemble de formes spectrales typiques. Chaque modèle est appris sur une base d'entraînement qui doit être représentative de la classe sonore à laquelle la source est affectée (par exemple la parole, la musique, un instrument de musique particulier, etc.). Ces modèles sont appelés ici *modèles généraux*, car ils sont censés couvrir l'ensemble des propriétés observables pour les sources appartenant à la classe sonore correspondante. Enfin, les sources peuvent être estimées, étant donné l'enregistrement à séparer et les modèles.

Ces méthodes semblent très prometteuses. Cependant, elles souffrent de limitations majeures qui les rendent difficilement utilisables en pratique pour la séparation de sources appartenant à des classes sonores de grande variabilité. En effet, cette modélisation est assez fine, puisque chaque événement sonore doit être représenté par une forme spectrale typique. Par conséquent, pour bien modéliser des classes sonores de grande variabilité, une quantité importante de formes spectrales est nécessaire, c'est-à-dire qu'il y a besoin de modèles de grande taille et d'une quantité importante de données pour les apprendre. Ainsi, les problèmes suivants se posent :

1. Pour une classe sonore très riche, il devient difficile de construire une base d'entraînement représentative.
2. La complexité calculatoire peut devenir excessive à cause de la grande taille des modèles (par ex. des milliers de formes spectrales).

Considérons par exemple la musique, qui est une classe sonore d'une variabilité extrême. A la variabilité des conditions d'enregistrement s'ajoutent les différents instruments qui peuvent jouer des notes et des accords différents. En effet, il semble utopique d'essayer de décrire toute cette richesse par un ensemble de formes spectrales typiques. Ainsi, il paraît indispensable de proposer des solutions permettant de surmonter les limitations annoncées, en particulier pour la tâche de séparation voix / musique.

Dans cette thèse, nous proposons de recourir à l'*adaptation* des modèles qui permet dans certains cas de surmonter les limites d'utilisation des modèles généraux. Par exemple, on peut adapter des modèles généraux aux nouvelles conditions d'enregistrement de chaque source particulière. En s'inspirant des techniques d'adaptation de modèles utilisées en reconnaissance de la parole ou du locuteur, nous proposons donc d'ajuster les modèles de sources à leurs réalisations dans l'enregistrement. Les modèles obtenus ainsi sont appelés *modèles adaptés*.

L'idée d'*adaptation a posteriori* des *connaissances a priori* est assez naturelle, et nous la rencontrons partout dans la vie courante. L'être humain a souvent besoin d'adapter ses connaissances quand il se retrouve dans une situation nouvelle (ou bien déjà vécue, mais oubliée) et pour cela il a besoin d'un temps d'adaptation. Considérons par exemple quelqu'un qui commence à lire une lettre manuscrite avec une écriture qu'il n'a jamais vue avant. D'abord, il lit lentement, c'est la phase d'adaptation de ses connaissances *a priori* (la langue française, l'alphabet, etc.), et ensuite il commence à lire plus vite.

Cette thèse présente deux contributions principales. La première est plus théorique : elle consiste en un développement du concept d'adaptation des modèles pour la séparation de sources. Un formalisme général d'adaptation est introduit sous la forme d'un critère d'adaptation bayésienne Maximum *A Posteriori* (MAP), qui peut être optimisé à l'aide de l'algorithme EM (*Expectation-Maximisation*). Un effort particulier est apporté pour présenter les techniques proposées dans le cadre du formalisme des réseaux bayésiens.

La deuxième contribution, qui est plus applicative, a pour objet le développement d'un système de séparation voix / musique basé sur les principes d'adaptation proposés. Le module d'adaptation utilisé dans ce système repose sur les trois étapes suivantes :

- Segmentation de la chanson en parties vocales (avec voix chantée) et non vocales (sans voix chantée).
- Adaptation acoustique du modèle de musique sur les parties non vocales.
- Adaptation de certains paramètres (en particulier des conditions d'enregistrement) des modèles de voix et de musique sur toute la chanson.

Deux procédures d'évaluation ont été employées pour valider ce système et pour montrer l'intérêt de l'adaptation des modèles. Premièrement, ce système est évalué en utilisant une mesure objective de performance de séparation. Les résultats montrent que l'adaptation des modèles permet d'améliorer significativement (au moins de 5 dB) les performances de séparation par rapport aux modèles généraux. La deuxième procédure d'évaluation concerne l'extraction des métadonnées à partir de la voix séparée. Comme il a été remarqué auparavant, ceci peut être très utile pour l'indexation audio. Plus particulièrement, il s'agit de l'estimation du *pitch* de la voix chantée (c'est-à-dire de la fréquence fondamentale qui contient l'information sur la mélodie chantée) à partir de la voix séparée. Les résultats montrent que cette estimation est plus fiable que celle faite à partir de l'enregistrement non séparé. Ils montrent également que l'adaptation des modèles améliore l'estimation du pitch.

Plan de la thèse

Cette thèse se décompose en cinq parties :

- Partie I : Cadre du travail
- Partie II : Adaptation des modèles : cadre général
- Partie III : Application d'adaptation à la séparation voix / musique
- Partie IV : Evaluation
- Partie V : Conclusion et perspectives

suivies de deux annexes.

Partie I : Cadre du travail

Le but de cette partie est de présenter le sujet étudié dans cette thèse, ainsi que quelques premiers essais expérimentaux, afin de définir la problématique traitée par la suite.

Dans le chapitre 1, le problème de la séparation de sources audio est présenté de manière générale. Premièrement, cela permet de positionner le problème de séparation de sources avec un seul capteur par rapport au problème de séparation de sources avec plusieurs capteurs. Deuxièmement, cette présentation permet de comprendre quelles connaissances sont nécessaires pour pouvoir résoudre le problème de séparation de sources avec un seul capteur.

Dans le chapitre 2, le problème de séparation de sources avec un seul capteur est explicité. Ensuite, nous présentons un état de l'art assez exhaustif des méthodes de séparation basées sur des modèles probabilistes des sources, notamment des Modèles de Mélange de Gaussiennes (MMG).

Le chapitre 3 aborde la question d'évaluation et de diagnostic des algorithmes de séparation. Des mesures de performance de séparation y sont présentées et de nouvelles mesures normalisées sont développées. Ce chapitre se termine par la présentation des estimateurs oracles permettant de calculer les limites de performance des algorithmes de séparation.

Quelques expériences préliminaires dans le cadre de la séparation voix / musique sont présentées dans le chapitre 4. Ces expériences permettent d'identifier les points faibles des méthodes de séparation étudiées, ainsi que de fixer certains paramètres.

Dans le chapitre 5, les limitations majeures de l'utilisation des modèles statistiques des sources pour la séparation sont présentées et discutées en détails, en s'appuyant sur l'état de l'art et sur les premiers résultats expérimentaux. Ces limitations peuvent être résumées comme l'incapacité en pratique de modéliser assez finement des classes sonores de grande variabilité. Ceci définit la problématique à laquelle nous essayons d'apporter des réponses dans ce travail.

Partie II : Adaptation des modèles : cadre général

Cette partie présente le développement d'un formalisme général d'adaptation *a posteriori* des modèles statistiques aux caractéristiques des sources dans le mélange. Cette adaptation est censée permettre, dans certains cas, de dépasser les limitations des modèles statistiques non adaptés évoquées dans le chapitre 5.

Le chapitre 6 est consacré au développement du formalisme d'adaptation. En s'inspirant fortement des techniques d'adaptation utilisées pour la reconnaissance de la parole, ce formalisme est introduit sous la forme d'un critère d'adaptation bayésienne Maximum *A Posteriori* (MAP).

Dans le chapitre 7, il est expliqué comment appliquer l'algorithme EM pour l'optimisation du critère MAP introduit dans le chapitre précédent. L'algorithme d'adaptation ainsi développé est présenté aux trois différents niveaux de généralité : le plus général, pour des familles exponentielles, et enfin pour les MMG.

Le chapitre 8 présente deux extensions du formalisme d'adaptation. La première extension consiste à utiliser des contraintes paramétriques sur les modèles adaptés plutôt que des lois *a priori*. L'objet de la deuxième extension est l'intégration dans la procédure d'adaptation de diverses informations auxiliaires.

Partie III : Application d'adaptation à la séparation voix / musique

Le formalisme d'adaptation présenté dans la partie II reste assez abstrait pour l'instant. Le but de cette partie est de mettre en pratique ce formalisme pour la tâche de séparation voix / musique.

Le système de séparation voix / musique basé sur des modèles adaptés est présenté dans le chapitre 9. Le module d'adaptation de ce système est composé de trois blocs : la segmentation de la chanson traitée en parties vocales et non vocales, l'adaptation acoustique du modèle de musique sur les parties non vocales et l'adaptation de certains paramètres (des filtres et des gains de DSP) des deux modèles sur toute la chanson.

Dans le chapitre 10, la question de l'intégration de l'adaptation des filtres et des gains lors de l'apprentissage du modèle général est abordée.

Partie IV : Evaluation

Cette partie est consacrée au réglage final de certains paramètres du système de séparation voix / musique, ainsi qu'à l'évaluation de ce système sous deux angles : en utilisant une mesure de performance de séparation (RSDN) et au travers de l'estimation du pitch de la voix chantée.

Dans le chapitre 11, le module de segmentation automatique en parties vocales et non vocales est évalué indépendamment du module d'adaptation des modèles.

Le chapitre 12 présente l'évaluation du système de séparation voix / musique à l'aide de la mesure de performance RSDN. De plus, le seuil de décision du module de segmentation est réglé dans ce chapitre via la performance de séparation.

Enfin, dans le chapitre 13, l'apport de la méthode de séparation voix / musique pour l'estimation du pitch de la voix est mesuré.

Partie V : Conclusion et perspectives

La conclusion générale est présentée dans le chapitre 14.

Quelques perspectives de cette thèse, notamment le développement de techniques rapides d'adaptation (et de séparation) et de méthodes d'adaptation en ligne, sont mentionnées dans le chapitre 15.

Annexes

L'annexe A contient quelques rappels des notions de probabilités et de statistiques utilisées dans cette thèse.

Les démonstrations de certains résultats sont présentées dans l'annexe B.

Liste des articles

Cette thèse a donné lieu à deux articles de revues :

- A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot. Adaptation of Bayesian models for single channel source separation. Application to voice / music separation in popular songs. *IEEE Trans. on Audio, Speech and Lang. Proc.* special issue on Blind Signal Proc. for Speech and Audio Applications (submitted).
- A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot. Choix et adaptation des modèles pour la séparation de voix chantée à partir d'un seul microphone. *Traitement du signal* (accepté pour la publication).

et à deux publications en conférences :

- A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot. One microphone singing voice separation using source-adapted models. In *IEEE Worksh. on Apps. of Signal Processing to Audio and Acoustics (WASPAA'05)*, pages 90 - 93, Mohonk, NY, Oct. 2005.
- A. Ozerov, R. Gribonval, P. Philippe, and F. Bimbot. Séparation voix / musique à partir d'enregistrements mono quelques remarques sur le choix et l'adaptation des modèles. In *GRETSI'05 Symposium on Signal and Image Processing*, Louvain-la-Neuve, Belgique, Sept. 2005.

Démonstrations

Quelques exemples de séparation voix / musique obtenus à l'aide du système développé dans cette thèse se trouvent sur ma page personnelle : `www.irisa.fr/metiss/ozerov/demos.html`

Première partie

Cadre du travail

Chapitre 1

Séparation de sources audio

Ce chapitre est consacré à une introduction à la Séparation de Sources Audio (SSA). Pour avoir plus de détails sur la SSA le lecteur peut se reporter à [Vincent-03] et [Vincent-05]. Cette introduction à pour but de :

- présenter le problème de la SSA en général,
- positionner le problème de séparation de sources avec un seul capteur, qui est traité dans ce travail, par rapport au problème de séparation de sources avec plusieurs capteurs,
- expliquer de quelles connaissances on a besoin pour pouvoir résoudre le problème de séparation de sources avec un seul capteur.

1.1 Introduction au niveau acoustique

Considérons une scène sonore créée par plusieurs émetteurs sonores. Tout objet émettant du son est appelé *émetteur sonore* : cela peut inclure des instruments musicaux, des personnes qui parlent ou chantent, etc. Un son émis par un émetteur est appelé *source*. Supposons que cette scène sonore est enregistrée par un ou plusieurs microphones (Fig. 1.1). Par exemple un microphone est utilisé pour les enregistrements mono et deux microphones sont utilisés pour les enregistrement stéréo. L'enregistrement acquis par un des microphones, qui est constitué des contributions de toutes les sources, est appelé *mélange*.

Etant donnés les mélanges, le problème de la SSA est d'estimer les contributions de chacune des sources dans ces mélanges.

Une source sonore est *ponctuelle* si la taille de l'émetteur correspondant est négligeable par rapport à la longueur d'onde du son émis. Ainsi, cet émetteur peut être représenté par un seul point dans l'espace. Une source sonore ponctuelle est *omnidirectionnelle* si l'émetteur émet avec la même puissance dans toutes les directions de l'espace. En modélisant les scènes sonores, nous supposons toujours que les sources sont ponctuelles et omnidirectionnelles. Comme conséquence,

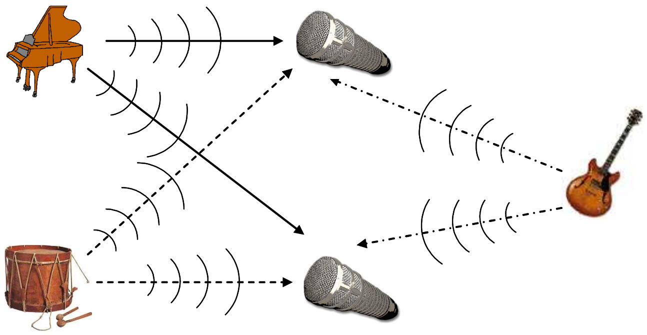


FIG. 1.1 – Scène sonore enregistrée par deux microphones.

les contributions d'une source dans chacun des mélanges sont des transformations d'un même son original. Dans le cas plus général, cela n'est pas vrai pour les sources qui ne sont pas ponctuelles, c'est-à-dire dont les émetteurs sont de taille comparable avec la longueur d'onde, par exemple un piano, une voiture, etc.

En supposant que les sources sont ponctuelles et omnidirectionnelles le problème de la SSA peut être formulé différemment : étant donnés les mélanges, estimer les sources originales (sons originaux) et non plus leurs contributions dans les mélanges. La différence entre une source originale et sa contribution dans un mélange résulte de toutes les transformations subies par cette source pendant le chemin entre son émission par l'émetteur sonore et son enregistrement par le microphone correspondant. Ces transformations sont conditionnées par la position de l'émetteur par rapport au microphone, par les caractéristiques de la pièce, du microphone, etc.

1.2 Formulation pour des signaux numériques

Il est supposé que K *sources* $\{s_k(n)\}_{k=1}^{K}$, où n est l'indice du temps discret, sont mélangées de façon à former L signaux $\{x_l(n)\}_{l=1}^{L}$ appelés *mélanges*. De plus, il est supposé que les mélanges sont des sommes des contributions des sources.

La *contribution* (ou l'*image*) de la k-ème source dans le l-ème mélange est notée $s_{l,k}^{\text{img}}(n)$ et définie comme suit :

$$s_{l,k}^{\text{img}}(n) = \mathcal{A}_{l,k}[s_k](n) \tag{1.1}$$

où $\mathcal{A}_{l,k}$ est le modèle de la transformation de la k-ème source dans le l-ème mélange. L'ensemble de ces modèles $\mathcal{A} = \{\mathcal{A}_{l,k}\}_{l,k}$ est appelé *modèle de mélange*. Quelques modèles de mélange fréquemment utilisés sont décrits dans la section suivante.

Ainsi, chaque mélange x_l étant la somme des contributions des sources s'écrit comme suit :

$$x_l(n) = \sum_{k=1}^{K} s_{l,k}^{\text{img}}(n) \tag{1.2}$$

1.2.1 Modèles de mélange

Un modèle de mélange $\mathcal{A}$ modélise l'effet de toutes les transformations appliquées aux sources avant d'être ajoutées pour créer des mélanges. Dans le cas des enregistrements faits en conditions réelles, c'est-à-dire en utilisant des microphones pour enregistrer directement une scène sonore (voir par exemple Fig. 1.1), ce modèle dépend des positions des émetteurs et des microphones, de l'acoustique de la salle d'enregistrement, des caractéristiques des microphones, etc. Dans le cas des enregistrements créés de manière artificielle, c'est-à-dire quand les sources sont d'abord enregistrées séparément et ensuite mixées, les effets appliqués aux sources définissent le modèle de mélange.

Le modèle de *mélange linéaire instantané* suppose que les mélanges sont des combinaisons linéaires de sources :

$$x_l(n) = \sum_{k=1}^{K} a_{l,k} s_k(n) \tag{1.3}$$

Dans ce cas, les paramètres de mixage sont des gains $a_{l,k}$ représentant les intensités avec lesquelles les sources contribuent à chaque mélange. Ces gains sont souvent réunis dans une matrice $A = [a_{l,k}]_{l,k}$ appelée *matrice de mélange*. Ce modèle est simple mais peu réaliste. En effet, dans des enregistrements réels, les sources subissent l'influence des réverbérations de la salle d'enregistrement, des échos etc. Ainsi, il semble plus pertinent de modéliser ces transformations complexes par des filtres plutôt que par des gains multiplicatifs.

Pour surmonter ces limitations du modèle de mélange linéaire instantané, le modèle de *mélange convolutif* peut être utilisé. Avec ce modèle, les mélanges $x_l(n)$ sont des sommes des sources $s_k(n)$ filtrées par des filtres linéaires, c'est-à-dire :

$$x_l(n) = \sum_{k=1}^{K} \sum_{m=-\infty}^{+\infty} a_{l,k}(m) s_k(n-m) \tag{1.4}$$

où $a_{l,k}(n)$ sont les réponses impulsionnelles des *filtres de mixage*. Ce modèle est plus général que le modèle de mélange linéaire instantané et il correspond mieux aux enregistrements réels.

Le modèle de *mélange anéchoïque* est un modèle intermédiaire entre le mélange linéaire instantané et le mélange convolutif. Ce modèle suppose que les sources sont mélangées avec des gains différents (comme pour le mélange linéaire instantané) et en plus avec des retards temporels différents.

Enfin, pour tous ces modèles, il est parfois supposé que les paramètres de mixage (les gains, les filtres etc.) varient (lentement) au cours du temps. Par exemple, cela modélise le fait que les émetteurs peuvent se déplacer dans l'espace.

1.2.2 Formulation du problème de la SSA

Etant donnés les mélanges $\{x_l(n)\}_{l=1}^{L}$, le problème de la SSA peut être formulé de deux façons différentes [Vincent-05] :

1. Estimer les sources originales $\{s_k(n)\}_{k=1}^{K}$.
2. Estimer les images des sources $\{s_{l,k}^{\text{img}}(n)\}_{l,k=1}^{L,K}$.

Remarquons d'abord que dans le cas de la première formulation, il faut avoir des connaissances très précises sur les sources ou le modèle de mélange pour pouvoir retrouver les sources proprement dites $\{s_k(n)\}_{k=1}^{K}$. Sans de telles connaissances, la plupart des méthodes de séparation de sources permettent d'estimer chaque source à une transformation $\mathcal{A}_{l,k}$ près. Par exemple, dans le cas du mélange linéaire instantané (1.3) ou convolutif (1.4), ces méthodes permettent d'estimer chaque source à un gain multiplicatif ou à un filtre près.

Deuxièmement, il faut noter que pour les deux formulations du problèmes, si il n'y a pas de connaissances spécifiques sur chacune des sources permettant les distinguer les unes des autres, les sources ne peuvent être estimées qu'à une permutation près. Considérons par exemple une personne qui écoute un extrait musical, mais qui ne connaît pas les noms des instruments jouant dans cet extrait. Cette personne ayant des connaissances en musique pourra prêter attention à chacun des instruments et même reproduire sa mélodie ou son rythme, mais elle ne sera pas capable de nommer ces instruments, c'est-à-dire les étiqueter. Les méthodes de séparation capables de n'estimer les sources qu'à une permutation près, c'est-à-dire qui ne sont pas capables de les étiqueter, sont parfois appelées *aveugles* [Vincent-03].

Toutes ces imprécisions sur les estimations de sources doivent être prises en compte dans les critères d'évaluation [Gribonval-03, Vincent-05a].

1.3 Pourquoi sépare-t-on ?

La SSA a de nombreux objectifs. Premièrement, la décomposition des enregistrements en sources originales ouvre la possibilité de créer de nouveaux enregistrements, par exemple en modifiant les positions des sources ou leurs intensités. Deuxièmement, cela peut faciliter l'analyse des enregistrements. Dans le cadre de l'indexation audio par exemple, on cherche à extraire à partir des enregistrements certaines métadonnées telles que des mots, des phrases, des partitions musicales, etc. Souvent, ces métadonnées semblent plus faciles à extraire à partir des sources séparées qu'à partir des enregistrements.

Selon l'article [Vincent-03], deux groupes d'applications de la SSA peuvent être distingués :

1. applications visant à *modifier le contenu audio* (pour la création de nouveaux enregistrements),
2. applications visant à *extraire des informations sémantiques* (pour l'extraction de métadonnées).

Quelques exemples d'applications pour chacun de ces groupes sont présentés ci-dessous :

1. Applications visant à modifier le contenu audio :
   - La restauration d'enregistrements anciens [Cappe-93].
   - Le remixage d'enregistrements [Vincent-04], c'est-à-dire la modification des effets de mixage, des positions des sources dans des enregistrements stéréo, etc.
   - L'élimination de la voix dans des chansons pour des application de karaoké.
2. Applications visant à extraire des informations sémantiques :
   - La reconnaissance automatique de la parole.
   - La reconnaissance / vérification du locuteur.
   - La transcription automatique de musique polyphonique, c'est-à-dire la recherche de la partition musicale jouée pour une source particulière du mélange.

Comme il est déjà remarqué dans l'introduction, l'application traitée dans cette thèse, c'est-à-dire la séparation voix / musique, peut être très utile puisque à partir de la voix bien séparée, il est plus facile d'extraire beaucoup de métadonnées importantes pour caractériser les chansons. Cela peut être, par exemple, la parole chantée, la mélodie chantée, l'identité du chanteur etc. Ces métadonnées peuvent être ensuite utilisées pour de nombreuses tâches d'indexation audio. Les applications suivantes peuvent être ainsi envisagées :

1. Applications visant à extraire des informations sémantiques :
   - La reconnaissance de la parole chantée.
   - La transcription de la mélodie chantée.
   - Estimation de la fréquence fondamentale (pitch) de la voix chantée. La transcription de la mélodie chantée peut être également effectuée à partir d'une estimation du pitch.
   - La reconnaissance de l'identité du chanteur.
2. Application visant à modifier le contenu audio :
   - Le remixage, par exemple l'amplification ou l'atténuation de la voix chantée.

Dans ce travail, nous avons choisi de mesurer l'apport des techniques de séparation voix / musique proposées pour l'estimation du pitch de la voix chantée. Cela permettra d'évaluer les méthodes proposées et la séparation de sources en général dans le cadre d'une tâche d'extraction de métadonnées pour l'indexation audio.

1.4 Classification des problèmes de la SSA par niveau de difficulté

Une classification assez classique des problèmes de la SSA par niveau de difficulté en fonction de leur dimensionnement (K, L) est présentée dans cette section. Cette classification permet de comprendre en quoi la séparation de sources avec un seul capteur est qualitativement différente de la séparation de sources avec plusieurs capteurs et dans un certain sens plus dur.

Commençons d'abord par une explication intuitive. Considérons un enregistrement stéréo $(L = 2)$ avec deux sources $(K = 2)$, par exemple deux instruments musicaux, mixées à l'aide d'un modèle de mélange linéaire instantané (1.3) avec la matrice de mélange $A = \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix}$. Dans ce cas, les rapports entre les gains de mixage pour deux canaux (gauche et droit), c'est-à-dire $r_k = a_{1,k}/a_{2,k}$, $k = 1, 2$, déterminent les directions de provenance des sources. En supposant qu'on arrive à estimer ces rapports r_k, c'est-à-dire les directions (nous ne décrivons pas la méthode d'estimation des r_k ici), l'estimation des sources devient simple et directe. Par exemple, en utilisant le rapport r_1, on peut égaliser l'énergie de la contribution de la source s_1 dans deux canaux. Ensuite, en soustrayant le canal gauche du canal droit, on arrive à éliminer la source s_1 et on obtient une estimation de la source s_2. Cette séparation est possible grâce à la diversité spatiale, c'est-à-dire le fait que les sources arrivent de directions différentes.

Maintenant, considérons un enregistrement stéréo, mais avec plus de deux sources $(K > 2)$, disons trois, et également avec un modèle de mélange linéaire instantané. Dans ce cas, on arrive toujours à estimer les directions (les rapports r_k), mais l'astuce basée sur l'égalisation et la soustraction ne marche plus. En effet, en éliminant une des trois sources dans le mélange, on n'arrive pas vraiment à estimer les deux autres. Cependant, il y a des techniques un peu plus élaborées qui permettent quand même de s'en sortir.

Dans le cas des enregistrements mono, la diversité spatiale n'est plus exploitable. Avec un seul capteur l'information sur les directions de provenance des sources est complètement perdue. Ainsi, il faut avoir d'autres connaissances pour pouvoir séparer les sources.

Passons maintenant à la présentation plus formelle de la classification des problèmes de la SSA par niveau de difficulté en fonction de leur dimensionnement (K, L) dans le cas du modèle de mélange linéaire instantané (1.3). Ce modèle particulier est choisi pour simplifier la présentation.

Cas déterminé $(K = L)$ ou surdéterminé $(K < L)$. Dans ce cas, la connaissance de la matrice de mélange A permet de reconstruire parfaitement les sources en appliquant sa pseudo inverse $A^+ \triangleq A^T(AA^T)^{-1}$ aux mélanges [Jutten-03]. Il suffit donc d'estimer la matrice de mélange. Pour cela, l'Analyse en Composantes Indépendantes (ACI) est souvent utilisée [Cardoso-98] en supposant que les sources sont mutuellement indépendantes.

Cas sous déterminé avec plusieurs capteurs ($K > L$ et $L > 1$). Dans ce cas, l'estimation de la matrice de mélange A seule ne permet plus d'avoir de bonnes estimations des sources [Gribonval-03]. Intuitivement, cela est assez facile à comprendre. En effet, puisque $K > L$, le but est de retrouver plus d'échantillons ($K \times N$, où N est la durée de chaque source) à partir de moins d'échantillons ($L \times N$), sachant seulement que ces échantillons sont reliés par une transformée linéaire A. Une hypothèse supplémentaire concernant *la parcimonie* des sources dans une représentation (une base ou une représentation redondante) permet de contourner la difficulté [Bofill-01, Gribonval-03a]. La parcimonie dans une représentation signifie qu'il y a très peu de coefficients ayant des valeurs significativement grandes.

Cas sous déterminé avec un seul capteur ($K > L$ et $L = 1$). L'estimation de la matrice de mélange A n'apporte aucune information utile pour la séparation dans ce cas. Autrement dit, il n'est plus possible d'utiliser la diversité spatiale des sources, c'est-à-dire de les distinguer grâce aux différentes directions de leurs provenances. L'hypothèse de la parcimonie seule n'est plus suffisante pour séparer les sources et il faut utiliser d'autres connaissances pour y arriver. Souvent, ces connaissances sont représentées sous la forme de modèles *a priori* des sources. Ces modèles décrivent assez finement les caractéristiques des différentes sources à séparer (par ex. la voix, la musique, la parole, etc.). Ainsi, la séparation devient possible grâce à la diversité des caractéristiques particulières des sources. Ce sont ces méthodes basées sur des modèles *a priori* des sources que nous allons étudier dans cette thèse.

1.5 Conclusion

Dans ce chapitre, le problème de la SSA à été formulé de manière assez générale d'abord au niveau acoustique et ensuite pour des signaux numériques. Désormais, nous ne considérerons que la formulation pour des signaux numériques.

Plusieurs applications potentielles de la SSA sont présentées et classées en deux groupes (applications visant à modifier le contenu audio et applications visant à extraire des informations sémantiques). Dans cette thèse, nous allons évaluer l'apport des techniques de séparation voix / musique proposées pour une application visant à extraire des informations sémantiques, notamment pour l'estimation du pitch de la voix chantée. Cette application est très utile pour certaines tâches d'indexation audio.

Une classification assez classique des tâches de la SSA par trois niveaux de difficulté (cas (sur) déterminé, sous déterminé avec plusieurs capteurs et sous déterminé avec un seul capteur) est présentée. Cette classification montre que la séparation de sources avec un seul capteur est plus difficile que la séparation de sources avec plusieurs capteurs, car la diversité spatiale n'est pas exploitable avec un seul capteur et il faut avoir d'autres connaissances pour pouvoir séparer. Une grande famille de méthodes utilise des modèles *a priori* des sources comme telles connaissances. Ces méthodes seront présentées dans le chapitre suivant.

Chapitre 2

Séparation de sources avec un seul capteur

Puisque dans le cas d'un seul capteur, la connaissance du modèle de mélange n'apporte pas d'information utilisable pour la séparation (voir Sec. 1.4), on peut chercher à estimer les contributions des sources $s_{1,k}^{\mathrm{img}}(n)$ (1.2) au lieu de chercher à estimer les sources elles mêmes $s_k(n)$. A la place de l'équation (1.2) nous considérons ainsi l'équation du mélange :

$$x(n) = \sum_k s_k(n) \tag{2.1}$$

et posons le problème comme suit : étant donné le mélange monophonique x, trouver les estimations des sources $\{\hat{s}_k\}_{k=1}^K$.

Pour simplifier les notations, nous avons remplacé dans l'équation (2.1) les contributions de sources $s_{1,k}^{\mathrm{img}}(n)$ (voir (1.2)) par les sources mêmes $s_k(n)$. On voit bien que le modèle initial du mélange $\mathcal{A}$ (linéaire instantané (1.3), convolutif (1.4) ou autre) n'apparait pas dans une telle formulation du problème. Ainsi, les méthodes qu'on pourrait utiliser pour résoudre ce problème ne reposent pas sur la structure de $\mathcal{A}$.

Puisque de toute manière, dans l'application traitée dans ce travail, c'est-à-dire la séparation voix / musique, il n'y a que deux sources (la voix et la musique) nous simplifions l'équation (2.1) pour deux sources. Une généralisation à plus de deux sources peut être faite, si cela est nécessaire. Par la suite, nous allons donc toujours considérer l'équation du mélange suivante :

$$x(n) = s_1(n) + s_2(n) \tag{2.2}$$

et le problème est de trouver les estimations des sources $\hat{s}_1$ et $\hat{s}_2$ étant donné le mélange x.

2.1 Présentation intuitive

Nous commençons par une présentation assez informelle et vulgarisée d'une approche permettant de séparer les sources à partir d'un seul microphone. Cette présentation permet de comprendre d'une part la difficulté de la tâche et d'autre part une manière de la résoudre. L'introduction plus formelle et théorique de cette approche sera faite par la suite (Sec. 2.4).

2.1.1 Hypothèse de travail : faible recouvrement dans le domaine de Fourier

La séparation est généralement effectuée dans un domaine temps - fréquence plutôt que dans le domaine temporel, en utilisant par exemple la Transformée de Fourier à Court Terme (TFCT). Puisque la TFCT est une transformée linéaire, l'équation de mélange (2.2) est préservée, c'est-à-dire :

$$X(t,f) = S_1(t,f) + S_2(t,f) \tag{2.3}$$

où $X(t,f)$, $S_1(t,f)$ et $S_2(t,f)$ sont des TFCT des signaux temporels $x(n)$, $s_1(n)$ et $s_2(n)$ pour la trame numéro $t = 1, 2, \ldots, T$ et d'indice de fréquence $f = 1, 2, \ldots, F$ (F est l'indice de la fréquence de Nyquist). Par la suite, les signaux temporels sont toujours notés par des lettres minuscules et leurs TFCT par les lettres majuscules correspondantes.

Le choix du domaine de la TFCT pour la séparation est motivé par le fait que les sources audio se recouvrent très peu dans ce domaine. Cette propriété a été montrée par exemple pour les signaux de parole [Rickard-02]. Ainsi, il parait plus facile d'effectuer la séparation dans le domaine de la TFCT plutôt que dans le domaine temporel. Pour donner un exemple, des signaux de voix chantée, de violon et de leur mélange sont représentés sur la figure 2.1 (A) et leurs spectrogrammes (modules de la TFCT en échelle logarithmique) sont représentés sur la figure 2.1 (B). On voit que dans le domaine de la TFCT, les sources (la voix et le violon) sont faciles à distinguer dans le mélange et on peut les séparer en supprimant par exemple les harmoniques de la source qu'on veut éliminer.

Dans la littérature, on trouve cette hypothèse du non-recouvrement dans le domaine de la TFCT sous le nom de WDO (*W-Disjoint Orthogonality*) [Rickard-02]. Cette hypothèse suppose que les supports des sources sont disjoints dans le domaine de la TFCT, c'est-à-dire

$$S_1(t,f)S_2(t,f) = 0 \tag{2.4}$$

En réalité cette hypothèse n'est jamais vérifiée exactement, car d'une part chaque source contient souvent une part de bruit qui apparaît partout dans le plan temps - fréquence et d'autre part les harmoniques de sources différentes peuvent se croiser. Cependant, dans beaucoup de cas, les régions les plus énergétiques des sources audio se recouvrent très peu dans le domaine de la TFCT. Par exemple, sur la figure 2.1 (C) nous avons représenté les régions les plus énergétiques (gardant 99 % de l'énergie totale des signaux) des spectrogrammes de la voix et du violon ainsi que leur intersection dans le mélange. On voit en effet que ces régions se recouvrent très peu. Cette dernière hypothèse du faible recouvrement dans le domaine de la TFCT est appelée *WDO approchée* [Rickard-02]. Nous ne chercherons pas à donner ici une définition mathématique rigoureuse de cette hypothèse et nous contenterons de l'explication intuitive donnée.

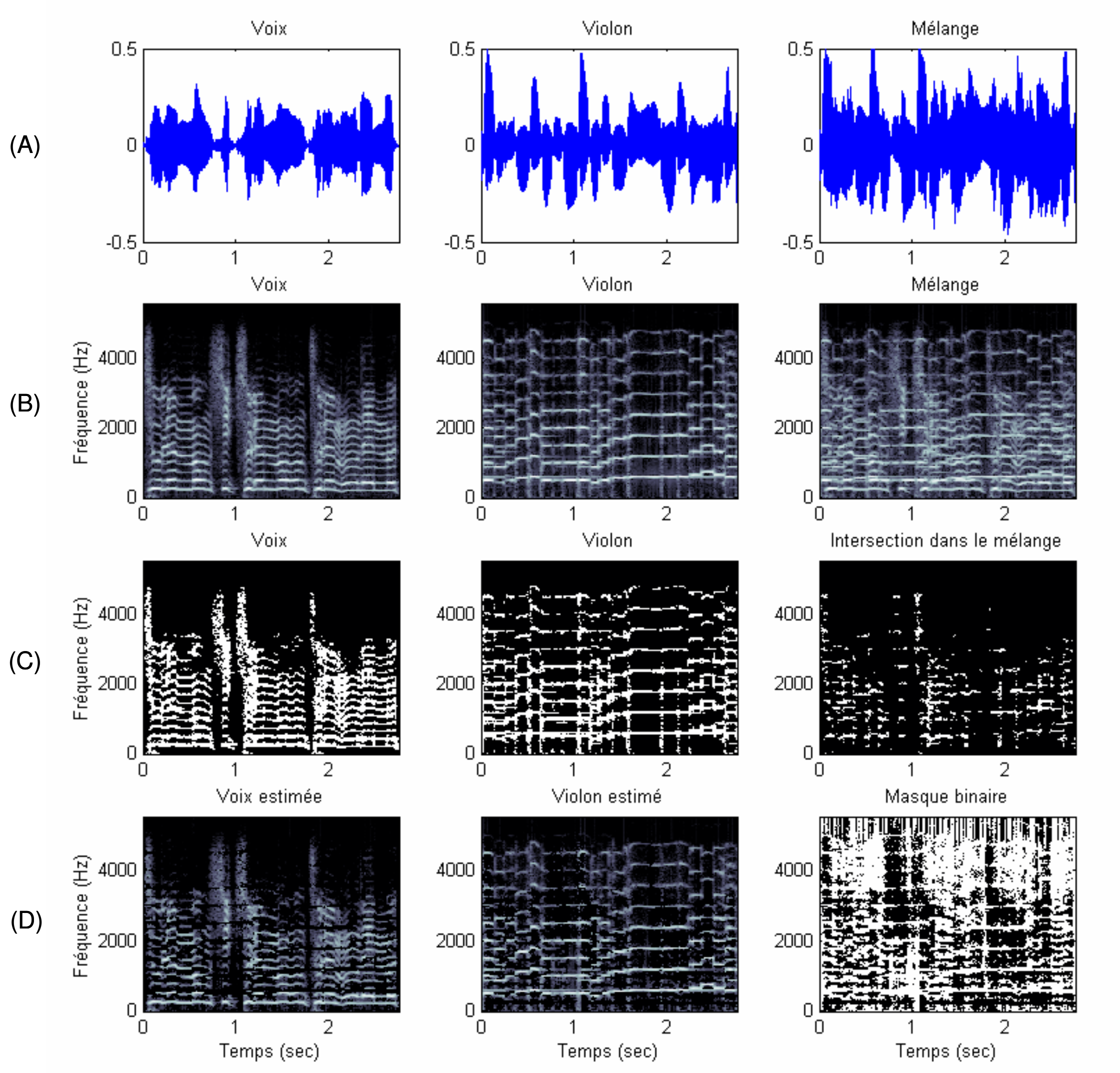


FIG. 2.1 – **(A)** : Voix chantée, violon et leur mélange dans le domaine temporel. **(B)** : Spectrogrammes de ces signaux. **(C)** : Régions les plus énergétiques des spectrogrammes de la voix et du violon (en blanc) et l'intersection de ces régions dans le mélange (en blanc). **(D)** : Estimations des spectrogrammes obtenues par l'application du masque binaire (représenté à droite) et de son complémentaire au spectrogramme du mélange.

2.1.2 Masquage temps - fréquence

Dans le domaine de la TFCT, le problème de séparation reste le même : étant donné la TFCT **complexe** du mélange $X(t,f)$ et l'équation du mélange (2.3), trouver des estimations des TFCT **complexes** des sources $\widehat{S}_k(t,f)$, $k = 1, 2$.

Pour y arriver l'opération suivante appelée *masquage temps - fréquence* [1] est utilisée fréquemment [Benaroya-03] :

$$\widehat{S}_k(t,f) = \mathcal{M}_k(t,f)X(t,f) \tag{2.5}$$

où $\mathcal{M}_k(t,f)$ est un gain **réel** entre 0 et 1 appliqué à la TFCT complexe du mélange $X(t,f)$. L'ensemble des gains $\mathcal{M}_k = [\mathcal{M}_k(t,f)]_{t,f}$ s'appelle *masque*.

Cette opération correspond également à un filtrage adaptatif avec le filtre $\mathcal{M}_k(t) = [\mathcal{M}_k(t,f)]_f$ variant au cours du temps.

Il faut aussi remarquer que la phase de la TFCT d'une source n'est pas réestimée dans (2.5), c'est-à-dire que la phase de la TFCT du mélange $X(t,f)$ est gardée dans l'estimation de la TFCT de cette source $\widehat{S}_k(t,f)$. En effet, dans l'équation (2.5) la TFCT complexe du mélange $X(t,f)$ est multipliée par un gain réel $\mathcal{M}_k(t,f)$. La phase n'est donc pas modifiée.

2.1.3 Masquage oracle

La question cruciale est ainsi de construire des masques menant à de bonnes estimations des sources.

On peut définir des *masques oracles*, c'est-à-dire des masques construits en utilisant la connaissance des TFCT des sources $S_k(t,f)$, $k = 1, 2$. Bien évidemment, il n'est possible de construire des masques oracles que dans des conditions expérimentales, car en réalité les sources S_k ne sont pas accessibles [2].

Le *masque oracle binaire* est calculé comme suit (de même pour $\mathcal{M}_2$) :

$$\mathcal{M}_1^{\text{orac\_bin}}(t,f) = \begin{cases} 1 & \text{si} \quad |S_1(t,f)| > |S_2(t,f)|, \\ 0 & \text{sinon} \end{cases} \tag{2.6}$$

L'hypothèse de WDO (2.4) étant vérifiée, le masquage avec ce masque mène à une estimation exacte (c'est-à-dire $\widehat{S}_k(t,f) = S_k(t,f)$).

[1] Le terme "masquage temps - fréquence" est largement utilisé dans la communauté de recherche sur la séparation de sources. Cependant, ce terme n'a rien à avoir avec les notions du "masquage temporel" et du "masquage fréquentiel" concernant le masquage des sons pendant leur perception par l'oreille.

[2] Puisque par la suite le traitement est souvent effectué dans le domaine de la TFCT, nous omettons parfois "la TFCT" dans des phrases où il est clair, d'après le contexte, qu'il s'agit de la TFCT. Par exemple, nous disons "la source S_k" au lieu de "la TFCT de la source S_k" ou "le mélange X" au lieu de "la TFCT du mélange X".

Comme les sources se recouvrent partiellement, c'est-à-dire que l'hypothèse de WDO n'est vérifiée qu'approximativement, on donne plus de liberté à chaque gain réel $\mathcal{M}_k(t,f)$, en supposant $\mathcal{M}_k(t,f) \in [0,1]$ et en utilisant par exemple le *masque oracle de Wiener* (de même pour $\mathcal{M}_2$) :

$$\mathcal{M}_1^{\text{orac\_wien}}(t,f) = \frac{|S_1(t,f)|^2}{|S_1(t,f)|^2 + |S_2(t,f)|^2} \tag{2.7}$$

La règle de construction de ce masque oracle est inspirée par le filtrage de Wiener [Wiener-49] $\mathcal{M}_1^{\text{wien}}(t,f) = \frac{r_1^2(f)}{r_1^2(f)+r_2^2(f)}$ en remplaçant les variances *a priori* $r_k^2(f)$, $k = 1,2$ par des modules au carré des TFCT des sources $|S_k(t,f)|^2$, $k = 1,2$.

Sur la figure 2.1 (D), les estimations des TFCT de la voix et du violon obtenues en appliquant le masque binaire (2.6) et son complémentaire à la TFCT du mélange sont représentées. On voit que grâce au faible recouvrement des sources dans le domaine temps - fréquence, les spectrogrammes sont convenablement reconstruits par rapport aux spectrogrammes originaux (Fig. 2.1 (B)). De plus, en écoutant les signaux temporels reconstruits à partir des estimations des TFCT (Fig. 2.1 (D)) et en les comparant avec les sources originales (Fig. 2.1 (A)), on s'aperçoit que les résultats de séparation sont très satisfaisants.

Ainsi, il est possible d'obtenir de bons résultats de séparation des sources audio en utilisant des masques oracles (par exemple binaires (2.6) ou ceux de Wiener (2.7)). Ceci montre en particulier la validité d'hypothèse du faible recouvrement dans le domaine de la TFCT. Le problème est qu'en pratique, les sources mélangées S_k ne sont pas connues (c'est justement elles que l'on cherche à estimer). Par conséquent, les masques oracles ne sont pas accessibles. Il faut donc trouver un autre moyen pour construire de bons masques. Ainsi, on peut dire que notre problème de séparation se restreint à l'estimation d'un masque.

2.1.4 Exemple de construction d'un masque

Comme remarqué dans la section précédente, il est nécessaire de trouver un moyen de construire des masques temps - fréquence sans utiliser les sources S_k (comme pour les masques oracles). Il est clair qu'il est impossible de trouver de bons masques en ne se basant que sur la connaissance du mélange X. En effet, si seul le mélange est disponible, on ne sait vraiment pas ce qu'il faut séparer. Il faut ainsi avoir d'autres connaissances sur les sources.

De telles connaissances sont souvent représentées sous la forme de modèles *a priori* de sources (dont on donne un exemple ci-après). Ces modèles sont appris sur des données d'entraînement y_k, $k = 1,2$ dont les caractéristiques sont similaires à celles des sources mélangées s_k.

Pour donner un exemple de modèles *a priori* de sources, nous considérons ici des *ensembles de Densités Spectrales de Puissance (DSP)*. Chaque source S_k est caractérisée par un ensemble de Q_k formes spectrales typiques (ou DSP) noté $\{r^2_{k,i}\}_{i=1}^{Q_k}$. Chaque forme spectrale typique $r^2_{k,i} = [r^2_{k,i}(f)]_f$ est un vecteur fréquentiel dont les composantes $r^2_{k,i}(f)$ sont des variances *a priori* de la source S_k au sein de cette forme spectrale.

L'apprentissage peut être par exemple effectué de la manière suivante. Soit Y_1 et Y_2, les TFCT des données d'entraînement y_1 et y_2. Premièrement, les modules au carré des spectres des données d'entraînement [3] [4] $|Y_k(t)|^2$ sont groupés en Q_k groupes en utilisant par exemple l'algorithme des K-moyennes (*K-means*) [McQueen-67]. Deuxièmement, les DSP $r^2_{k,i}$ sont calculées comme les moyennes des groupes obtenus.

Pour calculer un masque $\mathcal{M}(t)$ pour la trame numéro t, l'idéal serait de connaître les spectres de puissance (modules des spectres au carré) des sources $|S_1(t)|^2$ et $|S_2(t)|^2$ et d'utiliser par exemple la formule du masque oracle de Wiener (2.7). Cependant, les spectres de puissance des sources $|S_1(t)|^2$ et $|S_2(t)|^2$ ne sont pas accessibles, et nous ne connaissons que le mélange X et les modèles (ensembles de DSP). Ainsi, l'idée est de remplacer dans la formule (2.7) les spectres de puissance des sources par un couple de DSP ($r^2_{1,i^*(t)}$ et $r^2_{2,j^*(t)}$) qui leur ressemblent le plus. N'étant pas capable de comparer directement les DSP aux spectres de puissance des sources $|S_1(t)|^2$ et $|S_2(t)|^2$, nous allons chercher un couple de DSP dont la somme $r^2_{1,i^*(t)} + r^2_{2,j^*(t)}$ ressemble le plus au spectre de puissance du mélange $|X(t)|^2$. Cette ressemblance est calculée à l'aide d'une mesure de ressemblance $\Theta(\cdot,\cdot)$ qui peut être par exemple l'inverse de la distance euclidienne.

2.1.5 Exemple d'algorithme de séparation

L'algorithme 1 résume l'algorithme de séparation présenté dans les sections précédentes (2.1.1 à 2.1.4). Cet algorithme est également schématisé sur la figure 2.2.

L'algorithme qu'on vient de présenter sera développé par la suite de façon théorique et nous verrons que les ensembles de DSP considérés ici peuvent être représentés par des Modèles de Mélange de Gaussiennes (MMG). Mais avant cela, nous présentons un état de l'art plus complet des techniques de séparation de sources avec un seul capteur, basées sur des modèles *a priori*.

2.2 Méthodes basées sur des modèles *a priori* : état de l'art

Comme déjà mentionné dans l'introduction de cette thèse, nous étudions des méthodes de séparation basées sur des modèles *a priori* des sources. Cette section est consacrée à la présentation d'un état de l'art de ces méthodes.

[3] Rappel : Quand la TFCT d'un signal est écrite avec l'indice temporel t seul (par ex. $Y_k(t)$), cela représente le spectre à court terme, c'est-à-dire $Y_k(t) = [Y_k(t,f)]_f$.

[4] Dans cette thèse, les opérations $\log(\cdot)$, $\exp(\cdot)$, $|\cdot|$ et $(\cdot)^2$ appliquées aux vecteurs et aux matrices s'effectuent élément par élément.

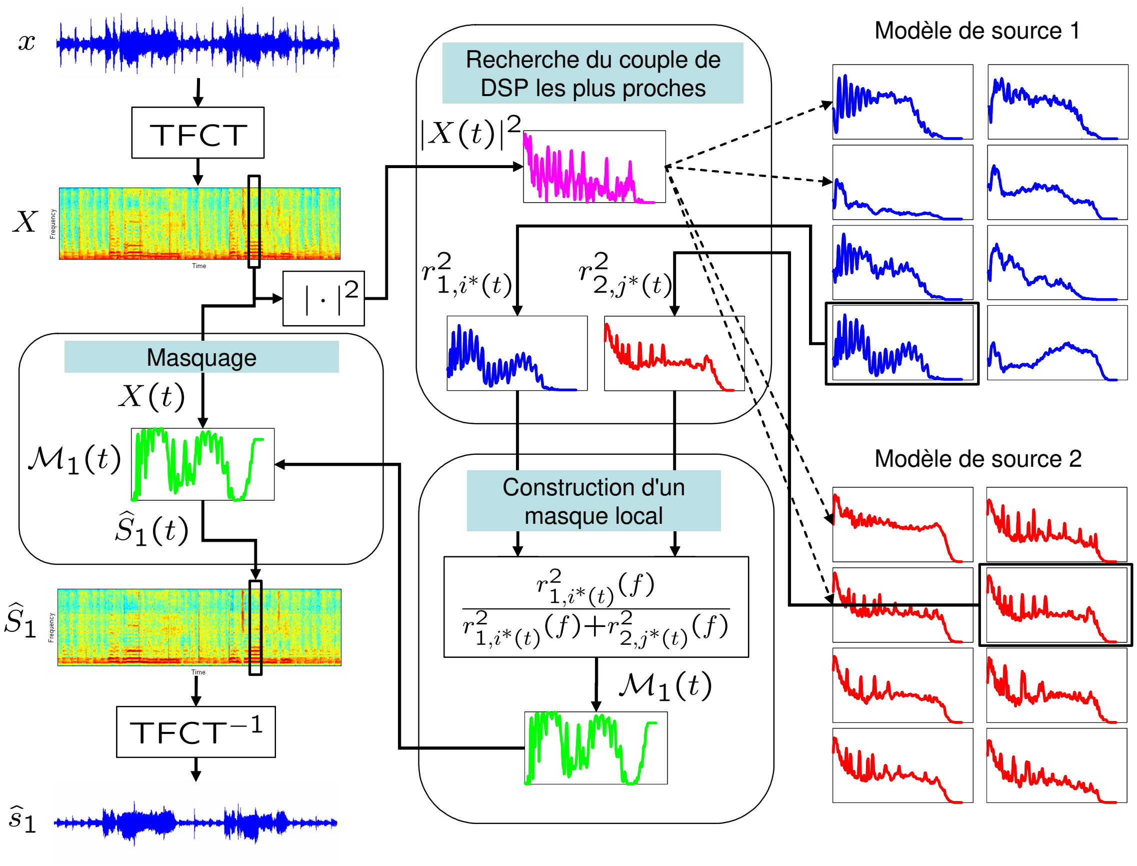


FIG. 2.2 – Schéma d'un algorithme de séparation utilisant des ensembles de DSP comme connaissances *a priori* sur les sources.

Algorithme 1 Séparation de sources avec un seul capteur.

1. Calculer la TFCT du mélange X à partir du signal temporel x.
2. Pour chaque $t = 1, 2, \ldots, T$:
   (a) Trouver le couple de DSP les plus proches du spectre du mélange $|X(t)|^2$, c'est-à-dire

$$(i^*(t), j^*(t)) = \arg\max_{(i,j)} \Theta\left(|X(t)|^2, r_{1,i}^2 + r_{2,j}^2\right), \tag{2.8}$$

   où $\Theta(\cdot,\cdot)$ est une mesure de ressemblance.

   (b) Construire un masque temps - fréquence local :

$$\mathcal{M}_1(t,f) = \frac{r_{1,i^*(t)}^2(f)}{r_{1,i^*(t)}^2(f) + r_{2,j^*(t)}^2(f)} \tag{2.9}$$

   (c) Appliquer ce masque local au spectre du mélange $X(t)$ pour obtenir l'estimation du spectre de la source :

$$\widehat{S}_1(t,f) = \mathcal{M}_1(t,f)X(t,f) \tag{2.10}$$

3. Reconstruire l'estimation de la source dans le domaine temporel $\hat{s}_1$ à partir de l'estimation de la TFCT $\widehat{S}_1$ en utilisant la méthode *OverLap and Add* (OLA) (voir par exemple [Peeters-99]).

Au préalable, il faut remarquer qu'il existe d'autres approches pour la séparation de sources avec un seul capteur, qui ne sont pas basées sur des modèles *a priori* des sources, c'est-à-dire quand il y a un modèle *a priori* pour chaque source particulière (par ex. la voix, un instrument musical particulier, etc.), mais sur une modélisation globale de toutes les sources à la fois. Par exemple, on peut mentionner l'Analyse Computationnelle de Scènes Auditives (*Computational Auditory Scene Analysis*) (CASA) [Cooke-93, Brown-94, Ellis-96, Hu-03], l'Analyse en Sous-espaces Indépendants (ASI) (basée sur l'ACI) [Casey-00, Vincent-01], méthodes basées sur la Factorisation en Matrices Non Négatives (*Non negative Matrix Factorisation*, NMF) [Smaragdis-04, Wang-05, Helen-05, Vembu-05, Kim-06, Schmidt-06], etc. Nous ne présentons pas toutes ces approches, car elles sortent du cadre de l'étude menée dans cette thèse.

Le schéma représenté figure 2.3 résume le principe des méthodes basées sur des modèles *a priori* des sources. Les modèles *a priori* Λ_1 et Λ_2 des deux sources sont les seules connaissances utilisées pour les séparer. Pour chaque source, un modèle est appris à partir d'une base d'entraînement. Les sources sont ensuite estimées à partir du mélange et des modèles. L'algorithme présenté dans la section 2.1.5 appartient à cette famille de méthodes.

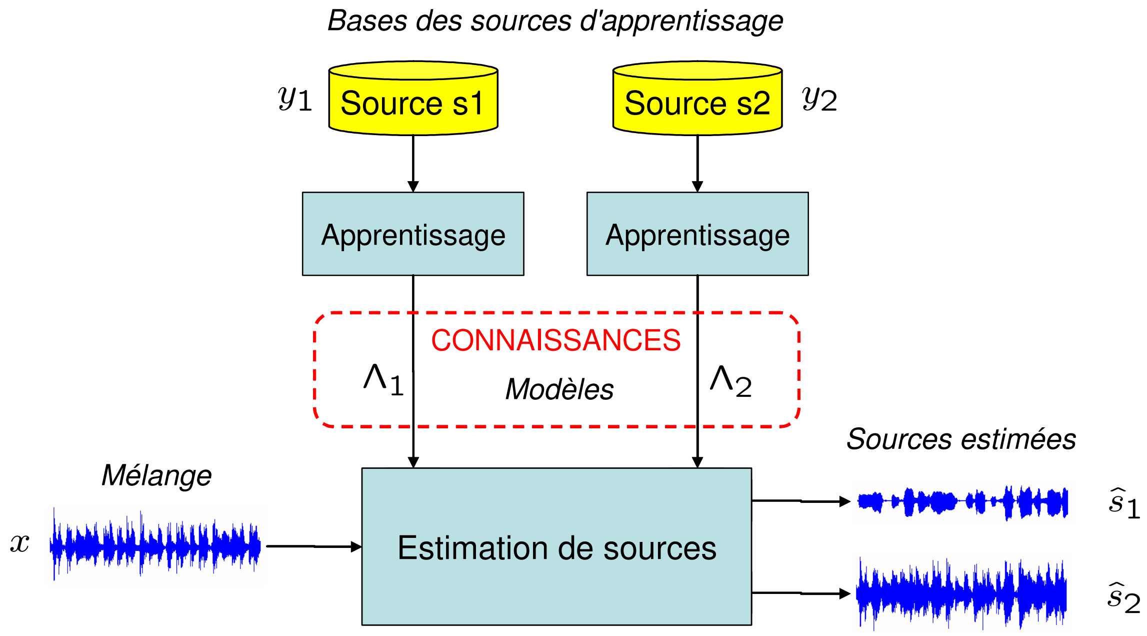


FIG. 2.3 – Méthodes basées sur des modèles *a priori* des sources.

2.2.1 Réseaux bayésiens (modèles graphiques orientés)

Dans la plupart des cas, les modèles *a priori* utilisés sont des modèles probabilistes. Une des originalités de ce travail consiste à présenter ces modèles, ainsi que des algorithmes (par exemple l'apprentissage des modèles, l'estimations de sources (Fig. 2.3)) sous la forme de *réseaux bayésiens*.

Les *réseaux bayésiens* (ou les *modèles graphiques orientés*) [Jordan-98, Murphy-02] peuvent être définis comme un moyen de représenter des modèles probabilistes sous forme de graphes (Fig. 2.4). Les noeuds de ces graphes correspondent aux variables aléatoires et les flèches entre les noeuds décrivent les dépendances conditionnelles entre ces variables. Notons que cette représentation ne précise ni la nature des variables aléatoires ni les lois les reliant.

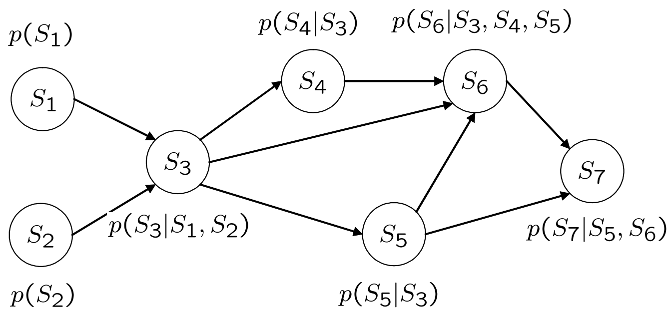


FIG. 2.4 – Exemple d'un réseau bayésien.

Il peut être intéressant d'utiliser des réseaux bayésiens pour les raisons suivantes :

- Visualisation des dépendances entre les variables aléatoires d'un modèle. Cela peut être très utile pour la compréhension et l'interprétation plus rapide des modèles probabilistes.
- Description d'un modèle. Le fait d'avoir dessiné un graphe évite d'avoir à prononcer toutes les hypothèses d'indépendance conditionnelle entre les variables aléatoires.
- Groupement des modèles probabilistes décrits par un même réseau bayésien. Ceci peut mener au développement d'algorithmes génériques d'apprentissage et d'inférence pour des modèles appartenant à un même groupe.

Donnons une description plus détaillée des *réseaux bayésiens* (*modèles graphiques orientés*) [Jordan-98, Murphy-02]. Chaque variable aléatoire S_i d'un modèle probabiliste est représentée par le noeud d'un graphe (Fig. 2.4). Ici nous considérons des modèles graphiques *orientés*, c'est-à-dire que les graphes associés aux modèles probabilistes sont *orientés* (les noeuds sont reliés par des flèches) et *acycliques* (il n'y a pas de chemin qui commence et finit dans le même noeud) [Jordan-98, Murphy-02]. A chaque noeud S_i, une densité de probabilité conditionnelle $p(S_i|\{S_j\}_{j\in\pi(i)})$ est associée, où $\pi(i)$ est l'ensemble des indices des *noeuds parents*, c'est-à-dire les noeuds d'où partent les flèches qui rentrent dans S_i. En particulier, cela signifie que, conditionnellement aux variables aléatoires $\{S_j\}_{j\in\pi(i)}$, la variable aléatoire S_i est indépendante de toutes les autres variables $\{S_j\}_{j\neq i, j\notin\pi(i)}$ et la densité conjointe de toutes les variables aléatoires du modèle est le produit de ces densités conditionnelles :

$$p(S_1, S_2, \ldots, S_I) = \prod_{i=1}^{I} p(S_i|\{S_j\}_{j\in\pi(i)}) \tag{2.11}$$

Parfois on fait aussi la distinction entre les noeuds du graphe représentant différents types de variables aléatoires (v. a.). Les noeuds ronds correspondent aux v. a. continues et les noeuds carrés aux v. a. discrètes. Les noeuds noirs sont observés, c'est-à-dire que la v. a. correspondante est remplacée par une de ses réalisations qui a été observée, et les noeuds blancs sont cachés (Fig. 2.5).

La figure 2.5 représente les réseaux bayésiens pour certains modèles probabilistes des sources. Le Modèle de Mélange de Gaussiennes (MMG) ayant pour observations les spectres à court terme de la source s_k peut être décrit de manière suivante : à chaque instant t un état $q_k(t)$ est émis selon une loi de probabilité discrète. Conditionnellement à l'état $q_k(t)$, le spectre à court terme $S_k(t)$ est distribué selon une loi normale (gaussienne) dont les paramètres dépendent de l'état $q_k(t)$. Pour le Modèle de Markov Caché (MMC) d'une source, la seule différence par rapport au MMG est que la loi discrète d'émission d'un état à l'instant t dépend de l'état émis à l'instant $t-1$. Les modèles de mélanges (2.3) correspondants, notamment le MMG factoriel et le MMC factoriel, sont également représentés graphiquement sur la figure 2.5.

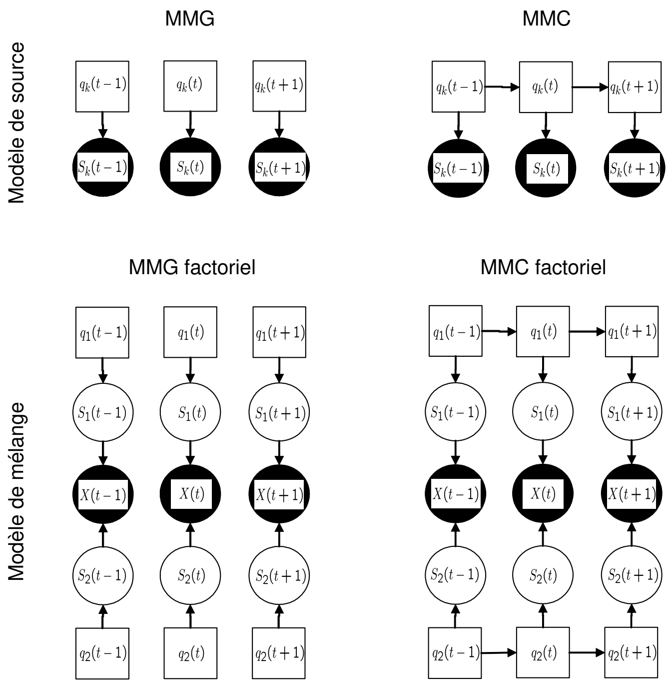


FIG. 2.5 – Réseaux bayésiens de modèles de sources et des modèles de mélanges correspondants. Formes des noeuds : variables aléatoires continues (ronds), variables aléatoires discrètes (carrés). Coloration des noeuds : noeuds observés (noir), noeuds cachés (blanc).

2.2.2 Méthodes basées sur les MMG / MMC

Pour la séparation de sources avec un seul capteur, on attribue en général à Roweis [Roweis-01] la première proposition d'une méthode basée sur des modèles *a priori* de sources. Dans cette proposition, les sources sont modélisées par des MMC ayant pour observations les logarithmes des spectres à court terme. Ensuite, de nombreuses variantes de cette approche avec les MMG ou les MMC ont été proposées [Benaroya-03, Pontoppidan-03, Vincent-04, Kristjansson-04, Beierholm-04, Reddy-04]. Ces approches ont été évaluées pour la tâche de séparation de parole homme / femme [Roweis-01, Pontoppidan-03, Kristjansson-04, Beierholm-04, Reddy-04], mais aussi pour la séparation des signaux musicaux [Benaroya-03, Vincent-04].

2.2.2.1 Quelques remarques sur les MMG

Nous ne rentrons pas ici dans les détails techniques des méthodes basées sur les MMG, cela sera fait par la suite. Notons seulement que ces méthodes possèdent de fortes parentés avec l'algorithme 1 (Sec. 2.1.5).

Ainsi, remarquons que la complexité calculatoire de la séparation utilisant les MMG est de l'ordre $O(Q_1 Q_2 T F)$, où Q_k est la taille du k-ème modèle, c'est-à-dire le nombre de DSP (Alg. 1), et T et F sont respectivement le nombre de trames et l'indice de la fréquence de Nyquist de la TFCT du mélange X. En effet, pour chaque trame, il faut faire une recherche exhaustive d'un couple de DSP parmi Q_1Q_2 couples possibles (Alg. 1). Roweis [Roweis-01] et Pontoppidan [Pontoppidan-03] proposent certaines astuces permettant d'éviter cette recherche exhaustive. Cependant, ces astuces sont développées dans les cadres particuliers des méthodes proposées et elles ne sont pas directement généralisables pour toutes autres méthodes basées sur des MMG.

2.2.2.2 Quelques remarques sur les MMC

Les MMC sont des extensions assez naturelles des MMG. En comparant leurs réseaux bayésiens (Fig. 2.5), on remarque que la seule différence est que, dans le cas des MMC, il y a des dépendances entre les états discrets cachés. Par exemple, si ces états représentent les notes d'une oeuvre musicale, les lois de transition entre les états modélisent les durées des notes et les probabilités de transition entre les notes, qui peuvent par exemple être estimées à partir de la clé musicale de cette oeuvre (voir [Ryynanen-05]). Pour les MMG, il n'y a pas du tout de dépendances temporelles, même pas entre les états cachés, ainsi toutes les observations sont traitées indépendamment.

Pour l'utilisation des MMC, la seule différence algorithmique par rapport aux MMG est que l'étape de recherche des couples de DSP (Alg. 1) doit être remplacée par l'application de l'algorithme de Viterbi (voir par ex. [Benaroya-03]) sur toute la séquence des trames.

Par la suite nous présenterons toutes les méthodes particulières dans le cas des MMG, sachant que l'extension aux MMC peut être faite facilement quand cela est nécessaire.

2.2.3 Méthodes similaires pour le débruitage de la parole avec un seul capteur

Le débruitage de la parole avec un seul capteur est un problème plus ancien que celui de la séparation de sources. Il est traité depuis plusieurs décennies [Curtis-78, Boll-79, Berouti-79]. Cependant, ce problème peut être vu comme un cas particulier de séparation des sources. Ainsi, certaines approches pour le débruitage de la parole sont basées aussi sur des MMG / MMC pour modéliser la parole et le bruit. Les premiers travaux sur ces approches sont généralement attribués à Ephraim [Ephraim-92, Ephraim-92a], mais d'autres propositions ont été faites depuis [Moreno-96, Burshtein-99]. Les approches pour la séparation que nous traitons ici ont donc beaucoup en commun avec ces approches pour le débruitage.

Il existe cependant des différences entre ces deux groupes d'approches. Notamment, pour le débruitage de la parole, les modèles utilisés sont plus simples que ceux utilisés pour la séparation de sources. Deux différences peuvent être soulignées :

1. Pour le débruitage de la parole, le bruit est parfois modélisé comme un signal stationnaire, ce qui n'est pas acceptable pour la plupart des sources audio, telles que la musique, la

parole etc. Dans ce cas, le modèle du bruit est très simple, c'est un MMG à un état, ou plus simplement une seule DSP [Ephraim-92a].

2. Pour le débruitage de la parole, le bruit et la parole sont souvent représentés localement par des modèles AutoRegressifs (AR) d'ordre faible (entre 10 et 12) [Ephraim-92a] modélisant des enveloppes spectrales. Cela permet d'atténuer de manière lisse le bruit dans des zones fréquentielles où il domine la parole (Fig. 2.6 (A)). Dans des zones où le bruit est dominé par la parole il ne s'entend pas dans la plupart des cas, car il est souvent masqué [5] par des harmoniques de la parole. En général, une telle modélisation grossière n'est plus suffisante pour la séparation de sources. En effet, par exemple pour la séparation de parole femme / homme, l'application d'un filtre de Wiener lisse ne permet pas de supprimer les harmoniques de la parole masculine dans l'estimation de la parole féminine (voir Fig. 2.6 (B)). Ces harmoniques vont s'entendre dans l'estimation, car ils sont mal masqués. Ainsi, pour bien séparer, il faut avoir un filtre de Wiener plus haché, qui est construit en utilisant la différence entre les fréquences fondamentales (pitchs) des sources. Les DSP locales utilisées pour la séparation ouvrent une telle possibilité (Fig. 2.6 (C)).

2.2.4 Extensions des méthodes basées sur les MMG et les MMC

Remarquons que la modélisation par les MMG n'est invariante ni par rapport aux intensités globales des sources, ni par rapport à leurs intensités locales. En effet, selon l'algorithme 1 et la figure 2.2, si l'intensité du mélange x change, le choix du couple de DSP peut également changer, car en général la mesure de ressemblance Θ n'est invariante ni à l'intensité du mélange X, ni aux intensités des DSP. Pour remédier à cela, Benaroya [Benaroya-03] propose d'associer un *facteur de gain* à chaque DSP. Ces facteurs de gains sont réestimés pour chaque trame, en adaptant ainsi les intensités des DSP aux intensités locales des sources. C'est une idée similaire à celle utilisée par Ephraim [Ephraim-92a] pour le débruitage de la parole.

Dans le cadre d'un système de séparation de sources et de transcription d'enregistrements musicaux, Vincent [Vincent-04] utilise aussi des modèles proches des MMC. Un modèle représente un instrument monophonique harmonique et chaque DSP d'un tel modèle représente une note. En plus de l'intensité (facteur de gain), Vincent [Vincent-04] propose d'associer d'autres paramètres descriptifs à chaque DSP. Ces paramètres représentent le volume, la hauteur et le timbre de la note correspondante.

[5] Ici il s'agit bien du *masquage fréquentiel* des sons pendant leur perception par l'oreille, et non pas du *masquage temps - fréquence* qui est un traitement (Sec. 2.1.2).

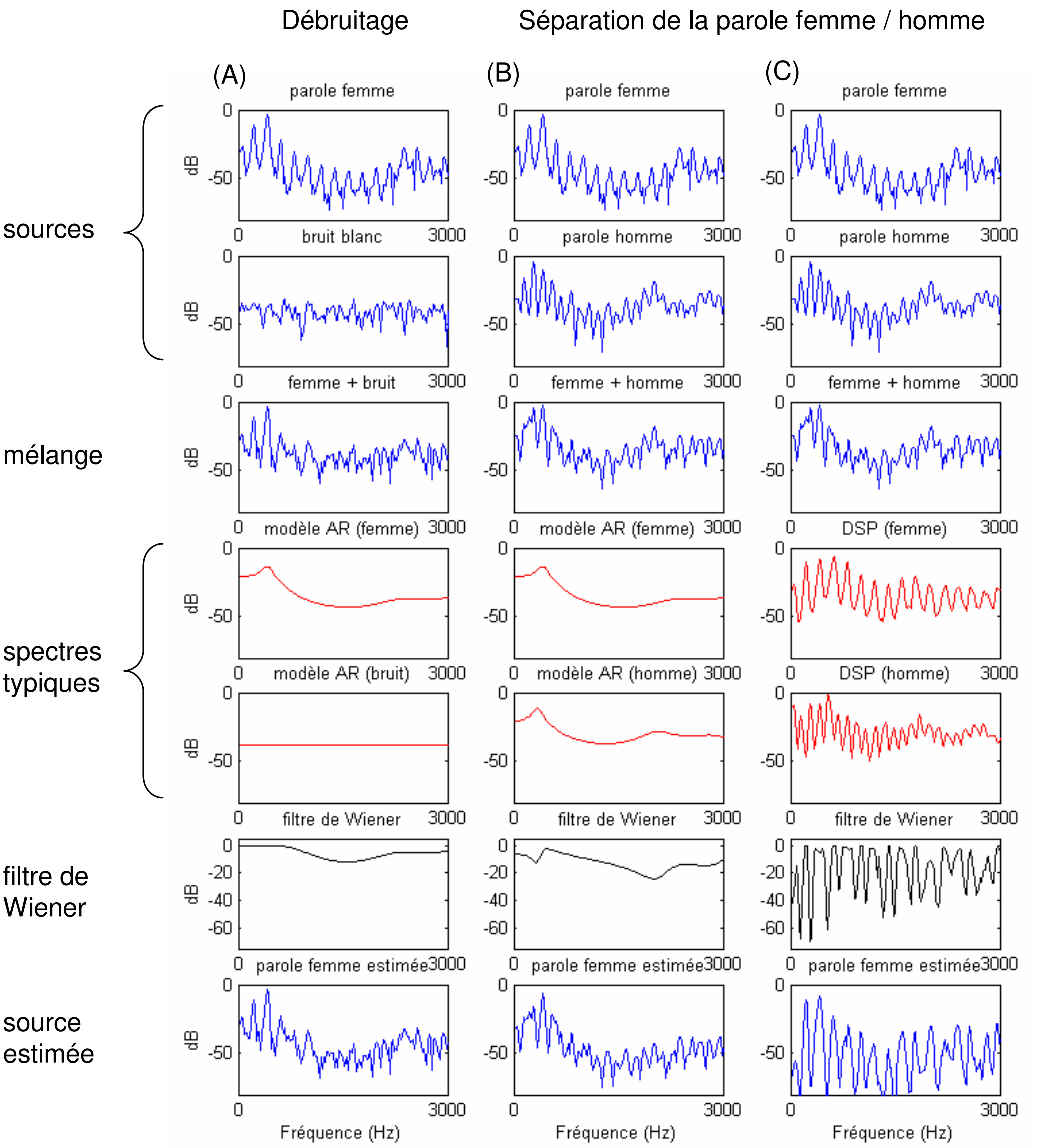


FIG. 2.6 – Illustration sur un spectre du débruitage de la parole et de la séparation de parole femme / homme en utilisant deux modèles : grossier (AR d'ordre 10) et fin (DSP). Les scénarios suivants sont illustrés. **(A)** : Débruitage de la parole féminine avec les modèles AR. **(B)** : Séparation de parole femme / homme avec les modèles AR. **(C)** : Idem avec les DSP (modèles fins des spectres). Pour chaque scénario qui se déroule selon la figure 2.2 (Alg. 1), les objets suivant sont représentés : les spectres des sources et du mélange ($S_1(t)$, $S_2(t)$ et $X(t)$), les spectres typiques ($r^2_{1,i^*(t)}$ et $r^2_{2,j^*(t)}$), le filtre de Wiener (masque temps - fréquence $\mathcal{M}(t)$) et l'estimation du spectre de la première source ($\widehat{S}_1(t)$).

2.2.5 Autres modèles

Mentionnons brièvement encore quelques approches. Pour la séparation des signaux de parole, Hershey et Casey [Hershey-01] utilisent des MMC factoriels, en modélisant indépendamment les harmoniques et les formants de la parole. Benaroya [Benaroya-03] propose d'utiliser la NMF (*Non negative Matrix Factorisation*) parcimonieuse [Hoyer-02] pour obtenir des modèles *a priori*. Il appelle ces modèles *dictionnaires de DSP*. Jang et Lee [Jang-03] utilisent comme modèles *a priori* les ensembles de fonctions de base dans le domaine temporel appris en utilisant l'ACI. Reyes-Gomez *et al.* [Reyes-Gomez-04b] proposent de découper la TFCT d'une source en quelques bandes fréquentielles, de modéliser dans un premier temps chaque bande indépendamment par un MMC et d'introduire dans un deuxième temps des dépendances entre les états des MMC dans des bandes voisines. Pour la séparation des signaux de parole, Ellis et Weiss [Ellis-06] utilisent comme modèles *a priori* des dictionnaires de formes spectrales caractéristiques appris à l'aide d'un algorithme de Quantification Vectorielle (QV).

2.2.6 Modèles utilisés dans cette thèse

Dans cette thèse, nous utilisons les MMG. Comme il est remarqué section 2.2.2.2, l'extension aux MMC peut être faite facilement en remplaçant la recherche des couples de DSP de l'algorithme 1 par l'algorithme de Viterbi. Ceci est valable pour les techniques existantes, ainsi que pour les nouvelles méthodes que nous allons proposer.

Notons quelques avantages de ces modèles :

1. Les MMG / MMC et leurs extensions sont utilisés dans la majorité des méthodes basées sur des modèles *a priori* des sources. Une telle popularité est un indice de la validité de ces modèles.
2. De plus, les méthodes basées sur des MMG sont assez générales, c'est-à-dire qu'elles ne sont pas spécifiques à certaines classes des signaux audio.
3. Les MMG ont été déjà appliqués pour la séparation des signaux de parole [Roweis-01, Kristjansson-04], ainsi que pour la séparation des signaux musicaux [Benaroya-03, Vincent-04], et des résultats plutôt satisfaisants ont été rapportés. Par conséquent, l'utilisation des MMG semble être une direction prometteuse pour notre tâche de séparation voix / musique.

Cependant, malgré tous les avantages des MMG, il y a des limitations majeures auxquelles nous serons confrontés dans ce travail. Notamment, puisque cette modélisation est très fine, il devient difficile de construire des modèles représentatifs pour des classes sonores de grande variabilité. D'une part, ceci est lié à la difficulté de la construction de bases d'entraînement représentatives pour des classes sonores très riches. D'autre part, étant limité en pratique par des ressource calculatoires, on ne peut pas traiter des modèles de très grand taille (composés par exemple de milliers de spectres typiques).

Puisque dans le cadre de la séparation voix / musique, la classe sonore de musique est d'une variabilité exorbitante, il semble indispensable dans ce travail d'apporter des solutions aux problèmes annoncés. Nous y reviendrons par la suite, où ces problèmes seront présentés et expliqués de manière plus détaillée.

Le reste de ce chapitre est composé d'une présentation technique générale des approches basées sur des modèles probabilistes *a priori*, approfondie ensuite pour des MMG.

2.3 Méthodes basées sur des modèles probabilistes *a priori* : présentation technique générale

Comme nous l'avons vu dans la section 2.2 il existe de nombreuses méthodes de séparation de sources avec un seul capteur basées sur des modèles statistiques *a priori*, notamment sur des MMG. Une des contributions de notre travail consiste en une tentative de réunir toutes ces méthodes sous la forme d'un schéma générique, représenté figure 2.7. Avant de passer à l'explication de chaque bloc de ce schéma, nous introduisons brièvement les trois transformées suivantes associées à chaque méthode de séparation :

- La transformée $\mathcal{F}$ transforme le signal temporel dans un domaine particulier où tout le traitement est ensuite effectué. Souvent, $\mathcal{F} = \text{TFCT}$ est utilisée [Roweis-01, Benaroya-03, Pontoppidan-03, Kristjansson-04, Beierholm-04, Reddy-04]. D'autres représentations temps - fréquence sont parfois utilisées [Vincent-04]. Enfin, certaines méthodes restent dans le domaine temporel ($\mathcal{F} = \text{Id}$) [Jang-03].
- La transformée $\mathcal{L}$ définit (dans le domaine transformé par $\mathcal{F}$) le domaine de modélisation des sources (par exemple, $\mathcal{L} = \text{Id}$ dans [Benaroya-03, Ephraim-85, Beierholm-04] et $\mathcal{L} = \log|\cdot|$ dans [Moreno-96, Burshtein-99, Roweis-01, Vincent-04, Kristjansson-04], voir section 2.4).
- La transformée $\mathcal{D}$ définit (dans le domaine transformé par $\mathcal{F}$) le domaine dans lequel sera calculée l'Erreur Quadratique Moyenne (EQM) minimisée pour l'estimation des sources (par exemple, $\mathcal{D} = \text{Id}$ dans [Benaroya-03, Beierholm-04, Reddy-04] et $\mathcal{D} = \log|\cdot|$ dans [Ephraim-85, Moreno-96, Burshtein-99, Roweis-01, Kristjansson-04], voir section 2.4).

2.3.1 Domaine du traitement

Généralement, le traitement est effectué dans un domaine autre que le domaine temporel. Nous supposons que les signaux subissent une transformée $\mathcal{F}$ qui peut être redondante en général.

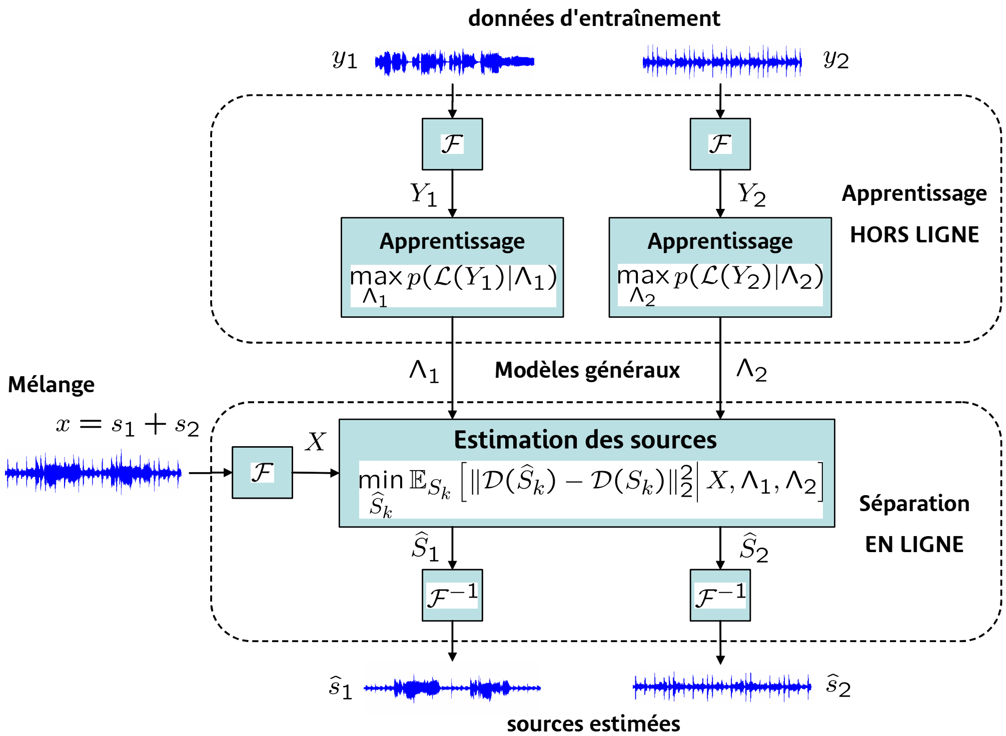


FIG. 2.7 – Schéma général de la séparation de sources basée sur des modèles probabilistes *a priori*.

Après le traitement, les signaux traités sont reconstruits dans le domaine temporel à l'aide d'une transformée de reconstruction $\mathcal{F}^{-1}$.

Si la transformée $\mathcal{F}$ est redondante (comme l'est la TFCT par exemple), la transformée de reconstruction $\mathcal{F}^{-1}$ n'est pas unique. Ainsi, il faut choisir une transformée parmi plusieurs transformées de reconstruction possibles. Cela peut être fait à l'aide d'un critère, par exemple celui des moindres carrés.

Comme dans le cas particulier de la TFCT, les signaux temporels sont notés par des lettres minuscules (par ex. s_k) et les signaux dans le domaine transformé sont notés par les lettres majuscules correspondantes (par ex. $S_k = \mathcal{F}(s_k)$).

2.3.2 Apprentissage de modèles

Dans la plupart des approches [Roweis-01, Benaroya-03, Pontoppidan-03, Vincent-04, Kristjansson-04] l'apprentissage des modèles *a priori* Λ_1 et Λ_2 des deux sources se fait indépendamment à partir des données d'entraînement y_1 et y_2 en utilisant le critère du Maximum de Vraisemblance (MV) :

$$\Lambda_k = \arg\max_{\Lambda'_k} p(\mathcal{L}(Y_k)|\Lambda'_k), \qquad k = 1, 2, \tag{2.12}$$

où $Y_k = \mathcal{F}(y_k)$ sont des données d'entraînement transformées, $\mathcal{L}(Y_k)$ est le processus aléatoire modélisé par le modèle Λ_k et $\mathcal{L}$ est une transformée.

2.3.3 Estimation de sources

Pour estimer les sources, on cherche à minimiser la mesure de distorsion $d(\widehat{S}_k, S_k)$ entre la source estimée et la vraie source. Puisque la vraie source S_k n'est pas observée pendant la séparation, la valeur de la mesure de distorsion est remplacée par son espérance conditionnelle calculée par rapport à S_k, sachant le mélange X et les modèles Λ_1 et Λ_2. C'est cette espérance conditionnelle qui est ensuite minimisée. Les sources sont donc estimées comme suit :

$$\widehat{S}_k = \arg\min_{S'_k} \mathbb{E}_{S_k}\left[d(S'_k, S_k)|X, \Lambda_1, \Lambda_2\right] \tag{2.13}$$

Supposons que $d(\widehat{S}_k, S_k) = \|\mathcal{D}(\widehat{S}_k) - \mathcal{D}(S_k)\|_2^2$ est l'Erreur Quadratique Moyenne (EQM) de $\mathcal{D}(S_k)$, où $\mathcal{D}$ est une transformée inversible. En utilisant l'expression pour l'estimateur minimisant l'EQM [Kay-93] nous obtenons :

$$\widehat{S}_k = \mathcal{D}^{-1}\left(\mathbb{E}_{S_k}\left[\mathcal{D}(S_k)|X, \Lambda_1, \Lambda_2\right]\right) \tag{2.14}$$

2.4 Méthodes de séparation basées sur des Modèles de Mélange de Gaussiennes (MMG)

Dans cette section, nous présentons trois méthodes de séparation de sources avec un seul capteur basées sur des MMG [Benaroya-03, Ephraim-85, Burshtein-99]. Les deux dernières méthodes [Ephraim-85, Burshtein-99] ont été utilisées à l'origine pour le débruitage de la parole.

Nous avons choisi de présenter seulement ces trois méthodes, car toutes les autres méthodes basées sur des MMG [Ephraim-92, Ephraim-92a, Moreno-96, Roweis-01, Kristjansson-04, Beierholm-04, Reddy-04] ressemblent en principe à ces trois-là.

Le principe des méthodes, étant déjà présenté de manière intuitive dans la section 2.1.5 (voir Alg. 1 et Fig. 2.2), il s'agit ici d'une présentation théorique suivant le schéma général de la figure 2.7.

Pour toutes les méthodes une TFCT avec un recouvrement des fenêtres d'analyse à 50 % est utilisée pour la représentation des signaux (ce qui définit $\mathcal{F}$). Les signaux sont reconstruits dans le domaine temporel en utilisant la méthode OLA [Peeters-99] (ce qui définit $\mathcal{F}^{-1}$).

Selon le schéma de la figure 2.7, il reste à spécifier les transformées $\mathcal{L}$ et $\mathcal{D}$ définissant respectivement le *domaine de modélisation* et le *domaine de minimisation de l'EQM*. Nous allons considérer les possibilités suivantes :

1. Domaine de modélisation : $\mathcal{L} = \mathrm{Id}$ ou $\mathcal{L} = \log|\cdot|$, la méthode correspondante est appelée *MMG spectral* ou *MMG log spectral*.
2. Domaine de minimisation de l'EQM : $\mathcal{D} = \mathrm{Id}$ ou $\mathcal{D} = \log|\cdot|$, la méthode correspondante est appelée *EQM spectrale* ou *EQM log spectrale*.

Chaque méthode est ainsi définie par un duo MMG / EQM et les trois méthodes que nous allons décrire sont :

1. **MMG spectral / EQM spectrale** (Sec. 2.4.1) [Benaroya-03],
2. **MMG spectral / EQM log spectrale** (Sec. 2.4.1) [Ephraim-85],
3. **MMG log spectral / EQM log spectrale** (Sec. 2.4.2) [Burshtein-99].

Pour ces trois méthodes, le bloc "séparation" du schéma représenté figure 2.7 est décrit par la figure 2.8. A partir de la TFCT du mélange $X(t, f)$ et des paramètres des modèles Λ_1 et Λ_2, des masques temps - fréquence $\mathcal{M}_k$, $k = 1, 2$ (Sec. 2.1.2), c'est-à-dire des ensembles des gains réels $\mathcal{M}_k(t, f) \geq 0$, sont calculés pour deux sources. Ensuite, les estimations de la TFCT des sources $\widehat{S}_k(t, f)$ sont obtenues en multipliant $X(t, f)$ par ces gains (2.5). Chaque masque $\mathcal{M}_k$ est calculé de telle façon que l'estimation $S'_k(t, f) = \mathcal{M}_k(t, f)X(t, f)$ minimise une mesure de distorsion donnée (2.13). Les sources estimées $\hat{s}_k(n)$ sont enfin reconstruites dans le domaine temporel.

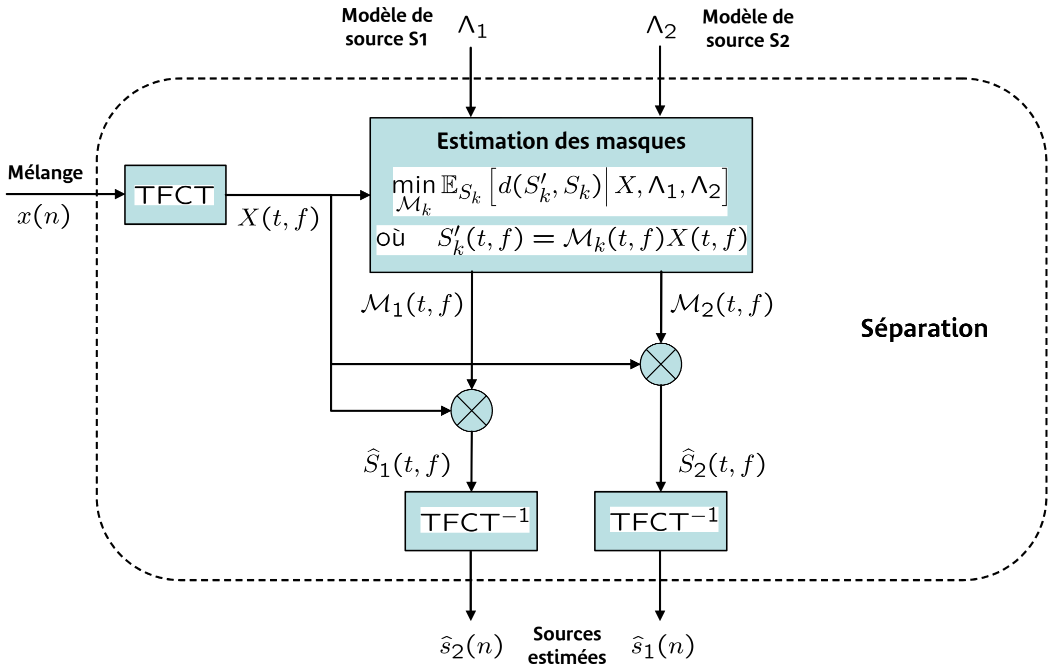


FIG. 2.8 – Séparation avec des méthodes basées sur des MMG.

2.4.1 Modélisation des spectres par des MMG

Benaroya [Benaroya-03] propose de modéliser les spectres à court terme par des MMG. On appellera ces modèles *MMG spectraux* et on les notera Λ_k^{spec}, $k = 1, 2$. Les spectres à court terme des deux sources $S_k(t)$ sont modélisés comme des vecteurs aléatoires complexes circulaires de densité MMG, avec des vecteurs moyens nuls et des matrices de covariance diagonales $R_{k,i} = \text{diag}[r_{k,i}^2(f)]_f$, c'est-à-dire :

$$p(S_k(t)|\Lambda_k^{\text{spec}}) = \sum_i u_{k,i} N_C(S_k(t); \bar{0}, R_{k,i}), \qquad k = 1, 2, \tag{2.15}$$

où $u_{k,i} \geq 0$ sont les poids des gaussiennes satisfaisants $\sum_i u_{k,i} = 1$. La densité de probabilité d'un vecteur aléatoire gaussien complexe circulaire $N_C(\cdot)$ est définie dans l'annexe A.1 (Eq. (A.2)).

Les MMG spectraux sont paramétrisés comme suit : $\Lambda_k^{\text{spec}} = \{u_{k,i}, R_{k,i}\}_i$, $k = 1, 2$.

La diagonale de chaque matrice de covariance $[r_{k,i}^2(f)]_f$ représente une Densité Spectrale de Puissance (DSP) locale. Ainsi, chaque modèle explique la source modélisée par un nombre fini de formes spectrales caractéristiques, ou bien un ensemble de DSP (Sec. 2.1.4). Pour donner un exemple, un MMG spectral à 16 états est représenté sur la figure 2.9.

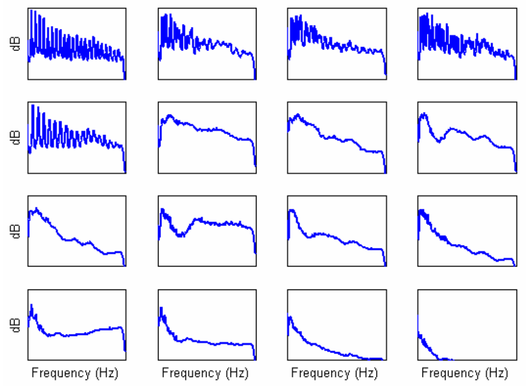


FIG. 2.9 – MMG spectral à 16 états. Chaque état i est représenté par sa DSP : $\log r_i^2(f)$.

2.4.1.1 Apprentissage des MMG spectraux

L'apprentissage des modèles est basé sur le critère du MV (2.12) avec $\mathcal{L} = \mathrm{Id}$. En pratique, l'apprentissage utilise l'algorithme EM (*Expectation-Maximisation*) [Dempster-77]. Pour les MMG spectraux, les formules de réestimation des paramètres [Benaroya-03a] sont résumées par l'algorithme 2. L'algorithme des K-moyennes [McQueen-67] est utilisé pour l'initialisation de EM.

Algorithme 2 Algorithme EM pour l'apprentissage d'un MMG spectral $\Lambda_k^{\text{spec}} = \{u_{k,i}, R_{k,i}\}_i$ à partir des données d'entraînement Y_k (les paramètres estimés à la l-ème itération de EM sont notés par (l) en exposant).

1. Calculer les poids $\gamma_i^{(l)}(t)$ satisfaisant $\sum_i \gamma_i^{(l)}(t) = 1$ et

$$\gamma_i^{(l)}(t) \propto u_{k,i}^{(l)} N_C \left(Y_k(t); \bar{0}, R_{k,i}^{(l)} \right), \tag{2.16}$$

   où le symbole $\propto$ signifie la proportionnalité et $N_C(\cdot)$ est défini selon (A.2).

2. Mettre à jour les poids de gaussiennes $u_{k,i}$:

$$u_{k,i}^{(l+1)} = \frac{1}{T} \sum_t \gamma_i^{(l)}(t) \tag{2.17}$$

3. Mettre à jour les matrices de covariances $R_{k,i}$:

$$r_{k,i}^{2,(l+1)}(f) = \frac{\sum_t \gamma_i^{(l)}(t) |Y_k(t,f)|^2}{\sum_t \gamma_i^{(l)}(t)} \tag{2.18}$$

2.4.1.2 Estimateur minimisant l'EQM spectrale

Considérons la mesure de distorsion suivante :

$$d_{\text{spec}}(\widehat{S}_k, S_k) = \|\widehat{S}_k - S_k\|_2^2 = \sum_{t,f} |\widehat{S}_k(t,f) - S_k(t,f)|^2, \tag{2.19}$$

Si l'on souhaite minimiser cette mesure de distorsion, appelée ensuite *l'EQM spectrale*, en utilisant l'expression (2.14), on arrive à la formule suivante pour le masque $\mathcal{M}_1$ [Benaroya-03] :

$$\mathcal{M}_1^{\text{wien\_ada}}(t,f) = \sum_{i,j} \gamma_{i,j}(t) \frac{r_{1,i}^2(f)}{r_{1,i}^2(f) + r_{2,j}^2(f)}, \tag{2.20}$$

où $\gamma_{i,j}(t)$ est la probabilité de choisir la paire d'états (i,j) pour l'observation $X(t)$, satisfaisant $\sum_{i,j} \gamma_{i,j}(t) = 1$ et

$$\gamma_{i,j}(t) \triangleq P(q_1(t) = i, q_2(t) = j | X(t), \Lambda_1, \Lambda_2) \propto u_{1,i} u_{2,j} N_C(X(t); \bar{0}, R_{1,i} + R_{2,j}), \tag{2.21}$$

où $N_C(\cdot)$ est défini selon (A.2).

Ce masque satisfait $\mathcal{M}_1^{\text{wien\_ada}}(t,f) \in [0,1]$ et l'estimation des sources revient à effectuer un *filtrage de Wiener pondéré* ou bien *adaptatif*.

Notons que quand chaque modèle Λ_k est composé d'un seul état, c'est-à-dire d'une seule DSP $r_k^2 = [r_k^2(f)]_f$, on revient au filtrage de Wiener "simple" ("non" adaptatif) [Wiener-49] :

$$\mathcal{M}_1^{\text{wien}}(t,f) = \frac{r_1^2(f)}{r_1^2(f) + r_2^2(f)} \tag{2.22}$$

2.4.1.3 Estimateur dur vs. estimateur doux

En faisant des expériences sur la séparation voix / musique, nous avons remarqué que parmi les probabilités $\gamma_{i,j}(t)$ pour un t donné, il y a souvent une probabilité $\gamma_{i^*(t),j^*(t)}(t)$ qui domine beaucoup les autres, c'est-à-dire que cette probabilité vaut presque 1 et par conséquent les autres valent presque 0, car leur somme vaut 1. Ainsi, il est possible de faire une approximation pour le calcul du masque $\mathcal{M}_1^{\text{wien\_ada}}$ en remplaçant la somme des filtres de Wiener dans l'équation (2.20) par un seul filtre de Wiener correspondant au couple d'états le plus probable :

$$\mathcal{M}_1^{\text{wien\_ada(dur)}}(t,f) = \frac{r_{1,i^*(t)}^2(f)}{r_{1,i^*(t)}^2(f) + r_{2,j^*(t)}^2(f)}, \tag{2.23}$$

avec

$$(i^*(t), j^*(t)) = \arg\max_{(i,j)} \gamma_{i,j}(t) \tag{2.24}$$

Maintenant, on voit que c'est cette approximation qui est utilisée dans l'algorithme 1 schématisé sur la figure 2.2. Les équations (2.24) et (2.23) correspondent aux équations (2.8) et (2.9) avec la mesure de ressemblance Θ définie par l'expression (2.21).

Nous appelons les estimateurs du type (2.23) *estimateurs durs* par opposition aux estimateurs du type (2.20) que nous appelons *estimateurs doux*.

L'utilité potentielle d'un estimateur dur est que son utilisation pourra permettre d'accélérer significativement l'algorithme de séparation, si l'on trouve une astuce permettant d'éviter la recherche exhaustive d'un couple d'états (2.24) comme le font Roweis [Roweis-01] et Pontoppidan [Pontoppidan-03] (voir aussi Sec. 2.2.2.1). Remarquons également qu'on peut envisager d'utiliser une solution intermédiaire entre l'estimateur dur et l'estimateur doux en calculant le masque comme une somme pondérée de quelques filtres de Wiener correspondant aux quelques couples d'états les plus probables.

Nous verrons dans le chapitre 4 consacré aux expérimentations préliminaires que pour la séparation voix / musique, l'utilisation des estimateurs durs ne change pas significativement les performances de séparation par rapport aux estimateurs doux.

Par la suite, tous les estimateurs sont introduits sous la forme douce, mais il est sous-entendu qu'ils peuvent être également utilisés sous la forme dure.

2.4.1.4 Estimateur minimisant l'EQM log spectrale

Considérons maintenant une autre mesure de distorsion, qu'on va appeler *l'EQM log spectrale* :

$$d_{\log}(\widehat{S}_k, S_k) = \sum_{t,f} \left| \log|\widehat{S}_k(t,f)| - \log|S_k(t,f)| \right|^2 \tag{2.25}$$

En utilisant (2.14) avec $\mathcal{D}(S_k) = \log|S_k|$, il est possible d'obtenir le masque pour l'EQM log spectrale [Ephraim-85] :

$$\mathcal{M}_1^{\text{spec\_log}}(t,f) = \sum_{i,j} \gamma_{i,j}(t) \left[\log \frac{r_{1,i}^2(f)}{r_{1,i}^2(f) + r_{2,j}^2(f)} + \frac{E_1(\theta_{i,j})}{2} \right], \tag{2.26}$$

où $\theta_{i,j} = \frac{r_{1,i}^2(f)|X(t,f)|^2}{[r_{1,i}^2(f)+r_{2,j}^2(f)]r_{2,j}^2(f)}$ et $E_1(\theta) = \int_\theta^\infty \frac{e^{-t}}{t} dt$ est connue sous le nom de *l'intégrale exponentielle* et peut être calculé numériquement de façon efficace (voir [Press-92], pages 222 – 226). Les probabilités $\gamma_{i,j}(t)$ sont calculées en utilisant (2.21).

2.4.1.5 Une remarque sur la phase

Il faut remarquer que dans l'expression de l'EQM log spectrale (2.25), la phase de la TFCT $\angle S_k(t,f) = \arg S_k(t,f)$ n'est pas prise en compte. C'est-à-dire que, contrairement aux exigences de la section 2.3.3, la transformée $\mathcal{D}(S_k) = \log|S_k|$ n'est pas inversible.

Formellement, il faut ajouter l'EQM de la phase

$$d_{\mathrm{phase}}(\widehat{S}_k, S_k) = \sum_{t,f} \left| e^{j\angle \widehat{S}_k(t,f)} - e^{j\angle S_k(t,f)} \right|^2 \tag{2.27}$$

à l'EQM log spectrale (2.25). En utilisant la définition de MMG spectral (2.15) et la définition d'un vecteur gaussien complexe [Kay-93] (voir Eq. (A.2) et Fig. A.1), on peut en déduire que la phase est distribuée uniformément entre 0 et 2π :

$$\angle S_k(t,f) \sim \mathcal{U}(0, 2\pi) \tag{2.28}$$

C'est-à-dire qu'il n'y a aucune connaissance *a priori* sur la phase. Dans ce cas, Ephraim [Ephraim-92] montre que l'estimation minimisant l'EQM de la phase (2.27) est la phase du mélange $\angle X(t,f)$. Cela reste vrai pour toutes les méthodes considérées ici. Sur la figure 2.8, cela transparaît par le fait que la TFCT complexe du mélange $X(t,f)$ est multipliée par un gain réel $\mathcal{M}_k(t,f)$.

Cet estimateur de la phase rejoint l'intuition. En effet, si l'on n'a pas de connaissances *a priori* sur la phase, il vaut mieux ne rien modifier et laisser la phase de mélange.

2.4.2 Modélisation des log spectres par des MMG

Burshtein et Gannot [Burshtein-99] proposent de modéliser les logarithmes des spectres par des MMG. On appellera ces modèles *MMG log spectraux* et on les notera $\Lambda_k^{\log}$, $k = 1, 2$. Les logarithmes des spectres des deux sources $\mathbf{S}_k(t) \triangleq \log |S_k(t)|$ sont modélisés par des MMG avec des vecteurs moyens $\mu_{k,i}$ et des matrices de covariance diagonales $R_{k,i}$:

$$p\left(\mathbf{S}_k(t) \middle| \Lambda_k^{\log}\right) = \sum_i u_{k,i} N(\mathbf{S}_k(t); \mu_{k,i}, R_{k,i}), \qquad k = 1, 2, \tag{2.29}$$

où la densité d'un vecteur gaussien réel $N(\cdot)$ est définie selon l'équation (A.1).

De plus, il est supposé que, comme pour les MMG spectraux (2.15), la phase de la TFCT est distribuée uniformément entre 0 et 2π. Ces MMG sont paramétrisés comme suit : $\Lambda_k^{\log} = \{u_{k,i}, \mu_{k,i}, R_{k,i}\}_i$, $k = 1, 2$.

Avec cette modélisation, les DSP locales sont représentées par les vecteurs moyens $\mu_{k,i}$ plutôt que par les diagonales des matrices de covariance $[r_{k,i}^2(f)]_f$, comme dans le cas des MMG spectraux. En effet, selon l'équation (2.29) les moyennes $[\mu_{k,i}(f)]_f$ déterminent la forme spectrale typique et les variances $[r_{k,i}^2(f)]_f$ la variation de cette forme. Pour donner un exemple, un MMG log spectral à 16 états est représenté figure 2.10. Ce modèle est appris sur les même données que celui représenté figure 2.9.

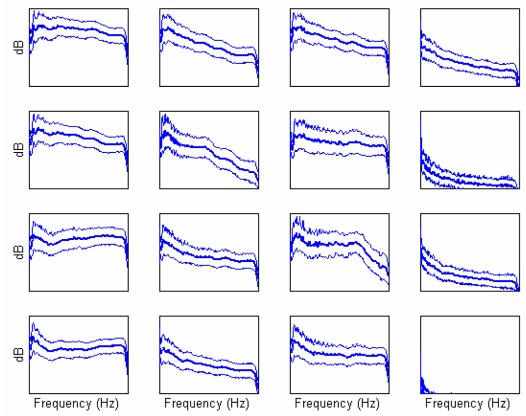


FIG. 2.10 – MMG log spectral à 16 états. Chaque état est représenté par sa DSP : $\mu_i(f)$ et DSP $\pm$ l'écart-type : $\mu_i(f) \pm r_i(f)$.

2.4.2.1 Apprentissage des MMG log spectraux

Comme pour les MMG spectraux (Sec. 2.4.1), l'apprentissage est basé sur le critère du MV (2.12) avec $\mathcal{L} = \log|\cdot|$ et il est réalisé en pratique à l'aide de l'algorithme EM [Dempster-77]. Les formules de réestimation des paramètres pour les MMG log spectraux [Rabiner-89] sont résumées par l'algorithme 3. L'algorithme des K-moyennes [McQueen-67] est utilisé pour l'initialisation de EM.

2.4.2.2 Distribution approchée du log spectre de mélange

Pour calculer l'estimateur minimisant l'EQM log spectrale, il faut pouvoir calculer la distribution du log spectre du mélange $\mathbf{X}(t) = \log|X(t)|$. Nous approchons l'équation du mélange (2.3) par l'expression suivante [Burshtein-99, Reddy-04] [6] :

$$|X(t,f)|^2 \approx |S_1(t,f)|^2 + |S_2(t,f)|^2 \tag{2.34}$$

on obtient :

$$\mathbf{X}(t,f) \approx \frac{1}{2}\log\left[\exp\{2\mathbf{S}_1(t,f)\} + \exp\{2\mathbf{S}_2(t,f)\}\right] \triangleq G[\mathbf{S}_1(t,f), \mathbf{S}_2(t,f)] \tag{2.35}$$

Les distributions de $\mathbf{S}_1(t)$ et $\mathbf{S}_2(t)$ sont connues (2.29). Pour simplifier le calcul de la distribution de $\mathbf{X}(t)$, la fonction non linéaire G est souvent approchée [Nadas-89, Moreno-96, Roweis-01].

[6] Dans la littérature [Burshtein-99, Reddy-04] l'approximation (2.34) est expliquée de la manière suivante : il est supposé que $S_1(t,f)$ et $S_2(t,f)$ sont des variables aléatoires complexes et qu'elles sont décorrélées, c'est-à-dire $\mathbb{E}[S_1(t,f)\overline{S_2(t,f)}] = 0$, où $\bar{V}$ est le conjugué d'un nombre complexe V. Ensuite, en partant de l'équation du mélange (2.3), on montre, grâce à la décorrélation, que $\mathbb{E}[|X(t,f)|^2] = \mathbb{E}[|S_1(t,f)|^2] + \mathbb{E}[|S_2(t,f)|^2]$. Enfin, en remplaçant l'opération $\mathbb{E}[|\cdot|^2]$ par $|\cdot|^2$ on obtient l'approximation (2.34). Toutefois, cette explication reste très discutable, puisque $S_1(t,f)$ et $S_2(t,f)$ ne sont pas des variables aléatoires

Algorithme 3 Algorithme EM pour l'apprentissage d'un MMG log spectral $\Lambda_k^{\log} = \{u_{k,i}, \mu_{k,i}, R_{k,i}\}_i$ à partir des données d'entraînement $\mathbf{Y}_k \triangleq \log |Y_k|$ (les paramètres estimés à la l-ème itération de EM sont notés par (l) en exposant).

1. Calculer les poids $\tilde{\gamma}_i^{(l)}(t)$ satisfaisant $\sum_i \tilde{\gamma}_i^{(l)}(t) = 1$ et

$$\tilde{\gamma}_i^{(l)}(t) \propto u_{k,i}^{(l)} N\left(\mathbf{Y}_k(t); \mu_{k,i}, R_{k,i}^{(l)}\right), \tag{2.30}$$

   où $N(\cdot)$ est défini selon (A.1).

2. Mettre à jour les poids de gaussiennes $u_{k,i}$:

$$u_{k,i}^{(l+1)} = \frac{1}{T} \sum_t \tilde{\gamma}_i^{(l)}(t) \tag{2.31}$$

3. Mettre à jour les vecteurs moyens $\mu_{k,i}$:

$$\mu_{k,i}^{(l+1)}(f) = \frac{\sum_t \tilde{\gamma}_i^{(l)}(t) \mathbf{Y}_k(t,f)}{\sum_t \tilde{\gamma}_i^{(l)}(t)} \tag{2.32}$$

4. Mettre à jour les matrices de covariances $R_{k,i}$:

$$r_{k,i}^{2,(l+1)}(f) = \frac{\sum_t \tilde{\gamma}_i^{(l)}(t) \left(\mathbf{Y}_k(t,f) - \mu_{k,i}^{(l+1)}(f)\right)^2}{\sum_t \tilde{\gamma}_i^{(l)}(t)} \tag{2.33}$$

Les approximations suivantes se trouvent dans la littérature :

- **MIXMAX** (*Mixture Maximum*) [Nadas-89] : la fonction G est approchée par le maximum de $\mathbf{S}_1(t,f)$ et $\mathbf{S}_2(t,f)$:

$$G[\mathbf{S}_1(t,f), \mathbf{S}_2(t,f)] \approx \max[\mathbf{S}_1(t,f), \mathbf{S}_2(t,f)] \tag{2.36}$$

- **VTS** (*Vector Taylor Series*) [Moreno-96] : conditionnellement à la paire d'états (i,j), la fonction G est approchée par son développement en série de Taylor d'ordre 1, calculé au point $(\mu_{1,i}(f), \mu_{2,j}(f))$:

$$\begin{aligned} G[\mathbf{S}_1(t,f), \mathbf{S}_2(t,f)] \approx& \\ \approx G[\mu_{1,i}(f), \mu_{2,j}(f)] + \nabla G[\mu_{1,i}(f), \mu_{2,j}(f)] &\left[\begin{array}{c} \mathbf{S}_1(t,f) - \mu_{1,i}(f) \\ \mathbf{S}_2(t,f) - \mu_{2,j}(f) \end{array} \right] \end{aligned} \tag{2.37}$$

  Toujours conditionnellement à (i,j), la distribution de $\mathbf{X}(t)$ est gaussienne, car l'approximation est linéaire et les distributions de $\mathbf{S}_1(t)$ et $\mathbf{S}_2(t)$ sont gaussiennes.

- **MeanMAX** (*Mean Maximum*) [Roweis-01] : conditionnellement à (i,j), le mélange $\mathbf{X}(t)$ est supposé gaussien avec les paramètres suivants :

$$\begin{cases} \mathbf{X}(t,f) \sim \mathcal{N}(\mu_{1,i}(f), r_{1,i}^2(f)), & \mu_{1,i}(f) \geq \mu_{2,j}(f) \\ \mathbf{X}(t,f) \sim \mathcal{N}(\mu_{2,j}(f), r_{2,j}^2(f)), & \mu_{1,i}(f) \leq \mu_{2,j}(f) \end{cases} \tag{2.38}$$

Ces approximations sont comparées sur la figure 2.11. Premièrement, on peut remarquer qu'en éloignant les moyennes de $\mathbf{S}_1(t,f)$ et $\mathbf{S}_2(t,f)$, toutes les approximations s'approchent de la distribution exacte. Deuxièmement, la précision des approximations semble décroître dans l'ordre de leur présentation. Nous verrons dans le chapitre 4 consacré aux expérimentations préliminaires que les performances de séparation décroissent aussi dans le même ordre. L'approximation MIXMAX est donc la plus précise et c'est celle que nous utiliserons dans la section suivante pour présenter l'estimateur minimisant l'EQM log spectrale.

2.4.2.3 Estimateur minimisant l'EQM log spectrale

Avec l'approximation MIXMAX (2.36), on peut obtenir l'estimateur minimisant l'EQM log spectrale (2.25). Le gain $\mathcal{M}_1(t,f)$ de cet estimateur se calcule comme suit [Burshtein-99] :

$$\mathcal{M}_1^{\text{log\_log}}(t,f) = \exp\left[\sum_{i,j} \tilde{\gamma}_{i,j}(t) \frac{\left[\mu_{1,i}(f) - r_{1,i}^2(f)\Upsilon_{1,i}(t,f) - \mathbf{X}(t,f)\right] \Upsilon_{2,j}(t,f)}{\Upsilon_{1,i}(t,f) + \Upsilon_{2,j}(t,f)}\right], \tag{2.39}$$

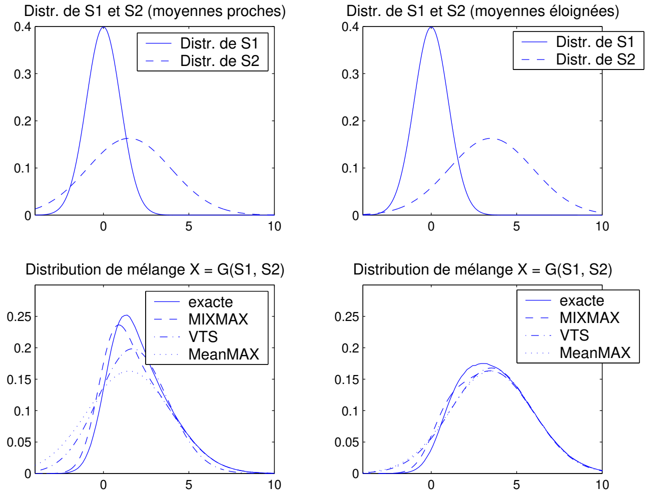


FIG. 2.11 – En haut : Distributions gaussiennes de $\mathbf{S}_1(t,f)$ et de $\mathbf{S}_2(t,f)$ conditionnellement à la paire d'états (i,j). Deux scénarios sont représentés : moyennes proches (à gauche) et moyennes éloignées (à droite). En bas : distribution exacte du mélange $\mathbf{X}(t,f)$ et des approximations.

où les probabilités $\tilde{\gamma}_{i,j}(t)$ (satisfaisant $\sum_{i,j}\tilde{\gamma}_{i,j}(t)=1$) et les grandeurs $\Upsilon_{k,i}(t,f)$ sont calculées selon les formules suivantes :

$$\tilde{\gamma}_{i,j}(t) \propto u_{1,i}u_{2,j}\prod_f \left[\phi_{1,i}(t,f)\Phi_{2,j}(t,f)+\phi_{2,j}(t,f)\Phi_{1,i}(t,f)\right] \tag{2.40}$$

$$\Upsilon_{k,i}(t,f) \triangleq \phi_{k,i}(t,f)/\Phi_{k,i}(t,f) \tag{2.41}$$

avec les grandeurs $\phi_{k,i}(t,f)$ et $\Phi_{k,i}(t,f)$ définies comme :

$$\phi_{k,i}(t,f) \triangleq N\left(\mathbf{X}(t,f);\mu_{k,i}(f),r_{k,i}^2(f)\right) \tag{2.42}$$

$$\Phi_{k,i}(t,f) \triangleq \Phi\left[\frac{\mathbf{X}(t,f)-\mu_{k,i}(f)}{r_{k,i}(f)}\right] \tag{2.43}$$

où la densité $N(\cdot)$ est définie par (A.1) [7] et la fonction $\Phi(\cdot)$ (la fonction de répartition de la loi normale centrée de variance unitaire) est définie comme :

$$\Phi(\tau)=\int_{-\infty}^{\tau}\frac{1}{\sqrt{2\pi}}e^{-\frac{w^2}{2}}dw=\frac{1}{2}\left[1+\operatorname{erf}\left(\frac{\tau}{\sqrt{2}}\right)\right] \tag{2.44}$$

où `erf` s'appelle *fonction d'erreur* ($\operatorname{erf}(\tau)=\frac{2}{\sqrt{\pi}}\int_0^\tau e^{-w^2}dw$).

Pour ce qui concerne l'estimation de la phase de la TFCT, la même remarque que dans la

[7] Car $\mathbf{X}(t,f)$ dans (2.42) est un scalaire, c'est un cas particulier de la formule (A.1) pour une variable aléatoire, c'est-à-dire un vecteur aléatoire de taille 1.

section 2.4.1.5 peut être faite.

2.5 Conclusion

Le problème de séparation de sources avec un seul capteur à été formulé et un état de l'art sur les méthodes basées sur des modèles *a priori* à été présenté.

Dans cette thèse, nous allons utiliser les MMG comme modèles des sources, sachant que l'extension aux MMC est assez facile à faire. Les méthodes basées sur les MMG semblent prometteuses pour de nombreuses tâches de séparation de sources avec un seul capteur. Cependant, ces méthodes souffrent de problèmes majeures, notamment dans le cas des classes sonores de grande variabilité :

1. il est difficile de construire des bases d'entraînement représentatives,
2. étant limité en pratique par des ressource calculatoires, on ne peut pas traiter des modèles de très grand taille.

Ainsi, nous avons besoin de passer aux expérimentations pour

- déterminer, si les problèmes annoncés se manifestent pour la tâche de séparation voix / musique,
- choisir une méthode particulière parmi les méthodes basées sur les MMG, présentées section 2.4.

Avant de passer aux expérimentations nous avons besoin d'introduire des outils d'évaluation et de diagnostic des algorithmes de séparation, ce qui sera fait dans le chapitre 3. Les expérimentations préliminaires seront ensuite présentées dans le chapitre 4.

Chapitre 3

Evaluation et diagnostic

Dans ce chapitre nous présentons d'abord différentes techniques d'évaluation des algorithmes de séparation. Ensuite, nous introduisons des estimateurs oracle permettant de calculer les limites de performance de séparation.

3.1 Evaluation de la qualité de séparation

La "qualité de séparation" n'est pas une notion absolue. En effet, comme il est remarqué dans la section 1.3, la séparation est souvent effectuée en visant une application particulière. Dans ce cas, la qualité de séparation est conditionnée par l'application visée, puisqu'une méthode de séparation peut donner de meilleurs résultats pour une application que pour une autre. Par exemple, pour l'estimation du pitch de la voix chantée, il faut bien séparer les sons voisés de la voix, tandis que les sons non voisés peuvent être séparés moins bien (à la limite ils peuvent même être supprimées dans l'estimation de source de la voix). En revanche, cela ne sera pas acceptable pour la reconnaissance de la parole chantée, car les sons non voisés contiennent des informations sur les consonnes prononcées. Ainsi, si une méthode de séparation est développée pour une application particulière, il est souvent plus judicieux d'évaluer cette méthode au travers de cette application.

Cependant, parfois, on a besoin de développer une méthode de séparation en s'affranchissant de toute application éventuelle ou bien de la développer pour plusieurs applications différentes. Dans ce cas, on peut utiliser des mesures objectives de performance de séparation qui ne dépendent pas (en tous cas pas directement) de l'application visée.

Ainsi, nous distinguons des procédures d'évaluation de deux types :

1. **Evaluation dépendant de l'application**, c'est-à-dire au travers de l'application visée. Les sources séparées sont utilisées par l'application visée et un résultat est obtenu. Ensuite, la qualité de ce résultat est mesurée en utilisant une mesure de performance pour cette

application (Fig. 3.1 (A)). Par exemple, si l'application visée est la reconnaissance automatique de la parole, la précision de reconnaissance des mots (voir par ex. [Ozerov-03]) peut être choisie comme mesure de performance d'application.

2. **Evaluation indépendant de l'application**, c'est-à-dire en utilisant une mesure objective de performance de séparation. La performance de séparation est souvent estimée comme un degré de ressemblance des sources estimées $\hat{s}_k$ aux sources originales s_k (Fig. 3.1 (B)). Ainsi, les sources originales doivent être disponibles dans le cadre expérimental. Le degré de ressemblance peut être calculé différemment, d'où les différentes mesures de performance.

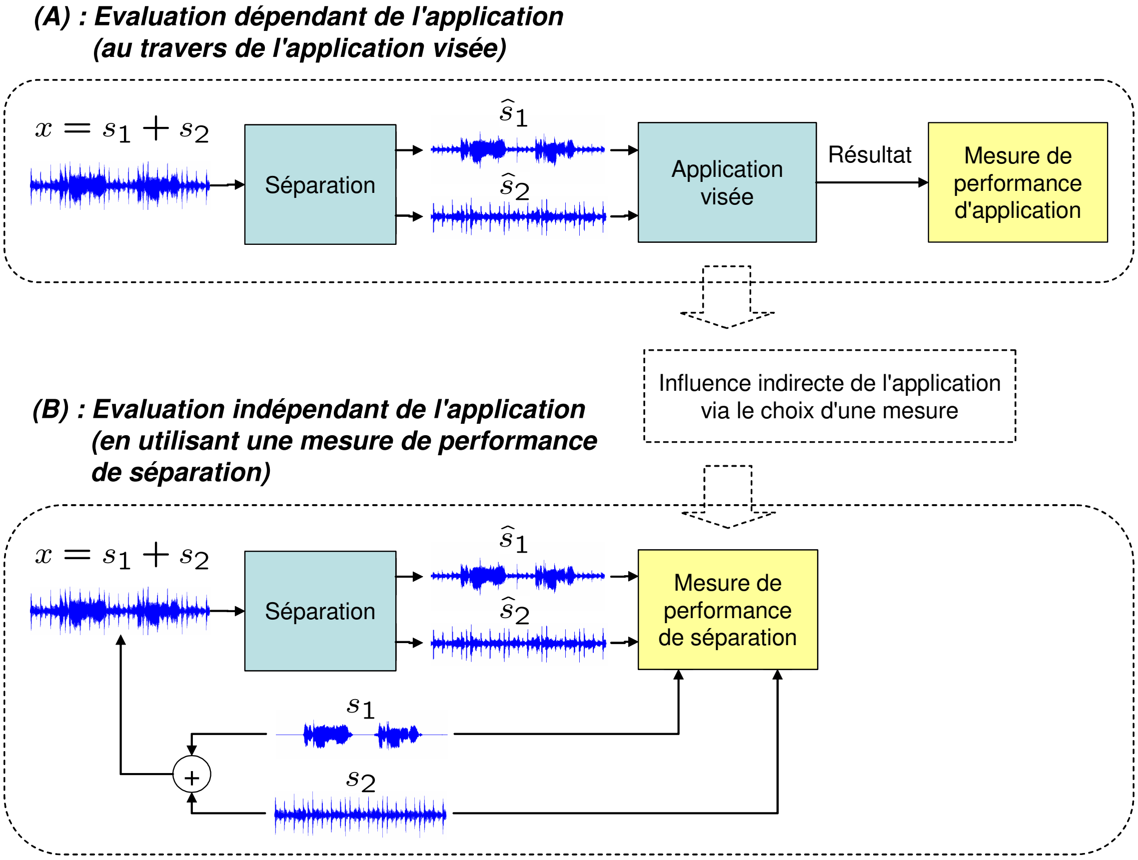


FIG. 3.1 – Deux types de procédures d'évaluation d'un système de séparation de sources. **(A)** : Evaluation dépendant de l'application. **(B)** : Evaluation indépendant de l'application.

Notons que l'évaluation indépendant de l'application (à l'aide d'une mesure de performance de séparation) peut être influencée par une application visée de manière indirecte. En effet, pour une application donnée certaines mesures de performance de séparation sont plus pertinentes que d'autres. Ainsi, une application visée peut influencer la procédure d'évaluation indépendant de l'application via le choix d'une mesure particulière (Fig. 3.1).

Nous allons évaluer le système de séparation voix / musique sous les deux angles à la fois, c'est-à-dire indépendamment et dépendamment de l'application. Le système sera développé en utilisant une mesure de performance de séparation (indépendamment de l'application). Certaines mesures de performance de séparation seront présentées dans la section suivante. A la fin du développement du système de séparation, son apport pour l'estimation du pitch de la voix chantée sera mesuré (évaluation dépendant de l'application). La mesure utilisée pour évaluer la qualité de l'estimation du pitch (mesure de performance pour l'application visée) sera présentée à son tour dans le chapitre 13.

3.2 Mesures de performance de séparation

Comme il est remarqué section 1.2.2, certaines méthodes de la séparation de sources permettent d'estimer chaque source originale s_k à une transformation près (par ex. à un gain près ou à un filtre près) et à une permutation près des indices. Ainsi, une des particularités des mesures de performance pour la séparation est qu'elles doivent être invariantes par rapport à ces transformations et aux permutations des indices des sources [Gribonval-03, Vincent-05a]. Dans le cas du problème de séparation de sources avec un seul capteur, tel qu'il est formulé dans le chapitre 2 (voir équation (2.2)), il n'y a pas besoin que la mesure de performance soit invariante ni par rapport aux transformations, ni par rapport aux permutations des indices. Ainsi, les mesures de performance pour le débruitage de la parole qui ne possèdent pas ces invariances peuvent être réutilisées directement pour la séparation.

La suite de cette section est organisée de manière suivante :

- présentation des mesures de performance héritées du débruitage de la parole,
- présentation des mesures de performance pour la séparation de sources,
- introduction des *mesures normalisées* qui sont développées dans le cadre de cette thèse et possèdent des propriétés intéressantes.

3.2.1 Mesures héritées du débruitage de la parole

Les mesures suivantes développées au départ pour le débruitage de la parole sont considérées ici :

- Le Rapport Signal à Bruit (RSB) [Deller-99] :

$$\mathrm{RSB}(\hat{s}_k, s_k) = 10 \log_{10} \left[\frac{\|s_k\|_2^2}{\|\hat{s}_k - s_k\|_2^2} \right] \tag{3.1}$$

- Le RSB Segmental qui est la moyenne des RSB calculés sur de segments temporels de courte durée (typiquement 15 - 25 msec.) [Deller-99].
- La Distorsion du Log Spectre (DLS) [Valin-04] :

$$\mathrm{DLS}(\hat{s}_k, s_k) = \frac{1}{T}\sum_{t=1}^{T}\left[\frac{1}{F}\sum_{f=1}^{F}\left(10\log_{10}\frac{|S_k(t,f)|^2+\epsilon}{|\widehat{\mathbb{S}}_k(t,f)|^2+\epsilon}\right)^2\right]^{\frac{1}{2}} \tag{3.2}$$

où $S_k(t,f)$ et $\widehat{\mathbb{S}}_k(t,f)$ [1] sont des Transformées de Fourier à Court Terme (TFCT) de s_k et $\hat{s}_k$. Pour calculer cette TFCT, nous utiliserons la même fenêtre d'analyse que celle utilisée pour la séparation. La constante ϵ est ajoutée pour éviter que ce critère ne devienne égal à $-\infty$ quand $S_k(t,f) = 0$. Elle correspond à un bruit dont l'énergie est 100 dB plus petite que celle de la source, autrement dit, elle est calculée comme suit : $\epsilon = 10^{-\mathbf{100}/10}\|S_k\|_2^2$.

3.2.2 Mesures pour la séparation de sources

Cette section présente des mesures de performance pour la séparation de sources. Bien que ces mesures [Gribonval-03, Vincent-05a] aient été introduites initialement pour des problèmes de séparation de sources avec plusieurs capteurs et plusieurs sources (1.3) ou (1.4), elles seront présentées ici pour notre cadre de travail (voir équation (2.2)).

Le Rapport Source à Distorsion (RSD) [Gribonval-03] est défini comme suit :

$$\mathrm{RSD}(\hat{s}_k, s_k) = 10\log_{10}\left[\frac{\langle \hat{s}_k, s_k\rangle^2}{\|\hat{s}_k\|_2^2\|s_k\|_2^2 - \langle \hat{s}_k, s_k\rangle^2}\right] \tag{3.3}$$

où $\hat{s}_k$ est une estimation de la source s_k.

Le RSD est invariant aux gains multiplicatifs des sources estimées. Cette propriété est attribuée au RSD puisque, pour le modèle du mélange linéaire instantané (1.3), les sources ne peuvent être estimées qu'aux gains multiplicatifs près.

Le RSD peut être aussi partagé entre un Rapport Source à Interférences (RSI) et un Rapport Source à Artefacts (RSA) [Gribonval-03] ce qui permet de faire de meilleurs diagnostics. De plus, ces mesures ont été généralisées pour devenir invariantes par rapport aux filtres constants ou variables au cours du temps [Vincent-05a].

3.2.3 Mesures normalisées

Un des problèmes des mesures présentées précédemment est qu'elles représentent des grandeurs absolues. Expliquons ceci sur un exemple.

Considérons deux mélanges $x' = s_1 + s_2'$ et $x'' = s_1 + s_2''$ et supposons que l'énergie de s_2' soit beaucoup moins importante que celle de s_2''. Ainsi, dans le mélange x' la première source s_1 est moins polluée par la deuxième source que dans le mélange x''. Il est facile de concevoir que

[1] Nous avons volontairement noté la *TFCT de l'estimation* $\widehat{\mathbb{S}}_k = \mathrm{TFCT}[\mathrm{TFCT}^{-1}(\widehat{S}_k)]$ par une lettre ajourée pour pouvoir la différencier de l'*estimation de la TFCT* $\widehat{S}_k$, qui en général ne coïncide pas avec $\widehat{\mathbb{S}}_k$, car la TFCT est une transformée redondante. Nous reviendrons sur ce point par la suite.

dans la plupart des cas, en appliquant le même algorithme pour estimer la source s_1 à partir de x' et x'', on obtiendra un meilleur (plus grand) RSD$(\hat{s}_1, s_1)$ dans le premier cas que dans le deuxième. Ceci n'est pas dû à une meilleure performance de l'algorithme dans le premier cas, mais au fait que la source s_1 est moins polluée par la deuxième source dès le départ.

Ainsi, la comparaison des RSD calculés sur des résultats de séparation des mélanges différents (sans prendre en compte les rapports des énergies des sources dans ces mélanges) pourra mener à des conclusions biaisées. Pour la même raison, le RSD moyen calculé pour un algorithme testé sur des séquences différentes ne reflètera pas la performance moyenne obtenue sur ces séquences.

De plus, nous avons observé que le RSD calculé en remplaçant l'estimation de source $\hat{s}_1$ par le mélange x, c'est-à-dire RSD(x, s_k), représente dans un certain sens le rapport des énergies des sources ou bien la difficulté de la tâche à accomplir. Par exemple, on voit sur la figure 3.2 que pour le 5-ème enregistrement, le RSD$(\hat{s}_k, s_k)$ est très petit, même négatif, et ceci est dû au fait que le RSD(x, s_k) est petit. Autrement dit, pour cet enregistrement la séparation est difficile dès le départ.

Nous proposons dans cette thèse d'utiliser des mesures normalisées représentant des grandeurs relatives. *Une mesure normalisée est l'amélioration de la mesure correspondante par rapport à la "séparation passive" qui consiste à prendre le mélange x pour l'estimation de source.* Ainsi, une mesure normalisée représente l'effort fait par un algorithme de séparation par rapport à ne rien faire en prenant x pour l'estimation de source ("séparation passive").

Par exemple, le RSD Normalisé (RSDN) [Ozerov-05a] mesure l'amélioration du RSD entre le signal non traité x et l'estimation $\hat{s}_k$:

$$\mathrm{RSDN}(\hat{s}_k, s_k, x) = \mathrm{RSD}(\hat{s}_k, s_k) - \mathrm{RSD}(x, s_k) \tag{3.4}$$

Un exemple de calcul du RSDN à partir de RSD$(\hat{s}_k, s_k)$ et RSD(x, s_k) (la difficulté de la tâche) est représenté sur la figure 3.2.

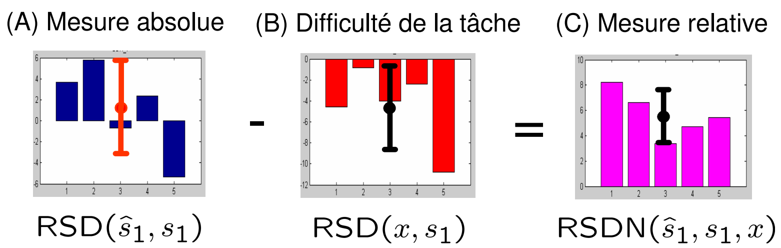


FIG. 3.2 – Interprétation du RSDN pour la séparation de 5 enregistrements différents. La barre verticale au milieu de chaque graphe représente l'écart-type.

De la même façon, la DLS Normalisée (DLSN) [Ozerov-05a] est définie comme l'amélioration de la DLS (3.2) entre x et $\hat{s}_k$:

$$\text{DLSN}(\hat{s}_k, s_k, x) = \text{DLS}(x, s_k) - \text{DLS}(\hat{s}_k, s_k), \tag{3.5}$$

Cette idée de normalisation était déjà apparue dans la littérature. Par exemple, Ayewah et Seidel [Ayewah-04] utilisent pour le débruitage de la parole *l'augmentation du RSB* (RSB $\uparrow$) calculée en utilisant le même principe.

Nous pensons que les mesures normalisées s'appliquent beaucoup plus largement aux problèmes de séparation de sources. C'est sont ces mesures, notamment le RSDN et la DLSN, que nous allons utiliser dans le but de notre évaluation.

3.3 Estimateurs oracles et limites de performance

Les masques oracles ont été mentionnés section 2.1.3. Ici nous faisons une introduction plus conséquente de la notion d'estimateur oracle [Vincent-05b] permettant de calculer les limites de performance qu'on ne peut pas envisager de dépasser avec la méthode de séparation choisie.

Remarquons que dans le cas des méthodes basées sur des MMG, l'estimation d'une source $\hat{s}_k$ ne dépend que du mélange x et du masque temps - fréquence $\mathcal{M}_k = [\mathcal{M}_k(t, f)]_{t,f}$ (Fig. 2.2 et 2.8). Il est donc possible d'exprimer cette estimation comme $\hat{s}_k = g(x, \mathcal{M}_k)$, où $g(\cdot, \cdot)$ est une certaine transformée. Le masque $\mathcal{M}_k$ appartient à un ensemble de masques admissibles $\mathbb{M}$ dépendant de la méthode de séparation. Ici, on considère $\mathbb{M}_{[0,1]} = \{\mathcal{M}_k | \mathcal{M}_k(t, f) \in [0, 1]\}$ pour le filtrage de Wiener pondéré (2.20) et $\mathbb{M}_+ = \{\mathcal{M}_k | \mathcal{M}_k(t, f) \geq 0\}$ pour les méthodes minimisant l'EQM log spectrale (2.26) et (2.39). Etant donné une mesure de performance $h(\hat{s}_k, s_k, x)$, *l'estimateur oracle* consiste à trouver le masque $\mathcal{M}_k^* \in \mathbb{M}$ qui donne la meilleure performance [Vincent-05b] :

$$s_k^* = g(x, \mathcal{M}_k^*), \quad \mathcal{M}_k^* = \arg \max_{\mathcal{M}_k \in \mathbb{M}} h(g(x, \mathcal{M}_k), s_k, x) \tag{3.6}$$

Cet estimateur permet de calculer, pour un jeu de données, la limite de performance qui ne peut pas être dépassée avec la méthode correspondante.

Remarquons que généralement des limites de performances calculées à l'aide d'oracles sont trop optimistes. En effet, chaque gain d'un masque temps - fréquence oracle $\mathcal{M}_k^*$ est finement réglé en utilisant la connaissance des sources s_k. De plus, par analogie à ce qui est remarqué section 2.1.3, l'hypothèse de WDO (2.4) étant vérifiée, une masque temps - fréquence oracle $\mathcal{M}_k^*$ mène à une estimation exacte de la source ($s_k^* = s_k$). Ainsi, d'une part il ne faut pas être déçu tout de suite si on est loin des limites de performances indiquées par des oracles, d'autre part, il faut être très content si on arrive à s'en approcher.

En pratique il est difficile de calculer l'oracle (3.6) pour les mesures de performance RSDN et DLSN, car il devient compliqué d'optimiser analytiquement le problème de maximisation (3.6).

En remplacement, on calcule les estimateurs oracles pour le Rapport Signal à Bruit (RSB) spectral défini directement dans le domaine de la TFCT comme :

$$\mathrm{RSB}_{\mathrm{spec}}(\widehat{S}_k, S_k) = 10 \log_{10} \left[\frac{\|S_k\|_2^2}{\|\widehat{S}_k - S_k\|_2^2} \right] \tag{3.7}$$

Puis les valeurs du RSDN et de la DLSN sont calculées sur les sources estimées à l'aide des masques oracles correspondants.

Cela peut être résumé par des étapes suivantes :

1. Calculer le masque oracle pour le RSB spectral (3.7) et un ensemble de masques admissibles $\mathbb{M}$ ($\mathbb{M}_{[0,1]}$ ou $\mathbb{M}_+$) :

$$\mathcal{M}_k^* = \arg \max_{\mathcal{M}_k \in \mathbb{M}} \mathrm{RSB}_{\mathrm{spec}}(\mathcal{M}_k \times X, S_k), \tag{3.8}$$

   où l'opération $\times$ signifie la multiplication des matrices élément par élément. Selon l'ensemble des masques admissibles ($\mathbb{M}_{[0,1]}$ ou $\mathbb{M}_+$), ce masque oracle se calcule comme suit :

$$\mathcal{M}_k^{*+}(t,f) = \max \left[0, \frac{1}{|X(t,f)|^2} \Re \left(X(t,f) \overline{S_k(t,f)} \right) \right] \quad \text{ou} \tag{3.9}$$

$$\mathcal{M}_k^{*[0,1]}(t,f) = \min \left[\mathcal{M}_k^{*+}(t,f), 1 \right], \tag{3.10}$$

   où $\bar{V}$ représente le conjugué d'un nombre complexe V et $\Re V$ représente sa partie réelle.
2. Calculer la source estimée à l'aide du masque oracle : $s_k^* = \mathrm{TFCT}^{-1}(\mathcal{M}_k^* \times X)$.
3. Calculer des valeurs du RSDN et de la DLSN (voir (3.4), (3.5)) à partir de s_k^*, s_k et x.

Les valeurs du RSDN et de la DLSN ainsi obtenues ne sont plus de véritables limites de performances qui ne peuvent pas être dépassée. Elles peuvent être dépassées en général, car nous avons utilisé le RSB spectral pour le calcul du masque oracle (3.8) au lieu d'utiliser le RSDN ou la DLSN. Cependant, ces *oracles approchés* donnent des performances presque aussi optimistes que des vrais oracles, et c'est toujours un défi d'essayer de s'en approcher.

Nous utiliserons les oracles approchés dans le cadre des expérimentations menées dans le chapitre suivant. Leur utilisation nous permettra de calculer des performances (trop optimistes en général) qui peuvent être potentiellement atteintes avec les techniques de masquage étudiées.

3.4 Résumé

Après avoir passé en revue les mesures de performance utilisées pour la séparation de sources ainsi que pour le débruitage de la parole, nous avons introduit de nouvelles mesures normalisées.

Ces mesures semblent plus appropriées pour exprimer la performance moyenne d'un algorithme de séparation sur plusieurs enregistrements différents.

De plus, nous avons présenté des estimateurs oracles permettant de calculer les limites de performance d'un algorithme de séparation. Pour certaines mesures de performance, il est difficile de calculer les oracles. Dans ce cas, nous avons proposé de calculer des oracles approchés qui ne permettent plus d'obtenir des limites de performances, mais des performances assez élevées et très optimistes qui peuvent être atteintes avec les techniques de masquage utilisées.

Certains outils d'évaluation et de diagnostic présentés ici seront utilisés pour les expérimentations préliminaires menées dans le chapitre suivant. Notamment, nous utiliserons des mesures normalisées, telles que le RSDN et la DLSN, et des oracles approchés calculés pour ces mesures.

Chapitre 4

Expérimentations préliminaires dans le cadre de la séparation voix / musique

Ce chapitre présente quelques expérimentations dans le cadre de séparation de la voix chantée par rapport à la musique ambiante dans des chansons populaires. Nous avons procédé à ces expérimentations pour fixer certains paramètres, pour consolider la méthode de séparation utilisée par la suite, et pour identifier des limitations des méthodes actuelles. Ceci permettra de définir ensuite la problématique traitée dans cette thèse. Nous avons déjà annoncé certaines limitations majeures, qui peuvent être résumées comme l'incapacité de modéliser assez finement des classes sonores de grande variabilité. Ainsi, ces expérimentations nous permettrons de voir comment ces limitations se manifestent pour la séparation voix / musique.

4.1 Problème de séparation voix / musique

Dans le contexte de la tâche particulière de séparation voix / musique, il est supposé que chaque enregistrement (mono) d'une chanson

$$x(n) = s_v(n) + s_m(n) \tag{4.1}$$

est une somme de deux sources : la voix $s_v(n)$ et de la musique $s_m(n)$ [1]. Le problème reste le même (Chap. 2), c'est-à-dire trouver des estimations de la voix $\hat{s}_v(n)$ et de la musique $\hat{s}_m(n)$.

[1]Pour ne pas confondre ces deux sources particulières (c'est-à-dire la voix et la musique), à chaque fois quand il s'agit de la séparation voix / musique, les indices v et m sont utilisés à la place des indices 1 et 2.

4.2 Objectifs des expérimentations préliminaires

Les objectifs principaux des expérimentations préliminaires sont :

- identifier les limitations des méthodes basées sur des MMG,
- choisir certains paramètres (la taille et le type de la fenêtre d'analyse de la TFCT),
- affiner la méthode de séparation basée sur des MMG (MMG spectral / log spectral, EQM spectrale / log spectrale),
- illustrer expérimentalement certaines affirmations faites dans la section 2.4 (par ex. la précision des estimateurs durs),

4.3 Mesures de performance

Pour mesurer la performance de séparation, nous utilisons les mesures normalisées que nous avons introduites dans la section 3.2.3, notamment le RSDN (3.4) et la DLSN (3.5).

Dans le cadre de l'application de la séparation voix / musique à l'indexation audio (Sec. 1.3), nous sommes surtout intéressés par l'estimation de la voix $\hat{s}_v$. Ainsi, les performances que nous présentons par la suite (par ex. les RSDN) sont calculées pour la voix (c'est-à-dire RSDN$(\hat{s}_v, s_v, x)$) et non pas pour la musique.

Pour estimer la performance globale, nous utilisons la moyenne des RSDN ou des DLSN calculés pour tous les enregistrements de test.

4.4 Description des données expérimentales pour la séparation

Avant de passer aux expérimentations, nous présentons les données expérimentales, c'est-à-dire les données d'apprentissage des modèles généraux Λ_v et Λ_m et les données de test pour la séparation.

La base d'entraînement du modèle général de voix Λ_v contient 34 extraits de voix chantée issus de chansons populaires. Chaque extrait dure approximativement une minute. Le modèle général de musique Λ_m est appris sur 30 extraits de musique populaire sans voix. Chaque extrait dure également environ une minute.

La base d'évaluation contient 6 chansons du même genre pour lesquelles les pistes de voix et de musique sont disponibles séparément, ce qui permet d'évaluer la performance de la séparation en comparant l'estimation à l'original. La taille médiocre de cette base d'évaluation est liée au fait qu'il est difficile de trouver des données de ce type.

La base d'entraînement et celle d'évaluation sont bien distinctes entre elles, notamment les oeuvres d'un même artiste n'interviennent jamais dans les deux bases.

Tous les enregistrements utilisés sont en mono et échantillonnés à 11025 Hz. Nous avons choisi cette fréquence d'échantillonnage car elle nous semble être un bon compromis entre la qualité et la complexité calculatoire. En particulier, ce choix est basé sur le fait qu'à l'heure actuelle, la qualité audio des signaux qu'on peut obtenir à l'aide des techniques de séparation de sources mono-capteur est assez basse. De plus, pour l'application que nous allons traiter par la suite, c'est-à-dire pour l'estimation du pitch de la voix chantée, la fréquence d'échantillonnage de 11025 Hz paraît suffisante.

4.5 Expérimentations et résultats

Les quatre expérimentations présentées par la suite apportent des réponses aux questions suivantes :

1. choix des paramètres de la TFCT (taille et type de la fenêtre d'analyse),
2. effet de l'hétérogénéité entre données d'apprentissage et de test et effet du dimensionnement des modèles (nombre d'états),
3. choix du domaine de modélisation (MMG spectral / log spectral) et de la mesure de distorsion minimisée (EQM spectrale / log spectrale),
4. précision des estimateurs durs par rapport aux estimateurs doux,

4.5.1 Choix de la fenêtre d'analyse

En utilisant le masque oracle $\mathcal{M}_v^{*[0,1]}$ (3.10) pour estimer les sources, nous avons étudié le comportement du RSDN (3.4) moyen en fonction de la taille et du type de fenêtre d'analyse de la TFCT (Fig. 4.1). Le meilleur résultat est obtenu avec une fenêtre de Hamming de taille 1024 échantillons (soit 93 ms), qui sera utilisée pour le reste des expériences menées dans cette thèse.

Notons que la taille optimale de la fenêtre (93 ms) est beaucoup plus longue que celle habituellement utilisée pour l'analyse de la parole parlée (20 - 50 ms). Ceci est peut-être lié au fait que pour la voix chantée, ainsi que pour la musique, la durée moyenne de la stationnarité locale est plausiblement plus longue que celle de parole parlée.

4.5.2 Effet de l'hétérogénéité entre données d'apprentissage et de test et effet du dimensionnement des modèles

En plus des modèles généraux Λ_v et Λ_m, nous considérons ici les *modèles idéaux* λ_v^{Idl} et λ_m^{Idl}. Ces modèles sont appris sur les sources séparées s_v et s_m disponibles dans le contexte expérimental. La performance de séparation obtenue avec ces modèles, qui est inaccessible dans un cadre d'application réelle, joue le rôle d'une borne empirique supérieure pour les performances

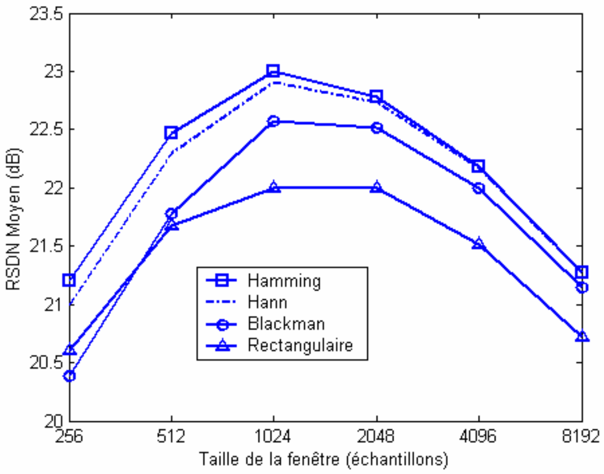


FIG. 4.1 – Le comportement du RSDN moyen pour l'estimateur oracle en fonction de la taille et du type de fenêtre d'analyse de la TFCT.

qui pourraient être atteintes avec des modèles de même structure et même taille obtenus dans un contexte réaliste (par ex. modèles généraux).

Avec les MMG spectraux (2.15) et l'estimateur minimisant l'EQM spectrale (2.20), nous avons testé l'effet sur le RSDN du nombre de gaussiennes (d'états) des MMG de voix et de musique $Q = Q_v = Q_m$ dans les configurations suivantes :

1. **Modèles généraux** Λ_v et Λ_m appris sur les données d'entraînement y_v et y_m (Alg. 2).
2. **Modèles idéaux** λ_v^{Idl} et λ_m^{Idl} appris sur les sources séparées s_v et s_m (inaccessibles dans un cadre réel). L'algorithme 2 a été également utilisé pour cet apprentissage, en remplaçant Y_k par S_k.

Les performances moyennes pour six chansons de test sont résumées sur la figure 4.2.

Avec deux modèles généraux Λ_v et Λ_m, l'augmentation du nombre de gaussiennes Q n'améliore pas sensiblement la performance par rapport au filtrage de Wiener monogaussien ($Q = 1$), voire la fait légèrement décroître. Remarquons que dans le cas $Q = 1$, le RSDN moyen de 5.2 dB est obtenu par un filtrage linéaire simple avec un filtre de Wiener passe haut dont la fréquence de coupure est située vers 300 Hz (Fig. 4.4). En regardant les modèles généraux à 16 états (Fig. 4.5 (A) et (C)), on pourrait dire que le modèle de voix est plus structuré que celui de musique.

Cependant, avec les modèles idéaux λ_v^{Idl} et λ_m^{Idl} qui sont bien adaptés aux sources, mais irréalistes en pratique, les performances sont bien meilleures qu'avec les modèles généraux et ces performances peuvent être sensiblement améliorées en augmentant le nombre de gaussiennes. En effet, le RSDN moyen s'améliore de 6 dB en passant de $Q = 1$ à $Q = 32$. Un modèle de musique idéal est représenté sur la figure 4.5 (D).

Les résultats détaillés (le RSDN pour chaque chanson de test) de la même expérience représentés sur la figure 4.3 confirment les tendances observées en moyenne.

La conclusion importante qui peut être tirée de cette expérience est que pour cette tâche de séparation, il n'y a pas d'intérêt à utiliser des modèles généraux, car ils ne permettent pas d'améliorer sensiblement les performances moyennes par rapport au cas trivial du filtrage de Wiener monogaussien ($Q = 1$). Par contre, il existe des modèles de même structure, tels que des modèles idéaux, dont l'utilisation permet de dépasser considérablement cette limitation. Très vraisemblablement, ceci est dû au fait que les modèles idéaux sont bien adaptés aux sources séparées. Cependant, l'utilisation de ces modèles n'est pas réaliste. Ainsi, la question cruciale est d'obtenir de manière réaliste des modèles bien adaptés aux sources.

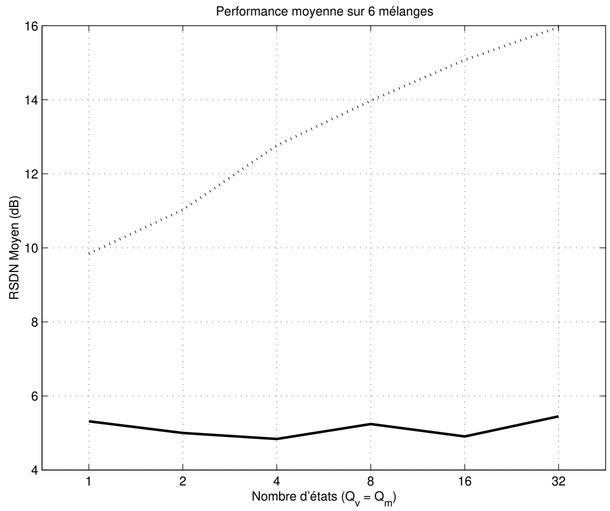


FIG. 4.2 – RSDN moyen pour six chansons de test en fonction du nombre d'états des modèles $Q = Q_v = Q_m$ et pour les différents types de modèles. Ligne continue : Modèles généraux (Λ_v, Λ_m) ; Pointillés : Modèles idéaux (borne empirique) $(\lambda_v^{\mathrm{Idl}}, \lambda_m^{\mathrm{Idl}})$.

4.5.3 Effets du domaine de modélisation et de la mesure de distorsion

Nous allons comparer les trois méthodes de séparation basées sur des MMG, qui sont présentées dans la section 2.4, afin de choisir la méthode qui sera utilisée par la suite. Chaque méthode est définie par les deux caractéristiques suivantes :

– domaine de modélisation : MMG spectral (Sec. 2.4.1) ou MMG log spectral (Sec. 2.4.2),

– critère d'erreur minimisé : EQM spectrale (2.19) ou EQM log spectrale (2.25),

et les méthodes étudiées correspondent aux trois combinaisons suivantes :

1. MMG spectral / EQM spectrale (Sec. 2.4.1) [Benaroya-03],
2. MMG spectral / EQM log spectrale (Sec. 2.4.1) [Ephraim-85],

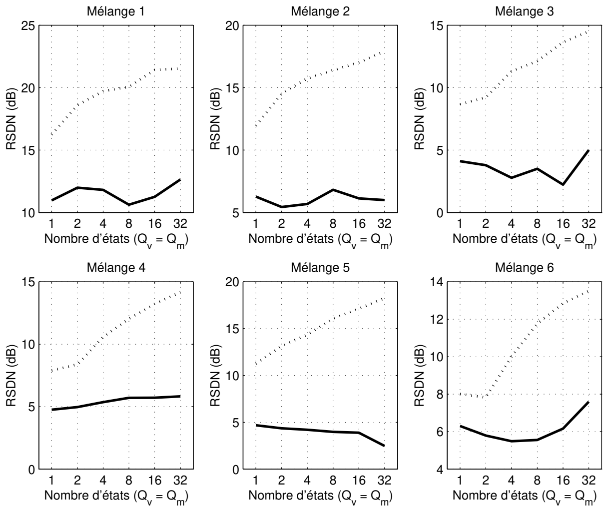


FIG. 4.3 – RSDN détaillé pour six chansons de test en fonction du nombre d'états des modèles $Q = Q_v = Q_m$ et pour différents types de modèles. Ligne continue : Modèles généraux (Λ_v, Λ_m) ; Pointillés : Modèles idéaux (borne empirique) $(\lambda_v^{\text{Idl}}, \lambda_m^{\text{Idl}})$.

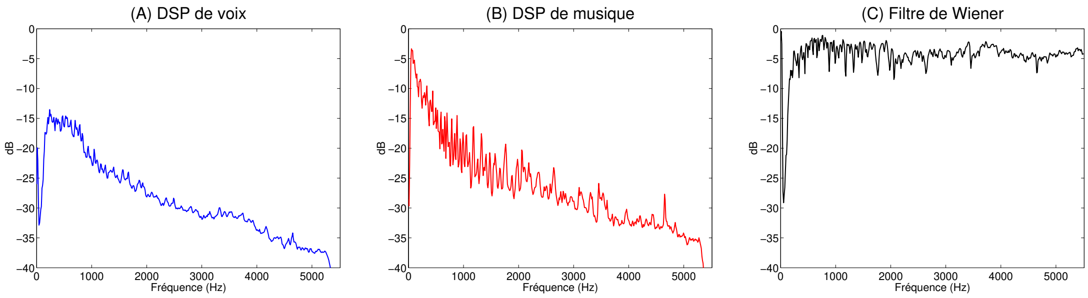


FIG. 4.4 – **(A)** : MMG général de voix à 1 état (c'est-à-dire la DSP de voix). **(B)** : MMG général de musique (la DSP de musique). **(C)** : Filtre de Wiener pour l'estimation de voix.

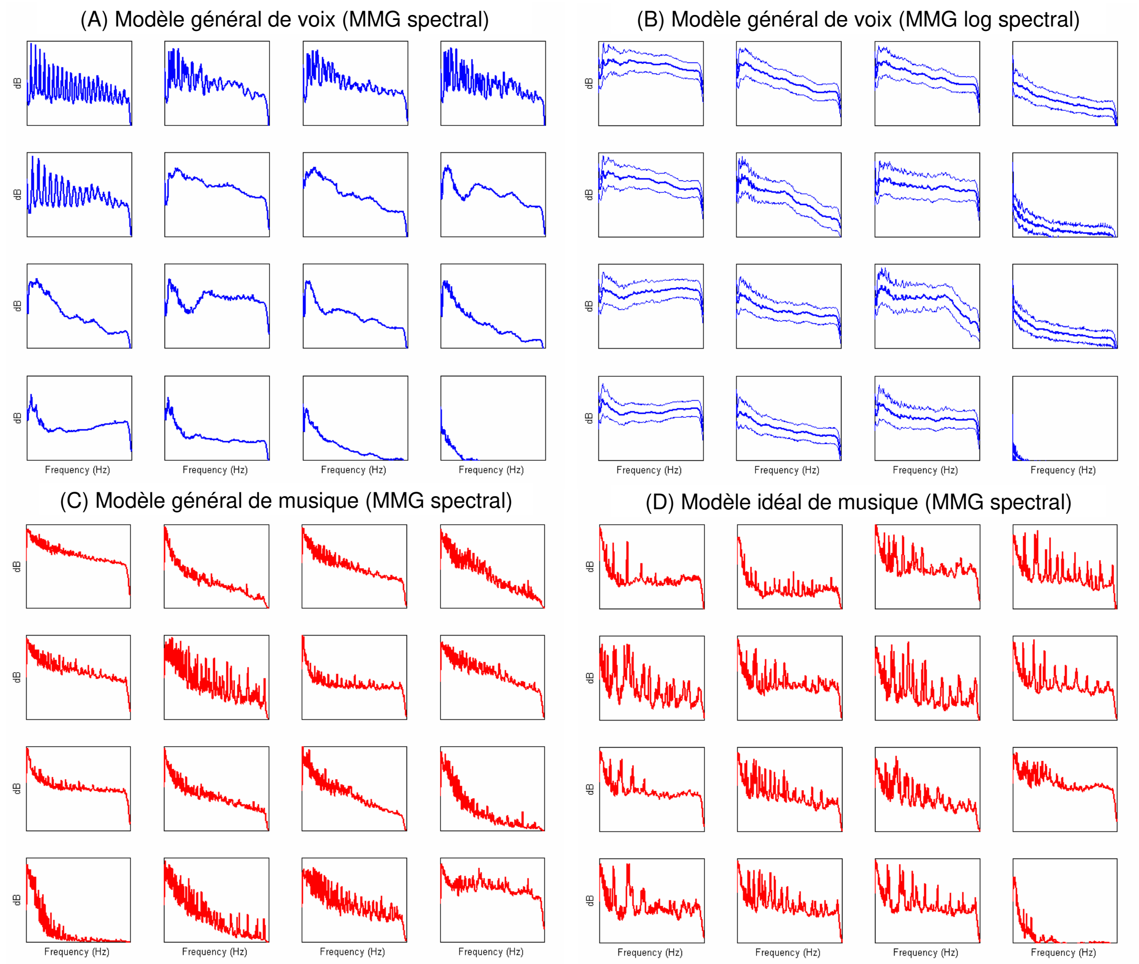


FIG. 4.5 – MMG à 16 états. Pour les MMG spectraux (2.15) chaque état i est représenté par sa DSP : $\log r_i^2(f)$. Pour le MMG log spectral (2.29) chaque état i est représenté par sa DSP : $\mu_i(f)$ et DSP $\pm$ l'écart-type : $\mu_i(f) \pm r_i(f)$. **(A)** : Modèle général de voix (spectral). **(B)** : Modèle général de voix (log spectral). **(C)** : Modèle général de musique (spectral). **(D)** : Modèle idéal de musique (spectral).

3. MMG log spectral / EQM log spectrale (Sec. 2.4.2) [Burshtein-99].

Nous allons comparer ces méthodes en utilisant deux mesures de performance : le RSDN et la DLSN.

Notons que pour un meilleur RSDN, il semble plus approprié d'utiliser des MMG spectraux et de minimiser l'EQM spectrale. En effet, les mesures RSB (3.1) et RSD (3.3) sont toutes les deux des fonctions de l'EQM dans le domaine temporel. La seule différence qualitative entre ces deux mesures est l'invariance aux gains multiplicatifs des sources pour le RSD. Par conséquent, pour avoir un meilleur RSDN, il faut minimiser l'EQM temporelle. Ainsi, sachant que la TFCT est une transformée linéaire, il semble plus judicieux de modéliser les spectres et de minimiser l'EQM spectrale, plutôt que de faire la même chose dans le domaine log spectral.

De même, pour une meilleure DLSN, il semble plus approprié d'utiliser des MMG log spectraux et de minimiser l'EQM log spectrale. Cela peut être expliqué par un raisonnement similaire à celui donné pour le RSDN dans paragraphe précédent.

Ainsi, nous nous attendons *a priori* à ce qu'en basculant progressivement du domaine spectral au domaine log spectral, le RSDN se dégrade et la DLSN s'améliore.

Nous avons comparé les trois méthodes annoncées avec des modèles idéaux λ_v^{Idl} et λ_m^{Idl} à 32 états chacun ($Q_v = Q_m = 32$). Les algorithmes 2 et 3 ont été utilisés pour l'apprentissage des MMG spectraux et des MMG log spectraux. Pour chaque méthode, les deux mesures de performance (le RSDN et la DLSN) sont calculées. De plus, pour la troisième méthode (MMG log spectral / EQM log spectrale) les différentes approximations (MIXMAX (2.36), VTS (2.37) et MeanMAX (2.38)) sont testées.

Les résultats, accompagnés par des références de performance obtenues avec des oracles approchés, sont résumés dans le tableau 4.1.

Comparons d'abord les trois approximations pour la méthode MMG log spectral / EQM log spectrale. Notons que les DLSN sont comparables pour les approximations MIXMAX et VTS. Cependant, le RSDN est bien meilleur pour l'approximations MIXMAX par rapport à VTS. L'approximation MeanMAX donne les plus mauvaises performances pour les deux mesures. Ces résultats rejoignent notre supposition sur les précisions de ces approximations (voir Sec. 2.4.2.2). L'approximation MIXMAX est la plus précise, ensuite c'est VTS, et enfin MeanMAX.

Ensuite, comparons les performances des trois méthodes de séparation étudiées. Comme on s'y attendait, en passant progressivement du domaine spectral dans le domaine log spectral, le RSDN se dégrade. Par contre, la DLSN ne s'améliore pas de façon monotone. En effet, la DLSN est plus mauvaise pour la deuxième méthode (MMG spectral / EQM log spectrale) que pour la première (MMG spectral / EQM spectrale).

Comme il est déjà remarqué dans la section 3.2.1, l'*estimation de la TFCT* $\widehat{S}_k$ ne coïncide pas en général avec la *TFCT de l'estimation* $\widehat{\mathbb{S}}_k = \text{TFCT}[\text{TFCT}^{-1}(\widehat{S}_k)]$ $(\widehat{S}_k(t,f) \neq \widehat{\mathbb{S}}_k(t,f))$, puisque la TFCT est une transformée redondante. Nous avons rajouté dans le tableau 4.1 les valeurs de la mesure DLSN', calculée en remplaçant la TFCT d'estimation $\widehat{\mathbb{S}}_k(t,f)$ par l'estimation de la TFCT $\widehat{S}_k(t,f)$ dans l'équation (3.2). Cette mesure est ajoutée à titre informatif, puisqu'elle ne peut pas être calculée quand la séparation est terminée, car $\widehat{S}_k(t,f)$ n'est plus accessible (voir Fig. 2.8). Remarquons que la mesure DLSN' peut avoir du sens dans le cas où l'on n'est pas intéressé par la reconstruction du signal dans le domaine temporel et que l'on utilise directement l'estimation de la TFCT $\widehat{S}_k(t,f)$. Par exemple, pour la reconnaissance automatique de la parole il est possible de calculer les coefficients cepstraux [Vergin-99] directement à partir de $\widehat{S}_v(t,f) = \widehat{S}_1(t,f)$.

Pour la DLSN', cette amélioration monotone est vérifiée, vraisemblablement parce que la DLSN' est plus cohérente avec le critère d'EQM log spectrale que la DLSN. Le meilleur RSDN est toujours obtenu pour la première méthode (MMG spectral / EQM spectrale) et la meilleure DLSN pour la troisième (MMG log spec. / EQM log spec. / approx. VTS).

Dans le contexte de ce travail, nous avons choisi d'utiliser par la suite la méthode MMG spectral / EQM spectrale pour les raisons suivantes :

1. Cette méthode donne le meilleur RSDN.
2. Elle donne également des performances satisfaisantes en termes de DLSN.
3. Parmi les trois méthodes étudiées, cette méthode a la moindre complexité calculatoire.
4. Pour les MMG spectraux, l'estimateur minimisant l'EQM spectrale peut être calculé de façon exacte, par opposition aux MMG log spectraux pour lesquels il est nécessaire d'avoir recours à un calcul approché.

MMG	**EQM**	Approximation	RSDN	DLSN	DLSN'
spectral (2.15)	spectrale (2.19)	-	**16.0** (23.0)	16.0 (25.5)	*13.9*
spectral (2.15)	log spec. (2.25)	-	15.3 (23.6)	13.3 (25.6)	*14.4*
log spec. (2.29)	log spec. (2.25)	MIXMAX (2.36)	14.0 (23.6)	17.9 (25.6)	*20.3*
		VTS (2.37)	12.2 (23.6)	**18.1** (25.6)	*20.0*
		MeanMAX (2.38)	5.9 (23.6)	-1.3 (25.6)	*-1.1*

Tab. 4.1 – Performances des méthodes en fonction du modèle MMG, de l'EQM minimisée et de l'approximation pour la méthode MMG log spec. / EQM log spec. (estimateurs **doux**, voir par ex. (2.20)). Les références de performance obtenues à l'aide des oracles approchés sont indiquées dans les parenthèses. La DLSN' est ajoutée à titre informatif.

4.5.4 Précision des estimateurs durs par rapport aux estimateurs doux

Cette expérience a pour but de tester l'efficacité des estimateurs durs par rapport aux estimateurs doux (Sec. 2.4.1.3). Autrement dit, nous vérifions de combien sont altérées les performances

MMG	**EQM**	Approximation	RSDN	DLSN	DLSN'
spectral (2.15)	spectrale (2.19)	-	**15.9** (23.0)	15.7 (25.5)	*12.9*
spectral (2.15)	log spec. (2.25)	-	15.2 (23.6)	13.2 (25.6)	*14.4*
log spec. (2.29)	log spec. (2.25)	MIXMAX (2.36)	14.0 (23.6)	17.0 (25.6)	*19.7*
		VTS (2.37)	12.2 (23.6)	**17.5** (25.6)	*19.5*
		MeanMAX (2.38)	5.9 (23.6)	-1.3 (25.6)	*-1.1*

Tab. 4.2 – Performances des méthodes en fonction du modèle MMG, de l'EQM minimisée et de l'approximation pour la méthode MMG log spec. / EQM log spec. (estimateurs **durs**, voir par ex. (2.23)).

de séparation si, au lieu de calculer la somme sur toutes les paires d'états (voir par ex. (2.20)), seule la paire d'états la plus probable est utilisée (voir par ex. (2.23)).

Les mêmes simulations que dans la section précédente sont réalisées, mais en utilisant les estimateurs durs. Les résultats sont résumés dans le tableau 4.2. En les comparant avec ceux du tableau 4.1 on observe que les performances sont dégradées d'au plus 1 dB.

Ainsi, comme il a été déjà remarqué dans la section 2.4.1.3, la complexité calculatoire des algorithmes de séparation peut être diminuée significativement si l'on trouve une procédure rapide (probablement approchée) de recherche de la paire d'états la plus probable permettant d'éviter la recherche exhaustive (2.24).

4.6 Conclusion

Le chapitre a présenté quelques expérimentations préliminaires dans le cadre de la séparation voix / musique. Ces expérimentations ont permis de fixer certains paramètres et de préciser la méthode de séparation que nous allons utiliser dans la suite de ce travail. Nous résumons les choix faits :

- fenêtre d'analyse de la TFCT : Hamming de taille 93 ms,
- méthode de séparation : MMG spectral (2.15) / EQM spectrale (2.19).

De plus, nous avons observé qu'avec des modèles généraux, l'augmentation de taille des modèles n'améliore presque pas les performances moyennes. C'est un véritable problème. En effet, cela signifie que la technique de séparation élaborée avec des MMG multigaussiens ne marche pas mieux qu'un filtrage linéaire simple (MMG monogaussiens). Vraisemblablement, cette limitation est liée à la difficulté de construction de modèles généraux pertinents pour des classes sonores très riches, comme la musique par exemple. Cependant, l'utilisation de modèles idéaux, irréalistes en pratique, donne des performances bien meilleures par rapport aux modèles généraux et ces performances s'améliorent en augmentant les tailles de modèles.

Les observations expérimentales faites dans ce chapitre serviront pour préciser la problématique liée à l'utilisation de modèles généraux qui sera présentée et discutée en détails dans le chapitre suivant.

Chapitre 5

Problématique

Dans le chapitre 2, nous avons présenté un état de l'art sur les méthodes de séparation basées sur des modèles probabilistes *a priori*, notamment des MMG. Ces méthodes semblent assez prometteuses. Cependant, comme il a été déjà mentionné (Sec. 2.2.6), elles souffrent de limitations importantes. Premièrement, pour des classes sonores de grande variabilité, il est difficile en pratique d'accumuler une quantité suffisante de données d'entraînement représentatives. Deuxièmement, la modélisation de telles classes sonores nécessite des modèles de grande taille, et comme conséquence des ressources calculatoires importantes.

Ainsi, après avoir présenté dans le chapitre 3 certains outils d'évaluation et de diagnostic, nous sommes passés dans le chapitre 4 aux expérimentation préliminaires dans le cadre de la séparation voix / musique. Ces expérimentations nous montrent qu'en utilisant des modèles généraux (ou *a priori*), on n'arrive pas à améliorer sensiblement les performances de séparation en augmentant les tailles des modèles, ainsi que la complexité calculatoire. Vraisemblablement, ceci est dû aux limitations d'utilisation des modèles généraux qu'on vient de mentionner.

Dans ce chapitre ces limitations seront discutées de manière plus détaillée, afin de définir la problématique qui sera traitée dans la suite de ce travail.

5.1 Limites des modèles probabilistes *a priori*

Notons qu'il existe une petite imperfection dans la terminologie actuelle utilisée pour la séparation de sources avec des modèles *a priori*. En effet, d'une part on appel la *source* le signal S_k qu'on essaye de séparer à partir d'un enregistrement particulier. D'autre part, on appel le modèle de *source* le modèle Λ_k qui modélise l'ensemble des signaux Y_k (par exemple la parole, la voix chantée, etc.) dont les caractéristiques sont censées être proches de celles de la source S_k. Ainsi, Λ_k n'est pas un modèle de la source S_k, mais de l'ensemble des sources. C'est d'ailleurs pour cela que ce modèle est appelé ici *modèle général*. Cette différence est du même ordre que celle entre une variable aléatoire et sa réalisation particulière. Pour souligner cette distinction, nous utilisons ici le terme *classe sonore* (par ex. la voix, la musique, etc.) pour désigner l'ensemble des sources modélisées par un modèle général Λ_k.

Une des particularités des méthodes de séparation basées sur des modèles MMG spectraux est que, pour avoir des résultats de séparation satisfaisants, ces modèles doivent couvrir assez finement les propriétés des classes sonores modélisées. En effet, chaque événement sonore doit être modélisé assez précisément par un spectre typique (DSP). Ainsi, il devient difficile en pratique de bien modéliser des classes sonores de grande variabilité (comme par exemple la musique) pour les raisons suivantes.

Premièrement, il est difficile d'obtenir des bases d'entraînement représentatives pour des classes sonores très vastes. Considérons par exemple la classe de la musique qui est extrêmement vaste. En effet, à la variabilité des différentes combinaisons d'instruments s'ajoute la variabilité des notes et la variabilité des accords pour chaque instrument. On peut facilement imaginer l'apparition dans un extrait musical à séparer d'une forme spectrale (correspondant par exemple à une combinaison particulière de notes) qui n'apparaît jamais dans la base d'entraînement utilisée. Par conséquent, dans un MMG de musique appris sur cette base, il n'y a pas d'état correspondant à cette forme spectrale, et donc cette forme ne peut pas être séparée correctement.

Deuxièmement, même si l'on arrive à obtenir des bases d'entraînement représentatives, elles doivent être de taille considérable pour couvrir toutes les propriétés des classes sonores de grande variabilité que nous considérons ici. Par conséquent, il faut avoir beaucoup de spectres typiques (DSP) pour pouvoir approcher assez finement ces données. Autrement dit, cela nécessite des modèles de grande taille. Par ailleurs, l'utilisation de modèles de grande taille mène à une complexité calculatoire élevée pour la séparation de l'ordre $O(Q_1 Q_2)$ opérations par échantillon du signal (voir par exemple équations (2.20) et (2.21)). Nous rappelons que Q_k est le nombre d'états (de DSP) du modèle Λ_k et représente ainsi sa taille. Les classes sonores traitées dans la littérature sont souvent la parole masculine et féminine [Roweis-01, Pontoppidan-03, Kristjansson-04, Beierholm-04] ou des instruments musicaux particuliers [Benaroya-03a, Vincent-04a]. Même dans le cas des ces classes, qui ne sont pas encore extrêmement vastes, des modèles de grande taille sont utilisés. Par exemple Kristjansson [Kristjansson-04] utilise $Q_1 = Q_2 = 512$ pour la séparation de la parole homme / femme, ce qui donne la complexité de l'ordre $O(512 \cdot 512)$ opérations par échantillon du signal. La classe de la musique que nous traitons dans cette thèse est considérablement plus vaste que la classe de la parole ou la classe d'un instrument musical particulier. Il est donc crucial de proposer des solutions permettant d'utiliser des modèles de taille raisonnable.

Ainsi, les deux problèmes suivants se posent :

1. La difficulté d'obtenir des bases d'entraînement représentatives pour des classes sonores très riches.

2. La complexité calculatoire ingérable due à la nécessité d'utiliser des modèles de grande taille.

Comme nous l'avons observé en faisant les expériences préliminaires, l'augmentation de tailles des modèles généraux ne permet par d'améliorer sensiblement les performances de séparation (Fig. 4.2). Vraisemblablement, cela est lié à ces deux problèmes à la fois. Premièrement, vu la variabilité considérable de la musique, la base d'entraînement de musique que nous utilisons est très probablement peu représentative. Par contre, cela concerne moins la base d'entraînement de voix, car la classe sonore de voix chantée est beaucoup moins variable que celle de musique. Deuxièmement, étant limité en pratique par les ressources calculatoires, on ne peut pas augmenter considérablement les tailles des modèles utilisés, notamment, parce que la complexité calculatoire de séparation est de l'ordre du produit de ces tailles ($O(Q_1 Q_2)$). L'utilisation de modèles de taille raisonnable mène à une modélisation trop grossière (voir par ex. le MMG de musique à 16 états représenté sur la figure 4.5 (C)).

Il est utile de comprendre comment ces problèmes s'expriment en termes de la modélisation statistique utilisée. Supposons que la source S_k est une réalisation d'un processus aléatoire $\mathcal{S}_k$, et que l'ensemble d'entraînement Y_k est une réalisation d'un autre processus aléatoire $\mathcal{Y}_k$. Supposons également que ces processus aléatoires possèdent des densités $p_{\mathcal{S}_k}(\cdot)$ et $p_{\mathcal{Y}_k}(\cdot)$. Pour bien estimer les sources $S_k, k = 1, 2$ (Sec. 2.3.3) l'idéal serait de connaître leurs vraies densités $p_{\mathcal{S}_k}(\cdot)$. Comme les sources S_k ne sont pas observées directement, leurs densités sont difficiles à estimer, et elles sont remplacées par les densités des ensembles d'entraînement $p_{\mathcal{Y}_k}(\cdot)$. Ces dernières sont enfin approchées par les densités des MMG $p(\cdot|\Lambda_k)$ dont les paramètres sont estimés en utilisant l'algorithme 2 présenté dans la section 2.4.1.1. Ainsi, les densités $p_{\mathcal{S}_k}(\cdot)$ sont remplacées par $p(\cdot|\Lambda_k)$ en deux étapes :

$$p_{\mathcal{S}_k}(\cdot) \to p_{\mathcal{Y}_k}(\cdot) \approx p(\cdot|\Lambda_k) \tag{5.1}$$

- Le remplacement $p_{\mathcal{S}_k}(\cdot) \to p_{\mathcal{Y}_k}(\cdot)$ a une influence négative quand la base d'entraînement Y_k ne contient pas d'exemples dont les caractéristiques ressemblent à celles de la source S_k. Cela est donc lié au problème de construction de bases d'entraînement représentatives.
- L'approximation $p_{\mathcal{Y}_k}(\cdot) \approx p(\cdot|\Lambda_k)$ est liée au problème des modèles de grande taille, car il faut que le modèle Λ_k soit suffisamment riche pour pouvoir bien approcher la densité $p_{\mathcal{Y}_k}(\cdot)$.

5.2 L'adaptation comme solution

Pour pouvoir dépasser ces limitations, nous proposons de recourir, quand c'est possible, à un schéma d'*adaptation* qui vise à ajuster *a posteriori* certaines caractéristiques des modèles *a priori* à celles des sources dans le mélange. Autrement dit, le but de cette adaptation est de rapprocher dans la mesure du possible les densités modélisées par les MMG généraux des vraies densités des sources $p_{\mathcal{S}_k}(\cdot)$. Comme nous le verrons par la suite, cette approche permet dans certains cas d'améliorer les modèles de sources, ainsi que d'utiliser des modèles de taille raisonnable.

Le reste de ce manuscrit est organisé de la manière suivante. Un formalisme général d'adaptation des modèles pour la séparation de sources avec un seul capteur sera introduit dans la partie II. Ensuite, un système de séparation voix / musique basé sur ce formalisme d'adaptation sera développé dans la partie III et évalué dans la partie IV. Enfin, la conclusion générale, ainsi que quelques perspectives de ce travail, seront présentées dans la partie V.

Deuxième partie

Adaptation des modèles : développement d'un formalisme général

Chapitre 6

Formalisme d'adaptation

Dans le chapitre 5, nous avons présenté et discuté en détails les limitations importantes d'utilisation des modèles statistiques pour la séparation de sources avec un seul capteur. Ces limitations sont une complexité calculatoire ingérable et la difficulté d'accumuler des données d'entraînement représentatives quand il s'agit de modéliser des classes sonores de grande variabilité. Pour pouvoir, dans certains cas, dépasser ces limitations, nous proposons dans cette thèse d'adapter *a posteriori* des modèles généraux, en rapprochant certaines de leurs caractéristiques de celles des sources dans le mélange.

Nous développons dans ce chapitre un formalisme d'adaptation des modèles basé sur un critère d'adaptation bayésienne Maximum *A Posteriori* (MAP). Ce formalisme étant introduit ici de manière très générale, il sera appliqué pour la séparation voix / musique dans la partie III.

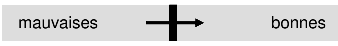
6.1 Cahier des charges pour l'adaptation

Nous commençons doucement, en définissant une sorte de "cahier des charges" pour l'adaptation des modèles, dont le but est d'expliquer ce que nous attendons de cette adaptation. Ce cahier des charges est représenté sous la forme du tableau 6.1.

D'une part, les modèles généraux Λ_1 et Λ_2 sont appris à partir des données d'entraînement Y_1 et Y_2. Leur utilisation est tout à fait réaliste, mais ils donnent des performances médiocres, en tous cas dans le cas de notre tâche de séparation.

D'autre part, les modèles idéaux λ_1^{Idl} et λ_2^{Idl} sont appris à partir des sources séparées S_1 et S_2. Ces modèles ont été déjà testés section 4.5.2 et ils permettent d'obtenir des performances de séparation bien meilleures que celles obtenues avec des modèles généraux (Fig. 4.2). Par contre, l'utilisation de ces modèles n'est pas réaliste, car les sources séparées ne sont pas observées, c'est justement elles que l'on cherche à estimer.

Ainsi, nous souhaitons introduire de nouveaux modèles, les *modèles adaptés* λ_1 et λ_2, qui

Modèles	Obtenus à partir de	Système	Performances
généraux Λ_1 et Λ_2	données d'éntraînement Y_1, Y_2	réaliste	
adaptés λ_1 et λ_2	modèles généraux Λ_1, Λ_2 et le mélange X	réaliste	
idéaux λ_1^{Idl} et λ_2^{Idl}	sources séparées S_1, S_2	irréaliste	

TAB. 6.1 – Cahier des charges pour l'adaptation des modèles.

sont intermédiaires entre les modèles généraux et les modèles idéaux. Premièrement, nous exigeons que, comme pour les modèles généraux, l'utilisation des modèles adaptés soit réaliste. Deuxièmement, nous attendons que les modèles adaptés donnent de meilleures performances par rapport aux modèles généraux et que ces performances s'approchent dans la mesure du possible de celles des modèles idéaux.

Puisque l'utilisation des modèles adaptés doit être réaliste, les sources séparées S_1 et S_2 ne peuvent pas être utilisées pour leur construction. Ainsi, les modèles adaptés doivent être obtenus à partir de toutes les autres connaissances disponibles dans un contexte réaliste, c'est-à-dire à partir des modèles généraux Λ_1, Λ_2 et du mélange X. On pourrait aussi imaginer réutiliser les données d'entraînement Y_1 et Y_2, mais cela n'a pas été l'approche adoptée dans ce travail.

6.2 Formalisme d'adaptation basé sur le critère MAP

Dans cette section, nous présentons de manière assez générale notre proposition concernant l'adaptation des modèles sous la forme d'un critère d'adaptation MAP.

Nous introduisons dans cette thèse des *modèles adaptés* dont les caractéristiques, par opposition aux modèles généraux Λ_1 et Λ_2, sont dans la mesure du possible rapprochées de celles des sources à séparer S_1 et S_2. Bien que les modèles adaptés aient exactement la même structure que les modèles généraux, afin de bien distinguer ces deux types de modèles, de nouvelles notations sont utilisées pour les modèles adaptés, ainsi que pour leurs paramètres. Les modèles adaptés sont donc notés λ_k, $k = 1, 2$ et paramétrisés comme $\lambda_k = \{\omega_{k,i}, \Sigma_{k,i}\}_i$, où $\omega_{k,i}$ sont des poids de gaussiennes et $\Sigma_{k,i} = \mathrm{diag}[\sigma_{k,i}^2(f)]_f$ sont des matrices de covariance.

Comment pourra-t-on procéder pour adapter les modèles ? L'idéal serait de les apprendre directement sur les sources S_k, comme les modèles idéaux λ_k^{Idl}, en maximisant la vraisemblance

$p(S_k|\lambda_k)$. Par exemple, Benaroya [Benaroya-03a] évalue ses algorithmes de séparation dans un contexte similaire. Il apprend les modèles sur les sources séparées (accessibles dans le cadre expérimental) venant de la première partie d'un extrait musical et il sépare ensuite la deuxième partie du même extrait. Cela donne de bonnes performances, mais malheureusement une telle approche n'est possible que dans un contexte expérimental. Dans le contexte d'une application réelle, les sources S_k ne sont pas observées directement, mais via le mélange X.

Une autre piste est d'essayer d'inférer les paramètres des modèles directement à partir du mélange X. Par exemple Attias [Attias-03] utilise une telle approche dans le cas multicapteur, où il y a au moins autant de capteurs que de sources. Dans ce cas, la diversité spatiale (le fait que les sources arrivent de directions différentes) crée une structure rigide permettant d'estimer correctement les modèles sans aucune autre connaissance *a priori*. Dans le cas d'un seul capteur traité ici la diversité spatiale n'est pas exploitable, ainsi cette approche n'est pas applicable directement. En effet, on peut par exemple essayer de chercher des modèles λ_1 et λ_2 en optimisant le critère du MV suivant :

$$(\lambda_1^{\mathrm{MV}}, \lambda_2^{\mathrm{MV}}) = \arg \max_{(\lambda'_1, \lambda'_2)} p(X|\lambda'_1, \lambda'_2) \tag{6.1}$$

Toutefois, cela ne mène pas à une bonne estimation des modèles, car dans ce critère il n'y a plus aucune connaissance *a priori* sur la nature des sources, et il peut exister une infinité de solutions non satisfaisantes.

Ainsi, nous proposons de garder les modèles généraux Λ_1 et Λ_2 comme des connaissances *a priori* et de les adapter (ou déformer certaines de leurs caractéristiques) par rapport au mélange X en utilisant les techniques d'adaptation bayésienne au Maximum *A Posteriori* (MAP) [Gauvain-94]. Ces techniques d'adaptation ont été récemment appliquées avec succès pour la reconnaissance de la parole [Lee-Huo-00] et la vérification du locuteur [Reynolds-00]. Le critère d'estimation MAP consiste à maximiser la densité *a posteriori* $p(\lambda_1, \lambda_2|X)$ plutôt que la vraisemblance $p(X|\lambda_1, \lambda_2)$ (6.1). En utilisant la loi de Bayes on peut montrer que cette densité *a posteriori* se décompose comme :

$$p(\lambda_1, \lambda_2|X) = \frac{p(X|\lambda_1, \lambda_2)p(\lambda_1, \lambda_2)}{p(X)} \propto p(X|\lambda_1, \lambda_2)p(\lambda_1, \lambda_2) \tag{6.2}$$

avec le facteur de proportion $1/p(X)$ qui ne dépend pas des modèles λ_k, $k = 1, 2$ et n'a donc pas d'influence sur l'optimisation du critère. Dans un critère MAP, les paramètres des modèles sont considérés comme des réalisations des variables aléatoires et il faut définir leurs densités *a priori* $p(\lambda_1, \lambda_2)$. Nous supposons que les paramètres du modèle λ_1 sont indépendants de ceux du modèle λ_2 et que la densité des paramètres de chaque modèle est définie en fonction des paramètres du modèle général correspondant Λ_k. Autrement dit, nous posons $p(\lambda_1, \lambda_2) \triangleq p(\lambda_1|\Lambda_1)p(\lambda_2|\Lambda_2)$ et nous avons le critère MAP suivant :

$$(\lambda_1^{\mathrm{MAP}}, \lambda_2^{\mathrm{MAP}}) = \arg \max_{(\lambda_1', \lambda_2')} p(X|\lambda_1', \lambda_2') p(\lambda_1'|\Lambda_1) p(\lambda_2'|\Lambda_2) \tag{6.3}$$

Remarquons que la différence de ce critère MAP par rapport au critère du MV (6.1) est qu'on impose des contraintes sur les modèles adaptés qui sont définies par des lois *a priori* dépendant des modèles généraux Λ_1 et Λ_2. Ainsi, les modèles généraux jouent toujours le rôle de connaissances *a priori* sur la nature des sources. Si les lois *a priori* sont choisies de manière judicieuse (nous abordons la question du choix de ces lois un peu plus tard), on espère qu'avec le critère MAP (6.3) il est possible d'obtenir des modèles adaptés qui donnent de meilleures performances de séparation par rapport à l'utilisation de modèles généraux. La figure 6.1 illustre comment le module d'adaptation *a posteriori* des modèles à l'aide du critère MAP (6.3) s'intègre dans le schéma de séparation représenté sur la figure 2.7. Puisque nous avons choisi d'utiliser la méthode MMG spectral / EQM spectrale (Sec. 4.5.3), par rapport au schéma représenté figure 2.7, la figure 6.1 utilise des transformations $\mathcal{F}$, $\mathcal{L}$ et $\mathcal{D}$ particulières, notamment $\mathcal{F} = \mathrm{TFCT}$, $\mathcal{L} = \mathrm{Id}$ et $\mathcal{D} = \mathrm{Id}$.

Nous appelons l'approche proposée *adaptation des modèles avec des données acoustiques manquantes*. Ce titre reflète les deux idées suivantes :

1. *L'adaptation des modèles* correspond à l'attachement des modèles adaptés aux modèles généraux à l'aide des lois *a priori* $p(\lambda_k|\Lambda_k)$ (6.3).
2. L'utilisations des *données acoustiques manquantes* correspond au fait que les paramètres des modèles sont estimés à partir du mélange X et non pas à partir des sources S_k (*données acoustiques*) qui sont inaccessibles (*manquantes*). L'adjectif "acoustiques" a pour but de ne pas introduire de confusion avec des *données manquantes* utilisées dans la terminologie de l'algorithme EM [Dempster-77].

6.2.1 Représentation à l'aide des réseaux bayésiens

Pour mieux expliquer les blocs du schéma représenté figure 6.1 et pour les comparer entre eux, nous les représentons ici à l'aide des réseaux bayésiens (Sec. 2.2.1). Ceci rejoint notre intention d'utiliser le long de ce travail les réseaux bayésiens pour représenter les modèles probabilistes utilisés, ainsi que les algorithmes d'estimation de leurs paramètres, des sources, etc. Les réseaux correspondant aux processus aléatoires impliqués dans les procédures d'apprentissage des modèles, d'estimation des sources et d'adaptation *a posteriori* des modèles (Fig. 6.1) sont représentés sur la figure 6.2. Rappelons que $q_k = [q_k(t)]_t$, $k = 1, 2$ sont des séquences des états cachés des modèles MMG. Une des différences avec la représentation faite figure 2.5 est que chaque séquence temporelle (par ex. $S_k = [S_k(t)]_t$) est représentée par un seul noeud pour

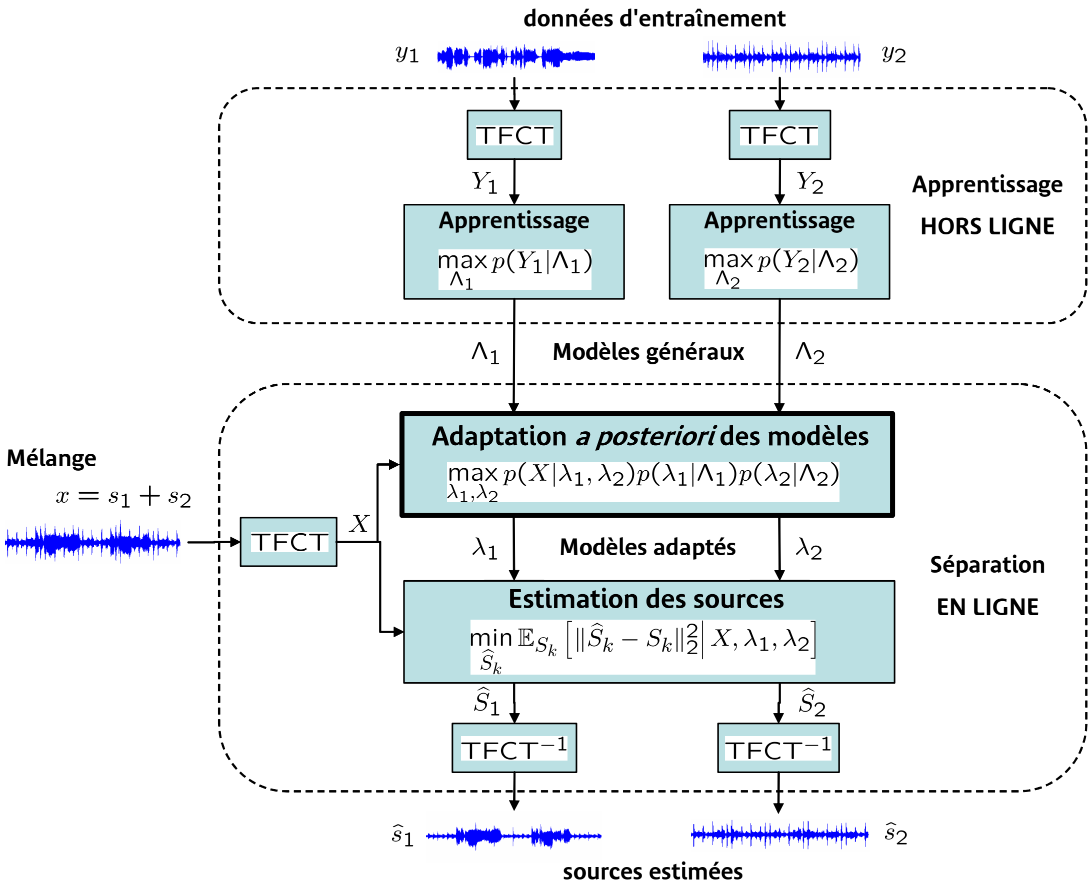


FIG. 6.1 – Séparation avec des modèles adaptés *a posteriori* (par rapport au schéma représenté figure 2.7 ici $\mathcal{F} = \mathrm{TFCT}$, $\mathcal{L} = \mathrm{Id}$ et $\mathcal{D} = \mathrm{Id}$).

des raisons de compacité. De plus, les modèles sont également représentés sur ces réseaux en forme d'hexagone. Enfin, nous utilisons ici une nouvelle coloration, notamment les noeuds cachés dont on souhaite estimer les valeurs ponctuelles conditionnellement aux noeuds observés (noirs) sont colorés en gris. Les autres noeuds cachés restent colorés en blanc. Cette nouvelle coloration permet de représenter sur le réseau bayésien le but de la procédure d'estimation correspondante.

Expliquons un peu plus en détails les réseaux bayésiens représentés figure 6.2 :

- **Apprentissage des modèles :** L'apprentissage du modèle Λ_k est effectué à l'aide du critère du MV (2.12) à partir des données d'entraînement Y_k. Une séquence d'états q_k est générée à partir du modèle Λ_k. Conditionnellement à cette séquence, Y_k est un processus gaussien avec des paramètres définis par le modèle Λ_k, d'où les dépendances représentées figure 6.2 (A). Les données Y_k sont observées et le but est d'estimer les paramètres du modèle Λ_k, d'où leurs colorations respectives en noir et en gris.
- **Estimation des sources :** L'estimation des sources S_1 et S_2 est effectuée en utilisant le critère (2.13) à partir du mélange X et des modèles λ_1 et λ_2. Ainsi, les sources sont colorées en gris et le mélange et les modèles en noir. Les dépendances entre λ_k, q_k et S_k $(k = 1, 2)$ sont exactement les mêmes qu'entre Λ_k, q_k et Y_k pour l'apprentissage (Fig. 6.2 (A)). Enfin, selon l'équation (2.3), le mélange X est obtenu à partir des deux sources S_1 et S_2.
- **Adaptation *a posteriori* des modèles :** Pour l'adaptation des modèles à l'aide du critère MAP (6.3), le réseau bayésien a la même allure que pour l'estimation des sources (Fig. 6.2 (B)), sauf les différences suivantes : au lieu d'estimer les sources S_1 et S_2, le but est d'estimer les modèles adaptés λ_1 et λ_2, ils sont donc colorés en gris. Chaque modèle adapté λ_k est relié avec le modèle général correspondant Λ_k par la loi *a priori* $p(\lambda_k|\Lambda_k)$. Les modèles généraux sont supposés connus (observés) et sont donc colorés en noir.

6.2.2 Rôle des lois *a priori*

Abordons maintenant la question du choix des lois *a priori* définies par les densités $p(\lambda_k|\Lambda_k)$. Il est clair que nous nous retrouvons devant un compromis dans ce choix. D'une part, comme l'adaptation est effectuée à partir du mélange X, ces lois devraient être assez restrictives pour que les modèles adaptés λ_k soient bien attachés aux modèles généraux Λ_k, en maximisant ainsi les chances de converger vers des modèles pertinents. D'autre part, ces lois devraient quand même laisser de la liberté aux modèles pour qu'ils puissent s'adapter aux caractéristiques des sources. Les deux cas suivants sont des cas extrêmes de ce compromis :

1. Les modèles λ_k sont trop attachés aux modèles généraux Λ_k, c'est-à-dire qu'il n'y a plus aucune liberté et ils doivent rester égaux aux modèles généraux. On revient donc au cas sans adaptation des modèles (Fig. 2.7).

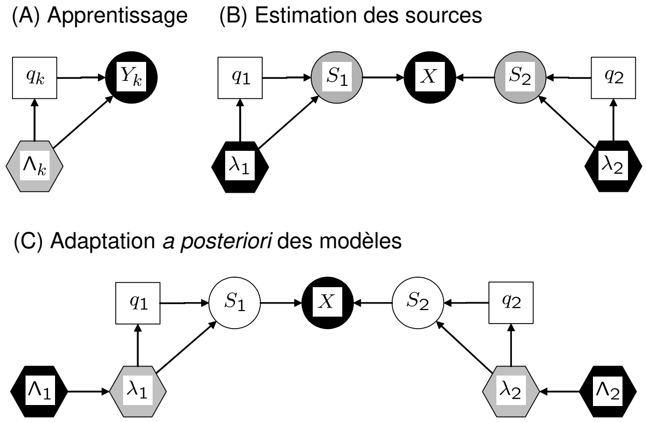


FIG. 6.2 – Réseaux bayésiens correspondants aux processus aléatoires impliqués dans les procédures d'apprentissage des modèles, d'estimation des sources et d'adaptation *a posteriori* des modèles (Fig. 6.1). Formes des noeuds : processus continus (ronds), processus discrets (carrés), modèles (hexagones). Coloration des noeuds : noeuds observés (noir), noeuds cachés estimés (gris), autres noeuds cachés (blanc).

2. Les modèles λ_k ne sont plus du tout attachés aux modèles généraux Λ_k, c'est-à-dire les lois *a priori* sont uniformes non informatives ($p(\lambda_k|\Lambda_k) \propto const$). Ainsi, on se retrouve dans le cas du critère du MV (6.1). Comme cela a déjà été dit, ce critère ne mène pas à une bonne adaptation, car il n'y a plus aucune connaissance *a priori* sur les sources.

L'adaptation peut être considérée aussi comme *la déformation de certaines caractéristiques des modèles généraux sur le mélange X ayant pour but de les rapprocher au mieux des sources.* Au sein d'une telle définition de l'adaptation, les lois *a priori* définissent *quelles caractéristiques* devraient être déformées et *jusqu'à quel point.*

Faire un bon choix des lois *a priori* est donc très important. Malheureusement, il n'y a pas de règles précises indiquant comment choisir, car ce choix dépend de nombreux facteurs différents. Par exemple, si les modèles généraux sont déjà bien représentatifs des sources à séparer, il faut probablement mieux dans ce cas ne pas les adapter du tout. L'adaptation n'est pas une solution miracle qui marche à tous les coups. Le seul moyen de vérifier que les lois *a priori* sont choisies de manière appropriée pour une tâche de séparation particulière est de tester l'adaptation avec ces lois et de montrer qu'elle mène à des meilleures performances de séparation par rapport aux modèles généraux.

Néanmoins nous donnons quelques conseils généraux pour choisir, les lois *a priori* étant vues comme les descriptions des caractéristiques des modèles généraux qui sont déformées (ou adaptées) sur le mélange X :

– Premièrement, il faut mieux choisir peu de caractéristiques à déformer. En effet, comme il est déjà remarqué au début de cette section, en choisissant beaucoup de caractéristiques déformables on perd l'attachement aux connaissances *a priori* (modèles généraux), le critère du MV (6.1) étant le cas extrême.

– Deuxièmement, parmi ces caractéristiques, il vaut mieux choisir celles dont la déformation diminue au plus vite la discordance entre les propriétés des modèles généraux et celles des sources à séparer.
– Enfin, il faut qu'il y ait suffisamment de nouvelles données pour l'adaptation d'une caractéristique particulière. Au cas contraire cela mènera à une suradaptation de cette caractéristique.

Les exemples de lois utilisables potentiellement peuvent être tirés des nombreuses techniques d'adaptation appliquées à la reconnaissance de la parole et à la vérification du locuteur, telles que MAP [Gauvain-94, Reynolds-00], MLLR (*Maximum Likelihood Linear Regression*) [Leggetter-95, Gales-96], SMAP (*Structural MAP*) [Shinoda-97], EMLLR (*Eigenspace-Based MLLR*) [Chen-00], etc. Lee et Huo [Lee-Huo-00] proposent un bon récapitulatif de ces techniques d'adaptation.

Enfin, pour éviter des ambiguïtés, remarquons que la technique d'adaptation MAP [Gauvain-94, Reynolds-00] correspond à un choix particulier de lois *a priori*, ce sont les lois *a priori* conjuguées [Gauvain-94] (la loi Normal - Wishart inverse pour les matrices de covariance et la loi de Dirichlet pour les poids des gaussiennes). Ici nous appelons *adaptation MAP* n'importe quelle procédure qui peut être représentée sous la forme du critère MAP (6.3), quelles que soient les lois *a priori*.

6.2.3 Positionnement par rapport à l'état de l'art

Le problème de la grande variabilité des données d'entraînement soulevé dans la section 5.1 a été déjà abordé dans la littérature. Il existe des travaux qui proposent d'introduire dans les modèles de source des invariances par rapport à certaines caractéristiques physiques, en diminuant ainsi l'influence de la variabilité des données d'entraînement. Par exemple la proposition de Benaroya *et al.* [Benaroya-06] mentionnée section 2.2.4 concerne l'utilisation de facteurs de gains variant au cours du temps, en introduisant ainsi une invariance par rapport à l'énergie locale du signal. Pour la séparation des instruments musicaux, Vincent [Vincent-04a] propose d'utiliser en plus d'autres paramètres descriptifs représentant le volume, la hauteur et le timbre de la note musicale correspondant à un état du modèle. Ces paramètres supplémentaires sont estimés *a posteriori* pour chaque trame, car ils varient au cours du temps. Ainsi, ces approches peuvent être vues comme des adaptations locales. Par rapport à ces travaux, notre formalisme d'adaptation se différencie de la manière suivante. Dans ces propositions d'adaptation locale, l'introduction des paramètres supplémentaires modifie les structures des modèles, tandis que dans notre proposition d'adaptation globale, les modèles gardent la même structure, le nombre des paramètres des modèles n'est pas modifié non plus, et ce sont seulement les valeurs des paramètres qui sont modifiées.

Rappelons que notre approche appelée "adaptation des modèles avec des données acoustiques manquantes" réunit deux aspects à la fois : l'adaptation et l'estimation des modèles à partir du mélange X. L'aspect "adaptation" est inspiré par des techniques d'adaptation utilisées pour la reconnaissance de la parole et la vérification du locuteur [Gauvain-94, Leggetter-95, Gales-96, Shinoda-97, Reynolds-00, Chen-00, Lee-Huo-00]. L'aspect "estimation des modèles à partir du mélange" est inspiré par des travaux sur l'identification du locuteur en présence du bruit [Rose-94] et sur le groupement aveugle des chansons populaires [Tsai-04]. Dans ces travaux, un modèle est estimé à partir du mélange, avec l'autre modèle fixé *a priori*, mais il n'y a pas de notion d'adaptation, c'est-à-dire qu'il n'y a pas d'attachement aux modèles généraux.

Ainsi, les deux contributions principales de notre proposition sont :

1. Regroupement des aspects de l'adaptation et de l'estimation des modèles à partir du mélange dans un même formalisme.
2. Application de ce formalisme pour la séparation de sources avec un seul capteur.

6.3 Conclusion

La séparation de sources par approche statistique est limitée par les problèmes de la complexité excessive des modèles et du manque de données d'entraînement représentatives.

Pour y remédier, nous avons proposé dans ce chapitre une approche originale d'adaptation *a posteriori* des modèles, qui est susceptible de pouvoir dépasser ces limitations. Inspirée par des stratégies utilisées en reconnaissance de la parole, cette approche est formulée comme un critère d'adaptation bayésienne Maximum *A Posteriori* (MAP). L'idée de cette adaptation est de déformer certaines caractéristiques des modèles généraux pour les adapter aux propriétés des sources dans le mélange. Ces caractéristiques sont définies par des loi *a priori* représentant le degré d'attachement des modèles adaptés aux modèles généraux. Le rôle de ces lois a été discuté en détail.

L'approche proposée semble être capable de dépasser des limitations des modèles généraux, ce qui sera étayé par une validation expérimentale (partie IV). Au préalable, nous allons nous focaliser sur l'algorithme d'adaptation proprement dit.

Chapitre 7

Algorithme d'adaptation

Dans ce chapitre, nous développons un algorithme d'adaptation des modèles, qui est basé sur l'algorithme EM [Dempster-77] explicité pour l'optimisation du critère MAP (6.3).

Comme il a été remarqué dans la section 6.2.3, notre approche sur l'adaptation réunit à la fois les deux aspects suivants : "adaptation" et "estimation des modèles à partir du mélange". Ainsi, pour le développement de l'algorithme d'adaptation que nous allons présenter, nous nous sommes inspirés des deux articles suivants :

- [Gauvain-94] traitant l'aspect "adaptation",
- [Rose-94] traitant l'aspect "estimation des modèles à partir du mélange".

Chacun de ces deux travaux traite un des deux aspects indépendamment de l'autre et les deux utilisent l'algorithme EM. Ainsi, l'originalité de l'algorithme qui sera développé dans ce chapitre est son caractère de généralisation réunissant les deux algorithmes présentés dans [Gauvain-94] et [Rose-94].

Par ailleurs, nous avons choisi d'utiliser une forme particulière que prend l'algorithme EM dans le cas des familles exponentielles (Déf. 3, Annexe A.2), dont les MMG font partie.

Ainsi, l'algorithme d'adaptation sera présenté en trois étapes, qui correspondent à différents niveaux de généralité. D'abord il sera présenté sous sa forme générale, ensuite des précisions seront apportées pour le cas des familles exponentielles, et enfin il sera explicité pour les MMG spectraux.

7.1 Algorithme d'adaptation sous sa forme générale

L'algorithme EM pour l'optimisation d'un critère MAP est présenté dans l'annexe A.3. Pour pouvoir appliquer cet algorithme dans le cas du critère MAP (6.3) on a besoin de spécifier les données observées $\mathcal{X}$, les données complètes $\mathcal{Z}$, les paramètres estimés θ et la densité *a priori* sur les paramètres $p(\theta)$. Nous choisissons :

– le mélange X comme données observées ($\mathcal{X} \triangleq X$),
– les suites des états cachés q_1 et q_2 et les sources S_1 et S_2 comme données complètes ($\mathcal{Z} \triangleq \{q_1, S_1, q_2, S_2\}$),
– les modèles adaptés λ_1 et λ_2 comme paramètres estimés ($\theta \triangleq \{\lambda_1, \lambda_2\}$),
– le produit des densités *a priori* sur les modèles $p(\lambda_1|\Lambda_1)$ et $p(\lambda_2|\Lambda_2)$ comme densité *a priori* sur les paramètres ($p(\theta) \triangleq p(\lambda_1|\Lambda_1)p(\lambda_2|\Lambda_2)$).

Remarquons que les données observées et les données complètes sont choisies de façon à ce que l'on puisse utiliser l'algorithme EM. En effet, selon l'équation de mélange (2.3) les données observées $\mathcal{X}$ s'expriment de façon unique à partir des données complètes $\mathcal{Z}$, notamment $X = S_1 + S_2$, ce qui détermine la transformée Ω reliant les données complètes avec les données observées ($\mathcal{X} = \Omega(\mathcal{Z})$, voir Annexe A.3).

La vraisemblance des données observées et celle des données complètes s'écrivent comme :
– $p(\mathcal{X}|\theta) = p(X|\lambda_1, \lambda_2)$, la vraisemblance des données observées,
– $p(\mathcal{Z}|\theta) = p(S_1, q_1|\lambda_1)p(S_2, q_2|\lambda_2)$, la vraisemblance des données complètes

Avec ces nouvelles notations, le critère MAP (6.3) s'exprime sous la forme du critère (A.4). On peut montrer que dans le cas particulier du critère MAP (6.3), l'algorithme EM (A.6), (A.7) s'écrit comme suit :

$$\begin{aligned}
\textbf{E :} \quad Q_k(\lambda_k, \lambda_k^{(l)}) &= \mathbb{E}_{S_k, q_k}\left[\log p(S_k, q_k|\lambda_k) \,\middle|\, X, \lambda_1^{(l)}, \lambda_2^{(l)}\right] + \log p(\lambda_k|\Lambda_k), \quad k = 1, 2 && (7.1)\\
\textbf{M :} \quad \lambda_k^{(l+1)} &= \arg\max_{\lambda_k} Q_k(\lambda_k, \lambda_k^{(l)}), \quad k = 1, 2 && (7.2)
\end{aligned}$$

où $\lambda_1^{(l)}$ et $\lambda_2^{(l)}$ désignent les paramètres des modèles estimés à la l-ème itération.

Remarquons que, grâce à l'indépendance conditionnelle des données complètes ($p(\mathcal{Z}|\theta) = p(S_1, q_1|\lambda_1)p(S_2, q_2|\lambda_2)$) et des modèles ($p(\theta) = p(\lambda_1|\Lambda_1)p(\lambda_2|\Lambda_2)$) à chaque itération, les étapes E et M évoluent indépendamment pour chaque source $k = 1, 2$. Parallèlement, il y a toujours des interactions entre les estimations des modèles $\lambda_1^{(l)}$ et $\lambda_2^{(l)}$, car ces deux estimations interviennent pour le calcul de l'espérance conditionnelle à l'étape E (7.1).

7.2 Algorithme d'adaptation pour les familles exponentielles

Maintenant, supposons que les familles des densités $\{p(S_k, q_k|\lambda_k)\}_{\lambda_k}$, $k = 1, 2$ sont des familles exponentielles (Déf. 3, Annexe A.2) et que $\mathbf{T}_k(S_k, q_k)$ sont des statistiques naturelles (Déf. 3, Annexe A.2) correspondant à ces familles.

En utilisant la définition des familles exponentielles (A.5) et le fait que la vraisemblance des données complètes $p(\mathcal{Z}|\theta)$ se factorise comme $p(\mathcal{Z}|\theta) = p(S_1, q_1|\lambda_1)p(S_2, q_2|\lambda_2)$, on peut déduire que $\{p(\mathcal{Z}|\theta)\}_\theta$ est aussi une famille exponentielle avec la statistique naturelle $\mathbf{T}(\mathcal{Z}) = \{\mathbf{T}_1(S_1, q_1), \mathbf{T}_2(S_2, q_2)\}$. Ainsi, on peut utiliser pour l'optimisation du critère MAP (6.3) l'algorithme EM (A.8), (A.9) présenté annexe A.3.1, qui prend la forme suivante :

$$\mathbf{E}: \quad \mathbf{T}_k^{(l)}(S_k, q_k) = \mathbb{E}_{S_k, q_k}\left[\mathbf{T}_k(S_k, q_k) \,\middle|\, X, \lambda_1^{(l)}, \lambda_2^{(l)}\right], \quad k = 1, 2 \tag{7.3}$$

$$\mathbf{M}: \quad \lambda_k^{(l+1)} = \mathbf{f}_k\left(\mathbf{T}_k^{(l)}(S_k, q_k)\right), \quad k = 1, 2 \tag{7.4}$$

où les fonctions $\mathbf{f}_k\left(\mathbf{T}_k(S_k, q_k)\right)$, $k = 1, 2$ sont définies comme des solutions des critères MAP des données complètes :

$$\mathbf{f}_k\left(\mathbf{T}_k(S_k, q_k)\right) \triangleq \arg\max_{\lambda'_k} p(S_k, q_k|\lambda'_k) p(\lambda'_k|\Lambda_k), \quad k = 1, 2 \tag{7.5}$$

L'existence de telles fonctions, qui ne dépendent que des statistiques naturelles (suffisantes) $\mathbf{T}_k(S_k, q_k)$, est assurée par la propriété 1 (Annexe A.2). Remarquons que ces critères MAP (7.5) correspondent au critère (6.3) en supposant de plus que les données complètes $\mathcal{Z} = \{q_1, S_1, q_2, S_2\}$ sont observées.

L'interprétation suivante peut être donnée à cet algorithme EM (7.3), (7.4). Si les données complètes $\mathcal{Z} = \{q_1, S_1, q_2, S_2\}$ étaient observées, on pourrait utiliser les critères MAP (7.5) et leurs solutions seraient $\lambda_k = \mathbf{f}_k\left(\mathbf{T}_k(S_k, q_k)\right)$. Cependant, puisqu'elles ne sont pas observées, les valeurs des statistiques naturelles $\mathbf{T}_k(S_k, q_k)$ sont remplacées par leurs espérances conditionnellement aux données observées X (Eq. (7.3)) et aux modèles estimés à l'itération précédente. Ainsi, l'étape E (7.3) consiste à calculer les espérances conditionnelles des statistiques naturelles et l'étape M (7.4) consiste à estimer les nouveaux paramètres des modèles en utilisant ces espérances.

L'algorithme EM (7.3), (7.4) est schématisé sur la figure 7.1. Ce schéma permet de mieux comprendre les interactions entre les différents blocs de cet algorithme. Remarquons que les données observées X (le mélange) interviennent seulement dans l'étape E et les loi *a priori* sur les modèles $p(\lambda_k|\Lambda_k)$, $k = 1, 2$ seulement dans l'étape M. Les étapes E et M interagissent entre elles de la manière suivante. L'étape E envoie à l'étape M les statistiques naturelles qui sont traitées indépendamment pour chaque source $k = 1, 2$. L'étape M lui renvoie ensuite les nouvelles estimations des modèles $\lambda_1^{(l)}$ et $\lambda_2^{(l)}$.

De plus, on voit maintenant que l'adaptation est effectuée vraiment à partir du mélange X sans être passée par des estimations intermédiaires des sources. En effet, comme on le voit sur la figure 7.1, toutes les grandeurs nécessaires pour réestimer les paramètres des modèles (c'est-à-dire les statistiques naturelles) sont estimées directement à partir du mélange X.

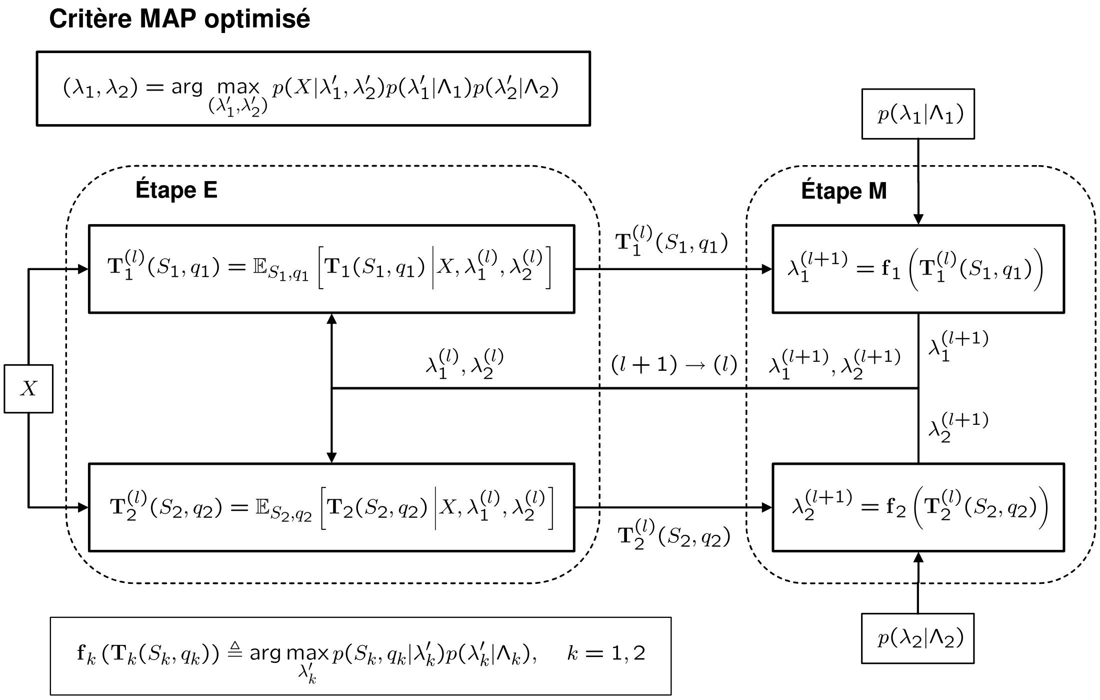


FIG. 7.1 – Algorithme EM pour l'optimisation du critère MAP (6.3) dans le cas des familles exponentielles.

7.3 Statistiques naturelles des MMG et leurs espérances conditionnelles

Pour les MMG spectraux, les familles des densités $\{p(S_k, q_k|\lambda_k)\}_{\lambda_k}$ sont des familles exponentielles et leurs statistiques naturelles s'écrivent comme suit (voir Annexe B.1 pour une preuve) :

$$\mathbf{T}_k(S_k, q_k) = \left\{\mathbf{t}_{k,i}^0, \{\mathbf{t}_{k,i}^2(f)\}_{f=1}^F\right\}_{i=1}^{Q_k}, \quad k = 1, 2 \tag{7.6}$$

c'est-à-dire chaque statistique $\mathbf{T}_k(S_k, q_k)$ est un ensemble de $Q_k + Q_k F$ statistiques scalaires $\{\mathbf{t}_{k,i}^0\}_{i=1}^{Q_k}$ et $\{\mathbf{t}_{k,i}^2(f)\}_{i,f=1}^{Q_k,F}$. Ces statistiques scalaires sont définies comme suit :

1. La statistique $\mathbf{t}_{k,i}^0$ compte le nombre de fois que l'état i à été observé, c'est-à-dire

$$\mathbf{t}_{k,i}^0 \triangleq \sum_t \delta(q_k(t), i), \quad \text{et} \tag{7.7}$$

   où $\delta(i, j)$ est le symbole de Kronecker qui vaut 1, si $i = j$, et vaut 0, si $i \neq j$.

2. La statistique $\mathbf{t}_{k,i}^2(f)$ représente l'énergie de la TFCT S_k associée à l'état i pour la fréquence f, c'est-à-dire

$$\mathbf{t}_{k,i}^2(f) \triangleq \sum_t |S_k(t,f)|^2 \delta(q_k(t), i), \tag{7.8}$$

L'algorithme 4 résume le calcul des espérances conditionnelles (7.3) des statistiques naturelles (7.6) (voir Annexe B.2 pour une preuve).

Algorithme 4 Calcul des espérances des statistiques naturelles (7.6) conditionnellement au mélange X et aux modèles $\lambda_k = \{\omega_{k,i}, \Sigma_{k,i}\}_i$, $k = 1, 2$. Ce calcul est présenté pour la source S_1 (pour la source S_2 le calcul est analogue).

---

1. Calculer les poids $\gamma_{i,j}^{(l)}(t)$ satisfaisant $\sum_{i,j} \gamma_{i,j}^{(l)}(t) = 1$ et

$$\gamma_{i,j}^{(l)}(t) \triangleq P(q_1(t) = i, q_2(t) = j | X, \lambda_1^{(l)}, \lambda_2^{(l)}) \propto \omega_{1,i}^{(l)} \omega_{2,j}^{(l)} N_C(X(t); \bar{0}, \Sigma_{1,i}^{(l)} + \Sigma_{2,j}^{(l)}) \tag{7.9}$$

2. Calculer l'espérance de la DSP pour la paire d'états (i, j) :

$$\begin{aligned} \left\langle |S_1(t,f)|^2 \right\rangle_{i,j}^{(l)} &\triangleq \mathbb{E}_{S_1}\left[|S_1(t,f)|^2 \,\middle|\, q_1(t) = i, q_2(t) = j, X, \lambda_1^{(l)}, \lambda_2^{(l)} \right] = \\ &= \frac{\sigma_{1,i}^{2,(l)}(f)\sigma_{2,j}^{2,(l)}(f)}{\sigma_{1,i}^{2,(l)}(f) + \sigma_{2,j}^{2,(l)}(f)} + \left| \frac{\sigma_{1,i}^{2,(l)}(f)}{\sigma_{1,i}^{2,(l)}(f) + \sigma_{2,j}^{2,(l)}(f)} X(t,f) \right|^2 \end{aligned} \tag{7.10}$$

3. Calculer l'espérance conditionnelle de $\mathbf{t}_{1,i}^0$:

$$\mathbf{t}_{1,i}^{0,(l)} \triangleq \mathbb{E}_{S_1,q_1}\left[\mathbf{t}_{1,i}^0 \,\middle|\, X, \lambda_1^{(l)}, \lambda_2^{(l)} \right] = \sum_t \sum_j \gamma_{i,j}^{(l)}(t), \tag{7.11}$$

4. Calculer l'espérance conditionnelle de $\mathbf{t}_{1,i}^2(f)$:

$$\mathbf{t}_{1,i}^{2,(l)}(f) \triangleq \mathbb{E}_{S_1,q_1}\left[\mathbf{t}_{1,i}^2(f) \,\middle|\, X, \lambda_1^{(l)}, \lambda_2^{(l)} \right] = \sum_t \sum_j \left\langle |S_1(t,f)|^2 \right\rangle_{i,j}^{(l)} \gamma_{i,j}^{(l)}(t), \tag{7.12}$$

Ainsi, nous avons donné tous les éléments pour pouvoir adapter des modèles en utilisant l'algorithme EM (7.3), (7.4). Il reste à résoudre les critères MAP (7.5) quand les lois *a priori* $p(\lambda_k | \Lambda_k)$ sont spécifiées.

Sous sa forme actuelle, l'algorithme présente un bon degré de généralité puisqu'il ne fait pas d'hypothèses sur la nature exacte des lois *a priori* sur les modèles λ_k.

7.4 Conclusion

L'algorithme EM pour l'optimisation du critère d'adaptation MAP (6.3) a été présenté aux différents niveaux de généralité suivants :

– le plus général : cet algorithme EM peut être appliqué pour l'adaptation de n'importe quel modèle probabiliste à états cachés,
– pour les modèles dont les vraisemblances conjointes des observations et des états cachées sont des familles exponentielles,
– pour les MMG spectraux traités dans cette thèse,

De cette présentation découlent assez naturellement les algorithmes à mettre en oeuvre pour les différents types de modèles et de lois *a priori*. C'est ce qui sera mis en évidence dans la partie III de cette thèse, où ce formalisme sera décliné dans différentes situations.

Auparavant, nous présentons plusieurs extensions du formalisme d'adaptation à partir de diverses connaissances, qui font intervenir des contraintes paramétriques ou des informations auxiliaires, même si celles-ci ne sont pas formulées par un modèle probabiliste.

Chapitre 8

Extensions du formalisme d'adaptation

Dans de nombreuses circonstances, les connaissances sur les sources ne s'expriment pas seulement par le mélange, les modèles généraux et les lois *a priori*. Dans certaines situations, il peut être approprié d'utiliser d'autre types de contraintes sur les modèles adaptés que des lois probabilistes *a priori*. En outre, si en plus du mélange, il existe d'autres informations sur les sources, il est important de les intégrer également dans le processus d'adaptation des modèles.

Ainsi, nous introduisons dans ce chapitre deux extensions du formalisme d'adaptation présenté dans le chapitre précédent :

- l'adaptation contrainte, où les lois *a priori* sont remplacées par des contraintes paramétriques. Ceci est inspiré de technique d'adaptation basées sur des transformations, telles que par exemple *Maximum Likelihood Linear Regression* (MLLR) [Leggetter-95].
- l'intégration dans le processus d'adaptation de diverses informations auxiliaires sur les sources ou les états cachés des modèles.

Ces extensions sont également formulées par des critères MAP et représentées par des réseaux bayésiens. Par ailleurs, nous montrons dans ce chapitre que l'algorithme d'adaptation présenté dans le chapitre 7 se généralise sans difficulté pour ces deux extensions.

8.1 Adaptation contrainte

Le principe d'*adaptation contrainte* consiste à supposer que les paramètres de chaque modèle adapté λ_k appartiennent à un *sous ensemble de paramètres admissibles* $\Xi_k(\Lambda_k)$ qui dépend éventuellement des paramètres du modèle général Λ_k. Comme avant, il est supposé que, sur ce sous ensemble, les paramètres de λ_k possèdent une densité *a priori* qui dépend aussi du modèle général. La façon la plus simple de le concevoir est de supposer que les paramètres du modèle adapté λ_k et ceux du modèle général Λ_k sont reliés entre eux à l'aide d'une déformation paramétrique Ψ_k, avec les paramètres de déformation C_k (que nous appelons aussi *paramètres libres*), notamment $\lambda_k = \Psi_k(C_k, \Lambda_k)$.

Ainsi, la procédure d'adaptation consiste à trouver les paramètres libres C_1 et C_2 en utilisant le critère MAP suivant :

$$(C_1, C_2) = \arg\max_{(C_1', C_2')} p\left(X|\lambda_1' = \Psi_1(C_1', \Lambda_1), \lambda_2' = \Psi_2(C_2', \Lambda_2)\right) p(C_1'|\Lambda_1)p(C_2'|\Lambda_2), \quad (8.1)$$

où $p(C_k|\Lambda_k)$, $k = 1, 2$ sont des lois *a priori* sur les paramètres libres. Les modèles adaptés sont ensuite obtenus comme $\lambda_k = \Psi_k(C_k, \Lambda_k)$, $k = 1, 2$.

Formellement, ce critère MAP (8.1) est différent du critère (6.3), mais en pratique on peut s'y ramener en considérant la loi *a priori* $p(\lambda_k|\Lambda_k)$ non plus comme une loi probabiliste, mais simplement comme une contrainte sur le modèle λ_k. Dans ce cas, le critère (8.1) devient un cas particulier de (6.3) avec la contrainte composée d'une contrainte paramétrique sur le modèle $\lambda_k = \Psi_k(C_k, \Lambda_k)$ et d'une loi *a priori* sur les paramètres libres $p(C_k|\Lambda_k)$.

L'adaptation contrainte est représentée sur la figure 8.1 par un réseau bayésien.

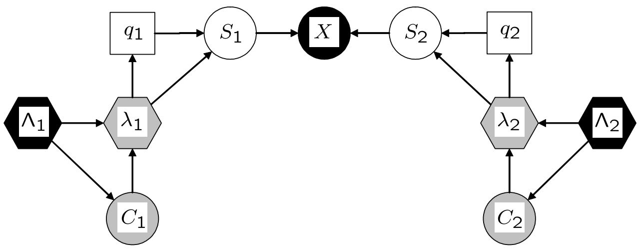


FIG. 8.1 – Réseau bayésien représentant l'adaptation contrainte. Formes des noeuds : processus continus (ronds), processus discrets (carrés), modèles (hexagones). Coloration des noeuds : noeuds observés (noir), noeuds cachés estimés (gris), autres noeuds cachés (blanc).

Par ailleurs, on peut montrer que le fait de considérer la loi *a priori* simplement comme une contrainte ne change pas en principe l'algorithme EM (7.3), (7.4) qui prend la forme suivante dans le cas du critère (8.1) :

$$\mathbf{E:} \quad \mathbf{T}_k^{(l)}(S_k, q_k) = \mathbb{E}_{S_k, q_k}\left[\mathbf{T}_k(S_k, q_k) \middle| X, \lambda_1^{(l)}, \lambda_2^{(l)}\right], \quad k = 1, 2 \quad (8.2)$$

$$\mathbf{M:} \quad C_k^{(l+1)} = \mathbf{g}_k\left(\mathbf{T}_k^{(l)}(S_k, q_k)\right), \quad \lambda_1^{(l+1)} = \Psi_k(C_k^{(l+1)}, \Lambda_k), \quad k = 1, 2 \quad (8.3)$$

où les fonctions $\mathbf{g}_k\left(\mathbf{T}_k(S_k, q_k)\right)$, $k = 1, 2$ sont définies comme des solutions des critères MAP des données complètes :

$$\mathbf{g}_k\left(\mathbf{T}_k(S_k, q_k)\right) \triangleq \arg\max_{C'_k} p\left(S_k, q_k | \lambda'_k = \Psi_k(C'_k, \Lambda_k)\right) p(C'_k|\Lambda_k), \quad k = 1, 2 \tag{8.4}$$

Enfin, notons que l'adaptation contrainte permet de sélectionner des caractéristiques adaptables définies par les paramètres libres C_k et de figer toutes les autres caractéristiques. Le degré d'adaptabilité est défini par la loi *a priori* $p(C_k|\Lambda_k)$. Cela permet de choisir des contraintes *a priori* assez prudemment, ce qui peut être essentiel (voir la discussion sur le rôle des lois *a priori* présentée dans la section 6.2.2).

Remarquons aussi que parmi les techniques d'adaptation utilisées pour la reconnaissance de la parole et la vérification du locuteur, l'adaptation MLLR [Leggetter-95, Gales-96] et l'adaptation EMLLR [Chen-00] sont des techniques d'adaptation contrainte.

8.2 Utilisation d'informations auxiliaires

Quand seul le mélange X est observé, l'adaptation des modèles est une tâche assez difficile, car on ne peut pas laisser beaucoup de latitude aux modèles adaptés, c'est-à-dire que les lois *a priori* doivent être assez restrictives. Ainsi, si jamais d'autres informations sur les sources S_k et (ou) sur les séquences d'états q_k sont disponibles, il est très souhaitable de les utiliser pour l'adaptation. L'utilisation de ces informations, appelées *informations auxiliaires* et notées I, pourra par exemple mener à une meilleure identification des états cachés. Ainsi, on pourra choisir des lois *a priori* moins strictes, ce qui mènera vraisemblablement à une meilleure adaptation et éventuellement une meilleure séparation.

On peut distinguer deux types d'informations auxiliaires :

– les informations auxiliaires connues *a priori*,
– les informations auxiliaires estimées de manière fiable en utilisant certaines méthodes d'estimation sur le mélange.

Voici quelques exemples d'informations auxiliaires :

1. Considérons par exemple une segmentation temporelle en zones d'activité des sources de trois types :
   – la source s_1 est active,
   – la source s_2 est active,
   – les deux sources sont actives à la fois.

   La connaissance d'une telle segmentation peut être très utile pour améliorer la qualité des modèles adaptés. Il est ainsi très souhaitable de l'utiliser comme information auxiliaire.

Dans ce travail, nous allons utiliser pour la séparation voix / musique une segmentation temporelle de la chanson traitée en parties vocales (avec voix chantée) et non vocales (sans voix chantée), ce qui nous permettra d'améliorer considérablement le modèle de musique.

2. Considérons maintenant un cas plus général, quand au lieu de segments temporels, on connaît des régions temps - fréquence où une seule source est active. Ainsi, il s'agit d'une sorte de "séparation incomplète", où l'on identifie laquelle des deux sources domine l'autre dans certaines régions temps - fréquence. L'utilisation de ces informations auxiliaires au sein de notre formalisme d'adaptation ressemble beaucoup à la "théorie des données manquantes" [Ghahramani-93] utilisée pour la reconnaissance de la parole dans un environnement bruité [Cooke-01].

3. Supposons que dans le contexte de séparation des signaux de parole, nous avons des modèles dont les états correspondent aux phonèmes. Dans ce cas, la connaissance des phrases prononcées peut considérablement restreindre l'ensemble des séquences d'états possibles.

4. Pour la séparation des signaux de parole, Hershey et Casey [Hershey-01] utilisent comme informations auxiliaires des paramètres extraits à partir des vidéos des mouvements des lèvres correspondant aux signaux audio séparés. L'utilisation de ces informations vidéo permet d'améliorer le résultat de séparation des signaux audio. Dans le cadre de séparation de sources multicapteur, cas déterminé, Rivet *et al.* [Rivet-04] montrent que l'utilisation des informations vidéo permet de résoudre le problème des permutations en fréquence pour des mélanges convolutifs. Ces informations pourraient être également utilisées pour l'adaptation des modèles.

La figure 8.2 illustre comment les informations auxiliaires I s'intègrent dans le réseau bayésien représentant l'adaptation (Fig. 6.2 (C)). Ces informations dépendent des sources S_k, $k = 1, 2$ et des séquences d'états q_k, $k = 1, 2$. Elles peuvent éventuellement dépendre des modèles adaptés λ_k, $k = 1, 2$. Dans ce cas, les modèles adaptés doivent être complétés par des paramètres qui génèrent I.

En intégrant les informations auxiliaires I dans le critère MAP (6.3), on obtient le critère suivant :

$$(\lambda_1, \lambda_2) = \arg \max_{(\lambda'_1, \lambda'_2)} p(X, I|\lambda'_1, \lambda'_2) p(\lambda'_1|\Lambda_1) p(\lambda'_2|\Lambda_2) \tag{8.5}$$

On voit que la seule nouveauté de ce critère par rapport au critère (6.3) est que le mélange X est complété par des observations supplémentaires I. Ainsi, ce critère peut être optimisé par l'algorithme EM (7.3), (7.4) en y remplaçant les données observées X par $\{X, I\}$.

De la même façon, des informations auxiliaires I peuvent être intégrées dans la procédure

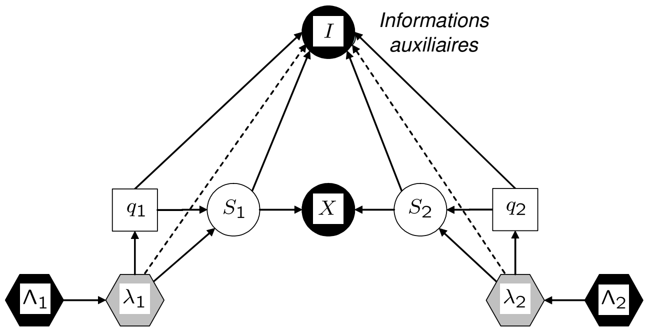


FIG. 8.2 – Réseau bayésien représentant la prise en compte des informations auxiliaires I dans la procédure d'adaptation des modèles (Fig. 6.2 (C)). Formes des noeuds : processus continus (ronds), processus discrets (carrés), modèles (hexagones). Coloration des noeuds : noeuds observés (noir), noeuds cachés estimés (gris), autres noeuds cachés (blanc).

d'estimation des sources (Fig. 6.2 (B)).

8.3 Conclusion

Dans ce chapitre, deux extensions importantes du formalisme d'adaptation ont été présentées. Ces extensions permettent d'unifier dans le formalisme des réseaux bayésiens différentes modalités d'incorporation des connaissance *a priori* sous la forme des critères MAP. Ceci fait converger dans un même paradigme la possibilité d'exploiter diverses sources d'information hétérogènes pour l'estimation des propriétés statistiques des sources et leur séparation.

Dans la partie III, nous allons maintenant décliner et utiliser ce formalisme ainsi que ses extensions dans le contexte pratique de séparation voix / musique.

Troisième partie

Application d'adaptation à la séparation voix / musique

Chapitre 9

Système de séparation voix / musique

Dans la partie II, nous avons développé un formalisme d'adaptation des modèles *a priori* aux caractéristiques des sources dans le mélange. Ce formalisme étant présenté de manière assez abstraite et théorique, nous allons l'appliquer dans ce chapitre à un problème concret, la séparation voix / musique.

Comme il a été déjà remarqué dans l'introduction et rediscuté dans la section 1.3, bien séparer la voix par rapport à la musique ambiante peut être très utile pour de nombreuses tâches de l'indexation audio. En effet, la voix chantée contient beaucoup d'informations sémantiques caractérisant la chanson (la mélodie, la parole chantée, l'identité du chanteur, etc.), et il paraît plus facile d'extraire ces informations à partir de la voix bien séparée. Ainsi, il semble important d'apporter des solutions au problème de séparation voix / musique avec un seul capteur. Cependant, ce problème est assez difficile, et il a été assez peu traité dans la littérature [Ozerov-05b, Vembu-05, Li-06].

Comme nous l'avons observé dans le chapitre 4, l'utilisation des modèles généraux pour cette tâche de séparation donne des performances médiocres, qui ne peuvent être sensiblement améliorées, même en augmentant les tailles des modèles. Nous supposons que ceci est dû aux limitations principales des modèles *a priori* présentées chapitre 5, c'est-à-dire la nécessité de modèles de grande taille et le manque de données d'entraînement représentatives.

Ainsi, le formalisme d'adaptation étant développé pour pouvoir dépasser dans certains cas ces limitations, nous espérons que, pour la séparation voix / musique, il permettra d'améliorer les performances des modèles généraux et d'utiliser des modèles de taille raisonnable. Nous allons donc développer dans ce chapitre un système de séparation voix / musique basé sur des modèles adaptés.

9.1 Système de séparation

Le système de séparation voix / musique est construit selon le schéma représenté figure 6.1. Les modules d'apprentissage des modèles généraux (MMG spectraux) sont implémentés selon l'algorithme 2. L'estimation des sources en minimisant l'EQM spectrale est effectuée à l'aide du filtrage de Wiener adaptatif présenté section 2.4.1.2.

Nous présentons ensuite le module d'adaptation dont les blocs sont basés sur différentes formes du formalisme d'adaptation présenté dans la partie II.

9.2 Description du module d'adaptation

Remarquons que dans les chansons, il y a souvent beaucoup de zones temporelles où la musique est présente seule sans voix chantée. Ces zones sont appelées ici *parties non vocales* par opposition aux *parties vocales*, où la voix est présente. La tâche de segmentation des chansons en parties vocales et non vocales a été déjà traitée dans la littérature [Berenzweig-01, Kim-02, Nwe-04, Tsai-04a].

L'idée principale, inspirée initialement par les travaux de Tsai *et al.* [Tsai-04], est d'utiliser les parties non vocales pour adapter le modèle de musique. Ensuite, le modèle de musique ainsi obtenu et le modèle général de voix sont encore adaptés sur toute la chanson.

Ainsi, le module d'adaptation schématisé figure 9.1 consiste en trois étapes :

1. La chanson X (dans le domaine de la TFCT) est segmentée en parties vocales $\{X(t)\}_{t \in \mathbf{voc}}$ et non vocales $\{X(t)\}_{t \notin \mathbf{voc}}$, où **voc** dénote l'ensemble des indices des trames vocales.

2. Le modèle général de musique Λ_m est adapté sur les parties non vocales $\{X(t)\}_{t \notin \mathbf{voc}}$, en réestimant seulement les paramètres des gaussiennes qui sont suffisamment observées au sens des nouvelles *données acoustiques*, c'est-à-dire des trames des parties non vocales. Cette adaptation est appelée *adaptation acoustique* et le nouveau modèle de musique ainsi obtenu $\tilde{\lambda}_m$ est appelé *modèle adapté acoustiquement.*

3. Le modèle de musique adapté acoustiquement $\tilde{\lambda}_m$ et le modèle général de voix Λ_v sont adaptés sur toute la chanson X en utilisant la technique d'adaptation des filtres et des gains de DSP présentée par la suite. Cette technique d'adaptation consiste à normaliser les modèles par rapport aux nouvelles conditions d'enregistrement (adaptation des filtres), ainsi que par rapport aux énergies relatives des DSP (adaptation des gains de DSP).

Les modèles adaptés λ_v et λ_m sont enfin utilisés pour estimer les sources (Fig. 6.1).

Comme nous le verrons par la suite, les deux blocs d'adaptation (adaptation acoustique et adaptation des filtres et des gains de DSP) sont des cas particuliers du formalisme d'adaptation général développé dans la partie II. Le bloc de segmentation en parties vocales et non vocales peut être vu au sens de ce formalisme comme un module d'extraction des informations auxiliaires (Sec. 8.2) utilisées par le bloc d'adaptation acoustique.

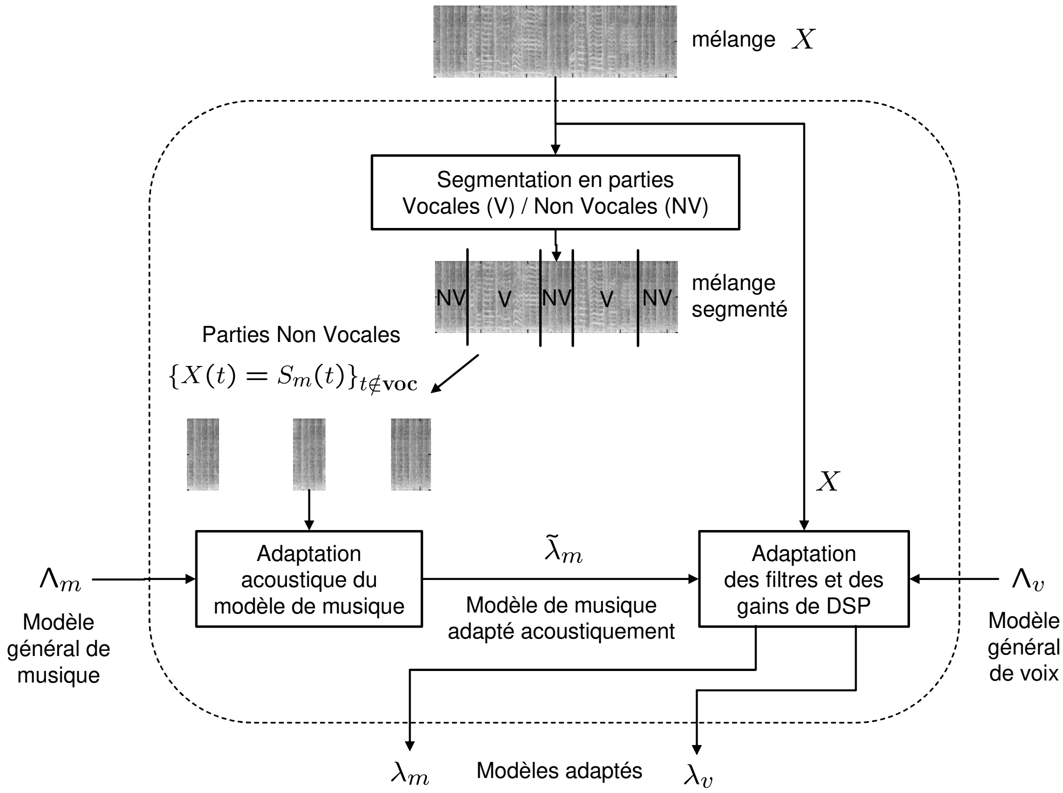


FIG. 9.1 – Module d'adaptation pour la séparation voix / musique.

Dans les trois sections suivantes, les trois blocs fonctionnels de ce schéma d'adaptation (Fig. 9.1) sont présentés en détails.

9.3 Segmentation en parties vocales et non vocales

La tâche de segmentation des chansons en parties vocales et non vocales a déjà été traitée dans la littérature (voir par ex. [Berenzweig-01, Kim-02, Nwe-04, Tsai-04a]). Nous avons choisi d'utiliser une approche assez classique basée sur des MMG [Tsai-04, Tsai-04a].

La TFCT de la chanson $X = \{X(t)\}_t$, qui est une suite de spectres à court terme, est transformée en une suite de vecteurs de paramètres acoustiques $B = \{B(t)\}_t$ (typiquement MFCC [Vergin-99]). Pour décider si le vecteur $B(t)$ correspond à la trame vocale ou non vocale, on utilise les MMG Γ_V et Γ_N, qui modélisent respectivement les trames vocales et non vocales. La structure de ces MMG est la même que celle des MMG log spectraux (Sec. 2.4.2), c'est-à-dire que les observations sont des vecteurs réels (ici c'est $B(t)$), les vecteurs moyens ne sont pas nuls et les matrices de covariance sont diagonales. Ces MMG sont appris à partir de données d'entrainement composées de chansons segmentées manuellement en parties vocales et non vocales. Comme ces modèles ont la même structure que les MMG log spectraux, l'algorithme 3 peut être utilisé pour cet apprentissage.

La décision pour la t-ème trame peut être obtenue en comparant le logarithme du rapport de vraisemblance avec un seuil de décision η :

$$\log p\left(B(t)|\Gamma_V\right) - \log p\left(B(t)|\Gamma_N\right) \underset{\text{non voc}}{\overset{\text{voc}}{\gtrless}} \eta \tag{9.1}$$

Cependant, la performance de segmentation peut être significativement augmentée en faisant la décision non plus sur une trame, mais sur un bloc [Tsai-04, Tsai-04a], c'est-à-dire un groupe de plusieurs trames consécutives. Pour cette décision par blocs, le logarithme du rapport de vraisemblance (9.1) doit être remplacé par sa moyenne calculée sur toutes les trames du bloc :

$$\frac{1}{2L+1}\sum_{l=t-L}^{t+L}\left[\log p\left(B(l)|\Gamma_V\right) - \log p\left(B(l)|\Gamma_N\right)\right] \underset{\text{non voc}}{\overset{\text{voc}}{\gtrless}} \eta \tag{9.2}$$

où $U = 2L+1$ est la taille d'un bloc en nombre de trames. Remarquons que, dans le choix de cette taille U, on se retrouve devant un compromis. Plus U est grand, plus la décision est robuste, car elle est faite sur un plus grand nombre de trames, mais en même temps moins les frontières des parties vocales et non vocales identifiées sont précises.

Cette procédure de segmentation en parties vocales et non vocales avec la décision par blocs est également schématisée figure 9.2.

9.4 Adaptation acoustique du modèle de musique

L'adaptation acoustique du modèle général de musique Λ_m aux nouvelles données acoustiques, qui sont les trames non vocales $\{X(t)\}_{t\notin\mathbf{voc}}$, consiste à optimiser le critère MAP suivant :

$$\tilde{\lambda}_m = \arg\max_{\lambda'_m} p(\{X(t)\}_{t\notin\mathbf{voc}}|\lambda'_m)p(\lambda'_m|\Lambda_m) \tag{9.3}$$

où la loi *a priori* $p(\lambda_m|\Lambda_m)$ définit la façon selon laquelle le modèle adapté λ_m est attaché au modèle général Λ_m. Comme il a été discuté dans la section 6.2.2, il existe de nombreuses possibilités pour spécifier cette loi *a priori*.

Ici, nous supposons que cette loi est constituée des lois *a priori* conjuguées sur les paramètres du MMG λ_m (la loi Normal - Wishart inverse pour les matrices de covariance $\Sigma_{m,j}$ et la loi de Dirichlet pour les poids des gaussiennes $\omega_{m,j}$) [Gauvain-94]. Dans ce cas, en appliquant l'algorithme EM [Dempster-77] pour l'optimisation du critère MAP (9.3), la formule suivante peut être obtenue pour la réestimation des variances $\sigma^2_{m,j}(f)$ (pour les poids $\omega_{m,j}$ il y a une

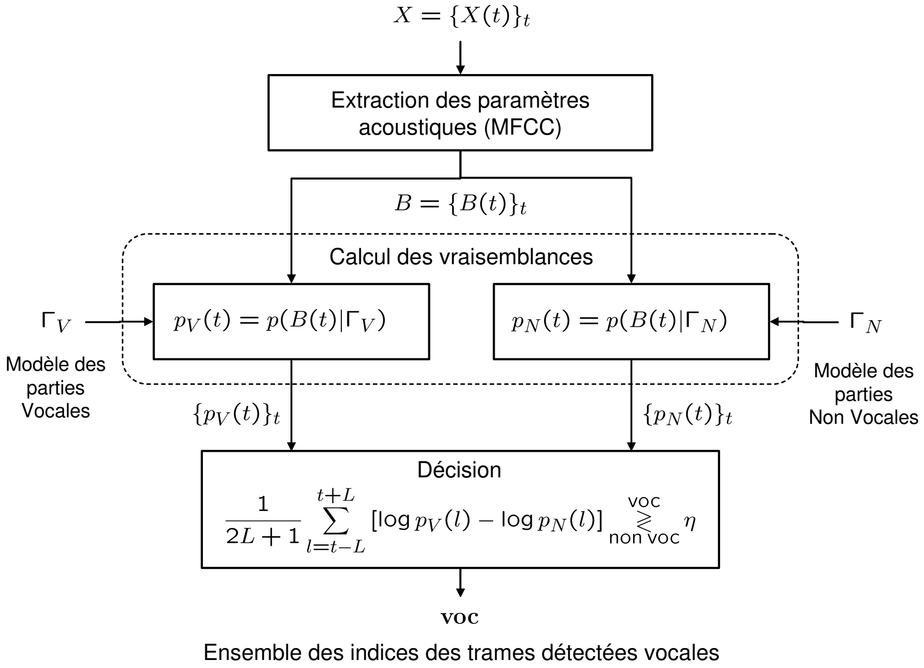


FIG. 9.2 – Segmentation en parties vocales et non vocales avec la décision par blocs.

formule analogue) [Gauvain-94] :

$$\sigma^{2,(l+1)}_{m,j}(f) = \alpha^{(l)}_j(\tau)\frac{\sum_{t\notin\mathbf{voc}} |X(t,f)|^2\gamma^{(l)}_j(t)}{\sum_{t\notin\mathbf{voc}}\gamma^{(l)}_j(t)} + \left(1-\alpha^{(l)}_j(\tau)\right) r^2_{m,j}(f), \tag{9.4}$$

où :

- l'exposant (l) dénote les paramètres estimés en la l-ème itération de EM,
- les poids $\gamma^{(l)}_j(t)$ satisfont $\sum_j \gamma^{(l)}_j(t) = 1$ et sont calculés comme :

$$\gamma^{(l)}_j(t) \propto \omega^{(l)}_{m,j} N_C\left(X(t);\bar{0},\Sigma^{(l)}_{m,j}\right), \tag{9.5}$$

  avec la densité d'un vecteur gaussien complexe circulaire $N_C(\cdot)$ définie selon l'équation (A.2),
- pour chaque état j le paramètre $\alpha^{(l)}_j(\tau)$ est calculé comme :

$$\alpha^{(l)}_j(\tau) = \frac{\zeta^{(l)}_j}{\zeta^{(l)}_j + \tau} \tag{9.6}$$

  avec $\zeta^{(l)}_j = \sum_{t\notin\mathbf{voc}} \gamma^{(l)}_j(t)$ et $\tau > 0$ étant un paramètre appelé *facteur de confiance* [Ben-04].

Cette procédure d'adaptation peut être interprétée de la manière suivante. Si il y a beaucoup de nouvelles données observées pour une gaussienne (une DSP), ses paramètres sont réestimés sur ces données. Si par contre il y a peu de nouvelles données, les paramètres du modèle général sont gardés pour cette gaussienne. En effet, la grandeur $\zeta_j^{(l)}$ représente la quantité de nouvelles données observées pour la j-ème gaussienne. Si $\zeta_j^{(l)}$ est grand $\alpha_j^{(l)}(\tau)$ vaut presque 1 et les paramètres sont réestimés. Si $\zeta_j^{(l)}$ est petit $\alpha_j^{(l)}(\tau)$ vaut presque 0 et les paramètres *a priori* sont conservés (par ex. $[r^2_{m,j}(f)]_f$ pour les variances). Cette précaution permet d'éviter la suradaptation d'une gaussienne sur peu de nouvelles données.

Le facteur de confiance τ règle le niveau d'attachement du modèle adapté λ_m au modèle général Λ_m. Quand τ tend vers $+\infty$, la loi *a priori* $p(\lambda_m|\Lambda_m)$ tend vers une distribution de Dirac, et on s'attache de plus en plus au modèle général Λ_m. Dans le cas extrême ($\tau = \infty$), il n'y a pas d'adaptation ($\lambda_m = \Lambda_m$). Quand τ tend vers 0 la loi *a priori* tend vers une loi uniforme non informative, c'est-à-dire $p(\lambda_m|\Lambda_m) \propto const$, et on perd complètement l'attache au modèle général Λ_m. Dans le cas extrême ($\tau = 0$), le critère MAP (9.3) se transforme en critère du MV suivant :

$$\tilde{\lambda}_m = \arg\max_{\lambda'_m} p(\{X(t)\}_{t\notin\mathbf{voc}}|\lambda'_m), \tag{9.7}$$

et la formule de réestimation des variances se simplifie comme suit :

$$\sigma^{2,(l+1)}_{m,j}(f) = \frac{\sum_{t\notin\mathbf{voc}} |X(t,f)|^2 \gamma_j^{(l)}(t)}{\sum_{t\notin\mathbf{voc}} \gamma_j^{(l)}(t)} \tag{9.8}$$

Cette dernière formulation correspond au réapprentissage complet du modèle de musique sur les parties non vocales, car il n'y a plus aucune attache au modèle général.

On sent qu'il serait plus pertinent d'utiliser l'adaptation MAP (9.3) plutôt que le réapprentissage au MV (9.7), au moins pour éviter le surapprentissage du modèle quand il y a très peu, voire pas du tout, de trames détectées comme non vocales. De plus, comme il a été remarqué, le critère du MV (9.7) est un cas extrême du critère MAP (9.3). Ainsi, en utilisant le critère MAP, on peut se rapprocher autant qu'on veut du critère du MV en choisissant le facteur τ suffisamment petit.

Pour vérifier cette supposition, nous allons faire une petite étude expérimentale.

9.4.1 Illustration expérimentale

Nous étudions l'influence sur les performances de séparation du facteur de confiance τ utilisé pour l'adaptation acoustique du modèle de musique.

Les données de test et d'entraînement sont décrites dans la section 4.4. Tous les modèles utilisés sont à 32 états ($Q_v = Q_m = 32$). Les expériences effectuées pour différentes valeurs du facteur τ sont résumées par les étapes suivantes :

1. Les modèles généraux Λ_v et Λ_m sont appris à partir des données d'entraînement (Alg. 2).
2. Chaque chanson de test est segmentée manuellement en parties vocales et non vocales.
3. Pour chaque valeur du facteur de confiance τ et pour chaque chanson de test :
   (a) Le modèle de musique est adapté acoustiquement sur les parties non vocales $\{X(t)\}_{t \notin \mathbf{voc}}$ en utilisant la formule de réestimation (9.4).
   (b) La séparation est effectuée en utilisant le modèle général de voix Λ_v et le modèle de musique adapté acoustiquement $\tilde{\lambda}_m$.

Le RSDN moyen en fonction du logarithme à base 2 du facteur de confiance τ est représenté sur la figure 9.3. Ces résultats sont complétés par les trois cas spéciaux :

1. Maximum de Vraisemblance (MV) : le réapprentissage complet en utilisant la formule de réestimation (9.8). L'algorithme des K-moyennes [McQueen-67] est utilisé pour l'initialisation.
2. $\tau = 0$: la seule différence par rapport au cas précédent est que le modèle général de musique Λ_m est utilisé pour l'initialisation.
3. $\tau = +\infty$: le modèle de musique Λ_m n'est pas adapté, c'est-à-dire les deux modèles généraux Λ_v et Λ_m sont utilisés pour la séparation.

La figure 9.3 nous confirme que quand $\tau \to 0$ on s'approche du réapprentissage au MV et quand $\tau \to +\infty$ on s'approche de l'utilisation des modèles généraux. Cependant, contrairement à ce qu'on attendait, le meilleur résultat est obtenu pour le réapprentissage complet au MV. Ainsi, dans le cas du système de séparation et de la base d'évaluation utilisés, l'attache au modèle général de musique n'apporte rien par rapport au réapprentissage complet. Vraisemblablement, ceci est lié au fait que dans les chansons de la base d'évaluation, il y a suffisamment de zones temporelles sans voix chantée pour réapprendre le modèle de musique. Notons que cette propriété est vérifiée pour la majorité des chansons populaires. Ainsi, nous avons choisi d'utiliser par la suite le réapprentissage complet du modèle de musique.

Toutefois, lorsqu'il est probable de rencontrer des chansons avec très peu (voir pas du tout) de zones temporelles sans voix chantée, il faut revenir à l'adaptation à l'aide du critère MAP (9.3). Même dans le cas des résultats que nous avons obtenus (voir Fig. 9.3), l'adaptation avec un facteur de confiance assez petit (disons $\log_2(\tau) \leq 2$) est une solution satisfaisante. En effet, la différence des RSDN pour le réapprentissage et pour l'adaptation avec un petit facteur τ ne semble pas très significative. Ainsi, en faisant l'adaptation, la petite perte en performances (probablement non significative) sera compensée par la robustesse du système dans les cas où il n'y a pas assez de trames non vocales pour un réapprentissage complet.

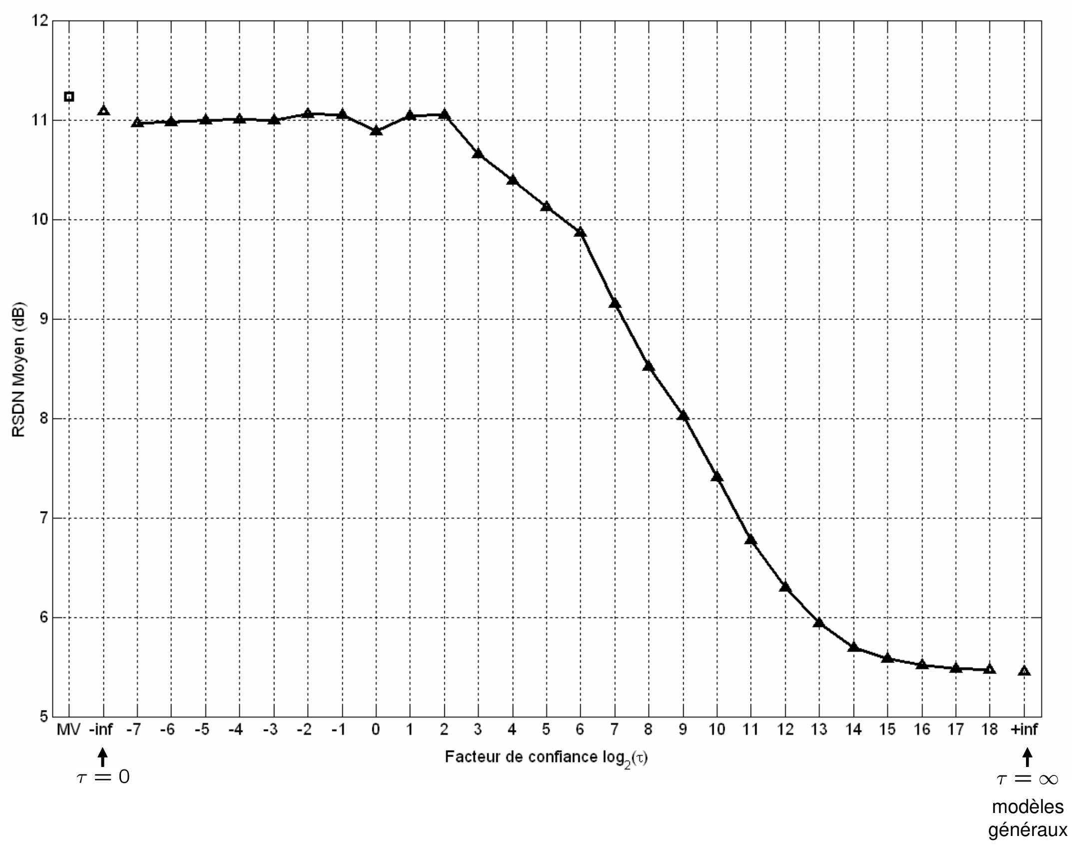


FIG. 9.3 – RSDN moyen en fonction du logarithme à base 2 du facteur de confiance τ pour l'adaptation acoustique du modèle de musique. Trois cas spéciaux : Maximum de Vraisemblance (MV) (initialisation par K-moyennes), $\tau = 0$ (initialisation par le modèle général de musique Λ_m), $\tau = +\infty$ (modèles généraux Λ_v et Λ_m).

9.4.2 Explication du réapprentissage sur les parties non vocales à l'aide du formalisme d'adaptation

Nous illustrons maintenant le principe de réapprentissage complet sur les parties non vocales dans le cadre du formalisme d'adaptation présenté dans la partie II. Premièrement, cette illustration montrera que le formalisme d'adaptation proposé englobe également cette procédure de l'apprentissage simple. Deuxièmement, elle permettra de comprendre comment procéder dans des situations un peu plus complexes, ce qui donnera des idées de généralisations possibles. En particulier, cela apportera des réponses aux questions suivantes :

- L'astuce d'utiliser directement l'ensemble des trames non vocales $\{X(t)\}_{t \notin \mathbf{voc}}$ pour l'apprentissage marche proprement pour les MMG, car les dépendances entre les trames $X(t)$ ne sont pas modélisées (Fig. 2.5), et l'ordre des trames n'a donc pas d'importance. Tandis que pour les modèles modélisant ces dépendances temporelles, comme MMC par exemple, il n'est plus pertinent d'utiliser l'ensemble $\{X(t)\}_{t \notin \mathbf{voc}}$ pour l'apprentissage. En effet, dans l'ensemble $\{X(t)\}_{t \notin \mathbf{voc}}$ il y a des trames qui sont à côté et qui n'y sont pas dans le mélange X, ce qui peut mener à une estimation incorrecte de la matrice des transitions d'un MMC. Ainsi, on se demande comment procéder pour apprendre de manière pertinente un MMC ou un autre modèle avec des dépendances temporelles sur les parties non vocales.
- Une segmentation en parties vocales et non vocales contient des connaissances sur des segments temporels où la musique est présente seule. Imaginons maintenant une situation plus générale quand, au lieu des segments temporels, nous connaissons des régions temps - fréquence où la musique est seule. La question est, comment peut-on utiliser des régions temps - fréquence non vocales $\{X(t,f)\}_{(t,f) \notin \mathbf{voc}}$ pour apprendre le modèle de musique ? Ceci n'est pas évident. En effet, on sait apprendre sur des spectres entiers $\{X(t)\}_{t \notin \mathbf{voc}}$, mais comment apprendre sur des "bouts des spectres" $\{X(t,f)\}_{(t,f) \notin \mathbf{voc}}$?

Le formalisme d'adaptation peut apporter des réponses à ces questions.

Maintenant nous passons directement à l'explication de l'apprentissage sur des parties non vocales à l'aide du formalisme d'adaptation. Le fait que le modèle de musique $\tilde{\lambda}_m$ est appris (pas adapté) se traduit par un *a priori* uniforme non informatif sur le modèle $p(\tilde{\lambda}_m|\Lambda_m) \propto const$ qui ne dépend pas d'un modèle général Λ_m.

Supposons que le modèle de voix $\tilde{\lambda}_v$ n'est pas adapté, c'est-à-dire $\tilde{\lambda}_v = \Lambda_v$, et que c'est un modèle bimodal (à deux états 0 et 1) défini comme suit. Conditionnellement à l'état 0, la source de voix est nulle, c'est-à-dire $p(S_v(t)|q_v(t) = 0, \tilde{\lambda}_v) = \prod_f \delta(S_v(t,f))$, où $\delta(\cdot)$ est la distribution de Dirac. Conditionnellement à l'état 1, cette source suit une loi uniforme non informative $p(S_v(t)|q_v(t) = 1, \tilde{\lambda}_v) \propto const$, c'est-à-dire que dans cet état il n'y a pas de connaissances sur la source de voix. Ce modèle bimodal est représenté figure 9.4.

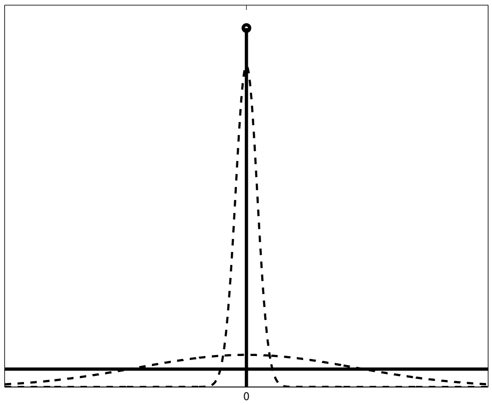


FIG. 9.4 – Modèle bimodal $\tilde{\lambda}_v$ composé de la distribution de Dirac et d'une loi uniforme non informative (ligne continue) et son approximation par un MMG à deux états λ_v^* avec une petite et une grande variance (pointillés).

Nous associons l'état 0 du modèle $\tilde{\lambda}_v$ aux parties non vocales et l'état 1 aux parties vocales. Ainsi, la connaissance d'une segmentation en parties vocales et non vocales est équivalente à la connaissance de la séquence d'états q_v. Cette séquence est considérée comme des informations auxiliaires qui peuvent être intégrées dans l'adaptation selon l'extension présentée dans la section 8.2. La figure 9.5 contient le réseau bayésien correspondant à l'apprentissage du modèle de musique sur des parties non vocales. Ce réseau est une simplification du réseau représenté figure 8.2 pour ce cas particulier.

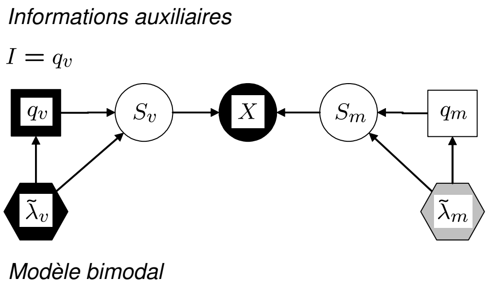


FIG. 9.5 – Réseau bayésien correspondant à l'apprentissage du modèle de musique sur les parties non vocales. Formes des noeuds : processus continus (ronds), processus discrets (carrés), modèles (hexagones). Coloration des noeuds : noeuds observés (noir), noeuds cachés estimés (gris), autres noeuds cachés (blanc).

L'application de l'algorithme EM présenté section 7.2 avec la séquence d'états q_v comme informations auxiliaires (Sec. 8.2) mène à la formule de réestimation des variances suivante :

$$\sigma_{m,j}^{2,(l+1)}(f) = \beta_j^{(l)} \frac{\sum_{t \notin \mathbf{voc}} |X(t,f)|^2 \gamma_j^{(l)}(t)}{\sum_{t \notin \mathbf{voc}} \gamma_j^{(l)}(t)} + (1 - \beta_j^{(l)}) \sigma_{m,j}^{2,(l)}(f), \tag{9.9}$$

avec $\beta_j^{(l)} = \frac{\zeta_j^{(l)}}{\zeta_j^{(l)} + \omega_{m,j}^{(l)} \#(\mathbf{voc})}$ ($\#(\mathbf{voc})$ est le nombre d'éléments de l'ensemble **voc**), $\zeta_j^{(l)} = \sum_{t \notin \mathbf{voc}} \gamma_j^{(l)}(t)$ et $\gamma_j^{(l)}(t)$ calculé selon (9.5).

La formule (9.9) peut être expliquée à partir de l'équation (7.10). Considérons un MMG à deux états $\lambda_v^* = \{\omega_{v,i}, \Sigma_{v,i}\}_{i=0,1}$. Remarquons que le modèle bimodal $\tilde{\lambda}_v$ peut être approché par ce MMG λ_v^*, si les variances $\sigma_{v,0}^2(f)$ sont assez petites et les variances $\sigma_{v,1}^2(f)$ sont assez grandes (voir Fig. 9.4). Enfin, le MMG λ_v^* tend vers $\tilde{\lambda}_v$ quand $\sigma_{v,0}^2(f) \to 0$ et $\sigma_{v,1}^2(f) \to \infty$. Considérons maintenant l'équation (7.10) en remplaçant les indices des sources 1 et 2 par m et v. Si dans cette équation $\sigma_{v,0}^2(f) \to 0$, on obtient $|X(t,f)|^2$, et si $\sigma_{v,1}^2(f) \to \infty$, on obtient $\sigma_{m,i}^2(f)$. Ceci explique la formule (9.9). Si pour la trame numéro t, le modèle $\tilde{\lambda}_v$ est dans l'état 0, c'est-à-dire $S_v(t) = \bar{0}$, on utilise le mélange $X(t) = S_m(t)$ pour estimer les paramètres du modèle $\tilde{\lambda}_m$. Si $\tilde{\lambda}_v$ est dans l'état 1, c'est-à-dire qu'il n'y a pas de connaissances sur $S_v(t)$, on garde les variances $\sigma_{m,j}^{2,(l)}(f)$ estimées à l'itération précédente.

La seule différence entre la formule de réestimation (9.9) et la formule (9.4) pour le critère MAP (9.3) est que les paramètres *a priori* $r_{m,j}^2(f)$ sont remplacés par des paramètres estimés à l'itération précédente $\sigma_{m,j}^{2,(l)}(f)$. La formule (9.9) est juste une version lissée de la formule (9.8) avec un coefficient de lissage $\beta_j^{(l)}$ qui dépend de l'état j et des observations X. Même si ces deux formules ne mènent pas exactement au même résultat, car avec la formule (9.9) les paramètres sont réestimés moins vite qu'avec la formule (9.8) et avec des vitesses différentes pour chaque gaussienne (facteurs $\beta_j^{(l)}$), ces formules sont assez proches. Ainsi, nous considérons ce développement théorique menant à la formule (9.9) comme une explication de l'apprentissage sur des parties non vocales (formule (9.8)) à l'aide du formalisme d'adaptation.

En se basant sur le formalisme d'adaptation et en faisant à peu près la même démarche, il est possible de développer proprement des procédures d'apprentissage (ou d'adaptation) dans des situations plus complexes, comme celles mentionnées au début de cette section.

Par exemple, imaginons qu'au lieu des segments temporels non vocaux, nous connaissons des régions temps - fréquences non vocales. Dans ce cas, il faut associer un état du modèle bimodal présenté figure 9.4 à chaque point temps - fréquence (t, f) plutôt qu'à chaque trame t. Plus précisément, il faut associer des Diracs aux points issus des régions non vocales (fiables pour l'apprentissage) et des lois uniformes aux points issus des régions vocales (non fiables pour l'apprentissage). Ceci mènera à l'apprentissage ou à l'adaptation des modèles dans le cadre de la "théorie des données manquantes" [Ghahramani-93] utilisée par exemple pour la reconnaissance de la parole dans un environnement bruité [Cooke-01]. D'ailleurs, Weiss et Ellis [Weiss-06] ont récemment proposé d'utiliser cette théorie pour la séparation de sources avec un seul capteur, mais pas dans le cadre de l'adaptation des modèles.

9.5 Adaptation des filtres et des gains de DSP

L'adaptation des filtres et des gains de DSP (Fig. 9.1) fait partie des techniques d'adaptation contrainte présentées section 8.1. Nous introduisons d'abord séparément l'adaptation d'un filtre et l'adaptation des gains de DSP. Nous expliquons ensuite comment faire l'adaptation des filtres et des gains de DSP conjointement.

9.5.1 Adaptation d'un filtre

Une des techniques d'adaptation contrainte que nous proposons est l'adaptation d'un filtre. Cette adaptation rend la modélisation invariante à toute variation entre enregistrements qui peut être représentée par un filtre linéaire global, par exemple la variation de l'acoustique de la salle, de certaines caractéristiques du microphone etc. Il est supposé (par exemple pour le modèle de la voix) que la discordance entre le modèle général Λ_v et le modèle adapté λ_v est un filtre linéaire h_v. Autrement dit, chaque source modélisée par λ_v est considérée comme le résultat du filtrage avec le filtre h_v d'une autre source modélisée par Λ_v. Le filtre h_v est supposé inconnu et le but d'adaptation est de l'estimer.

Soit $H_v = [H_v(f)]_f$, la TFCT de la réponse impulsionnelle du filtre h_v. Dans le domaine de la TFCT, le filtrage peut être réalisé approximativement en multipliant chaque spectre à court terme par H_v élément par élément. Les DSP du modèle adapté λ_v et celles du modèle général Λ_v sont donc reliées comme suit :

$$\sigma_{v,i}^2(f) = |H_v(f)|^2 r_{v,i}^2(f), \quad f = 1, \ldots, F, \tag{9.10}$$

Ainsi, en introduisant la matrice diagonale $\mathcal{H}_v \triangleq \mathrm{diag}\,[\mathcal{H}_v(f)]_f$ avec $\mathcal{H}_v(f) \triangleq |H_v(f)|^2$ (par la suite, on va appeler *filtre* cette matrice $\mathcal{H}_v$), on peut écrire la relation suivante entre les modèles λ_v et $\Lambda_v = \{u_{v,i}, R_{v,i}\}_i$:

$$\lambda_v = \mathcal{H}_v \Lambda_v \triangleq \{u_{v,i}, \mathcal{H}_v R_{v,i}\}_i \tag{9.11}$$

Au sein des notations de la section 8.1 le filtre $\mathcal{H}_v$ joue le rôle des paramètres libres C_v et $\mathcal{H}_v \Lambda_v$ joue le rôle de la déformation paramétrique $\Psi_v(C_v, \Lambda_v)$. Pour estimer le filtre $\mathcal{H}_v$, le critère suivant correspondant au critère (8.1), est utilisé :

$$\mathcal{H}_v = \arg\max_{\mathcal{H}'_v} p(X|\lambda'_v = \mathcal{H}'_v \Lambda_v, \tilde{\lambda}_m) \tag{9.12}$$

Remarquons que dans ce critère (9.12) le modèle de musique adapté acoustiquement $\tilde{\lambda}_m$ est utilisé au lieu d'un modèle général Λ_m, car l'adaptation est faite en deux étapes (voir Fig. 9.1). Notons également qu'il n'y a pas de contraintes supplémentaires sur le filtre $\mathcal{H}_v$, ou bien il y a un *a priori* uniforme non informatif ($p(\mathcal{H}_v|\Lambda_v) \propto const$).

Pour une meilleure compréhension, nous avons représenté le modèle général Λ_v et le modèle adapté λ_v par des *matrices des DSP*, c'est-à-dire des matrices dont les colonnes sont des DSP (diagonales des matrices de covariances) (voir Fig. 9.6 (A) et (B)). On s'aperçoit qu'au sein d'une telle représentation, l'adaptation d'un filtre correspond à la multiplication de la matrice des DSP du modèle général $\mathbf{R}_v$ par la matrice diagonale $\mathcal{H}_v$ à gauche.

L'application de l'algorithme EM (8.2), (8.3) pour le critère (9.12) mène à la formule de réestimation suivante (voir Annexe B.3.1 pour une démonstration) :

$$\mathcal{H}_v^{(l+1)}(f) = \frac{1}{T} \sum_{i=1}^{Q_v} \frac{\mathbf{t}_{v,i}^{2,(l)}(f)}{r_{v,i}^2(f)} \tag{9.13}$$

où les espérances $\mathbf{t}_{v,i}^{2,(l)}(f)$ sont calculées à l'aide de l'algorithme 4 (Eq. (7.12)), conditionnellement aux modèles $\lambda_v^{(l)} = \mathcal{H}_v^{(l)}\Lambda_v$ et $\lambda_m^{(l)} = \tilde{\lambda}_m$.

Enfin, il faut remarquer que l'adaptation d'un filtre peut être vue comme l'adaptation MLLR contrainte. En effet, la technique MLLR [Leggetter-95, Gales-96] consiste à adapter une transformation affine de l'espace des paramètres, tandis que pour l'adaptation d'un filtre, seules les dilatations et les contractions de l'espace le long des axes sont autorisées, car la matrice $\mathcal{H}_v$ est diagonale.

9.5.2 Adaptation des gains de DSP

Chaque état d'un MMG spectral est décrit par une DSP et correspond à un événement sonore particulier, par exemple une note ou un accord. Les énergies relatives moyennes de ces événements sonores varient entre les enregistrements. Par exemple, pour un enregistrement, la note *la* peut être jouée plus fort en moyenne que la note *ré* et vice versa pour un autre enregistrement. Pour prendre en compte cette variation de l'énergie, un gain réel positif $g_{v,i} > 0$ est associé à chacune des DSP $[r_{v,i}^2(f)]_f$ du modèle Λ_v. Ce gain appelé *gain de DSP* correspond à l'énergie de l'événement sonore représenté par cette DSP. Chaque DSP étant la diagonale de la matrice de covariance correspondante $R_{v,i}$, ceci se traduit par la multiplication de chaque matrice $R_{v,i}$ par le gain $g_{v,i}$. Ainsi, la technique d'adaptation des gains de DSP consiste à rechercher le modèle adapté λ_v sous la forme suivante :

$$\lambda_v = g_v \bullet \Lambda_v \triangleq \{u_{v,i}, g_{v,i}R_{v,i}\}_i, \tag{9.14}$$

où $g_v = [g_{v,i}]_i$ est le vecteur des gains de DSP et le symbole "$\bullet$" signifie une opération non standard utilisée ici pour différencier l'application des gains de l'application d'un filtre (9.11).

Par rapport à l'adaptation d'un filtre (9.11), où le but est d'adapter l'énergie de chaque bande

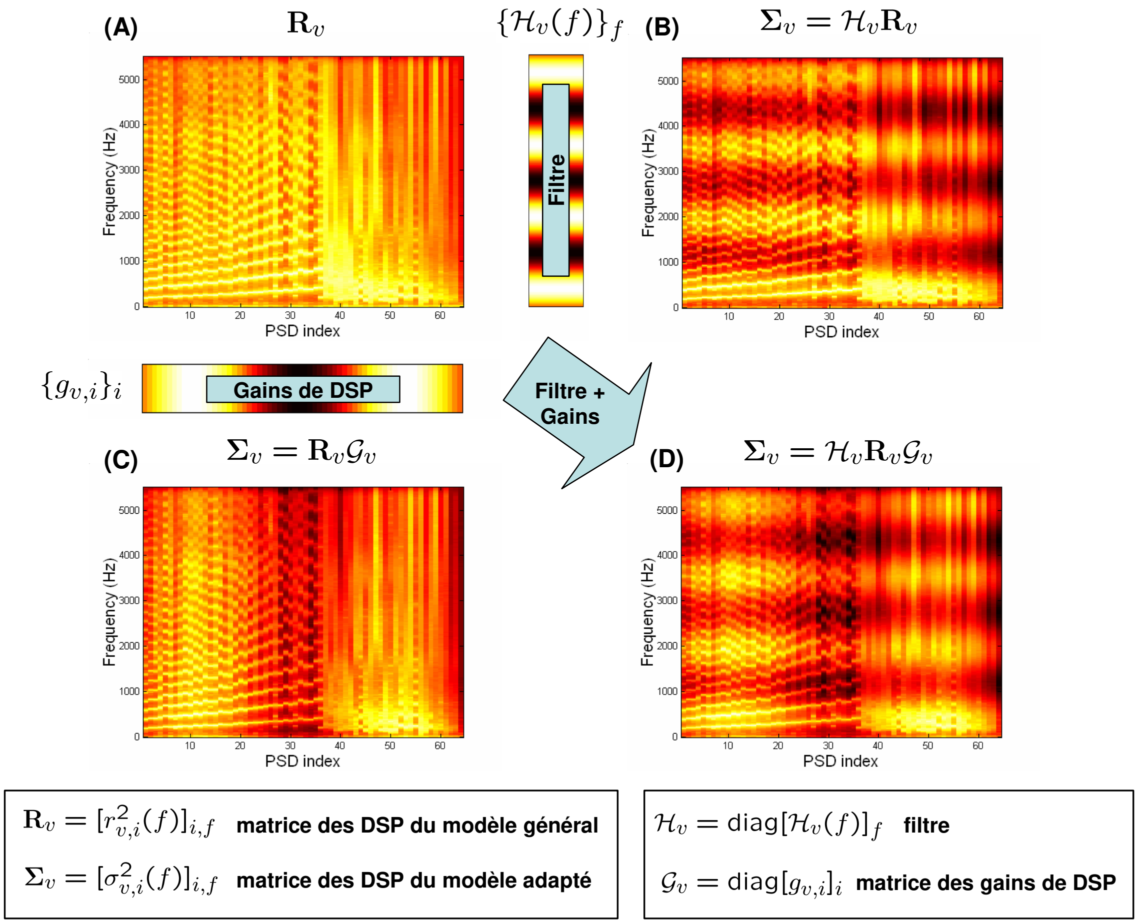


FIG. 9.6 – Interprétation matricielle de l'adaptation d'un filtre et des gains de DSP. Les modèles $\Lambda_v = \{u_{v,i}, R_{v,i}\}_i$ et $\lambda_v = \{\omega_{v,i}, \Sigma_{v,i}\}_i$ sont représentés par leurs matrices des DSP $\mathbf{R}_v = [r^2_{v,i}(f)]_{i,f}$ et $\mathbf{\Sigma}_v = [\sigma^2_{v,i}(f)]_{i,f}$. Le filtre et les gains de DSP sont représentés par des matrices diagonales $\mathcal{H}_v = \mathrm{diag}[\mathcal{H}_v(f)]_f$ et $\mathcal{G}_v = \mathrm{diag}[g_{v,i}]_i$. **(A)** : Modèle de voix général à 64 états $\mathbf{R}_v$. **(B)** : Adaptation d'un filtre (multiplication de $\mathbf{R}_v$ par $\mathcal{H}_v$ à gauche). **(C)** : Adaptation des gains de DSP (multiplication de $\mathbf{R}_v$ par $\mathcal{G}_v$ à droite). **(D)** : Adaptation conjointe d'un filtre et des gains de DSP (multiplication de $\mathbf{R}_v$ par $\mathcal{H}_v$ à gauche et par $\mathcal{G}_v$ à droite).

fréquentielle f au signal traité, le but de l'adaptation des gains de DSP est d'adapter l'énergie de chaque DSP i. Cette différence est également représentée sur la figure 9.6. On voit que l'adaptation des gains de DSP correspond à la multiplication de la matrice des DSP du modèle général $\mathbf{R}_v$ par la matrice diagonale des gains de DSP $\mathcal{G}_v = \text{diag}[g_{v,i}]_i$ à droite (Fig. 9.6 (A) et (C)).

Il est aussi important de comparer cette technique avec celle à base de facteurs de gains [Benaroya-06] (Sec. 2.2.4). Dans [Benaroya-06], il est proposé d'estimer un gain pour chaque trame, tandis que l'adaptation des gains de DSP consiste à estimer un gain pour toutes les trames associées à l'état i.

Ensuite, l'explication est analogue à celle de section 9.5.1. Les gains de DSP g_v jouent le rôle des paramètres libres et le critère suivant est utilisé pour les estimer :

$$g_v = \arg\max_{g'_v} p(X|\lambda'_v = g'_v \bullet \Lambda_v, \tilde{\lambda}_m) \tag{9.15}$$

L'application de l'algorithme EM (8.2), (8.3) mène à la formule de réestimation suivante (voir Annexe B.3.2 pour une démonstration) :

$$g_{v,i}^{(l+1)} = \frac{1}{F \cdot \mathbf{t}_{v,i}^{0,(l)}} \sum_{f=1}^{F} \frac{\mathbf{t}_{v,i}^{2,(l)}(f)}{r_{v,i}^2(f)} \tag{9.16}$$

avec des espérances $\mathbf{t}_{v,i}^{0,(l)}$ et $\mathbf{t}_{v,i}^{2,(l)}(f)$ calculées à l'aide de l'algorithme 4 (Eqs. (7.11) et (7.12)), conditionnellement aux modèles $\lambda_v^{(l)} = g_v^{(l)} \bullet \Lambda_v$ et $\lambda_m^{(l)} = \tilde{\lambda}_m$.

9.5.3 Adaptation conjointe des filtres et des gains de DSP

Ici, nous expliquons comment adapter conjointement les filtres et les gains de DSP pour les deux modèles, c'est-à-dire le modèle de voix et le modèle de musique. Les modèles adaptés de voix et de musique sont recherchés sous la forme $\lambda_v = g_v \bullet \mathcal{H}_v \Lambda_v$ et $\lambda_m = g_m \bullet \mathcal{H}_m \tilde{\lambda}_m$, où $\mathcal{H}_m$ et g_m sont respectivement un filtre et des gains de DSP du modèle de musique λ_m. Pour adapter tous ces paramètres, le critère suivant est utilisé :

$$(\mathcal{H}_v, g_v, \mathcal{H}_m, g_m) = \arg\max_{\mathcal{H}'_v, g'_v, \mathcal{H}'_m, g'_m} p(X|\lambda'_v = g'_v \bullet \mathcal{H}'_v \Lambda_v, \lambda'_m = g'_m \bullet \mathcal{H}'_m \tilde{\lambda}_m) \tag{9.17}$$

Pour le modèle général de voix Λ_v, cette adaptation est très importante car ce modèle n'a pas encore été adapté. Pour le modèle de musique $\tilde{\lambda}_m$, l'adaptation est moins cruciale car il est déjà adapté sur les parties non vocales. Cependant, cette adaptation peut potentiellement améliorer le modèle de musique puisqu'il y a toujours une petite discordance entre la musique dans les parties non vocales, où le modèle $\tilde{\lambda}_m$ est adapté, et la musique dans les parties vocales. L'adaptation d'un filtre et des gains de DSP pour le modèle de voix est également visualisée figure 9.6 (A) et (D).

En essayant comme avant d'appliquer l'algorithme EM (8.2), (8.3) pour optimiser le critère (9.17), on s'aperçoit qu'il est difficile de résoudre l'étape M (8.3) conjointement pour les filtres $\{\mathcal{H}_v, \mathcal{H}_m\}$ et les gains de DSP $\{g_v, g_m\}$ (voir Annexe B.3.3). Une solution à ce problème serait d'utiliser l'algorithme SAGE (*Space-Alternating Generalized EM*) [Fessler-94, McLachlan-97] en alternant les itérations de EM entre $\{\mathcal{H}_v, \mathcal{H}_m\}$ et $\{g_v, g_m\}$. Un des inconvénients de cette approche par rapport à l'algorithme EM est que l'on a besoin de deux itérations de EM au lieu d'une seule pour réestimer une fois tous les paramètres $\{\mathcal{H}_v, g_v, \mathcal{H}_m, g_m\}$. Par conséquent, la complexité calculatoire double. En analysant séparément les complexités calculatoires des étapes E (8.2) et M (8.3), nous avons remarqué que la complexité de l'étape M est souvent négligeable par rapport à celle de l'étape E. En effet, la complexité de E (le calcul des espérances des statistiques naturelles) est de l'ordre $O(TFQ_1Q_2)$ (voir (7.11) et (7.12)) et la complexité de M (mise à jour des paramètres) est de l'ordre $O(F(Q_1 + Q_2))$ (voir par exemple (9.13) et (9.16)). Ainsi, pour ne pas doubler la complexité calculatoire, au lieu de l'algorithme SAGE, nous proposons de faire à chaque itération une étape E suivie par plusieurs étapes M en alternant entre le mise à jour des filtres $\{\mathcal{H}_v, \mathcal{H}_m\}$ et des gains de DSP $\{g_v, g_m\}$ (voir Annexe B.3.3 pour plus d'explications). L'algorithme 5 résume cette proposition.

9.6 Conclusion

Nous avons présenté un système de séparation voix / musique basé sur l'adaptation de modèles. Le module d'adaptation de ce système consiste en trois blocs principaux :

1. la segmentation en parties vocales et non vocales (Sec. 9.3),
2. l'adaptation acoustique du modèle de musique sur les parties non vocales (Sec. 9.4),
3. l'adaptation du modèle de musique $\tilde{\lambda}_m$ et du modèle général de voix Λ_v sur toute la chanson en utilisant la technique d'adaptation des filtres et des gains de DSP (Sec. 9.5).

Ces différentes adaptations peuvent être formulées comme des cas spécifiques du formalisme présenté dans la partie II de cette thèse et, bien que résultant de considérations hétérogènes, les algorithmes correspondants dérivent de l'algorithme EM générique présenté dans le chapitre 7.

Le bloc d'adaptation acoustique du modèle de musique sur les parties non vocales a été partiellement évalué dans ce chapitre, et nous avons choisi de complètement réapprendre ce modèle en perdant l'attache au modèle de musique général. Les autres blocs seront évalués dans la partie IV.

Algorithme 5 Adaptation conjointe des filtres et des gains de DSP pour les modèles $\Lambda_v = \{u_{v,i}, R_{v,i}\}_i$ et $\tilde{\lambda}_m = \{\tilde{\omega}_{m,j}, \tilde{\Sigma}_{m,j}\}_i$.

---

1. **Etape E** : Calculer les espérances $\left\{\mathbf{t}_{v,i}^{0,(l)}, \{\mathbf{t}_{v,i}^{2,(l)}(f)\}_f\right\}_i$ et $\left\{\mathbf{t}_{m,j}^{0,(l)}, \{\mathbf{t}_{m,j}^{2,(l)}(f)\}_f\right\}_j$ des statistiques naturelles, conditionnellement aux modèles $\lambda_v^{(l)} = g_v^{(l)} \bullet \mathcal{H}_v^{(l)} \Lambda_v$ et $\lambda_m^{(l)} = g_m^{(l)} \bullet \mathcal{H}_m^{(l)} \tilde{\lambda}_m$ à l'aide de l'algorithme 4.
2. **Etape M** : Mettre à jour les paramètres.
   (a) Initialiser $g_v^{[0]} = g_v^{(l)}$, $g_m^{[0]} = g_m^{(l)}$,
   (b) Effectuer W étapes de maximisation en alternant entre $\{\mathcal{H}_v, \mathcal{H}_m\}$ et $\{g_v, g_m\}$, pour $w = 1, 2, \ldots, W$:

$$\mathcal{H}_v^{[w]}(f) = \frac{1}{T}\sum_{i=1}^{Q_v} \frac{\mathbf{t}_{v,i}^{2,(l)}(f)}{g_{v,i}^{[w-1]} r_{v,i}^2(f)}, \tag{9.18}$$

$$g_{v,i}^{[w]} = \frac{1}{F \cdot \mathbf{t}_{v,i}^{0,(l)}} \sum_{f=1}^{F} \frac{\mathbf{t}_{v,i}^{2,(l)}(f)}{\mathcal{H}_v^{[w]}(f) r_{v,i}^2(f)}, \tag{9.19}$$

$$\mathcal{H}_m^{[w]}(f) = \frac{1}{T}\sum_{j=1}^{Q_m} \frac{\mathbf{t}_{m,j}^{2,(l)}(f)}{g_{m,j}^{[w-1]} \tilde{\sigma}_{m,j}^2(f)}, \tag{9.20}$$

$$g_{m,j}^{[w]} = \frac{1}{F \cdot \mathbf{t}_{m,j}^{0,(l)}} \sum_{f=1}^{F} \frac{\mathbf{t}_{m,j}^{2,(l)}(f)}{\mathcal{H}_m^{[w]}(f) \tilde{\sigma}_{m,j}^2(f)} \tag{9.21}$$

   (c) Poser $\mathcal{H}_v^{(l+1)} = \mathcal{H}_v^{[W]}$, $g_v^{(l+1)} = g_v^{[W]}$, $\mathcal{H}_m^{(l+1)} = \mathcal{H}_m^{[W]}$, $g_m^{(l+1)} = g_m^{[W]}$,

---

Avant de passer à cette évaluation, nous mettons en évidence dans le chapitre suivant que l'adaptation peut nécessiter des changements dans la procédure de l'apprentissage des modèles généraux.

Chapitre 10

Intégration de l'adaptation des filtres et des gains lors de l'apprentissage du modèle général

Après avoir présenté les grandes lignes de l'approche utilisée en pratique, nous abordons les questions suivantes. En prenant en compte le fait que les modèles généraux sont adaptés, la procédure de leurs apprentissage devrait-elle changer ? Si oui, comment ?

10.1 Apprentissage du modèle général à filtres adaptés

Considérons d'abord une technique particulière d'adaptation, notamment l'adaptation d'un filtre (Sec. 9.5.1) pour le modèle général de voix Λ_v. Selon cette technique, un filtre est adapté pour chaque nouvel enregistrement à séparer. Par conséquent, le modèle général est invariant par rapport à un filtre pour chaque enregistrement. Cependant, cette invariance n'est pas prise en compte pendant l'apprentissage du modèle. Ceci pose le problème suivant. Supposons que dans la base d'entraînement Y_v il y a une grande variabilité entre les morceaux des conditions d'enregistrement qui peuvent être modélisées par des filtres linéaires (par ex. les acoustiques des salles, les caractéristique des microphones). Dans ce cas, les états (les DSP) du modèle général vont être gaspillés pour modéliser ces différentes conditions d'enregistrement, alors que de toute manière, l'adaptation d'un filtre va normaliser le modèle par rapport à ces conditions.

Nous proposons donc de modifier l'apprentissage en adaptant également un filtre pour chaque enregistrement de la base d'entraînement. Autrement dit, nous essayons de rendre la procédure d'apprentissage des modèles généraux homogène à leur utilisation.

Supposons que la base d'entraînement Y_v du modèle général de voix Λ_v soit composée de plusieurs enregistrements, soit $Y_v = \{E_{v,z}\}_z$, où $E_{v,z}$ est la TFCT du z-ème enregistrement.

Au lieu d'estimer comme avant un modèle général Λ_v modélisant les enregistrements $E_{v,z}$, nous proposons d'estimer un modèle général $\Lambda_v^{\mathcal{H}}$ de telle manière que chaque enregistrement $E_{v,z}$ soit modélisé par un modèle $\lambda_{v,z} = \mathcal{H}_{v,z}\Lambda_v^{\mathcal{H}}$, où $\mathcal{H}_{v,z}$ est un filtre estimé spécialement pour cet enregistrement.

Soit $\mathbf{H}_v = \{\mathcal{H}_{v,z}\}_z$ un ensemble de filtres inconnus (un filtre par enregistrement). Au lieu de chercher Λ_v en optimisant le critère du MV (2.12) la nouvelle procédure d'apprentissage consiste à estimer conjointement les filtres $\mathbf{H}_v$ et le modèle $\Lambda_v^{\mathcal{H}}$ avec le critère du MV suivant :

$$(\mathbf{H}_v, \Lambda_v^{\mathcal{H}}) = \arg\max_{\mathbf{H}'_v,\Lambda'_v} p(Y_v|\mathbf{H}'_v, \Lambda'_v) \triangleq \arg\max_{\mathbf{H}'_v,\Lambda'_v} \prod_z p(E_{v,z}|\lambda'_{v,z} = \mathcal{H}'_{v,z}\Lambda'_v) \tag{10.1}$$

En essayant appliquer directement l'algorithme EM (Annexe A.3) pour optimiser le critère (10.1) avec les données observées $\mathcal{X} = Y_v$, les données complètes $\mathcal{Z} = \{Y_v, q_v\}$ et les paramètres estimés $\theta = \{\mathbf{H}_v, \Lambda_v\}$ on rencontre exactement le même problème que dans la section 9.5.3. L'étape M n'est pas facile à résoudre conjointement pour $\mathbf{H}_v$ et Λ_v. Ainsi, il est possible de s'en sortir à l'aide de l'algorithme SAGE [Fessler-94, McLachlan-97] ou bien en alternant l'étape M, comme il est proposé dans la section 9.5.3. Cependant, par rapport à l'adaptation, la complexité calculatoire n'est pas si cruciale pour l'apprentissage du modèle général, car il est effectué hors ligne (Fig. 6.1). Ainsi, nous avons choisi d'utiliser l'algorithme SAGE.

Les formules de réestimation de l'algorithme SAGE pour l'optimisation du critère (10.1) sont représentées dans l'algorithme 6. Chaque itération de cet algorithme consiste en deux itérations de EM. La première itération est pour mettre à jour l'ensemble des filtres $\mathbf{H}_v$ avec le modèle Λ_v fixé, la deuxième itération est pour mettre à jour le modèle Λ_v avec l'ensemble des filtres $\mathbf{H}_v$ fixé.

10.2 Illustration expérimentale

Pour mesurer l'apport de cette nouvelle procédure d'apprentissage à filtres adaptés par rapport à l'apprentissage conventionnel, nous avons recours à une petite expérimentation.

Les données de test et d'entraînement sont décrites dans la section 4.4. La segmentation manuelle en parties vocales et non vocales est utilisée pour l'instant dans le module d'adaptation (Fig. 9.1). La séparation des chansons de la base d'évaluation est faite avec des modèles de voix et de musique à 32 états chacun ($Q_v = Q_m = 32$) dans les configurations suivantes :

1. **Modèles généraux** Λ_v et Λ_m appris sur les données d'entraînement en utilisant l'algorithme 2.
2. **Modèles adaptés acoustiquement** : modèle général de voix Λ_v et modèle de musique adapté acoustiquement $\tilde{\lambda}_m$ (appris sur les parties non vocales, car nous avons décidé de

Algorithme 6 Algorithme SAGE pour l'apprentissage du modèle général de voix à filtres adaptés, c'est-à-dire l'estimation conjointe de l'ensemble des filtres $\mathbf{H}_v = \{\mathcal{H}_{v,z}\}_z$ et du modèle général $\Lambda_v^{\mathcal{H}} = \{u_{v,i}, R_{v,i}\}_i$ à l'aide du critère (10.1) à partir des données d'entraînement $Y_v = \{E_{v,z}\}_z$.

1. Première itération de EM ($\mathbf{H}_v$ est mis à jour, $\Lambda_v = \Lambda_v^{(l)}$ est fixé) :
   (a) Pour chaque enregistrement $E_{v,z}$ calculer les poids $\gamma_{z,i}^{(l)}(t)$ satisfaisant $\sum_i \gamma_{z,i}^{(l)}(t) = 1$ et
   $$\gamma_{z,i}^{(l)}(t) \propto u_{v,i}^{(l)} N_C(E_{v,z}(t); \bar{0}, \mathcal{H}_{v,z}^{(l)} R_{v,i}^{(l)}), \tag{10.2}$$
   où $N_C(\cdot)$ est défini selon (A.2).
   (b) Mettre à jour l'ensemble des filtres $\mathbf{H}_v$:
   $$\mathcal{H}_{v,z}^{(l+1)}(f) = \frac{1}{T_z} \sum_{t=1}^{T_z} \sum_i \frac{|E_{v,z}(t,f)|^2}{r_{v,i}^{2,(l)}(f)} \gamma_{z,i}^{(l)}(t), \tag{10.3}$$
   où T_z dénote le nombre de trames dans la TFCT de z-ème enregistrement $E_{v,z}$.
2. Deuxième itération de EM (Λ_v est mis à jour, $\mathbf{H}_v = \mathbf{H}_v^{(l+1)}$ est fixé) :
   (a) Recalculer les poids $\gamma_{z,i}^{(l+0.5)}(t)$ satisfaisants $\sum_i \gamma_{z,i}^{(l+0.5)}(t) = 1$ et
   $$\gamma_{z,i}^{(l+0.5)}(t) \propto u_{v,i}^{(l)} N_C(E_{v,z}(t); \bar{0}, \mathcal{H}_{v,z}^{(l+1)} R_{v,i}^{(l)}) \tag{10.4}$$
   (b) Mettre à jour les poids de gaussiennes $u_{v,i}$:
   $$u_{v,i}^{(l+1)} = \frac{1}{\sum_z T_z} \sum_z \sum_{t=1}^{T_z} \gamma_{z,i}^{(l+0.5)}(t), \tag{10.5}$$
   (c) Mettre à jour les matrices de covariances $R_{v,i}$:
   $$r_{v,i}^{2,(l+1)}(f) = \frac{\sum_z \sum_{t=1}^{T_z} \gamma_{z,i}^{(l+0.5)}(t) \frac{|E_{v,z}(t,f)|^2}{\mathcal{H}_{v,z}^{(l+1)}(f)}}{\sum_z \sum_{t=1}^{T_z} \gamma_{z,i}^{(l+0.5)}(t)}, \tag{10.6}$$

ne pas garder d'attache au modèle général Λ_m).

3. **Modèles adaptés (à partir de Λ_v)** : modèle de musique adapté acoustiquement $\tilde{\lambda}_m$ et modèles de voix à filtre adapté $\lambda_v = \mathcal{H}_v \Lambda_v$ (Sec. 9.5.1) obtenu à partir du modèle général Λ_v (appris en utilisant l'algorithme 2).
4. **Modèles adaptés (à partir de $\Lambda_v^{\mathcal{H}}$)** : idem, sauf que le modèle de voix à filtre adapté $\lambda_v = \mathcal{H}_v \Lambda_v^{\mathcal{H}}$ est obtenu à partir du modèle général de voix à filtres adaptés $\Lambda_v^{\mathcal{H}}$ (appris en utilisant l'algorithme 6).

Types de modèles	Modèle de voix	Modèle de musique	RSDN moyen
Modèles généraux	Λ_v	Λ_m	5.4
Modèles adaptés acoustiquement	Λ_v	$\tilde{\lambda}_m$	11.3
Modèles adaptés (à partir de Λ_v)	$\lambda_v = \mathcal{H}_v \Lambda_v$	$\tilde{\lambda}_m$	12.1
Modèles adaptés (à partir de $\Lambda_v^{\mathcal{H}}$)	$\lambda_v = \mathcal{H}_v \Lambda_v^{\mathcal{H}}$	$\tilde{\lambda}_m$	**12.3**

Tab. 10.1 – Apport de la procédure d'apprentissage à filtres adaptés. La performance de séparation (RSDN moyen) est affichée pour chaque type de modèles testés.

Les résultats obtenus sont résumés dans le tableau 10.1. Par rapport aux modèles adaptés acoustiquement (Λ_v et $\tilde{\lambda}_m$), l'adaptation du filtre améliore les performances moyennes de 0.8 dB et l'apprentissage à filtres adaptés les améliore encore de 0.2 dB. Ainsi, on voit que c'est surtout l'adaptation du filtre du modèle de voix qui permet de gagner en performance, et non pas la nouvelle procédure d'apprentissage du modèle général. Ceci est probablement lié au fait que le modèle général de voix n'est pas très grand (il est composé seulement de 32 DSP). Ainsi, les états du modèle ne sont pas encore gaspillés pour la modélisation des différentes conditions d'enregistrement, et cette normalisation par rapport aux filtres n'est pas vraiment nécessaire lors de son apprentissage.

Toutefois, cette expérimentation ne remet pas en cause la nouvelle procédure d'apprentissage proposée. L'adaptation des filtres lors de l'apprentissage n'améliore pas significativement les performances dans notre cas, mais il ne les dégrade pas non plus. Nous allons donc utiliser par la suite cet apprentissage à filtres adaptés pour le modèle général de voix.

10.3 Apprentissage prenant en compte l'adaptation contrainte

De manière analogue, il est possible d'intégrer dans la procédure d'apprentissage l'adaptation des gains de DSP (Sec. 9.5.2), ainsi que l'adaptation conjointe d'un filtre et des gains de DSP (Sec. 9.5.3). Toutes ces méthodes font partie des techniques d'adaptation contrainte (Sec. 8.1). Nous pouvons ainsi représenter les procédures apprentissage prenant en compte l'adaptation des différents paramètres libres (filtres, gains de DSP, etc.) par un critère d'apprentissage correspondant au critère MAP d'adaptation contrainte (8.1). Ce critère est une généralisation du critère (10.1) et s'écrit comme suit :

$$(\mathbf{C}_k, \Lambda_k^C) = \arg \max_{\mathbf{C}'_k, \Lambda'_k} \prod_z p\left(E_{k,z} \left| \lambda'_{k,z} = \Psi_k(C'_{k,z}, \Lambda'_k)\right.\right) p(C'_{k,z}|\Lambda'_k), \tag{10.7}$$

où $Y_k = \{E_{k,z}\}_z$ sont les données d'entraînement, $C_{k,z}$, $\Psi_k(C_{k,z}, \Lambda_k)$ et $p(C_{k,z}|\Lambda_k)$ sont les paramètres libres, la déformation paramétrique et la loi *a priori* définis section 8.1 et $\mathbf{C}_k = \{C_{k,z}\}_z$ est l'ensemble des paramètres libres.

Le réseau bayésien pour cette procédure d'adaptation est représenté figure 10.1. Si la loi *a priori* sur les paramètres libres est uniforme non informative $p(C_{k,z}|\Lambda_k) \propto const$, ce qui est le cas pour l'adaptation d'un filtre et des gains de DSP, les $C_{k,z}$ ne devraient pas être reliés avec Λ_k par des flèches. C'est pour cela que ces flèches sont représentées figure 10.1 par des lignes pointillées.

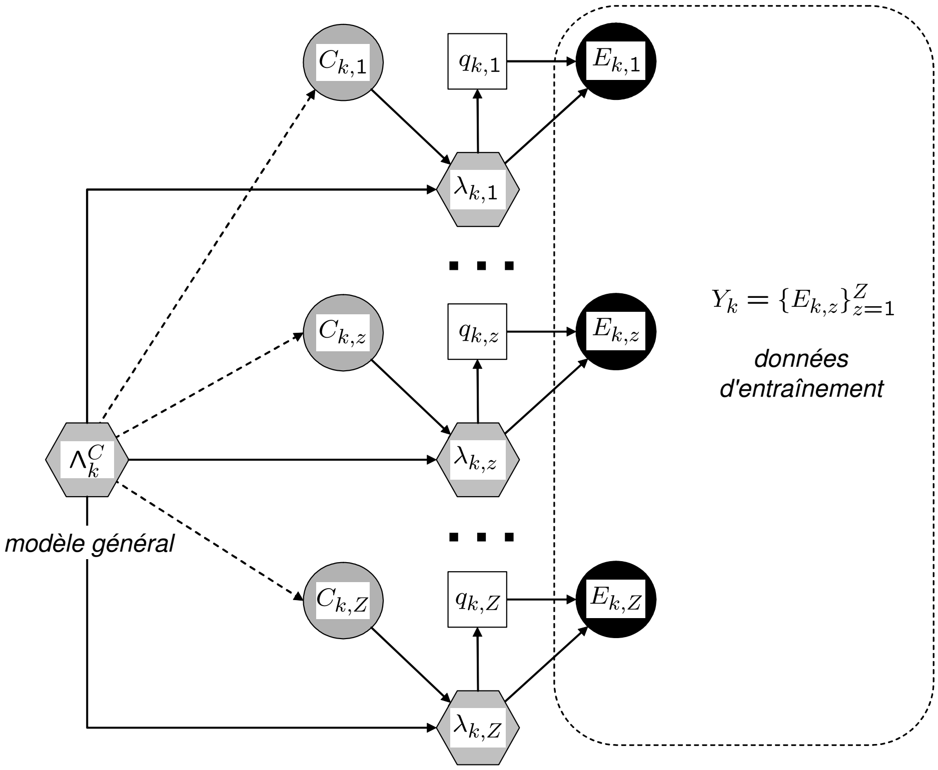


FIG. 10.1 – Réseau bayésien correspondant à l'apprentissage prenant en compte l'adaptation contrainte (10.7). Formes des noeuds : processus continus (ronds), processus discrets (carrés), modèles (hexagones). Coloration des noeuds : noeuds observés (noir), noeuds cachés estimés (gris), autres noeuds cachés (blanc).

10.4 Conclusion

Sachant que le modèle général est adapté pour chaque enregistrement, nous avons étudié dans ce chapitre s'il existe une procédure de son apprentissage plus pertinente que l'apprentissage conventionnel au maximum de vraisemblance (2.12) présenté section 2.3.2. La réponse est oui. En effet, sachant que pour chaque nouvel enregistrement le modèle est adapté, il sera plus pertinent de chercher un modèle qui, une fois adapté, modélise bien chaque enregistrement de la base d'entraînement, plutôt qu'un modèle qui modélise ces enregistrements tout simplement.

Ainsi, nous présentons une procédure alternative d'apprentissage qui prend en compte l'adaptation. Cette procédure est d'abord introduite pour une technique particulière de l'adaptation d'un filtre sous le nom d'*apprentissage à filtres adaptés*. Il est expliqué comment mettre en oeuvre cet apprentissage à l'aide de l'algorithme SAGE.

Une évaluation expérimentale montre que, par rapport à l'apprentissage conventionnel, l'apprentissage à filtres adaptés n'améliore pas significativement les performances de séparation. Nous pensons que ceci est parce que le modèle général étudié n'est pas très grand. Ainsi, la variation des paramètres adaptables (des filtres) n'est pas représentée par ce modèle et, par conséquent, la normalisation par rapport à ces paramètres lors de l'apprentissage n'est pas une étape cruciale. Cependant, cela ne remet pas en cause notre proposition dont le but est de rendre l'apprentissage de modèles conforme à leur utilisation.

Enfin, nous généralisons la procédure d'apprentissage prenant en compte l'adaptation pour d'autres techniques d'adaptation contrainte (Sec. 8.1), comme par exemple l'adaptation des gains de DSP.

La parie suivante est entièrement consacrée à une évaluation complète du système de séparation voix / musique présenté dans le chapitre 9.

Quatrième partie

Evaluation du système de séparation voix / musique

Chapitre 11

Segmentation en parties vocales et non vocales

Dans ce chapitre, le module de segmentation automatique en parties vocales et non vocales présenté dans la section 9.3 est évalué indépendamment du module d'adaptation résumé sur la figure 9.1. Cependant, un des paramètres du module de segmentation, précisément le seuil de décision η, sera réglé dans le chapitre suivant via la performance de séparation, quand ce module sera intégré dans le module d'adaptation.

11.1 Description des données expérimentales pour la segmentation

La base d'entraînement des modèles des parties vocales Γ_V et non vocales Γ_N (Sec. 9.3) consiste en 52 chansons populaires. Pour évaluer la performance de segmentation, 21 chansons sont utilisées, parmi lesquelles les six chansons de la base d'évaluation pour la séparation (Sec. 4.4).

Les oeuvres d'un même artiste n'interviennent jamais à la fois dans la base d'entraînement et celle d'évaluation.

Tous ces enregistrements de chansons sont en mono, échantillonnés à 11025 Hz et segmentés à la main en parties vocales et non vocales.

11.2 Protocole expérimental

Les modèles Γ_V et Γ_N sont appris à partir de la base d'entraînement à l'aide de l'algorithme 3 (rappelons que les MMG Γ_V et Γ_N utilisés pour la segmentation ont la même structure que les MMG log spectraux présentés dans la section 2.4.2). La base d'évaluation est ensuite segmentée automatiquement en parties vocales et non vocales par le module de segmentation représenté figure 9.2. Enfin, la performance est calculée en comparant les segmentations automatiques obtenues avec les segmentations manuelles correspondantes. La mesure de performance utilisée est présentée ci-dessous.

11.3 Mesure de performance

Comme dans l'article [Tsai-04a], pour évaluer la performance de la segmentation en parties vocales et non vocales, les courbes DET (*Detection Error Tradeoff*) [Martin-97] sont utilisées. Pour un seuil de décision η donné (voir équations (9.1) et (9.2)), la performance de segmentation peut être exprimée en termes de deux mesures d'erreur : le VMER (*Vocal Miss Error Rate*), qui est le taux de trames vocales détectées comme non vocales, et le VFAR (*Vocal False Alarm Rate*), qui est le taux de trames non vocales détectées comme vocales. Ces mesures d'erreur sont calculées en comparant la segmentation automatique évaluée avec une segmentation manuelle. Cependant, parce qu'il est difficile de marquer précisément à la main les transitions entre les parties vocales et non vocales, une certaine tolérance est utilisée. Notamment, comme cela est fait dans l'article [Tsai-04a], les trames qui se trouvent dans un intervalle de 0.25 secondes de part et d'autre d'un point de transition ne sont pas prises en compte pour le calcul du VMER et du VFAR. Les coordonnées de chaque point d'une courbe DET sont le VMER et le VFAR pour un certain seuil de décision η.

11.4 Paramètres acoustiques

Nous utilisons comme paramètres acoustiques (Sec. 9.3) les paramètres classiques basés sur des coefficients cepstraux (MFCC pour *Mel Frequency Cepstral Coefficients*) [Vergin-99]. Pour chaque trame, nous avons pris les 12 premiers coefficients MFCC plus l'énergie (ainsi 13 paramètres) complétés par leurs dérivés Δ et leurs accélérations $\Delta\Delta$ (ainsi 39 paramètres). Les coefficients MFCC sont obtenus à partir de la TFCT calculée en utilisant la même fenêtre d'analyse que pour la séparation (Sec. 4.5.1), c'est-à-dire la fenêtre de Hamming de taille 93 ms avec un recouvrement à 50 %. Pour diminuer l'influence du bruit convolutif et additif, les paramètres sont normalisés en utilisant les techniques CMS (*Cepstral Mean Subtraction*) et VN (*Variance Normalization*) [Chen-02].

11.5 Simulations

Les objectifs des deux expérimentations présentées par la suite sont :

1. Mesurer l'influence sur les performances de segmentation de deux types de décision : par trame et par bloc. Dans le cas de la deuxième décision, tester les différentes tailles du bloc.
2. Mesurer l'influence sur les performances de segmentation du nombre d'états des modèles Γ_V et Γ_N.

11.5.1 Décision par trame vs. décision par bloc, taille du bloc

En utilisant des MMG Γ_V et Γ_N à 32 états, nous mesurons les performances du système de segmentation avec la décision par trame (Eq. (9.1)), puis la décision par bloc (Eq. (9.2)) de taille 0.5, 1 et 2 secondes. Les résultats sont représentés sur la figure 11.1. En faisant la décision par trame, la performance en termes de EER (*Equal Error Rate*) (c'est-à-dire VMER = VFAR) est de 29 %. Il faut remarquer que le EER est de 50 % pour un segmenteur aléatoire. Avec la décision par bloc de taille 0.5 secondes, le EER baisse significativement jusqu'au 20 %. Ce résultat s'améliore encore un peu quand la taille du bloc de décision est augmentée jusqu'à 1 seconde (EER = 17 %). L'augmentation de cette taille jusqu'à 2 secondes ne change plus sensiblement le EER. Ainsi, nous avons décidé d'utiliser par la suite la décision par bloc de taille 1 seconde.

11.5.2 Nombre d'états des modèles

Ensuite, nous étudions l'influence sur les performances de segmentation du nombre d'états $Q_V = Q_N$ des MMG Γ_V et Γ_N. Les résultats obtenus en utilisant la décision par bloc de taille 1 seconde et les modèles à 8, 16, 32 et 64 états sont résumés sur la figure 11.2. On voit que le EER de 17 % obtenu avec des modèles à 32 états ne s'améliore plus en augmentant le nombre d'états jusqu'à 64. Nous continuons donc à utiliser par la suite les modèles à 32 états.

11.6 Conclusion

Nous avons évalué le module de segmentation des chansons en parties vocales et non vocales et réglé certains paramètres de ce module. Précisément, nous avons choisi d'utiliser les modèles à 32 états et de faire la décision par bloc de taille 1 seconde. Les performances obtenues dans cette configuration (EER = 17 %) sont du même ordre de grandeur que les performances rapportées dans la littérature pour la même tâche et avec les mêmes techniques basées sur des modèles MMG ou MMC. Par exemple, dans les articles [Nwe-04], [Berenzweig-01] et [Tsai-04a] le meilleur EER rapporté est de 13 %, de 19 % et de 15 % respectivement. Même si l'on ne peut pas comparer ces résultats rigoureusement, car les bases d'évaluation sont différentes, cela nous donne une petite idée du positionnement du système développé par rapport aux systèmes existants basés sur les mêmes techniques.

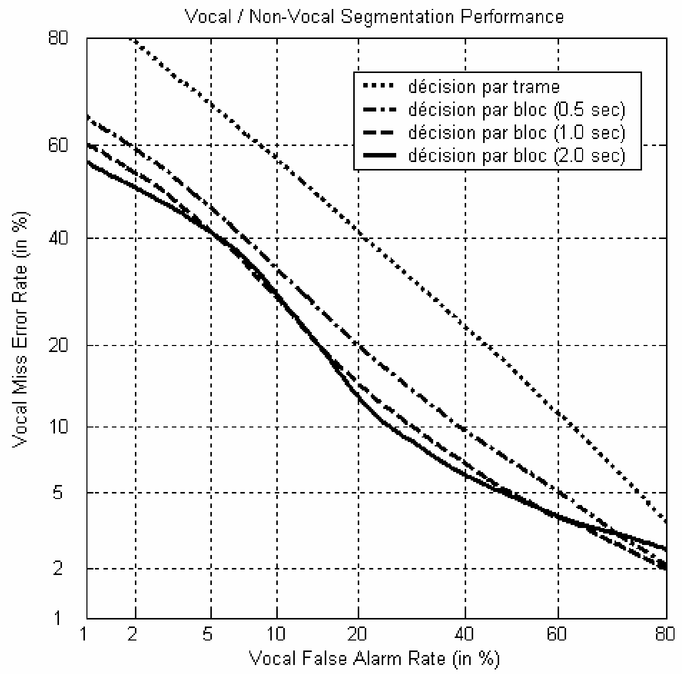


FIG. 11.1 – Influence de la taille du bloc de décision sur les performances de segmentation en parties vocales et non vocales (modèles Γ_V et Γ_N à 32 états sont utilisés). Décision par trame, EER = 29 % (pointillés). Décision par bloc de taille 0.5 secondes, EER = 20 % (points-tirets). Décision par bloc de taille 1 seconde, EER = 17 % (tirets). Décision par bloc de taille 2 secondes, EER = 17 % (ligne continue).

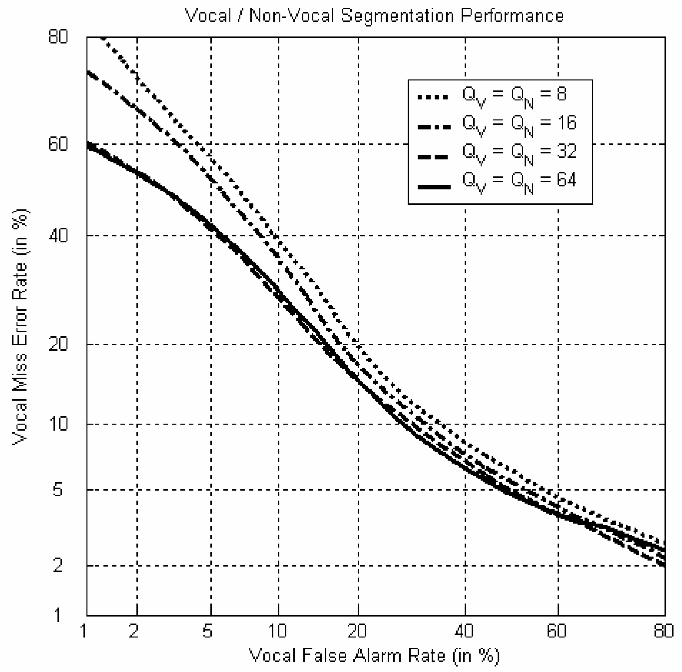


FIG. 11.2 – Influence du nombre d'états $Q = Q_V = Q_N$ des modèles Γ_V et Γ_N sur les performances de segmentation en parties vocales et non vocales (décision par bloc de taille 1 seconde est utilisée). $Q = 8$, EER = 20 % (pointillés) ; $Q = 16$, EER = 19 % (points-tirets) ; $Q = 32$, EER = 17 % (tirets) ; $Q = 64$, EER = 17 % (ligne continue).

Cependant, il reste à choisir le seuil de décision η (correspondant à un point de fonctionnement sur la courbe DET). Puisque le but final de ce travail est d'améliorer les performances de séparation, la valeur de ce seuil sera choisie à travers de ces performances, quand le bloc de la segmentation automatique sera intégré dans le module d'adaptation (Fig. 9.1). Ce choix sera fait dans le chapitre suivant.

Remarquons que dans ce choix, il y a un compromis entre la pureté et la quantité des données. En effet, les parties non vocales sont utilisées pour l'adaptation acoustique du modèle de musique $\tilde{\lambda}_m$ (Fig. 9.1). Ainsi, d'une part il faut que le VMER soit petit pour que les parties non vocales soient pures ou peu perturbées par des trames vocales détectées par erreur. D'autre part, il faut que le VFAR soit petit pour qu'il y ait beaucoup de trames non vocales détectées correctement, sinon il y a peu de données pour adapter le modèle de musique.

Chapitre 12

Séparation voix / musique

Le but de ce chapitre est d'évaluer le système de séparation voix / musique présenté chapitre 9. La mesure de performance de séparation RSDN sera utilisée pour cette évaluation.

Le bloc de segmentation en parties vocales et non vocales qui fait partie du module d'adaptation (Fig. 9.1) à déjà été évalué dans le chapitre précédent et il reste seulement à régler le seuil de décision η.

12.1 Protocole expérimental

Les données expérimentales sont les mêmes que celles utilisées chapitre 4. Elles sont décrites dans la section 4.4.

Les modèles généraux Λ_v et Λ_m sont appris à partir des données d'entraînement. Les chansons de la base d'évaluation sont ensuite séparées avec l'adaptation des modèles (Fig. 6.1) ou sans adaptation (Fig. 2.7). La séparation avec des modèles idéaux λ_v^{Idl} et λ_m^{Idl} (appris à partir des sources) est effectuée également.

L'apprentissage des MMG spectraux est effectué en utilisant l'algorithme 2. Les sources sont estimées à l'aide du filtrage de Wiener adaptatif présenté dans la sections 2.4.1.2. Le module d'adaptation des modèles est présenté dans la section 9.2.

Enfin, pour estimer la performance de séparation, la mesure RSDN (3.4) est utilisée. La moyenne des RSDN sur six chansons de test sert à estimer la performance globale.

Toutes les autres particularités seront expliquées dans la section suivante au fur et à mesure de la présentation des expériences.

12.2 Simulations

Les expérimentations que nous allons présenter sont organisées en trois parties :

1. D'abord nous étudions, comment évoluent les performances de séparation en fonction du seuil de décision du module de segmentation utilisé pour l'adaptation acoustique du modèle de musique.
2. Ensuite, nous mesurons l'apport aux performances des différentes techniques d'adaptation intégrées dans le module d'adaptation schématisé sur la figure 9.1.
3. Enfin, nous étudions l'effet sur les performances du nombre d'états des modèles. Cette expérimentation reprend celle présentée dans la section 4.5.2, mais avec l'adaptation des modèles en plus.

12.2.1 Seuil de décision de la segmentation automatique

Notons que les courbes DET que nous avons utilisées pour évaluer la segmentation ne représentent pas un système de segmentation, mais plutôt un ensemble de systèmes qu'on peut obtenir en choisissant différemment le seuil η. Ainsi, pour utiliser un système de segmentation au sein du module d'adaptation, il faut choisir une valeur particulière de ce seuil.

Pour faire ce choix, nous faisons l'expérimentation suivante. En utilisant les modèles à 32 états et la décision par bloc de taille 1 seconde, la segmentation est effectuée pour différentes valeurs du seuil η. Pour chaque valeur, un modèle de musique $\tilde{\lambda}_m$ est adapté acoustiquement (c'est-à-dire, dans notre cas, appris sur les parties non vocales). Ensuite, les chansons de test sont séparées en utilisant ce modèle de musique $\tilde{\lambda}_m$ et le modèle général de voix Λ_v. Les deux modèles ($\tilde{\lambda}_m$ et Λ_v) sont à 32 états.

Selon les résultats représentés sur la figure 12.1 le meilleur RSDN est obtenu pour $\eta = 0.25$. En observant la courbe DET, notons que le point de fonctionnement pour ce seuil (VMER = 20 %, VFAR = 15 %) n'est par très loin ni du point de fonctionnement du EER = 17 %, ni du point de fonctionnement pour le seuil bayésien $\eta = 0$ (VMER = 15 %, VFAR = 19 %), qui est le seuil théorique si l'on suppose que les deux classes (vocale et non vocale) sont équiprobables.

Nous avons choisi d'utiliser par la suite le seuil $\eta = 0.25$, puisqu'il donne la meilleure performance de séparation.

12.2.2 Apport des différentes adaptations

Pour estimer l'importance de chaque bloc du module d'adaptation (Fig. 9.1), ainsi que l'importance d'adaptation des différentes combinaisons des paramètres (filtres, gains de DSP) les expérimentations sur la séparation sont faites avec des modèles de voix et de musique à 32 états chacun dans les configurations suivantes :

1. **Modèles généraux** Λ_v et Λ_m appris sur les données d'entraînement.

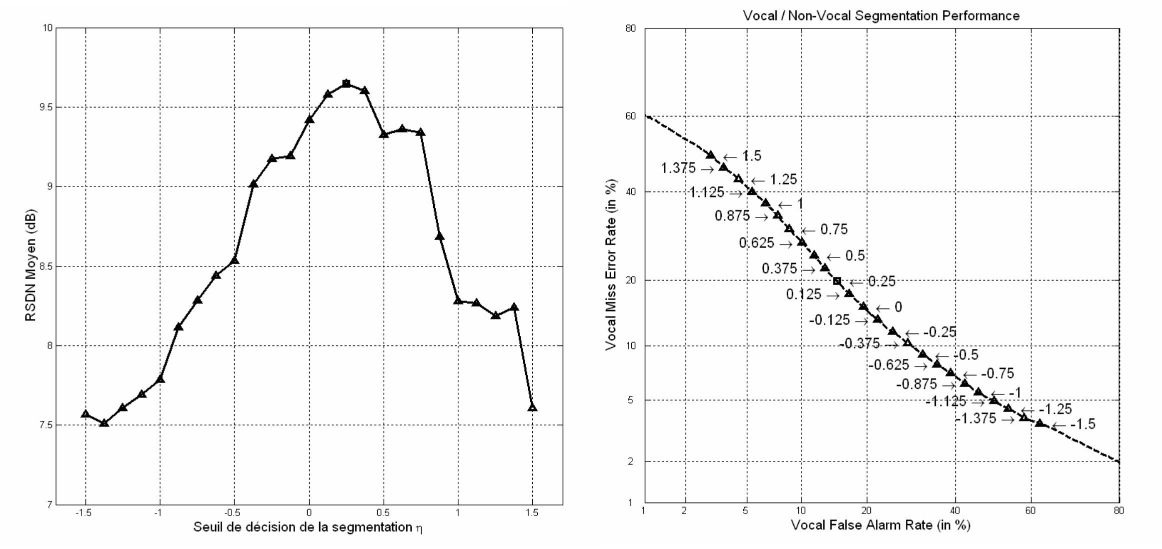


FIG. 12.1 – Performances de séparation en fonction du seuil de décision de la segmentation automatique. A gauche : RSDN moyen en fonction du seuil η. A droite : points de fonctionnement sur la courbe DET correspondants aux seuils de décision testés.

2. **Modèles adaptés acoustiquement** : modèle général de voix Λ_v et modèle de musique adapté acoustiquement $\tilde{\lambda}_m$ (c'est-à-dire, dans notre cas, appris sur des parties non vocales).
3. **Modèles adaptés** λ_v et λ_m obtenus à partir des modèles $\Lambda_v^{\mathcal{H}}$ [1] et $\tilde{\lambda}_m$ en adaptant les différentes combinaisons des paramètres (Alg. 5) [2] :
   (a) un filtre adapté pour le modèle de voix $\{\mathcal{H}_v\}$,
   (b) un filtre et des gains adaptés pour le modèle de voix $\{\mathcal{H}_v, g_v\}$,
   (c) un filtre adapté pour le modèle de voix et des gains adaptés pour les deux modèles $\{\mathcal{H}_v, g_v, g_m\}$,
   (d) des filtres et des gains adaptés pour les deux modèles $\{\mathcal{H}_v, g_v, \mathcal{H}_m, g_m\}$.
4. **Modèles idéaux** λ_v^{Idl} et λ_m^{Idl} appris sur les sources séparées S_v et S_m disponibles dans le cadre expérimental. La performance de séparation obtenue avec ces modèles, qui est inaccessible dans un cadre d'application réelle, joue le rôle de borne empirique supérieure pour les performances qu'on pourrait atteindre en adaptant les modèles.

Puisque l'adaptation acoustique du modèle de musique $\tilde{\lambda}_m$ nécessite une segmentation en parties vocales et non vocales, les tests utilisant ce modèle sont effectués avec la segmentation manuelle, puis automatique.

[1] Rappelons que le modèle $\Lambda_v^{\mathcal{H}}$ est appris en utilisant l'apprentissage à filtres adaptés, c'est-à-dire l'algorithme 6.

[2] Notons que quand seulement une partie des paramètres $\{\mathcal{H}_v, g_v, \mathcal{H}_m, g_m\}$ est adaptée l'algorithme 5 est toujours applicable avec des modifications légères. Par exemple, pour adapter $\{\mathcal{H}_v, \mathcal{H}_m, g_m\}$ l'équation (9.19) devrait être sautée.

Types de modèles	Modèle de voix	Modèle de musique	Segmentation Man.	Segmentation Aut.
Modèles généraux (état de l'art)	Λ_v	Λ_m	5.4	
Modèles adaptés acoustiquement	Λ_v	$\tilde{\lambda}_m$	11.3	9.6
Modèles adaptés (filters, gains de DSP)	$\lambda_v = \mathcal{H}_v \Lambda_v^{\mathcal{H}}$	$\lambda_m = \tilde{\lambda}_m$	12.3	10.5
	$\lambda_v = g_v \bullet \mathcal{H}_v \Lambda_v^{\mathcal{H}}$	$\lambda_m = \tilde{\lambda}_m$	**12.8**	**10.7**
	$\lambda_v = g_v \bullet \mathcal{H}_v \Lambda_v^{\mathcal{H}}$	$\lambda_m = g_m \bullet \tilde{\lambda}_m$	12.5	10.7
	$\lambda_v = g_v \bullet \mathcal{H}_v \Lambda_v^{\mathcal{H}}$	$\lambda_m = g_m \bullet \mathcal{H}_m \tilde{\lambda}_m$	12.2	10.7
Modèles idéaux (borne empirique)	λ_v^{Idl}	λ_m^{Idl}	*15.9*	

TAB. 12.1 – Importance d'adaptation des différentes combinaisons des paramètres. La performance de séparation (RSDN moyen) est affichée pour chaque type de modèles testés.

Les résultats sont représentés dans le tableau 12.1. L'amélioration principale de la performance est obtenue grâce à l'adaptation acoustique du modèle de musique sur des parties non vocales. Cette amélioration est de 5.9 dB et de 4.2 dB pour les segmentations manuelle et automatique respectivement. Elle est ainsi de 1.7 dB plus faible pour la segmentation automatique.

L'adaptation d'un filtre $\mathcal{H}_v$ pour le modèle de voix avec l'apprentissage du modèle général $\Lambda_v^{\mathcal{H}}$ à filtres adaptés (Alg. 6) améliorent toujours la performance de 1 dB et de 0.9 dB pour ces deux types de segmentations. L'adaptation supplémentaire des gains de DSP g_v pour ce modèle augmente la performance encore un peu. L'adaptation des paramètres pour le modèle de musique (g_m et $\mathcal{H}_m$) n'augmente plus la performance. Ceci signifie que le modèle de musique $\tilde{\lambda}_m$ est déjà suffisamment bien adapté.

En résumant, par rapport aux modèles généraux de l'état de l'art, notre proposition permet de gagner 7.4 dB avec une légère intervention humaine (pour faire la segmentation manuelle) et de gagner 5.3 dB de manière complètement automatique. Notons que ces résultats sont 3.1 dB et 5.2 dB au-dessous de la borne empirique obtenue à l'aide des modèles idéaux. Il reste donc un défi à diminuer ces marges en améliorant le schéma d'adaptation des modèles.

12.2.3 Effet du nombre d'états des modèles

Enfin, nous testons l'effet sur les performances du nombre d'états des modèles $Q = Q_v = Q_m$ dans les trois configurations suivantes :

1. Modèles généraux Λ_v et Λ_m.
2. Modèles adaptés $\lambda_v = g_v \bullet \mathcal{H}_v \Lambda_v^{\mathcal{H}}$ et $\lambda_m = \tilde{\lambda}_m$ qui donnent le meilleur résultat (Tab. 12.1) avec la segmentation manuelle.
3. Modèles adaptés $\lambda_v = g_v \bullet \mathcal{H}_v \Lambda_v^{\mathcal{H}}$ et $\lambda_m = \tilde{\lambda}_m$ avec la segmentation automatique.
4. Modèles idéaux λ_v^{Idl} et λ_m^{Idl}.

Les résultats moyens sont résumés sur la figure 12.2. Comme il a été déjà remarqué section 4.5.2 avec les modèles généraux, l'augmentation du nombre d'états Q n'améliore pas sensiblement la performance par rapport au filtrage de Wiener ($Q = 1$). Très vraisemblablement, ceci est dû au problème de non représentativité des bases d'entraînement pour des classes sonores très riches. En revanche, l'adaptation des modèles permet de dépasser ces limites. Avec les modèles adaptés l'augmentation des tailles des modèles, c'est-à-dire l'augmentation de la complexité calculatoire, est rémunérée par l'amélioration sensible des performances de séparation. Cette expérience montre donc que pour la séparation des classes sonores très riches, comme par exemple la musique, l'adaptation des modèles est indispensable.

Les résultats de la même expérience sont représentés sur la figure 12.3 pour chaque chanson de test. Ces résultats montrent que le comportement de notre schéma d'adaptation observé en moyenne est régulier.

Pour faire une analyse plus approfondie des chutes des performances entre l'adaptation avec la segmentation manuelle et celle avec la segmentation automatique (Fig. 12.3) nous avons représenté sur la figure 12.4 les performances de segmentation en parties vocales et non vocales pour chaque chanson de test. En plus des VMER et VFAR (Sec. 11.3), nous utilisons une mesure globale d'erreur appelée TER (*Total Error Rate*), qui est le taux de trames détectées incorrectement (c'est-à-dire non vocales détectées comme vocales et vocales détectées comme non vocales) parmi toutes les trames de la chanson.

Considérons les chansons pour lesquelles les performances de séparation ne changent pas beaucoup entre les segmentations manuelle et automatique (mélanges 2, 4 et 6, Fig. 12.3). Pour ces chansons, le VMER est inférieur ou égal à 10 % et le TER est inférieur à 20 % (Fig. 12.4). On voit aussi que pour le mélange 1, il y a un comportement anormal : les performances de séparation diminuent beaucoup, tandis que le VMER et le TER ne sont pas trop grands (10 et 16 %). La particularité de ce morceau est qu'il est très court (35 sec.) par rapport aux autres chansons qui durent entre 3 et 5 min. Remarquons également que pour la chanson 5, la segmentation est très mauvaise (TER = 50 %, ce qui est égal aux performances d'un segmenteur aléatoire). Cette chanson a deux particularités. Premièrement, le niveau de la voix chantée est très faible par rapport au niveau de la musique, il est donc difficile de distinguer la voix. Deuxièmement, dans ce morceau il y a beaucoup de bombarde (un instrument dont les caractéristiques ressemblent à la voix) qui est prise par erreur pour la voix chantée. Cependant, même avec une si mauvaise segmentation, l'adaptation permet d'améliorer les performances pour la chanson 5 par rapport aux modèles généraux (Fig. 12.3).

Enfin, remarquons que la complexité calculatoire du système développé est assez raisonnable. Pour les modèles à 32 états ($Q_v = Q_m = 32$), la séparation (c'est-à-dire l'adaptation et l'estimation des sources) de 23 minutes d'enregistrement (durée totale des 6 chansons de test) dure

4 heures en utilisant un ordinateur portable muni d'un processeur Pentium M 1.7 GHz.

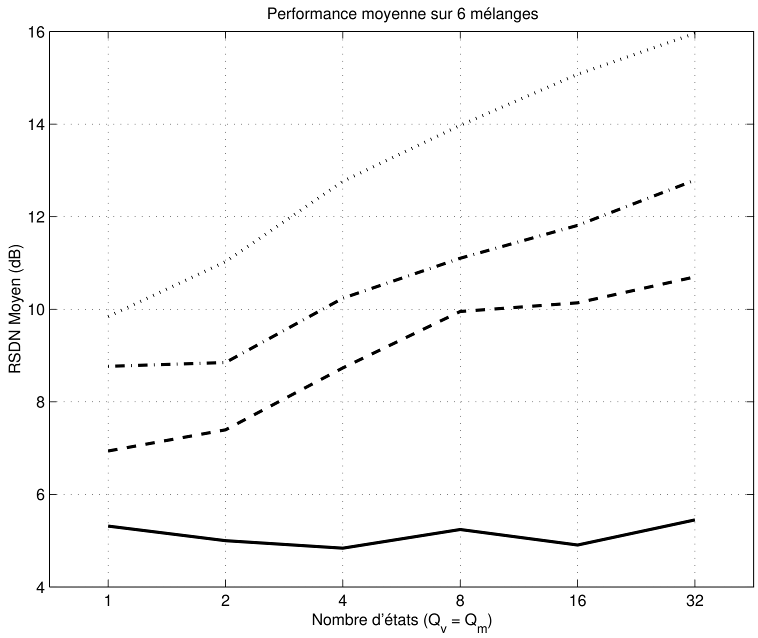


FIG. 12.2 – RSDN moyen pour six chansons de test en fonction du nombre d'états des modèles $Q = Q_v = Q_m$ et pour différents types de modèles. Ligne continue : Modèles généraux ; Tirets : Modèles adaptés avec la segmentation automatique ; Points-tirets : Modèles adaptés avec la segmentation manuelle ; Pointillés : Modèles idéaux (borne empirique).

12.3 Conclusion

Le problème de séparation de la voix par rapport à la musique dans des enregistrement monophoniques de chansons populaires est une tâche difficile qui n'a pas été beaucoup traitée dans la littérature ([Ozerov-05b, Vembu-05, Li-06]). Pour cette tâche, nous avons développé un système de séparation qui a les avantages suivants grâce à l'adaptation des modèles :

- Les performances moyennes sont améliorées de 5 dB par rapport à l'utilisation des modèles généraux.
- Le système marche de manière automatique, c'est-à-dire sans intervention humaine.
- La complexité calculatoire est assez raisonnable (pas plus que 10 fois le temps réel).
- Les expérimentations ont été effectuées sans restrictions particulières ni sur le style musical (en restant cependant dans le cadre des chansons populaires) ni sur la langue de la chanson.

Quelques exemples de séparation voix / musique obtenus à l'aide de ce système se trouvent sur ma page personnelle [Ozerov-www].

Remarquons cependant quelques limitations. Premièrement, la chanson traitée doit contenir suffisamment de parties non vocales pour qu'il y ait assez de données pour l'adaptation acous-

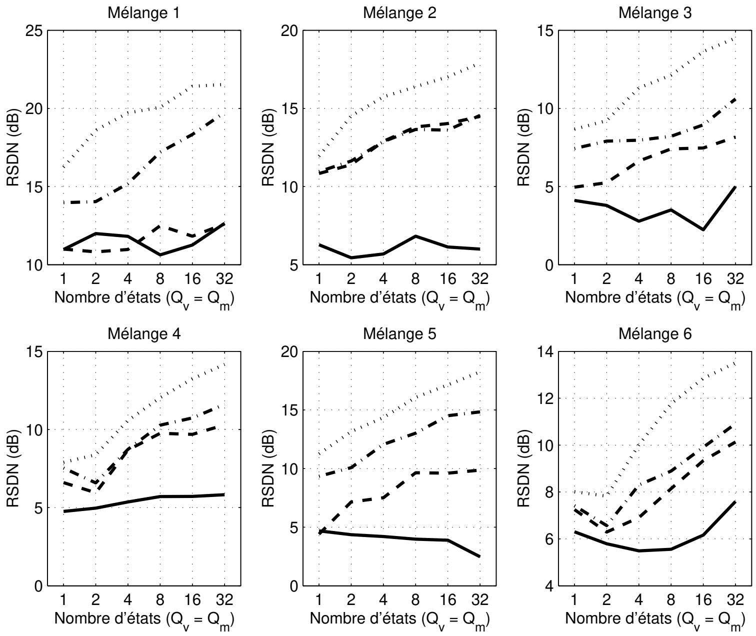


FIG. 12.3 – RSDN détaillé pour six chansons de test en fonction du nombre d'états des modèles $Q = Q_v = Q_m$ et pour différents types de modèles. Ligne continue : Modèles généraux ; Tirets : Modèles adaptés avec la segmentation automatique ; Points-tirets : Modèles adaptés avec la segmentation manuelle ; Pointillés : Modèles idéaux (borne empirique).

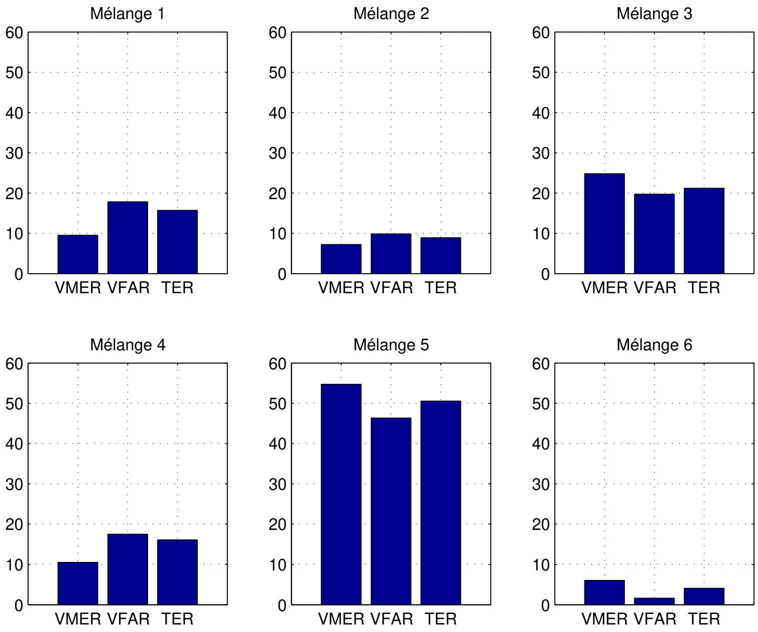


FIG. 12.4 – Performances de segmentation en parties vocales et non vocales pour chaque chanson de test : VMER (*Vocal Miss Error Rate*), VFAR (*Vocal False Alarm Rate*) et TER (*Total Error Rate*).

tique du modèles de musique. Deuxièmement, la musique issue des parties non vocales doit être assez similaire à celle issue des parties vocales. Enfin, il est préférable qu'à chaque instant, il n'y ait qu'une seule personne qui chante, c'est-à-dire qu'il n'y ait ni de choeur ni de back vocal. Ces suppositions sont vérifiées pour la majorité des chansons populaires.

Il faut noter également que les performances de ce système sont 5.2 dB au-dessous de la borne empirique obtenue avec des modèles idéaux. Il reste donc un défi à diminuer cette marge en améliorant le schéma d'adaptation ainsi que le module de segmentation en parties vocales et non vocales qui fait partie de ce schéma.

Il est important de noter qu'avec les modèles généraux, l'augmentation du nombre d'états des modèles n'améliore pas les performances moyennes, tandis que l'adaptation des modèles permet de dépasser sensiblement ces limitations. Ainsi, une conclusion que nous tirons de ces expérimentations est que, pour la séparation des classes sonores très riches (comme la musique), l'adaptation des modèles est indispensable.

Chapitre 13

Apport de la séparation pour l'estimation du pitch de la voix

Nous allons mesurer ce qu'apporte la séparation avec ou sans adaptation des modèles à l'estimation du pitch de la voix chantée, en utilisant un estimateur de pitch développé pour la voix seule (non polluée par la musique).

13.1 Estimateur de pitch

Nous utilisons un estimateur de pitch développé pour la voix seule et basé sur l'autocorrélation. A partir du signal de voix seule s_v, cet estimateur sort l'estimation de pitch ρ dans le format suivant $\rho = \{\rho(n)\}_n$, où n est l'indice du temps discret correspondant à la fréquence d'échantillonnage 200 Hz, $\rho(n) = 0$ si la trame correspondante à été détectée comme non voisée. Sinon $\rho(n)$ est l'estimation de la fréquence fondamentale $f0$ en hertz.

13.2 Description des données expérimentales

Dans un premier temps, nous utiliserons pour ces expérimentions les six chansons de la base d'évaluation utilisée pour la séparation (Sec. 4.4).

Dans un deuxième temps, nous utiliserons quelques extraits de la base d'évaluation du concours sur l'extraction de la mélodie organisé pendant la conférence ISMIR 2004 [ISMIR-04]. Cela nous permettra d'avoir une idée sur les performances d'estimation du pitch que nous allons obtenir, en les comparant avec les performances des systèmes proposés par les participants du concours. La base d'évaluation du concours qui est disponible sur le site [ISMIR-04] est composées de 20 extraits musicaux de différents genres (pop, opéra, jazz, etc.). Parmi ces extraits, nous avons choisi d'utiliser les quatre extraits issus des chansons pop, qui sont notés dans la base *pop1* - *pop4*. Les autres extraits nous semblent peu appropriés pour le système de séparation développé dans cette thèse. Chaque extrait parmi les quatre choisis dure approximativement 20 secondes.

13.3 Protocole expérimental

Pour chaque chanson (ou extrait) traitée x, le pitch est estimé (avec l'estimateur utilisé) d'abord directement à partir du mélange x et ensuite à partir de la voix estimée $\hat{s}_v$, en utilisant les différents types de modèles, notamment les modèles généraux, adaptés et idéaux (Sec. 12.2.2). A chaque fois, une estimation de pitch $\hat{\rho}$ est obtenue. Pour mesurer la précision de cette estimation, elle doit être comparée avec un pitch de référence.

13.4 Pitch de référence

Pour chaque chanson de la base de test de la séparation, nous prenons comme pitch de référence le pitch ρ estimé (avec l'estimateur utilisé) à partir de la voix seule s_v qui est disponible dans le cadre expérimental. Pour les extraits de la base d'évaluation du concours de l'ISMIR 2004, les pitchs de référence sont fournis avec la base. D'ailleurs, ces références sont obtenues de la même manière, c'est-à-dire qu'elles sont estimées à partir de la voix seule avec une vérification manuelle en plus [ISMIR-04].

Ainsi, l'estimation de pitch $\hat{\rho}$ est comparée avec le pitch de référence ρ en utilisant les mesures de performance décrites dans la section suivante.

13.5 Mesures de performance

Nous avons l'intention de comparer les performances que nous allons obtenir avec celles des systèmes présentés au concours de l'ISMIR 2004 [ISMIR-04]. Ainsi, pour comparer le pitch estimé $\hat{\rho}$ avec le pitch de référence ρ nous allons utiliser les mesures de performance proposées dans le cadre de ce concours. Ces mesures sont composées des deux options présentées ci-dessous.

13.5.1 Mesures de performance : Option 1

Cette option est composée des trois mesures suivantes :

1. **unpitchMatch** : mesure de concordance pour les trames non voisées (dans la référence) seulement :

$$M_{\text{unpitch}} = 100 \frac{\#\left(\{n : \hat{\rho}(n)\rho(n) = 0\}\right)}{\#\left(\{n : \rho(n) = 0\}\right)} \tag{13.1}$$

où $\#(A)$ dénote le nombre d'éléments de l'ensemble A. C'est donc le pourcentage des trames détectées correctement comme non voisées parmi toutes les trames non voisées de la référence.

2. **pitchMatch** : mesure de concordance pour les trames voisées (dans la référence) seulement :

$$M_{\text{pitch}} = 100 \frac{\sum_{\hat{\rho}(n)\rho(n)>0} \min\left(1 - e_{\text{demiton}}[\hat{\rho}(n), \rho(n)], 0\right)}{\#\left(\{n : \rho(n) > 0\}\right)}, \quad \text{avec} \tag{13.2}$$

$$e_{\text{demiton}}[\hat{\rho}(n), \rho(n)] = 12|\log_2 \hat{\rho}(n) - \log_2 \rho(n)| \tag{13.3}$$

C'est donc la différence moyenne absolue entre les notes saturée à 1 demi-ton (100 %) pour l'erreur maximale (parmi les trames voisées de la référence).

3. **totalMatch** : mesure de concordance combinée :

$$M_{\text{total}} = 100 \frac{\#\left(\{n : \hat{\rho}(n)\rho(n) = 0\}\right) + \sum_{\hat{\rho}(n)\rho(n)>0} \min\left(1 - e_{\text{demiton}}[\hat{\rho}(n), \rho(n)], 0\right)}{N}, \tag{13.4}$$

où N est le nombre total de trames ($\rho = \{\rho(n)\}_{n=1}^{N}$).

13.5.2 Mesures de performance : Option 2

Les mesures sont les mêmes que pour l'option 1 à une seule exception : il est autorisé de se tromper d'octave. Ainsi, la formule (13.3) doit être remplacée par la formule suivante :

$$e'_{\text{demiton}}[\hat{\rho}(n), \rho(n)] = \min_k \left(12|k + \log_2 \hat{\rho}(n) - \log_2 \rho(n)|\right) \tag{13.5}$$

13.6 Simulations

Comme nous l'avons déjà mentionné, les expérimentations sont organisées en deux parties :
- d'abord, avec les données utilisées pour la séparation,
- ensuite, avec les données du concours de l'ISMIR 2004.

13.6.1 Expérimentations avec les données utilisées pour la séparation

Pour chaque chanson de la base d'évaluation utilisée pour la séparation, les estimations de pitch $\hat{\rho}$ sont obtenues dans les configurations suivantes :

1. **Mélange** : $\hat{\rho}$ est estimée à partir du mélange x.

2. **Modèles généraux** : $\hat{\rho}$ est estimée à partir de la voix $\hat{s}_v$ séparée en utilisant les modèles généraux Λ_v et Λ_m.
3. **Modèles adaptés** : $\hat{\rho}$ est estimée à partir de la voix $\hat{s}_v$ séparée en utilisant les modèles adaptés $\lambda_v = g_v \bullet \mathcal{H}_v \Lambda_v^{\mathcal{H}}$ et $\lambda_m = \tilde{\lambda}_m$ qui donnent les meilleures performances de séparation (Tab. 12.1) avec la segmentation automatique.
4. **Modèles généraux (+ segm.)** : $\hat{\rho}$ est estimée à partir de la voix $\hat{s}_v$ séparée en utilisant les modèles généraux Λ_v et Λ_m et la segmentation automatique en parties vocales et non vocales. Plus précisément, dans l'estimation $\hat{s}_v$, les parties non vocales de x sont mises à zéro et les parties vocales sont séparées à l'aide des modèles généraux. Cette configuration est rajoutée pour le diagnostic, à savoir pour mesurer l'impact de la segmentation dans le processus d'adaptation des modèles.
5. **Modèles idéaux** : $\hat{\rho}$ est estimée à partir de la voix $\hat{s}_v$ séparée en utilisant les modèles idéaux λ_v^{Idl} et λ_m^{Idl}.
6. **Solution triviale** : Enfin, nous considérons une solution triviale $\hat{\rho}(n) = 0$ pour tous n ; pour cette solution le totalMatch est égal au pourcentage des trames non voisées dans la référence.

Les résultats moyens sont représentés dans les tableaux 13.1 et 13.2 en utilisant les options 1 et 2 des mesures de performance, respectivement. On voit que les résultats pour ces deux options sont cohérents entre eux. Ainsi, pour des raisons de simplicité, nous n'allons analyser et utiliser par la suite que l'option 1.

Remarquons d'abord qu'avec la solution triviale, le totalMatch est assez élevé, il est de 75 %. Ceci signifie simplement qu'il y a 75 % de trames non voisées dans les données traitées. Bien évidemment, nous ne pouvons pas nous contenter de cette solution triviale. Elle est présentée juste pour monter qu'il faut analyser les trois mesures ensemble, et pas seulement le totalMatch, même si il est censé combiner le unpitchMatch et le pitchMatch.

Les résultats représentés tableau 13.1 nous montrent que l'utilisation des modèles généraux améliore significativement le pitchMatch de 32 % par rapport à la configuration sans séparation. A son tour, l'adaptation des modèles améliore de 25 % le unpitchMatch. Puisque l'adaptation ne change pas le pitchMatch, on peut croire que c'est surtout la segmentation en parties vocales et non vocales qui joue le rôle principal. C'est pour vérifier cette hypothèse que nous avons testé la configuration "modèles généraux (+ segm.)". Cependant, on observe que l'utilisation des modèles généraux avec la segmentation dégrade légèrement le unpitchMatch et dégrade le pitchMatch de 11 % par rapport aux modèles adaptés. Ainsi, parmi les quatre configurations testées, [1] les modèles adaptés donnent les meilleures performances moyennes avec chacune des

[1]Dans cette comparaison, nous ne prenons en considération ni les modèles idéaux (car c'est irréaliste) ni la solution triviale.

trois mesures.

Données pour la séparation	Option 1		
	unpitchMatch (%)	pitchMatch (%)	totalMatch (%)
Mélange	67	12	53
Modèles généraux	67	45	61
Modèles adaptés	**92**	**46**	**80**
Modèles généraux (+ segm.)	88	35	75
Modèles idéaux	*86*	*70*	*82*
Solution triviale $\hat{\rho}(n) = 0$	100	0	75

TAB. 13.1 – Résultats moyens de l'estimation de pitch en utilisant les données pour la séparation et l'option 1 des mesures de performance.

Données pour la séparation	Option 2		
	unpitchMatch (%)	pitchMatch (%)	totalMatch (%)
Mélange	67	21	55
Modèles généraux	67	**49**	62
Modèles adaptés	**92**	**49**	**81**
Modèles généraux (+ segm.)	88	38	75
Modèles idéaux	*86*	*74*	*83*
Solution triviale $\hat{\rho}(n) = 0$	100	0	75

TAB. 13.2 – Résultats moyens de l'estimation de pitch en utilisant les données pour la séparation et l'option 2 des mesures de performance.

Les mêmes résultats (en utilisant l'option 1) sont détaillés pour chaque chanson de test dans le tableau 13.3. On voit que le comportement observé en moyenne est plutôt régulier. Pour le unpitchMatch et le pitchMatch, l'utilisation des modèles adaptés donne de meilleures performances pour quatre chansons sur six. Quand ce n'est pas le cas, la différence entre les meilleures performances et celles obtenues avec les modèles adaptés ne dépasse jamais 10 %.

Remarquons aussi que les plus mauvaises performances sont obtenues pour la chanson 5 (unpitchMatch = 84 % et pitchMatch = 29 %). Vraisemblablement, ceci est lié au fait que la segmentation en parties vocales et non vocales est très mauvaise pour cette chanson (TER = 50 %, voir Fig. 12.4).

13.6.2 Expérimentations avec les données de l'ISMIR 2004

Exactement les mêmes expérimentations sont effectuées pour les quatre extraits de la base d'évaluation du concours de l'ISMIR 2004. Seule la configuration avec des modèles idéaux n'est pas reproduite, puisque nous n'avons pas accès aux sources séparées nécessaires pour l'apprentissage de ces modèles.

Option 1	Données pour la séparation	Chanson					
		1	2	3	4	5	6
unpitchMatch (%)	Mélange	70	68	60	71	64	73
	Modèles généraux	88	63	54	83	56	79
	Modèles adaptés	**96**	**98**	**90**	91	**84**	94
	Modèles généraux (+ segm.)	94	91	87	**95**	71	**96**
	Modèles idéaux	*95*	*96*	*74*	*89*	*85*	*78*
	Solution triviale $\hat{\rho}(n) = 0$	100	100	100	100	100	100
pitchMatch (%)	Mélange	22	6	8	19	24	5
	Modèles généraux	**67**	40	**54**	54	28	49
	Modèles adaptés	58	**55**	43	**56**	**29**	**51**
	Modèles généraux (+ segm.)	53	39	35	47	13	46
	Modèles idéaux	*88*	*66*	*78*	*79*	*63*	*68*
	Solution triviale $\hat{\rho}(n) = 0$	0	0	0	0	0	0
totalMatch (%)	Mélange	60	57	47	63	53	42
	Modèles généraux	84	59	54	79	48	65
	Modèles adaptés	**88**	**90**	**78**	86	**68**	**75**
	Modèles généraux (+ segm.)	86	81	74	**88**	55	73
	Modèles idéaux	*93*	*91*	*75*	*87*	*79*	*74*
	Solution triviale $\hat{\rho}(n) = 0$	79	82	75	85	72	55

TAB. 13.3 – Résultats de l'estimation de pitch détaillés pour chaque chanson de la base de test pour la séparation en utilisant l'option 1 des mesures de performance.

Les résultats moyens sont représentés tableau 13.4, accompagnés par les résultats des quatre participants du concours de l'ISMIR 2004. Pour les participants du concours, seul le totalMatch est indiqué, puisque les valeurs du unpitchMatch et du pitchMatch ne sont pas affichées sur le site [ISMIR-04]. Ces résultats sont détaillés pour chaque extrait de test tableau 13.5.

On voit que cette fois-ci, c'est l'utilisation des modèles généraux qui donne le meilleur totalMatch (Tab. 13.4) et l'adaptation des modèles dégrade les performances moyennes. Vraisemblablement, ceci est dû au fait que chaque extrait traité est très court et contient peu de trames non vocales, ce qui est insuffisant pour apprendre un modèle de musique. En effet, chaque extrait dure approximativement 20 secondes. De plus, comme nous l'indique le totalMatch de la solution triviale (Tab. 13.4), il y a seulement 22 % de trames non voisées en moyenne, et donc il y a encore moins de trames non vocales. En faisant un petit calcul à partir de ces chiffres, on peut déduire qu'en moyenne chaque extrait contient au plus une centaine de trames non vocales, ce qui n'est pas beaucoup pour apprendre un modèle à 32 états. C'est justement la situation où l'adaptation MAP du modèle de musique (testée dans la section 9.4.1) semble être plus appropriée que le réapprentissage complet.

Toutefois, ces résultats ne remettent pas en cause le schéma d'adaptation proposé dans cette thèse, puisque ce schéma n'a pas été développé pour traiter des extraits de 20 secondes, mais pour traiter des chansons entières qui durent en moyenne entre 3 et 5 minutes.

Ainsi, le meilleur totalMatch de 49 % est obtenu avec des modèles généraux. On voit que ce résultat est comparable à ceux des participants du concours de l'ISMIR 2004. Ceci nous donne une idée sur le positionnement de notre proposition sur l'estimation du pitch par rapport aux propositions existantes.

Notons en plus, que le système d'estimation du pitch de la voix chantée dans la musique polyphonique que nous étudions ici consiste en l'application successive des deux modules :

1. module de séparation voix / musique,
2. estimateur de pitch pour la voix seule.

Ces deux modules ont été développés indépendamment l'un de l'autre et n'ont pas été modifiés après les avoir combiné. Cependant, un petit réglage d'un module en fonction de l'autre pourra probablement améliorer le système. Par exemple, on peut essayer de régler certains paramètres de l'estimateur de pitch pour le rendre plus robuste aux erreurs de la séparation.

13.7 Conclusion

Dans ce chapitre, nous avons mesuré l'apport de la séparation voix / musique pour l'estimation du pitch de la voix chantée. Les expérimentations on été organisées en deux parties.

Données de l'ISMIR 2004	Option 1		
	unpitchMatch (%)	pitchMatch (%)	totalMatch (%)
Mélange	70	8	22
Modèles généraux	75	**41**	**49**
Modèles adaptés	85	28	41
Modèles généraux (+ segm.)	**88**	33	45
Solution triviale $\hat{\rho}(n) = 0$	100	0	22
Participant 1 (ISMIR'04)	-	-	**67**
Participant 2 (ISMIR'04)	-	-	23
Participant 3 (ISMIR'04)	-	-	57
Participant 4 (ISMIR'04)	-	-	48

TAB. 13.4 – Résultats moyens de l'estimation de pitch en utilisant les données de l'ISMIR 2004 et l'option 1 des mesures de performance.

Option 1	Données de l'ISMIR 2004	Extrait			
		pop1	pop2	pop3	pop4
unpitchMatch (%)	Mélange	75	48	73	87
	Modèles généraux	68	64	94	76
	Modèles adaptés	65	79	96	**99**
	Modèles généraux (+ segm.)	**84**	**85**	**99**	84
	Solution triviale $\hat{\rho}(n) = 0$	100	100	100	100
pitchMatch (%)	Mélange	3	19	0	12
	Modèles généraux	**30**	**64**	31	**43**
	Modèles adaptés	8	49	**38**	24
	Modèles généraux (+ segm.)	**30**	57	27	23
	Solution triviale $\hat{\rho}(n) = 0$	0	0	0	0
totalMatch (%)	Mélange	17	27	16	28
	Modèles généraux	37	**64**	45	**50**
	Modèles adaptés	19	57	**51**	40
	Modèles généraux (+ segm.)	**40**	**64**	42	36
	Solution triviale $\hat{\rho}(n) = 0$	20	26	21	21
	Participant 1 (ISMIR'04)	**61**	**64**	**73**	71
	Participant 2 (ISMIR'04)	17	19	26	32
	Participant 3 (ISMIR'04)	55	58	46	71
	Participant 4 (ISMIR'04)	26	29	63	**73**

TAB. 13.5 – Résultats de l'estimation de pitch détaillés pour chaque extrait de la base de test de l'ISMIR 2004 en utilisant l'option 1 des mesures de performance.

D'abord, la base de test de la séparation a été utilisée pour l'évaluation. Dans ce cas, la séparation de la voix chantée facilite l'estimation du pitch. De plus, le meilleur résultat moyen est obtenu en utilisant les modèles adaptés.

Pour pouvoir comparer le système d'estimation du pitch proposé avec d'autres systèmes existants, nous avons ensuite évalué ce système sur quelques extraits utilisés pour le concours sur l'extraction de la mélodie de l'ISMIR 2004. Dans ce cas, nous avons observé que l'adaptation des modèles dégrade les performances par rapport aux modèles généraux. Nous pensons que ceci résulte du fait que les extraits traités sont trop courts. Toutefois, les résultats obtenus montrent que l'approche basée sur

- la séparation voix / musique et
- l'estimation du pitch à partir de la voix séparée

semble prometteuse et capable de concourir avec d'autres systèmes d'estimation du pitch de la voix chantée dans la musique polyphonique.

Cinquième partie

Conclusion et perspectives

Chapitre 14

Conclusion

Dans cette thèse, nous étudions les techniques de séparation de sources avec un seul capteur qui sont basées sur des modèles statistiques des sources. Ces approches souffrent de limitations majeures qui les rendent peu performantes pour la séparation des sources appartenant à des classes sonores de grande variabilité. Ceci est lié au fait qu'il est difficile en pratique d'apprendre et de traiter des modèles représentant bien de telles classes.

Pour pouvoir dépasser ces limitations, nous proposons un formalisme général d'adaptation des modèles de sources à leurs réalisations particulières dans le mélange traité. Nous présentons ce formalisme sous la forme d'un critère d'adaptation Maximum *A Posteriori* (MAP). Puis, nous développons un algorithme général permettant d'optimiser ce critère. Enfin, nous appliquons des techniques d'adaptation dérivées du formalisme général au problème de séparation de la voix chantée par rapport à la musique ambiante dans des chansons. Les résultats obtenus montrent la validité de notre proposition.

Notons que la tâche de séparation voix / musique semble être un cas favorable pour l'adaptation. En effet, dans les chansons, il y a beaucoup de parties non vocales (sans voix chantée), et nous profitons de ce fait pour adapter le modèle de musique. De plus, c'est cette adaptation qui permet de gagner le plus en termes des performances de séparation. Cependant, nous avons montré que l'adaptation peut prendre des formes plus sophistiquées, telles que l'adaptation des filtres et l'adaptation des gains de DSP à partir du mélange, ou bien l'adaptation à partir des régions temps-fréquences dans le cadre de la "théorie des données manquantes" (Sec. 9.4.2).

En général, le principe d'adaptation proposé dans cette thèse est de rapprocher les caractéristiques des modèles à celles des sources dans le mélange, en utilisant toutes les informations disponibles sur ces sources. Le mélange constitue toujours la principale information sur les sources, mais dans certains cas on peut avoir à disposition d'autres informations, appelées informations auxiliaires (Sec. 8.2), telles qu'une segmentation temporelle ou temps-fréquentielle en régions d'activité des sources, des informations visuelles, etc. Puisque souvent ces informations ne sont pas suffisantes pour pouvoir complètement réapprendre les modèles, il est important de garder une attache aux modèles *a priori*. Cependant, comme on l'a vu dans le cas de l'adaptation acoustique du modèle de musique (Sec. 9.4.1), cette attache peut être perdue, quand il y a suffisamment de nouvelles observations.

Du point de vue théorique, cette thèse peut être considérée comme un travail de convergence de différentes techniques d'apprentissage ou d'adaptation à partir du mélange ou des données manquantes vers un seul formalisme d'adaptation MAP dans le cadre bayésien. Du point de vue applicatif, ce travail élargit le champ d'application des méthodes probabilistes pour la séparation de sources appartenant à des classes sonores de grande variabilité. Quelques perspectives de ce travail sont présentées dans le chapitre suivant.

Chapitre 15

Perspectives

Le formalisme d'adaptation étant formulé de manière assez générale, il pourra trouver des applications pour d'autres tâches de séparation que celle de séparation voix / musique. Cependant, comme il est mentionné section 6.2.2, il faut être très prudent en choisissant les lois *a priori* sur les modèles. En fait, la clé de la réussite de la méthode réside dans un bon choix des ces lois.

Ainsi, une piste de recherche intéressante sera d'essayer de proposer des techniques de construction de lois *a priori* de manière plus ou moins automatisée. On peut s'inspirer par exemple par la technique EMLLR (*Eigenspace-Based MLLR*) [Chen-00] selon laquelle l'adaptation des modèles est effectuée le long des directions de plus forte variation des paramètres qui sont sélectionnées auparavant de manière automatique.

L'adaptation des modèles peut être étendue à des méthodes de séparation de sources avec plusieurs capteurs, qui sont basées à la fois sur l'utilisation de l'information spatiale et sur des modèles *a priori* des sources (voir [Vincent-04]). Par exemple, une idée consiste à utiliser des techniques d'apprentissage ou d'adaptation des modèles à partir de régions temps - fréquence dans lesquelles une seule source est active à la fois (voir Sec. 9.4.2). Ces régions peuvent être estimées d'abord en utilisant l'information spatiale, puis les modèles appris ou adaptés sur ces régions peuvent être utilisés pour améliorer la séparation dans des régions où il y a plusieurs sources actives à la fois.

Vers des techniques rapides de séparation / adaptation

Comme il est remarqué section 2.2.2, la complexité calculatoire de l'algorithme d'estimation des sources est proportionnelle au produit des tailles des modèles ($Q_1 Q_2$), car il faut calculer la somme sur tous les états d'un MMG factoriel. La complexité calculatoire d'une itération de l'algorithme EM pour l'adaptation (Sec. 7.2) est du même ordre de grandeur, car les espérances des statistiques naturelles sont également calculées en sommant sur tous les états d'un MMG factoriel. Ainsi, il semble très important de proposer des techniques (probablement approchées) d'estimation des sources et d'adaptation des modèles dont la complexité calculatoire est inférieure à $O(Q_1 Q_2)$. Ceci permettra soit de baisser le coût de calcul, soit pour le même coût de calcul d'utiliser des modèles de plus grande taille, ce qui mènera, dans certains cas, à de meilleures performances de séparation.

Une des pistes mentionnées section 2.4.1.3 est d'utiliser des estimateurs durs pour l'estimation de sources ainsi que pour l'adaptation, c'est-à-dire pour le calcul des espérances des statistiques naturelles, et de trouver une astuce rapide de recherche d'un couple d'états le plus probable dans le modèle factoriel. Par exemple, en s'inspirant par les travaux de Pontoppidan et Dyrholm [Pontoppidan-03], on peut essayer de représenter les DSP du MMG factoriel sous la forme d'un arbre binaire en utilisant l'algorithme de K-moyennes [McQueen-67]. La recherche dans un tel arbre pourra s'effectuer considérablement plus vite par rapport à la recherche exhaustive. Cependant, une telle solution est en général approchée.

Une autre direction consiste à utiliser des techniques d'approximation variationnelle [Jordan-98] pour les MMG / MMC factoriels [Ghahramani-97]. Ces techniques ont été déjà appliquées pour la séparation de sources avec plusieurs capteurs [Attias-03], ainsi que pour la séparation de sources avec un seul capteur [Hershey-01].

Adaptation en ligne

Une des limitations de la technique d'adaptation proposée dans cette thèse est qu'on a besoin de traiter chaque enregistrement en entier. Ceci n'est pas acceptable pour des applications qui doivent fonctionner en ligne, comme par exemple la reconnaissance vocale d'un service téléphonique. Pour ces applications, le signal arrive au fur et à mesure et à chaque moment une décision sur le signal en cours de traitement doit être effectuée avant que le signal futur ne soit disponible. En fonction de l'application, un délai plus ou moins long peut être toléré.

Cependant, la technique d'adaptation proposée peut être assez facilement modifiée pour être applicable en ligne. Au lieu d'utiliser l'algorithme EM classique [Dempster-77], on peut par exemple utiliser EM récursif [Krishnamurthy-93]. L'idée consiste à recalculer les espérances des statistiques naturelles (étape E) au fur et à mesure le long d'une fenêtre glissante (par ex. rectangulaire ou exponentielle) en mettant à chaque fois à jour les paramètres des modèles adaptés (étape M). Ainsi, les modèles ne sont plus adaptés à tout l'enregistrement, mais ils varient au cours du temps en s'adaptant aux nouvelles conditions. La taille de la fenêtre glissante utilisée pour le calcul des espérances des statistiques naturelles règle la vitesse de cette variation.

Le désavantage d'un tel schéma d'adaptation en ligne par rapport à l'adaptation sur tout l'enregistrement est que l'information future n'est pas utilisée. Par contre, il y a plusieurs avantages. Premièrement, elle est applicable en ligne, c'est-à-dire qu'elle est causale. Deuxièmement, il devient possible de régler la vitesse d'adaptation en ajustant la taille de la fenêtre glissante.

Il semble même possible d'adapter les différents paramètres avec des vitesses différentes. Par exemple, puisque les conditions d'enregistrement varient en moyenne plus lentement que l'énergie locale du signal, il paraît raisonnable d'utiliser une fenêtre glissante longue pour l'adaptation des filtres (Sec. 9.5.1) et une fenêtre glissante courte pour l'adaptation des gains de DSP (Sec. 9.5.2).

Enfin, on voit que cette adaptation en ligne est quelque chose d'intermédiaire entre l'adaptation locale (par ex. l'adaptation des facteurs de gains [Benaroya-06, Vincent-04a]) et d'adaptation globale traitée dans cette thèse (voir la discussion section 6.2.3). En effet, si la taille de la fenêtre glissante est comparable avec la taille d'enregistrement, il s'agit plutôt d'adaptation globale. Par contre, si la taille de la fenêtre est de l'ordre d'une trame, il s'agit d'adaptation locale [Benaroya-06, Vincent-04a].

Annexes

Annexe A

Rappels de probabilités et de statistiques

Cette annexe contient quelques rappels des notions de probabilités et de statistiques utilisées dans cette thèse.

A.1 Densité d'un vecteur aléatoire gaussien réel / complexe

La densité d'un vecteur aléatoire gaussien réel $V \in \mathbb{R}^F$ avec comme vecteur moyen $\mu = [\mu(f)]_f \in \mathbb{R}^F$ et comme matrice de covariance diagonale $R = \mathrm{diag}[r^2(f)]_f \in \mathbb{R}^{F\times F}$ est définie comme suit [Kay-93] :

$$N(V;\mu,R) = \prod_f \frac{1}{\sqrt{2\pi r^2(f)}} \exp\left[-\frac{1}{2}\frac{(V(f)-\mu(f))^2}{r^2(f)}\right], \tag{A.1}$$

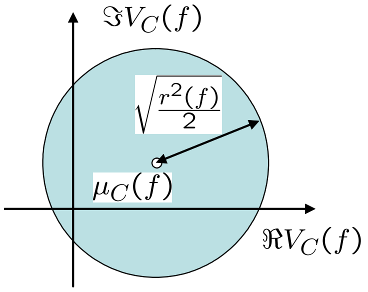
alors que la densité d'un vecteur aléatoire gaussien complexe *circulaire* [1] $V_C \in \mathbb{C}^F$ avec comme vecteur moyen $\mu_C = [\mu_C(f)]_f \in \mathbb{C}^F$ et comme matrice de covariance diagonale $R = \mathrm{diag}[r^2(f)]_f \in \mathbb{R}^{F\times F}$ est définie un peu différemment [Neeser-93] :

$$N_C(V_C;\mu_C,R) = \prod_f \frac{1}{\pi r^2(f)} \exp\left[-\frac{|V_C(f)-\mu_C(f)|^2}{r^2(f)}\right] \tag{A.2}$$

La différence entre ces deux formules peut être expliquée en supposant que chaque vecteur $[\Re V_C(f), \Im V_C(f)]^T \in \mathbb{R}^2$ est un vecteur aléatoire gaussien réel bidimensionnel avec comme

[1] Les moments centrés d'ordre 2 d'un vecteur aléatoire complexe V sont définis par une matrice de covariance $R = \mathbb{E}[(V-\mu)(V-\mu)^H]$ et une matrice de *pseudo-covariance* $\tilde{R} = \mathbb{E}[(V-\mu)(V-\mu)^T]$, où $\mu = \mathbb{E}[V]$ est le vecteur moyen et H et T en exposant d'un vecteur complexe signifient sa transposée-conjuguée et sa transposée, respectivement. Ce vecteur aléatoire complexe est appelé *circulaire* (*proper* en anglais) si la matrice de pseudo-covariance est nulle [Neeser-93, Picinbono-96].

vecteur moyen $[\Re\mu_C(f), \Im\mu_C(f)]^T$ et comme matrice de covariance diagonale $\begin{bmatrix} \frac{r^2(f)}{2} & 0 \\ 0 & \frac{r^2(f)}{2} \end{bmatrix}$ (Fig. A.1).



$$V_C(f) \sim \mathcal{N}(\mu_C(f), r^2(f)) \qquad V_C(f), \mu_C(f) \in \mathbb{C}$$

$$r^2(f) \in \mathbb{R}$$

$$\begin{bmatrix} \Re V_C(f) \\ \Im V_C(f) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \Re\mu_C(f) \\ \Im\mu_C(f) \end{bmatrix}, \begin{bmatrix} \frac{r^2(f)}{2} & 0 \\ 0 & \frac{r^2(f)}{2} \end{bmatrix} \right)$$

FIG. A.1 – Distribution d'une variable aléatoire gaussienne complexe circulaire $V_C(f) \in \mathbb{C}$.

A.2 Familles exponentielles et statistiques naturelles

Nous rappelons ici les notions des statistiques, des statistiques suffisantes, des familles exponentielles et des statistiques naturelles.

Considérons une famille de densités paramétriques $\{p(\mathcal{X}|\theta)\}_\theta$, où $\mathcal{X}$ et θ notent les données et les paramètres, respectivement.

Définition 1. [Kay-93] *Une fonction quelconque des données $\mathbf{T}(\mathcal{X})$ s'appelle statistique.*

Définition 2. [Kay-93] *Une statistique $\mathbf{T}(\mathcal{X})$ s'appelle statistique suffisante (ou exhaustive), si il existe des fonctions $a(\cdot)$ et $b(\cdot,\cdot)$ telles que*

$$p(\mathcal{X}|\theta) = a(\mathcal{X})b(\mathbf{T}(\mathcal{X}), \theta) \tag{A.3}$$

pour tous $\mathcal{X}$ et θ.

Propriété 1. *Soit $p(\theta)$ une loi a priori sur les paramètres θ. Si $\mathbf{T}(\mathcal{X})$ est une statistique suffisante, alors l'estimation MAP des paramètres θ*

$$\theta^{\text{MAP}} = \arg\max_{\theta'} p(\mathcal{X}|\theta')p(\theta') \tag{A.4}$$

doit être une fonction de $\mathbf{T}(\mathcal{X})$.

En effet, en substituant (A.3) dans (A.4), on peut omettre le facteur $a(\mathcal{X})$, car il n'y a pas d'influence sur l'optimisation des paramètres θ, c'est-à-dire $p(\mathcal{X}|\theta)p(\theta) \propto b(\mathbf{T}(\mathcal{X}), \theta)p(\theta)$. Ainsi, l'estimation θ^{MAP} (A.4) ne dépend plus que de $\mathbf{T}(\mathcal{X})$.

Définition 3. [Dempster-77, McLachlan-97] *La famille des densités paramétriques* $\{p(\mathcal{X}|\theta)\}_\theta$ *est appelée* <u>*famille exponentielle*</u>*, si* $p(\mathcal{X}|\theta)$ *peut s'exprimer sous la forme suivante :*

$$p(\mathcal{X}|\theta) = \exp\left\{\langle g(\theta), \mathbf{T}(\mathcal{X})\rangle + d(\theta) + h(\mathcal{X})\right\}, \tag{A.5}$$

où $d(\theta), h(\mathcal{X}) \in \mathbb{R}$ *sont des fonctions scalaires,* $g(\theta), \mathbf{T}(\mathcal{X}) \in \mathbb{R}^L$ *sont des fonctions vectorielles et* $\langle\cdot,\cdot\rangle$ *dénote le produit scalaire. La fonction* $\mathbf{T}(\mathcal{X})$ *s'appelle* <u>*statistique naturelle*</u> *pour cette famille exponentielle.*

Remarquons que la statistique naturelle est également une statistique suffisante. En effet, pour représenter la densité $p(\mathcal{X}|\theta)$ définie par (A.5) sous la forme (A.3), il suffit de poser $a(\mathcal{X}) \triangleq \exp\{h(\mathcal{X})\}$ et $b(\mathbf{T}(\mathcal{X}), \theta) \triangleq \exp\left\{\langle g(\theta), \mathbf{T}(\mathcal{X})\rangle + d(\theta)\right\}$. Ainsi, la propriété 1 est également vérifiée pour les statistiques naturelles.

A.3 Algorithme EM pour l'estimation MAP

L'algorithme EM (*Expectation - Maximization*) [Dempster-77, McLachlan-97] est un des outils principaux utilisés dans cette thèse. Il est présenté ici pour l'estimation Maximum *A Posteriori* (MAP).

Considérons une famille de densités paramétriques $\{p(\mathcal{X}|\theta)\}_\theta$ et une loi *a priori* sur les paramètres $p(\theta)$. Les données $\mathcal{X}$ sont supposées connues et sont appelées ainsi *données observées*. Le but est de trouver l'estimation MAP des paramètres (A.4).

Supposons de plus qu'il existe d'autres données $\mathcal{Z}$ applées *données complètes* qui ne sont pas observées directement et que les données observées s'expriment de façon unique à partir des données complètes, c'est-à-dire il existe une transformée Ω telle que $\mathcal{X} = \Omega(\mathcal{Z})$. Ainsi, les données complètes $\mathcal{Z}$ sont partiellement observées via $\mathcal{X}$.

Il se trouve que le critère MAP (A.4) est difficile à optimiser pour une raison ou l'autre. Il se trouve aussi que ce critère serait facile à optimiser, si les données complètes $\mathcal{Z}$ étaient observées, c'est-à-dire si $\mathcal{X}$ était remplacé par $\mathcal{Z}$ dans (A.4).

Dans une telle situation, il est très favorable d'utiliser l'algorithme EM. Cet algorithme est une procédure itérative dont chaque itération consiste en deux étapes [Dempster-77] :

$$\textbf{Etape E :} \quad Q(\theta, \theta^{(l)}) = \mathbb{E}_{\mathcal{Z}}\left[\log p(\mathcal{Z}|\theta) \,\middle|\, \mathcal{X}, \theta^{(l)}\right] + \log p(\theta) \tag{A.6}$$

$$\textbf{Etape M :} \quad \theta^{(l+1)} = \arg\max_\theta Q(\theta, \theta^{(l)}) \tag{A.7}$$

où $\theta^{(l)}$ dénote les paramètres estimés à la l-ème itération. L'étape E (*Expectation*) (A.6) consiste à calculer une fonction auxiliaire $Q(\theta, \theta^{(l)})$ et l'étape M (*Maximization*) (A.7) consiste à estimer les nouveaux paramètres maximisant cette fonction.

L'algorithme EM assure la convergence des paramètres θ vers un point stationnaire de la loi *a posteriori* $p(\theta|\mathcal{X}) \propto p(\mathcal{X}|\theta)p(\theta)$ qui possède en général plusieurs maxima locaux. Il est donc important de bien choisir les paramètres initiaux $\theta^{(0)}$ pour éviter la convergence vers des maxima locaux trop éloignés des maxima globaux.

A.3.1 Cas particulier des familles exponentielles

Supposons que la famille des densités paramétriques des données complètes $\{p(\mathcal{Z}|\theta)\}_\theta$ est une famille exponentielle (Déf. 3) avec la statistique naturelle $\mathbf{T}(\mathcal{Z})$. Dans ce cas, l'algorithme EM (A.6), (A.7) peut être réécrit dans une forme qui est plus facile à comprendre et à utiliser [Dempster-77, McLachlan-97] :

$$\begin{aligned} \textbf{Etape E :} \quad \mathbf{T}^{(l)}(\mathcal{Z}) &= \mathbb{E}_{\mathcal{Z}}\left[\mathbf{T}(\mathcal{Z}) \,\middle|\, \mathcal{X}, \theta^{(l)}\right] && \text{(A.8)} \\ \textbf{Etape M :} \quad \theta^{(l+1)} &= \mathbf{f}\left(\mathbf{T}^{(l)}(\mathcal{Z})\right) && \text{(A.9)} \end{aligned}$$

où la fonction $\mathbf{f}(\mathbf{T}(\mathcal{Z}))$ est définie comme la solution du critère MAP des données complètes, qui est défini comme suit :

$$\mathbf{f}(\mathbf{T}(\mathcal{Z})) \triangleq \arg\max_{\theta'} p(\mathcal{Z}|\theta')p(\theta') \tag{A.10}$$

Rappelons qu'une telle fonction dépendant de $\mathbf{T}(\mathcal{Z})$ existe selon la propriété 1, car $\mathbf{T}(\mathcal{Z})$ est une statistique naturelle et donc suffisante.

Annexe B

Démonstration de certains résultats

Les démonstrations de certains résultats sont présentées dans cette annexe.

B.1 Familles exponentielles des MMG

Montrons que pour un MMG spectral λ, défini selon (2.15), la famille des densités $\{p(S, q|\lambda)\}_\lambda$ est une famille exponentielle et que la statistique définie par les équations (7.6), (7.7) et (7.8) est une statistique naturelle pour cette famille. Pour alléger les notations, nous omettons ici l'indice k de source.

En utilisant la loi de Bayes, la vraisemblance $p(S, q|\lambda)$ peut être décomposée comme suit :

$$p(S, q|\lambda) = p(S|q, \lambda)p(q|\lambda) \tag{B.1}$$

En utilisant la propriété de l'indépendance des observations $S(t)$ et des états $q(t)$, les logarithmes des vraisemblances de la partie droite de (B.1) peuvent être représentés sous la forme suivante :

$$\log p(S|q, \lambda) = \sum_t \sum_i \delta(q(t), i) \log p(S(t)|q(t) = i, \lambda) \tag{B.2}$$

$$\log p(q|\lambda) = \sum_t \sum_i \delta(q(t), i) \log P(q(t) = i|\lambda) \tag{B.3}$$

Par définition du MMG spectral (2.15), conditionnellement à l'état i, le spectre à court terme $S(t)$ est un vecteur aléatoire gaussien complexe circulaire centré avec comme matrice de covariance Σ_i. Ainsi, la vraisemblance $p(S(t)|q(t) = i, \lambda)$ est égale à $N_C(S(t); \bar{0}, \Sigma_i)$ qui se calcule selon (A.2). Nous avons donc :

$$\begin{aligned}
\log p(S(t)|q(t)=i,\lambda) &= -\sum_f \left[\frac{|S(t,f)|^2}{\sigma_i^2(f)} + \log\left\{\pi\sigma_i^2(f)\right\} \right] && \text{(B.4)}\\
\log P(q(t)=i|\lambda) &= \log \omega_i && \text{(B.5)}
\end{aligned}$$

En substituant les équations (B.4) et (B.5) dans (B.2) et (B.3) qui sont substituées à leurs tour dans (B.1), après quelques développements on obtient :

$$\log p(S,q|\lambda) = \sum_i \left(\left[\log\omega_i - \sum_f \log\left\{\pi\sigma_i^2(f)\right\} \right] \mathbf{t}_i^0 - \sum_f \frac{\mathbf{t}_i^2(f)}{\sigma_i^2(f)} \right) = \langle g(\lambda), \mathbf{T}(S,q)\rangle \quad \text{(B.6)}$$

où $\mathbf{T}(S,q)$, $\mathbf{t}_i^0$ et $\mathbf{t}_i^2(f)$ sont définis selon (7.6), (7.7) et (7.8), et $g(\lambda)$ est une fonction vectorielle. Ainsi, la densité $p(S,q|\lambda) = \exp\{\langle g(\lambda), \mathbf{T}(S,q)\rangle\}$ est représentée sous la forme (A.5), la famille $\{p(S,q|\lambda)\}_\lambda$ est donc une famille exponentielle avec $\mathbf{T}(S,q)$ comme statistique naturelle.

B.2 Calcul des espérances conditionnelles des statistiques naturelles des MMG

Dans cette section, nous présentons quelques éléments de la démonstration des formules (7.9), (7.10), (7.11) et (7.12) impliquées dans l'algorithme 4 de calcul des espérances conditionnelles (7.3) des statistiques naturelles des MMG. Comme toutes les espérances (7.3) sont conditionnellement à X, $\lambda_1^{(l)}$ et $\lambda_2^{(l)}$, nous utilisons une nouvelle notation $\xi^{(l)} \triangleq \{X, \lambda_1^{(l)}, \lambda_2^{(l)}\}$ pour que les développements soient moins encombrants.

Les lignes suivantes contiennent la démonstration de l'équation (7.11) :

$$\begin{aligned}
\mathbf{t}_{1,i}^{0,(l)} &= \mathbb{E}_{S_1,q_1}\left[\sum_t \delta(q_1(t),i) \middle| \xi^{(l)} \right] = \sum_t \mathbb{E}_{q_1}\left[\delta(q_1(t),i) \middle| \xi^{(l)} \right]\\
&= \sum_t \sum_j \mathbb{E}_{q_1,q_2}\left[\delta(q_1(t),i)\delta(q_2(t),j) \middle| \xi^{(l)} \right]\\
&= \sum_t \sum_j P\left(q_1(t)=i, q_2(t)=j \middle| \xi^{(l)} \right) \stackrel{(7.9)}{=} \sum_t \sum_j \gamma_{i,j}^{(l)}(t) && \text{(B.7)}
\end{aligned}$$

L'équation (7.12) peut être démontrée de manière analogue :

$$\begin{aligned}
\mathbf{t}_{1,i}^{2,(l)}(f) &= \mathbb{E}_{S_1,q_1}\left[\sum_t |S_1(t,f)|^2\delta(q_1(t),i)\left|\xi^{(l)}\right.\right] = \sum_t \mathbb{E}_{S_1,q_1}\left[|S_1(t,f)|^2\delta(q_1(t),i)\left|\xi^{(l)}\right.\right] \\
&= \sum_t\sum_j \mathbb{E}_{S_1,q_1,q_2}\left[|S_1(t,f)|^2\delta(q_1(t),i)\delta(q_2(t),j)\left|\xi^{(l)}\right.\right] \\
&= \sum_t\sum_j \mathbb{E}_{S_1}\left[|S_1(t,f)|^2\left|q_1(t)=i,q_2(t)=j,\xi^{(l)}\right.\right] P\left(q_1(t)=i,q_2(t)=j\left|\xi^{(l)}\right.\right) \\
&\overset{(7.9),(7.10)}{=} \sum_t\sum_j \left\langle|S_1(t,f)|^2\right\rangle_{i,j}^{(l)}\gamma_{i,j}^{(l)}(t) \qquad \text{(B.8)}
\end{aligned}$$

Enfin, il faut remarquer que la formule (7.9) est analogue à la formule (2.21) et l'expression pour l'espérance de $|S_1(t,f)|^2$ conditionnellement à la paire d'états (i,j) (7.10) peut être trouvée par exemple dans l'article de Rose *et al.* [Rose-94].

B.3 Formules de réestimation pour l'adaptation des filtres et des gains de DSP

Ici, nous présentons des éléments de démonstrations des formules de réestimation pour l'adaptation d'un filtre (Sec. 9.5.1), des gains de DSP (Sec. 9.5.2) et pour l'adaptation conjointe de ces paramètres (Sec. 9.5.3).

Pour obtenir toutes ces formules de réestimation à l'aide de l'algorithme EM d'adaptation contrainte (8.2), (8.3), il suffit de résoudre les critères MAP des données complètes (8.4) en fonction des statistiques naturelles $\mathbf{T}_k(S_k,q_k)$.

B.3.1 Adaptation d'un filtre

Dans le cas du critère d'adaptation d'un filtre (9.12), les critères MAP (8.4) deviennent (éventuellement, il peut n'y avoir qu'un seul critère car seulement un modèle est adapté) :

$$\mathcal{H}_v^* = \arg\max_{\mathcal{H}_v'} p(S_v,q_v|\lambda_v' = \mathcal{H}_v'\Lambda_v) \qquad \text{(B.9)}$$

En substituant $\lambda_v' = \mathcal{H}_v'\Lambda_v$ dans la formule (B.6) et en annulant la dérivée par rapport au filtre $\mathcal{H}_v'$, on peut montrer que $\mathcal{H}_v^*(f) = \frac{1}{T}\sum_i \frac{\mathbf{t}_{v,i}^2(f)}{r_{v,i}^2(f)}$. Ensuite, en remplaçant les statistiques $\mathbf{t}_{v,i}^2(f)$ par leurs espérances conditionnelles (7.12), on obtient la formule de réestimation (9.13).

B.3.2 Adaptation des gains de DSP

La formule de réestimation des gains de DSP (9.16) se démontre en suivant exactement le même raisonnement que dans la section précédente. Il suffit de remplacer $\lambda'_v = \mathcal{H}'_v \Lambda_v$ par $\lambda'_v = g'_v \bullet \Lambda_v$.

B.3.3 Adaptation conjointe des filtres et des gains de DSP

Pour l'optimisation du critère (9.17) avec l'algorithme EM (8.2), (8.3) il faut résoudre les deux critères MAP des données complètes (8.4), car les deux modèles Λ_v et $\tilde{\lambda}_m$ sont adaptés. Puisque ces critères sont identiques nous considérons celui du modèle de voix :

$$(\mathcal{H}^*_v, g^*_v) = \arg \max_{\mathcal{H}'_v, g'_v} p(S_v, q_v | \lambda'_v = g'_v \bullet \mathcal{H}'_v \Lambda_v) \tag{B.10}$$

Si on essaye de faire les mêmes développements comme dans la section B.3.1, c'est-à-dire de substituer $\lambda'_v = g'_v \bullet \mathcal{H}'_v \Lambda_v$ dans la formule (B.6) et d'annuler les dérivées par rapport à $\mathcal{H}'_v$ et g'_v, on trouve que la solution pour le filtre $\mathcal{H}^*_v$ s'exprime en utilisant la solution pour les gains de DSP g^*_v et vise versa, c'est-à-dire

$$\mathcal{H}^*_v(f) = \frac{1}{T} \sum_{i=1}^{Q_v} \frac{\mathbf{t}^2_{v,i}(f)}{g^*_{v,i} r^2_{v,i}(f)}, \tag{B.11}$$

$$g^*_{v,i} = \frac{1}{F \cdot \mathbf{t}^0_{v,i}} \sum_{f=1}^{F} \frac{\mathbf{t}^2_{v,i}(f)}{\mathcal{H}^*_v(f) r^2_{v,i}(f)} \tag{B.12}$$

Ainsi, on décide de chercher la solution en alternant entre ces deux expressions. Autrement dit, $\mathcal{H}_v$ est recalculé selon (B.11) ayant g_v fixé, puis g_v est recalculé selon (B.12) ayant $\mathcal{H}_v$ fixé, et ainsi de suite, d'où les formules (9.18), (9.19), (9.20) et (9.21) de l'algorithme 5.

On se demande si cette procédure d'alternance entre (B.11) et (B.12) converge vers la solution du critère (B.10). Tous ce qu'on peut affirmer est que les règles de mise à jour (B.11) et (B.12) ne font pas décroître la vraisemblance $p(S_v, q_v | \lambda'_v = g'_v \bullet \mathcal{H}'_v \Lambda_v)$ et que cette procédure converge vers un point stationnaire de cette vraisemblance dans l'espace des paramètres $\{\mathcal{H}_v, g_v\}$. Par contre, la vraisemblance $p(S_v, q_v | \lambda'_v = g'_v \bullet \mathcal{H}'_v \Lambda_v)$ peut posséder en général plusieurs maxima. Il est donc possible que cette procédure converge vers un maximum local.

Bibliographie

[Attias-03] H. Attias. New EM algorithms for source separation and deconvolution. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'03)*, volume 5, pages 297–300, 2003.

[Ayewah-04] N. Ayewah and P.-M. Seidel. Fused models for noise reduction in speech processing. In *38th Asilomar Conference on Signals, Systems and Computers*, 2004.

[Beierholm-04] T. Beierholm, B. D. Pedersen, and O. Winther. Low complexity bayesian single channel source separation. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'04)*, volume 5, pages 529–532, 2004.

[Ben-04] M. Ben. *Approches robustes pour la vérification automatique du locuteur par normalisation et adaptation hiérarchique.* PhD thesis, Université de Rennes 1, IRISA, Novembre 2004.

[Benaroya-03] L. Benaroya. *Séparation de plusieurs sources sonores avec un seul microphone.* PhD thesis, Université de Rennes 1, 2003.

[Benaroya-03a] L. Benaroya and F. Bimbot. Wiener based source separation with HMM/GMM using a single sensor. In *Intl. Conf. on Indep. Component Analysis and Blind Source Separation (ICA'03)*, pages 957–961, Nara, Japan, April 2003.

[Benaroya-06] L. Benaroya, F. Bimbot, and R. Gribonval. Audio source separation with a single sensor. *IEEE Trans. Audio, Speech and Language Proc.*, 14(1) :191–199, january 2006.

[Berenzweig-01] A. Berenzweig and D. P. W. Ellis. Locating singing voice segments within music signals. In *IEEE Worksh. on Apps. of Signal Processing to Audio and Acoustics (WASPAA'01)*, pages 119 – 122, 2001.

[Berouti-79] M. Berouti, R. Schwartz, and J. Makhoul. Enhancement of speech corrupted by additive noise. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'79)*, pages 208–211, 1979.

[Bofill-01] P. Bofill and M. Zibulevsky. Underdetermined blind sourc separation using sparse representations. *Signal Processing*, 81 :2353–2362, 2001.

[Boll-79] S. F. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Trans. Acoustics, Speech and Signal Processing*, 2(27) :112–120, April 1979.

[Brown-94] G. J. Brown and M. P. Cooke. Computational auditory scene analysis. *Computer Speech and Language*, 8 :297–336, 1994.

[Burshtein-99] D. Burshtein and S. Gannot. Speech enhancement using a mixture-maximum model. In *European Conf. on Speech Communication and Technology (EuroSpeech'99)*, volume 6, pages 2591–2594, Budapest, Hungary, Sep 1999.

[Cappe-93] O. Cappé. *Techniques de réduction de bruit pour la restauration d'enregistrements musicaux*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, Paris, Septembre 1993.

[Cardoso-98] J.-F. Cardoso. Blind signal separation : Statistical principles. *Proc. IEEE*, 86 :2009–2025, Oct. 1998.

[Casey-00] M. Casey and A. Westner. Separation of mixed audio sources by independent subspace analysis. In *International Computer Music Conference (ICMC'00)*, 2000.

[Chen-00] K.-T. Chen, W.-W. Liau, H.-M. Wang, and L.-S. Lee. Fast speaker adaptation using eigenspace-based maximum likelihood linear regression. In *Intl. Conf. on Spoken Language Proc. (ICSLP'00)*, pages 742–745, Beijing, China, Oct 2000.

[Chen-02] C.-P. Chen, J. Bilmes, and K. Kirchhoff. Low-resource noise-robust feature post-processing on aurora 2.0. In *Intl. Conf. on Spoken Language Proc. (ICSLP'02)*, pages 2445–2448, 2002.

[Cooke-01] M. P. Cooke, P. D. Green, L. Josifovski, and A. Vizinho. Robust automatic speech recognition with missing and unreliable acoustic data. *Speech Communication*, pages 267–285, 2001.

[Cooke-93] M. P. Cooke, G. J. Brown, M. D. Crawford, and P. Green. Computational auditory scene analysis : Listening to several things at once. *Endeavour*, 17(4) :186–190, 1993.

[Curtis-78] R. A. Curtis and R. J. Niederjohn. An investigation of several frequency domain processing methods for enhancing the intelligibility of speech in wideband random noise. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'78)*, pages 602–605, 1978.

[Deller-99] J.R. Deller, Jr.J.H.L. Hansen, and J.G. Proakis. *Discrete-Time Processing of Speech Signals*. Wiley-IEEE Press, Sept. 1999.

[Dempster-77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39 :1–38, 1977.

[Ellis-06] D. Ellis and R. Weiss. Model-based monaural source separation using a vector-quantized phase-vocoder representation. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'06)*, volume 5, pages 957–960, Toulouse, France, May 2006.

[Ellis-96] D. P. W. Ellis. *Prediction-driven computational auditory scene analysis*. PhD thesis, Department of Electrical Engineering and Computer Science, M.I.T, 1996.

[Ephraim-85] Y. Ephraim and D. Malah. Speech enhancement using a minimum mean square error log-spectral amplitude estimator. In *IEEE Trans. on Acoust., Speech, and Sig. Proc.*, volume ASSP-33, pages 443–445, Apr 1985.

[Ephraim-92] Y. Ephraim. A bayesian estimation approach for speech enhancement using hidden Markov models. *IEEE Trans. Signal Processing*, SP-40 :725–735, April 1992.

[Ephraim-92a] Y. Ephraim. A bayesian estimation approach for speech enhancement using hidden markov models. *IEEE Trans. Signal Processing*, SP-40 :725–735, April 1992.

[Fessler-94] J. A. Fessler and A. O. Hero. Space-alternating generalized expectation-maximization algorithm. *IEEE Trans. on Signal Processing*, 42(10) :2664 – 2677, Oct. 1994.

[Gales-96] M. Gales, D. Pye, and P. Woodland. Variance compensation within the MLLR framework for robust speech recognition and speaker adaptation. In *Intl. Conf. on Spoken Language Proc. (ICSLP'96)*, volume 3, pages 1832–1835, Philadelphia, PA, 1996.

[Gauvain-94] J. Gauvain and C. Lee. Maximum a-posteriori estimation for multivariate gaussian mixture observations of markov chains. *IEEE Trans. on Speech and Audio Proc.*, 2(2) :291 – 298, April 1994.

[Ghahramani-93] Z. Ghahramani and M. I. Jordan. Supervised learning from incomplete data via an em approach. In *Neural Info. Processing Systems (NIPS'93)*, pages 120–127, 1993.

[Ghahramani-97] Z. Ghahramani and M. Jordan. Factorial hidden Markov models. *Machine Learning*, 29 :245–273, 1997.

[Gribonval-03] R. Gribonval, L. Benaroya, E. Vincent, and C. Févotte. Proposals for performance measurement in source separation. In *Intl. Conf. on Indep. Component Analysis and Blind Source Separation (ICA'03)*, pages 763–768, April 2003.

[Gribonval-03a] R. Gribonval. Piecewise linear source separation. In *SPIE-03 "Wavelets : Applications in Signal and Image Processing"*, volume 5207, pages 297–310, San Diego, California, USA, August 2003.

[Helen-05] M. Helén and T. Virtanen. Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine. In *European Signal Processing Conference (EUSIPCO'05)*, 2005.

[Hershey-01] J. Hershey and M. Casey. Audio-visual sound separation via hidden Markov models. In *Advances in Neural Information Processing Systems (NIPS'01)*, 2001.

[Hoyer-02] P. O. Hoyer. Non-negative sparse coding. In *Neural Networks for Signal Processing XII (Proc. IEEE Workshop on Neural Networks for Signal Processing)*, Martigny, Switzerland, 2002.

[Hu-03] G. Hu and D.L. Wang. Monaural speech separation. In *Neural Info. Processing Systems (NIPS'02)*, 2003.

[ISMIR-04] ISMIR 2004, Melody Extraction Contest. `http://ismir2004.ismir.net/melody_contest/results.html`.

[Jang-03] G.-J. Jang and T.-W. Lee. A maximum likelihood approach to single-channel source separation. *Journal of Machine Learning Research*, (4) :1365–1392, 2003.

[Jordan-98] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Learning in Graphical Models*, 37(2) :183–233, 1999.

[Jutten-03] C. Jutten and R. Gribonval. L'analyse en composantes indépendantes : un outil puissant pour le traitement de l'information. In *GRETSI'05 Symposium on Signal and Image Processing*, ENST, Paris, France, September 2003.

[Kay-93] S. M. Kay. *Fundamentals of Statistical Signal Processing, Estimation Theory.* Prentice Hall, 1993.

[Kim-02] Y. E. Kim and B. Whitman. Singer identification in popular music recordings using voice coding features. In *Intl. Sympos. on Music Information Retrieval (ISMIR'02)*, pages 164–169, Oct 2002.

[Kim-06] M. Kim and S. Choi. Monaural music source separation : Nonnegativity, sparseness, and shift-invariance. In *Intl. Conf. on Indep. Component Analysis and Blind Source Separation (ICA'06)*, 2006.

[Krishnamurthy-93] V. Krishnamurthy and J. B. Moore. On-line estimation of hidden Markov model parameters based on the Kullback–Leibler Information measure. *IEEE Trans. Signal Process.*, 41 :2557–2573, 1993.

[Kristjansson-04] T. Kristjansson, H. Attias, and J. Hershey. Single microphone source separation using high resolution signal reconstruction. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'04)*, volume 2, pages 817–820, 2004.

[Lee-Huo-00] C.-H. Lee and Q. Huo. On adaptive decision rules and decision parameter adaptation for automatic speech recognition. *Proceedings of the IEEE*, 88(8) :1241–1269, 2000.

[Leggetter-95] C. Leggetter and P. Woodland. Flexible speaker adaptation using maximum likelihood linear regression. In *ARPA Spoken Language Technology Workshop*, pages 104–109, 1995.

[Li-06] Y. Li and D. L. Wang. Singing voice separation from monaural recordings. In *ISMIR'06*, 2006.

[Martin-97] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki. The DET curve in assessment of detection task performance. In *European Conf. on Speech Communication and Technology (EuroSpeech'97)*, pages 1895–1898, 1997.

[McLachlan-97] G. McLachlan and T. Krishnan. *The EM Algorithm and Extensions.* Wiley, New York, USA, 1997.

[McQueen-67] J. McQueen. Some methods for classification and analysis of multivariate observations. In *5th Berkeley Symposium on mathematics, Statistics and Probability*, pages 281–298, 1967.

[Moreno-96] P. J. Moreno, B. Raj, and R. M. Stern. A vector taylor series approach for environment-independent speech recognition. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'96)*, volume 2, 1996.

[Murphy-02] K. P. Murphy. *Dynamic Bayesian Networks : Representation, Inference and Learning.* PhD thesis, UC Berkeley, July 2002.

[Nadas-89] A. Nádas, D. Nahamoo, and M. A. Picheny. Speech recognition using noise-adaptive prototype. In *IEEE Trans. on Speech and Audio Proc.*, pages 1495–1505, 1989.

[Neeser-93] F. D. Neeser and J. L. Massey. Proper complex random processes with applications to information theory. *IEEE Trans. Inform. Theory*, 39(4) :1293–1302, July 1993.

[Nwe-04] T. L. Nwe, A. Shenoy, and Y. Wang. Singing voice detection in popular music. In *ACM Multimedia Conference*, pages 324 – 327, New York, NY, USA, October 2004.

[Ozerov-03] A. Ozerov. Représentations robustes pour la reconnaissance automatique de la parole. Master's thesis, DESS CSA, Université de Bordeaux 1, 2003.

[Ozerov-05a] A. Ozerov, R. Gribonval, P. Philippe, and F. Bimbot. Séparation voix / musique à partir d'enregistrements mono quelques remarques sur le choix et l'adaptation des modèles. In *GRETSI'05 Symposium on Signal and Image Processing*, Louvain-la-Neuve, Belgique, Sept. 2005.

[Ozerov-05b] A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot. One microphone singing voice separation using source-adapted models. In *IEEE Worksh. on Apps. of Signal Processing to Audio and Acoustics (WASPAA'05)*, pages 90 – 93, Mohonk, NY, Oct. 2005.

[Ozerov-www] Alexey Ozerov's singing voice separation demos. `http://www.irisa.fr/metiss/ozerov/demos.html`.

[Peeters-99] Geoffroy Peeters and Xavier Rodet. SINOLA : A new analysis/synthesis method using spectrum peak shape distortion, phase and reassigned spectrum.

[Picinbono-96] B. Picinbono. Second-order complex random vectors and normal distributions. *IEEE Trans. Signal Processing*, 44(10) :2637–2640, October 1996.

[Pontoppidan-03] N. H. Pontoppidan and M. Dyrholm. Fast monaural separation of speech. In *Audio Engineering Society (AES) 23rd Conf. on Signal Proc. in Audio Recording and Reproduction*, 2003.

[Press-92] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C : The Art of Scientific Computing.* Cambridge University Press, 2 edition, October 1992.

[Rabiner-89] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2) :257– 286, 1989.

[Reddy-04] A. M. Reddy and B. Raj. Soft mask estimation for single channel speaker separation. In *ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing (SAPA-04)*, Jeju, Korea, October 2004.

[Reyes-Gomez-04b] M. Reyes-Gomez, D. Ellis, and N. Jojic. Multiband audio modeling for single-channel acoustic source separation. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'04)*, volume 5, pages 641–644, May 2004.

[Reynolds-00] A. Reynolds, T.F. Quatieri, and R.B. Dunn. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, (10) :19 – 41, 2000.

[Rickard-02] S. Rickard and O. Yilmaz. On the approximate W-disjoint orthogonality of speech. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'02)*, volume 3, pages 3049–3052, Florida, USA, May 2002.

[Rivet-04] B. Rivet, L. Girin, C. Jutten, and J.-L. Schwartz. Using audiovisual speech processing to improve the robustness of the separation of convolutive speech mixtures. In *IEEE Int. Workshop on Multimedia Signal Processing (MMSP'04)*, pages 47–50, Sienna, Italy, October 2004.

[Rose-94] R. C. Rose, E. M. Hofstetter, and D. A. Reynolds. Integrated models of signal and background with application to speaker identification in noise. *IEEE Trans. Speech and Audio*, 2(2) :245–257, April 1994.

[Roweis-01] S. T. Roweis. One microphone source separation. In *Advances in Neural Information Processing Systems*, volume 13, pages 793–799. MIT Press, 2001.

[Ryynanen-05] M. P. Ryynänen and A. Klapuri. Polyphonic music transcription using note event modeling. In *IEEE Worksh. on Apps. of Signal Processing to Audio and Acoustics (WASPAA'05)*, Mohonk, NY, Oct. 2005.

[Schmidt-06] M. N. Schmidt and M. Mørup. Nonnegative matrix factor 2-d deconvolution for blind single channel source separation. In *Intl. Conf. on Indep. Component Analysis and Blind Source Separation (ICA'06)*, 2006.

[Shinoda-97] K. Shinoda and C.-H. Lee. Structural MAP speaker adaptation using hierarchical priors. In *IEEE Workshop on Speech Recognition and Understanding*, pages 381–388, Santa Barbara, Dec 1997.

[Smaragdis-04] P. Smaragdis. Non-negative matrix factor deconvolution ; extraction of multiple sound sources from monophonic inputs. In *Intl. Conf. on Indep. Component Analysis and Blind Source Separation (ICA'04)*, pages 494–499, 2004.

[Tsai-04] W.-H. Tsai, D. Rogers, and H.-M. Wang. Blind clustering of popular music recordings based on singer voice characteristics. *Computer Music Journal*, 28(3) :68 — 78, 2004.

[Tsai-04a] W. H. Tsai and H. M. Wang. Automatic detection and tracking of target singer in multi-singer music recordings. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'04)*, volume 4, pages 221 – 224, Montreal, Canada, 2004.

[Valin-04] J.-M. Valin, J. Rouat, and F. Michaud. Microphone array post-filter for separation of simultaneous non-stationary sources. In *IEEE Intl. Conf. on Acoustics, Speech and Signal Proc. (ICASSP'04)*, 2004.

[Vembu-05] S. Vembu and S. Baumann. Separation of vocals from polyphonic audio recordings. In *Intl. Sympos. on Music Information Retrieval (ISMIR'05)*, pages 337–344, 2005.

[Vergin-99] R. Vergin, D. O'Shaughnessy, and A. Farhat. Generalized mel frequency cepstral coefficients for large-vocabulary speaker-independent continuous-speech recognition. *IEEE Trans. on Speech and Audio Proc.*, 7(5) :525 – 532, Sep 1999.

[Vincent-01] E. Vincent. Séparation de signaux audio : principes statistiques de l'analyse en composantes indépendantes et applications au signal monophonique. Master's thesis, DEA ATIAM, IRCAM, Paris, France, 2001.

[Vincent-03] E. Vincent, C. Févotte, R. Gribonval, and al. A tentative typology of audio source separation tasks. In *Intl. Conf. on Indep. Component Analysis and Blind Source Separation (ICA'03)*, Nara, Japan, April 2003.

[Vincent-04] E. Vincent. *Modèles d'instruments pour la séparation de sources et la transcription d'enregistrements musicaux*. PhD thesis, Université Paris VI, Paris, France, 2004.

[Vincent-04a] E. Vincent and X. Rodet. Underdetermined source separation with structured source priors. In *Intl. Conf. on Indep. Component Analysis and Blind Source Separation (ICA'04)*, pages 327–334, Granada, Spain, September 2004.

[Vincent-05] E. Vincent, M. G. Jafari, S. A. Abdallah, M D. Plumbley, and M. E. Davies. Blind audio source separation. Technical Report C4DM-TR-05-01, Centre for Digital Music, Queen Mary University of London, November 2005.

[Vincent-05a] E. Vincent, C. Févotte, and R. Gribonval. Performance measurement in blind audio source separation. *IEEE Trans. Speech and Audio Processing*, 14(4) :1462–1469, 2005.

[Vincent-05b] E. Vincent and R. Gribonval. Construction d'estimateurs oracles pour la séparation de sources. In *GRETSI'05 Symposium on Signal and Image Processing*, Louvain-la-Neuve, Belgium, 2005.

[Wang-05] B. Wang and M. D. Plumbley. Musical audio stream separation by non-negative matrix factorization. In *UK Digital Music Research Network (DMRN) Summer Conf.*, 2005.

[Weiss-06] R. J. Weiss and D. P. W. Ellis. Estimating single-channel source separation masks : Relevance vector machine classifiers vs. pitch-based masking. In *(SAPA'06) ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing*, 2006.

[Wiener-49] N. Wiener. *Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications.* New York, Wiley, 1949.

Résumé

La séparation de sources avec un seul capteur est un problème très récent, qui attire de plus en plus d'attention dans le monde scientifique. Cependant, il est loin d'être résolu et, même plus, il ne peut pas être résolu en toute généralité. La difficulté principale est que, ce problème étant extrêmement sous déterminé, il faut disposer de fortes connaissances sur les sources pour pouvoir les séparer. Pour une grande partie des méthodes de séparation, ces connaissances sont représentées par des modèles statistiques des sources, notamment par des Modèles de Mélange de Gaussiennes (MMG), qui sont appris auparavant à partir d'exemples.

L'objet de cette thèse est d'étudier les méthodes de séparation basées sur des modèles statistiques en général, puis de les appliquer à un problème concret, tel que la séparation de la voix par rapport à la musique dans des enregistrements monophoniques de chansons. Apporter des solutions à ce problème, qui est assez difficile et peu étudié pour l'instant, peut être très utile pour faciliter l'analyse du contenu des chansons, par exemple dans le contexte de l'indexation audio.

Les méthodes de séparation existantes donnent de bonnes performances à condition que les caractéristiques des modèles statistiques utilisés soient proches de celles des sources à séparer. Malheureusement, il n'est pas toujours possible de construire et d'utiliser en pratique de tels modèles, à cause de l'insuffisance des exemples d'apprentissage représentatifs et des ressources calculatoires.

Pour remédier à ce problème, il est proposé dans cette thèse d'adapter *a posteriori* les modèles aux sources à séparer. Ainsi, un formalisme général d'adaptation est développé. En s'inspirant de techniques similaires utilisées en reconnaissance de la parole, ce formalisme est introduit sous la forme d'un critère d'adaptation Maximum *A Posteriori* (MAP). De plus, il est montré comment optimiser ce critère à l'aide de l'algorithme EM à différents niveaux de généralité.

Ce formalisme d'adaptation est ensuite appliqué dans certaines formes particulières pour la séparation voix / musique. Les résultats obtenus montrent que pour cette tâche, l'utilisation des modèles adaptés permet d'augmenter significativement (au moins de 5 dB) les performances de séparation par rapport aux modèles non adaptés. Par ailleurs, il est observé que la séparation de la voix chantée facilite l'estimation de sa fréquence fondamentale (pitch), et que l'adaptation des modèles ne fait qu'améliorer ce résultat.

Mots clés : séparation de sources avec un seul capteur - modèles statistiques - adaptation bayésienne - maximum a posteriori - réseaux bayésiens - expectation maximization - modèles de mélange de gaussiennes - filtrage de Wiener adaptatif